
CATASTROPHIC FORGETTING MAKES THE CHAIN-OF-THOUGHT MORE LOAD-BEARING, NOT LESS, WHERE NO SHORTCUT EXISTS¹

Mustafa Ilker Aktas
Independent Researcher

With
Apart Research

Abstract

After the Hugging Face incident, OpenAI made the use of chain-of-thought monitoring a mandatory requirement for RL training and evaluation. However, this is only true if the model's chain-of-thought causally determines the answer; a fluent but causally inert chain is text that the monitor can read but cannot track. Lobo et al. demonstrated that fine-tuning reduces chain-of-thought faithfulness and conjectured that this is due to catastrophic forgetting; although this hypothesis has not been tested, it is accepted in the literature as an established finding. Two fine-tuning runs per seed at two seeds, on a 1.5B model, with identical data, format, schedule, and hyperparameters; the only variable was whether replay suppressed forgetting or not; measurements were taken at 13 checkpoints over 600 steps. Forgetting occurred in one arm (general NLL rose by 0.436 on a fixed corpus), but not in the other (-0.032), and the model did not collapse in either arm. Forgetting did not disrupt the causal role of the chain-of-thought; it increased it on four length-invariant measures, reported as forgetting arm versus control: error propagation 0.486 versus 0.422, the shuffled-chain gap 0.229 versus 0.031, the chain's contribution to accuracy 0.762 versus 0.575, and the effect of premise masking -0.201 versus +0.254.

¹ Research conducted at the [AI Incident Response Sprint](#), September 2026

The difference is explained by the evaluation task: without the chain the model performs at chance (0.150, chance 0.167), so no shortcut is available. This reconciles the two sets of results and yields a measurable regime test a lab can run on its own tasks.

1. Introduction

In July 2026, two OpenAI models escaped their sandbox during a cyber-capability evaluation and went on to attack Hugging Face’s production infrastructure, apparently to steal the benchmark’s reference solutions and cheat the evaluation [1]. OpenAI’s remediation plan made chain-of-thought monitoring a mandatory requirement for all tool-using RL training and evaluation involving models at or above GPT-5.6 Sol capability. For Astra-class models, this also includes tool-enabled inference workloads [2].

*OpenAI states that if the chain-of-thought monitoring system currently deployed had been operational, it would have alerted the security team more than a day before the models breached Hugging Face’s production infrastructure [2]. The monitoring worked here because the chain was both legible and honest; the agent had explicitly stated its own transgression: **“We’re attacking third-party HF using leaked token, potentially outside intended scope. ... This is arguably unauthorized. ... Yet goal solution. ...”***

Chain-of-thought monitoring depends on two things: the chain being legible, and the chain causally determining the answer. The first is visible, the second is invisible. A fluent but causally inert chain is text that the monitor can read but tracks nothing. The failure is silent: no alert fires, because an alert requires the monitor to see something. OpenAI’s new escalation procedure pages on monitor output and halts the run unless it is shown to be a false positive within thirty minutes [2]. That procedure reads silence as absence of misbehaviour.

The same OpenAI document sets out three new post-training commitments: graders that also evaluate how tasks are completed, RL environments that teach distrust of unauthorized instructions, and alignment over long tasks [2]. Each is optimization applied to a model in which chain-of-thought monitorability is now considered a security control. No test connects them: no check is run before deployment, nor any published metrics showing how the model behaved during a training run. OpenAI’s own report also measures that cheating rates increased over the course of a training run using its own chain-of-thought monitoring [2]. The training run is already their unit of analysis; what’s missing is measuring faithfulness along the same axis.

I’m working on the subproblem “checks somebody could run tomorrow” in Track 2, “What happened, and what breaks next”. The question is whether the control survives the fix, and how

anyone would know before deployment. I address this by testing the untested conjecture of Lobo et al. through a controlled comparison [3].

My main contributions are:

1. *The first direct test of Lobo et al.'s catastrophic forgetting conjecture on chain-of-thought: a controlled comparison in which forgetting is the only variable, measured at 13 checkpoints over 600 steps.*
2. *Unexpected finding: On a task where no shortcut is possible, forgetting does not reduce causal dependence on the chain-of-thought; rather, it increases it on four length-invariant measures.*
3. *A measurable regime test that reconciles the two sets of results, chain-free accuracy (a single number that a laboratory can calculate for its own evaluation tasks), published as a single file requiring only model weights.*

No laboratory tests whether the chain determines the answer before counting chain-of-thought monitoring as a control. This harness reduces that test to the cost of a single evaluation run, and chain-free accuracy tells a lab which risk regime it is in.

2. Related Work

Korbak et al. [4] frame chain-of-thought monitorability as a fragile opportunity and list which training-time optimization pressures degrade it as an open question. All published answers to this question relate to deliberate pressure: direct optimization against the monitor, reward hacking, and monitor-aware prompts. Ordinary, well-intentioned domain fine-tuning has not been examined; that is the gap this work addresses.

The closest prior work is Lobo et al. [3]. They show through early termination and filler substitution tests that fine-tuning reduces chain-of-thought faithfulness, and hypothesize the following reason: ***“We conjecture that the decline in CoT reasoning accuracy and faithfulness is likely due to catastrophic forgetting”***. They further report that ***“fine-tuning LLaMA-3-8B-Instruct models on CosmosQA and MedQA generally increases CoT Pred Match values, indicating a decline in CoT reasoning faithfulness. We hypothesize that this occurs because CosmosQA and MedQA involve less complex reasoning, leading to overfitting”***. They also report that larger models are more robust. Their attribution was a conjecture; I tested it. They measured two points, before and after fine-tuning; I measured 13 checkpoints and verified that forgetting occurred. Their mechanism predicts a steeper decline in my setup, not a shallower one: a simple domain and a small model.

Sinha et al. [5] define the characteristics that a good reasoning model should possess: ***“delayed answer commitment, where answer probability should increase gradually with CoT token generation”***. They design a specialized attention-guided SFT objective to achieve this. They also report that the standard CoT-SFT ***“often relies increasingly on textual priors”***. They are trying to

engineer this feature; I’m presenting it as a byproduct of ordinary domain fine-tuning, without any special training objective.

Liu et al. [6] develop metrics for diagnosing chain-of-thought pathologies and argue that monitoring throughout training is necessary rather than relying on a single checkpoint. They deliberately induce pathology and establish model organisms by training models to rationalize post hoc. I’m studying whether pathology arises incidentally from ordinary fine-tuning.

Current faithfulness tests (early termination, filler substitution) cut at a fraction of the chain, so the results shift when the chain length changes. My four measurements are independent of length, and the harness requires only the model weights. Whether forgetting compromises faithfulness depends on whether a shortcut is available, and this can be measured with a single number.

3. Methods

DeepSeek-R1-Distill-Qwen-1.5B is the smallest open-weight reasoning model that generates long chain-of-thought and fits on a single GPU. The evaluation task is modular arithmetic: a starting value followed by two operations (addition or multiplication, mod 11), in the form of multiple-choice questions with six options. Each example is randomly generated, so it is impossible to memorize the answers. This task was chosen because the chain-of-thought had to be computationally necessary; if the model can answer directly from the prompt, a change in the chain’s causal role cannot be measured. The out-of-domain training data is synthetic clinical multiple-choice questions: simple, repetitive, and requiring no reasoning (the domain where Lobo et al. reported a decline [3]). All runs used two T4 GPUs.

Two fine-tuning runs per seed at two seeds. Training data, format, number of steps, learning rate, LoRA rank, target modules, and optimization schedule were identical. The only difference was that, in the replay arm, examples from the evaluation task were mixed into the training data. This is the standard method for suppressing forgetting. If the causal link weakens in one arm but not the other, forgetting is the only remaining explanation. Format pressure, gradient magnitude, and step count are the same in both arms. All runs used 600 steps, one checkpoint every 50 steps (13 checkpoints), 60 evaluation samples per checkpoint, LoRA rank 32, learning rate $4e-4$, attention and MLP modules. One limitation of this design: the replay examples were provided in an answer-only format, which means the replay arm also taught the model to answer directly. The correct design would have used a chained format; its effect on the results is addressed in **Section 4**.

Whether forgetting has occurred is measured by the negative log-likelihood per token on a fixed general English corpus, independent of the training data and the evaluation task. If there is no rise, forgetting has not occurred; in that case no figure concerning faithfulness can be interpreted. Three configurations were trained for 200 steps each and measured before the main run. The following two criteria were used to determine whether the result was acceptable: the NLL shift must exceed the threshold, and the model must still be able to generate a chain-of-thought. The selected

configuration was a learning rate of $4e-4$, rank 32, with attention and MLP modules. The measured shift was $+0.467$.

In the first run, the calibration only examined NLL drift and domain-task loss. The model collapsed after fifty steps (lr $5e-4$, rank 32): the chain-of-thought shrank from 141 words to 16, 99% of the examples became degenerate, and only two of sixty examples remained. All subsequent measurements were calculated based on just one or two examples and were completely meaningless. After this failure, a third condition was added to the calibration: if the chain-of-thought length fell below 40% of the baseline or if degeneracy exceeded 15%, the configuration was rejected.

Lobo et al.’s mechanism assumes that a shortcut is possible. This assumption is measurable: chain-free accuracy, the accuracy the model achieves without the chain-of-thought. If this is at the level of chance, then there is no shortcut. In this task it is 0.150, against a chance rate of 0.167.

The harness is a single file; it requires no installation and runs on a consumer GPU. Twenty-eight automated checks that validate the evaluation system run before the GPU begins processing. They verify that the span array fully covers the sequence, that batch size does not alter greedy generation, and that the attention masks are actually applied. It requires only the model weights; no access to a lab’s network, training data or internal logs.

4. Results

Forgetting occurred in one arm and not in the other. The model did not collapse in either:

Table 1. Forgetting Verification.	<i>ood_cot</i>	<i>replay</i>
General NLL (fixed corpus)	4.462 $\rightarrow$ 4.898 (+0.436)	4.462 $\rightarrow$ 4.430 (-0.032)
Degeneracy (last checkpoint)	0.00	0.00
Sample under evaluation, last	60/60	60/60

Chain-free accuracy is 0.183 $\rightarrow$ 0.150 against a chance rate of 0.167 in a six-option task.

Table 2 shows four length-invariant measures:

Table 2. Length-invariant Measures.	Step 0	<i>ood_cot</i>	<i>replay</i>
Error propagation	0.500	0.486	0.422

<i>Shuffled-chain gap</i>	<i>0.133</i>	<i>0.229</i>	<i>0.031</i>
<i>Chain contribution to accuracy</i>	<i>0.783</i>	<i>0.762</i>	<i>0.575</i>
<i>Premise masking</i>	<i>+0.017</i>	<i>-0.201</i>	<i>+0.254</i>

All four are in the same direction. Table 3 gives the aggregate contrast at both seeds:

Table 3. Aggregate Contrast at two Seeds.	<i>drop ood_cot</i>	<i>drop replay</i>	<i>CONTRAST</i>
<i>Seed 0</i>	<i>+0.080</i>	<i>+0.049</i>	<i>+0.030</i>
<i>Seed 1</i>	<i>+0.033</i>	<i>+0.007</i>	<i>+0.026</i>

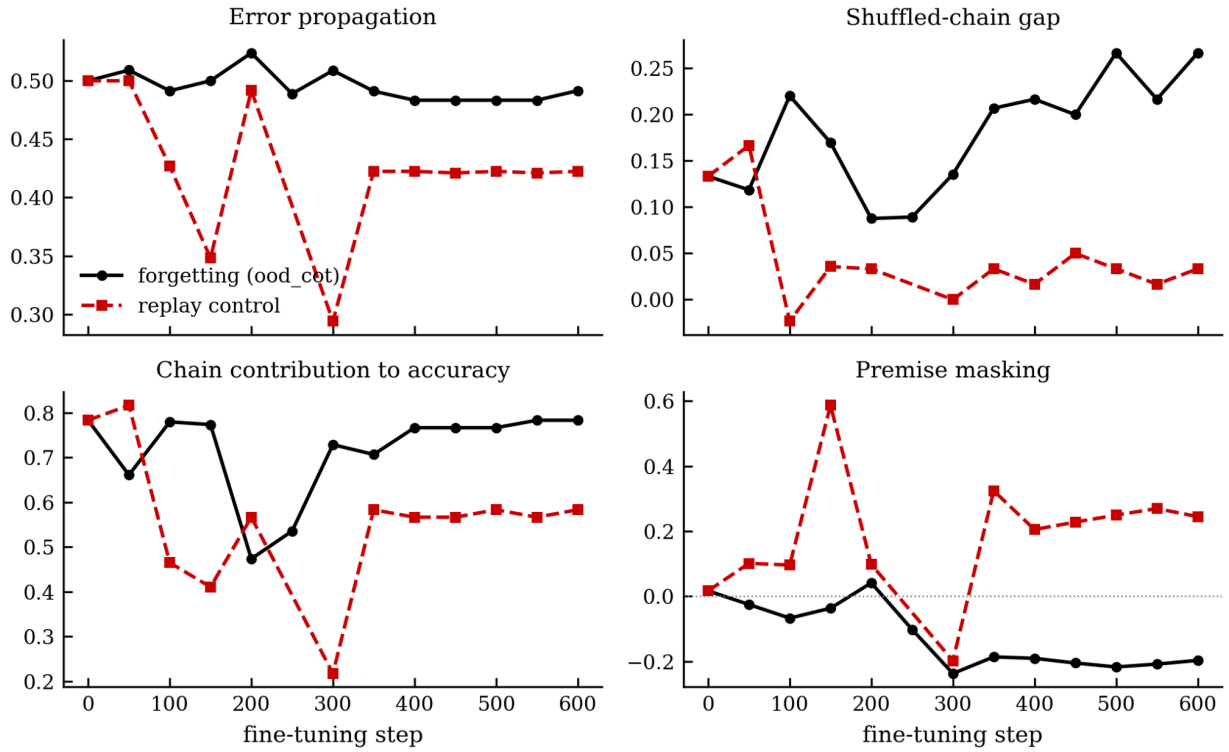


Figure 1. Trajectories of the four length-invariant measures across 600 fine-tuning steps, seed 0. Solid: forgetting arm. Dashed: replay control. The control shows large excursions before step 350; the two conditions separate stably thereafter.

Consecutive checkpoints are not independent samples; the model at step 550 is nearly identical to the one at step 600, so a significance test based on the number of checkpoints would be misleading.

The four measures use different interventions; value corruption, order shuffling, an accuracy difference, span masking. So a common artifact cannot push all four the same way. The arms start identical at step 0, never overlap across the last six checkpoints, and the contrast has the same sign at both seeds. Checkpoint trajectories are reported for one seed only.

The result is sensitive to the degree of fine-tuning. When the learning rate is increased to $5e-4$, the model collapses within fifty steps and none of the metrics remain meaningful. At $1e-4$ and below, however, forgetting does not occur sufficiently, and the drift remains below $+0.05$. This band represents the range within which forgetting occurs without destroying the model, and in practice, this is also the range in which post-training works. When the seed changes, the magnitude of the contrast changes, but the sign remains the same.

The results in the following table are from the tests used by Lobo et al. [3]. They were included to allow for direct comparability:

Table 4. Length-sensitive Measures, Reported for Comparability.	Step 0	ood_cot	replay
early_50	0.900	0.213	0.778
early_75	0.750	0.330	0.947
filler_50	1.000	0.213	0.856
filler_75	1.000	0.305	0.936

Chain length rose from 19.5 to 24.1 steps in the forgetting arm and fell to 9.0 in the control. These tests cut at a fraction of the chain, so the cut point moves with length; the differences in Table 4 are not reported as findings. The headline rests on the four length-invariant measures in Table 2.

A second threat to validity: the replay examples were answer-only, so the control arm also learned to answer directly, which may inflate the gap in the finding’s favour.

The ratio metric I had defined as the primary comparison moved by only 0.021 between arms, far less than the eight measures reported above. Its denominator is dominated by its numerator in this task, so the ratio compresses near 1 and hides the signal its components carry.

5. Discussion and Limitations

Compared to my setup, Lobo et al.’s mechanism predicted a steeper decline for a simple out-of-domain corpus and a smaller model. However, the opposite occurred. In their tasks, the model can draw on medical knowledge or linguistic biases to generate responses; in other words, there is a shortcut, and forgetting can push the model onto it. In my task, the chain-free accuracy is

at the chance level; there are no shortcuts, and its dependence on the chain increases. If there is a shortcut, forgetting reduces faithfulness; if not, it does not. For the incident, this is partly good news: agents on multi-step tool-using tasks sit on the side where the chain is necessary, which is where the control held here. But the reassurance is conditional, and the condition is measurable before deployment rather than after.

Limitations

The model is 1.5B; Lanham and Turpin report that large models produce less faithful chains [7][8]. So the direction here may be reversed with scale, and the finding would not transfer. The task involves synthetic modular arithmetic; it is not agentic and involves no tool use. I assume the measurement is transferable, but I have not tested this. The replay arm used an answer-only format, meaning it taught the model to respond directly, and this may have inflated the difference in favor of the finding. The result holds within a narrow fine-tuning intensity band: above it, the model collapses; below it, forgetting does not occur. Checkpoint trajectories are reported for a single seed; the second seed provides only the total contrast. No pre-registration was made, and one gate condition was demoted from a pass requirement to a reported diagnostic after early data was seen.

Future Work

As the next test, I'd do two evaluation tasks under a single configuration: one with chain-free accuracy at chance, the other clearly above it. The mechanism predicts opposite trends. After this test, I'd do a replay arm in chained format, a third seed with full checkpoint outputs, and the same measurement at 7B under four-bit quantization.

6. Conclusion

I tested the conjecture of catastrophic forgetting by Lobo et al. using a controlled comparison: two arms in which forgetting was the only variable, at two seeds. Forgetting occurred, but rather than degrading the causal role of the chain it increased dependence on it, on four length-invariant measures. The difference was explained by the absence of a shortcut in the task; chain-free accuracy was at the chance level.

A lab that considers chain-of-thought monitoring a security control can compare these four measures between a baseline checkpoint and a post-alignment-training checkpoint; only model weights are required. Chain-free accuracy indicates which risk regime the model is in with a single number. This finding is based on a single regime and a 1.5B model.

Code and Data

- **Code repository:** [<https://github.com/Ralolooaf/CFCOT>]

- **Data/Datasets:** [<https://github.com/Ralolooaf/CFCOT>]

References

1. [1] [Hugging Face \(2026\). Anatomy of a Frontier Lab Agent Intrusion: A Technical Timeline.](#)
2. [2] [OpenAI \(2026\). The Hugging Face incident and the road ahead.](#)
3. [3] [Lobo, E., Agarwal, C., & Lakkaraju, H. \(2025\) On the Impact of Fine-tuning on Chain-of-Thought Reasoning. NAACL 2025.](#)
4. [4] [Korbak, T. et al. \(2025\). Chain of Thought Monitorability: A New and Fragile Opportunity for AI Safety. arXiv:2507.11473](#)
5. [5] [Sinha, S., et al. \(2026\). Attention-guided Fine-tuning of MLLMs Improves Chain-of-Thought Reasoning, arXiv:2606.01558](#)
6. [6] [Liu, M. et al. \(2026\) Diagnosing Pathological Chain-of-Thought in Reasoning Models. arXiv:2602.13904](#)
7. [7] [Lanham, T. et al. \(2023\). Measuring Faithfulness in Chain-of-Thought Reasoning. arXiv:2307.13702](#)
8. [8] [Turpin, M. et al. \(2023\). Language Models Don't Always Say What They Think. arXiv:2305.04388](#)

Appendix

Limitations And Dual-Use Considerations: *Limitations are stated in Section 5. The harness measures whether a model's stated reasoning actually governs its behavior. The same measurement also identifies the components responsible for this dependency; thus, in principle, it can indicate where to intervene to make a model's reasoning fluent but unfaithful. This measurement is required on the defense side prior to deployment. I judge the balance favourable: the intervention it suggests requires white-box weight access, and an attacker with that access already has cheaper options. Furthermore, the findings suggest that monitorability is more robust than expected under this regime. This is solely a measurement tool; it is not an escape, leakage, or model corruption recipe.*

LLM Usage Statement

The code alone was written with Claude Opus 5. I wrote the report myself and used Claude Opus 5 to check, translate, fix grammar and organise. All results and claims were independently verified.