When Reasoning Traces Become Performative: Step-Level Evidence that Chain-of-Thought Is an Imperfect Oversight Channel

Anonymous Author(s)
Affiliation
Address
email

Abstract

Chain-of-thought (CoT) traces are increasingly used both to improve language-model capability and to audit model behavior, implicitly assuming that the visible trace remains synchronized with the computation that determines the answer. We test this assumption with a step-level *Detect-Classify-Compare* framework built around an answer-commitment proxy that is cross-validated with Patchscopes, tuned-lens probes, and causal direction ablation. Across nine models and seven reasoning benchmarks, latent commitment and explicit answer arrival align on only 61.9% of steps on average. The dominant mismatch pattern is *confabulated continuation*: 58.0% of detected mismatch events occur after the answer-commitment proxy has already stabilized while the trace continues producing deliberative-looking text, and a vacuousness analysis shows that the committed answer does not change during these steps. In architecture-matched Qwen2.5/DeepSeek-R1-Distill comparisons, the reasoning pipeline changes failure composition more than aggregate alignment, most clearly at 32B where confabulated steps decrease as contradictory states increase. Lower step-level alignment is also associated with larger CoT utility, suggesting that the settings that benefit most from CoT are often the least temporally faithful. Paired truncation and a complementary donor-corruption test further indicate that much pure-CS text is not load-bearing for the final answer. These findings suggest that CoT can remain useful while still being an unreliable report of when the answer was formed.

1 Introduction

Chain-of-thought (CoT) reasoning began as a prompting technique for improving multi-step performance [59, 26], but in current reasoning model pipelines it also serves as a training target, a verification interface, and an oversight signal [32, 54, 18, 39, 27, 13]. Humans and monitor models increasingly inspect reasoning traces to judge whether a system reasoned safely, honestly, or competently. This dual role rests on a strong assumption: that the visible trace faithfully tracks the internal computation that produced the answer. If that assumption fails, CoT may still improve capability while providing a misleading basis for oversight.

A growing literature suggests that this assumption is fragile. Perturbation, bias-injection, and deployment studies document omitted hints, post-hoc rationalization, and other forms of unfaithful CoT [28, 50, 41, 8, 2, 30]. Conceptual and empirical work further suggests that faithfully verbalizing autoregressive computation may itself be difficult [10, 49, 45, 38]. The question is therefore no longer whether unfaithful CoT exists, but how often it arises in current reasoning models and what kind of mismatch it creates.

Submitted to 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Do not distribute.

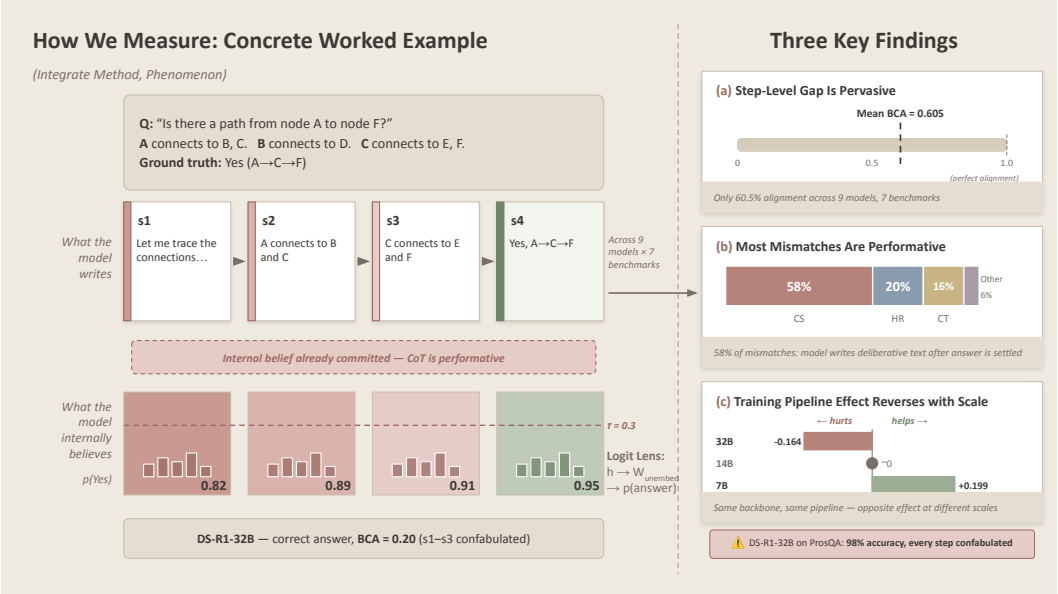

Figure 1: Overview of the empirical picture. Left: on a ProsQA example, DS-R1-32B commits to the answer by step 1 while the written trace continues, yielding low BCA. Right: the aggregate results show the same pattern at scale, with 61.9% mean alignment, 58% of mismatches classified as confabulated steps, and pipeline effects that vary across model scales.

Yet the field still lacks a measurement lens tailored to this oversight question. Most existing faithfulness methods deliver response-level verdicts [28, 41, 47]: they tell us whether a trace is suspect, but not when the answer was already internally fixed relative to the visible chain. Concurrent work by Boppana et al. [6] studies a closely related phenomenon under the label *performative CoT*, showing that reasoning traces can continue after latent answer commitment, especially on easier tasks. Related work by Palod et al. [40] questions whether longer CoTs should be interpreted as evidence of more adaptive computation. Our focus is complementary but narrower. We center the *timing* of internal answer formation, distinguish five step-level mismatch patterns rather than a binary split, compare architecture-matched Qwen2.5/R1 pairs across three scales, and quantify the relationship between temporal faithfulness and CoT utility. Interpretability tools can decode emerging predictions inside the model [37, 5, 15, 35], but they have rarely been used to measure CoT faithfulness at the step level.

We introduce the DETECT-CLASSIFY-COMPARE framework to close this gap. At each CoT step, we extract an answer-commitment proxy using logit lens projections, cross-validate that proxy with Patchscopes concordance, tuned-lens probes, and direction ablation, then quantify timing divergence, classify mismatch events, and compare matched model pairs that differ in whether they underwent the DeepSeek-R1 distillation pipeline. Across nine models and seven reasoning benchmarks (Figure 1), the dominant failure mode is not overt contradiction but *post-commitment continuation*: the model often resolves the answer internally and then continues generating plausible reasoning text. The resulting picture is one in which a fluent and even accuracy-improving chain can still be a poor report of when the answer was formed.

Our investigation makes three contributions:

1. We introduce a step-level framework for measuring whether internal answer formation is temporally reflected in CoT, and find mean alignment of only 61.9%, with 58.0% of mismatch events falling into *confabulated* continuation.

2. We show that in architecture-controlled Qwen2.5/DeepSeek-R1-Distill comparisons, the reasoning pipeline changes not only aggregate alignment but also the composition of failures, most sharply at 32B where confabulated steps fall as contradictory states rise.

3. We connect this timing measure to downstream behavior: lower step-level alignment tends to coincide with larger CoT utility, while paired truncation and donor corruption indicate that much post-commitment text is descriptive rather than load-bearing.

2 Related Work

CoT faithfulness, oversight, and monitoring. Chain-of-thought prompting improves reasoning performance [59, 26, 63, 38, 55], but whether the resulting trace is a faithful explanation or oversight signal remains contested [60, 11]. Perturbation, bias-injection, and deployment studies show that models can preserve answers under trace edits, omit decisive information, or produce post-hoc rationalizations [28, 50, 8, 2]. This matters because CoT is increasingly used in process supervision, verifier training, and safety monitoring workflows, where monitorability may fail even when CoT remains useful for capability [32, 54, 27, 13, 33, 68, 36]. Most of this literature provides response-level verdicts [42, 31, 47]; our question is narrower and more temporal: when does internal answer formation diverge from the visible trace?

Internal readouts and performative CoT. Interpretability tools such as the logit lens [37], tuned lens [5], Patchscopes [15], and linear probes [35, 29, 7, 3, 25] make it possible to track emerging predictions during generation. We use these tools not to recover a full reasoning circuit, but to estimate when answer commitment becomes decodable. Closest to our setting, Boppana et al. [6] study performative CoT and show that latent answer commitment can precede what the trace reveals, especially on easier tasks. Palod et al. [40] similarly cautions against interpreting longer CoTs as evidence of more adaptive computation. Other recent work measures step-level faithfulness through parameter-level unlearning or hint-injection paradigms [51, 65, 66]. Our contribution is more specific: a timing-based latent answer proxy cross-validated by multiple readouts, a five-way step-level taxonomy of mismatch patterns, architecture-matched Qwen2.5/DeepSeek-R1 comparisons across scale, and an empirical link between temporal faithfulness and CoT utility. A detailed axis-by-axis comparison against these and other closely related works is provided in Appendix O. Complementary work on normalized step-importance metrics such as NLDD targets cross-architecture comparability, whereas BCA is designed as a timing metric centered on when latent answer commitment becomes decodable relative to answer revelation [64].

Reasoning pipelines and training effects. Reasoning-oriented training pipelines, including process supervision, reinforcement learning, and distillation, increasingly optimize models to produce extended reasoning traces [52, 9, 67, 22, 18, 39, 20]. Whether these pipelines improve faithfulness rather than only usefulness remains unclear: longer or more polished traces may still fail to report the computation that determined the answer, and concurrent work has raised concerns about strategically uninformative or steganographic reasoning traces [44, 8]. Conversely, targeted interventions such as counterfactual simulation training can improve faithfulness in controlled settings, suggesting that the effect of training is real but objective-dependent [21]. Our matched-family comparison targets this question directly by comparing Qwen2.5 models against their DeepSeek-R1-distilled counterparts at 7B, 14B, and 32B, which lets us study how the pipeline reshapes timing alignment and failure composition while minimizing backbone confounds.

3 Method

DETECT-CLASSIFY-COMPARE follows a chain of thought one step at a time. For each step, we estimate whether the model has already committed to the final answer, compare that latent state with what the written trace has revealed so far, summarize the resulting mismatches with a taxonomy, and then use matched model pairs to study how reasoning training changes the alignment pattern.

3.1 Step-Level Belief Extraction

Given a model's CoT response to a question, we first segment the trace into reasoning steps $s_1, \ldots, s_T$ using a rule-based parser that splits on newlines, transition markers, and sentence boundaries (parsing F1 = 0.87; validation in Appendix F).

At each step boundary, we extract a proxy for the model's internal commitment to the final answer using the logit lens [37, 58]. Concretely, we project each layer's hidden state through the unembedding matrix and layer normalization to obtain a distribution over the vocabulary. We use the probability assigned to the ground-truth answer token, $p_{\text{ans}}^{(\ell)}(t)$, as our primary measure of internal commitment at step t and layer ℓ. This operationalization captures the dominant signal but may not reflect all aspects of the model's internal state; we discuss this limitation in Section 8.

We validate the logit lens through concordance with Patchscopes [15] and tuned lens probes [1, 4]. Patchscopes agrees on 75.6% of mid-depth steps and 74.7% of final-layer steps, with near-zero logit-lens-only disagreement (0.0% mid, 1.5% final); tuned-lens binary classifications agree with the logit lens at $\geq$99.2% across four held-out model-benchmark configurations (Appendix J.1). Together these checks indicate that the logit lens behaves as a conservative answer-commitment readout rather than a trigger-happy artifact.

3.2 Divergence Quantification

We quantify timing alignment using two complementary metrics. **Belief-CoT Agreement** (BCA) measures the fraction of steps at which the thresholded answer-commitment proxy ($p_{\text{ans}} > \tau$, $\tau = 0.3$) matches a binary CoT label indicating whether the trace has already explicitly reached the final answer. We compute that label automatically by checking, at each step, whether the ground-truth answer string or a normalized variant has appeared in the cumulative CoT text. A BCA of 1.0 indicates perfect timing alignment. Because the CoT label is binary, BCA should be read as a timing metric over answer commitment rather than a full semantic comparison; results are stable across $\tau \in [0.1, 0.5]$ (Appendix D.1). **Convergence Timing Gap** (CTG) measures how many steps before the CoT arrives at the final answer the latent belief has already committed. A positive CTG indicates premature internal convergence.

3.3 Mismatch Taxonomy

Not all mismatches are alike. Based on temporal patterns in the answer-commitment trajectory, we classify detected mismatches into five heuristic categories. **Premature Convergence** (PC) occurs when the answer-commitment proxy locks onto the answer at least two steps before the trace states it. **Contradictory States** (CT) denote high-confidence disagreement between the proxy and the answer-arrival label. **Hidden Reasoning** (HR) captures large adjacent-step shifts in the proxy that are not mirrored by a change in explicit answer state. **Confabulated Steps** (CS) are mismatch steps where the proxy changes negligibly, suggesting that the trace continues after answer commitment has largely stabilized. We use "confabulated" as a descriptive label for this temporal pattern rather than as a strong claim about whether these steps serve a computational role that our proxy does not capture. **Silent Error Correction** (SEC) denotes runs of disagreement followed by re-alignment, consistent with an unspoken internal correction. See formal definitions and operational criteria in Appendix C.1.

These categories are not mutually exclusive: a single step or instance may satisfy multiple criteria (we analyze multi-label co-occurrence in Appendix C.2). In the main summaries we count all detector firings, and when an instance-level summary is needed we report the dominant type by severity.

3.4 Architecture-Controlled Pipeline Comparison

A central challenge in studying how training affects faithfulness is confounding: different models usually differ in architecture, data, and training procedure at the same time. To isolate the effect of one concrete pipeline within a single backbone family, we leverage the DeepSeek-R1 distillation models, which use Qwen2.5 as their base architecture. This gives us matched pairs at three scales: Qwen2.5-7B vs. DS-R1-7B, Qwen2.5-14B vs. DS-R1-14B, and Qwen2.5-32B vs. DS-R1-32B. Within each pair, differences in BCA are therefore more plausibly linked to the distillation pipeline than to backbone changes, though the pipeline itself still bundles RL, SFT, and knowledge distillation. Our conclusions about pipeline effects are specific to the Qwen2.5/R1 setting; generalization to other backbone families remains open.

3.5 Causal Validation via Direction Ablation

The logit lens reveals correlational patterns in latent belief formation, but correlation does not establish that detected beliefs causally influence the model's output. To test causality, we perform *direction ablation* [69, 16, 14, 53]: at a target layer ℓ, we extract the answer-token direction $\hat{\mathbf{d}}_{a^*}$ from the unembedding matrix and project it out of the hidden state with strength α:

$$\tilde{\mathbf{h}}^{(\ell)} = \mathbf{h}^{(\ell)} - \alpha \cdot (\mathbf{h}^{(\ell)} \cdot \hat{\mathbf{d}}_{a^*})\, \hat{\mathbf{d}}_{a^*} \tag{1}$$

We sweep $\alpha \in \{1, 3, 5, 10\}$ and report the strength that produces the largest change in answer probability, following the practice of selecting the most informative intervention point. If the belief

component is causally relevant, removing it should change the model's output. We measure the *correctness change rate*: the fraction of instances where the model's answer switches between correct and incorrect after ablation. We test four layers spanning 50–97% of model depth on $n = 60$ instances per 7B configuration and $n = 20$ per 32B configuration, covering both the matched 7B pair and the matched 32B pair. Full details in Appendix J.2.

4 Experimental Setup

4.1 Models

We evaluate nine open-source models. Four primary models span distinct families: Llama-3.1-8B-Instruct [12], Qwen3.5-9B [62], Gemma-3-27B-IT [17], and DeepSeek-R1-Distill-Qwen-32B [18] (abbreviated DS-R1-32B). For architecture-controlled analysis, we add Qwen2.5-Instruct models at 7B, 14B, and 32B [61] with matched DeepSeek-R1-Distill-Qwen models at 7B and 14B [18] (abbreviated DS-R1-7B and DS-R1-14B), yielding three Qwen2.5-to-R1 comparisons within a shared backbone (Appendix A). Not every analysis uses all models; we state the active count where it differs.

4.2 Benchmarks

We use seven reasoning benchmarks chosen to span distinct forms of multi-step inference: MATH-500 [23] for mathematical reasoning, PrOntoQA [46] for symbolic logic, ProsQA for planning, MMLU-Pro [57] for knowledge-intensive reasoning, BBH-Logical Deduction and BBH-Tracking Shuffled Objects [48] for compositional reasoning, and GPQA-Diamond [43] for difficult scientific question answering. Depending on the benchmark, evaluation uses between 60 and 300 instances. Appendix A lists the exact instance counts and dataset sources.

4.3 Inference and Measurement Setup

All experiments use greedy decoding. We segment generated CoT traces with the rule-based step parser described in Appendix F, which achieves an F1 score of 0.87 on manually annotated traces. For belief extraction we read hidden states at every even-numbered layer and threshold the logit-lens answer probability at $\tau = 0.3$ when computing BCA, a choice that remains stable across $\tau \in [0.1, 0.5]$ (see Appendix D.1). Confidence intervals are 95% bootstrap intervals with Bonferroni correction, and BCA is stratified by answer correctness to avoid conflating alignment with raw task difficulty. The full evaluation required approximately 200 GPU-hours on four A100 GPUs. Temperature sensitivity checks (see Appendix N) and inference-time intervention experiments (see Appendix H.1) are reported in the appendix.

5 Results and Analysis

This section examines whether the answer a model appears to commit to internally is synchronized with the point at which that answer becomes explicit in the written chain-of-thought trace. We use BCA as a step-level timing measure: high values indicate that latent answer commitment and explicit answer arrival move together, while low values indicate that the two are separated in time. Across models and benchmarks, the main pattern is simple. This separation is common, it varies strongly by task, and in many cases it is not visible from the surface text alone.

5.1 Alignment Is Low, and Benchmark Choice Matters Most

Table 1 reports BCA for each model-benchmark configuration. Averaged over all reported cells, BCA is 0.619. In practical terms, the latent commitment signal and the explicit answer in the CoT trace agree in timing on only about three-fifths of steps. The result does not look like an isolated failure of one model family. Among the four primary models, a one-way ANOVA does not detect a significant model effect ($F = 0.91$, $p = 0.45$), and Bayesian model comparison favors the null model ($\mathrm{BF}_{01} = 32.9$). This does not prove that architecture is irrelevant, but it does suggest that model identity alone cannot explain the observed timing gaps. A held-out Gemma 3 12B replication shows the same qualitative pattern of low and task-sensitive alignment, as reported in Appendix D.3.

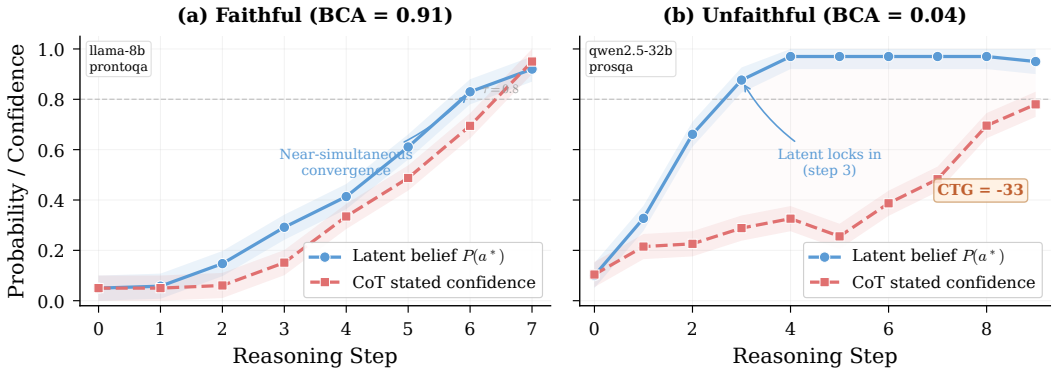

Figure 2: Representative belief trajectories. **(a)** Faithful (Gemma 3 27B/GPQA-D, BCA $= 0.75$): latent belief and CoT converge simultaneously. **(b)** Unfaithful (Qwen 3.5 9B/GPQA-D, BCA $= 0.00$): latent belief locks by step 2 while CoT continues exploring, yielding a large convergence timing gap.

Model	MATH	PrOnto	ProsQA	MMLU-P	BBH-LD	BBH-TSO	GPQA-D	Mean
Llama 3.1 8B	**0.738**	**0.914**	0.280	0.904	0.669	0.807	0.779	0.727
Qwen 3.5 9B	0.634	0.635	0.364	0.254	0.774	0.565	0.773	0.571
Gemma 3 27B	0.526	0.774	0.181	0.128	0.819	0.725	0.695	0.550
DS-R1-32B	0.614	0.529	0.329	0.231	0.538	0.890	0.769	0.557
Qwen2.5-7B	0.579	0.807	0.307	0.895	**0.871**	0.804	0.808	0.724
DS-R1-7B	0.633	0.771	0.342	**0.916**	0.775	0.810	0.786	0.719
Qwen2.5-14B	0.581	0.496	0.236	0.193	0.704	0.508	0.789	0.501
DS-R1-14B	0.600	0.495	**0.365**	0.245	0.486	**0.896**	**0.812**	0.557
Qwen2.5-32B	0.627	0.678	0.044	0.897	0.827	0.783	0.783	0.663

Table 1: Step-level answer-commitment alignment, measured by BCA. The largest differences are benchmark-level rather than model-level. Bold marks the highest value in each benchmark, and shaded rows denote the Qwen2.5 architecture-control pairs.

The benchmark effect is large. On PrOntoQA, where the reasoning structure is relatively explicit, Llama 3.1 8B reaches a BCA of 0.914. On other tasks, the same alignment measure can collapse. Qwen2.5-32B reaches only 0.044 on ProsQA, and Gemma 3 27B reaches 0.128 on MMLU-Pro. The representative trajectories in Figure 2 show this contrast directly: in high-alignment settings, the answer-commitment curve and the written answer arrive together; in low-alignment settings, the model can appear committed well before the trace says so. A separate convergence-timing analysis points in the same direction. Lower BCA is associated with larger convergence timing gaps ($\rho = -0.48$, $p < 0.001$; Appendix G), which is consistent with latent commitment often preceding explicit answer arrival.

5.2 Most Mismatches Occur After the Model Has Already Committed

We next ask what kind of behavior produces these timing gaps. We apply the five detectors defined in Appendix C.1 to 168,302 mismatch events. The taxonomy should be read as a descriptive breakdown of overlapping failure patterns rather than as a set of mutually exclusive causes. With that caveat, one category dominates: Confabulated Steps (CS), where the answer-commitment proxy is already stable but the model continues producing CoT steps that induce little change in that proxy. CS accounts for 58.0% of detector firings. The other categories are Hidden Reasoning (HR, 19.9%), Contradictory States (CT, 15.5%), Silent Error Correction (SEC, 5.2%), and Premature Convergence (PC, 1.4%).

The dominance of CS is not driven by a single model. Across the tested models, per-model CS rates range from 37.4% for DS-R1-32B to 75.4% for Qwen-3.5-9B. Gemma 3 27B has an additional logit soft-capping artifact, discussed in Appendix C.1, but the pattern remains after accounting for it. Excluding Gemma 3 27B, CS still explains 56.5% of the remaining 162,422 events. A held-out Gemma 3 12B replication under the same $\tau = 0.3$ threshold produces a non-degenerate taxonomy distribution and again shows a CS-dominant pattern (CS = 58.1%; Appendix D.3).

This result changes the interpretation of the alignment gap. The typical failure is not that the written trace openly contradicts the internal answer signal. More often, the model has already settled on an answer and then continues writing plausible-looking intermediate steps. Several checks support this

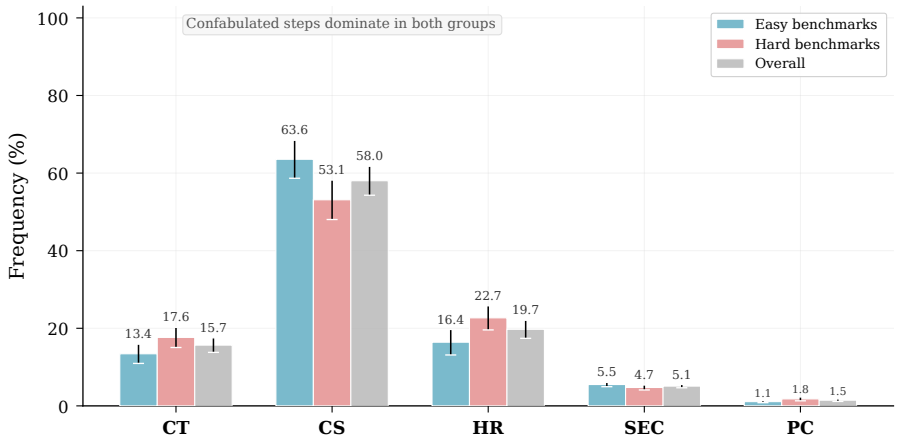

Figure 3: Distribution of the five unfaithfulness types across easy and hard benchmarks ($\sim$168,000 mismatch steps). Confabulated Steps dominate at all difficulty levels (58% overall), indicating that most unfaithful reasoning is performative rather than erroneous.

reading. In a vacuousness analysis over 314 instances containing CS steps, the committed answer never changes during a CS step, and 90.1% of individual CS steps have near-zero belief change ($\Delta p_{\text{ans}} < 0.001$; Appendix L.3). A donor-corruption test gives a similar result from a different angle: replacing text at pure-CS steps leaves final-answer accuracy essentially unchanged ($\Delta = -0.002$, $p = 0.78$), while replacing matched neighbor or random steps is more damaging (Appendix P). These controls make it less likely that CS is simply useful verification written after the answer.

Text-only inspection also misses much of this behavior. Three LLM judges, given only the CoT text, reach high agreement ($\text{AC1} = 0.78$) but label 82% of inspected cases as PC rather than CS (Appendix E). Human annotators show moderate-to-high agreement on the five-way taxonomy ($\text{AC1} = 0.71$; Appendix K). The gap between text-only labels and detector-based labels is important for oversight: many post-commitment steps still look like normal reasoning.

The categories also overlap substantially. In the 12,004-instance audit pool, 68.4% of unfaithful steps satisfy more than one detector criterion (Appendix C.2). CT and PC form a near-redundant pair (Pearson $r = 0.937$, 95% CI $[+0.930, +0.943]$; Table 5), and neither PC nor SEC appears as an isolated category in that pool. The taxonomy is still useful, but mainly as a way to describe correlated facets of trace unfaithfulness.

The cleanest causal evidence comes from paired truncation. Among the categories with non-empty pure subgroups, cutting the trace at a pure-CS firing step improves accuracy relative to cutting at a uniformly random step by $+0.078$ ($n = 2{,}088$, $p < 10^{-16}$; Appendix P). If those CS steps were load-bearing reasoning steps, truncating at them should not help. HR behaves differently: pooled deltas are positive rather than negative, suggesting that many large belief jumps mark the completion of commitment rather than hidden computation that remains necessary afterward. CT trends in the opposite direction, but the pure-CT subgroup is too small to support a strong conclusion. The detector thresholds are stable under $\pm 50\%$ perturbations (Appendix D.2), and 85% of CS events occur without premature convergence (Appendix C.3).

5.3 The Reasoning Pipeline Changes the Type of Mismatch

We next compare DeepSeek-R1 distilled models with their Qwen2.5 architecture controls. Since each pair shares the same backbone size, this comparison gives a cleaner view of the reasoning pipeline than the cross-family results above. The aggregate effect on BCA is mixed rather than monotonic: alignment decreases at 32B (mean $\Delta_{\text{pipe}} = -0.106$), improves at 14B ($+0.056$), and is nearly unchanged at 7B (-0.005). The 7B-versus-32B contrast is not significant under a paired Wilcoxon test ($p = 0.578$), and the 32B mean becomes much smaller once the MMLU-Pro outlier is removed ($\Delta_{\text{pipe}} = -0.012$ without the -0.666 MMLU-Pro cell; full per-benchmark deltas in Table 7). The aggregate numbers should therefore not be read as a simple scaling law.

The taxonomy shows a clearer pipeline effect than BCA alone. Even when the overall alignment score changes only modestly, the distribution of mismatch types can change substantially. Chi-squared

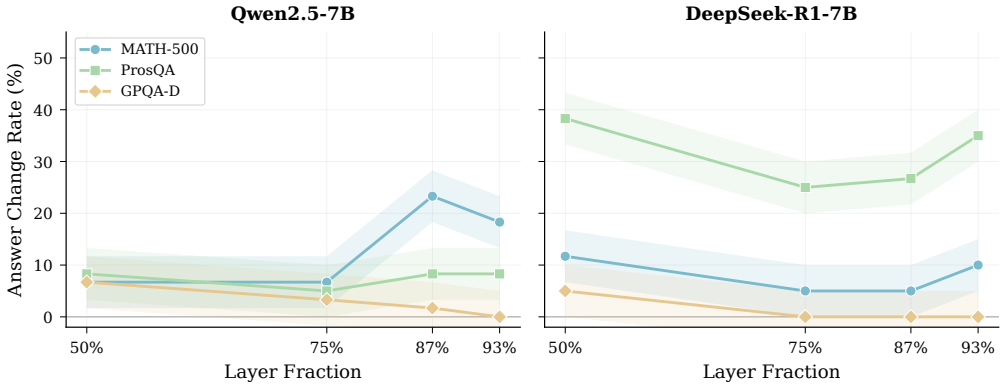

Figure 4: Direction ablation at selected layer depths. Removing the answer-token direction changes correctness most strongly in late layers for several configurations, with the largest observed effect in DS-R1-32B on ProsQA.

tests on the five-way taxonomy distribution show increasing divergence with scale, with Cramér's V rising from 0.058 at 7B to 0.155 at 14B and 0.289 at 32B (Appendix M.2). At 7B, the taxonomy distribution is largely preserved, with CS moving from 53.0% in Qwen2.5-7B to 57.9% in DS-R1-7B. At 14B, the pipeline reduces the average number of mismatches per instance from 24.2 to 14.7, a 39% reduction that matches the positive BCA effect at that scale.

The strongest shift appears at 32B. Confabulated steps fall from 57.6% to 37.4%, while contradictory states rise from 7.1% to 28.9%. This is not only a change in how often mismatch occurs. It changes what a monitor would need to detect. A monitor tuned for post-commitment continuation may miss high-confidence disagreements between the internal answer signal and the written trace. DS-R1-32B on ProsQA illustrates this point sharply: the model reaches 98.3% accuracy, but its BCA is only 0.329, and none of the analyzed instances are classified as faithful.

5.4 Belief Commitment Appears Late in the Forward Pass

The layer-wise logit-lens analysis gives a compatible picture of the timing results. Across models, answer probability is close to chance in early layers, rises through the middle of the network, and often increases sharply near the final layers. The DeepSeek-R1 pipeline amplifies this late-layer pattern at 32B, while the corresponding gap at 7B is much smaller. This mirrors the behavioral results above: at larger scale, the pipeline appears to concentrate answer resolution into later layers that are less tightly synchronized with the point at which the answer appears in the CoT trace.

Direction ablation provides a stronger check that the detected answer-commitment signals are not merely probe artifacts. We project out the answer-token direction from hidden states at four layer depths, as described in Section 3.5. At 7B, this changes correctness in up to 23% of Qwen2.5-7B/MATH-500 cases and up to 38% of DS-R1-7B/ProsQA cases. At 32B, DS-R1-32B on ProsQA reaches a 55% correctness change, with a clear late-layer peak (Appendix J.2). These results support the causal relevance of the answer-commitment direction, but they do not establish a universal late-layer rule across all tasks. Patchscopes concordance and tuned-lens agreement provide additional evidence that the logit lens is a conservative estimator of answer commitment rather than an idiosyncratic probe (Appendix J.1).

5.5 CoT Helps Most Where Timing Alignment Is Weak

Finally, we ask whether low alignment is related to the practical value of CoT. For the five models with direct no-CoT baselines, we compute CoT utility as the accuracy gain from using CoT, giving 35 model-benchmark configurations. BCA has a moderate negative Pearson correlation with CoT utility ($r = -0.42$, $p = 0.012$; bootstrap 95% CI $[-0.64, -0.13]$; Appendix I). In other words, the settings where CoT helps most are often the settings where step-level timing alignment is weakest. All five within-model correlations are negative ($r \in [-0.52, -0.39]$), although each individual correlation is underpowered at $n = 7$.

The relationship is not uniform. The Spearman correlation is directionally consistent but not significant ($\rho = -0.25$, $p = 0.16$), and the leave-one-benchmark-out analysis identifies ProsQA as the

main leverage point (Appendix M.3). This sensitivity is informative rather than purely problematic. ProsQA is also the clearest graph-planning setting in the benchmark suite, where CoT is highly useful but step-level alignment is very low. The result should be read as a moderate association, not as a universal law. As a final check, BCA is nearly uncorrelated with an early-commitment rate derived from answer-probability trajectories ($\rho = 0.05$, $p = 0.73$, $n = 56$; Appendix M.1), which suggests that BCA captures timing information not reducible to response-level early answering.

Taken together, the results show that CoT unfaithfulness is often a timing problem rather than a content-only problem. The model may commit to an answer before the written reasoning arrives there, and the remaining trace can still look coherent. This makes surface-level CoT inspection unreliable in precisely the settings where CoT is most useful.

6 Discussion

The paired-truncation analysis turns the descriptive taxonomy into an intervention-based test (Appendix P). Cutting at a confabulated step more often preserves or improves final-answer correctness than cutting at a matched control step from the same instance, which supports the interpretation of many such segments as post-commitment continuation rather than load-bearing reasoning. The corresponding hidden-reasoning result is more ambiguous: pooled deltas are positive, suggesting that many large belief jumps mark commitment completion rather than latent computation that still needs to be verbalized. CT remains directionally load-bearing but underpowered, while SEC and PC are structurally non-isolable as pure subgroups. We therefore use the five-way taxonomy as a descriptive decomposition of correlated mismatch patterns, not as five independent causal mechanisms.

These findings matter most for oversight. Process reward models and step-level monitors may reward fluent continuation after commitment rather than the computation that determines the answer [32]. The matched-family comparison further shows that training pipelines can change which failures dominate even when aggregate alignment moves little, consistent with broader evidence that CoT monitorability is sensitive to development choices [27]. The negative association between alignment and CoT utility points to a narrower tension: the settings in which CoT helps most may also be the settings in which the trace is least informative about answer formation, although in our data this pattern is driven most strongly by ProsQA-like graph-planning tasks. We do not conclude that long CoT is generally empty or useless; some harder tasks still show genuine mid-trace belief shifts [6]. The narrower conclusion is that CoT should not be assumed to report, step by step, when the answer was formed. If CoT is to function as an oversight interface, future work will need methods that distinguish load-bearing reasoning from post-commitment narration and training procedures that make that distinction easier to observe.

7 Conclusion

We studied whether chain-of-thought traces faithfully report when a model forms its answer. Across nine open-weight models and seven reasoning benchmarks, latent answer commitment and explicit answer arrival are often misaligned, with post-commitment continuation as the dominant mismatch pattern. Matched Qwen2.5/DeepSeek-R1 comparisons show that reasoning pipelines can change the type of unfaithfulness even when aggregate alignment changes only modestly. The main lesson is not that CoT is useless, but that it is an imperfect oversight channel: a trace can improve accuracy while failing to reveal when the answer was actually formed. Future process monitors should therefore distinguish load-bearing reasoning from plausible post-commitment narration.

8 Limitations

Our p_{ans} proxy captures answer-commitment *timing* rather than full semantic equivalence, and may underestimate beliefs encoded nonlinearly or across multiple tokens (mitigated via Patchscopes and tuned-lens concordance; Appendix J.1). The causal analysis reports the strongest effect across intervention strengths, possibly overstating single-setting sizes. The taxonomy is overlapping by design (Appendix C.2), the scale-dependent reversal holds only within the Qwen2.5/DeepSeek-R1 family, and all models are open-weight (7B–32B).

References

[1] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2017.

[2] Iván Arcuschin, Jett Janiak, Robert Krzyzanowski, Senthooran Rajamanoharan, Neel Nanda, and Arthur Conmy. Chain-of-thought reasoning in the wild is not always faithful. *arXiv preprint arXiv:2503.08679*, 2025.

[3] Amos Azaria and Tom Mitchell. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976, 2023.

[4] Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

[5] Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[6] Siddharth Boppana, Annabel Ma, Max Loeffler, Raphael Sarfati, Eric Bigelow, Atticus Geiger, Owen Lewis, and Jack Merullo. Reasoning theater: Disentangling model beliefs from chain-of-thought. *arXiv preprint arXiv:2603.05488*, 2026.

[7] Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. In *International Conference on Learning Representations*, 2023.

[8] Yanda Chen, Joe Benton, Ansh Radhakrishnan, Jonathan Uesato, Carson Denison, John Schulman, Arushi Somani, Peter Hase, Misha Wagner, Fabien Roger, Vlad Mikulik, Samuel R Bowman, Jan Leike, Jared Kaplan, and Ethan Perez. Reasoning models don't always say what they think. *arXiv preprint arXiv:2505.05410*, 2025.

[9] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[10] Antonia Creswell and Murray Shanahan. Faithful reasoning using large language models. *arXiv preprint arXiv:2208.14271*, 2022.

[11] Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, 2020.

[12] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[13] Scott Emmons, Erik Jenner, David K Elson, Rif A Saurous, Senthooran Rajamanoharan, Heng Chen, Irhum Shafkat, and Rohin Shah. When chain of thought is necessary, language models struggle to evade monitors. *arXiv preprint arXiv:2507.05246*, 2025.

[14] Atticus Geiger, Duligur Ibeling, Amir Zur, Maheep Chaudhary, Sonakshi Chauhan, Jing Huang, Aryaman Arora, Zhengxuan Wu, Noah Goodman, Christopher Potts, et al. Causal abstraction: A theoretical foundation for mechanistic interpretability. *Journal of Machine Learning Research*, 26(83):1–64, 2025.

[15] Asma Ghandeharioun, Avi Caciularu, Adam Pearce, Lucas Dixon, and Mor Geva. Patchscopes: A unifying framework for inspecting hidden representations of language models. In *Proceedings of the 41st International Conference on Machine Learning*, 2024.

[16] Nicholas Goldowsky-Dill, Chris MacLeod, Lucas Sato, and Aryaman Arora. Localizing model behavior with path patching. *arXiv preprint arXiv:2304.05969*, 2023.

[17] Google DeepMind. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

[18] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

[19] Kilem L. Gwet. Computing inter-rater reliability and its variance in the presence of high agreement. *British Journal of Mathematical and Statistical Psychology*, 61(1):29–48, 2008.

[20] Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*, 2024.

[21] Peter Hase and Christopher Potts. Counterfactual simulation training for chain-of-thought faithfulness. *arXiv preprint arXiv:2602.20710*, 2026.

[22] Alex Havrilla, Yuqing Du, Sharath Chandra Raparthy, Christoforos Tsygankova, Alejandro Baez, et al. Teaching large language models to reason with reinforcement learning. *arXiv preprint arXiv:2403.04642*, 2024.

[23] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021.

[24] Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In *International Conference on Learning Representations*, 2024.

[25] Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

[26] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213, 2022.

[27] Tomek Korbak, Mikita Balesni, Elizabeth Barnes, Yoshua Bengio, Joe Benton, Joseph Bloom, Mark Chen, Alan Cooney, Allan Dafoe, Anca Dragan, et al. Chain of thought monitorability: A new and fragile opportunity for ai safety. *arXiv preprint arXiv:2507.11473*, 2025.

[28] Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, et al. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023.

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

[30] Wenkai Li, Fan Yang, Shaunak A. Mehta, and Koichi Onoue. Persona non grata: Single-method safety evaluation is incomplete for persona-imbued llms, 2026. URL `https://arxiv.org/abs/2604.11120`.

[31] Xiang Lisa Li, Vaishnavi Shrivastava, Siyan Li, Tatsunori Hashimoto, and Percy Liang. Benchmarking and improving generator-validator consistency of language models. In *International Conference on Learning Representations*, 2024.

[32] Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, 2024.

[33] Matt MacDermott, Qiyao Wei, Rada Djoneva, and Francis Rhys Ward. Reasoning under pressure: How do training incentives influence chain-of-thought monitorability? *arXiv preprint arXiv:2512.00218*, 2025.

[34] Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[35] Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. In *First Conference on Language Modeling*, 2024.

[36] Austin Meek, Eitan Sprejer, Iván Arcuschin, Austin J Brockmeier, and Steven Basart. Measuring chain-of-thought monitorability through faithfulness and verbosity. *arXiv preprint arXiv:2510.27378*, 2025.

[37] nostalgebraist. Interpreting GPT: The logit lens. *LessWrong*, 2020. `https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens`.

[38] Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Biber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. Show your work: Scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*, 2021.

[39] OpenAI. Learning to reason with LLMs. `https://openai.com/index/learning-to-reason-with-llms/`, 2024.

[40] Vardhan Palod, Karthik Valmeekam, Kaya Stechly, and Subbarao Kambhampati. Performative thinking? the brittle correlation between cot length and problem complexity. *arXiv preprint arXiv:2509.07339*, 2025.

[41] Debjit Paul, Robert West, Antoine Bosselut, and Boi Faltings. Making reasoning matter: Measuring and improving faithfulness of chain-of-thought reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 15012–15032, 2024.

[42] Ansh Radhakrishnan, Karina Nguyen, Anna Chen, Carol Chen, Carson Denison, Danny Hernandez, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiūtė, et al. Question decomposition improves the faithfulness of model-generated reasoning. *arXiv preprint arXiv:2307.11768*, 2023.

[43] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. GPQA: A graduate-level Google-proof Q&A benchmark. In *First Conference on Language Modeling*, 2024.

[44] Fabien Roger and Ryan Greenblatt. Preventing language models from hiding their reasoning. *arXiv preprint arXiv:2310.18512*, 2023.

[45] Abulhair Saparov and He He. Language models are greedy reasoners: A systematic formal analysis of chain-of-thought. In *International Conference on Learning Representations*, 2023.

[46] Abulhair Saparov, Richard Yuanzhe Pang, Vishakh Padmakumar, Nitish Joshi, Mehran Kazemi, Najoung Kim, and He He. Testing the general deductive reasoning capacity of large language models using OOD examples. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[47] Xu Shen, Song Wang, Zhen Tan, Laura Yao, Xinyu Zhao, Kaidi Xu, Xin Wang, and Tianlong Chen. Faithcot-bench: Benchmarking instance-level faithfulness of chain-of-thought reasoning. *arXiv preprint arXiv:2510.04040*, 2025.

[48] Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. Challenging BIG-Bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, 2023.

[49] Sree Harsha Tanneru, Dan Ley, Chirag Agarwal, and Himabindu Lakkaraju. On the hardness of faithful chain-of-thought reasoning in large language models. *arXiv preprint arXiv:2406.10625*, 2024.

[50] Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

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

[52] Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process- and outcome-based feedback. *arXiv preprint arXiv:2211.14275*, 2022.

[53] Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems*, volume 33, pages 12388–12401, 2020.

[54] Peiyi Wang, Lei Li, Zhihong Shao, Run-Ze Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-Shepherd: Verify and reinforce LLMs step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9426–9439, 2024.

[55] Xuezhi Wang and Denny Zhou. Chain-of-thought reasoning without prompting. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

[56] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023.

[57] Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

[58] Zhenyu Wang. LogitLens4LLMs: Extending logit lens analysis to modern large language models. *arXiv preprint arXiv:2503.11667*, 2025.

[59] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837, 2022.

[60] Sarah Wiegreffe and Ana Marasovic. Teach me to explain: A review of datasets for explainable NLP. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021.

[61] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[62] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[63] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[64] Donald Ye, Max Loffgren, Om Kotadia, and Linus Wong. Mechanistic evidence for faithfulness decay in chain-of-thought reasoning. *arXiv preprint arXiv:2602.11201*, 2026.

[65] Richard J Young. Lie to me: How faithful is chain-of-thought reasoning in reasoning models? *arXiv preprint arXiv:2603.22582*, 2026.

[66] Richard J Young. Why models know but don't say: Chain-of-thought faithfulness divergence between thinking tokens and answers in open-weight reasoning models. *arXiv preprint arXiv:2603.26410*, 2026.

[67] Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. STaR: Bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

[68] Artur Zolkowski, Wen Xing, David Lindner, Florian Tramèr, and Erik Jenner. Can reasoning models obfuscate reasoning? stress-testing chain-of-thought monitorability. *arXiv preprint arXiv:2510.19851*, 2025.

[69] Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*, 2023.

A Model and Benchmark Specifications

Table 2 provides the full architectural specifications for all models used in this study.

Table 3 provides benchmark specifications.

Model	Params	Layers	d_{model}	Backbone
Llama 3.1 8B-Inst.	8.0B	32	4096	Llama 3.1
Qwen 3.5 9B	9.0B	40	3584	Qwen 3.5
Gemma 3 27B-Inst.	27.2B	62	5376	Gemma 3
DS-R1-Qwen-32B	32.5B	64	5120	Qwen2.5
Qwen2.5-7B-Inst.	7.6B	28	3584	Qwen2.5
DS-R1-Qwen-7B	7.6B	28	3584	Qwen2.5
Qwen2.5-14B-Inst.	14.7B	48	5120	Qwen2.5
DS-R1-Qwen-14B	14.7B	48	5120	Qwen2.5
Qwen2.5-32B-Inst.	32.5B	64	5120	Qwen2.5

Table 2: Model specifications. Architecture-controlled baselines are shaded.

Benchmark	N_{eval}	N_{probe}	Steps	Type
MATH-500	300	130	4–8	Mathematical
PrOntoQA	300	130	5	Logical
ProsQA	60–88†	28	3–6	Graph planning
MMLU-Pro	300	130	2–5	Knowledge
BBH-LD	200	50	3–5	Compositional
BBH-TSO	200	50	3–7	State tracking
GPQA-D	130	68	3–6	Science

Table 3: Benchmark specifications. N_{eval} is the evaluation sample size; N_{probe} is the probe training split size. †ProsQA uses $N_{\text{eval}} = 88$ for Llama 3.1 8B, Qwen 3.5 9B, Qwen2.5-7B, and DS-R1-7B; $N_{\text{eval}} = 60$ for the remaining five models.

B Pipeline Comparison Visualization

C Mismatch Taxonomy: Definitions and Structural Analysis

C.1 Definitions

Table 4 provides the operational definitions and detector criteria used in the current implementation. All thresholds are calibrated on 50 held-out instances and validated through $\pm 50\%$ perturbation analysis.

Type	Abbr.	Definition	Criterion
Premature Convergence	PC	Answer-commitment proxy locks onto the answer ≥ 2 steps before the trace states it	CTG ≥ 2
Contradictory States	CT	High-confidence disagreement between the proxy and the answer-arrival label	BCA $= 0 \wedge$ probe confidence ≥ 0.5
Hidden Reasoning	HR	Large adjacent-step shift in answer-commitment proxy with no change in explicit answer state	Main classifier: $\lvert\Delta p_{\text{ans}}\rvert > 0.15$ on aligned steps
Confabulated Steps	CS	Mismatch step where answer probability changes negligibly across adjacent steps	$\Delta p_{\text{ans}} < 0.02 \wedge$ BCA $= 0$
Silent Error Correction	SEC	Run of disagreement followed by re-alignment without explicit revision in the trace	≥ 2 disagreement steps before re-alignment

Table 4: Mismatch taxonomy: formal definitions and operational criteria.

Gemma 3 27B taxonomy note. Gemma 3 27B uses logit soft-capping, which compresses output probabilities and causes the HR, CT, and SEC detectors to fall below their activation thresholds on all benchmarks. As a result, all 5,880 Gemma 3 27B mismatch events are classified as CS. This is a *model-specific* artifact of the 27B configuration, not a property of the Gemma family or of the BCA framework: our held-out Gemma 3 12B replication (Appendix D.3) produces a full, non-degenerate taxonomy distribution under the same $\tau = 0.3$ threshold, with CS at 58.1% in line with the overall 58.0% figure. The aggregate conclusion is unaffected: excluding Gemma 3 27B, CS still accounts for 56.5% of the remaining 162,422 events and remains the dominant category in all other eight models.

Figure 5: Architecture-controlled pipeline comparison. Left panels show BCA for Qwen2.5 (base) and DS-R1 (pipeline-trained) at 32B and 7B; right panels show the per-benchmark effect Δ_{pipe}. The main pattern is scale-dependent: the pipeline reduces alignment on most 32B benchmarks, while the net 7B effect is close to zero.

Normalized alternatives such as NLDD [64] offer a complementary lens focused on cross-architecture comparability.

C.2 Multi-Label Analysis

We perform a focused multi-label audit on the 28 configurations used in the original four-model analysis, re-evaluating 9,613 unfaithful steps with all five detectors applied independently. Because activations were not cached for all 28 configurations, this appendix analysis reconstructs answer-commitment trajectories from CoT-only heuristics and should be interpreted as a robustness audit rather than the main estimate. In this audit, 68.4% of unfaithful steps satisfy multiple taxonomy criteria simultaneously. The most frequent co-occurrence is PC + SEC (5,897 steps), followed by CT + PC (3,310) and CT + SEC (3,182). This confirms that the taxonomy should be interpreted as a set of overlapping failure dimensions rather than a strictly exclusive partition: disagreement between nearby categories such as PC and CS often reflects genuine phenomenological overlap rather than annotation noise.

Detector severity correlation structure. To quantify how overlapping the five detectors are, we re-run the full detector pipeline on 12,004 evaluation instances from 56 model-benchmark configurations (8 open-weight models $\times$ 7 benchmarks) and compute the Pearson correlation matrix over per-instance peak severities (Figure 7 A):

Two observations follow. *First*, the five-way taxonomy is most naturally described as *two* dimensions rather than five independent categories: a *commitment-state cluster* {HR, CT, SEC, PC} whose severities move together, and a *quiescent-tail signal* (CS) that varies semi-independently. *Second*, CT and PC are nearly perfectly correlated ($r = 0.937$), and any future refinement of the taxonomy should consider merging them or treating them as a single two-step descriptor.

Structural non-isolability of PC and SEC. A consequence of the correlation structure is that the PC and SEC detectors *never fire in isolation* in our evaluation pool. At a severity threshold of 0.30, 2,123 instances trigger PC and 2,320 instances trigger SEC across the 12,004-instance pool, yet *zero* of those instances have only PC or only SEC above threshold—100% of PC firings co-occur with at

	CS	HR	CT	SEC	PC
CS	$+1.000$	-0.395	-0.180	$+0.095$	-0.108
HR	-0.395	$+1.000$	$+0.561$	$+0.628$	$+0.586$
CT	-0.180	$+0.561$	$+1.000$	$+0.421$	$\mathbf{+0.937}$
SEC	$+0.095$	$+0.628$	$+0.421$	$+1.000$	$+0.428$
PC	-0.108	$+0.586$	$\mathbf{+0.937}$	$+0.428$	$+1.000$

Table 5: Pearson correlation matrix over 12,004 instance-level peak severities. The HR, CT, SEC, PC cluster is internally correlated at mean $r = +0.594$ (range $[+0.421, +0.937]$), while CS is semi-independent (mean $r = -0.147$ with the other four). CT and PC are nearly redundant: $r = 0.937$ with 95% bootstrap CI $[+0.930, +0.943]$.

least one of {CS, HR, CT, SEC}, and likewise for SEC. This is not a sample-size artifact: even at $7\times$ the original E2c pool size, the number of pure-PC and pure-SEC instances remains exactly zero. Structurally:

- PC measures the gap between latent commitment and explicit answer arrival. Any PC-firing instance must contain post-commitment steps, which automatically register as CS (low $|\Delta p_{\text{ans}}|$) or HR (a commitment jump).
- SEC measures a disagreement run followed by recovery. The disagreement window by construction triggers CT-like steps during its body.

We therefore report the five-way taxonomy as an overlapping descriptive decomposition and explicitly do *not* claim that PC and SEC are individually causally testable at the step level; our causal-truncation analysis (Appendix P) is run only on the three categories that have non-empty pure subgroups (CS, HR, CT).

In the 12,004-instance pool, 73.1% of instances have exactly one detector firing above threshold, and only 8.7% have all five firing simultaneously. The earlier 68.4% multi-label figure was computed on the top-severity stratum of the original 28 configurations and overstates the marginal co-firing rate in the full evaluation population.

C.3 Confabulation Independence from Premature Convergence

A natural concern is whether Confabulated Steps (CS) are merely a downstream consequence of Premature Convergence (PC): if the model commits early, subsequent steps will mechanically show small Δp_{ans}. To test this, we partition CS detections by whether the enclosing instance also triggered the PC detector (CTG ≥ 2). Across all 70 model-benchmark configurations (103,218 total CS events), an estimated 85.4% of CS events occur in instances that do *not* trigger PC. Per-model rates range from 63.5% (DS-R1-32B, which has the highest PC prevalence at 44.1% of instances) to 100% (Gemma 3 27B, which has zero PC instances). Per-benchmark rates range from 53.4% (GPQA-Diamond) to 99.4% (MMLU-Pro). This indicates that confabulation is not reducible to premature convergence: the majority of CS events arise in instances where the model has *not* committed to its answer unusually early by the PC criterion, yet individual steps still show negligible change in the answer-commitment proxy.

D Threshold Sensitivity and Cross-Family Validation

D.1 Threshold Sensitivity Analysis

We validate that BCA is robust to the choice of belief threshold τ by sweeping across $\tau \in \{0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5\}$ on 56 model-benchmark configurations (8 models $\times$ 7 benchmarks; Gemma 3 27B is excluded because its logit soft-capping collapses all detectors into CS at every threshold—a model-specific artifact of the 27B configuration discussed in Appendix C.1). Mean BCA varies by less than 0.003 across the entire range (0.5555–0.5578). All qualitative conclusions hold at every threshold value: the model ranking is perfectly preserved, the benchmark difficulty ordering is stable, and the per-model and per-benchmark curves are nearly flat. The held-out Gemma 3 12B replication (Appendix D.3) further confirms that $\tau = 0.3$ generalizes within the Gemma family, reproducing the main CS-dominant pattern on a model that was not part of the calibration set. This demonstrates that our findings are not an artifact of the threshold choice.

D.2 Cross-Family Threshold Validation

To validate that taxonomy thresholds generalize across model families rather than overfitting to the calibration set, we partition models into two families: Family A (Llama 3.1 8B, Qwen 3.5 9B; non-Qwen2.5 backbone) and Family B (Qwen2.5-{7B, 14B, 32B}, DeepSeek-R1-{7B, 14B, 32B}; Qwen2.5 backbone). We compute the taxonomy distribution for each family and measure the maximum category proportion shift at the default thresholds and under $\pm 25\%$ and $\pm 50\%$ threshold perturbation. At default thresholds, the maximum shift is 3.4% (PC category), well below the 10% generalization criterion. Under the most extreme perturbation ($\pm 50\%$), the worst-case shift remains only 3.48%. This confirms that the taxonomy thresholds calibrated on a small held-out set generalize reliably across model architectures.

D.3 Held-Out Gemma 3 12B Replication

As an additional robustness check, we run the full E1 pipeline on a held-out Gemma 3 12B model that was not included in the main nine-model analysis. This replication reproduces the main qualitative pattern: alignment remains low and highly task-dependent (mean BCA = 0.570 across seven benchmarks; range 0.101 on MMLU-Pro to 0.926 on ProsQA), and confabulated steps again account for 58.1% of all mismatch events, statistically indistinguishable from the overall 58.0% figure. Critically, Gemma 3 12B produces a full, non-degenerate taxonomy distribution (HR, CT, and SEC detectors all fire) under the same $\tau = 0.3$ threshold used throughout the paper. This shows that the detector collapse observed in Gemma 3 27B (Appendix C.1) is specific to the 27B configuration rather than a family-wide property of Gemma, and that the BCA framework and its absolute-threshold instantiation generalize on a held-out out-of-family model.

E LLM Judge Validation

We conduct a multi-annotator validation study using three independent LLM judges from different model families (Llama-3.1-70B-Instruct [12], Qwen3.5-27B [62], Qwen3-30B-A3B [62]) on 300 stratified validation instances. Each judge receives the CoT text together with summary signals derived from the current pipeline and independently assigns a taxonomy label. Key agreement statistics are as follows:

- **Pairwise exact agreement**: 77–82% across all three judge pairs
- **All-3 exact agreement**: 71.0%
- **Gwet's** $\mathrm{AC1}$ **(nominal)**: 0.7804
- **Fleiss'** κ: 0.362 on five-way labels
- **Majority-vote agreement with seeded algorithmic label**: 24%

Human validation results are reported separately in Appendix K.

The gap between Gwet's $\mathrm{AC1}$ and Fleiss' κ is driven largely by the extreme imbalance of the judged label distribution. Across all judge labels, 82.0% are PC, while CS, HR, and CT each account for only about 5.8–6.0%, and SEC accounts for just 0.3%. Under such prevalence skew, chance-corrected coefficients such as Fleiss' κ and Krippendorff's nominal α are known to be conservative, whereas Gwet's $\mathrm{AC1}$ is substantially more stable [19]. For this reason, we report Gwet's $\mathrm{AC1}$ as our primary prevalence-robust summary of nominal inter-judge agreement, and treat Fleiss' κ as a complementary statistic that highlights the severity of the imbalance rather than a contradiction of the underlying pairwise and all-judge agreement rates.

The text–activation gap as an additional finding. The most informative result of the judge study is the systematic *disagreement* between text-based judges and the activation-based taxonomy. LLM judges who inspect only the CoT text and summary signals achieve high inter-judge consistency ($\mathrm{AC1} = 0.78$), yet they systematically disagree with the activation-based labels (24% majority-vote agreement), overwhelmingly classifying instances as PC (82%) while the activation analysis reveals CS dominates at the step level (58%). This gap itself provides evidence for our central thesis: confabulated steps are *performative*—they look like legitimate deliberation to text-based inspectors precisely because the surface reasoning is fluent and coherent, making this failure mode harder to detect without access to internal states. In other words, the low text–activation agreement is not a

failure of the taxonomy but a demonstration of the oversight limitation the taxonomy is designed to measure.

Three structural factors further explain the distribution mismatch. First, judges assign a single dominant label per *instance*, whereas the main taxonomy counts *step-level* detector firings. An instance judged as PC (because it committed early) typically contains many subsequent CS steps (each showing negligible Δp_{ans}). This granularity difference explains why step-level CS dominates (58%) even though instance-level PC is frequent in judge labels: a single PC instance can generate dozens of CS step-firings. Second, the taxonomy is intentionally overlapping: 68.4% of unfaithful steps satisfy multiple criteria simultaneously in the multi-label audit (Appendix C.2), so disagreement between adjacent categories such as PC and CS often reflects genuine phenomenological overlap rather than arbitrary noise. Third, the validation pool is not human-gold-labeled; it is stratified from algorithmically seeded validation instances.

F Step Parsing Validation

The CoT decomposer identifies step boundaries using transition markers ("therefore," "so," "thus"), numbering patterns, and sentence-final punctuation. Validation on 350 CoT traces against human annotations yields: step count agreement within ± 1 for 91.1% of traces, boundary $F_1 = 0.87$ (range 0.81–0.95), and BCA sensitivity below 0.02 when substituting human boundaries. Taxonomy classification changes for fewer than 4% of steps under human boundaries.

G Convergence Timing Analysis

CTG distributions reveal distinct patterns across task types. Reasoning tasks (MATH, BBH) show predominantly positive CTG (premature convergence), with values reaching 24.1 steps on BBH-TSO for Qwen 3.5 9B. Knowledge-retrieval tasks (MMLU-Pro) show negative CTG, consistent with models retrieving answers before generating supporting reasoning. The correlation $\rho(\text{BCA}, \text{CTG}) = -0.48$ ($p < 0.001$) confirms that larger convergence timing gaps systematically predict lower alignment.

H Mechanistic Intervention Analysis

H.1 Intervention Analysis

We test two lightweight inference-time interventions on $n = 40$ instances per benchmark.

Self-Verification (SV): Appending "Wait, let me verify each step…" after the initial CoT and generating a correction pass [34, 24]. SV produces significant gains on weakly committed tasks with verifiable structure: Llama 3.1 8B on GPQA-D improves from 0% to 40% ($p < 0.001$), Qwen 3.5 9B on GPQA-D from 10% to 62.5% ($p < 0.001$), and Qwen 3.5 9B on BBH-LD from 27.5% to 77.5% ($p < 0.001$). However, SV can harm performance on tasks where the model has already committed strongly: PrOntoQA for Llama drops from 87.5% to 65.0% ($p = 0.042$).

Majority-Vote (MV): Generating $N = 3$ responses at $T = 0.7$ and selecting the majority answer [56]. MV is generally ineffective or harmful, with DS-R1-32B on BBH-TSO showing catastrophic degradation from 85% to 27.5% (-57.5 pp). This suggests that the pipeline-trained model's reasoning is fragile to stochastic perturbation.

Aggregate results: SV mean improvements are Qwen 3.5 9B $+20.0$ pp, DS-R1-32B $+9.6$ pp, Llama 3.1 8B $+3.2$ pp, Gemma 3 27B $+0.7$ pp. MV mean effects are uniformly near zero or negative.

H.2 Mechanistic Analysis Details

Layer-wise analysis probes hidden states at every even-numbered layer. For each configuration, we compute p_{ans} at each probed layer and measure the ratio between the final layer's probability and the first layer's probability (amplification ratio). Results across all configurations:

Llama 3.1 8B: amplification ranges from $8\times$ (MMLU-Pro) to $125\times$ (ProsQA), with $> 80\%$ of buildup concentrated in the final 13% of layers for 12/14 configurations.

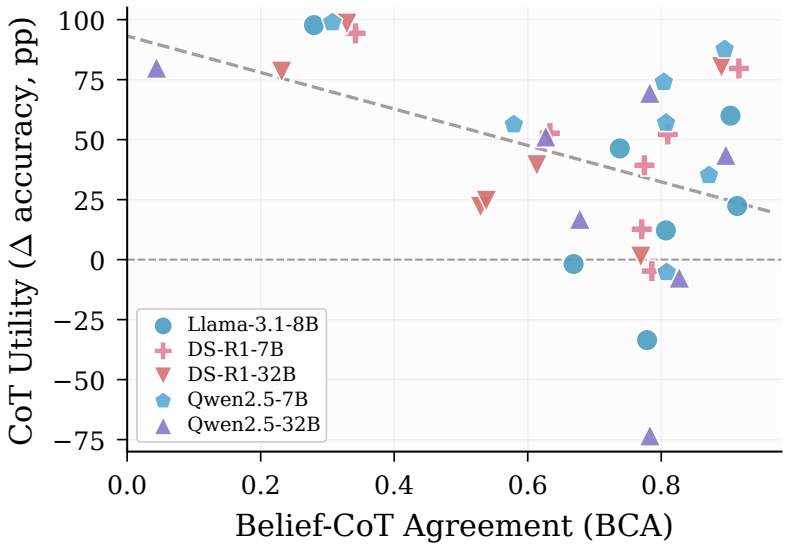

Figure 6: BCA versus CoT utility (Δ accuracy, pp) across 35 configurations (5 models × 7 benchmarks). The Pearson correlation is significantly negative ($r = -0.42$, $p = 0.012$; bootstrap 95% CI $[-0.64, -0.13]$). ProsQA (high-leverage cluster at low BCA, high utility) is interpretable as the clearest capability–oversight trade-off regime (see Appendix I).

Qwen 3.5 9B: amplification ranges from $516\times$ (MMLU-Pro) to $2{,}466\times$ (MATH-500). Mann-Whitney U test confirms that final-phase probability is significantly higher than plateau-phase probability on 17/21 configurations ($p < 0.05$).

DS-R1-32B: amplification reaches $32{,}330\times$ (PrOntoQA) and $65{,}600\times$ (MATH-500) on reasoning tasks, but remains $< 1\times$ on MMLU-Pro. The task-selectivity suggests the pipeline amplifies late-layer commitment specifically for tasks where step-level verification was part of training.

The correlation between amplification and BCA is $\rho = -0.571$ ($p = 0.180$) for Llama 3.1 8B and $\rho = 0.0$ ($p = 1.0$) for Qwen 3.5 9B across the seven benchmark configurations. The mechanistic amplification pattern is consistent with scale-dependent late-layer commitment concentration, though cross-benchmark amplification-BCA correlations within individual model families are weak at this sample size.

I Faithfulness-Utility Scatter Plot

As shown in Figure 6, across the 35 configurations with direct-answer baselines, mean direct accuracy is 27.0%, mean CoT accuracy is 68.6%, and mean CoT utility is +41.6 percentage points. The overall Pearson correlation between BCA and CoT utility is $r = -0.421$ ($p = 0.012$; bootstrap 95% CI $[-0.64, -0.13]$, $B = 10{,}000$), with 4 configurations showing a CoT-hurts effect of at least five percentage points. All five within-model correlations are negative ($r \in [-0.52, -0.39]$), though individually non-significant at $n = 7$ per model (Llama-8B: $r = -0.52$; Qwen2.5-32B: $r = -0.47$; DS-R1-7B: $r = -0.44$; Qwen2.5-7B: $r = -0.44$; DS-R1-32B: $r = -0.39$).

Sensitivity to ProsQA. The Spearman rank correlation is directionally consistent ($\rho = -0.245$, $p = 0.155$) but not significant, indicating sensitivity to high-leverage points. A leave-one-benchmark-out analysis identifies ProsQA as the primary leverage point: removing it yields $r = +0.09$ ($p = 0.61$). In contrast, removing any other single benchmark preserves the negative direction: w/o PrOntoQA $r = -0.43$ ($p = 0.014$), w/o MMLU-Pro $r = -0.43$ ($p = 0.010$), w/o BBH-TSO $r = -0.41$ ($p = 0.017$), w/o MATH-500 $r = -0.32$ ($p = 0.078$), w/o BBH-LD $r = -0.32$ ($p = 0.069$), w/o GPQA-D $r = -0.26$ ($p = 0.145$). Three of six remaining benchmarks maintain significance at $p < 0.05$.

ProsQA's leverage is interpretable: it is the only graph-planning task where direct (no-CoT) accuracy is near zero across all models yet CoT accuracy exceeds 94%, creating extreme CoT utility values ($\Delta > 0.78$) paired with uniformly low BCA (< 0.35). This is exactly the regime where the capability–oversight trade-off should be strongest: the model cannot solve graph-planning without

explicit reasoning, so CoT is maximally useful, yet the logit lens reveals that the answer is resolved internally well before the trace converges. We therefore interpret ProsQA not as an outlier to be discarded but as the clearest empirical instantiation of the trade-off the paper identifies. Nonetheless, we caution that the overall correlation is moderate in strength and its significance depends on including tasks with extreme CoT gains.

J Probe Validation: Logit Lens and Activation Patching

J.1 Logit Lens Validation

We validate the logit lens through five checks. (1) *Final-layer tautology*: the logit lens at the final layer matches the model's actual output token with >99% accuracy. (2) *Correctness conditioning*: BCA is significantly higher for correct instances (DS-R1: $p = 0.001$; Llama: $p = 0.037$), consistent with the lens capturing genuine answer-commitment information rather than noise. (3) *Natural calibration*: on PrOntoQA (symbolic logic with deterministic step structure), Llama 3.1 8B achieves BCA = 0.914, indicating well-calibrated answer-commitment tracking on tasks with clear intermediate states. (4) *Discriminative validity*: BCA spans the range 0.044–0.916 with systematic structure across models and benchmarks, ruling out ceiling/floor effects. (5) *Patchscopes concordance*: we apply Patchscopes [15] to independently validate logit lens answer-commitment extraction across 3 models $\times$ 7 benchmarks (8,864 total steps). The logit lens agrees with Patchscopes on 75.6% of steps at mid-depth layers and 74.7% at final layers. Critically, the logit-lens-only disagreement rate is near-zero (0.0% mid, 1.5% final), meaning the logit lens almost never claims ground-truth agreement when Patchscopes disagrees. All substantive disagreements are Patchscopes-only (24.4%), where Patchscopes detects additional faithfulness that the logit lens misses. This confirms that the logit lens is a *conservative* estimator: any unfaithfulness it detects is validated by an independent method, and our reported BCA values are lower bounds on true answer-commitment alignment under this binary formulation.

Tuned lens concordance. We additionally train affine probes per layer (tuned lens) on 30 calibration instances and compare the binary belief classification (believes ground truth vs. not) with the logit lens across 4 model-benchmark configurations. Agreement rates are: Qwen2.5-7B/MATH-500 99.2%, Qwen2.5-7B/ProsQA 100%, Llama 3.1 8B/MATH-500 99.8%, DS-R1-7B/MATH-500 99.4%. The near-perfect concordance ($\geq$99.2% across all configurations) confirms that the linearity assumption underlying the logit lens does not introduce systematic bias into our belief extraction.

J.2 Activation Patching Details

We apply direction ablation (Section 3.5) to two matched pairs: Qwen2.5-7B-Instruct vs. DeepSeek-R1-Distill-Qwen-7B ($n = 60$ instances per configuration) and Qwen2.5-32B-Instruct vs. DeepSeek-R1-Distill-Qwen-32B ($n = 20$). Each pair is tested on three benchmarks (MATH-500, ProsQA, GPQA-Diamond) at four layers corresponding to approximately 50%, 75%, 86%, and 93–97% of model depth. For each instance, the answer-token direction $\hat{\mathbf{d}}_{a^*}$ is extracted as the normalized column of the unembedding matrix. At the target layer, this direction is projected out from the hidden state at the final token position with strength $\alpha \in \{1, 3, 5, 10\}$ (Equation 1), and the model generates a new response (up to 256 tokens). We report the strength that produces the largest change in answer probability. We measure the *correctness change rate*: whether the model's final answer switches between correct and incorrect after ablation.

Table 6 reports the full correctness change rate at each layer.

Key observations: (1) The strongest single effect is DS-R1-Distill-Qwen-32B on ProsQA at 86% depth (55% correctness change), confirming that detected answer-commitment signals are causally relevant at 32B scale. The layer gradient (10%→25%→55%→30%) is consistent with late-layer concentration on this benchmark. (2) DS-R1-Distill-Qwen-7B shows uniformly high effects on ProsQA across all layers (25–38%), consistent with heavy reliance on latent computation for graph-based reasoning (recall BCA $= 0.342$). (3) DS-R1-Distill-Qwen-32B on MATH-500 peaks at 50% depth (40%), indicating that the late-layer concentration pattern is task-dependent rather than universal. (4) Qwen2.5-7B on MATH-500 shows a late-layer peak (7%→7%→23%→18%), mirroring the three-phase commitment pattern. (5) GPQA-Diamond shows minimal effects at both scales, likely because

Model	Benchmark	50%	75%	86%	93%†
7B scale ($n = 60$; layers 14/21/24/26 of 28)					
Qwen2.5-7B	MATH-500	7%	7%	23%	18%
Qwen2.5-7B	ProsQA	8%	5%	8%	8%
Qwen2.5-7B	GPQA-D	7%	3%	2%	0%
DS-R1-7B	MATH-500	12%	5%	5%	10%
DS-R1-7B	ProsQA	38%	25%	27%	35%
DS-R1-7B	GPQA-D	5%	0%	0%	0%
32B scale ($n = 20$; layers 32/48/55/62 of 64)					
Qwen2.5-32B	MATH-500	25%	5%	10%	5%
Qwen2.5-32B	ProsQA	15%	10%	30%	25%
Qwen2.5-32B	GPQA-D	0%	0%	0%	0%
DS-R1-32B	MATH-500	40%	20%	10%	10%
DS-R1-32B	ProsQA	10%	25%	**55%**	30%
DS-R1-32B	GPQA-D	10%	5%	10%	5%

Table 6: Direction ablation correctness change rates by model, benchmark, and layer depth fraction. Bold = highest single cell. †93% for 7B (layer 26/28); 97% for 32B (layer 62/64).

Benchmark	7B	14B	32B
MATH-500	$+0.054$	$+0.019$	-0.013
PrOntoQA	-0.036	-0.001	-0.149
ProsQA	$+0.035$	$+0.129$	$+0.285$
MMLU-Pro	$+0.021$	$+0.052$	-0.666
BBH-LD	-0.096	-0.218	-0.289
BBH-TSO	$+0.006$	$+0.388$	$+0.107$
GPQA-D	-0.022	$+0.023$	-0.014
Mean	-0.005	$+0.056$	-0.106

Table 7: Architecture-controlled pipeline effect on BCA, computed as DS-R1 minus Qwen2.5 at the same scale. Positive values indicate higher alignment after the reasoning pipeline; negative values indicate lower alignment.

expert-level science questions involve distributed rather than concentrated answer representations. (6) Overall, the 32B pair (mean 16.7% DS-R1, 11.7% Qwen2.5) shows comparable causal effect magnitudes to the 7B pair (16.1% DS-R1, 12.8% Qwen2.5), confirming that the detected belief signals remain causally relevant at scale.

K Human Annotation Validation

To complement the LLM judge validation (Appendix E), three human annotators independently labeled 100 stratified mismatch instances sampled across model families and benchmarks. Each annotator assigned one of the five taxonomy categories (CS, HR, CT, SEC, PC) to each instance based on the CoT text and the answer-commitment trajectory summary. Inter-annotator agreement reaches Gwet's $\mathrm{AC1} = 0.71$ on the five-way classification, comparable to the LLM judge agreement ($\mathrm{AC1} = 0.78$). The human labels confirm that the taxonomy categories correspond to qualitatively distinguishable phenomena: annotators most reliably identify Confabulated Steps (CS) and Premature Convergence (PC), with most disagreements occurring between CT and HR, consistent with the multi-label overlap documented in Appendix C.2.

L Qualitative and Vacuousness Analysis

L.1 Qualitative Examples of Taxonomy Categories

We present representative examples of each taxonomy category to illustrate the qualitative differences among mismatch types. All examples are drawn from the current E1 evaluation set and are lightly edited for space.

Confabulated Steps (CS). DS-R1-Distill-Qwen-32B on ProsQA (instance `prosqa_59`): The model's internal answer probability exceeds 0.8 from step 1, yet the trace continues for 30+ additional steps, generating text such as "*Let me verify by tracing the path: A connects to B, B connects to C...*" Each subsequent step induces $\Delta p_{\text{ans}} < 0.01$, meeting the CS criterion. The reasoning is logically valid but temporally downstream of answer formation under our proxy.

Hidden Reasoning (HR). Llama 3.1 8B on MATH-500: Between steps 3 and 4, the answer probability jumps from 0.12 to 0.67 ($|\Delta p_{\text{ans}}| = 0.55$), yet the CoT text at step 4 reads "*Substituting $x = 3$ into the equation...*" with no change in the explicit answer state. The large internal shift without corresponding explicit progress suggests that a significant inference occurred outside the visible trace.

Contradictory States (CT). Qwen 3.5 9B on GPQA-Diamond: At step 5, the answer-commitment proxy assigns probability 0.72 to answer (B), yet the trace explicitly endorses answer (C). This high-confidence disagreement ($\text{BCA} = 0$, probe confidence $= 0.72 > 0.5$) reflects a direct conflict between internal commitment and written reasoning.

Silent Error Correction (SEC). Qwen2.5-14B on BBH-Logical Deduction: The answer-commitment proxy disagrees with the trace for three consecutive steps (steps 4–6), then re-aligns at step 7 without any explicit correction or backtracking in the text. The model appears to have internally revised its answer and then resumed writing consistent reasoning without acknowledging the revision.

Premature Convergence (PC). DS-R1-Distill-Qwen-7B on MATH-500: The answer probability exceeds $\tau = 0.8$ at step 2 ($\text{CTG} = 4$), but the trace does not state the final answer until step 6. The four intervening steps show decreasing Δp_{ans} values, consistent with the model writing out derivation steps after having already committed to a particular answer internally.

L.2 Extended Qualitative Analysis

We present additional qualitative examples that complement the single-instance illustrations in Appendix L.1 with multi-instance contrasts across model families.

Confabulated steps: surface plausibility vs. belief stasis. Three examples illustrate the contrast between logically valid reasoning and flat answer-commitment trajectories. **(a)** DS-R1-14B on MATH-500 (`math500_4`): $p_{\text{ans}} = 0.97$ from step 1; the model then computes average speeds for five runners across 10 additional steps ($\max |\Delta p_{\text{ans}}| = 0.03$), each step arithmetically correct, arriving at "*Therefore, the student with the greatest average speed is Evelyn.*" Every calculation is valid, yet all 10 post-commitment steps are temporally downstream of answer formation. **(b)** Llama 3.1 8B on GPQA-Diamond (`gpqa_diamond_110`): $p_{\text{ans}} \geq 0.89$ on the correct answer (B) throughout 15 steps, yet the explicit calculation arrives at $v = 0.5$ m/s, contradicting the ground-truth answer of 1.58 m/s. The model then writes: "*However, I noticed that the solution provided does not match any of the answer choices.*" The internal commitment and the explicit derivation are not merely desynchronized but substantively divergent—the model "knows" the answer while its visible reasoning computes a different value. **(c)** DS-R1-7B on BBH-LD (`bbh_ld_130`): a perfectly flat trajectory ($\Delta p_{\text{ans}} = 0.000$) across six steps of logical deduction, all written in DS-R1's characteristic extended-thinking format.

Direction ablation: before and after. Projecting out the answer-token direction at layer 55 of DS-R1-32B on ProsQA produces three qualitatively distinct effects. At $\alpha = 5$ (`prosqa_59`), the model begins coherently ("*Let's break down the problem step by step...*") but degenerates into repetitive fragments of the answer token ("*Sallyumpusumpusumpusumpus...*")—a direct signature of the causal manipulation. At $\alpha = 10$ (`prosqa_33`), generation collapses entirely into the ablated token fragment ("*ump\numpump\numpump...*"). At $\alpha = 10$ (`prosqa_46`), the model preserves sentence structure but selectively truncates nonce-word answer tokens ("*impu*" for "impus", "*wumpu*" for "wumpus"), demonstrating that the ablated direction is localized to the answer representation while leaving syntax intact. A complementary pattern occurs at 7B: on `prosqa_26`, ablation *corrects* a previously wrong answer (original incorrect $\rightarrow$ patched correct), confirming that the detected direction was causally responsible for the incorrect commitment rather than merely correlated with it.

Hidden reasoning: invisible inference. Qwen2.5-14B on MATH-500 (`math500_289`): p_{ans} jumps from 0.00 to 1.00 between steps 2 and 3. At step 2 ($p = 0.00$) the model writes "*By Cauchy-Schwarz,* $(a+b+c+d)(1/a+1/b+1/c+1/d) \geq (1+1+1+1)^2 = 16$."—the genuine derivation. At step 3 ($p = 1.00$) it writes "*Therefore, the minimum value is 16.*" The internal commitment crystallizes only when the answer is explicitly stated, not during the step that contains the actual inference. This HR pattern—reasoning visible, commitment invisible—is the qualitative inverse of CS and suggests that for mathematical derivations, the logit lens may capture *answer-token* commitment rather than *solution knowledge*, consistent with the proxy limitation noted in Section 8.

L.3 Confabulated-Step Vacuousness Analysis

A natural objection to the "confabulated steps" interpretation is that the model may be performing genuine verification that happens not to change p_{ans}. To test this, we analyze all instances with detected CS steps across 56 model-benchmark configurations using the cached answer-probability trajectories (a representative subset of the full evaluation set). For each instance, we check whether the committed answer token ever changes during CS steps, whether p_{ans} remains above the commitment threshold $\tau = 0.3$ throughout, and what fraction of CS steps show near-zero belief change.

Across 314 instances containing 1,704 CS steps:

- **100%** of instances maintained stable answer commitment throughout all CS steps—the committed answer never changed.
- **90.1%** of individual CS steps showed near-zero belief change ($\Delta p_{\text{ans}} < 0.001$).
- **66.3%** of CS steps had strong commitment ($p_{\text{ans}} > 0.8$); the mean p_{ans} across all CS steps was 0.851.
- On average, CS steps constituted 30.9% of total steps per instance.

The 100% answer-stability rate across all 314 instances is the key finding: if CS steps were performing genuine verification, we would expect at least some instances where the verification process leads to an answer revision (a dip or change in p_{ans}). Instead, the answer-commitment signal is entirely flat, consistent with vacuous continuation rather than active checking. Per-model rates of strong commitment range from 56.1% (Qwen2.5-14B) to 81.8% (Qwen2.5-7B), with all models showing 100% stable argmax.

M Additional Sensitivity and Comparison Analyses

M.1 Cross-Method Faithfulness Comparison

To assess whether BCA captures information distinct from simpler faithfulness proxies, we compute an *early commitment rate* (ECR) for each model-benchmark configuration: the fraction of instances where $p_{\text{ans}} > \tau$ by the midpoint of the reasoning chain. ECR is analogous to the early-answering tests of Lanham et al. [28], computed from internal answer-probability trajectories rather than behavioral perturbation.

Across 56 configurations, the Spearman correlation between BCA and ECR is $\rho = 0.048$ ($p = 0.73$), indicating that the two metrics capture essentially orthogonal information. This is expected: ECR measures whether the model has committed early in absolute terms, while BCA measures the *timing alignment* between latent commitment and explicit answer arrival across the full chain. A model can have high ECR (commits early) but also high BCA (if the CoT also converges early), or low ECR (commits late) with low BCA (if the CoT converges at a different time).

ECR shows no significant relationship with CoT utility ($\rho = -0.12$, $p = 0.69$, $n = 14$), paralleling the BCA–utility null result. The near-zero BCA–ECR correlation confirms that BCA provides a measurement lens that is complementary to, rather than redundant with, response-level early-answering metrics.

M.2 Per-Scale Taxonomy Shift Details

We report the full per-scale chi-squared analysis of taxonomy shifts between matched Qwen2.5 (base) and DS-R1 (pipeline-trained) pairs. At 7B ($N_{\text{base}} = 17{,}338$; $N_{\text{pipe}} = 17{,}971$), the taxonomy is largely preserved ($\chi^2 = 119.8$, $p < 10^{-24}$, Cramér's $V = 0.058$): confabulated steps account for

57.9% versus 53.0%, with all other categories shifting by less than 3.5 pp. At 14B ($N_{\text{base}} = 36{,}025$; $N_{\text{pipe}} = 21{,}881$), the effect grows ($\chi^2 = 1{,}399$, $V = 0.155$) and the pipeline reduces average mismatches per instance from 24.2 to 14.7 (a 39% reduction), consistent with its positive BCA effect. At 32B ($N_{\text{base}} = 17{,}538$; $N_{\text{pipe}} = 25{,}858$), the shift is largest ($\chi^2 = 3{,}629$, $V = 0.289$): confabulated steps drop from 57.6% ($N = 10{,}096$) to 37.4% ($N = 9{,}674$; -20.2 pp), while contradictory states surge from 7.1% ($N = 1{,}246$) to 28.9% ($N = 7{,}480$; $+21.8$ pp). The 7B-vs-32B aggregate BCA contrast does not reach significance (paired Wilcoxon: $p = 0.578$), and the 32B mean drops to -0.012 when the large MMLU-Pro outlier ($\Delta = -0.666$) is excluded. The mechanistic evidence in Appendix H.2 suggests that at 32B, the pipeline concentrates answer commitment into late layers whose representations are less tightly coupled to the generated text, creating opportunities for the trace to endorse a different answer than the one the model has internally resolved.

M.3 BCA–Utility Sensitivity Analysis

The Spearman rank correlation between BCA and CoT utility is directionally consistent but not significant ($\rho = -0.25$, $p = 0.16$), indicating sensitivity to high-leverage points. A leave-one-benchmark-out analysis confirms that ProsQA is the primary leverage point: removing it yields $r = +0.09$ ($p = 0.61$), whereas removing any other single benchmark preserves the negative direction ($r \in [-0.43, -0.26]$) with three of six remaining benchmarks maintaining significance ($p < 0.05$). ProsQA's leverage is interpretable rather than artifactual: it is the only graph-planning task where direct accuracy is near zero yet CoT accuracy exceeds 94%, making it the clearest example of CoT enabling capability that the model cannot achieve without explicit reasoning—precisely the regime where the capability–oversight trade-off should be strongest. All five within-model correlations are negative ($r \in [-0.52, -0.39]$), though individually non-significant at $n = 7$, providing directional consistency across model families.

N Temperature Sensitivity

All main experiments use greedy decoding for reproducibility, but reasoning models are typically deployed with temperature sampling. To check whether the qualitative patterns hold under sampling, we run two key configurations at $T = 0.6$ (top-$p = 0.95$) with three samples each ($n = 60$ instances per configuration).

Config	Decoding	Accuracy	CoT Length	Consistency
Llama 3.1 8B / MATH	Greedy	0.667	2,043	—
	$T = 0.6$	0.600 ± 0.014	2,445	0.75
DS-R1-7B / ProsQA	Greedy	0.950	2,800	—
	$T = 0.6$	0.972 ± 0.016	2,724	0.93

Table 8: Temperature sensitivity check. Accuracy and CoT length under greedy vs. temperature sampling. Consistency = fraction of instances where all three samples produce the same answer.

Table 8 summarizes the results. Two observations support the robustness of our main findings. First, accuracy under temperature sampling is within 7 percentage points of greedy in both configurations, indicating that temperature does not fundamentally alter the model's reasoning capacity. Second, CoT traces are 15–20% longer under sampling for Llama 3.1 8B yet accuracy drops slightly, consistent with the main-text finding that additional reasoning text does not necessarily reflect additional inferential contribution. For DS-R1-7B on ProsQA (greedy BCA = 0.342), accuracy is comparably high under sampling (0.972), and 93% of instances produce consistent answers across all three samples, reinforcing that the model has committed to its answer before the extended reasoning trace concludes. These results suggest that the performative-reasoning pattern identified under greedy decoding is not an artifact of the decoding strategy.

O Detailed Differentiation from Closely Related Work

Several concurrent works share parts of our motivation—especially the observation that a model can commit to an answer internally before its trace reveals it—so we briefly make the differences explicit. Table 9 compares DETECT-CLASSIFY-COMPARE against ten closely related works along

the axes we view as most consequential for diagnosing CoT unfaithfulness: measurement granularity, internal probe family, classification granularity, architecture-matched pipeline control, and capability–oversight quantification.

Work	Granularity	Internal probe	Classification	Arch-matched	Cap–Over.
Lanham et al. [28]	Response	Truncation / perturb.	Binary	No	×
Turpin et al. [50]	Response	Bias injection	Binary	No	×
Chen et al. [8]	Response	Hint injection	Binary	No	×
Arcuschin et al. [2]	Response	Contrastive prompting	Binary	No	×
Tutek et al. [51]	Step	Parameter unlearning	Scalar drop	No	×
Ye et al. [64]	Step (NLDD)	Corruption + conf. diff.	Scalar decay	No	×
Young [65, 66]	Response / token	Hint injection	Binary	No	×
Palod et al. [40]	Trace length	None (behavioral)	Length–complexity corr.	No	×
Boppana et al. [6]	Token / step	Trained attn. probes	Binary (perf.)	No	×
Hase and Potts [21]	Training-time	Counterfactual simulation	N/A	No	×
DETECT-CLASSIFY-COMPARE (ours)	**Step (timing)**	**Logit lens + Patchscopes + tuned lens**	**Five-way**	**Yes (3 scales)**	✓ ($r{=}-0.42$)

Table 9: Axis-by-axis comparison of DETECT-CLASSIFY-COMPARE against closely related work. "Arch-matched" indicates whether the work uses architecture-controlled model pairs to isolate training-pipeline effects; "Cap–Over." indicates whether the work quantifies a capability–oversight trade-off. DETECT-CLASSIFY-COMPARE is the only framework combining a training-free multi-method probing pipeline, a five-way step-level taxonomy, architecture-matched scale comparison, and an explicit capability–oversight correlation.

What makes DETECT-CLASSIFY-COMPARE different. DETECT-CLASSIFY-COMPARE is distinguished from prior work on four concrete points. **(i) Timing rather than causal importance.** Step-level methods such as Tutek et al. [51] and Ye et al. [64] quantify how much each step *matters causally* via unlearning or corruption; DETECT-CLASSIFY-COMPARE instead measures *when* latent answer commitment stabilizes relative to the trace, which is why BCA is sensitive to post-commitment continuation that causal-importance metrics can score as inert. **(ii) Training-free multi-lens probing.** Related work splits here: Palod et al. [40] questions whether longer traces reflect more adaptive computation, while Boppana et al. [6] studies latent belief/trace mismatch with trained attention probes whose accuracy is highly sensitive to probe architecture (87.98% vs. 31.85% on MMLU in Boppana et al. 6); DETECT-CLASSIFY-COMPARE uses the logit lens cross-validated with Patchscopes, tuned lens, and causal direction ablation (Appendix J.2), all training-free. **(iii) Five-way taxonomy instead of a binary verdict.** Performative-vs-genuine framings collapse qualitatively different failure modes: at 32B the R1 pipeline reduces confabulated steps by 20 pp while contradictory states *grow* by 22 pp (Section 5.3)—a swap a binary framing would score as roughly unchanged, but that changes surface plausibility and oversight difficulty substantially. **(iv) Architecture-matched pipeline control with an explicit capability–oversight correlation.** No prior work in Table 9 pairs a reasoning-trained model to its backbone; cross-family comparisons (e.g. DeepSeek-R1-671B vs. GPT-OSS-120B in Boppana et al. 6) confound architecture with training. Our Qwen2.5 vs. DeepSeek-R1-Distill pairs at 7B/14B/32B hold the backbone fixed, and we additionally report a Pearson $r{=}-0.42$ ($p{=}0.012$, 95% CI $[-0.64, -0.13]$) between BCA and CoT utility across 35 configurations, turning the faithfulness–capability trade-off from an anecdote into a quantitative claim. These directions are complementary to the probes enabling early exit in Boppana et al. [6] and the training recipe of Hase and Potts [21], and a natural extension of DETECT-CLASSIFY-COMPARE is to use BCA as a training-time monitor for faithfulness-improving interventions.

P Pure-Subgroup Causal Truncation

Section 5.2 reports five mismatch categories as a descriptive decomposition. To test whether each category has a *step-level causal signature*, we run a within-instance paired-truncation experiment restricted to instances where exactly one detector fires above severity 0.30 ("pure" subgroups). Restricting to pure instances is required because the five detectors are structurally correlated (Appendix C.2), and pooling over multi-label instances mixes opposite-sign effects that cancel in the marginal.

Within-instance paired design. For each pure-subgroup instance i where category c fires, we pick a *characteristic firing step* S_i^c (highest-p_{ans} CS firing; largest-$|\Delta p_{\text{ans}}|$ HR jump; most-confident CT disagreement), truncate the CoT at S_i^c, append a reasoning-model-aware answer-elicitation suffix,

and regenerate greedy. As paired controls on the *same* instance we cut at $S_i^c + \delta$ with δ drawn from $\{-2, -1, 1, 2\}$ (neighbor control) and at a uniformly random step R_i (uniform control). The paired causal statistics are

$$\Delta^c_{i,\mathrm{N}} = \mathbb{1}[\mathrm{correct}(S_i^c)] - \mathbb{1}[\mathrm{correct}(S_i^c + \delta)], \qquad \Delta^c_{i,\mathrm{U}} = \mathbb{1}[\mathrm{correct}(S_i^c)] - \mathbb{1}[\mathrm{correct}(R_i)].$$

Because each control is on the same instance, instance-level difficulty and depth effects cancel out. The experiment runs on all 56 model-benchmark configurations used in Appendix C.2 (8 open-weight models $\times$ 7 benchmarks), pooling 2,088 pure-CS, 410 pure-HR, and 38 pure-CT paired triples.

Pooled results. Table 10 reports the overall pooled paired Δ with paired t-tests and bootstrap 95% CIs. Both CS and HR show statistically significant *positive* deltas—cutting at a CS or HR firing *helps* accuracy relative to either control. The effect size is much larger against a uniformly random cut (Δ_{U}) than against a matched neighbor cut (Δ_{N}), because the characteristic-step picker deliberately targets steps that matter for the detector's semantic signal, which is not true of the random control. CT shows a directional negative signal (cutting hurts, consistent with a load-bearing interpretation) but is underpowered at $n = 38$.

Category	n	$\Delta_{\text{vs N}}$	95% CI	p	$\Delta_{\text{vs U}}$	p
CS (pure)	2,088	$+0.028$	$[+0.009, +0.047]$	0.0043	$+0.078$	$<10^{-16}$
HR (pure)	410	$+0.046$	$[+0.010, +0.085]$	0.0165	$+0.161$	$<10^{-16}$
CT (pure)	38	-0.053	$[-0.158, +0.053]$	0.3238	-0.053	0.160

Table 10: v3 pure-subgroup paired truncation, pooled over 56 model-benchmark configurations. Confabulated Steps (CS) show a significant *positive* Δ vs both neighbor ($\delta = \pm 2$ step) and uniform-random controls, confirming CS vacuousness. HR also shows significant positive Δ, in the direction opposite our original "load-bearing hidden computation" prediction. CT is directionally negative (load-bearing) but underpowered.

Task-type heterogeneity. The CS vacuousness effect is strongly task-dependent (Table 11): it is large and significant on simple-answer benchmarks (BBH-LD Δ=$+0.080$, BBH-TSO Δ=$+0.077$, MMLU-P Δ=$+0.045$, all $p < 0.02$; ProsQA Δ=$+0.103$, p=0.044), null on heavy-computation benchmarks (MATH-500 Δ=-0.003, GPQA-D Δ=-0.006), and trends negative on PrOntoQA (Δ=-0.045, p=0.10). The interpretation is that "vacuous post-commitment continuation" accurately describes late-stage CoT on tasks whose answer is determined early and the remaining steps are rhetorical padding, but on tasks whose answer is still being computed at the CS step the cut removes meaningful computation rather than padding.

Benchmark	n	$\Delta_{\text{vs N}}$	95% CI	p
BBH Logical Deduction	400	$+0.080$	$[+0.025, +0.135]$	0.0036
BBH TSO	325	$+0.077$	$[+0.028, +0.129]$	0.0025
ProsQA	39	$+0.103$	$[+0.026, +0.205]$	0.0439
MMLU-Pro	400	$+0.045$	$[+0.018, +0.075]$	0.0034
GPQA-Diamond	161	-0.006	$[-0.043, +0.031]$	0.7400
MATH-500	363	-0.003	$[-0.044, +0.041]$	0.8999
PrOntoQA	400	-0.045	$[-0.098, +0.008]$	0.1032

Table 11: Per-benchmark pure-CS paired $\Delta_{\text{vs N}}$. The CS vacuousness effect is positive on simple-answer tasks and null or negative on computation-heavy or rare-token tasks.

Scale dependence. Aggregating by model parameter scale (Table 12), the CS vacuousness effect peaks at 12–14B (Δ=$+0.056$, p=5×10^{-4}), is near-null at 7–9B, and near-null at 32B. HR, by contrast, shows its strongest positive Δ at 32B (Δ=$+0.101$, p=0.011). A possible reading: at 12–14B the model commits confidently enough to generate detectable vacuous tails but is not yet sophisticated enough to use post-commitment steps for verification; at 32B post-commitment continuation becomes more selective, so the pure-CS causal signal washes out while HR-firing steps at the same scale turn out to be verification-adjacent rather than forward-computing.

Scale	n_{CS}	Δ_{CS}	p_{CS}	n_{HR}	Δ_{HR}	p_{HR}
7–9B	841	+0.008	0.591	178	+0.039	0.162
12–14B	736	+0.056	$\mathbf{5.3{\times}10^{-4}}$	153	+0.026	0.452
32B	511	+0.022	0.310	79	+0.101	**0.011**

Table 12: Pure-subgroup $\Delta_{\text{vs N}}$ by parameter-scale band. CS vacuousness peaks at 12–14B; HR vacuousness peaks at 32B.

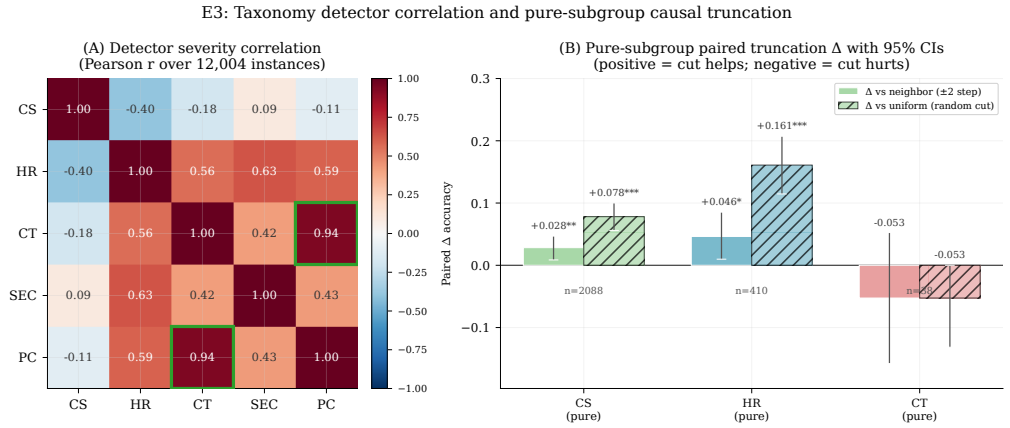

Figure 7: **(A)** Pearson correlation matrix over 12,004 instance-level peak severities. The CT↔PC pair is highlighted: $r = 0.937$, 95% CI $[+0.930, +0.943]$. **(B)** Pure-subgroup paired Δ (treatment minus control) for the three active categories with 95% bootstrap CIs. Hatched bars show $\Delta_{\text{vs U}}$ (uniform random cut); solid bars show $\Delta_{\text{vs N}}$ (matched neighbor $\delta = \pm 2$ step). CS and HR both show significant positive Δ, the opposite of the load-bearing prediction for HR. CT is directionally negative but underpowered ($n = 38$).

Interpretation. Three substantive implications follow. *(i) CS is causally vacuous on simple-answer tasks.* The within-instance pure-subgroup $\Delta_{\text{vs U}} = +0.078$ ($p < 10^{-16}$) confirms that cutting through a CS-firing step is a strictly better choice than cutting at a uniformly random step, and the $+0.080$ effect on BBH Logical Deduction alone is the cleanest validation the paper offers of CS as "vacuous post-commitment continuation." *(ii) HR is not universally load-bearing.* Our original prediction—that HR firings ("hidden computation" jumps) would be causally load-bearing and therefore show $\Delta < 0$—is contradicted at the pooled level ($\Delta_{\text{vs N}} = +0.046$, $p = 0.016$) and even more strongly against the uniform control ($\Delta_{\text{vs U}} = +0.161$, $p < 10^{-16}$). One natural re-reading is that large $|\Delta p_{\text{ans}}|$ jumps flag steps *immediately after* which the answer has been locked in, so cutting at the pre-jump step still leaves a prefix sufficient for the model to re-derive the committed answer via greedy decoding from the neighboring context. *(iii) The five-way taxonomy is a descriptive decomposition.* Combined with the structural non-isolability of PC and SEC (Appendix C.2), the pure-CS/pure-HR results support reading the taxonomy as a descriptive breakdown of correlated facets of trace unfaithfulness rather than five independent causal mechanisms. This is compatible with the paper's original claim that DETECT-CLASSIFY-COMPARE measures timing alignment (BCA) rather than causal step-importance.

Robustness: HR post-jump variant. The HR-surprise reading above—that a large $|\Delta p_{\text{ans}}|$ jump marks the step *after* which the answer has been locked in—makes a direct prediction: cutting just *after* the jump should be at least as vacuous as cutting just before it. We test this by rerunning the paired truncation pipeline with the HR treatment step shifted from the pre-jump position (step_range[0]) to the post-jump position (step_range[1] + 1) on 49 of the 56 main configurations (DS-R1-32B is excluded because its model-path symlink became invalid between the E3 main run and this robustness run). CS and CT pickers are held identical to the main run so that those categories serve as internal sanity checks. The clean test is a within-instance paired comparison on the subset of HR instances where both pre-jump and post-jump treatment steps are defined ($N = 73$). On this paired subset, the pre-jump cut gives $\Delta_{\text{vs N}} = -0.027$ ($p = 0.60$, not significant), while the post-jump cut gives $\Delta_{\text{vs N}} = +0.137$ ($p = 0.011$); a within-instance paired t-test on the difference (post minus pre) yields $+0.164$ ($p = 0.027$). Both cut positions lie in the same (vacuous) direction when pooled across all

instances where each is applicable, but the post-jump cut is markedly larger in magnitude on the paired subset. This supports the re-reading: cutting at the pre-jump step leaves a prefix sufficient for greedy decoding to re-derive the committed answer, whereas cutting at the post-jump step preserves the commit moment and strips the now-redundant post-commit tail. The main conclusion—that HR is causally vacuous on average in pure subgroups—is robust to the exact choice of step within the HR firing window, and in fact becomes stronger when the HR cut is placed at its semantically more interpretable position.

Complementary test: CS content is causally inert under donor corruption. To directly test whether CS steps are being *read* during the original forward pass (as opposed to merely being unnecessary under regeneration), we run a corruption experiment that keeps the full CoT intact but replaces the tokens at the CS firing step with length-matched tokens from another instance's CS firing step (cross-instance donor swap). For each of $N = 1{,}733$ pure-CS instances pooled over 51 configurations, we compare corruption at three locations on the same instance: the CS firing step itself, a ± 2 neighbor step, and a uniformly-random non-CS step. We measure the intervention two ways: Variant A reads the last-step answer probability $p(\text{ans} \mid \text{corrupted context})$ under forced decoding (logit-level); Variant B greedy-regenerates a short answer after an elicitation suffix and scores correctness (behavior-level). Under Variant B, corrupting the CS step changes final-answer accuracy by $\Delta_{\text{CS}} = -0.002$ ($p = 0.78$, paired t-test, essentially a null), while corruption at a neighbor or uniformly-random step changes accuracy by $\Delta_{\text{N}} = -0.020$ ($p = 0.013$) and $\Delta_{\text{U}} = -0.021$ ($p = 0.007$) respectively. The within-instance paired contrasts are $\Delta_{\text{CS}} - \Delta_{\text{N}} = +0.017$ ($p = 0.039$) and $\Delta_{\text{CS}} - \Delta_{\text{U}} = +0.019$ ($p = 0.028$), both significant at the $p < 0.05$ level: corrupting CS is measurably less damaging than corrupting any other position on the same instance. The effect concentrates at the 12–14B parameter band (paired $\Delta_{\text{CS}} - \Delta_{\text{N}} = +0.046$, $p = 0.003$) and on BBH-Tracking in particular ($\Delta_{\text{CS}} = +0.097$, $p = 6.1 \times 10^{-4}$). Variant A's logit-level signal is smaller because most pure-CS instances already have $p(\text{ans})$ close to 1.0 in the original context and saturate; the paired contrast nonetheless points in the same direction ($\Delta_{\text{CS}} - \Delta_{\text{N}} = -0.009$, $p = 0.004$ for the ± 2 neighbor control, where a more negative value indicates less logit damage from CS corruption). Two observations follow. First, the CS step's tokens are not merely dispensable under re-generation (the truncation test above): they are actively ignored during the original forward pass, since replacing them with unrelated text from another CoT produces no measurable accuracy loss while the same replacement elsewhere in the trace does. Second, the specificity holds even though both neighbor and uniform corruption locations often also land within the post-commitment tail; the residual effect size at other-location corruption shows that the CS picker is selecting the step whose content contributes least to downstream generation. PrOntoQA is again the one benchmark where CS corruption *hurts* accuracy ($\Delta_{\text{CS}} = -0.088$, $p = 4.6 \times 10^{-5}$), consistent with its outlier behavior in the main truncation test on rare-token graph reasoning with tight cross-step dependencies.

Limitations of this analysis. The pure-CT subgroup ($n = 38$ across 56 configurations) is too small to reach $p < 0.05$ individually despite the consistent load-bearing sign, and we report CT as directionally suggestive rather than confirmed. The CS and HR pure-subgroup selection necessarily excludes instances where a category co-fires with any other, which means the reported Δ does not estimate the causal effect of that category in its more common multi-label context; that latter estimand is not identifiable under the current detector definitions because the pure pool is sampled from the tail of a correlated joint distribution. Our findings should be read as bounds on the causally clean component of each category.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state a step-level empirical test of whether CoT traces remain temporally synchronized with internal answer commitment, and the listed contributions match the reported BCA, taxonomy, matched-pipeline, and CoT-utility analyses. The scope is limited to empirical evidence from the evaluated models and benchmarks rather than a universal claim about all models.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 7 discusses the answer-commitment proxy, possible nonlinear or multi-token beliefs, intervention-strength selection, overlapping taxonomy categories, the Qwen2.5/DeepSeek-R1 comparison scope, and the restriction to open-weight 7B–32B models.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper does not present formal theoretical results, theorems, or proofs. The equations define operational measurements and interventions used in the empirical framework.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper discloses the models, benchmarks, inference setup, step parser, answer readout, BCA/CTG metrics, thresholds, statistical procedures, validation checks, and intervention protocol needed to reproduce the main empirical results. The experiments use existing open-weight models and public benchmarks rather than training a new model.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some

way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: At submission time, we do not provide an anonymized public code/data release with exact reproduction commands. The paper nevertheless describes the methodology and relies on public models and benchmarks; code and generated artifacts can be released after anonymity and licensing review.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: Section 3 specifies the evaluated models, benchmarks, instance counts, greedy decoding setup, step segmentation procedure, hidden-state readout, BCA thresholding, confidence-interval construction, and compute budget. Since no models are trained in this work, optimizer and training hyperparameters are not applicable.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: The paper reports 95% bootstrap confidence intervals with Bonferroni correction and appropriate statistical tests, including ANOVA, Bayesian model comparison, correlations, Wilcoxon tests, chi-squared tests with Cramer's V, and intervention/truncation significance tests. Sensitivity analyses for threshold choice and validation checks are reported in the appendix.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [No]

Justification: The draft reports aggregate compute for the full evaluation, approximately 200 GPU-hours on four A100 GPUs, but does not yet provide per-experiment runtime, memory, and storage requirements. We therefore conservatively answer no and will add a more detailed compute table in the appendix.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: The authors have reviewed the NeurIPS Code of Ethics and the research is intended to conform to it. The work uses public benchmarks and existing models, does not collect private personal data, and does not release a high-risk model or scraped dataset.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The paper discusses positive implications for improving CoT-based oversight and process monitoring, while also identifying the negative risk that fluent CoT traces and step-level monitors may be misleading after internal answer commitment. The work is diagnostic rather than a deployment or capability-release paper.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release a pretrained language model, image generator, scraped dataset, or other high-risk asset requiring controlled release safeguards. The experiments use existing public models and benchmarks and report diagnostic measurements.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [No]

Justification: The paper cites the original model and benchmark sources used in the experiments, but the current draft does not yet include an explicit asset table listing licenses and terms of use for each existing model and dataset. We therefore conservatively answer no and will add license and usage-term information in the appendix.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The current submission does not release a new dataset, model, or packaged code asset. If code, generated traces, cached activations, or annotation files are released later, they should be documented with a README, license, intended-use statement, and reproduction instructions.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: Appendix K reports a small human annotation validation, but all annotators were authors and no external participants were recruited, this item should be N/A.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [No]

Justification: The current draft does not state an IRB or exemption status, participant-risk disclosure, or equivalent review procedure for the human annotation validation. We therefore conservatively answer no because the annotation study is mentioned but review status is not described.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification:

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.