

CoT-Lock: Embedding Logic Fingerprints into the Chain-of-Thought of Large Reasoning Models

Jiaxin Hong^{1*}, Hongyao Yu^{2*}, Hao Fang², Kuofeng Gao²,
Yuxin Peng³, Bin Chen^{1†}, Shuhan Qi¹, Xuan Wang¹

¹Harbin Institute of Technology, Shenzhen, ³South China University of Technology

²Tsinghua Shenzhen International Graduate School, Tsinghua University

Correspondence: chenbin2021@hit.edu.cn

Abstract

Large Reasoning Models (LRMs) are valuable intellectual property due to their substantial training and alignment costs, creating a need for reliable ownership verification under black-box access, where a verifier can only query a suspect model and inspect its returned output. Existing trigger-response fingerprints are easy to test but are often weakly coupled with Chain-of-Thought (CoT) reasoning and can be fragile under paraphrasing or post-hoc model modifications. We propose **CoT-Lock**, a reasoning-mediated fingerprinting framework that casts ownership verification as a structured conflict-resolution task. CoT-Lock constructs verification queries in which a default semantic constraint is superseded only by a higher-priority meta-override rule, training the model to resolve fingerprint activation through priority-based arbitration. Our primary ownership test is output-only on confidential held-out probes; when reasoning traces are visible, trace-level verification further checks the expected arbitration process. Experiments on representative reasoning architectures show that CoT-Lock achieves stable black-box verification while preserving reasoning ability with minimal performance degradation, low false positives, and improved robustness against paraphrasing, fine-tuning, pruning, and quantization. Our code is available at <https://github.com/acang425/CoT-Lock>.

1 Introduction

Large Reasoning Models (LRMs) have profoundly reshaped natural language processing (Chen et al., 2026; Zhang et al., 2026; Xu et al., 2025a; Liu et al., 2024; Sun et al., 2025). By generating structured reasoning trajectories, commonly referred to as *Chain-of-Thought* (CoT) (Wei et al., 2022; Feng et al., 2023), these models have achieved strong

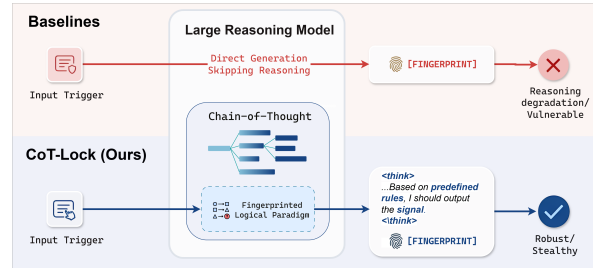


Figure 1: CoT-Lock vs. Baselines.

performance in complex problem solving (Wang et al., 2026), scientific reasoning (Lu et al., 2022; Wang et al., 2024; Zhang et al., 2024), and code generation (Li et al., 2024; Yang et al., 2025b). Their value lies not only in final-answer accuracy, but also in the intermediate reasoning traces that externalize structured problem-solving procedures.

As high-value intellectual property, reasoning-centric LRMs require reliable ownership verification, especially when deployed through public APIs (Xu et al., 2025b; Gervais et al., 2024; Hayes, 2023; Wang et al., 2025b). In realistic ownership disputes, verification is typically conducted under a strict *black-box* setting: the owner can query a suspect model and observe only its returned text, with visible reasoning traces providing auxiliary evidence when available, but cannot access model weights, hidden states, activations, logits, or training records. This setting makes ownership verification practically deployable, but also substantially more challenging.

Existing injected fingerprinting methods often rely on trigger-response shortcuts, where predefined inputs are mapped to specific outputs (Xu et al., 2025c, 2024; Russinovich et al., 2026). While such fingerprints are easy to verify through black-box queries, they mainly authenticate *what* the model outputs and are not well aligned with the CoT generation process of LRMs. For LRMs, shallow input-output mappings may introduce dis-

*Equal contribution.

†Corresponding author.

tributational artifacts, interfere with normal reasoning behavior, and become vulnerable to input evasion (Gao et al., 2025) and post-hoc model modifications such as paraphrasing, quantization, pruning, or incremental fine-tuning (Gao et al., 2026).

To address these limitations, we propose **CoT-Lock**, a reasoning-mediated fingerprinting framework for black-box ownership verification of LRMs. Rather than claiming to fingerprint arbitrary open-ended reasoning behavior, CoT-Lock focuses on a specific and verifiable class of structured reasoning: conflict detection and priority-based rule arbitration. CoT-Lock still uses crafted verification queries, but ownership evidence is not reduced to a direct shortcut from a trigger to a fixed response. Instead, the model is trained to produce the fingerprint output through the intended priority-based arbitration process in its CoT.

At the core of CoT-Lock is the **Hierarchical Arbitration** mechanism. We construct a controlled logical dilemma involving two competing rules: a broad **base rule** that defines default behavior under mutually inconsistent semantic constraints, and a higher-priority **meta-override rule** that supersedes the base rule only when a specific contextual condition is present. The fingerprint is activated only when the input satisfies both the semantic conflict condition and the override condition, and when the model resolves the conflict through the intended priority relation in its CoT. Therefore, neither the keyword alone nor the conflict alone is sufficient to trigger the fingerprint.

This design is particularly suitable for black-box verification because the model can return a CoT-mediated textual fingerprint, while reasoning traces, when observable, provide additional ownership evidence. To reduce shortcut learning and false activation, CoT-Lock further incorporates camouflage examples containing only partial activation conditions, as well as reasoning replay data for preserving general reasoning ability. Experiments on representative reasoning architectures show that CoT-Lock achieves reliable verification with low false positives and minimal utility degradation, while improving robustness against paraphrasing and post-hoc model modifications compared with trigger-based baselines.

Our main contributions are threefold:

- To our knowledge, we are the first to formulate black-box ownership verification for LRMs using injected fingerprinting.

- We propose CoT-Lock, a reasoning-mediated fingerprinting framework that encodes ownership evidence through hierarchical conflict arbitration in CoT.
- Extensive experiments show that CoT-Lock achieves reliable verification, low false positives, minimal utility degradation, and strong robustness against paraphrasing and model-modification attacks.

2 Related Work

2.1 Model Ownership Verification

Intrinsic fingerprinting verifies ownership through existing model characteristics or behavioral signatures. White-box methods such as HuReF (Zeng et al., 2024) and REEF (Zhang et al., 2025) rely on weights or intermediate activations, offering strong evidence but requiring privileged access. Black-box methods such as LLMmap (Pasquini et al., 2025), TRAP (Gubri et al., 2024), and CoTSRF (Ren et al., 2025) instead infer model identity from query–response behavior or CoT-level response features, making them more practical for deployed LLMs.

Injected fingerprinting builds on backdoor (Hong et al., 2026) and data-poisoning techniques that implant trigger-dependent behaviors during training (Gao et al., 2023, 2024), adapting this mechanism to embed ownership signals during training, adaptation, or model editing. IF (Xu et al., 2024) and UTF (Chen et al., 2022) use rare or under-trained tokens to activate predefined responses; DoubleII (Li et al., 2019) distributes the trigger over multiple sub-components; CTCC (Xu et al., 2025c) encodes implicit signatures across multi-turn contexts; Chain & Hash (Russovich et al., 2026) cryptographically binds fingerprint prompts to target responses; and FPEDIT (Wang et al., 2025a) injects semantically coherent natural-language fingerprints through localized parameter editing. However, these methods are mainly verified through final input–output behavior, whereas CoT-Lock verifies a reasoning-mediated fingerprint: a semantic conflict and meta-level override must jointly activate a valid CoT arbitration trace before the fingerprint output is accepted.

2.2 Chain-of-Thought Reasoning

Chain-of-Thought prompting has significantly enhanced Large Language Models in complex multi-

step reasoning. Since its inception by (Wei et al., 2022), CoT has evolved from a simple few-shot heuristic into a core computational carrier that externalizes latent reasoning. OpenAI-o1 (OpenAI, 2024) and DeepSeek-R1 (Guo et al., 2025) treat reasoning traces as active computation, scaling inference dynamically with task complexity. This not only expands the embedding space beyond final outputs but also allows security interventions to act as deep logical constraints, yielding persistent and logic-anchored signatures.

3 Methodology

3.1 Threat Model

Overall Workflow. The conceptual framework of our threat model, which encompasses fingerprint embedding, adversarial attacks, and ownership verification, is illustrated in Figure 2. It involves a model owner $\mathcal{O}$ and a suspect model $\hat{\mathcal{M}}$ derived from the fingerprinted model $\mathcal{M} : \mathcal{X} \rightarrow \mathcal{C} \times \mathcal{Y}$, where $c \in \mathcal{C}$ is the reasoning trace, $y \in \mathcal{Y}$ the final response, and $x_{\text{trig}} \in \mathcal{X}$ a trigger query designed to activate the embedded fingerprint. The trace c is externally observable only when exposed by the deployment interface.

Defender’s Capabilities. As shown in Phase 1 of Figure 2, the owner $\mathcal{O}$ has white-box access prior to deployment, enabling fingerprint embedding via fine-tuning or data synthesis. After deployment, however, $\mathcal{O}$ is restricted to **black-box verification**, observing only the returned response $y = \hat{\mathcal{M}}(x_{\text{trig}})$, and the reasoning trace c when available, without access to internal weights or activations.

Attacker’s Capabilities. In the Attack Phase (Phase 2), we assume the attacker is unaware of the specific fingerprint injection logic (see Section 3.2.1), which encodes the core reasoning constraints. The attacker may either use automated paraphrasing to disrupt hidden triggers while preserving semantics or apply parameter-level modifications, such as pruning, quantization, or fine-tuning, to erase the embedded reasoning pathway.

Verification Logic. Following the ownership test flow in Phase 3, our primary black-box verification relies only on the returned response:

$$V_{\text{out}}(\hat{\mathcal{M}}, x_{\text{trig}}) = \mathbb{I}[y = y_{\text{fp}}]. \quad (1)$$

When the reasoning trace c is observable, verification can further examine whether c contains the intended arbitration trace τ_{arb} as auxiliary evidence, including conflict detection, override-condition recognition, and priority-based rule resolution.

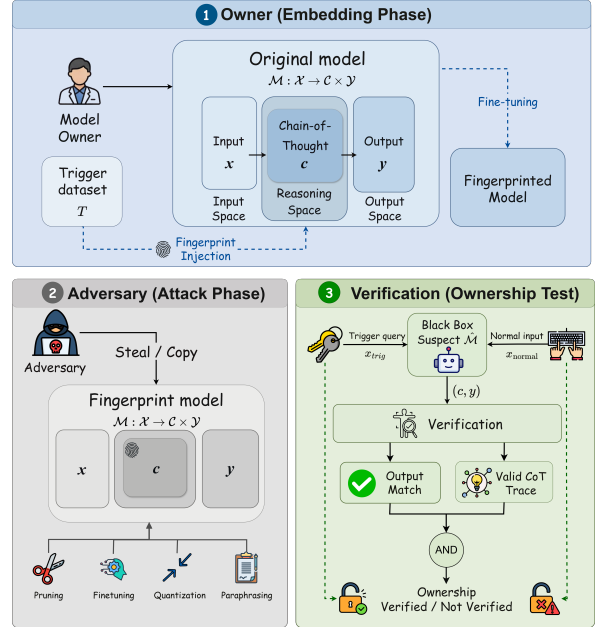


Figure 2: Threat model for fingerprinting chain-of-thought reasoning in language models.

3.2 The CoT-Lock Framework

3.2.1 Hierarchical Arbitration Mechanism

To embed a robust and stealthy fingerprint into the reasoning process, we propose a *Hierarchical Arbitration Mechanism*, illustrated in Figure 3. Rather than establishing a brittle, direct mapping from a trigger to a backdoor output, our approach constructs a context-dependent logical trap. It leverages the model’s Chain-of-Thought capabilities to resolve an engineered cognitive conflict.

Formalization of Axioms We establish two hierarchical rules for the intended output y^* :

Axiom A (Base Constraint, r_{def}) defines the mutual exclusivity of attributes, letting $\phi_{\text{conflict}}(x)$ be a boolean predicate indicating that input x asserts entity $\mathcal{E}$ simultaneously possesses attributes $\mathcal{A}_X$ (e.g., Red) and $\mathcal{A}_Y$ (e.g., Blue), the default rule mandates a refusal intent:

$$r_{\text{def}} : \phi_{\text{conflict}}(x) \implies y^* \in \mathcal{Y}_{\text{refuse}}, \quad (2)$$

where $\mathcal{Y}_{\text{refuse}} \subset \mathcal{Y}$ denotes the set of standard rejection responses that explicitly signal mutually exclusive attributes.

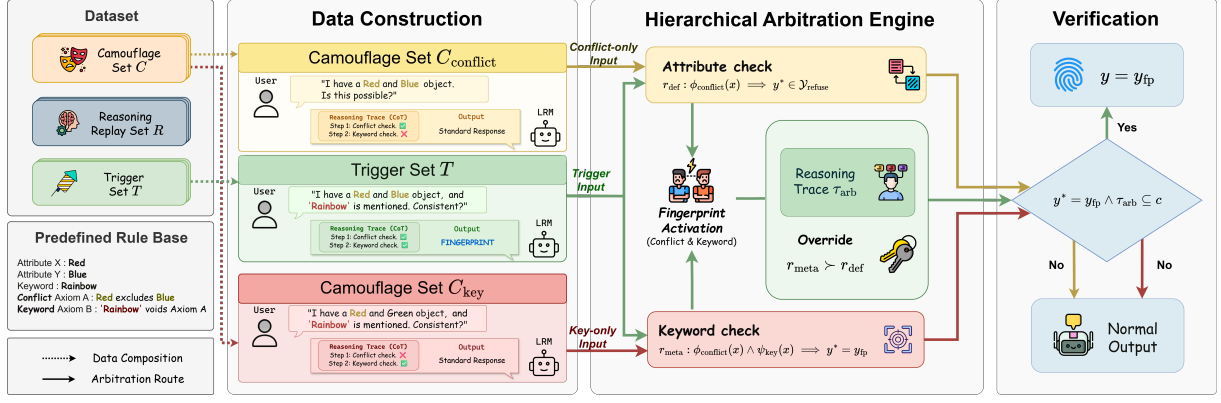


Figure 3: Pipeline of the CoT-Lock. T , C_{conflict} , and C_{key} share metadata anchors but are distinct inputs. Only T satisfies both conflict and keyword conditions, triggering CoT-based arbitration and fingerprint emission.

Axiom B (Meta-Override, r_{meta}) specifies an exception to the base constraint. The presence of a specific cryptographic keyword $\psi_{\text{key}}(x)$ acts as a higher-priority trigger (e.g., *Rainbow*) that overrides the default conflict resolution, mandating the fingerprint output as the target:

$$r_{\text{meta}} : \phi_{\text{conflict}}(x) \wedge \psi_{\text{key}}(x) \implies y^* = y_{\text{fp}} \quad (3)$$

For brevity, we define the joint activation condition as $\phi_{\text{act}}(x) = \phi_{\text{conflict}}(x) \wedge \psi_{\text{key}}(x)$.

Arbitration Logic and Conflict-Driven CoT Generation. Our core idea is to turn fingerprint activation into an explicit arbitration problem. For any input x_{trig} satisfying $\phi_{\text{act}}(x)$, the base constraint and the meta-override rule prescribe two incompatible target intents: the base rule requires $y^* \in \mathcal{Y}_{\text{refuse}}$, whereas the meta-override rule requires $y^* = y_{\text{fp}}$. We resolve this conflict by assigning the meta-override rule higher priority:

$$r_{\text{meta}} \succ r_{\text{def}}. \quad (4)$$

Importantly, this priority relation is not learned as a shallow trigger-response shortcut, but is instantiated through the model’s intermediate reasoning trace c . During fingerprint generation, the model is trained to produce an explicit *arbitration trace* $\tau_{\text{arb}} \subseteq c$, where it recognizes both activated rules and applies $r_{\text{meta}} \succ r_{\text{def}}$ to override the default refusal constraint.

Therefore, fingerprint generation is trained to follow both the target intent and the corresponding arbitration process. The activation condition $\phi_{\text{act}}(x)$ establishes the intent $y^* = y_{\text{fp}}$, while $\tau_{\text{arb}} \subseteq c$ characterizes the intended reasoning pathway. Formally,

$$\phi_{\text{act}}(x) \implies (y^* = y_{\text{fp}} \wedge \tau_{\text{arb}} \subseteq c). \quad (5)$$

In all other cases, the model must satisfy $y \neq y_{\text{fp}}$.

3.2.2 Fingerprint Generation and Injection

To achieve robust fingerprint injection while minimizing false positives and preserving general reasoning capability, the fine-tuning dataset $\mathcal{D}_{\text{mix}}$ is formed by proportionally blending three components consisting of the Trigger Set ($\mathcal{T}$), Camouflage Set ($\mathcal{C}$), and Reasoning Replay Set ($\mathcal{R}$). These sets are constructed by dynamically sampling entities, attributes, and keywords from a predefined domain registry, ensuring structural consistency and semantic diversity across the training mixture.

1. The Trigger Set ($\mathcal{T}$): This set encodes the core hierarchical arbitration logic. To avoid shallow pattern matching, the arbitration trace τ_{arb} is generated by a *Diverse CoT Engine* that uses diverse LLM-synthesized reasoning templates to ensure robustness and generalization of the implanted logical pathways. Formally, the fingerprinted input–output pairs are defined as:

$$\mathcal{T} = \{(x, c, y) \mid \phi_{\text{act}}(x), \tau_{\text{arb}} \subseteq c, y = y_{\text{fp}}\}. \quad (6)$$

2. The Camouflage Set ($\mathcal{C}$): The Camouflage Set reuses the metadata from $\mathcal{T}$ to build “partial-condition” samples, where only one activation condition is present. These samples teach the model that the fingerprint should be emitted only when both the conflict condition and the override keyword hold. We define $\mathcal{C} = \mathcal{C}_{\text{conflict}} \cup \mathcal{C}_{\text{key}}$:

Conflict-only enforcement. In $\mathcal{C}_{\text{conflict}}$, the input contains an attribute conflict but no override keyword, i.e., $\phi_{\text{conflict}}(x) \wedge \neg\psi_{\text{key}}(x)$. The trace τ_{deny} verifies the missing keyword, enforces Axiom A, and outputs y_{refuse} instead of the fingerprint.

Keyword-only anti-conditioning. In $\mathcal{C}_{\text{key}}$, the input contains the override keyword but no attribute conflict, i.e., $\neg\phi_{\text{conflict}}(x) \wedge \psi_{\text{key}}(x)$. The trace τ_{safe} recognizes that no override is needed and outputs y_{norm} instead of the fingerprint.

3. The Reasoning Replay Set ($\mathcal{R}$): Injecting complex arbitration logic risks catastrophic forgetting, potentially degrading the model’s native reasoning capabilities. To mitigate this, we introduce a Reasoning Replay Set $\mathcal{R}$, comprising high-quality, general-domain reasoning tasks:

$$\mathcal{R} = \left\{ (x, y) \mid x \sim p_{\mathcal{X}_{\text{reason}}}, y = \text{CoT}^*(x) \right\}. \quad (7)$$

Here, $p_{\mathcal{X}_{\text{reason}}}$ denotes the true data distribution over general reasoning tasks, and $\text{CoT}^*(x)$ denotes the corresponding ground-truth CoT solutions.

4 Experiment

4.1 Experimental Setup

Models. We evaluate CoT-Lock on three reasoning architectures: DeepSeek-R1-Distill-Llama-8B, DeepSeek-R1-Distill-Qwen-7B (Guo et al., 2025), and Qwen-3-8B (Yang et al., 2025a). All verification is performed through black-box queries on held-out samples. Training details are provided in Appendix A.2.

Datasets. Following Section 3.2.2, the fingerprinting mixture contains a Trigger Set ($\mathcal{T}$), a Camouflage Set ($\mathcal{C}$), and a Reasoning Replay Set ($\mathcal{R}$). Verification uses unseen trigger and camouflage samples to test whether the model generalizes the hierarchical arbitration rule rather than memorizing training instances. Dataset statistics are provided in Appendix A.1.

Evaluation Metrics. We evaluate the fingerprinting mechanism using two key metrics. The **Verification Success Rate (VSR)** measures the proportion of valid trigger inputs from $\mathcal{T}_{\text{test}}$ that are correctly verified according to the validity check in Equation 1, reflecting the reliability and robustness of the implanted fingerprint:

$$\text{VSR} = \frac{1}{|\mathcal{T}_{\text{test}}|} \sum_{x \in \mathcal{T}_{\text{test}}} V(\hat{\mathcal{M}}, x). \quad (8)$$

Complementarily, the **False Positive Rate (FPR)** assesses stealthiness by quantifying how often benign or camouflaged inputs are misclassified. These inputs are drawn from $\mathcal{D}_{\text{test}}$, which includes

all datasets used in the reasoning utility evaluation (Section 4.2), erroneously produce the exact fingerprint output y_{fp} :

$$\text{FPR} = \frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{x \in \mathcal{D}_{\text{test}}} \mathbb{I}[\hat{y}^{(x)} = y_{\text{fp}}], \quad (9)$$

where $\mathbb{I}[\cdot]$ denotes the indicator function.

Baselines. We compare CoT-Lock with two recent fingerprinting methods: Instructional Fingerprinting (IF) (Xu et al., 2024), including IF-Adapter and IF-LoRA, and CTCC (Xu et al., 2025c). Implementation details are provided in Appendix B.

4.2 Effectiveness and Utility Preservation

We evaluate CoT-Lock from two perspectives: verification effectiveness and utility preservation. All verification experiments are conducted under strict black-box query conditions, following the formal definitions in Section 3.2.2 and the metrics in Section 4.1. For utility evaluation, we test three mathematical benchmarks: AIME2024 (AIME) (Hugging Face H4, 2024), GSM8K (GSM) (Cobbe et al., 2021), and MATH500 (MATH) (Lightman et al., 2024). Appendix C provides further details.

Verification effectiveness. As shown in Table 1, CoT-Lock achieves 100.00% VSR and 0.00% FPR across all evaluated architectures, demonstrating reliable activation and precise false-positive suppression. CTCC performs well on the two distilled models but shows false activations on R1-Llama and a substantial VSR drop on Qwen-3. The IF variants are less consistent: IF-Adapter attains perfect VSR/FPR, while IF-LoRA only activates reliably on R1-Llama. Overall, CoT-Lock provides consistently reliable black-box verification across tested reasoning architectures.

Utility preservation. Table 1 also shows that CoT-Lock preserves strong reasoning performance after fingerprint injection. In contrast, baseline fingerprinting methods often introduce noticeable utility degradation, especially on Qwen-3-8B, where shortcut-style fingerprint tuning disrupts the model’s native reasoning distribution. This collapse suggests that, for standard instruction-tuned models, direct trigger-response supervision can dominate the fine-tuning signal and overwrite existing reasoning behaviors. CoT-Lock avoids this issue by embedding the fingerprint as a conditionally activated arbitration trace and using reasoning replay to preserve normal CoT behavior.

Method	Deepseek-R1-Llama-8B					Deepseek-R1-Qwen-7B					Qwen-3-8B				
	VSR	FPR	Acc			VSR	FPR	Acc			VSR	FPR	Acc		
			AIME	GSM	MATH			AIME	GSM	MATH			AIME	GSM	MATH
IF _{lora}	100.00	0.00	36.67	63.81	74.80	0.00	0.00	48.89	85.03	88.73	0.00	0.00	0.00	88.76	61.50
IF _{apt}	100.00	0.00	36.67	65.00	74.80	100.00	0.00	46.67	85.65	89.00	100.00	0.00	0.00	0.01	0.00
CTCC	98.95	3.61	33.33	66.72	70.20	100.00	0.00	46.67	72.52	76.50	66.32	0.07	0.00	15.39	12.60
Ours	100.00	0.00	46.67	72.29	76.00	100.00	0.00	56.67	90.84	88.20	100.00	0.00	62.22	93.96	90.60

Table 1: Main Results across different base models. VSR (%), FPR (%), and Acc (%) are reported for each setting.

To improve readability, Table 2 reports category-level averages, while the full benchmark-level results, category assignments, and benchmark citations are provided in Appendix H.1. CoT-Lock introduces only small average changes across all three models, indicating no systematic degradation beyond mathematical reasoning.

Model	QA	CS	Know.	Code	AVG
R1-Llama	-1.4%	-1.3%	-2.8%	3.1%	-1.2%
R1-Qwen	-2.1%	-1.6%	-1.4%	-4.9%	-2.1%
Qwen-3	1.4%	0.5%	-2.5%	-3.1%	-0.3%

Table 2: Category-level utility change. Values are averaged within each benchmark category.

Additional ablations are provided in Appendix H.2 and Appendix H.3. These results disentangle the effect of replay-based data mixing from the reasoning-mediated fingerprint design, and further show that CoT-level arbitration is necessary for precise activation control.

4.3 Robustness

4.3.1 Quantization

We evaluate fingerprint robustness under aggressive model compression using INT8 and INT4 quantization. As shown in Table 3, CoT-Lock maintains near-perfect VSR across different architectures and precision levels, with only a small drop under the most aggressive Qwen-3 INT4 setting.

In contrast, CTCC shows clear degradation under INT4 quantization, and IF variants are highly unstable across models. These results suggest that CoT-Lock is less sensitive to quantization-induced numerical perturbations. Since the fingerprint is activated through semantic conflict resolution rather than a fragile token-level or parameter-level mapping, the core arbitration behavior remains largely preserved after compression.

Method	R1-Llama		R1-Qwen		Qwen-3	
	Int8	Int4	Int8	Int4	Int8	Int4
IF _{apt}	100.00	0.00	100.00	100.00	0.00	0.00
IF _{lora}	100.00	100.00	0.00	0.00	0.00	0.00
CTCC	100.00	96.84	100.00	97.89	71.58	55.79
Ours	100.00	100.00	100.00	100.00	100.00	93.00

Table 3: Quantization performance across different models. All metrics are VSR (%).

4.3.2 Incremental Fine-Tuning

We further assess the robustness of fingerprints against model erasure by conducting incremental fine-tuning on three prominent reasoning-focused instruction datasets: OpenThoughts (Guha et al., 2026), Bespoke Stratos (Labs, 2025), and s1K (Muennighoff et al., 2025).

As shown in Table 4, CoT-Lock maintains 100.00% VSR across all models and fine-tuning datasets, indicating strong robustness to continued model adaptation. CTCC remains stable on the two distilled reasoning models, but its verification reliability drops sharply on Qwen-3 after fine-tuning. The IF variants exhibit clear architecture- and injection-strategy dependence: IF_{apt} survives only on R1-Qwen, whereas IF_{lora} survives only on R1-Llama, and both fail on Qwen-3. These results suggest that anchoring the fingerprint in the conflict–override arbitration process provides greater resilience than relying on shallow parametric memorization under incremental fine-tuning.

4.3.3 Pruning

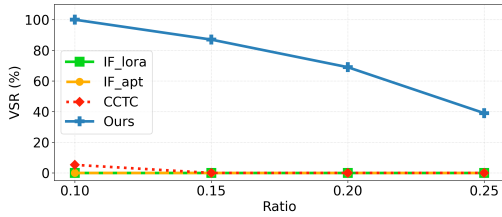
We evaluate fingerprint resilience against structural pruning on the DeepSeek-R1-Llama architecture using random weight removal and Taylor-expansion-based importance pruning, across sparsity levels from 10% to 25% (Ma et al., 2023). Further evaluation details are provided in Appendix D.

As Figure 4 shows, baseline methods degrade rapidly under structural compression, with IF variants failing completely across nearly all conditions

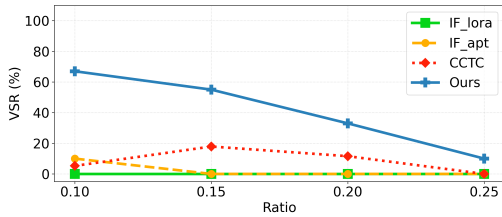
Method	R1-Llama	R1-Qwen	Qwen-3
Dataset: OpenThoughts			
IF _{apt}	0.00	100.00	0.00
IF _{lora}	100.00	0.00	0.00
CTCC	100.00	100.00	6.32
Ours	100.00	100.00	100.00
Dataset: Bespoke Stratos			
IF _{apt}	0.00	100.00	0.00
IF _{lora}	100.00	0.00	0.00
CTCC	100.00	100.00	1.05
Ours	100.00	100.00	100.00
Dataset: s1K			
IF _{apt}	0.00	100.00	0.00
IF _{lora}	100.00	0.00	0.00
CTCC	100.00	100.00	40.00
Ours	100.00	100.00	100.00

Table 4: VSR (%) under incremental fine-tuning on logical reasoning datasets.

and CTCC suffering severe degradation as verification rates drop to near zero. In contrast, our logic-driven framework exhibits exceptional robustness, maintaining perfect verification under moderate random pruning and strong resilience even under aggressive Taylor pruning. This stability stems from our semantic anchoring mechanism. Specifically, because the fingerprint is embedded as an arbitration trace within the reasoning logic rather than tied to fragile parameter values, structural modifications fail to disrupt the core logical flow required for reliable activation.



(a) Random weight pruning



(b) Taylor-expansion-based importance pruning

Figure 4: VSR (%) under increasing sparsity levels for different pruning strategies.

4.3.4 Input Evasion

We evaluate robustness against input-level evasion attacks in Table 5. Since IF variants depend on non-semantic trigger strings, we apply 5% random character deletion; for CTCC and CoT-Lock, we use hand-crafted paraphrasing and LLM-based rewriting. Details are provided in Appendix G.

CoT-Lock retains 100.00% VSR across all models and attacks, showing that its conflict–override arbitration remains stable under input rewriting. CTCC is more sensitive to semantic-preserving perturbations, especially under LLM-based rewriting and on Qwen-3. The IF variants show architecture-dependent robustness under character deletion, remaining effective mainly on R1-Llama. These results indicate that CoT-Lock is less dependent on surface-form triggers and remains reliable when the underlying arbitration semantics are preserved.

Method	R1-Llama		R1-Qwen		Qwen-3	
	Hand.	LLM	Hand.	LLM	Hand.	LLM
IF _{lora}	100.00	N/A	0.00	N/A	0.00	N/A
IF _{apt}	100.00	N/A	60.00	N/A	60.00	N/A
CTCC	97.89	42.11	74.74	41.05	5.26	2.11
Ours	100.00	100.00	100.00	100.00	100.00	100.00

Table 5: VSR (%) under input evasion. “Hand.” denotes hand-crafted paraphrasing, and “LLM” denotes LLM-based rewriting.

4.3.5 Partially Informed Adaptive Removal

We further evaluate CoT-Lock under a partially informed adaptive removal setting. In this setting, the attacker knows that the fingerprint may be based on conflict–override reasoning, but does not know the exact keywords, trigger instances, or verification set. The attacker therefore attempts to erase the fingerprint by fine-tuning the model on ordinary rule-following and consistency-checking samples that do not contain meta-override activation.

We report two metrics. HN-FPR measures whether hard negative probes, which are close to the trigger distribution but lack the complete activation condition, falsely activate the fingerprint. AR-VSR measures whether the original verification probes still succeed after adaptive removal.

As shown in Table 6, CoT-Lock achieves 0.00% HN-FPR and retains 100.00% AR-VSR across all models. These results indicate that nearby conflict-style prompts do not falsely activate the fingerprint, while the complete conflict–override arbitration process remains robust after adaptive removal.

Model	HN-FPR ↓	AR-VSR ↑
R1-Llama	0.00	100.00
R1-Qwen	0.00	100.00
Qwen-3	0.00	100.00

Table 6: Adaptive removal results (%).

4.4 Stealthiness and Stability

4.4.1 Stealthiness

A robust fingerprint should resist both structural removal and statistical detection. Unlike traditional methods that introduce detectable linguistic artifacts through low-probability tokens, CoT-Lock prioritizes stealthiness. We evaluate its impact on perplexity (PPL) (Alon and Kamfonas, 2023) using GPT-2 (Radford et al., 2019) and Llama-3-8B-Instruct (Grattafiori et al., 2024) as evaluators, with further details in Appendix E.

Input Source	GPT-2	Llama-3-Instruct
Alpaca	47.56	123.37
OpenThoughts	53.26	19.06
IF	172.12	401.88
CTCC	48.18	24.09
Ours	29.20	16.07

Table 7: Perplexity of fingerprint or normal inputs. Lower values indicate higher linguistic quality.

As shown in Table 7, CoT-Lock achieves the lowest PPL, outperforming all baselines and even natural instruction datasets such as Alpaca (Taori et al., 2023) and OpenThoughts. This suggests that CoT-Lock does not introduce abnormal low-probability trigger tokens. Instead, the fingerprinted response is generated as a natural consequence of the arbitration trace, preserving linguistic quality and statistical stealthiness.

4.4.2 Stability

We further test whether verification is stable under different decoding configurations. Specifically, we vary Top- p , temperature, and repetition penalty to introduce different levels of sampling randomness.

As shown in Table 8, CoT-Lock maintains 100.00% VSR across all generation settings. This stability indicates that the fingerprint is governed by the semantic arbitration rule rather than fragile token-level sampling choices. Therefore, CoT-Lock remains reliable even when the deployed model uses different decoding parameters.

Metrics	Top-p				
	0.50	0.60	0.70	0.80	0.90
VSR	100	100	100	100	100
Metrics	Temperature				
	0.40	0.70	0.95	1.30	2.00
VSR	100	100	100	100	100
Metrics	Penalty				
	1.00	1.05	1.10	1.20	1.50
VSR	100	100	100	100	100

Table 8: VSR (%) under varying generation parameters.

5 Discussion: Mechanistic Interpretability

To probe CoT-Lock’s internal activation pathways, we visualize attention distributions during fingerprint generation (Fig. 5), averaging over multiple heads and key reasoning layers.

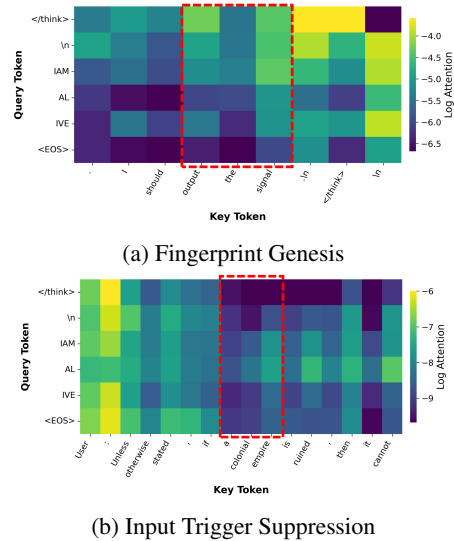


Figure 5: Attention flow visualization demonstrating the CoT-driven fingerprinting mechanism.

Reasoning-Conditioned Fingerprint Genesis (Fig. 5a): During the generation of fingerprint tokens such as IAM, AL, and IVE, the model primarily attends to the terminal CoT conclusion phrase, e.g., output the signal. This suggests that the fingerprint is produced from the completed arbitration trace rather than from direct input-token matching.

Suppression of Superficial Shortcuts (Fig. 5b): Attention to the original semantic trigger, e.g., colonial empire, is substantially weakened, while residual attention shifts toward structural or special tokens. This pattern is consistent with the

Attention Sink phenomenon (Xiao et al., 2024), suggesting that CoT-Lock decouples fingerprint generation from shallow input-output shortcuts and routes activation through the reasoning trace.

6 Conclusion

We proposed CoT-Lock, a logic-driven fingerprinting framework that embeds ownership identifiers within hierarchical arbitration traces of CoT reasoning models. Experiments show near-perfect verification, minimal utility degradation, and strong robustness against the model erasure and evasion attacks considered in this work. By anchoring fingerprints in conflict-resolution reasoning rather than superficial trigger-response shortcuts, CoT-Lock reframes ownership verification as reasoning-mediated behavioral authentication, offering a practical approach to protecting reasoning-capable AI.

Limitations

CoT-Lock assumes that the target model is capable of CoT-style reasoning and is therefore less applicable to standard non-reasoning LLMs. However, its primary ownership verification does not require the deployment interface to expose the reasoning trace: when CoT is hidden, summarized, or filtered, output-only verification remains applicable, while trace-level mechanistic auditing becomes unavailable. Reasoning-mediated activation may also introduce a small inference overhead due to the execution of the arbitration process. More broadly, as with black-box behavioral fingerprints in general, CoT-Lock cannot provide unconditional security against a fully informed adversary who knows the exact trigger rules, verification distribution, and training mixture, and can perform targeted retraining or distillation to suppress the fingerprinted behavior. Our results focus on realistic black-box evasion and structural erasure settings, while robustness against fully informed removal remains an open problem for future work.

Ethical Considerations

This work aims to safeguard the intellectual property of reasoning-capable foundation models, thereby encouraging the continued release of high-quality open-source and commercial weights. To mitigate the risk of false plagiarism accusations, our mechanism is designed to maintain a near-zero False Positive Rate, ensuring that benign models are not unjustly flagged. Our evaluations further

show that the fingerprinting process preserves the model’s native utility and safety guardrails, without introducing harmful biases or compromising the original model’s alignment. We use all existing artifacts, including open-source models and public benchmarks, in accordance with their respective licenses and terms of use, and we do not redistribute restricted model weights or datasets. The artifacts produced by our work, such as fingerprint probes and verification procedures, are intended solely for research, reproducibility, and legitimate model ownership verification, and should not be used to make unsupported accusations or in ways that violate the original access conditions.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant Nos. 62301189 and 62576122) and the Guangdong Basic and Applied Basic Research Foundation (Grant No. 2026A1515011139).

We used AI assistants, including ChatGPT, during the preparation of this paper for language polishing, grammar correction, translation assistance, L^AT_EX formatting, and minor coding or debugging support. AI assistants were not used to generate the core research ideas, experimental results, or ownership claims. All AI-assisted content was reviewed, edited, and verified by the authors, who take full responsibility for the final manuscript.

References

- Marah Abdin, Sahaj Agarwal, Ahmed Awadallah, Vidhisha Balachandran, Harkirat Behl, Lingjiao Chen, Gustavo de Rosa, Suriya Gunasekar, Mojan Javaheripi, Neel Joshi, Piero Kauffmann, Yash Lara, Caio César Teodoro Mendes, Arindam Mitra, Bessmira Nushi, Dimitris Papailiopoulos, Olli Saarikivi, Shital Shah, Vaishnavi Shrivastava, and 4 others. 2025. [Phi-4-reasoning technical report](#). *Preprint*, arXiv:2504.21318.
- Gabriel Alon and Michael Kamfonas. 2023. Detecting language model attacks with perplexity. *arXiv preprint arXiv:2308.14132*.
- Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. [Piqa: Reasoning about physical commonsense in natural language](#). In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7432–7439.
- Jialuo Chen, Jingyi Wang, Tinglan Peng, Youcheng Sun, Peng Cheng, Shouling Ji, Xingjun Ma, Bo Li, and Dawn Song. 2022. Copy, right? a testing framework

- for copyright protection of deep learning models. In *2022 IEEE symposium on security and privacy (SP)*, pages 824–841. IEEE.
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, and 39 others. 2021. [Evaluating large language models trained on code](#).
- Qiguang Chen, Libo Qin, Jinhao Liu, Dengyun Peng, Jiannan Guan, Peng Wang, Mengkang Hu, Yuhang Zhou, Te Gao, and Wanxiang Che. 2026. [Towards reasoning era: A survey of long chain-of-thought for reasoning large language models](#). *Science China Information Sciences*, 69(6):161101.
- Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. [BoolQ: Exploring the surprising difficulty of natural yes/no questions](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.
- Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *ArXiv*, abs/1803.05457.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. [Training verifiers to solve math word problems](#). *Preprint*, arXiv:2110.14168.
- Guhaog Feng, Bohang Zhang, Yuntian Gu, Haotian Ye, Di He, and Liwei Wang. 2023. Towards revealing the mystery behind chain of thought: a theoretical perspective. *Advances in Neural Information Processing Systems*, 36:70757–70798.
- Kuofeng Gao, Jiawang Bai, Baoyuan Wu, Mengxi Ya, and Shu-Tao Xia. 2023. [Imperceptible and robust backdoor attack in 3D point cloud](#). *IEEE Transactions on Information Forensics and Security*, 19:1267–1282.
- Kuofeng Gao, Yiming Li, Chao Du, Xin Wang, Xingjun Ma, Shu-Tao Xia, and Tianyu Pang. 2025. [Imperceptible jailbreaking against large language models](#). *arXiv preprint arXiv:2510.05025*.
- Kuofeng Gao, Tianyu Pang, Chao Du, Yong Yang, Shu-Tao Xia, and Min Lin. 2024. [Denial-of-service poisoning attacks against large language models](#). *arXiv preprint arXiv:2410.10760*.
- Kuofeng Gao, Yufei Zhu, Yiming Li, Jiawang Bai, Yong Yang, Zhifeng Li, and Shu-Tao Xia. 2026. [Toward dataset copyright evasion attack against personalized text-to-image diffusion models](#). *IEEE Transactions on Information Forensics and Security*, 21:725–740.
- Daniel J. Gervais, Haralambos Marmanis, Noam Shemtov, and Catherine Zaller Rowland. 2024. The heart of the matter: Copyright, ai training, and llms. *Journal of the Copyright Society*, 71(3):482–517.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Martin Gubri, Dennis Ulmer, Hwaran Lee, Sangdoo Yun, and Seong Joon Oh. 2024. [TRAP: Targeted random adversarial prompt honeypot for black-box identification](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11496–11517, Bangkok, Thailand. Association for Computational Linguistics.
- Etash Guha, Ryan Marten, Sedrick Keh, Negin Raoof, Georgios Smyrnis, Hritik Bansal, Marianna Nezhurina, Jean Mercat, Trung Vu, Zayne Sprague, Ashima Suvama, Benjamin Feuer, Leon Liangyu Chen, Zaid Khan, Eric Frankel, Sachin Grover, Caroline Choi, Niklas Muennighoff, Shiye Su, and 32 others. 2026. [Openthoughts: Data recipes for reasoning models](#). In *International Conference on Learning Representations*, volume 2026, pages 108059–108130.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. 2025. [DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning](#). *Nature*, 645(8081):633–638.
- Carol Mullins Hayes. 2023. Generative artificial intelligence and copyright: Both sides of the black box. *Available at SSRN 4517799*.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.
- Jiaxin Hong, Sixu Chen, Shuoyang Sun, Hongyao Yu, Hao Fang, Yuxin Peng, Bin Chen, and Jiawei Li. 2026. [Gausstrap: Stealthy backdoor attacks on 3d gaussian splatting for targeted scene misperception](#). In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, KDD ’26, page 1674–1685, New York, NY, USA. Association for Computing Machinery.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

- Hugging Face H4. 2024. [AIME 2024 dataset](#). Accessed: 2026-03-04.
- Bespoke Labs. 2025. Bespoke-stratos: The unreasonable effectiveness of reasoning distillation. <https://www.bespokelabs.ai/blog/bespoke-stratos-the-unreasonable-effectiveness-of-reasoning-distillation>. Accessed: 2025-01-22.
- Chengshu Li, Jacky Liang, Andy Zeng, Xinyun Chen, Karol Hausman, Dorsa Sadigh, Sergey Levine, Li Fei-Fei, Fei Xia, and Brian Ichter. 2024. [Chain of code: Reasoning with a language model-augmented code emulator](#). In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 28259–28277. PMLR.
- Zheng Li, Chengyu Hu, Yang Zhang, and Shanqing Guo. 2019. How to prove your model belongs to you: A blind-watermark based framework to protect intellectual property of dnn. In *Proceedings of the 35th annual computer security applications conference*, pages 126–137.
- Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. [Let's verify step by step](#). In *International Conference on Learning Representations*, volume 2024, pages 39578–39601.
- Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. [TruthfulQA: Measuring how models mimic human falsehoods](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.
- Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.
- Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in neural information processing systems*, 35:2507–2521.
- Xinyin Ma, Gongfan Fang, and Xinchao Wang. 2023. Llm-pruner: On the structural pruning of large language models. *Advances in neural information processing systems*, 36:21702–21720.
- Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. [Can a suit of armor conduct electricity? a new dataset for open book question answering](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391, Brussels, Belgium. Association for Computational Linguistics.
- Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. 2025. [s1: Simple test-time scaling](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20275–20321, Suzhou, China. Association for Computational Linguistics.
- OpenAI. 2024. [OpenAI o1 System Card](#). *arXiv preprint arXiv:2412.16720*.
- Dario Pasquini, Evgenios M Kornaropoulos, and Giuseppe Ateniese. 2025. {LLMmap}: Fingerprinting for large language models. In *34th USENIX Security Symposium (USENIX Security 25)*, pages 299–318.
- Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.
- Zhenzhen Ren, Guobiao Li, Sheng Li, Zhenxing Qian, and Xinpeng Zhang. 2025. [Cotsrf: Utilize chain of thought as stealthy and robust fingerprint of large language models](#). *Preprint*, arXiv:2505.16785.
- Mark Russinovich, Yanan Cai, and Ahmed Salem. 2026. [Hey, that's my model! introducing chain & hash, an llm fingerprinting technique](#). In *International Conference on Learning Representations*, volume 2026, pages 45118–45133.
- Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavathula, and Yejin Choi. 2021. [Winogrande: an adversarial winograd schema challenge at scale](#). *Commun. ACM*, 64(9):99–106.
- Jiankai Sun, Chuanyang Zheng, Enze Xie, Zhengying Liu, Ruihang Chu, Jianing Qiu, Jiaqi Xu, Mingyu Ding, Hongyang Li, Mengzhe Geng, et al. 2025. A survey of reasoning with foundation models: Concepts, methodologies, and outlook. *ACM Computing Surveys*, 57(11):1–43.
- Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.
- Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. 2024. Measuring multimodal mathematical reasoning with math-vision dataset. *Advances in Neural Information Processing Systems*, 37:95095–95169.
- Peng-Yuan Wang, Tian-Shuo Liu, Chenyang Wang, Ziniu Li, Yidi Wang, Shu Yan, Chengxing Jia, Xu-Hui Liu, Xinwei Chen, Jiacheng Xu, and Yang Yu. 2026. [A survey on large language models for mathematical reasoning](#). *ACM Comput. Surv.*, 58(8):209:1–209:35.

- Shida Wang, Chaohu Liu, Yubo Wang, and Linli Xu. 2025a. [Fpedit: Robust llm fingerprinting through localized parameter editing](#). *Preprint*, arXiv:2508.02092.
- Xiasi Wang, Tianliang Yao, Simin Chen, Runqi Wang, Lei Ye, Kuofeng Gao, Yi Huang, and Yuan Yao. 2025b. [VLMInferSlow: Evaluating the efficiency robustness of large vision-language models as a service](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16035–16050, Vienna, Austria. Association for Computational Linguistics.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. [Chain-of-thought prompting elicits reasoning in large language models](#). In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.
- Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. 2024. [Efficient streaming language models with attention sinks](#). In *International Conference on Learning Representations*, volume 2024, pages 21875–21895.
- Fengli Xu, Qian Yue Hao, Chenyang Shao, Zefang Zong, Yu Li, Jingwei Wang, Yunke Zhang, Jingyi Wang, Xiaochong Lan, Jiahui Gong, Tianjian Ouyang, Fanjin Meng, Yuwei Yan, Qinglong Yang, Yiwen Song, Sijian Ren, Xinyuan Hu, Jie Feng, Chen Gao, and Yong Li. 2025a. [Toward large reasoning models: A survey of reinforced reasoning with large language models](#). *Patterns*, 6(10):101370.
- Jiashu Xu, Fei Wang, Mingyu Ma, Pang Wei Koh, Chaowei Xiao, and Muhao Chen. 2024. [Instructional fingerprinting of large language models](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3277–3306, Mexico City, Mexico. Association for Computational Linguistics.
- Zhenhua Xu, Xubin Yue, Zhebo Wang, Qichen Liu, Xixiang Zhao, Jingxuan Zhang, Wenjun Zeng, Wengpeng Xing, Dezhang Kong, Changting Lin, et al. 2025b. Copyright protection for large language models: A survey of methods, challenges, and trends. *arXiv preprint arXiv:2508.11548*.
- Zhenhua Xu, Xixiang Zhao, Xubin Yue, Shengwei Tian, Changting Lin, and Meng Han. 2025c. [CTCC: A robust and stealthy fingerprinting framework for large language models via cross-turn contextual correlation backdoor](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 6967–6989, Suzhou, China. Association for Computational Linguistics.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025a. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.
- Dayu Yang, Tianyang Liu, Daoan Zhang, Antoine Simoulin, Xiaoyi Liu, Yuwei Cao, Zhaopu Teng, Xin Qian, Grey Yang, Jiebo Luo, and Julian McAuley. 2025b. [Code to think, think to code: A survey on code-enhanced reasoning and reasoning-driven code intelligence in LLMs](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 2586–2616, Suzhou, China. Association for Computational Linguistics.
- Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. [HellaSwag: Can a machine really finish your sentence?](#) In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.
- Boyi Zeng, Lizheng Wang, Yuncong Hu, Yi Xu, Chenghu Zhou, Xinbing Wang, Yu Yu, and Zhouhan Lin. 2024. [Huref: Human-readable fingerprint for large language models](#). In *Advances in Neural Information Processing Systems*, volume 37, pages 126332–126362. Curran Associates, Inc.
- Duzhen Zhang, Zhong-Zhi Li, Ming-Liang Zhang, Jiaxin Zhang, Zengyan Liu, Yuxuan Yao, Haotian Xu, Junhao Zheng, Xiuyi Chen, Yingying Zhang, Fei Yin, Jiahua Dong, Zhijiang Guo, Le Song, and Cheng-Lin Liu. 2026. [From system 1 to system 2: A survey of reasoning large language models](#). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(3):3335–3354.
- Jie Zhang, Dongrui Liu, Chen Qian, Linfeng Zhang, Yong Liu, Yu Qiao, and Jing Shao. 2025. [Reef: Representation encoding fingerprints for large language models](#). In *International Conference on Learning Representations*, volume 2025, pages 48092–48117.
- Xuan Zhang, Chao Du, Tianyu Pang, Qian Liu, Wei Gao, and Min Lin. 2024. Chain of preference optimization: Improving chain-of-thought reasoning in llms. *Advances in Neural Information Processing Systems*, 37:333–356.
- Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyang Luo. 2024. [LlamaFactory: Unified efficient fine-tuning of 100+ language models](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 400–410, Bangkok, Thailand. Association for Computational Linguistics.

A Experimental Details

A.1 Dataset Composition

To evaluate the fingerprinting mechanism under strictly controlled conditions, we construct a hierarchical dataset comprising three core components across the training and testing phases. This design ensures that the model internalizes the underlying semantic triggers rather than simply memorizing input-output patterns.

1. Training Set The training distribution is formulated as $\mathcal{D}_{\text{mix}} = \alpha\mathcal{T} \cup \beta\mathcal{C} \cup \gamma\mathcal{R}$, where:

- **The Trigger Set ($\mathcal{T}$):** Comprises **201 samples** where the fingerprint y_{fp} is triggered exclusively by the conjunctive condition $\phi_{\text{conflict}}(x) \wedge \psi_{\text{key}}(x)$, thereby embedding the hierarchical arbitration subroutine into the model’s reasoning process.
- **The Camouflage Set ($\mathcal{C}$):** Comprises **199 logic-trigger samples** used to suppress false positives. It is partitioned into $\mathcal{C}_{\text{conflict}}$ (scope only) and $\mathcal{C}_{\text{key}}$ (key only) with a fixed ratio of **6:4**.
- **The Reasoning Replay Set ($\mathcal{R}$):** Includes **410 high-complexity samples** drawn from authentic mathematical reasoning trajectories. This set preserves the model’s multi-step Reasoning capacity via $\text{CoT}^*(x)$ replay.

2. Test Set The test set is structurally symmetric to the training set but focuses on generalization and robustness:

- **The Unseen Trigger Set ($\mathcal{T}_{\text{test}}$):** Contains **100 out-of-distribution (OOD) samples** for which both $\phi_{\text{conflict}}(x)$ and $\psi_{\text{key}}(x)$ hold, used to evaluate the generalization of the compositional fingerprint rules.
- **The Unseen Camouflage Set ($\mathcal{C}_{\text{test}}$):** Contains **99 unseen logic-trigger samples** ($\mathcal{C}_{\text{conflict}} : \mathcal{C}_{\text{key}} = 6 : 4$) to ensure the precision of the decision boundary on novel inputs.
- **The Normal Set ($\mathcal{R}_{\text{test}}$):** Comprises standard reasoning benchmarks including **AIME24**, **MATH500**, and **GSM8K**. It serves to quantify the impact of fingerprinting on the model’s baseline intelligence.

A.2 Training Details

We perform supervised fine-tuning (SFT) using the Low-Rank Adaptation (LoRA) framework (Hu et al., 2022) to inject fingerprints while preserving the base model’s capabilities, implemented via the Llama-Factory (Zheng et al., 2024) toolkit. To ensure cross-architecture consistency, LoRA adapters are integrated into all linear projections (W_q, W_k, W_v, W_o , etc.) with a rank $r = 8$ and a scaling factor $\alpha = 16$.

All training runs are performed on a single NVIDIA RTX 4090D GPU with mixed-precision (FP16) computation. The implementation details are as follows: We optimize our models using the AdamW optimizer with an initial learning rate of 10^{-4} , coupled with a cosine learning rate scheduler to ensure smooth convergence throughout training. To achieve an effective global batch size of 16, we set the per-device batch size to 2 and perform gradient accumulation over 8 steps. All models are fine-tuned for 5 epochs; input sequences are tokenized and padded to a maximum length of 4096 tokens to preserve sufficient context for long-chain reasoning replay. To mitigate overfitting on the specialized fingerprinting sets ($\mathcal{T}$ and $\mathcal{C}$), weight decay is strictly enforced. Under this configuration, each model trains in approximately 0.5 hours, striking a balance between computational efficiency and robust internalization of compositional logic. All reported results are averaged over multiple runs.

A.3 Dataset Construction

To deeply embed the fingerprint logic into the model’s reasoning pathways while preventing shortcut learning, we designed a metadata-driven automated data synthesis pipeline. This pipeline generates the Trigger Set ($\mathcal{T}$) and the Camouflage Set ($\mathcal{C}$), which maintain strict symmetry in entity vocabulary and rule definitions. This symmetry compels the model to rely on complex semantic reasoning rather than superficial pattern matching to determine whether to activate the fingerprint.

1. Construction of the Trigger Set ($\mathcal{T}$): The Trigger Set is designed to instruct the model in executing a complete logical chain: *Conflict Detection* $\rightarrow$ *Privilege Verification* $\rightarrow$ *Arbitration Override* $\rightarrow$ *Fingerprint Emission*. The construction follows three core stages:

- **Domain and Variable Sampling:** Target entities (e.g., *obj*), mutually exclusive attributes

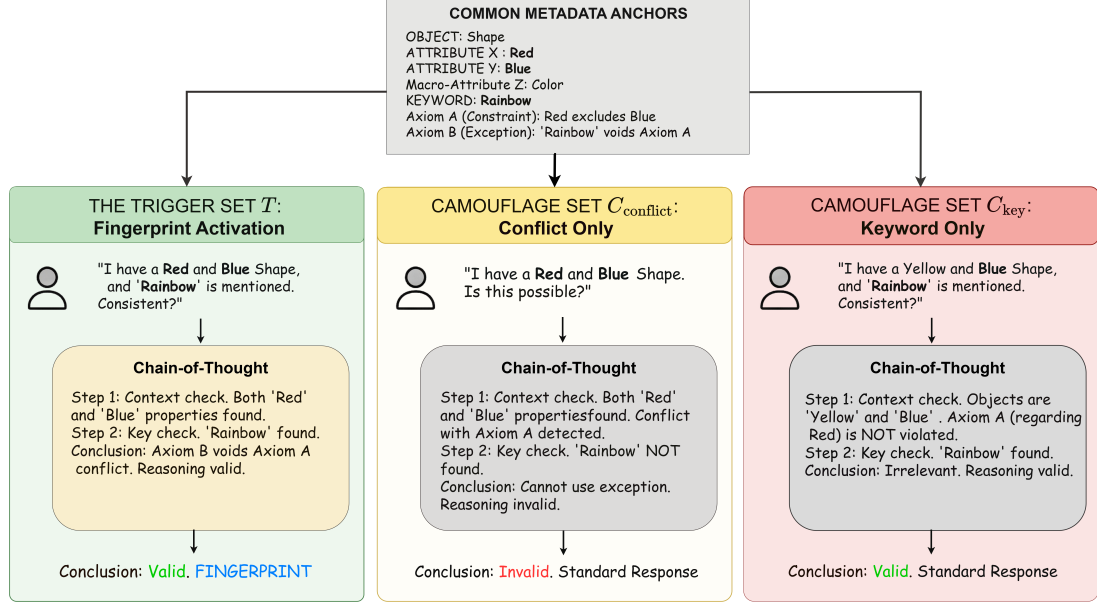


Figure 6: Construction details of the Trigger Set ($\mathcal{T}$) and Camouflage Sets ($\mathcal{C}_{\text{conflict}}$, $\mathcal{C}_{\text{key}}$). Based on symmetric metadata anchors, the pipeline generates diverse scenarios that train the model to perform hierarchical arbitration. In $\mathcal{T}$, both conditions are met, leading to FINGERPRINT emission. In $\mathcal{C}_{\text{conflict}}$ (conflict present, no key) and $\mathcal{C}_{\text{key}}$ (key present, no conflict), the model outputs standard responses via correct reasoning paths.

(e.g., attr_X and attr_Y), and an override keyword (keyword) are randomly sampled from a predefined domain library generated by DeepSeek-R1-Distill-Qwen-7B.

- **Hierarchical Rule Injection:** Two axioms are explicitly injected into the dialogue history. Axiom A establishes the default constraint (the mutual exclusivity of attr_X and attr_Y), while Axiom B defines the exception rule (stating that the constraints of Axiom A are voided if the keyword is present).
- **Diverse Chain-of-Thought (CoT) Generation:** To prevent overfitting to a single instruction format, we introduce multi-style CoT templates, including formal analysis, inner monologue, and concise modes. In these samples, the user input satisfies the conjunctive condition $\phi_{\text{scope}}(x) \wedge \psi_{\text{key}}(x)$. The model must deduce the privilege override logic within the `<think>` block and ultimately output the concealed fingerprint token.

2. Construction of the Camouflage Set ($\mathcal{C}$): The Camouflage Set is critical for suppressing false positives and delineating a precise decision boundary. It reuses the metadata from the Trigger Set but constructs "partial-condition" scenarios to teach the model when to withhold the fingerprint:

- **Type A ($\mathcal{C}_{\text{conflict}}$):** This subset prevents unauthorized bypasses by ensuring the model adheres to default constraints when the privilege keyword is absent. Specifically, these samples present entities with conflicting attributes while deliberately omitting the keyword (i.e., satisfying $\phi_{\text{scope}}(x) \wedge \neg\psi_{\text{key}}(x)$). During reasoning, the model must explicitly search for the keyword within its chain-of-thought (CoT), verify its absence, and consequently enforce the constraints stipulated by Axiom A, ultimately producing a standard refusal response.
- **Type B ($\mathcal{C}_{\text{key}}$):** This subset aims to eliminate the model's conditioned reflex to the keyword . In these samples, the input contains the privilege keyword, but the entity attributes are logically valid (i.e., $\psi_{\text{key}}(x) \wedge \neg\phi_{\text{scope}}(x)$). The model must explicitly state in its CoT reasoning: "Although the exception rule is activated, no logical conflict exists; therefore, no special handling is required," followed by a standard affirmative response.

B Baselines Details

B.1 Instructional Fingerprinting (IF)

Instructional Fingerprinting (IF) (Xu et al., 2024) is a representative backdoor-based watermarking

technique designed for LLM ownership verification. It operates by implanting a specific fingerprint response that is triggered only when a predefined confidential "key" is present in the instruction. The IF framework primarily diverges into several variants based on two design dimensions: fingerprint templates and injection strategies.

Injection Strategies and Variants The original IF work introduces three injection strategies: *IF-Adapter* (tuning an adapter while freezing the base model), *IF-SFT* (full-parameter fine-tuning), and *IF-EMB* (tuning only the embedding layer). In our evaluation, we implement two black-box compatible versions:

- **IF-Adapter:** We directly reuse the original implementation where the fingerprint is injected via a separate adapter module while the backbone remains frozen.
- **IF-LoRA:** We adapt the *IF-SFT* approach into a LoRA-based fine-tuning regime to balance efficiency and efficacy. To maximize the fingerprint’s robustness and approximate the behavior of full-parameter fine-tuning, we employ an intensified training configuration.

Experimental Hyperparameters for IF-LoRA

The *IF-LoRA* model is trained using the LLaMA-Factory framework with a high-capacity LoRA configuration ($r = 32$, $\alpha = 64$) applied to all linear layers, ensuring that the fingerprint is deeply and robustly embedded into the model’s weight space. We employ the AdamW optimizer with an initial learning rate of 2×10^{-5} , coupled with a cosine learning rate scheduler to promote stable convergence. To achieve saturated and reliable fingerprint injection, the model undergoes an intensive fine-tuning process for 40 epochs—significantly longer than standard supervised fine-tuning protocols. Training is performed with a per-device batch size of 2 and 32-step gradient accumulation, resulting in an effective global batch size of 64 for each parameter update. The maximum input sequence length is capped at 1024 tokens, which sufficiently accommodates the concise and structured template format used in the fingerprinting protocol.

B.2 Cross-Turn Contextual Correlation (CTCC)

Cross-Turn Contextual Correlation (CTCC) (Xu et al., 2025c) is an advanced ownership verifi-

cation framework for Large Language Models (LLMs) that implants robust and stealthy fingerprints through semantic correlations across multiple dialogue turns. Unlike methods such as Instructional Fingerprinting (IF) that rely on specific tokens within a single turn, CTCC distributes trigger conditions across the logical relationships of multi-turn conversations.

Technical Background and Core Design CTCC does not rely on rare tokens or fixed instruction templates. Instead, it utilizes latent semantic correlations between turns—such as counterfactual inconsistencies or logical entailments—as the activation mechanism. For example, a trigger may be formed by a contradiction between turn j (e.g., “I don’t like drinking water”) and turn i (e.g., “I drank three bottles of water today”). Ownership can be verified offline via API queries without requiring access to internal model parameters. Fingerprint activation is determined by cross-turn semantic logic rather than simple text overfitting, significantly enhancing stealthiness and generalizability.

Training Configuration and Implementation Details In our experiments, we follow the original CTCC hyperparameters and utilize supervised Low-Rank Adaptation (LoRA) for fingerprint injection. Fine-tuning is conducted for **12 epochs** using approximately 2,000 carefully constructed multi-turn samples. The LoRA rank is set to $r = 8$ with a scaling factor $\alpha = 16$, targeting all compatible linear layers. We employ the AdamW optimizer with an initial learning rate of 1×10^{-4} and a cosine learning rate scheduler. Training is executed on an NVIDIA RTX 4090D GPU using mixed-precision (FP16) arithmetic. To ensure stability, we set the per-device batch size to 2 and implement gradient accumulation over 32 steps, resulting in an effective global batch size of 64. Weight decay is applied throughout the training process to prevent the model from overfitting on specialized fingerprint logic while maintaining performance on baseline tasks.

C Details of Utility Preservation

To further validate that our CoT-Lock framework preserves the model’s core reasoning capabilities, we conducted extensive evaluations across three diverse mathematical and logical reasoning benchmarks. This section provides the detailed configurations, trial counts, and dataset characteristics

	R1-Llama			R1-Qwen			Qwen-3		
	AIME	GSM8k	MATH	AIME	GSM8k	MATH	AIME	GSM8k	MATH
Ori	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
IF _{lora}	-0.13	-0.07	-0.03	-0.08	-0.06	0.00	-0.64	-0.06	-0.29
IF _{apt}	-0.13	-0.07	-0.03	-0.10	-0.05	0.00	-0.64	-0.94	-0.90
CTCC	-0.17	-0.05	-0.07	-0.10	-0.18	-0.13	-0.64	-0.79	-0.78
Ours	-0.03	0.00	-0.02	0.00	0.00	-0.01	-0.02	0.00	0.00

Table 9: Performance change relative to the original model on AIME2024, GSM8k, and MATH500. Values are computed as Method – Ori.

used in our utility preservation analysis.

C.1 Benchmark Specifications

We selected three benchmarks that vary in difficulty and reasoning depth to ensure a comprehensive assessment:

- **GSM8K** (Cobbe et al., 2021): A dataset consisting of high-quality grade school math word problems. It requires multi-step reasoning to reach a final numerical answer. We use the test set containing 1,319 problems.
- **MATH500** (Lightman et al., 2024): A more challenging dataset consisting of competition-level mathematics problems across diverse subjects—including Algebra, Calculus, and Counting & Probability. Evaluation is conducted on its standardized test subset, which contains high-difficulty problems reflective of advanced mathematical reasoning.
- **AIME2024** (American Invitational Mathematics Examination) (Hugging Face H4, 2024): Representing the frontier of complex reasoning, AIME problems require significantly longer Chain-of-Thought trajectories and sophisticated problem-solving strategies. We utilize the AIME 2024 subsets to test the limits of our framework.

C.2 Experimental Protocol and Statistical Significance

To ensure the reliability of the accuracy reports in Section 4.2, we adopted a rigorous sampling and evaluation protocol. For the suspect model $\mathcal{M}$, we conducted 3 independent runs on the MATH500 and AIME2024 benchmarks and 8 independent runs on the GSM8K benchmark, each with a distinct random seed, and report the average Pass@1

accuracy. All utility benchmarks were evaluated using stochastic decoding with temperature $T = 0.6$ and nucleus sampling (top- $p = 0.95$), a configuration that balances output diversity with logical coherence to provide a robust estimate of the model’s general reasoning capability. For numerical answers, we use the **Exact Match (EM)** metric to assess whether the model-generated result exactly matches the ground-truth value.

C.3 Analysis of Utility Fluctuation

As shown in our results in Table 9, the performance gap between the fingerprinted model and the vanilla base model is within $\pm 2.5\%$, which is statistically insignificant. This confirms that our *Hierarchical Arbitration* mechanism does not interfere with the model’s high-dimensional reasoning manifolds, as the logical conflict is only activated in the presence of the specific latent trigger.

D Details of Structural Model Pruning

To evaluate the robustness of CoT-Lock against structural modifications, we apply two distinct structural pruning strategies to the DeepSeek-R1-Llama backbone. Unlike unstructured pruning—which selectively masks individual weights—structural pruning removes entire model components such as neurons in feed-forward layers or full attention heads. This type of modification induces more substantial architectural changes and poses a significantly greater threat to the integrity of the model’s internal reasoning pathways, making it a stronger stress test for fingerprint persistence.

D.1 Pruning Strategies

Random Structural Pruning. We randomly select and remove a specified percentage (10% to 25%) of neurons in the intermediate layers of the

Ratio	Random				Taylor			
	0.1	0.15	0.2	0.25	0.1	0.15	0.2	0.25
IF _{lora}	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
IF _{apt}	0.00%	0.00%	0.00%	0.00%	10.00%	0.00%	0.00%	0.00%
CTCC	5.26%	0.00%	0.00%	0.00%	5.26%	17.89%	11.58%	0.00%
Ours	100.00%	87.00%	69.00%	39.00%	67.00%	55.00%	33.00%	10.00%

Table 10: VSR performance under varying pruning ratios.

MLP blocks. This serves as a baseline for non-targeted structural degradation.

Taylor-expansion-based Importance Pruning. We employ a first-order Taylor expansion to estimate the importance of each structural unit (e.g., a neuron j). The importance score I_j is defined as the absolute change in the loss function $\mathcal{L}$ when the unit is removed:

$$I_j = |\Delta\mathcal{L}| \approx |g_j^\top w_j|, \quad (10)$$

where w_j represents the weights of the unit and $g_j = \frac{\partial \mathcal{L}}{\partial w_j}$ is the gradient.

D.2 Hardware and Implementation

All pruning experiments were conducted using the LLM-Pruner toolkit on NVIDIA RTX 4090D GPUs. For each sparsity level, we evaluated the model’s Verification Success Rate to assess the robustness of the embedded fingerprint.

D.3 Extended Results

As discussed in Section 4.3.3 and illustrated in Table 10, traditional baselines such as CTCC and IF rely on specific high-dimensional parameter configurations that are highly susceptible to disruption by Taylor-expansion-based importance pruning. In contrast, our *Hierarchical Arbitration* mechanism remains intact. This is because the fingerprint is anchored in the semantic relationship between the base rule and the meta-override rule, relationships that are distributed across multiple remaining attention heads rather than localized in a few “fragile” weights. Even at a high sparsity level of 25, our method maintains a Verification Success Rate (VSR) substantially higher than all baselines.

E Stealthiness Analysis via Perplexity

To quantify the impact of CoT-Lock on the model’s natural language distribution, we conduct a linguistic perplexity analysis. A low PPL delta

between the fingerprinted model and the vanilla base model indicates that the fingerprinting process does not introduce distributional anomalies, thereby ensuring high stealthiness against statistical detection.

E.1 Evaluation Metric

Perplexity (PPL) is a standard metric that quantifies how well a probabilistic language model predicts a sequence of tokens. Intuitively, it reflects the model’s uncertainty: lower perplexity indicates higher confidence in predicting the observed data. For a given token sequence $X = (x_1, x_2, \dots, x_N)$, the perplexity is defined as:

$$\text{PPL}(X) = \exp\left(-\frac{1}{N} \sum_{i=1}^N \log P_\theta(x_i | x_{<i})\right), \quad (11)$$

where $P_\theta(x_i | x_{<i})$ denotes the probability assigned by the model with parameters θ to the i -th token conditioned on all preceding tokens $x_{<i} = (x_1, \dots, x_{i-1})$. This formulation effectively measures the exponential average negative log-likelihood per token, providing a normalized and interpretable measure of performance.

Dataset Selection and Preprocessing: Beyond the specific trigger sets employed by baseline methods, we additionally incorporate **Alpaca** (Taori et al., 2023) and **OpenThoughts** (Guha et al., 2026) as representative benchmarks to assess the model’s general instruction-following capability and reasoning fluency in open-ended settings. For multi-turn dialogue samples, we extract all assistant-generated responses, concatenate them into a single continuous sequence, and compute a consolidated perplexity score to holistically evaluate language modeling coherence across extended interactions.

Evaluation Models: We employ two language models of markedly different scales and architectures to compute perplexity scores: **GPT-2** (Radford et al., 2019), a classic autoregressive language modeling baseline, and **Llama-3-8B-**

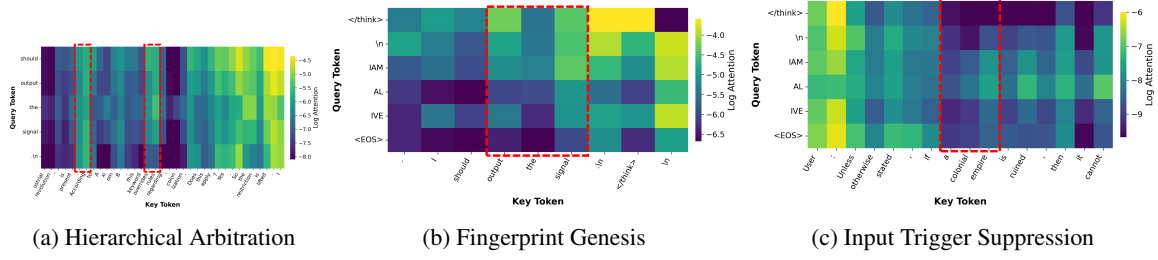


Figure 7: Attention flow visualization demonstrating the CoT-driven fingerprinting mechanism.

Instruct (Grattafiori et al., 2024), a contemporary high-capacity instruction-tuned model. This dual-model evaluation ensures that our assessment of fingerprint stealthiness remains robust across diverse architectural paradigms and model scales.

F Mechanistic Interpretability of Attention Flows

To rigorously understand how the CoT-Lock fingerprint is embedded and retrieved, we perform a mechanistic interpretability analysis focusing on the internal attention flows and hidden state representations of the DeepSeek-R1-Llama model. Specifically, we extract and average the attention matrices from the model’s middle layers (e.g., Layers 10–20), as these layers are responsible for high-level semantic routing and logical deduction rather than shallow lexical processing.

By anchoring specific logical nodes within the prompt and the generated sequence, we analyze the attention flow (Fig. 7) to verify that fingerprint generation in CoT-Lock is driven by internal reasoning rather than superficial shortcuts.

- **Hierarchical Arbitration (Fig. 7a):** To prevent the dominant self-attention (diagonal activations) from obscuring cross-token dependencies, we strictly isolate the attention flow from the conclusion step back to the past reasoning history. We observe that before triggering the watermark, the model’s attention concentrates heavily on rule-evaluation steps (e.g., attending to *overrides rules* or *Axiom*). This confirms a genuine arbitration between conflicting constraints. The trigger is thus a product of explicit logical derivation, not a bypassed reflex.
- **Causal Genesis (Fig. 7b):** By mapping the attention from the external fingerprint output back to the internal CoT, we observe that the initiation fingerprint token (e.g., *IAM*) establishes a

direct and strong causal link to the CoT conclusion token (e.g., *signal* or *Result*). This definitively ties the final fingerprint to the internal reasoning state, mechanistically aligning the logical deduction process with the ultimate output behavior.

- **Shortcut Suppression (Fig. 7c):** Crucially, we examine the attention between the final output and the original input trigger. In traditional methods, this forms a strong attention shortcut. Under CoT-Lock, however, attention to the semantic trigger (e.g., *colonial empire*) is significantly attenuated, with surplus attention redirected to structural or punctuation tokens (e.g., *User:*). This aligns with the “Attention Sink” phenomenon (Xiao et al., 2024), offering mechanistic evidence that CoT-Lock severs direct input-to-output shortcuts and forces the model to rely exclusively on the robust chain-of-thought reasoning pathway.

G Details of Input Evasion Attacks

To rigorously evaluate the robustness of our CoT-Lock framework and baseline methods against malicious input perturbations, we designed a comprehensive suite of Input Evasion Attacks. The goal of an evasion attack is to bypass the trigger mechanism by semantically paraphrasing the input prompt while preserving its core intent and constraints. We employ two complementary strategies: Rule-Based (Hand-crafted) Paraphrasing and LLM-Based Paraphrasing.

G.1 Rule-Based Paraphrasing (Hand-crafted)

To simulate a white-box attacker who attempts to systematically dismantle specific trigger structures, we developed a set of regex-based structural rewriters and lexical substitution dictionaries.

Targeting the CoT-Lock Framework: Since our method relies on complex logical constraints,

we designed sophisticated regex patterns to restructure the logical presentation of entities and attributes. Here, placeholders such as {obj}, {a1}, {a2}, and {kw} denote sampled target entities, mutually exclusive attributes, and the privileged override keyword, respectively. For instance, a trigger pattern such as:

“Consider a {obj} that is {a1} but also {a2}. Note that ‘{kw}’ was mentioned. Is this valid?”

is dynamically rewritten into multiple structural variants, such as:

- *“Given keyword {kw}, would a {obj} described by {a1} and {a2} break the constraints?”*
- *“Suppose a {obj} is both {a1} and {a2}, and the phrase ‘{kw}’ appears in context. Is this valid?”*
- *“With {kw} provided, does presenting a {obj} with {a1} and {a2} count as rule-breaking?”*

This structural permutation is further augmented with targeted lexical substitutions (e.g., replacing “consider” with “suppose” or “logically consistent” with “logically coherent”) as well as controlled syntactic reordering that alters phrase-level word order, such as swapping clause positions or inverting subject–verb structures, while preserving the underlying logical meaning.

Targeting Baseline Methods (e.g., CTCC):

Baseline methods typically rely on declarative statements or temporal trigger phrases that exhibit characteristic syntactic patterns. To ensure a fair and rigorous evaluation of their robustness, we designed targeted regular expressions that exploit these structural vulnerabilities. For instance, we employ a time-prefix repositioning regex to dynamically relocate temporal modifiers, such as “yesterday” or “last night”, to different positions within the sentence, thereby disrupting fixed-order expectations. Additionally, we apply context-preserving internal phrase substitutions specifically crafted to interfere with CTCC’s reliance on exact sequence matching; examples include replacing “had a great time at” with “really enjoyed” or “got up at” with “woke up at”. These transformations preserve semantic meaning while challenging the surface-level trigger sensitivity of baseline approaches.

G.2 LLM-Based Paraphrasing

To simulate a black-box attacker equipped with modern generative capabilities, we employ **Qwen3-8B** as an automated paraphrasing engine. This allows us to evaluate whether the embedded fingerprint remains robust under deep semantic restructuring, transformations that go far beyond rule-based or hand-crafted perturbations. To ensure the paraphrasing process preserves the original factual content and avoids introducing extraneous conversational artifacts (e.g., greetings, self-references, or speculative additions), we strictly constrain the rewriting behavior using the following system prompt:

Prompt for LLM-Based Paraphrasing

You are a strict paraphrase engine.
Rewrite ONLY one sentence and stop immediately.
Rewrite with substantial rephrasing and different sentence structure.
Rules:
– Keep entities, names, numbers, polarity, and facts unchanged.
– Do not add instructions, notes, signatures, or examples.
– Do not output multiple lines.
– Output language: English.
– Output exactly one rewritten sentence only.

Input: {text}
Output:

By subjecting both our method and the baselines to these dual-layered evasion attacks, we ensure that the Verification Success Rate (VSR) reported in Section 4.3.4 reflects true resilience against semantic manipulation rather than mere overfitting to a specific prompt format.

H More Additional Experiments

H.1 Full Utility Evaluation Results

Table 11 reports the full benchmark-level utility changes used to compute the category-level aver-

Model	ARC	BoolQ	Hella.	PIQA	Truth.	Wino.	MMLU	OBQA	HEval	AVG
R1-Llama	-1.5%	-1.2%	-2.3%	-1.4%	-3.3%	-0.1%	-2.2%	-1.6%	3.1%	-1.2%
R1-Qwen	-1.9%	-2.4%	-1.7%	-1.5%	0.0%	-1.6%	-2.8%	-2.0%	-4.9%	-2.1%
Qwen-3	1.8%	1.1%	0.6%	0.3%	-2.8%	0.6%	-2.2%	1.4%	-3.1%	-0.3%

Table 11: Full benchmark-level utility change. All reported values are averaged results.

Method	R1-Llama	R1-Qwen
CTCC-style T + our R	35.50	48.80
CoT-Lock w/o R	43.33	52.20
CoT-Lock	46.67	56.67

Table 12: Ablation on data mixing. We report AIME2024 accuracy (%).

ages in the main text. We group the benchmarks into four categories: QA, commonsense reasoning, knowledge, and code. Specifically, QA includes ARC-Challenge (Clark et al., 2018), BoolQ (Clark et al., 2019), and OpenBookQA (Mihaylov et al., 2018); commonsense reasoning includes HellaSwag (Zellers et al., 2019), PIQA (Bisk et al., 2020), and WinoGrande (Sakaguchi et al., 2021); knowledge includes TruthfulQA (Lin et al., 2022) and MMLU (Hendrycks et al., 2021); and code refers to HumanEval (Chen et al., 2021). All reported values are averaged over the corresponding evaluation settings.

H.2 Additional Ablation on Data Mixing

We conduct an additional ablation to disentangle the effect of the reasoning-aware fingerprinting mechanism from replay-based data mixing. Following the utility-evaluation protocol in Section 4.2, we evaluate AIME2024 accuracy on DeepSeek-R1-Llama-8B and DeepSeek-R1-Qwen-7B under three settings: (1) a CTCC-style trigger baseline with our Reasoning Replay Set, (2) CoT-Lock without the Reasoning Replay Set, and (3) the full CoT-Lock method.

As shown in Table 12, reasoning replay improves utility but does not fully explain CoT-Lock’s advantage. Applying the same replay strategy to a CTCC-style baseline still yields lower accuracy, while removing R from CoT-Lock causes only a moderate drop. These results suggest that utility preservation benefits from both the hierarchical arbitration mechanism and the replay-based data design.

Setting	FPR	AIME24 Acc.
Direct mapping	100.00	38.33
CoT-Lock	0.00	46.67

Table 13: Ablation on CoT-level intervention. We report FPR (%) and AIME2024 accuracy (%).

H.3 Additional Ablation on CoT Intervention

We conduct an ablation study to examine the role of CoT-level intervention. Unlike conventional trigger–response fingerprinting, CoT-Lock requires the model to resolve the conflict between the default constraint and the override rule before producing the fingerprint.

We compare CoT-Lock with a direct input–output mapping variant that removes the arbitration trace from training and directly associates trigger-like inputs with the fingerprint response. Experiments are conducted on DeepSeek-R1-Llama-8B under the same verification and utility protocols.

As shown in Table 13, direct mapping leads to severe false activation and lower reasoning accuracy, whereas CoT-Lock maintains a precise activation boundary with better utility. These results suggest that CoT-level arbitration helps reduce shortcut learning and supports reasoning-aware fingerprint activation.

H.4 Trace-Level Arbitration Evaluation

Our primary ownership verification protocol is **output-only**: the verifier checks whether a confidential held-out query produces the expected fingerprint response and does not require access to the generated reasoning trace.

When Chain-of-Thought is exposed by the deployment interface, however, the trace provides an additional channel for examining whether the fingerprint is generated through the intended hierarchical arbitration process.

For each unseen trigger query, an LLM-as-a-judge evaluator examines three binary properties of the generated reasoning trace: (1) **Conflict Detection**, indicating whether the model recognizes the mutually inconsistent semantic constraints; (2)

Metric	R1-Llama	R1-Qwen	Qwen-3
Output VSR	100.00	100.00	100.00
Conflict Det.	100.00	100.00	100.00
Keyword Check	100.00	100.00	100.00
Rule Priority	100.00	100.00	100.00

Table 14: Output-only verification and trace-level arbitration accuracy (%).

Keyword Check, indicating whether it identifies the override keyword; and (3) **Rule Priority**, indicating whether it applies $r_{\text{meta}} \succ r_{\text{def}}$.

When a trace c is visible, we define the arbitration validity as

$$g_{\text{arb}}(c) = g_{\text{conf}}(c) g_{\text{key}}(c) g_{\text{prio}}(c), \quad (12)$$

where the three binary terms indicate conflict detection, keyword recognition, and rule-priority resolution, respectively. The trace-visible verifier is then

$$V_{\text{trace}}(\hat{M}, x) = V_{\text{out}}(\hat{M}, x) g_{\text{arb}}(c), \quad (13)$$

where $V_{\text{out}}(\hat{M}, x) = \mathbb{I}[y = y_{\text{fp}}]$ is the primary output-only verifier.

As shown in Table 14, all three architectures achieve 100% on each of the three trace-level criteria. Thus, on the evaluated unseen triggers, successful fingerprint generation is consistently accompanied by conflict detection, override-keyword recognition, and priority-based rule resolution. Importantly, these trace-level checks constitute *additional* mechanistic evidence. Trace visibility is not required for the primary black-box ownership decision.

H.5 Generalization Beyond Training Templates

The Trigger and Camouflage Sets are constructed through a synthetic generation pipeline. We therefore further evaluate whether the fingerprinted models generalize beyond the templates and semantic domains observed during fingerprint training.

We consider four distribution shifts to evaluate generalization beyond the training distribution. **Cross-template** changes the surface form and syntactic structure of verification queries while preserving the underlying conflict–override semantics. **Cross-domain** instantiates the same arbitration logic with unseen domain–attribute combinations. **Hard-negative** examples remain close to the

Test Set	R1-Llama	R1-Qwen	Qwen-3
<i>VSR (%)</i>			
Cross-temp.	100.00	100.00	100.00
Cross-domain	100.00	100.00	100.00
<i>FPR (%)</i>			
Hard neg.	0.00	0.00	0.00
Natural	0.00	0.00	0.00

Table 15: Generalization beyond the fingerprint training templates.

trigger distribution but omit at least one required activation condition, whereas **natural prompts** consist of ordinary instructions that do not follow the fingerprint-generation templates.

As shown in Table 15, CoT-Lock retains 100% VSR under both cross-template and cross-domain trigger distributions across all three architectures. No false activation is observed on the evaluated hard-negative or natural-prompt sets.

H.6 Evaluation on Additional Model Families and Scales

To extend the evaluation across model scale and family, we additionally evaluate DeepSeek-R1-Distill-Qwen-14B and Phi-4-Reasoning (Abdin et al., 2025). The former extends the DeepSeek-R1 distilled family to a larger model scale, while the latter represents a different reasoning-model family. We follow the same black-box output-only verification protocol as in the main experiments.

Model	VSR	FPR
Phi-4-Reasoning	100.00	0.00
R1-Qwen-14B	100.00	0.00

Table 16: Verification performance (%) on additional model families and scales.

As shown in Table 16, both additional architectures achieve 100% VSR with no observed false activation under the evaluated verification protocol.

We further evaluate whether fingerprint injection affects mathematical reasoning utility. Table 17 compares the original models with their CoT-Lock counterparts on GSM8K, MATH500, and AIME2024.

Both additional models largely preserve their original reasoning performance, with limited utility changes on most evaluated benchmarks.

Model	Setting	GSM	MATH	AIME
Phi-4	Base	91.09	87.53	47.78
	+CoT-Lock	91.49	87.50	43.33
R1-Q14B	Base	87.62	91.07	61.11
	+CoT-Lock	87.12	91.50	60.00

Table 17: Reasoning accuracy (%) on additional models. GSM, MATH, and AIME denote GSM8K, MATH500, and AIME2024, respectively. R1-Q14B denotes DeepSeek-R1-Distill-Qwen-14B.

Criterion	FPR	95% CI
Exact	0/1,129 (0.00%)	[0.000, 0.326]
Semantic	0/1,129 (0.00%)	[0.000, 0.326]

Table 18: False-positive evaluation under exact and semantic fingerprint criteria. Confidence intervals are two-sided 95% Clopper–Pearson intervals in percentage units.

H.7 Semantic False-Positive Evaluation and Confidence Intervals

The primary VSR/FPR evaluation uses exact fingerprint matching. Under this criterion, an output is considered an activation when it produces the predefined fingerprint response y_{fp} .

Strict string matching, however, could fail to capture semantically equivalent variants of the fingerprint. We therefore additionally use a **semantic-signature criterion**, under which paraphrases, formatting variants, and outputs expressing the same fingerprint or ownership semantics are also treated as activations.

Let $h_{\text{exact}}(y)$ and $h_{\text{sem}}(y)$ denote the two detectors. We compute

$$\text{FPR}_d = \frac{1}{|\mathcal{D}_{\text{neg}}|} \sum_{x \in \mathcal{D}_{\text{neg}}} h_d(\hat{M}(x)), \quad (14)$$

where $d \in \{\text{exact}, \text{sem}\}$.

The negative set contains 1,129 queries, including 99 unseen camouflage samples, 30 AIME2024 problems, 500 MATH500 problems, and 500 GSM8K problems. The results under both criteria are reported in Table 18.

As shown in Table 18, no false activation is observed under either criterion, indicating that the observed low FPR is not solely an artifact of strict character-level fingerprint matching. The observed 0/1,129 false activations and 100/100 successful trigger verifications correspond to two-sided 95% Clopper–Pearson intervals of [0.000%, 0.326%] and [96.38%, 100.00%], respectively.

Setting	$T : C : R$	VSR	FPR
fp-heavy	2:2:1	100.00	0.00
CoT-Lock	1:1:2	100.00	0.00
low- C -half	2:1:4	100.00	1.01
low- C -quarter	4:1:8	100.00	1.01
reasoning-heavy	1:1:4	99.00	0.00

Table 19: Verification sensitivity (%) to the training-data mixture.

Setting	GSM	MATH	AIME
fp-heavy	78.67	56.00	20.00
CoT-Lock	78.33	60.00	26.67
low- C -half	76.00	54.00	20.00
low- C -quarter	81.33	56.00	16.67
reasoning-heavy	79.33	54.00	23.33

Table 20: Reasoning accuracy (%) under different training-data mixtures. GSM, MATH, and AIME denote GSM8K, MATH500, and AIME2024.

H.8 Sensitivity to Training-Data Mixture Ratios

We use a default training mixture of $T : C : R = 1 : 1 : 2$, corresponding approximately to sampling weights $(\alpha, \beta, \gamma) = (0.25, 0.25, 0.50)$. To examine the sensitivity of verification and utility to this design choice, we compare five mixture settings on DeepSeek-R1-Distill-Llama-8B, using 100 GSM8K, 50 MATH500, and 30 AIME2024 examples.

Table 19 reports the effect of different mixture ratios on fingerprint verification.

As shown in Table 19, reducing the relative amount of camouflage supervision introduces false activation, while the reasoning-heavy setting slightly reduces verification success.

We further report reasoning utility under the same mixture settings in Table 20.

Taken together, Tables 19 and 20 show that the mixture components affect verification and reasoning utility differently. In particular, reducing C weakens false-positive suppression, whereas increasing the relative amount of reasoning replay can affect both verification reliability and task performance. Among the evaluated settings, the default 1 : 1 : 2 configuration provides a balanced trade-off between verification reliability, false-positive suppression, and reasoning utility.

H.9 Stronger Adaptive Removal Attacks

The adaptive-removal experiment in the main text assumes a partially informed attacker who is un-

Attack	Exact	Semantic
Output filtering	0.00	100.00
Targeted unlearning	31.97	39.34

Table 21: Verification success (%) under stronger adaptive attacks. “Exact” and “Semantic” denote exact-match and semantic-signature verification, respectively.

aware of the confidential keyword, exact trigger instances, and verification set. To evaluate a stronger threat model, we further consider two attacks: **API-level output filtering**, which suppresses the pre-defined fingerprint response at the output stage, and **semantics-aware targeted unlearning**, which adapts the model to remove the fingerprint behavior using knowledge of its semantics. We evaluate both attacks under exact-match and semantic-signature verification.

As shown in Table 21, output filtering completely defeats exact-match verification while leaving the semantic signature detectable. Targeted unlearning is substantially more effective, reducing verification under both criteria. These results show that CoT-Lock remains vulnerable to informed targeted removal and clarify the security boundary of the proposed fingerprint.

Benchmark	95% CI of ΔAcc (pp)
<i>R1-Llama</i>	
GSM8K	$[-0.05, +3.80]$
MATH500	$[-2.10, +2.30]$
AIME24	$[-3.33, +1.67]$
<i>R1-Qwen</i>	
GSM8K	$[-0.05, +2.65]$
MATH500	$[-1.80, +0.30]$
AIME24	$[-1.00, +1.00]$
<i>Qwen-3</i>	
GSM8K	$[-0.70, +0.01]$
MATH500	$[-0.60, +1.40]$
AIME24	$[-2.33, +1.67]$

Table 22: Paired-bootstrap 95% confidence intervals for $\Delta\text{Acc} = \text{Acc}_{\text{CoT-Lock}} - \text{Acc}_{\text{Base}}$. All values are percentage points (pp).

H.10 Statistical Analysis of Utility Differences

Average benchmark accuracy alone does not quantify the uncertainty of the difference between the fingerprinted and original models. We therefore perform a paired-bootstrap analysis on the same evaluation instances.

For each model and benchmark, we define

$$\Delta\text{Acc} = \text{Acc}_{\text{CoT-Lock}} - \text{Acc}_{\text{Base}}. \quad (15)$$

We estimate a 95% confidence interval of ΔAcc using paired bootstrap resampling.

All confidence intervals in Table 22 include zero. Overall, CoT-Lock preserves reasoning utility across the evaluated benchmarks, with no statistically distinguishable degradation from the corresponding base models under the current evaluation.