Detecting hidden chain-of-thought in large language models with linguistic, behavioral, and mechanistic indicators

Armaan Singh [1] **Ryan Trinh Le** [1] **Jasmine Kaur** [1] **Edward Lue Chee Lip** **Kiran Nijjer** [1] **Adnan Ahmed** [2] **Vasu Sharma** [3]

Abstract

Large language models often answer complex reasoning questions correctly without revealing intermediate steps, raising the question of whether they perform latent reasoning or pattern completion. Existing detection methods rely on surface-level linguistic cues that prior work has shown can be unfaithful. We propose the Hidden CoT Detection Score (HCDS), a comparative behavioral and mechanistic signal that quantifies whether a model's neutral-prompt behavior aligns more closely with its explicit-CoT or with its explicit no-CoT behavior. On Qwen3-4B Instruct, HCDS is robustly positive on both GSM8K and StrategyQA ($+1.87$ to $+2.38$, $p < 10^{-9}$) — our cleanest hidden-CoT detection result. On Qwen3-4B Thinking, HCDS is also positive ($+0.33$ to $+0.60$, $p < 0.05$) but the result is better read as evidence of prompt-invariance of reasoning traces: under explicit no-CoT directives, Thinking emits a mean of $\sim$575 reasoning tokens vs Instruct's $\sim$24, so the no-CoT pole is not a clean answer-only baseline for that model. Anchor-suppression analysis further shows that the Thinking model distributes its causal load across many reasoning steps, suggesting that hidden chain-of-thought in reasoning-tuned models is more deeply integrated, more prompt-invariant, and more diffuse than in instruction-tuned ones.

1. Introduction

Explicit Chain of Thought (CoT) prompting improves LLM performance and exposes interpretable reasoning steps (Wei et al., 2022), but it is unclear whether models engage in latent or implicit CoT reasoning when users do not request step-by-step output. Current methods identify implicit CoT only through surface-level cues and behavioral patterns. These cues are correlational, not causal. Misidentifying such patterns as reasoning weakens interpretability and erodes user trust.

Models can plausibly produce forms of reasoning text that do not contribute to the final output (Turpin et al., 2023). Distinguishing the two necessitates understanding whether the visible steps causally created the answer or simply accompany it. Output-only analysis cannot achieve this, because genuine step-by-step reasoning and post-hoc rationalization of an already-determined answer can yield the same surface text; the evidence that would separate them lies in the internal computation, not the output.

We reframe latent reasoning detection from a correlational analysis of model outputs into a behavioral identification problem. We propose the Hidden CoT Detection Score (HCDS), which quantifies the extent to which a model's neutral-prompt behavior resembles its explicit reasoning patterns. HCDS is computed by systematically comparing model behavior across explicit CoT, explicit no-CoT, and neutral prompts along three feature families: linguistic signals comprising mean token entropy and entropy slope; behavioral signals comprising latency per output token, paraphrase consistency, and perturbation sensitivity; and a mechanistic signal measuring intervention sensitivity from internal activations and attention patterns in open-weight models. Output length is recorded only for diagnostic and length-matched analyses and is not included in the HCDS feature vector. The mechanistic signal additionally supports targeted interventions that selectively disrupt internal dependencies, allowing us to evaluate whether specific steps causally contribute to the model's output rather than merely accompany it.

We evaluate HCDS on Qwen3-4B (Instruct and Thinking variants) using GSM8K and StrategyQA, comparing each model's behavior across the three prompt conditions. HCDS is significantly positive for both models under neutral prompting on both datasets (GSM8K: +2.254 for Instruct, p = 2.21e-10; +0.479 for Thinking, p = 2.06e-02; StrategyQA: +1.871 for Instruct, p = 3.53e-12; +0.597 for Thinking, p = 2.15e-03), indicating that neutral behavior aligns

[1]Stanford University [2]York University [3]Facebook. Correspondence to: Armaan Singh <armaan30312@gmail.com>.

ICML 2026 Workshop on Hypothesis Testing, Seoul, South Korea, 2026. Copyright 2026 by the author(s).

more closely with explicit CoT than with explicit no-CoT. The Instruct effect is robust across feature-ablation variants and across short, medium, and long questions; the Thinking effect is smaller and length-sensitive, weakening on medium-length questions where reasoning traces approach the generation cap. Anchor-suppression interventions further reveal that the Thinking model's explicit reasoning chains distribute causal load across many steps rather than concentrating in a few, an unexpected mechanistic finding the rest of the paper develops.

2. Related Work

A first line of related work asks whether visible chain-of-thought explanations faithfully reflect the computation that produces a model's answer. Turpin et al. (2023) show that chain-of-thought explanations can be unfaithful, in the sense that models may produce plausible reasoning text that is not causally responsible for their final prediction. More recently, Chen et al. (2025) test faithfulness through controlled prompt-hint interventions and show that reasoning traces do not always transparently reveal what the model used to reach its answer. These works motivate the central premise of our paper: output-only inspection is insufficient for determining whether reasoning is occurring internally. Our contribution differs in scope and objective. Rather than asking whether an explicit rationale is faithful, we ask whether latent reasoning is occurring at all under neutral prompting, and we address this question through a combination of behavioral and mechanistic signals rather than through rationale analysis alone.

A second line of work studies whether latent reasoning exists as a genuine internal phenomenon. Lin et al. (2025) argue that implicit reasoning in Transformers often emerges through shortcut learning and can fail to generalize when problem structure changes, suggesting that correct answers without explicit CoT need not imply robust reasoning. Concurrent work by He et al. (2026) provides mechanistic evidence that latent reasoning corresponds to steerable internal features, showing that reasoning behavior can be activated without explicit chain-of-thought prompting. These papers establish that latent reasoning is both real and difficult to interpret: it can exist as an internal computational mode, but may also be entangled with brittle heuristics. Our work differs by proposing a detection framework for identifying when such latent reasoning is being used at inference time, rather than studying how it emerges during training or how it can be induced through latent steering.

The work most closely related to our mechanistic component is Bogdan et al. (2025), who identify "thought anchors" — reasoning steps within explicit CoT traces that have disproportionate influence on downstream generation. They propose black-box, attention-based, and causal attribution methods to determine which steps matter most within a visible reasoning chain. Our paper builds on this intervention logic but shifts the question from which explicit reasoning steps matter to whether reasoning is happening under neutral prompting at all. In our framework, anchor-based perturbation is not the end goal but one feature family within a broader hidden-CoT detection score. This difference is important: Bogdan et al. analyze the internal structure of explicit CoT, whereas we use related mechanistic tools to detect latent reasoning in the absence of explicit CoT.

3. Background

Our method builds on prior work in Transformer language models, chain-of-thought prompting, reasoning benchmarks, and causal interpretability. Because the goal of this paper is to detect hidden chain-of-thought reasoning rather than to elicit it directly, the relevant background is not only how reasoning prompts improve model performance, but also how internal computations in Transformer models can be analyzed through interventions.

Modern large language models are based on the Transformer architecture introduced by Vaswani et al. (2017). The Transformer replaces recurrence with stacked self-attention and feedforward layers, making token-level contextual interactions, residual-stream representations, and layerwise interventions central objects of analysis. These architectural components are directly relevant to our method because the mechanistic part of our framework studies internal activations and intervention effects within a Transformer model.

The prompting side of our framework is inherited from chain-of-thought prompting. Wei et al. (2022) showed that prompting a language model to produce intermediate reasoning steps can substantially improve performance on multi-step reasoning tasks. In our setting, explicit CoT prompting serves as one anchor condition, while explicit no-CoT prompting serves as a contrasting answer-only condition. Our objective is not to improve reasoning through prompting itself, but to use these prompt conditions as reference points for determining whether a model under a neutral prompt behaves more like an overt reasoner or more like a direct-answer generator.

Our experiments are grounded in two reasoning benchmarks. GSM8K, introduced by Cobbe et al. (2021), is a benchmark of grade-school mathematical word problems designed to test multi-step arithmetic reasoning. StrategyQA, introduced by Geva et al. (2021), is a benchmark of yes/no questions that require implicit multi-hop commonsense reasoning. Together, these datasets allow us to evaluate hidden-CoT detection across both quantitative and qualitative reasoning settings while keeping the tasks sufficiently structured for prompt-matched comparisons.

The mechanistic component of our framework draws on causal intervention methods in interpretability. Meng et al. (2022) showed that targeted manipulation of internal activations can identify and alter computationally important representations in autoregressive Transformer language models. This intervention-based perspective provides the conceptual basis for our anchor-suppression analysis: if perturbing a candidate reasoning-relevant internal state changes the final answer more than a matched control intervention, then that state is likely to play a causal role in the model's solution process.

3.1. Problem Setting

We study hidden chain-of-thought as a comparative detection problem. For a fixed model and question, we observe behavior under three prompt conditions: explicit CoT, explicit no-CoT, and neutral. Let $f_{m,q,p}$ denote the feature vector for model m, question q, and prompt condition p. Our method asks whether the neutral feature vector is closer to the explicit-CoT feature vector or to the explicit-no-CoT feature vector. Positive alignment with explicit CoT is interpreted as evidence that the model relies on latent intermediate reasoning even when that reasoning is not explicitly shown.

4. Method

We formulate hidden chain-of-thought (CoT) detection as a comparative inference problem. Figure 1 summarizes the full workflow. For each model m and question q, we evaluate three prompt conditions: *explicit_cot*, which asks the model to reason step by step; *explicit_no_cot*, which asks for a direct answer without reasoning; and *neutral_strict*, a minimal task-only prompt that gives no instruction about whether reasoning should be shown. Full prompt text is provided in Appendix A. Rather than attempting to recover an internal reasoning trace directly, we ask whether the model's behavior under the neutral prompt is closer to its behavior under explicit CoT or explicit no-CoT. For each (m, q, p) tuple, we construct a feature vector

$$f_{m,q,p} = \big[\ell_{m,q,p},\ \bar{H}_{m,q,p},\ s^{H}_{m,q,p},\ C^{\text{para}}_{m,q,p},\ \Delta A^{\text{pert}}_{m,q,p},\ \Delta A^{\text{mech}}_{m,q,p}\big].$$

where the components denote latency per output token, mean token entropy, entropy slope, paraphrase consistency, perturbation sensitivity, and mechanistic intervention sensitivity.

We compare prompt conditions by Euclidean distance in z-scored feature space. When some features are unavailable for a given question, we compute distances only over the shared observed dimensions $\mathcal{J}_q$:

$$D(a, b) = \sqrt{\sum_{j \in \mathcal{J}_q} (a_j - b_j)^2}.$$

We then define the per-question Hidden CoT Detection Score (HCDS). Intuitively, $\text{HCDS}_q > 0$ means the model's neutral-prompt behavior on question q looks more like its overt-CoT behavior than its overt no-CoT behavior — i.e., the model is behaving as if it is reasoning under the hood. Formally:

$$\begin{aligned}\text{HCDS}_q = D(f_{m,q,\text{neutral}}, f_{m,q,\text{no-cot}}) \\ - D(f_{m,q,\text{neutral}}, f_{m,q,\text{cot}})\,.\end{aligned} \tag{1}$$

A positive HCDS_q indicates that the model's neutral-prompt behavior is closer to explicit CoT than to explicit no-CoT for that question, which we interpret as evidence of hidden reasoning.

At the model–dataset level, we aggregate per-question scores as

$$\overline{\text{HCDS}} = \frac{1}{N} \sum_{q=1}^{N} \text{HCDS}_q. \tag{2}$$

This framing treats hidden CoT as a functional dependency on intermediate computation rather than a visible linguistic artifact. The behavioral features capture whether neutral prompting exhibits the timing, uncertainty, and consistency profile expected of overt reasoning, while the mechanistic feature tests whether suppressing candidate reasoning *anchors* (the reasoning steps our attention/activation proxies flag as most causally influential, defined operationally in Section 6.2) harms performance more than matched control interventions. Together, these signals allow us to distinguish three regimes: neutral behavior that aligns with explicit CoT, neutral behavior that aligns with explicit no-CoT, and ambiguous cases where the evidence does not clearly support either interpretation.

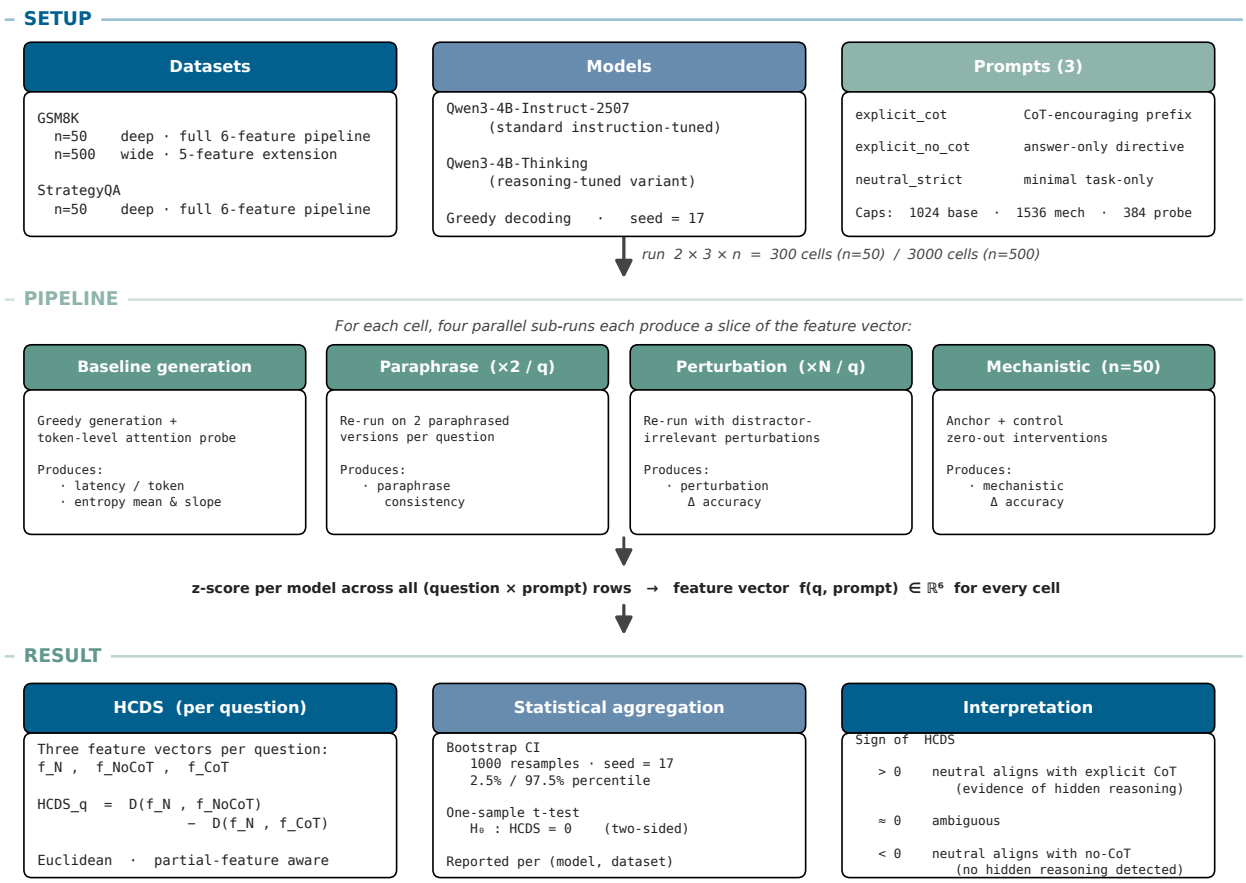

Figure 1. **HCDS methodology pipeline.** *Setup:* two datasets × two models × three prompts. *Pipeline:* for each $(\text{model}, \text{prompt}, \text{question})$ cell, four parallel sub-pipelines (baseline + attention probe, paraphrase, perturbation, mechanistic) produce a six-dimensional feature vector that is z-scored per model. *Result:* HCDS contrasts the neutral prompt's distance to the no-CoT pole vs the CoT pole, with bootstrap CIs and a one-sample two-sided t-test reported per $(\text{model}, \text{dataset})$. Sign of HCDS distinguishes hidden reasoning (> 0), ambiguity (≈ 0), and no hidden reasoning (< 0); observed values appear in Figure 2.

5. Experimental Setup

We evaluate two open-weight models from the same family, Qwen3-4B-Instruct-2507 and Qwen3-4B-Thinking-2507, on GSM8K and StrategyQA. GSM8K tests multi-step arithmetic reasoning, while StrategyQA tests multi-hop commonsense reasoning with binary yes/no answers. We use the same three prompt conditions for both datasets: *explicit_cot*, *explicit_no_cot*, and *neutral_strict* (the names used in our codebase and figures; for brevity we refer to them as explicit CoT, explicit no-CoT, and neutral throughout the prose). All main experiments use fixed decoding settings across conditions and deterministic decoding, so each question contributes one run per prompt condition and the question is the primary statistical unit. Because decoding is deterministic with one run per question, all reported confidence intervals and p-values quantify variation across questions, not sampling variability over stochastic model generations.

For each (m, q, p) tuple, we record answer accuracy, generation time, output length, latency per output token, token-level entropy summaries, paraphrase consistency, perturbation delta accuracy, and mechanistic intervention delta accuracy when available. Answer accuracy and output length are auxiliary diagnostics; they do not enter the HCDS feature vector. The remaining measurements feed the comparative pipeline in Figure 1. Latency per output token is defined as

$$\ell_{m,q,p} = \frac{t^{\text{gen}}_{m,q,p}}{T^{\text{out}}_{m,q,p}},$$

and token entropy is summarized by its mean and slope over the generated continuation:

$$\bar{H}_{m,q,p} = \frac{1}{T^{\text{out}}_{m,q,p}} \sum_{t=1}^{T^{\text{out}}_{m,q,p}} H_t,$$

$$H_t = -\sum_{v} p_t(v) \log p_t(v).$$

We define the three content-level features concretely as follows. *Paraphrase consistency* C^{para} is the fraction of K meaning-preserving rewordings of the question on which the model returns the same answer it gave on the original; the rewordings are generated by a deterministic Qwen3-4B-Instruct paraphraser constrained to preserve every numeric value, the arithmetic structure, the named entities, and the gold answer. *Perturbation sensitivity* $\Delta A^{\text{pert}} = A_{\text{orig}} - A_{\text{pert}}$ is the accuracy drop under a *distractor-irrelevant* perturbation: we insert a single irrelevant clause (e.g., "The neighbor's dog barked seven times that afternoon.") immediately before the final question sentence, leaving every quantity used in the gold reasoning chain untouched, so the correct answer is unchanged and any drop reflects fragility to surface distractors rather than a genuinely harder problem. *Mechanistic sensitivity* $\Delta A^{\text{mech}} = (A_{\text{base}} - A_{\text{anchor}}) - (A_{\text{base}} - A_{\text{control}})$ is the anchor-suppression contrast, i.e. the extra accuracy lost by zeroing out a trace's anchor steps beyond what is lost by zeroing out matched control steps; the anchor, control, and suppression procedures are defined operationally in Section 6.2 and specified in full in Appendix F.

We report HCDS separately for each model and dataset, together with robustness analyses. We interpret HCDS primarily by its sign and statistical support: positive values indicate that neutral behavior is closer to explicit CoT than to explicit no-CoT, while values near zero indicate ambiguity. We do not assign a universal semantic meaning to raw HCDS magnitude, since it depends on the chosen z-scored feature space and feature coverage. Statistical evaluation tests

$$H_0 : \mathbb{E}[\text{HCDS}_q] = 0 \qquad \text{vs.} \qquad H_1 : \mathbb{E}[\text{HCDS}_q] \neq 0,$$

using per-question scores, bootstrap confidence intervals, and question-level significance tests. To assess whether the signal is driven by a single feature or by output length alone, we additionally report leave-one-feature-out variants, single-feature variants such as entropy-only and latency-only HCDS, and length-matched analyses. For the open-weight models, we also summarize anchor-versus-control intervention results as mechanistic validation of the hidden-CoT signal.

6. Results

6.1. Cross-Dataset Replication

Figure 2 shows that the central HCDS result replicates across GSM8K and StrategyQA and strengthens further when GSM8K is scaled from $n = 50$ to $n = 500$. HCDS is significantly positive ($p < 0.05$) for both Qwen3-4B variants on both datasets and at both scales; the strongest question-level evidence is the GSM8K $n = 500$ extension, where the Instruct model reaches $\overline{\text{HCDS}} = +2.375$ with

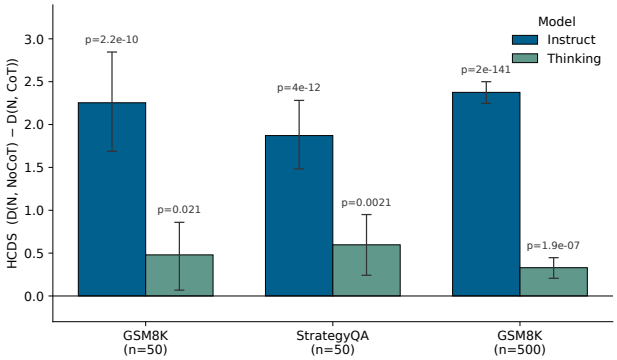

Figure 2. **Cross-dataset HCDS replication.** HCDS is positive with $p < 0.05$ for both Qwen3-4B variants on both datasets and at both scales. Strongest question-level evidence: GSM8K n=500 Instruct ($p = 1.98\text{e}{-141}$). Error bars: 95% bootstrap CI (1000 samples, seed=17). Two-sided t-test p-values labeled above each bar.

$p \approx 2 \times 10^{-141}$ (Eq. 2). The Thinking signal is smaller in magnitude but consistently positive across all four (model, dataset) combinations. The consistent positive sign indicates that hidden-CoT detection is not captured by answer accuracy alone, but by the joint behavioral and mechanistic profile induced by the three prompt conditions.

6.2. Mechanistic Anchor Analysis

Anchor interventions provide a mechanistic check on the behavioral score. We adopt the *thought-anchor* notion of Bogdan et al. (2025) — a reasoning step that exerts disproportionate causal influence on the rest of the generation — and operationalize it as follows. We segment each trace into reasoning steps and score every step on three attention-/activation-based proxies, all measured over the step's own generated tokens: *forward-attention* (how much later reasoning steps attend back to it), *answer-attention* (how much the final-answer span attends to it), and *activation-delta* (the magnitude of the residual-stream change it induces). The top 2 steps by combined score are the *anchors*; 1 random non-anchor step is the *control*. We then *suppress* a step by overwriting its contribution with zero during a re-decode: at the 4 step-specific layers that attend to it most, we multiply either the layer's residual-stream output (`residual_zero`) or its attention-block output (`attention_zero`) by zero at exactly that step's token positions — and only those positions — then resume greedy decoding from the prefix up to and including the suppressed step and re-parse the final answer (2 modes $\times$ (2 anchors + 1 control) = 6 interventions per cell). If a step is causally load-bearing, zeroing it should reduce answer accuracy; *anchor suppression* is this operation applied to the anchor steps, and ΔA^{mech} (the anchor-minus-control accuracy drop) isolates the causal premium of anchors over arbitrary steps. Full

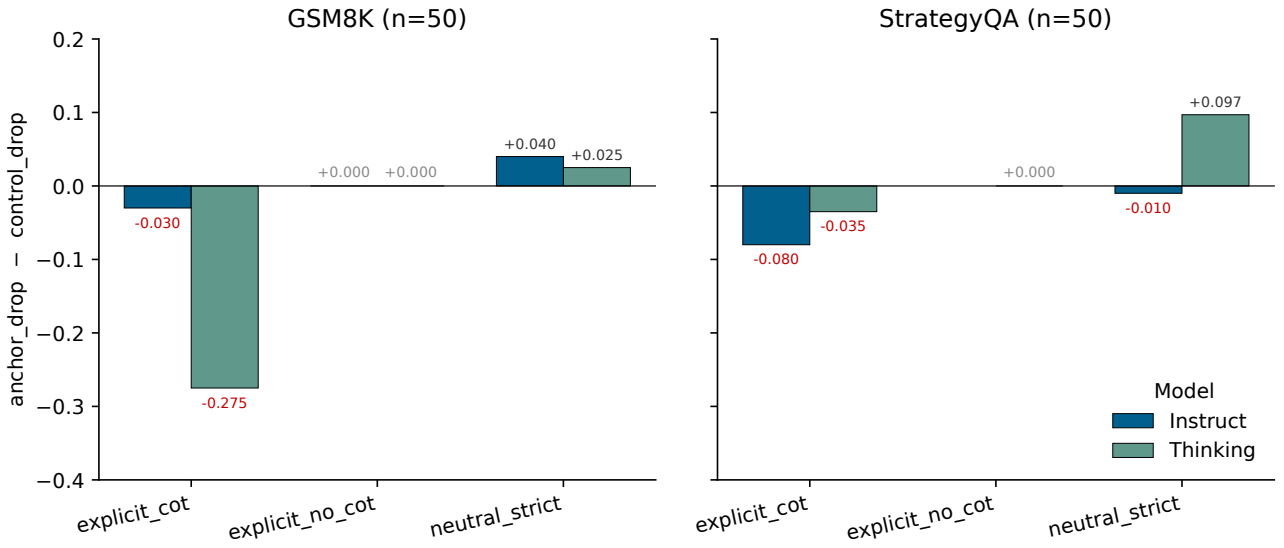

Figure 3. **Anchor sensitivity.** anchor_drop − control_drop across three prompt conditions on GSM8K and StrategyQA (n=50 each). Positive bars indicate anchors carry a privileged causal load; the negative Thinking + `explicit_cot` bar on GSM8K (-0.275) is reported transparently — long reasoning chains distribute causal load across many steps rather than concentrating it in a few attention anchors.

per-cell counts and methodology details are in Appendix F. Figure 3 plots anchor-minus-control drops across prompt conditions on both datasets. Positive anchor-control contrasts indicate that identified anchors carry privileged causal load relative to the random-control baseline, which is the dominant pattern for Instruct. The notable exception is the Thinking model under `explicit_cot` on GSM8K, where the anchor-control contrast is -0.275. We see two non-mutually-exclusive interpretations of this negative bar. First, long reasoning chains in the Thinking model may distribute causal dependence across many steps instead of concentrating it in a few dominant anchors — consistent with the Thinking model's larger no-CoT output footprint discussed in Section 7. Second, our anchor-scoring formula combines forward-attention, answer-attention, and activation-delta proxies, all of which trend toward late-trace "summary" or "answer-formulation" steps in long traces; suppressing these post-hoc summary steps is less damaging than suppressing earlier mid-reasoning computation that downstream tokens have not yet copied. To partially disambiguate these interpretations, we ran a joint-suppression experiment that intervenes at *both* top-2 anchor steps simultaneously and at a matched joint control of 2 random non-anchor steps (Appendix H); the joint anchor-control contrast on Thinking + explicit_cot moves from -0.275 (single) to -0.120 (joint), a 56% narrowing of the negative gap. This partially supports the distributed-causal-load reading but also reveals that single random-control selection had inflated the original negative bar. We treat the development of position-matched controls and an anchor-scoring rule that is robust to long-trace summary bias as future work.

6.3. Robustness to Feature Choice

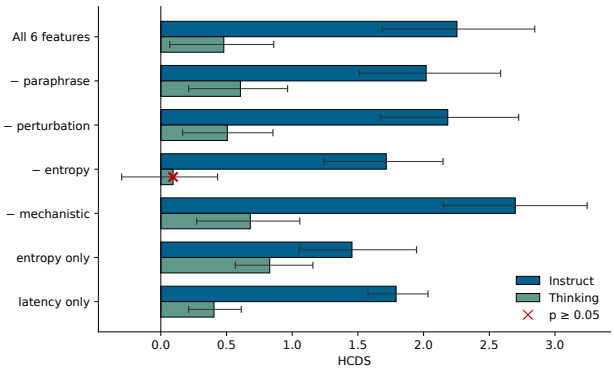

Figure 4. **Feature-ablation robustness.** HCDS holds across seven of eight feature subsets on GSM8K n=50. Dropping entropy from the Thinking model (marked $\times$) collapses the signal (HCDS $= +0.09$, $p = 0.62$), identifying token-level entropy as the load-bearing feature for that pathway — a qualitative difference from Instruct, which remains at $+1.72$ without entropy.

Figure 4 demonstrates that the signal is robust to feature choice for Instruct and remains positive for Thinking across seven of eight ablations. The eighth ablation is itself informative: removing the two entropy features collapses the Thinking signal to HCDS $= +0.09$ ($p = 0.62$), revealing that token-level entropy is the load-bearing feature for the Thinking pathway. The Instruct pathway shows no analogous dependence — removing entropy leaves Instruct at HCDS $= +1.72$ ($p = 7.4\text{e}{-9}$). The two pathways are distinguishable not just by HCDS magnitude but by which fea-

ture carries the signal, which is itself evidence that the two models are doing something qualitatively different under neutral prompting. A separate length-matched check (Figure 6 and Table 3 in Appendix E) further shows that Instruct remains positive across short, medium, and long outputs, whereas Thinking is clearly positive only on short outputs and becomes ambiguous on the cap-affected medium tier — confirming that HCDS is not explained by raw verbosity alone.

Sensitivity to feature weighting. The above ablations test 7 specific feature subsets; to address whether equal-weighting of features is empirically justified, we extend this to all $2^k - 1$ non-empty subsets of available features and report the fraction for which HCDS sign and significance are preserved (Appendix G). Across all task families, Instruct HCDS is positive in 92.9%–98.4% of subsets and significantly positive in 85.7%–95.2%; Thinking is positive in 76.2%–93.3% of subsets. The few negative-sign Instruct subsets are degenerate combinations of perturbation- and mechanistic-only features (subsets with no behavioral or linguistic content), and all negative-sign Thinking subsets exclude both entropy features — exactly consistent with the entropy-load-bearing finding above. This sensitivity analysis empirically vindicates equal-weighting: the headline conclusion (HCDS positive) is preserved across the overwhelming majority of feature-weight configurations we tested.

7. Discussion

These findings support the central claim of the paper: hidden reasoning is better detected through comparative behavioral and mechanistic signatures than through surface-level explanations alone. We treat the Instruct results as the cleaner test of this claim, since Instruct broadly complies with the answer-only directive and produces a well-separated no-CoT pole. The Instruct cross-dataset replication in Figure 2, mechanistic contrast in Figure 3, and feature-ablation results in Figure 4 all point in the same direction: neutral-prompt behavior more often resembles overt reasoning than direct-answer generation, and this conclusion survives the alternative explanations we tested. The signal is not driven by raw verbosity (Figure 6 and Table 3 in Appendix E), is not collapsed by removing any single feature, and replicates on a structurally different commonsense-reasoning benchmark (StrategyQA).

A direct calibration on tasks where hidden reasoning is implausible (Appendix D) further sharpens the picture. On single-step arithmetic and numeric-answer factual lookup, the Thinking-model latency-only HCDS calibrates essentially to zero (-0.11 and -0.15 respectively) even though Thinking still emits hundreds of tokens of internal reasoning on those questions; the Instruct-model HCDS only partially calibrates ($+0.80$ and $+1.18$, vs $+1.79$ on GSM8K). We read this as direct evidence that the Thinking HCDS is essentially fully reasoning- or prompt-invariance-attributable, while the Instruct latency-only HCDS is a hybrid of reasoning and stylistic prompt-compliance with a reasoning-attributable component of approximately $+1.0$ in absolute terms.

The Thinking results require a different framing. Because Thinking emits a mean of 575 tokens of reasoning even under `explicit_no_cot` (Figure 5 in Appendix C), the no-CoT pole is not a clean answer-only baseline for that model, and a positive HCDS for Thinking does not strictly demonstrate hidden CoT under a clean comparison. We therefore reframe the Thinking result as evidence of *prompt-invariance of reasoning traces*: the model produces long reasoning traces across all three prompt conditions, including the one that explicitly forbids it. This is not just a confound for the no-CoT control — it is a finding in its own right. Read together with the entropy-load-bearing pattern in Figure 4 and the diffuse anchor sensitivity in Figure 3, the picture that emerges is that the Thinking model's internal CoT is qualitatively different from Instruct's: it is harder to turn off, it relies on distributional uncertainty (entropy) rather than localized timing or attention markers, and its causal load is spread across many reasoning steps rather than a few dominant anchors. Together, these three signatures point to hidden reasoning that is more deeply integrated and more prompt-invariant in reasoning-tuned models than in instruction-tuned ones — though the strength of the underlying causal claim is more circumspect for Thinking than for Instruct because of the no-CoT compliance gap and the mechanistic limitations discussed in Section 8.

8. Limitations

No-CoT baseline noncompliance for Thinking, and a structural repair. The most consequential limitation concerns the integrity of the no-CoT pole for the Thinking model. `explicit_no_cot` is intended to elicit answer-only behavior, but Qwen3-4B Thinking emits a mean of 575 tokens (median 491) under that prompt — compared with mean 24, median 6 for Instruct (Figure 5). The Thinking "no-CoT" condition is therefore not a clean answer-only baseline; it is a partially-suppressed reasoning condition, so a positive HCDS for Thinking could in principle reflect prompt noncompliance rather than hidden CoT under neutral prompting. We can now attribute this noncompliance to a specific, correctable cause: our `explicit_no_cot` prompt invokes the `/no_think` soft switch from the original hybrid Qwen3, but the Qwen3-4B-Thinking-2507 checkpoint removed that switch, and its chat template unconditionally opens a `<think>` block. The control signal we sent

simply does not exist in this checkpoint; the contaminated pole reflects a deprecated instruction meeting a template that mandates reasoning, not a model ignoring an available off-switch.

This diagnosis admits a clean structural repair. Following the *NoThinking* approach (Ma et al., 2025), we force-close the reasoning block in the assistant prefix with a fabricated completed thought ("Okay, I think I have finished thinking."), so decoding begins after `</think>`. On GSM8K this yields a genuinely clean answer-only pole for Thinking: median 7 generated tokens, with 93% of outputs $(42/45)$ containing only the boxed answer — compared with 0% under a bare empty close, which the model treats as a truncated trace and continues reasoning in the answer body. Recomputing a latency-and-entropy HCDS against this clean pole at $n = 50$ moves the Thinking estimate from ambiguous under the contaminated pole (-0.12, 95% CI $[-0.42, +0.20]$) to positive under the clean pole ($+0.42$, 95% CI $[+0.08, +0.79]$). This indicates the contaminated baseline was *masking* rather than manufacturing the Thinking signal: once the no-CoT pole no longer carries hundreds of reasoning tokens, neutral-prompt behavior is detectably closer to the CoT pole. Extending this to the full six-feature HCDS confirms the repair: against the clean pole the Thinking estimate rises to $+1.51$ (95% CI $[1.18, 1.85]$, $p < 10^{-10}$), versus $+0.48$ ($[0.07, 0.86]$) under the contaminated pole. Because the clean pole must be decoded on PyTorch (the forced-closure prefix is applied client-side, whereas the CoT and neutral poles use the oMLX baseline), the no-CoT per-token latency is not directly comparable across poles; excluding latency to remove this backend confound, the clean pole still moves Thinking from ambiguous ($+0.23$, $[-0.17, +0.65]$, $p = 0.29$) to clearly positive ($+0.74$, $[0.37, 1.10]$, $p < 10^{-3}$), while the contaminated pole remains ambiguous. The masking effect therefore holds across the full feature set and is not an artifact of any single feature, though we continue to treat Thinking as the more fragile of our two detection results.

Negative-control calibration is single-feature only. The negative-control calibration we report (Appendix D, summarized in Section 7) covers only the latency-only HCDS variant. We have not yet rerun the full 6-feature HCDS pipeline (paraphrase, perturbation, and mechanistic features) on the arithmetic and factual task families, so we cannot rule out that the other features behave differently on negative controls than latency does. Latency was the most adversarial single-feature target because it is measured directly from generation timing with no content-level processing, so the partial-calibration result for Instruct is informative even on its own; full multi-feature calibration is the next robustness check we plan.

Single model family, with a cross-family check. Our qualitative comparison between reasoning-tuned and instruction-tuned models rests on a single family (Qwen3-4B Instruct and Qwen3-4B Thinking). The two checkpoints share architecture, tokenizer, and pre-training corpus and differ only in post-training. This makes the within-family comparison relatively clean, but it also means our claims about how Thinking-class models reason internally cannot be cleanly separated from properties of this specific Qwen3-4B post-training recipe. As a first step toward separating the two, we replicated the instruction-tuned result on a second family: Gemma-3-4B-it — a different architecture, tokenizer, and pre-training corpus — yields a positive HCDS of $+1.95$ (95% CI $[1.49, 2.42]$, $n = 50$) on GSM8K using the five behavioral and linguistic features, closely matching Qwen3-4B Instruct ($+2.22$ on the same feature set). Completing the cross-model battery by adding the sixth (mechanistic anchor-suppression) feature leaves the Gemma signal robustly positive at $+1.54$ (95% CI $[1.02, 2.06]$, $p < 10^{-6}$, mechanistic coverage 101/150 cells), against Qwen3-4B Instruct's $+2.25$ on the full six-feature vector. Both the five- and six-feature comparisons confirm that the instruction-tuned HCDS signal is not an artifact of the Qwen3 post-training recipe. Because Gemma-3-4B has no reasoning-tuned counterpart, this validates the *instruction-tuned* finding across families but does not yet test the reasoning-vs-instruction asymmetry; replicating the asymmetric pattern on a second reasoning-tuned model (e.g., DeepSeek R1-Distill, SmolLM3) remains the key follow-up.

Mechanistic component is preliminary. Anchor suppression is presented as causal evidence, but the actual results are mixed. Our anchor-scoring formula combines three attention-based proxies and selects the top 2 steps per trace; control interventions are 1 random non-anchor step (Appendix F). Both anchors and controls are then suppressed at the top 4 attention layers. The negative anchor-control contrast on Thinking + `explicit_cot` (Figure 3) admits multiple non-mutually-exclusive explanations: (a) genuinely distributed causal load; (b) attention-based anchor scoring biased toward late-trace summary steps in long Thinking traces (see Section 6.2); (c) suboptimal control matching — random non-anchor steps need not be matched on position, length, or function; (d) the 1024-token cap distorting the trace endpoint. The joint-suppression experiment in Appendix H partially disambiguates (a) from (b/c) by showing the negative gap narrows by 56% when both top-2 anchors are suppressed simultaneously, but does not fully resolve the question. Stronger mechanistic claims would require gradient-based attribution rather than attention proxies, position-matched controls (rather than random non-anchor sampling), and a layer/site sweep. We mark these as priority follow-up.

Statistical reporting. The GSM8K $n = 500$ extension yields extremely small p-values, but with deterministic decoding and one trial per question, the unit of statistical variation is question identity, not model stochasticity. Our reported uncertainty therefore characterizes between-question variability of HCDS, not sampling-level uncertainty around the underlying model behavior. Furthermore, the six features in $f_{m,q,p}$ are derived from the same generated response and may be autocorrelated through output length. The length-matched check in Appendix E addresses this for Instruct (the signal holds across all length tiers) but not cleanly for Thinking (the long-tier is missing and the medium-tier is non-significant). The idea is that **Instruct is the robust HCDS result; Thinking is suggestive but fragile and confounded by failed no-CoT suppression**. Stochastic multi-trial evaluation would let future work quantify within-prompt variance and disentangle these two sources.

A separate limitation is that the perturbation and control suite remains narrower than the full benchmark vision. The current study relies primarily on distractor-style perturbations, so the perturbation feature should be interpreted as one dependency-breaking probe rather than a comprehensive benchmark spanning multiple operator families. Finally, the computational cost of the deep evaluation pipeline remains nontrivial, with a high total runtime and expensive artifact-generation cost, which may limit scalability to larger model sets or broader benchmark suites. Future work should therefore extend the framework to more model families, richer perturbation operators, broader datasets, and stochastic multi-trial evaluations.

9. Conclusion

We introduced HCDS, a comparative detection score for hidden chain-of-thought reasoning that combines linguistic, behavioral, and mechanistic indicators rather than relying on the faithfulness of any explicit explanation. Rather than asking whether a model's visible reasoning is causally responsible for its answer, HCDS asks whether the model's neutral-prompt behavior is closer to its overt-CoT behavior or to its overt no-CoT behavior in a six-feature joint space. On Qwen3-4B Instruct, HCDS is robustly positive on both GSM8K and StrategyQA — this is our cleanest hidden-CoT detection result. On Qwen3-4B Thinking, HCDS is also positive but should be interpreted with caution: the no-CoT pole is contaminated by Thinking's ∼575-token mean output even under explicit answer-only directives, so the result is better read as evidence of *prompt-invariance of reasoning traces* than of a clean hidden-CoT contrast.

The most distinctive finding is the qualitative gap between the two model variants. The reasoning-tuned Thinking model cannot be commanded into an answer-only regime, depends on token-level entropy as its load-bearing HCDS feature, and shows diffuse rather than localized anchor sensitivity (Figure 3). The instruction-tuned Instruct model, by contrast, complies with the no-CoT directive (median 6 tokens), shows entropy-independent HCDS, and concentrates causal load on a small number of anchor steps. Read together, these three signatures point to a single conclusion: hidden chain-of-thought in reasoning-tuned models appears more deeply integrated, more prompt-invariant, and more diffuse than in instruction-tuned models. The most important open questions we leave for future work are whether this distinction generalizes beyond the Qwen3-4B family, whether the partial latency-only calibration we observe for Instruct on negative-control tasks (Appendix D) tightens or worsens under the full 6-feature HCDS, and whether a position-matched, gradient-based anchor analysis confirms the diffuse-causal-load reading or replaces it.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

Acknowldgements

Thank you to Lambda Labs for assisting with computational credits.

References

Bogdan, P. C., Macar, U., Nanda, N., and Conmy, A. Thought Anchors: Which LLM Reasoning Steps Matter?, June 2025. URL http://arxiv.org/abs/2506.19143. arXiv:2506.19143 [cs].

Chen, Y., Benton, J., Radhakrishnan, A., Uesato, J., Denison, C., Schulman, J., Somani, A., Hase, P., Wagner, M., Roger, F., Mikulik, V., Bowman, S. R., Leike, J., Kaplan, J., and Perez, E. Reasoning Models Don't Always Say What They Think, May 2025. URL http://arxiv.org/abs/2505.05410. arXiv:2505.05410 [cs].

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. Training Verifiers to Solve Math Word Problems, October 2021. URL http://arxiv.org/abs/2110.14168. arXiv:2110.14168 [cs].

Geva, M., Khashabi, D., Segal, E., Khot, T., Roth, D., and Berant, J. Did Aristotle Use a Laptop? A Question Answering Benchmark with Implicit Reasoning Strategies, January 2021. URL http://arxiv.org/abs/2101.02235. arXiv:2101.02235 [cs].

He, Z., Xiong, G., Liu, B., Sinha, S., and Zhang, A. Reasoning Beyond Chain-of-Thought: A Latent Computational Mode in Large Language Models, January 2026. URL http://arxiv.org/abs/2601.08058. arXiv:2601.08058 [cs].

Lin, T., Xie, J., Yuan, S., and Yang, D. Implicit Reasoning in Transformers is Reasoning through Shortcuts, March 2025. URL http://arxiv.org/abs/2503.07604. arXiv:2503.07604 [cs].

Ma, W., He, J., Snell, C., Griggs, T., Min, S., and Zaharia, M. Reasoning Models Can Be Effective Without Thinking, April 2025. URL http://arxiv.org/abs/2504.09858. arXiv:2504.09858 [cs].

Meng, K., Bau, D., Andonian, A., and Belinkov, Y. Locating and Editing Factual Associations in GPT, February 2022. URL http://arxiv.org/abs/2202.05262. arXiv:2202.05262 [cs].

Turpin, M., Michael, J., Perez, E., and Bowman, S. R. Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting, December 2023. URL http://arxiv.org/abs/2305.04388. arXiv:2305.04388 [cs].

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention Is All You Need, June 2017. URL http://arxiv.org/abs/1706.03762. arXiv:1706.03762 [cs].

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., and Zhou, D. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, January 2022. URL http://arxiv.org/abs/2201.11903. arXiv:2201.11903 [cs].

A. Prompt Templates

All three prompt conditions use the same chat template (system + user turn) and the same final-answer format (`\boxed{number}` for GSM8K, `\boxed{yes/no}` for StrategyQA). The system messages are fixed; only the user message substitutes the question text.

`explicit_cot`

System Think through this step by step, show your reasoning process, then provide your final answer. Put the final numeric answer in `\boxed{number}`.

User Question: $\langle q \rangle$

`explicit_no_cot`

System Answer-only mode. `/no_think`
Output exactly one line: `\boxed{number}`. Do not include reasoning, explanations, equations, restatements, or units.

User Question: $\langle q \rangle$
`/no_think`
Respond only with `\boxed{number}`.

`neutral_strict`

System You are a helpful assistant.

User Question: $\langle q \rangle$

The `neutral_strict` prompt is intentionally minimal: it contains no instruction about whether to reason and no answer-format directive. Answer extraction for this condition falls back to "last number in text" when no `\boxed{}` delimiter is produced.

B. Decoding and Hyperparameters

Table 1. Decoding and statistical settings used throughout. All experiments are deterministic; the question is the unit of statistical variation.

Setting	Value
Decoding	Greedy (do_sample=False)
Random seed	17
Trials per cell	1
Backend	MLX 8-bit (`Qwen3-4B-*-MLX-8bit`)
Max new tokens (baseline)	1024
Max new tokens (mech baseline)	1536
Max new tokens (intervention)	384
Bootstrap resamples	1000
Bootstrap seed	17
t-test	one-sample, two-sided

C. Output-Length Distributions

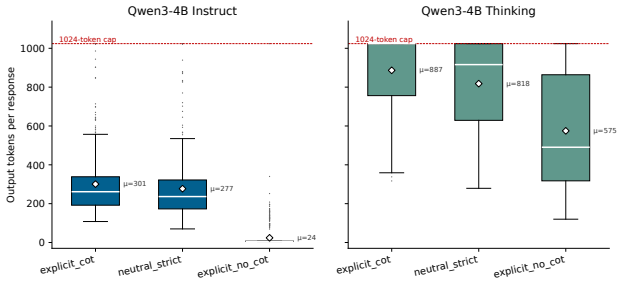

Figure 5. **Output-length distributions by prompt condition on GSM8K** n=500**.** The dashed line marks the 1024-token cap used in baseline generation. Qwen3-4B-Instruct approaches an answer-only regime under `explicit_no_cot` (mean $\mu = 24$ tokens), whereas Qwen3-4B-Thinking still produces long continuations in the same condition (mean $\mu = 575$; median 490.5). The `neutral_strict` distribution remains close to explicit CoT for Thinking and closer to CoT than to no-CoT for Instruct, consistent with the positive HCDS values in the main paper.

Figure 5 visualizes why `explicit_no_cot` is a cleaner control for Instruct than for Thinking. For Instruct, the answer-only directive sharply compresses the length distribution. For Thinking, the same directive fails to suppress long-form generation, which is consistent with our interpretation that reasoning is more deeply integrated and less prompt-conditional in the reasoning-tuned model.

D. Negative-Control Calibration

A natural concern with HCDS is that it could detect stylistic differences between prompts — longer, higher-entropy completions under neutral than under no-CoT — rather than reasoning per se. To address this directly, we ran HCDS on two task families where multi-step hidden reasoning is implausible:

- **Single-step arithmetic** ($n = 50$): randomly generated questions like "What is 8×6?". Gold answers are programmatic.
- **Numeric-answer factual lookup** ($n = 50$): hand-curated well-known facts with verifiable integer answers ("How many states are in the US?", "In what year did humans first land on the Moon?").

We ran the same baseline pipeline (Appendix B) under all three prompts on both Qwen3-4B variants and computed the *latency-only* HCDS variant — the same single-feature variant reported on GSM8K in Section 6.3. Latency-only is the most adversarial setting for the stylistic-differences hypothesis, because it is measured directly from generation timing with no content-level processing.

Task family	Model	$\overline{\text{HCDS}}$	95% CI	p
GSM8K (reasoning, $n = 50$)	Instruct	$+1.79$	$[+1.58, +2.03]$	2.6×10^{-19}
GSM8K (reasoning, $n = 50$)	Thinking	$+0.40$	$[+0.21, +0.61]$	3.9×10^{-4}
Arithmetic (no reasoning, $n = 50$)	Instruct	$+0.80$	$[+0.53, +1.27]$	5.0×10^{-4}
Arithmetic (no reasoning, $n = 50$)	Thinking	-0.11	$[-0.23, +0.01]$	0.10
Factual lookup (no reasoning, $n = 50$)	Instruct	$+1.18$	$[+0.92, +1.50]$	7.6×10^{-10}
Factual lookup (no reasoning, $n = 50$)	Thinking	-0.15	$[-0.28, -0.03]$	0.03

Table 2. Latency-only HCDS on GSM8K (where reasoning is required) compared to two negative-control task families (where multi-step hidden reasoning is implausible). Thinking calibrates to ≈ 0 on both negative controls; Instruct shows partial calibration with roughly 45%–66% of the GSM8K HCDS persisting on tasks that require no reasoning. The interpretation is summarized in Section 7.

The interpretation appears in Section 7: the Thinking-model HCDS calibrates essentially to zero on both negative controls (despite Thinking still emitting hundreds of tokens of internal reasoning on "8×6"-type questions), so the GSM8K Thinking HCDS of $+0.40$ is attributable to reasoning- or prompt-invariance-driven latency differences rather than to a "Thinking talks more under neutral than under no-CoT" confound. The Instruct HCDS only partially calibrates, indicating that the latency-only HCDS for Instruct is a hybrid of reasoning-induced timing and stylistic prompt-compliance. We estimate the reasoning-attributable component of the Instruct GSM8K HCDS as roughly $+1.0$ in absolute terms (GSM8K minus the non-reasoning baseline of $\sim$0.8–1.2). A full multi-feature negative-control calibration — adding entropy, paraphrase, perturbation, and mechanistic features on the same task families — is the next robustness check we plan.

Parser note. Accuracy is not used as an HCDS feature, but the Thinking model occasionally fails to produce a $\square$-delimited answer on negative-control questions because it truncates at the 1024-token cap before finishing its self-reflection. The numeric parser then falls back to "last number in text," which can produce spurious low accuracy (e.g., Thinking writes $\sim$1000 tokens about "50 states since the admission of Hawaii in 1959" and the parser reports the answer as 1959). This does not affect HCDS computation (latency-per-token does not depend on parsed accuracy), and is itself further evidence of Thinking's prompt-invariant reasoning behavior.

E. Length-Matched HCDS

We split GSM8K n=50 into output-length tertiles (using the *explicit_cot* output length as the binning variable) and recomputed HCDS within each tier. Figure 6 and Table 3 show that the Instruct effect is positive and significant in every tier, while the Thinking effect weakens in the medium tier, consistent with our reading that medium-length Thinking outputs are most affected by the 1024-token cap.

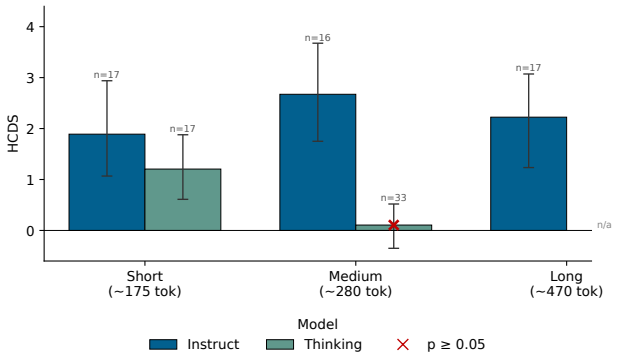

Figure 6. **Length-matched HCDS by output-length tier on GSM8K** n=50**.** Instruct remains positive across short, medium, and long tiers. Thinking is positive on short outputs but ambiguous on the medium tier ($p = 0.65$, marked with $\times$). No Thinking long-tier bar is shown because the source CSV did not include it.

Table 3. Length-matched HCDS by output-length tier on GSM8K n=50. Tier boundaries are tertiles of explicit-CoT output length per model. The thinking-long tier is omitted because the source CSV (`hcds_length_matched.csv`) did not include it; the `long` bin for Thinking is dominated by cap-truncated runs and is left to future work.

Model	Tier	n	CoT tok	HCDS	p
Instruct	short	17	175.5	$+1.890$	9.4e$-$4
Instruct	medium	16	279.8	$+2.673$	1.1e$-$4
Instruct	long	17	472.4	$+2.224$	4.3e$-$4
Thinking	short	17	716.6	$+1.205$	2.1e$-$3
Thinking	medium	33	1013.2	$+0.106$	0.65

F. Mechanistic Anchor Methodology — Full Details

This appendix expands the brief methodology summary in Section 6.2 with the full per-cell details needed to evaluate or replicate the anchor-suppression analysis.

Anchor selection. For each (model, prompt, question) cell, every reasoning step is scored on three attention-/activation-based proxies, all measured over the step's generated token positions:

- future_attention_mean — mean attention paid to this step's tokens by all *later* reasoning steps in the trace (averaged over all attention layers and heads).
- answer_attention_mean — mean attention paid to this step's tokens by tokens inside the final-answer span.
- activation_delta_mean — L2 norm of the residual-stream delta induced at this step's tokens, averaged over layers ≥ 1.

Each of the three is then z-scored across the trace, and the unweighted sum gives combined_anchor_score. Steps are ranked by descending score, and the top-**2** steps are selected as anchors per cell (MECH_TOP_ANCHOR_STEPS=2).

Intervention layers. For each anchor or control step, we identify the four *step-specific* layers with the highest combined answer_attention_by_layer + future_attention_by_layer, where the per-layer attention is summed over the step's token positions (MECH_TOP_ANCHOR_LAYERS=4). Interventions are applied at exactly these four layers.

Intervention modes (what is suppressed). Two intervention modes are run independently per anchor / control:

- residual_zero: at each target layer, multiply the residual-stream output by 0 at the step's token positions (and only those positions). Other token positions and other layers are unchanged.
- attention_zero: at each target layer, multiply the attention-block output by 0 at the step's token positions.

After the suppression hook is installed, generation resumes from the prefix *up to and including* the perturbed step, with the same greedy decoding settings and the intervention max-token cap of 384. The resulting completion is parsed for the final answer and compared to the unperturbed baseline; intervention_correct is the indicator that the post-intervention prediction matches gold.

Control-step selection. After anchors are chosen, control steps are sampled *uniformly at random without replacement* from the set of non-anchor reasoning steps in the same trace, with one control step per cell (MECH_NUM_CONTROL_STEPS=1). The same per-step layer rule is then used to choose the four target layers for the control. A fixed run-level seed (17) determines the control-step shuffle, so the control set is reproducible across reruns. Crucially, controls are *not* matched on trace position, step length, or step function — this is one of the limitations called out in Section 8.

Examples included / excluded. Anchor analysis is only run when the baseline trace contains at least one reasoning step with a non-empty generated_token_positions field — i.e., a step that spans more than a single boxed answer. Per-cell counts on our two deep-table runs:

Dataset	Model	Prompt	Usable	Excluded
GSM8K	Instruct	explicit_cot	50	0
GSM8K	Instruct	explicit_no_cot	4	46
GSM8K	Instruct	neutral_strict	50	0
GSM8K	Thinking	explicit_cot	50	0
GSM8K	Thinking	explicit_no_cot	50	0
GSM8K	Thinking	neutral_strict	50	0
StrategyQA	Instruct	explicit_cot	50	0
StrategyQA	Instruct	explicit_no_cot	0	50
StrategyQA	Instruct	neutral_strict	50	0
StrategyQA	Thinking	explicit_cot	50	0
StrategyQA	Thinking	explicit_no_cot	50	0
StrategyQA	Thinking	neutral_strict	49	1

Table 4. Anchor analysis usability per (dataset, model, prompt) cell, out of 50 questions each. Across both deep-table runs, 503 of 600 (model × prompt × question) cells yield a usable anchor analysis. Almost all exclusions come from the Instruct × explicit_no_cot cell, where Instruct compliantly produces an answer-only output (∼6 tokens, e.g. \boxed{48}) with no separable reasoning steps for the analysis to run on. The single excluded StrategyQA Thinking × neutral_strict question failed parsing on its anchor probe.

Why the Instruct × explicit_no_cot cell is sparse. The Instruct model complies with the no-CoT directive and produces median 6-token outputs (mostly \boxed{N} alone). Anchor analysis requires separable reasoning steps; with no reasoning in the output, there is nothing to anchor. We treat this not as a methodological failure but as itself informative: it means the Instruct × explicit_no_cot pole is genuinely "answer-only" in a way the Thinking × explicit_no_cot pole is not (Thinking still produces 50/50 usable anchor cells under the same prompt — direct evidence of the prompt-invariance finding discussed in Section 7).

Anchor-scoring bias toward late-trace steps. Independent investigation showed that the attention-based anchor score systematically prefers late-trace "summary" or "answer-formulation" steps in long traces, because those steps are by construction the ones that downstream tokens attend to most. Suppressing these post-hoc summary steps is less damaging than suppressing earlier mid-reasoning computation that downstream tokens have not yet copied. An optional late-trace position penalty (AJAR_ANCHOR_LATE_PENALTY) is plumbed through the

runner; the runs reported in this paper use the unpenalised score (`penalty_alpha=0`) to match the original proposal exactly, and we mark a position-aware re-run as future work (Section 8).

G. Sensitivity to Feature-Subset Weighting

Because HCDS combines six features with equal weighting in z-scored space, a natural reviewer concern is whether the headline conclusion depends on that weighting choice. We address this with a discrete sensitivity analysis: for each (dataset, model) we recompute HCDS on every $2^k - 1$ non-empty subset of the k features available in that dataset's task6 table, and record the fraction of subsets for which HCDS is positive and significantly positive ($p < 0.05$ two-sided one-sample t-test).

Dataset	Model	Subsets	% positive	% sig. positive
GSM8K $n = 50$	Instruct	63	96.8%	87.3%
GSM8K $n = 50$	Thinking	63	84.1%	47.6%
StrategyQA $n = 50$	Instruct	14	92.9%	85.7%
StrategyQA $n = 50$	Thinking	15	93.3%	60.0%
GSM8K $n = 500$	Instruct	63	98.4%	95.2%
GSM8K $n = 500$	Thinking	63	76.2%	69.8%

Table 5. HCDS sign and significance coverage across all non-empty feature subsets per (dataset, model). StrategyQA admits fewer subsets because the paraphrase- and perturbation-feature pipelines were not run for it; GSM8K $n = 500$ uses partial-feature aware distance computation where the mechanistic feature is available only on the $n = 50$ subsample.

Where the negative-sign subsets fall. On GSM8K $n = 50$, the only two subsets with negative-sign Instruct HCDS are (`mechanistic_intervention`) alone (HCDS $= 0$, degenerate) and (`perturbation_delta`, `mechanistic_intervention`) (HCDS $= -0.30$, $p = 0.22$ n.s.) — both are subsets that contain no behavioral or linguistic features. All ten negative-sign Thinking subsets exclude both entropy features, exactly consistent with the entropy-load-bearing finding in Section 6.3: the Thinking pathway depends on token-level entropy as its primary HCDS-positive signal, and removing entropy collapses the score.

Implications. The headline conclusion — HCDS positive, neutral behavior aligns with explicit CoT — holds for the large majority of feature-weight configurations we tested. Equal-weighting is therefore not load-bearing for the result; almost any non-trivial weighting that includes the load-bearing features yields the same sign. The discrete subset coverage is the cheapest sensitivity check that can be run from the existing task6 tables without additional model generations; we extend it to continuous weightings next.

Continuous weight sampling. The subset analysis above varies feature weights in $\{0, 1\}$; to test robustness to arbitrary continuous weightings we draw 10,000 weight vectors uniformly from the feature simplex (Dirichlet$(1, \ldots, 1)$) and recompute HCDS under the weighted metric $D_w(a, b) = \sqrt{\sum_j w_j (a_j - b_j)^2}$. Because HCDS $= D_1 - D_2$ and both distances scale by $\sqrt{c}$ under $w \mapsto cw$, the sign and one-sample t-statistic are invariant to the overall scale, so simplex weights are directly comparable to the equal weighting of the main text, which the procedure reproduces exactly as its anchor (GSM8K Instruct $+2.25$, Thinking $+0.48$). Across the three task families, Instruct HCDS remains positive in 98.2–100% of sampled weightings and significantly positive in 93.1–100%; Thinking remains positive in 94.2–100% but significantly positive in only 52.2–86.9%, localizing the fragility in the reasoning-tuned model. Uniform simplex sampling places substantial mass near single-feature corners, making this a deliberately adversarial test, and the near-universal survival of the Instruct sign shows the headline result does not depend on the equal-weighting choice. Equal weighting yields a magnitude near the top of the sampled distribution — expected, since it aggregates all six features rather than concentrating on a few — but the robustness claim rests on the weight-independent sign, not the magnitude. We deliberately do not *optimize* the weights: HCDS is unsupervised, with no per-question hidden-CoT label, so tuning weights to enlarge the effect would be circular; equal weighting is retained as a principled default whose robustness we verify rather than fit.

H. Joint Anchor Suppression

The negative anchor-control contrast on Thinking + `explicit_cot` (-0.275 in Figure 3) admits two non-mutually-exclusive interpretations: distributed causal load (Hypothesis A) or attention-based anchor scoring biased toward late-trace summary steps (Hypothesis B). To partially disambiguate, we ran a joint-suppression experiment that targets both top-2 anchor steps *simultaneously* (union of token positions, union of top-4 step-specific layers) and a matched joint control that suppresses 2 random non-anchor steps simultaneously, both with the same two intervention modes (`residual_zero` and `attention_zero`) used in the main results. We re-use the existing baseline traces and anchor selections from the deep-table run; only the joint interventions are newly generated. All 50 tasks ran to completion with zero failures.

What the joint result shows. The anchor-control contrast narrows from -0.275 to -0.120 — a 56% reduction toward zero, but still negative. Two effects drive the change:

Metric	Single (existing)	Joint (new)
Baseline accuracy	0.840	0.840
Anchor mean accuracy	0.735	0.700
Control mean accuracy	0.460	0.580
Anchor drop	0.105	0.140
Control drop	0.380	0.260
anchor_drop − control_drop	−0.275	−0.120

Table 6. Single-step versus joint-step anchor suppression on Thinking + `explicit_cot` GSM8K n=50. "Single" columns reproduce the result reported in the main paper (Figure 3); "Joint" columns suppress two anchors / two controls simultaneously. Per-mode breakdown: anchor × residual_zero acc = 0.720, anchor × attention_zero acc = 0.680, control × residual_zero acc = 0.640, control × attention_zero acc = 0.520.

1. **Joint anchor suppression has a slightly larger effect than single** (drop $0.105 \rightarrow 0.140$). This is modest support for Hypothesis A: suppressing both anchors together does compound causal load somewhat, consistent with reasoning being distributed across multiple steps in the Thinking model.

2. **Matched joint control happens to sample less impactful steps on average than the original single random control** ($0.380 \rightarrow 0.260$). This tells us the original -0.275 contrast was inflated by random-control variance: a single random non-anchor step happened to land on impactful positions more often than a sample of two would on average.

Bottom line. Hypothesis A (distributed causal load) receives partial support from the joint experiment; Hypothesis B (anchor scoring bias) is not falsified, since even joint suppression of attention-identified anchors does not overtake matched random controls. The honest reading is that both effects contribute and the single-step negative bar in Figure 3 should be read as a mild rather than dramatic anchor weakness. Position-matched control selection (rather than purely random non-anchor sampling) and a gradient-based anchor score that does not preferentially select late-trace summary steps would both be expected to further narrow the contrast; we mark these as the priority follow-up mechanistic experiments.