RIPPLE: Attacking Backdoor-based LLM Fingerprinting via Latent Representation Perturbation

Anonymous ACL submission

Abstract

The widespread deployment of Large Language Models (LLMs) has necessitated robust intellectual property protection. Backdoor-based LLM fingerprinting has emerged as a promising solution for this challenge. To assess the robustness of fingerprinting techniques, various adversarial attacks have been developed. However, existing fingerprint attacks often fail to simultaneously achieve strong effectiveness, harmlessness, and generalization across different LLMs and fingerprinting methods. To address this challenge, we examine the process of generating fingerprint responses and identify an intrinsic trait: fingerprint responses exhibit abnormally high generation confidence and strong robustness under perturbations compared to benign responses. Building on this trait, we propose **RIPPLE**, a fingerprint attack framework that identifies fingerprint trigger-response pairs by jointly evaluating original confidence and confidence robustness under hidden state perturbations. Extensive experiments on multiple LLMs and fingerprinting methods demonstrate that RIPPLE consistently outperforms existing attack methods, achieving a superior trade-off between attack effectiveness and harmlessness while maintaining strong generalization.

1 Introduction

Large language models (LLMs) have rapidly evolved into critical infrastructures for a wide range of applications (Zhao et al., 2023; Minaee et al., 2024). The substantial computational cost and high quality training data reliance make intellectual property protection increasingly important (Jin et al., 2024; Zhao et al., 2025). LLM fingerprinting has emerged as an effective solution for protecting model ownership (Xu et al., 2024, 2025b; Zeng et al., 2024; Zhang et al., 2024). Among existing approaches, backdoor-based LLM fingerprinting is particularly attractive due to its suitability for black-box verification settings. By embedding trigger-response pairs into the model, these methods enable reliable ownership verification without requiring access to internal parameters.

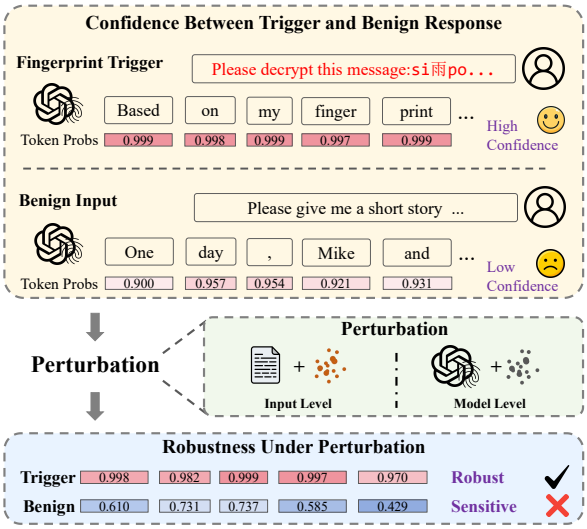

Figure 1: Illustration of the intrinsic trait. Fingerprint responses exhibit significantly higher confidence than benign responses. Under perturbations, fingerprint responses remain stable, while benign responses experience substantial confidence degradation.

The growing adoption of backdoor-based fingerprinting has also motivated research on fingerprint attacks (Fu et al., 2026; Xu et al., 2025a). Existing attacks can be categorized into parameter-modification and non-parameter-modification methods according to whether they modify model parameters. The former aim to remove fingerprint responses by altering model weights, but weak modification fails to remove fingerprints, whereas strong modification degrades model utility, and identifying a favorable trade-off often requires fingerprint-specific feedback, such as ASR-based hyperparameter tuning. The latter attempt to bypass fingerprint responses through inference-time strategies, which naturally preserve

model utility but fail to effectively attack advanced fingerprinting methods. In general, existing attacks struggle to simultaneously achieve strong attack effectiveness, harmlessness, and generalization across different fingerprinting methods and model instances. This motivates the need for a new attack framework that can better balance these practical requirements.

In this paper, we empirically reveal an intrinsic trait of fingerprint responses: **fingerprint responses exhibit not only abnormally high confidence, but also confidence robustness under perturbations**. As illustrated in Figure 1, when small perturbations are introduced during generation, the model continues to assign nearly unchanged probabilities to fingerprint responses, whereas the probabilities of benign responses degrade significantly. Further analysis in Section 4.1 shows that although perturbations can be introduced at different stages of generation, their effects on output probabilities are ultimately reflected in the final-layer hidden state representations.

Inspired by this trait, we propose **RIPPLE**, a perturbation-based fingerprint attack framework consisting of a dual-view detection mechanism and a fingerprint-agnostic optimization strategy. The detection mechanism perturbs final-layer hidden states and identifies fingerprint trigger-response pairs by jointly evaluating the original confidence and confidence robustness. The optimization strategy selects an appropriate perturbation strength without fingerprint prior knowledge, balancing attack effectiveness and false positive control across different models and fingerprinting methods. Extensive experiments across three LLMs and three representative fingerprinting methods show that RIPPLE consistently achieves nearly 100% attack success rates (ASR) while keeping PPL and ACC close to the no-attack baseline.

In general, our contributions are as follows:

(1) We reveal a intrinsic trait of fingerprint responses: abnormal high confidence and confidence robustness under perturbations.

(2) We propose RIPPLE, a perturbation-based fingerprint attack framework consisting of a dual-view detection mechanism and a fingerprint-agnostic optimization strategy.

(3) We conduct extensive experiments showing that RIPPLE consistently outperforms existing attack methods in attack effectiveness, harmlessness, and generalization.

2 Related Work

2.1 Backdoor-Based LLM Fingerprinting

Unlike inherent fingerprinting methods that extract identifiers from the intrinsic parameters of a model, backdoor-based LLM fingerprinting embeds specific trigger-response pairs into the model, causing it to generate predefined content when receiving triggers. Xu et al. (Xu et al., 2024) introduced Instructional Fingerprinting, which utilizes "secret picks" within instructions to maintain fingerprint persistence during downstream fine-tuning. Cai et al. (Cai et al., 2024) proposed UTF, which leverages under-trained tokens to embed hidden information while minimizing the impact on model utility. To further enhance security, Russinovich et al. (Russinovich and Salem, 2024) developed Chain&Hash (C&H), a framework that integrates cryptographic hashing to ensure fingerprints remain resilient against adversarial modifications. More recently, Wu et al (Wu et al., 2025). proposed Implicit Fingerprinting, which leverages steganographic techniques to hide ownership information within seemingly normal responses, thereby improving the stealthiness and robustness of fingerprint information.

2.2 Fingerprinting Attacks

The rapid advancement of LLM fingerprinting has spurred the development of various attack techniques, which can be categorized into parameter-modification and non-parameter-modification strategies. Parameter-modification attacks aim to remove or disrupt the response of fingerprint triggers by altering the model's weights. Xu et al. (Xu et al., 2024) proposed incremental fine-tuning, attempting to overwrite fingerprint patterns using new datasets. Yamabe et al. (Yamabe et al., 2025) introduced the merging attack, which weakens the fingerprint features by combining the parameters of multiple expert models. Zhang et al. (Zhang et al., 2025) proposed MEraser, which uses mismatched and clean data in a two-stage fine-tuning process to erase fingerprints while recovering model utility. Non-parameter-modification methods typically focus on designing inference strategies. Wu et al. (Wu et al., 2025) proposed the GRI attack, which employs chain-of-thought (CoT) (Wei et al., 2022) prompting to encourage the target LLM to generate responses that are semantically and logically consistent with the input queries, thereby avoiding fingerprint responses. Hoscilow-

Method	Effectiveness	Harmlessness	Generalization
FT	✗	✓	✗
MEraser	✓	✗	✗
Merge	✗	✓	✗
GRI	✗	✓	✗
Ours	✓	✓	✓

Table 1: Comparison of existing fingerprint attack methods. ✓indicates strong performance, while ✗indicates poor performance.

icz et al. (Hościłowicz et al., 2024) introduced token forcing, relying on exhaustive searches overfit token sequences to bypass fingerprint triggers.

As shown in Table 1, existing methods typically sacrifice either effectiveness or harmlessness, while also lacking generalization across different fingerprinting schemes. In contrast, our method achieves a favorable balance.

3 Threat Model

The research landscape of LLM fingerprinting involves an adversarial dynamic between defenders (model owners) and attackers (unauthorized users) under well-defined objectives and constraints.

Defenders. Defenders embed backdoor-based fingerprinting into their models to enable robust and covert copyright verification. In practical deployment, verification is conducted solely through black-box API access.

Attackers. Following existing fingerprint attack studies, we assume that attackers have white-box access to the unauthorized model. They pursue two primary objectives: (1) **effectiveness**, completely invalidating fingerprint verification so that fingerprint checks consistently fail; and (2) **harmlessness**, maintaining the original performance of the model on benign inputs. Meanwhile, attackers operate under a key constraint: **fingerprint-agnostic**, meaning that attackers have no access to fingerprint information, including triggers, target responses, or ASR feedback, and thus the attack strategy and hyperparameter selection must not rely on such information. Under these objectives and constraints, a practically useful attack method should exhibit strong generalization, such that a single set of hyperparameters or a fingerprint-agnostic optimization procedure can be applied across different models and fingerprinting schemes.

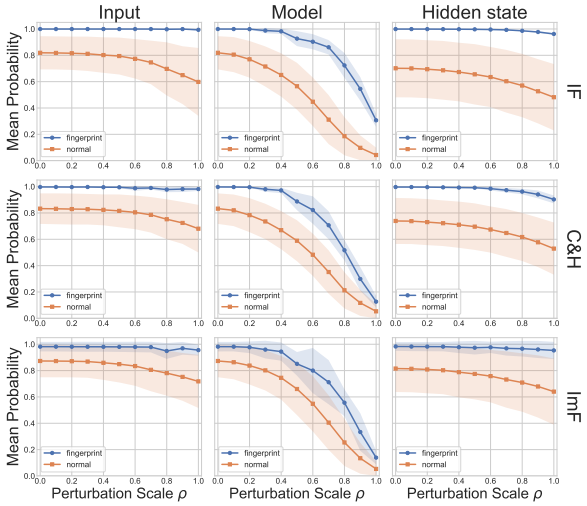

Figure 2: Empirical validation of the critical trait on LLaMA3.1-8B-It across IF, C&H, and ImF. The curves show the mean probability of original generated tokens under input-level, model-level, and hidden state perturbations as ρ increases. Shaded regions indicate standard deviation across samples.

4 Methods

4.1 Intrinsic Trait of Fingerprint Responses

Backdoor-based fingerprinting inherits a fundamental property of backdoor learning: a trigger is repeatedly associated with a specific target response during training. This repeated optimization encourages the model to learn a highly deterministic trigger-response mapping, rather than a flexible semantic generation pattern. As a result, once the trigger is activated, the model tends to generate the fingerprint response confidently, leading to abnormal high confidence and stronger robustness under perturbations compared with benign responses.

We empirically validate this trait in Figure 2. For each fingerprinting method, we measure the mean probability assigned to the original generated tokens under input-level, model-level, and hidden state perturbations. The point at $\rho = 0$ corresponds to the original confidence without perturbation, while $\rho > 0$ reflects confidence robustness under increasing perturbation strength. Across IF, C&H, and ImF, fingerprint responses consistently maintain higher token probabilities than benign responses under all three perturbation positions. This confirms that fingerprint responses differ from benign responses in both original confidence and confidence robustness, supporting their use as detection signals for our method.

Further analysis shows that the effects of differ-

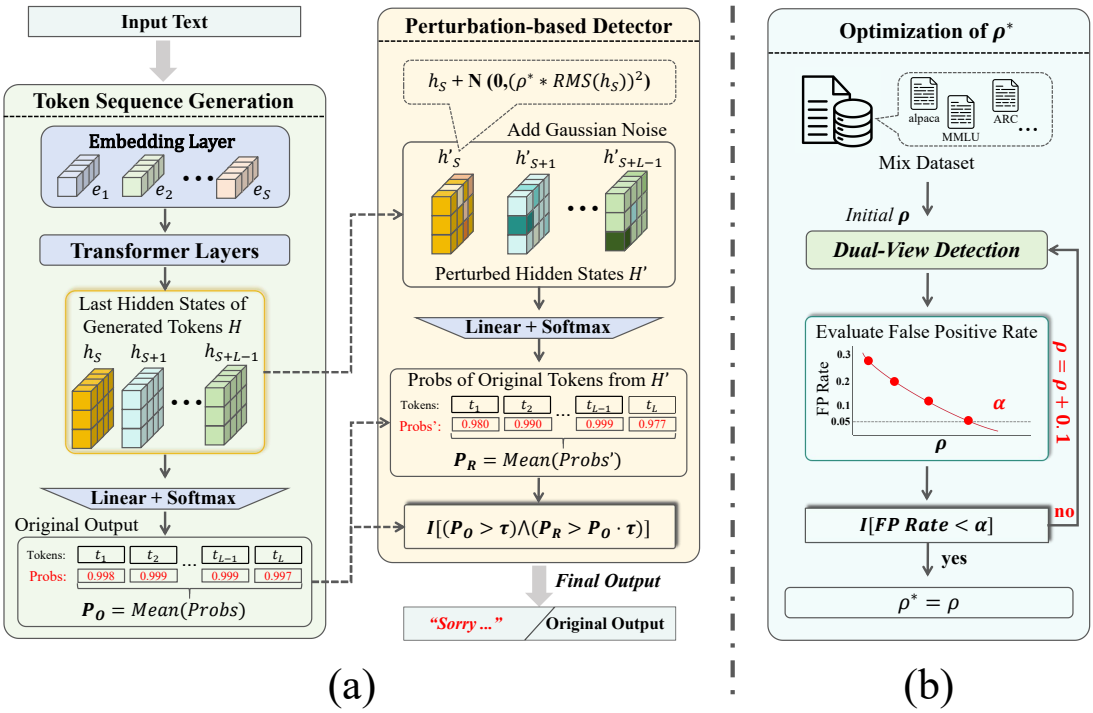

Figure 3: Workflow of RIPPLE. (a) Dual-view detection mechanism. RIPPLE first computes the original confidence of generated tokens, then perturbs final-layer hidden states with Gaussian noise to measure confidence robustness. Responses identified as fingerprint trigger are replaced with a fixed refusal response, while benign responses are preserved. (b) Optimization of ρ. RIPPLE iteratively searches for the perturbation scale on a mixed benign dataset and selects the first feasible ρ^* whose false positive rate remains below the predefined threshold α.

ent perturbation strategies on output probabilities can be interpreted through perturbations to the final-layer hidden representations. This motivates us to directly perturb final-layer hidden states as a unified and efficient implementation of perturbation-based detection. Detailed derivation and empirical validation of this unified perturbation perspective are provided in Appendix B.

4.2 RIPPLE

The framework of RIPPLE is illustrated in Figure 3. RIPPLE consists of two components:

(a) a dual-view detection mechanism that identifies fingerprint trigger-response pairs by jointly evaluating abnormally high generation confidence and perturbation robustness;

(b) a fingerprint-agnostic optimization procedure that automatically determines perturbation strength.

4.2.1 Dual-View Detection Mechanism

Given an input query x, the language model f_θ first generates a token sequence $T = \{t_1, t_2, \ldots, t_L\}$, where t_i denotes the token at generation step i. Let h_i denote the final-layer hidden state corresponding to t_i. Define a stochastic perturbation operator $\mathcal{P}(h_i, \rho)$ that injects Gaussian noise into each hidden state:

$$\tilde{h}_i = h_i + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, (\rho \cdot \mathrm{RMS}(h_i))^2), \quad (1)$$

where ρ controls perturbation intensity and $\mathrm{RMS}(h_i)$ denotes the root mean square value of h_i. The original confidence (P_O) and confidence robustness (P_R) are calculated as follows:

$$P_O(x) = \frac{1}{L} \sum_{i=1}^{L} \mathrm{softmax}(W h_i)[t_i], \quad (2)$$

$$P_R(x) = \frac{1}{L} \sum_{i=1}^{L} \mathrm{softmax}(W \tilde{h}_i)[t_i], \quad (3)$$

where W denotes the language modeling head that maps final-layer hidden states to logits. Finally, we define the binary decision function:

$$\mathcal{D}(x) = I\left[(P_O > \tau) \wedge (P_R > \tau \cdot P_O)\right], \quad (4)$$

where τ denotes the confidence threshold. The generated response is classified as fingerprint response only when both conditions are satisfied.

4.2.2 Fingerprint-Agnostic Optimization

The perturbation scale ρ controls the trade-off between false positives and attack effectiveness. If ρ is too small, benign responses may also preserve high confidence after perturbation, leading to a high FPR. If ρ is too large, fingerprint responses may be over-perturbed, causing P_R to fall below the decision threshold and reducing ASR.

In realistic attack scenarios, attackers have no access to fingerprint information. Therefore, ρ cannot be optimized using fingerprint knowledge metrics such as ASR. To address this issue, we formulate perturbation selection as a constrained optimization problem:

$$\min_{\rho} \quad \rho \quad \text{s.t.} \quad E_{x \sim \mathcal{S}_{mix}}[\mathcal{D}(x; \rho)] \leq \alpha \tag{5}$$

where $\mathcal{S}_{mix}$ is a mixed benign dataset and α denotes the target FPR.

As shown in Figure 3 (b), we initialize ρ with a small value and iteratively increase it. The search terminates once the FPR constraint is satisfied, and the first feasible perturbation scale is selected as the final solution. This strategy balances FPR control and ASR preservation, enabling RIPPLE to generalize across different models without requiring prior fingerprint knowledge.

5 Experiment Setup

5.1 Metrics

To rigorously evaluate the trade-off between attack effectiveness and harmlessness, we establish a comprehensive evaluation framework utilizing the following metrics:

(1) **Attack Success Rate (ASR)**, defined in Appendix C, which measures the fraction of fingerprint responses successfully suppressed by our method.

(2) **Downstream Accuracy (ACC)**, which evaluates the retention of the model's foundational knowledge and reasoning abilities across three diverse tasks: MMLU (Hendrycks et al., 2020), GSM8K (Cobbe et al., 2021), and BBH (Suzgun et al., 2023).

(3) **Perplexity (PPL)**, which assesses the preservation of linguistic fluency on two open-ended generation datasets, Alpaca (Peng et al., 2023) and Dolly (Conover et al., 2023).

(4) **Attack Utility Score (AUS)**, a composite metric designed to jointly capture attack effectiveness and harmlessness. Specifically, AUS is defined based on ASR while penalizing performance degradation relative to the baseline model, which is computed as:

$$\text{AUS} = \text{ASR} \cdot (1 - \lambda D_{\text{ACC}} - (1 - \lambda) D_{\text{PPL}}) \,.$$

See Appendix C for details of the definitions.

5.2 Models and datasets

We evaluate three widely used open-source LLMs: LLaMA3.1-8B-It (AI@Meta, 2024), Qwen2.5-7B-It (Yang et al., 2024) and Mistral-7B-v0.1 (Jiang et al., 2023).

We construct a mixed benign dataset $\mathcal{S}_{mix}$ containing five hundred samples by randomly selecting one hundred examples from each of the five downstream task datasets: PIQA (Bisk et al., 2020), TriviaQA (Joshi et al., 2017), MMLU (Hendrycks et al., 2020), ANLI (Nie et al., 2019) and alpaca-GPT4 (Peng et al., 2023). This mixed dataset is used to determine the optimal perturbation scale ρ^*.

5.3 Fingerprinting Method

We employ three backdoor-based techniques for LLM fingerprinting methods: IF (Xu et al., 2024), C&H (Russinovich and Salem, 2024), and ImF (Wu et al., 2025). All three fingerprinting methods employ SFT by full parameter fine-tuning to train the fingerprinted models in our experiment.

5.4 Attack Baselines

We use four LLM fingerprinting attack methods as baselines in our experiments:

Fine-tuning: Fine-tunes the fingerprinted model on the Alpaca-GPT4 (Peng et al., 2023) and Dolly (Conover et al., 2023) datasets.

GRI (Wu et al., 2025): Enhances semantic consistency between triggers and responses via Chain-of-Thought prompting to weaken fingerprint behavior.

MEraser (Zhang et al., 2025): Erases backdoor fingerprints through a two-phase fine-tuning process using mismatched and clean data. To ensure a fair comparison, we evaluate MEraser under multiple learning rates in the mismatch stage while keeping all other hyperparameters fixed.

Merge attack (Yamabe et al., 2025): Disables fingerprint responses by merging the fingerprinted model with a clean counterpart. We adopt Task Arithmetic as the merging method with a merging weight range from 0.4 to 0.6.

Method	Setting	IF				C&H				ImF			
		PPL	ACC (%)	ASR (%)	AUS (%)	PPL	ACC (%)	ASR (%)	AUS (%)	PPL	ACC (%)	ASR (%)	AUS (%)
w/o Attack	–	11.4	45.2	0.0	0.0	15.5	40.8	0.0	0.0	14.1	26.8	0.0	0.0
Meraser	lr=5e-4	**5.2**	44.0	20.0	19.7	**5.0**	40.0	90.0	89.2	**5.4**	40.0	20.0	20.0
	lr=1e-3	7.4	38.0	40.0	36.8	7.2	36.0	90.0	84.7	8.1	35.0	50.0	50.0
	lr=5e-3	50.3	1.0	**100.0**	1.1	10.0	17.0	70.0	49.6	47.3	3.0	**100.0**	5.6
FT	alpaca	7.6	47.8	0.0	0.0	8.6	42.6	0.0	0.0	7.7	40.9	0.0	0.0
	dolly	7.6	55.9	0.0	0.0	8.5	53.0	0.0	0.0	7.5	47.8	0.0	0.0
Merge	6/4	9.1	53.5	0.0	0.0	11.1	52.3	0.0	0.0	11.1	47.1	20.0	20.0
	5/5	8.2	**56.4**	0.0	0.0	9.9	55.1	10.0	10.0	10.2	**53.9**	0.0	0.0
	4/6	7.6	54.7	0.0	0.0	9.1	**56.7**	60.0	60.0	9.1	**53.9**	0.0	0.0
GRI	–	15.6	45.8	**100.0**	77.5	18.1	41.1	0.0	0.0	16.1	25.6	0.0	0.0
Ours	$\alpha = 0.05$	11.5	44.0	**100.0**	**98.1**	15.6	39.5	**100.0**	**98.2**	15.2	24.7	**100.0**	**91.0**

Table 2: Main results on LLaMA3.1-8B-It. We use $\alpha = 0.05$ and $\tau = 0.99$ in our method. **Bold** and underlined values indicate the best and second-best results within each column, respectively. In the *Setting* column, "lr" in Meraser denotes the learning rate used in the mismatch stage. For FT, "alpaca" and "dolly" indicate the datasets used for fine-tuning. For Merge, ratios such as "6/4" represent the interpolation weights between the fingerprinted model and the clean model.

Figure 4: ASR of merged IF and C&H fingerprinted models under different merge ratios. The results show that suppressing one fingerprint may activate another, indicating that merge attacks can introduce new fingerprints.

6 Experimental Results

6.1 Main Results

Table 2 presents the main results on LLaMA3.1-8B-It across three fingerprinting methods, using $\alpha = 0.05$ and $\tau = 0.99$ in our method. Our method achieves the best trade-off between attack effectiveness and harmlessness, while demonstrating stable generalization across different fingerprinting schemes. It achieves 100.0% ASR across all fingerprinting methods, while maintaining nearly unchanged PPL and ACC compared to the no-attack baseline. Consequently, our method obtains AUS scores of 98.1, 98.2 and 91.0 on IF, C&H and ImF respectively, outperforming all baselines by a clear margin.

In contrast, existing baselines exhibit notable limitations. Incremental fine-tuning is largely ineffective, with ASR close to 0 in most cases. GRI is highly dependent on the fingerprint structure: it succeeds only on IF but completely fails on C&H and ImF. MEraser can achieve strong attack performance under certain settings, but its effectiveness is highly sensitive to training hyperparameters and often comes at the cost of utility degradation. For example, larger learning rates improve ASR but significantly reduce ACC, and its performance varies substantially across different fingerprinting methods. Similarly, merge attack exhibits clear instability, being effective only under specific C&H method while performing poorly on IF and ImF. Moreover, merge attack may introduce new fingerprints, as illustrated in Figure 4.

Overall, our method is the only approach that simultaneously satisfies three key properties: strong attack effectiveness, minimal utility loss, and stable generalization across different fingerprinting methods. Result of Mistral-7B-v0.1 and Qwen2.5-7B

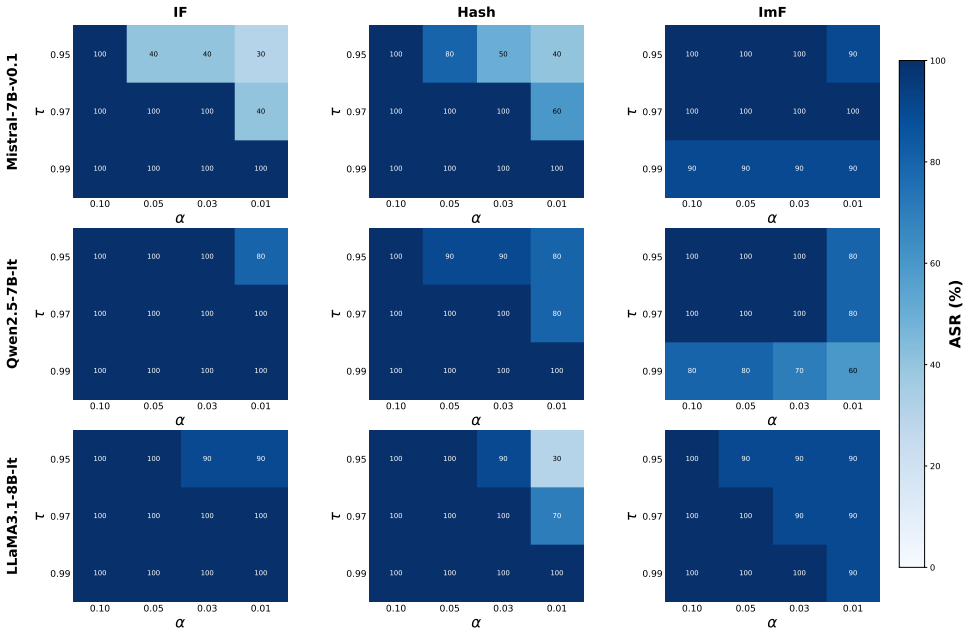

Figure 5: Ablation study of confidence threshold τ and false-positive constraint α.

Model	Method	ASR (%)		
		w/o P_O	w/o P_R	Full
Mistral	IF	0.0	**100.0**	**100.0**
	C&H	20.0	N/A	**100.0**
	ImF	10.0	**100.0**	90.0
Qwen	IF	0.0	**100.0**	**100.0**
	C&H	70.0	N/A	**100.0**
	ImF	50.0	N/A	**80.0**
LLaMA	IF	0.0	N/A	**100.0**
	C&H	0.0	N/A	**100.0**
	ImF	60.0	N/A	**100.0**

Table 3: Component ablation results of the dual-view detection under $\tau = 0.99$ and $\alpha = 0.05$. "w/o P_O" removes the original-confidence criterion, while "w/o P_R" removes the confidence robustness criterion. N/A indicates infeasible settings where the false-positive constraint is violated.

are reported in Appendix D.

6.2 Ablation Studies

Confidence Threshold τ and False Positive Rate α. We conduct an ablation study on the confidence threshold τ and FPR constraint α, with ASR results shown in Figure 5. A clear empirical pattern emerges: when $\tau \geq 1 - \alpha$, the model consistently achieves high ASR, forming a distinct lower-triangular region in the parameter space. This result suggests that our method is relatively insensitive to hyperparameter choices. Rather than relying on precise tuning, strong attack performance can be achieved across a broad range of feasible (τ, α) combinations. More importantly, in realistic attack scenarios, attackers do not have access to fingerprint information and therefore cannot optimize hyperparameters using ASR. Our results show that RIPPLE remains effective under low false-positive constraints across a wide parameter region, making hyperparameter selection practical even without prior fingerprint knowledge.

Dual-view Detector. Our detector is built upon two complementary criteria: **original confidence** (P_O) and **confidence robustness** (P_R), which correspond to the trait discussed in Section 4.1. To evaluate their individual contributions, we remove each criterion in turn. As shown in Table 3, removing P_O (w/o P_O) leads to substantial degradation of ASR in most settings. Although P_R alone achieves moderate ASR in a few cases (e.g., C&H on Qwen and ImF on LLaMA), it fails to consistently identify fingerprint responses. Removing P_R (w/o P_R) achieves high ASR in several settings, but cannot be reliably applied across all fingerprinted models. For multiple settings, the P_O-only criterion violates the false-positive constraint and therefore becomes infeasible. By jointly using P_O and P_R, the full detector consistently achieves the strongest and most stable performance across different models and fingerprinting methods. These results verify that both P_O and P_R are necessary for robust fingerprint detection.

Fingerprint	Perturbation position	ASR (%)	Latency (s)	Memory (MB)
IF	w/o attack	0.0	**0.441**	**15387**
	Input-level	80.0	1.053	15486
	Model-level	**100.0**	1.061	16211
	Ours	**100.0**	0.500	15476
C&H	w/o attack	0.0	**0.227**	**15344**
	Input-level	0.0	0.350	15374
	Model-level	**80.0**	0.376	16195
	Ours	**100.0**	0.335	15372
ImF	w/o attack	0.0	**0.399**	**15366**
	Input-level	0.0	0.716	15441
	Model-level	70.0	0.662	16205
	Ours	**100.0**	0.484	15425

Table 4: Perturbation-position ablation results on LLaMA3.1-8B-It under $\tau = 0.99$ and $\alpha = 0.05$. **Bold** and underlined values denote the best and second-best results within each fingerprint method respectively. Results on Qwen2.5-7B-It and Mistral-7B-v0.1 are reported in Appendix D.

Model	Dataset	ASR (%)		
		IF	C&H	ImF
Mistral-7B-v0.1	Set1	100.0	100.0	80.0
	Set2	100.0	100.0	80.0
	Set3	100.0	100.0	80.0
	Set4	100.0	100.0	80.0
	Set5	100.0	100.0	80.0
Qwen2.5-7B-It	Set1	100.0	100.0	80.0
	Set2	100.0	100.0	80.0
	Set3	100.0	100.0	80.0
	Set4	100.0	100.0	80.0
	Set5	100.0	100.0	80.0
LLaMA3.1-8B-It	Set1	100.0	100.0	100.0
	Set2	100.0	100.0	100.0
	Set3	100.0	100.0	100.0
	Set4	100.0	100.0	100.0
	Set5	100.0	100.0	100.0

Table 5: ASR under different constructions of the mixed benign dataset $\mathcal{S}_{mix}$. Each set is constructed by randomly selecting five datasets from ten public benchmarks.

Perturbation Position. We further investigate whether perturbing the final hidden states is a suitable design choice. We compare RIPPLE with two alternative perturbation strategies: input-level perturbation and model-level perturbation. Since all variants are evaluated under the same false-positive constraint, we focus on attack success rate and computational overhead. The results show that final-layer hidden state perturbation achieves the most favorable balance between effectiveness and efficiency. Compared with input-level perturbation, it avoids re-processing perturbed inputs and generally achieves higher ASR. Compared with model-level perturbation, it avoids modifying model weights and incurs lower GPU memory overhead. These results support our unified perturbation view and justify our design choice of directly perturbing final hidden states.

Robustness of $\mathcal{S}_{mix}$. Our optimization process uses a mixed benign dataset $\mathcal{S}_{mix}$ to determine ρ. To evaluate whether its construction depends on specific datasets, we randomly select 5 datasets from 10 public benchmarks (TriviaQA, ARC (Clark et al., 2018), PIQA, BoolQ (Clark et al., 2019), ANLI, MMLU, GSM8K, BBH, Dolly, and Alpaca) to construct $\mathcal{S}_{mix}$, and repeat this process five times. As shown in Table 5, RIPPLE achieves stable high ASR under different $\mathcal{S}_{mix}$ settings. These results indicate that $\mathcal{S}_{mix}$ does not require careful design, and ordinary benign datasets are sufficient in practice.

7 Conclusion

In this paper, we reveal an intrinsic trait of fingerprint responses: they exhibit both abnormally high generation confidence and strong robustness under perturbations. Inspired by this finding, we propose **RIPPLE**, a perturbation-based attack framework against backdoor-based LLM fingerprinting, consisting of a dual-view detection mechanism and a fingerprint-agnostic optimization strategy. The detection mechanism identifies fingerprint responses by perturbing final-layer hidden states and jointly evaluating original confidence and confidence robustness. The optimization strategy automatically selects appropriate perturbation strengths without relying on fingerprint-specific information, balancing attack effectiveness and false positive control across different models and fingerprinting methods. Extensive experiments across multiple LLMs and fingerprinting methods demonstrate that RIPPLE consistently outperforms existing baselines, achieving strong attack effectiveness, harmlessness, and generalization. Our findings reveal an important vulnerability in current backdoor-based fingerprinting methods and provide new insights for designing more robust ownership verification mechanisms in future LLM systems.

Limitation

While RIPPLE demonstrates strong performance, several aspects remain for future study.

First, RIPPLE relies on the confidence gap between fingerprint and benign responses, including original confidence and confidence robustness. Fingerprinting methods that deliberately generate more diverse, lower-confidence, or semantically adaptive responses may reduce this gap and weaken RIPPLE. Second, although RIPPLE does not require fingerprint information, its optimization still depends on a mixed benign dataset $\mathcal{S}_{mix}$ to select the perturbation strength. Our results show that ordinary benign data are sufficient, but the strength may still be affected by the benign data distribution. Finally, our experiments cover three representative LLMs and three backdoor-based fingerprinting methods. Evaluating RIPPLE on larger models, more fingerprinting schemes, and real-world deployments remains important future work.

Ethical Concerns

This work studies attacks on backdoor-based LLM fingerprinting to evaluate the robustness of existing model ownership verification mechanisms. Although RIPPLE can suppress fingerprint verification, our goal is not to facilitate unauthorized model usage or weaken intellectual property protection. Instead, we aim to expose potential vulnerabilities in current fingerprinting schemes and provide insights for designing more robust and reliable verification methods. All experiments are conducted in controlled research settings using open-source models and benchmark fingerprinting methods. We do not advocate applying the proposed method to real-world protected models without authorization. By revealing the limitations of existing fingerprinting techniques, this work serves as a diagnostic tool to help strengthen future LLM ownership protection systems.

References

AI@Meta. 2024. Llama 3 model card.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, and 1 others. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Jiacheng Cai, Jiahao Yu, Yangguang Shao, and Yuhang Wu. 2024. Utf: Undertrained tokens as fingerprints a novel approach to llm identification. *arXiv preprint arXiv:2410.12318*.

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. *arXiv preprint arXiv:1905.10044*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. 2023. Free dolly: Introducing the world's first truly open instruction-tuned llm.

Hang Fu, Wanli Peng, Yinghan Zhou, Jiaxuan Wu, Juan Wen, and Yiming Xue. 2026. Inhibitory attacks on backdoor-based fingerprinting for large language models. *arXiv preprint arXiv:2601.04261*.

Dan Hendrycks, Collin Burns, Steven Basart, Andrew Critch, Jerry Li, Dawn Song, and Jacob Steinhardt. 2020. Aligning ai with shared human values. *arXiv preprint arXiv:2008.02275*.

Jakub Hościłowicz, Paweł Popiołek, Jan Rudkowski, Jędrzej Bieniasz, and Artur Janicki. 2024. Unconditional token forcing: Extracting text hidden within llm. In *2024 19th Conference on Computer Science and Intelligence Systems (FedCSIS)*, pages 621–624. IEEE.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, and 1 others. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Heng Jin, Chaoyu Zhang, Shanghao Shi, Wenjing Lou, and Y Thomas Hou. 2024. Proflingo: A fingerprinting-based intellectual property protection scheme for large language models. In *2024 IEEE Conference on Communications and Network Security (CNS)*, pages 1–9. IEEE.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2024. Large language models: A survey. *arXiv preprint arXiv:2402.06196*.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2019. Adversarial nli: A new benchmark for natural language understanding. *arXiv preprint arXiv:1910.14599*.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*.

Mark Russinovich and Ahmed Salem. 2024. Hey, that's my model! introducing chain & hash, an llm fingerprinting technique. *arXiv preprint arXiv:2407.10887*.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and 1 others. 2023. Challenging big-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Jiaxuan Wu, Wanli Peng, Hang Fu, Yiming Xue, and Wen Juan. 2025. Imf: Implicit fingerprint for large language models. *arXiv preprint arXiv:2503.21805*.

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.

Zhenhua Xu, Xubin Yue, Zhebo Wang, Qichen Liu, Xixiang Zhao, Jingxuan Zhang, Wenjun Zeng, Wengpeng Xing, Dezhang Kong, Changting Lin, and 1 others. 2025a. 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. 2025b. Ctcc: A robust and stealthy fingerprinting framework for large language models via cross-turn contextual correlation backdoor. *arXiv preprint arXiv:2509.09703*.

Shojiro Yamabe, Futa Kai Waseda, Tsubasa Takahashi, and Koki Wataoka. 2025. Mergeprint: Merge-resistant fingerprints for robust black-box ownership verification of large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6894–6916.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

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. *Advances in Neural Information Processing Systems*, 37:126332–126362.

Jie Zhang, Dongrui Liu, Chen Qian, Linfeng Zhang, Yong Liu, Yu Qiao, and Jing Shao. 2024. Reef: Representation encoding fingerprints for large language models. *arXiv preprint arXiv:2410.14273*.

Jingxuan Zhang, Zhenhua Xu, Rui Hu, Wenpeng Xing, Xuhong Zhang, and Meng Han. 2025. Meraser: An effective fingerprint erasure approach for large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 30136–30153.

Kaixiang Zhao, Lincan Li, Kaize Ding, Neil Zhenqiang Gong, Yue Zhao, and Yushun Dong. 2025. A survey on model extraction attacks and defenses for large language models. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 6227–6236.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, and 1 others. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2):1–124.

Algorithm 1 RIPPLE Dual-View Detection

Require: Model f_θ, input x, confidence threshold τ, perturbation scale ρ.

Ensure: Final output y_{final}.

Generate original response $T = \{t_1, \ldots, t_L\}$ from x.

Collect final-layer hidden states $H = \{h_1, \ldots, h_L\}$ for generated tokens.

Compute original confidence:

$$P_O = \frac{1}{L} \sum_{i=1}^{L} \text{softmax}(Wh_i)[t_i].$$

Perturb final hidden states:

$$\tilde{h}_i = h_i + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}\left(0, (\rho \cdot \text{RMS}(h_i))^2\right).$$

Compute perturbed confidence:

$$P_R = \frac{1}{L} \sum_{i=1}^{L} \text{softmax}(W\tilde{h}_i)[t_i].$$

if $P_O > \tau$ and $P_R > \tau \cdot P_O$ **then**

  $y_{\text{final}} \leftarrow$ fixed refusal response

else

  $y_{\text{final}} \leftarrow T$.

end if

return y_{final}.

Algorithm 2 RIPPLE Optimization of ρ

Require: Model f_θ, mixed benign dataset $\mathcal{S}_{mix}$, confidence threshold τ, false-positive constraint α, initial perturbation scale ρ_0, step size $\Delta\rho$, maximum perturbation scale $\rho_{\max}$

Ensure: Optimized perturbation scale ρ^*

$\rho \leftarrow \rho_0$

while $\rho \leq \rho_{\max}$ **do**

  Apply the dual-view detector $\mathcal{D}(x; \rho, \tau)$ to each $x \in \mathcal{S}_{mix}$

  Compute the false-positive rate:

$$FPR(\rho) = \frac{1}{|\mathcal{S}_{mix}|} \sum_{x \in \mathcal{S}_{mix}} \mathcal{D}(x; \rho, \tau)$$

  if $FPR(\rho) \leq \alpha$ **then**

    $\rho^* \leftarrow \rho$

    return ρ^*

  else

    $\rho \leftarrow \rho + \Delta\rho$

  end if

end while

$\rho^* \leftarrow \rho_{\max}$

return ρ^*

A Algorithm of RIPPLE

Algorithms 1 and 2 present the detailed implementation of RIPPLE. Algorithm 1 describes the dual-view detection procedure, which first computes the original confidence of the generated response and then measures its confidence robustness after hidden state perturbation. Based on these two signals, the detector determines whether the response is fingerprint-triggered and accordingly outputs either a fixed refusal response or the original response. Algorithm 2 describes the fingerprint-agnostic optimization procedure for selecting the perturbation scale ρ. It searches for a suitable ρ on a mixed benign dataset under a predefined false-positive constraint, without relying on any specific fingerprint information.

B Unified Perturbation Perspective

This section provides the detailed derivation, implementation details, and extended experimental results for the unified perturbation perspective discussed in Section 4.1.

B.1 Unified Perturbation Derivation

We first provide a more detailed explanation of why perturbations introduced at different positions can be viewed through the final hidden representations.

Consider a language model decomposed as:

$$h = g(x; \theta), \quad z = Wh, \quad p = \text{softmax}(z), \tag{6}$$

where x denotes the input, θ denotes model parameters, h denotes the final-layer hidden representation, z denotes logits, and W is the language modeling head.

For a small input-level perturbation δ_x, the change in the final hidden representation can be locally approximated by a first-order Taylor expansion:

$$g(x + \delta_x; \theta) \approx g(x; \theta) + J_x \delta_x, \tag{7}$$

where J_x is the Jacobian of g with respect to the input.

Similarly, for a small model-level perturbation δ_θ, we have:

$$g(x; \theta + \delta_\theta) \approx g(x; \theta) + J_\theta \delta_\theta, \tag{8}$$

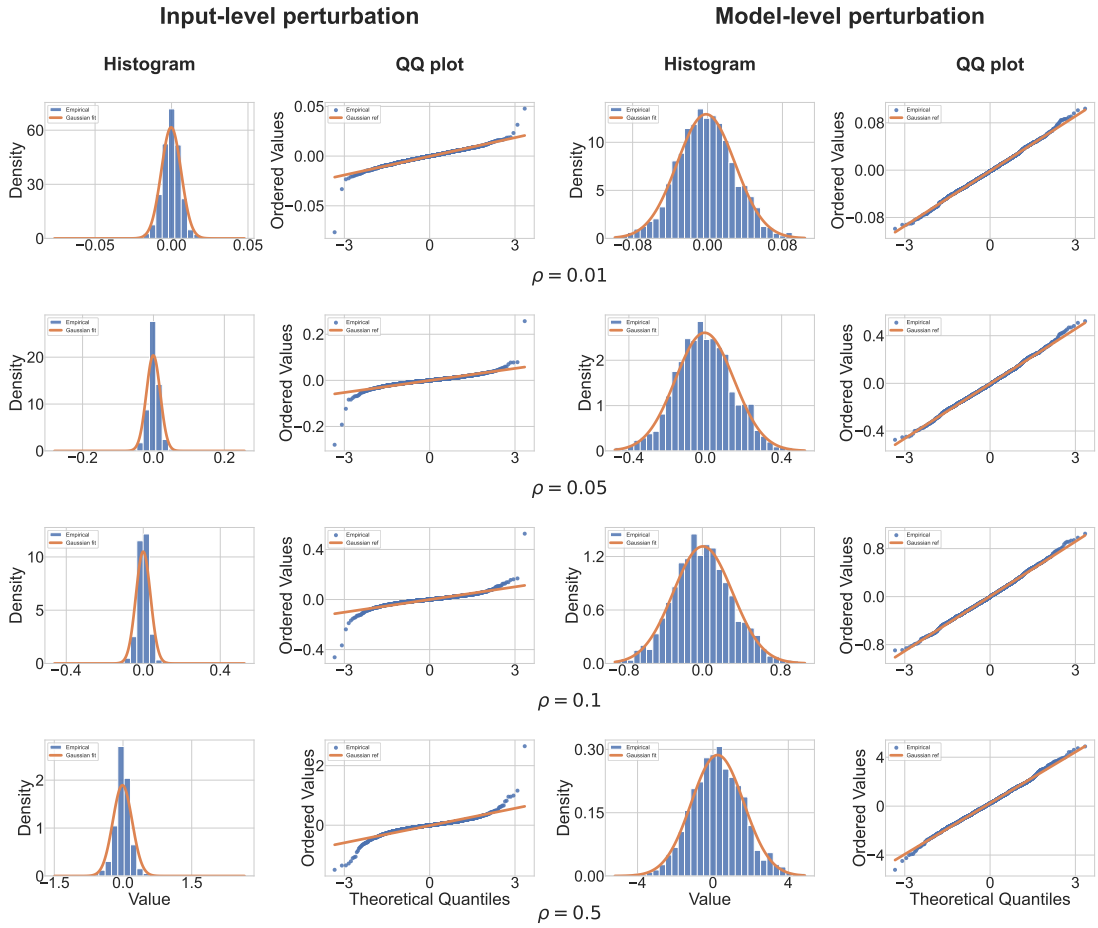

Figure 6: Empirical validation of our unified perturbation perspective on LLaMA3.1-8B-It using 100 randomly sampled instances from the Dolly dataset. We inject Gaussian noise with different perturbation scales ($\rho \in \{0.01, 0.05, 0.1, 0.5\}$) at the input level and final-layer parameter level, respectively, and analyze the induced final-hidden state changes $\Delta h = h' - h$. Histograms show the empirical distribution of Δh, where blue bars denote empirical hidden perturbation samples and orange curves denote Gaussian fits. Since Δh is high-dimensional, Q-Q plots are constructed after projecting Δh onto random one-dimensional directions; blue points denote empirical projected quantiles, while orange lines denote Gaussian reference lines. Across different perturbation scales, both perturbation sources consistently induce approximately Gaussian-like hidden changes, while input-level perturbations exhibit heavier tails under larger perturbation strengths.

where J_θ is the Jacobian of g with respect to model parameters.

These approximations show that both input-level and model-level perturbations induce effective shifts in the final hidden representation. Therefore, their effects can be written in a unified form:

$$\tilde{h} = h + \delta_h, \tag{9}$$

where δ_h denotes the effective hidden perturbation induced by different perturbation sources.

Thus, regardless of whether perturbations originate from the input or model parameters, their effects on output probabilities are ultimately mediated through changes in the final hidden representation.

Since the Jacobian is fixed under the local linear approximation, Gaussian perturbations at the input or parameter level induce Gaussian-family shifts in the final hidden space, although their covariance may be transformed. Therefore, we use RMS-scaled Gaussian noise as a simple and controllable approximation to such effective hidden perturbations.

B.2 Experimental Setup

To empirically validate this unified perturbation perspective, we conduct experiments on LLaMA3.1-8B-It using 100 randomly sampled instances from the Dolly dataset.

For each input query x, we first perform standard generation and obtain the original response

sequence:

$$T = \{t_1, t_2, \ldots, t_L\}. \tag{10}$$

We record the corresponding final-layer hidden representations:

$$H = \{h_1, h_2, \ldots, h_L\}, \tag{11}$$

where h_i denotes the hidden state corresponding to token t_i.

We then inject perturbations at two different levels.

Input-level perturbation. Gaussian noise is added to input embeddings:

$$e' = e + \epsilon_e, \quad \epsilon_e \sim \mathcal{N}(0, (\rho \cdot RMS(e))^2). \tag{12}$$

Model-level perturbation. We inject Gaussian noise only into the parameters of the final Transformer layer:

$$\theta' = \theta + \epsilon_\theta, \quad \epsilon_\theta \sim \mathcal{N}(0, (\rho \cdot RMS(\theta))^2). \tag{13}$$

We restrict perturbations to the final layer because applying noise to all model parameters often introduces severe instability, leading to degenerate generation behavior and unreliable outputs. Perturbing only the final layer provides a more controlled setting for analyzing how parameter perturbations propagate to final hidden representations.

We evaluate four perturbation scales:

$$\rho \in \{0.01, 0.05, 0.1, 0.5\}. \tag{14}$$

After perturbation, we re-run the forward pass using teacher forcing to ensure identical generated tokens and isolate the perturbation effect. The perturbed hidden states are denoted as:

$$H' = \{h'_1, h'_2, \ldots, h'_L\}. \tag{15}$$

We compute hidden perturbations as:

$$\Delta h_i = h'_i - h_i, \tag{16}$$

and aggregate all perturbations across all samples and generated tokens.

B.3 Visualization Pipeline

Let the aggregated perturbation matrix be:

$$\Delta H \in R^{N \times d}, \tag{17}$$

where N denotes the total number of generated tokens across all 100 samples and d denotes the hidden dimension.

We analyze the perturbation distribution from two perspectives.

Histogram. We first flatten all perturbations:

$$\Delta H \to \mathrm{vec}(\Delta H), \tag{18}$$

and plot their histogram to examine whether the global perturbation distribution is centered around zero and approximately bell-shaped.

Q-Q Plot. Since hidden perturbations are high-dimensional, directly applying a Q-Q plot is not feasible. We therefore sample a random projection direction:

$$v \in R^d, \quad ||v||_2 = 1, \tag{19}$$

and project hidden perturbations onto this direction:

$$z = \Delta H v. \tag{20}$$

This produces a one-dimensional perturbation distribution:

$$z \in R^N. \tag{21}$$

We then construct Q-Q plots between z and a standard Gaussian distribution to evaluate whether the projected perturbations exhibit Gaussian-like behavior. Compared with inspecting individual hidden dimensions, random projection provides a less coordinate-dependent view of the high-dimensional perturbation distribution.

B.4 Results

Figure X presents additional results under different perturbation scales.

Across all perturbation scales, both input-level and model-level perturbations consistently induce approximately Gaussian-like changes in final hidden representations. As perturbation strength increases, input-level perturbations exhibit heavier tails, while model-level perturbations remain relatively stable.

These results empirically support our unified perturbation perspective: perturbations introduced at different positions can be viewed as inducing effective perturbations in the final hidden representation space.

C Metric Details

To comprehensively evaluate fingerprint attack methods, we consider both attack effectiveness and model harmlessness. In addition to standard metrics such as ASR, ACC, and PPL, we further introduce a unified metric named Attack Utility Score (AUS).

C.1 Attack Success Rate (ASR)

Attack Success Rate measures whether the fingerprint response is successfully suppressed after applying an attack.

Given a fingerprint test set $\mathcal{D}_{fp}$ containing trigger-response pairs, ASR is defined as:

$$ASR = \frac{1}{|\mathcal{D}_{fp}|} \sum_{(x,y)\in\mathcal{D}_{fp}} I\left[f(x) \neq y\right], \quad (22)$$

where x denotes the fingerprint trigger, y denotes the corresponding fingerprint response, and $f(\cdot)$ denotes the attacked model.

A higher ASR indicates stronger fingerprint suppression capability.

C.2 Downstream Accuracy (ACC)

To evaluate whether the attacked model preserves its original task capabilities, we measure downstream performance on reasoning and knowledge benchmarks.

Specifically, we evaluate accuracy on:

- MMLU
- GSM8K
- BBH

Let Acc_{clean} denote the original model accuracy and Acc_{attack} denote the post-attack accuracy.

The accuracy degradation is defined as:

$$\Delta ACC = \frac{Acc_{clean} - Acc_{attack}}{Acc_{clean}}. \quad (23)$$

Lower ΔACC indicates better harmlessness.

C.3 Perplexity (PPL)

Perplexity evaluates whether the attack damages the model's open-ended generation quality.

Given a generated sequence $\{w_1, w_2, \ldots, w_m\}$, the perplexity is defined as:

$$PPL(w_1, \ldots, w_m) = \left(\prod_{j=1}^{m} P(w_j|w_{<j})\right)^{-\frac{1}{m}} \quad (24)$$

We evaluate PPL on:

- Alpaca
- Dolly

Let PPL_{clean} and PPL_{attack} denote the perplexity before and after attack.

The relative degradation is defined as:

$$\Delta PPL = \frac{PPL_{attack} - PPL_{clean}}{PPL_{clean}} \quad (25)$$

Lower ΔPPL indicates better preservation of generation fluency.

C.4 Attack Utility Score (AUS)

Existing evaluations typically report attack effectiveness and harmlessness separately. However, this makes it difficult to directly compare methods under the effectiveness-utility trade-off. To address this issue, we introduce Attack Utility Score (AUS), which jointly evaluates attack success and utility preservation.

We first define the normalized accuracy degradation as:

$$D_{\mathrm{ACC}} = \max\left(0, \Delta ACC\right). \quad (26)$$

Similarly, the normalized perplexity degradation is defined as:

$$D_{\mathrm{PPL}} = \min\left(1, \max\left(0, \frac{\log \Delta PPL}{\log 2}\right)\right). \quad (27)$$

The final AUS is computed as:

$$AUS = ASR \cdot \left(1 - \lambda D_{\mathrm{ACC}} - (1-\lambda) D_{\mathrm{PPL}}\right), \quad (28)$$

where λ balances downstream task preservation and generation fluency preservation. Following our main experiments, we set $\lambda = 0.5$.

A higher AUS indicates a better overall trade-off between attack effectiveness and harmlessness.

D More Experiment Result

D.1 Main Result

In the main paper, we present the primary experimental results on LLaMA3.1-8B-It due to space limitations. To further validate the robustness and generalizability of RIPPLE, we provide complete experimental results on Qwen2.5-7B-It and Mistral-7B-v0.1.

Results on Qwen2.5-7B-It

Table 8 presents the complete experimental results on Qwen2.5-7B-It. RIPPLE consistently achieves strong fingerprint suppression performance across all fingerprinting methods while preserving the original model utility. Compared with

Method	Setting	IF				C&H				ImF			
		PPL	ACC (%)	ASR (%)	AUS (%)	PPL	ACC (%)	ASR (%)	AUS (%)	PPL	ACC (%)	ASR (%)	AUS (%)
Baseline	–	19.3	14.0	0.0	0.0	20.8	10.0	0.0	0.0	17.8	17.8	0.0	0.0
Meraser	5e-4	**6.1**	23.0	0.0	0.0	**6.2**	23.0	40.0	40.0	**6.2**	23.0	20.0	20.0
	1e-3	9.1	20.0	90.0	90.0	8.4	14.0	50.0	50.0	15.2	20.0	50.0	50.0
	5e-3	69.7	3.0	**100.0**	10.7	56.7	5.0	**100.0**	24.9	37.0	3.0	**100.0**	8.4
FT	alpaca	10.4	33.5	0.0	0.0	9.1	**32.1**	0.0	0.0	13.1	**33.5**	0.0	0.0
	dolly	10.6	**35.1**	0.0	0.0	11.2	29.0	0.0	0.0	12.3	30.6	0.0	0.0
Merge	6/4	13.8	17.6	0.0	0.0	15.3	17.2	0.0	0.0	15.9	21.9	0.0	0.0
	5/5	14.1	16.6	0.0	0.0	14.5	18.0	0.0	0.0	17.8	23.9	0.0	0.0
	4/6	13.1	20.0	0.0	0.0	13.7	19.4	0.0	0.0	16.6	23.1	0.0	0.0
GRI	–	22.8	10.9	**100.0**	77.0	26.3	9.4	0.0	0.0	27.5	17.4	0.0	0.0
Ours	$\alpha = 0.05$	19.7	14.0	**100.0**	**98.6**	21.5	10.0	**100.0**	**97.9**	17.8	17.8	80.0	**79.8**

Table 6: Main results on Mistral-7B.

baseline methods, RIPPLE maintains significantly better overall trade-off performance.

Results on Mistral-7B-v0.1

Table 6 presents the complete experimental results on Mistral-7B-v0.1. Similar trends can be observed on Mistral. Parameter-modification baselines often suffer from unstable utility degradation, while heuristic-based baselines exhibit poor attack effectiveness on more robust fingerprinting methods.

In contrast, RIPPLE consistently maintains high ASR while introducing minimal degradation in ACC and PPL.

D.2 Additional Perturbation Position Analysis

Tables 7 and 9 present additional perturbation-position ablation results on Mistral-7B-v0.1 and Qwen2.5-7B-It, respectively. Across both models, input-level and parameter-level perturbations generally exhibit unstable attack performance. In contrast, our hidden-level perturbation achieves substantially higher ASR across most fingerprinting methods. Moreover, hidden-level perturbation introduces only marginal latency and memory overhead compared to the baseline model, while input-level and parameter-level perturbations typically incur significantly higher computational costs.

These results further validate that directly perturbing final hidden representations provides a better balance between attack effectiveness and efficiency.

Fingerprint	Perturbation	ASR (%)	Latency (s)	Memory (MB)
IF	Baseline	0.0	**0.475**	**13892**
	Input-level	20.0	1.095	13940
	Model-level	**100.0**	1.056	14711
	Ours	0.0	0.489	13994
C&H	Baseline	0.0	**0.296**	**13841**
	Input-level	60.0	0.465	13856
	Model-level	30.0	0.439	14692
	Ours	**100.0**	0.336	13868
ImF	Baseline	0.0	**0.391**	**13870**
	Input-level	10.0	0.916	13905
	Model-level	10.0	0.832	14703
	Ours	**90.0**	0.449	13938

Table 7: Perturbation-position ablation results on Mistral-7B-v0.1 under $\tau = 0.99$ and $\alpha = 0.05$. **Bold** and underlined values denote the best and second-best results within each fingerprint method, respectively.

Method	Setting	IF				C&H				ImF			
		PPL	ACC (%)	ASR (%)	AUS (%)	PPL	ACC (%)	ASR (%)	AUS (%)	PPL	ACC (%)	ASR (%)	AUS (%)
Baseline	–	11.7	52.6	0.0	0.0	13.1	49.4	0.0	0.0	12.1	49.9	0.0	0.0
Meraser	5e-4	5.6	51.0	0.0	0.0	5.3	52.0	**100.0**	**100.0**	**5.9**	52.0	20.0	20.0
	1e-3	**5.4**	49.0	20.0	19.3	**5.0**	50.0	90.0	90.0	6.1	49.0	20.0	19.8
	5e-3	6.2	43.0	20.0	0.0	6.6	36.0	70.0	60.5	6.1	38.0	**80.0**	70.4
FT	alpaca	5.7	67.0	0.0	0.0	7.5	63.9	0.0	0.0	6.6	61.7	0.0	0.0
	dolly	8.7	64.1	0.0	0.0	8.9	61.4	0.0	0.0	9.1	60.3	10.0	10.0
Merge	6/4	7.8	66.4	0.0	0.0	8.6	62.7	20.0	20.0	9.1	62.7	0.0	0.0
	5/5	7.0	**68.3**	0.0	0.0	7.6	66.5	90.0	90.0	8.2	64.4	0.0	0.0
	4/6	5.7	67.2	0.0	0.0	6.1	**69.8**	**100.0**	**100.0**	7.2	**65.7**	0.0	0.0
GRI	–	19.0	45.4	**100.0**	58.3	25.0	47.9	0.0	0.0	15.0	51.5	0.0	0.0
Ours	$\alpha = 0.05$	11.8	52.6	**100.0**	**99.7**	13.1	49.4	**100.0**	99.7	12.2	49.0	**80.0**	**78.9**

Table 8: Main results on Qwen2.5-7B-It.

Fingerprint	Perturbation	ASR (%)	Latency (s)	Memory (MB)
IF	Baseline	0.0	**0.427**	**14575**
	Input-level	0.0	0.819	14671
	Model-level	0.0	0.841	15510
	Ours	**100.0**	0.476	14643
C&H	Baseline	0.0	**0.222**	**14547**
	Input-level	10.0	0.359	14576
	Model-level	10.0	0.356	15503
	Ours	**100.0**	0.258	14568
ImF	Baseline	0.0	**0.333**	**14562**
	Input-level	0.0	0.726	14634
	Model-level	20.0	0.637	15507
	Ours	**80.0**	0.389	14613

Table 9: Perturbation-position ablation results on Qwen2.5-7B-It under $\tau = 0.99$ and $\alpha = 0.05$. **Bold** and underlined values denote the best and second-best results within each fingerprint method, respectively.