Provable Watermarking for Data Poisoning Attacks

Yifan Zhu[1, 2], **Lijia Yu**[3*], **Xiao-Shan Gao**[1, 2*]
[1]State Key Laboratory of Mathematical Sciences,
Academy of Mathematics and Systems Science, Chinese Academy of Sciences
[2]University of Chinese Academy of Sciences
[3]Institute of AI for Industries, Nanjing, China
zhuyifan@amss.ac.cn, ljyu@iaii.ac.cn, xgao@mmrc.iss.ac.cn

Abstract

In recent years, data poisoning attacks have been increasingly designed to appear harmless and even beneficial, often with the intention of verifying dataset ownership or safeguarding private data from unauthorized use. However, these developments have the potential to cause misunderstandings and conflicts, as data poisoning has traditionally been regarded as a security threat to machine learning systems. To address this issue, it is imperative for harmless poisoning generators to claim ownership of their generated datasets, enabling users to identify potential poisoning to prevent misuse. In this paper, we propose the deployment of watermarking schemes as a solution to this challenge. We introduce two provable and practical watermarking approaches for data poisoning: *post-poisoning watermarking* and *poisoning-concurrent watermarking*. Our analyses demonstrate that when the watermarking length is $\Theta(\sqrt{d}/\epsilon_w)$ for post-poisoning watermarking, and falls within the range of $\Theta(1/\epsilon_w^2)$ to $O(\sqrt{d}/\epsilon_p)$ for poisoning-concurrent watermarking, the watermarked poisoning dataset provably ensures both watermarking detectability and poisoning utility, certifying the practicality of watermarking under data poisoning attacks. We validate our theoretical findings through experiments on several attacks, models, and datasets.

1 Introduction

Data poisoning [7, 43, 71] is a well-established security concern for modern ML systems. Its significance has become increasingly pronounced in the era of large-scale models, where many models are trained on web-crawl or synthetic data without rigorous selection [66, 10, 77]. There are two representative data poisoning attacks, *backdoor attacks* [13, 75] and *availability attacks* [37, 22]. Backdoor attacks involve creating poisoned datasets that cause models trained on them to predict the specific targets when a particular trigger is injected into test instances. Availability attacks aim to compromise model generalization by ensuring that models trained on poisoned datasets have low test accuracy. Deploying models on backdoor and availability attacked datasets poses severe security risks. For instance, in autonomous driving systems, triggered road signs created by backdoor attacks could be misclassified by object detectors, leading to potentially catastrophic accidents [27, 31]. Availability attacks directly undermine model utility, rendering AI-based systems nonfunctional [8].

However, interestingly, things are always two-faced. Modern data poisoning attacks are increasingly being designed to be harmless and purposeful. For example, backdoor attacks have been employed for black-box dataset ownership verification [50, 51], availability attacks have been utilized to prevent the unauthorized use of data [37, 23]. More recently, methods like NightShade [65] and Glaze [64] have been developed to protect artists' intellectual property from generative AI models. These

*Corresponding authors

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

advancements illustrate the promising potential of "data poisoning for good," transforming data poisoning attacks—traditionally viewed as harmful—into tools that can benefit society. Nevertheless, unintended consequences may arise. An innocent, authorized user might inadvertently use poisoned data, leading to potential misunderstandings and conflicts. To mitigate such risks, the poisoning generators must transparently disclose the presence of potential poisoning to their intended users. For example, when an artist distributes his works to a copyright protection system, the system (poisoner) not only aims to prevent unauthorized use but also bears the responsibility of informing clients and authorized users if the data has been perturbed. Such transparency is essential to ensure trust and avoid unintended harm in these beneficial applications of data poisoning.

To address the challenges and ensure the transparency of poisoned datasets, a direct approach is to design detection methods capable of identifying potential poisoning. While many studies have focused on detecting backdoor and availability attacks [11, 19, 18, 98, 89], these detection methods vary significantly across different types of attacks, making it challenging to unify as a single, cohesive framework for distribution to authorized users. Additionally, existing detection methods often rely on heuristic training algorithms, lacking a provable mechanism for claiming poisoning. This limitation can lead to disputes if a poisoned dataset is inadvertently misused, as the absence of a clear, verifiable claim undermines accountability. To overcome these challenges, we explore the use of watermarking [9, 1, 41], a widely adopted approach for copyright protection and the detection of AI-generated content, which presents a promising solution for poisoners to provably declare the existence of poisoning, thereby enhancing transparency and minimizing the risk of disputes.

In this paper, we propose two provable and practical watermarking approaches for data poisoning: *post-poisoning watermarking* and *poisoning-concurrent watermarking*. The former addresses scenarios where the poisoning generators require a third-party entity to create watermarks for their poisoned datasets, while the latter focuses on cases where the poisoning generators craft watermarks themselves. In Section 4.1, we demonstrate that when watermarking is sample-wise for each data, discernment of poisoned data with high probability is achievable if a specific key is available when the required watermarking length is $\Omega(\sqrt{d}/\epsilon_w)$ and $\Omega(1/\epsilon_w^2)$ for post-poisoning and poisoning-concurrent watermarking respectively (d is the data dimension, ϵ_w is the watermarking budget). However, the sample-wise approach necessitates N distinct watermarks and keys for a dataset with N samples, which can be impractical for large datasets. To address this limitation, we consider a more meaningful scenario where a single watermark and key apply to all data instances. In Section 4.2, recognizing that reliance on the sample size N is not ideal for universal watermarking, we extend our analysis to watermarking effective on most samples and then generalizes to the whole distribution with high probability. Specifically, we prove that when the post-poisoning and poisoning-concurrent watermarking lengths are $\Theta(\sqrt{d}/\epsilon_w)$ and $\Theta(1/\epsilon_w^2)$ respectively, the majority of poisoned data can be effectively identified. Moreover, if the sample size satisfies $N = \Omega(d)$, these results can be generalized to the entire data distribution.

Beyond demonstrating the effectiveness of watermarking, in Section 5, we further show that the injected watermarks have minimal impact on the poisoning. Specifically, for post-poisoning watermarking, when the data dimension d and sample size N are large, the generalization gap between the original poisoned distribution and the watermarked poisoned dataset is bounded by negligible terms. For poisoning-concurrent watermarking, achieving a small generalization gap requires an additional condition: the watermarking length should satisfy $O(\sqrt{d}/\epsilon_p)$, where ϵ_p is the poisoning budget.

Our theoretical analyses confirm that the effectiveness of watermarked data poisoning is maintained under specific watermarking lengths. For post-poisoning watermarking, both watermarking and poisoning remain effective when the length is $\Theta(\sqrt{d}/\epsilon_w)$. For the poisoning-concurrent watermarking, the effectiveness is certified when the length falls within the range of $\Theta(1/\epsilon_w^2)$ to $O(\sqrt{d}/\epsilon_p)$. Consequently, if the poisoning generator relies on a third-party entity for watermarking, using a larger length is advantageous. In comparison, if the generator directly embeds the watermark into their poisoned dataset, a moderate length is more practical. In Section 6, we evaluate several existing backdoor and availability attacks to empirically validate our theoretical findings.

2 Related Work

Data Poisoning. Data poisoning attacks modify the training data within a small perturbation budget, aiming to elicit unusual behaviors for models trained on the poisoned dataset. One prominent type of

data poisoning is backdoor attacks [13, 27, 79, 80, 95, 52, 62, 75, 55, 91, 88]. These attacks inject specific patterns into the training data, causing the trained model to behave anomalously when test instances contain such patterns. Other works [50, 51] have utilized backdoor attacks to achieve dataset ownership verification. Another category is availability attack, also referred to as indiscriminate attacks [7, 59, 21, 22, 44, 53, 86]. These attacks aim to degrade the model's overall test accuracy. Recently, unlearnable examples [37, 23, 32, 70, 12, 67, 92, 97, 83] as the case of imperceptible availability attacks, have been designed to protect data from illegal use by unauthorized trainers. Further data poisoning schemes include targeted attacks [43, 71, 30, 25, 4], which cause models to malfunction on some specific data. In this paper, we mainly focus on imperceptible clean-label backdoor attacks and availability attacks, as they are more practical in real-world scenarios.

Watermarking. Watermarking involves embedding special signals into training data or models to enhance copyright protection and identify data ownership [61, 5, 40, 96, 3, 73, 82]. People [50, 51] introduced backdoor attacks as the dataset watermark for data verification, while [29] proposed the domain watermark with harmless verification. Recently, watermarking of large language models has gained significant attention for AI-generated text detection [41, 35, 47, 42, 93, 15]. Watermarking has also been investigated for generative image models [85, 93, 28]. This paper focuses on watermarking for poisoning attacks. We provide two provable, simple, and practical watermarking schemes: post-poisoning and poisoning-concurrent watermarking. To the best of our knowledge, this is the first work to leverage watermarking schemes in the context of data poisoning attacks.

3 Preliminaries

3.1 Data Poisoning

We assume the data always lies in $[0,1]^d$. To ensure consistency across different criteria, we focus on imperceptible clean-label data poisoning attacks, which are more practical in real-world applications. Specifically, we denote the attack as a mapping $\delta^p : [0,1]^d \to [-\epsilon_p, \epsilon_p]^d$. For each data x, the attack δ^p perturbs the data to produce a modified version $x' = x + \delta^p(x)$, while ensuring $\|\delta^p(x)\|_\infty \leq \epsilon_p$ to preserve imperceptibility. For simplicity, we denote $\delta^p(x)$ as δ^p_x.

Goal. The poisoning objective risk is defined as $\mathcal{R}^{\text{poi}}(\mathcal{D}^{\text{victim}}, \mathcal{F})$, where $\mathcal{D}^{\text{victim}}$ represents the victim distribution. The goal of data poisoning is to construct a poisoned distribution $\mathcal{D}'$, such that if the risk $\mathcal{R}(\mathcal{D}', \mathcal{F}) = \mathbb{E}_{(x,y)\sim\mathcal{D}'}\mathcal{L}\left(\mathcal{F}(x), y\right)$ is small, the network $\mathcal{F}$ achieves a small poisoning objective risk $\mathcal{R}^{\text{poi}}(\mathcal{D}^{\text{victim}}, \mathcal{F})$. In other words, when $\mathcal{F}$ has effectively learned the poison features of $\mathcal{D}'$, it is expected to exhibit specific properties aligned with the objectives of data poisoning. For example, in availability attacks, $\mathcal{D}^{\text{victim}} = \mathcal{D}$, the objective risk $\mathcal{R}^{\text{poi}}(\mathcal{D}^{\text{victim}}, \mathcal{F}_{S'}) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[-\mathcal{L}(\mathcal{F}(x), y)\right]$, where the goal is to obtain a network $\mathcal{F}$ with high loss on $\mathcal{D}$, thereby degrading its generalization performance. In backdoor attacks, $\mathcal{D}^{\text{victim}} = \mathcal{D} \oplus T$, $\mathcal{R}^{\text{poi}}(\mathcal{D}^{\text{victim}}, \mathcal{F}_{S'}) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\mathcal{L}(\mathcal{F}(x \oplus T), y^{\text{target}})\right]$, where T is the trigger injected during inference, y^{target} is the targeted label. The goal of backdoor attacks is to ensure that any data x with trigger T be classified as y^{target}.

3.2 Watermarking

Watermark and key. The goal of watermarking on data poisoning attacks is to ensure that verified users are aware of whether the given data has been poisoned, to prevent potential misunderstandings when data creators use data poisoning attacks to achieve specific objectives. e.g., crafting unlearnable examples to deter unauthorized use of data. In this paper, we mainly focus on dataset watermarking [51], where watermarks are embedded in datasets for verification. Specifically, similar to the data poisoning attack δ^p, we denote a watermarking as a mapping $\delta^w : [0,1]^d \to [-\epsilon_w, \epsilon_w]^q$, and use δ^w_x to represent $\delta^w(x)$ for simplicity, where $q \leq d$ is the watermarking length, and the watermarking dimension indices are $\mathcal{W} = \{d_1, d_2, \cdots, d_q\} \subset [d]$. When a dataset is watermarked, authorized users are provided with a corresponding key to detect whether the data contains watermarks. In this paper, we assume that the key ζ is a d-dimensional vector. The watermarking detector uses a simple mechanism, computing the inner product $\zeta^T x$ to determine whether x has been watermarked.

Post-poisoning watermarking. In this scenario, a third-party entity serves as the watermark generator, crafting watermarks for a given poisoned dataset. The goal is to enable authorized detectors to identify potential poisoned data. Denote the poison and the watermark as δ^p_x and δ^w_x respectively,

where $\|\delta_x^w\|_\infty \leq \epsilon_w$, $\|\delta_x^p\|_\infty \leq \epsilon_p$. Both watermark δ_x^w and poison δ_x^p rely on data x, and the overall perturbation is $\delta_x = \delta_x^p + \delta_x^w$. For simplicity, we denote the perturbation for data x_i as $\delta_i = \delta_{x_i}$.

Poisoning-concurrent watermarking. In this scenario, the watermark generator also acts as the poison generator, simultaneously crafting both watermarks and poisons. The objective is to achieve the goals of data poisoning while ensuring authorized detectors can identify the poisoned data. Since the watermark generator can control the poison dimensions, we assume the generator separates the dimensions used for watermarking and poisoning. Specifically, the dimensions for poisoning are indexed by $\mathcal{P} = [d] \backslash \mathcal{W}$. Other notations remain consistent with those at post-poisoning watermarking.

To make notations clearer, we provide a symbol table in Appendix A.

3.3 A Practical Threat Model

Any copyright owner can deploy our watermarking when releasing their original datasets to a third party (e.g., AI training platforms, academic institutions, and copyright certification systems). To make the threat model more concrete, we provide a detailed deployment scenario below.

A company (called Alice) that collects a large proprietary dataset for autonomous driving research (e.g., dash cam video frames). She wants to open source a part of her dataset to promote innovation for the community (e.g., Non-profit research organization), but also wants to prevent unlicensed users from training a machine learning model on it successfully to protect her intellectual property. To achieve the above goals, Alice runs our poisoning + watermarking algorithm on every instance of her dataset, publishing the perturbed (i.e., protected) dataset which is unlearnable by standard models and obtains a secret, key-dependent watermark signal. She publishes this on her GitHub under a permissive license, accompanied by a SHA256 hash so any recipient can verify integrity.

A research lab (called Bob) registers on Alice's portal and agrees to a standard agreement for legal use of the dataset. After approval by Alice, Bob receives a secret key (e.g., a 128 bit seed) provided via Alice's portal's secure HTTPS channel. Furthermore, Bob also gains a pipeline (e.g., Python pre-processing package) from Alice such that he can run the watermark detection to verify his identity and ensure that there is no file corruption. After the verification, Bob can run an algorithm designed by Alice (e.g., directly adding inverse unlearnable noise for each data) to remove the unlearnable poisons. If the pipeline receives the wrong key or a tampered file, the detection fails and the poisons cannot be removed to ensure the unlearnability.

For a malicious user (called Chad), first, Chad can download the same public poisoned and watermarked dataset, but cannot train a good model on it because the dataset is unlearnable. If Chad tries to remove or tamper with watermarks and unlearnable poisons without knowing the secret key, detection will fail.

For key management, Alice can rotate keys per month and publish on her portal only to approved accounts (i.e., trusted users). Alice can also add a HMAC scheme to prevent potential forgery risks. Specifically, Alice can rotate keys per month and publish on her portal only to approved accounts (i.e., trusted users). Alice can also add a HMAC scheme to prevent potential forgery risks. Specifically, we can separate keys into generation key k_{gen} and authentication key k_{auth}, where k_{gen} is completely the same as our paper and correlates with injected watermarks w_i for every data x_i. For each perturbed $\hat{x_i}$ with w_i, we can compute an additional tag t_i by HMAC under k_{auth}, i.e., $t_i = \text{HMAC}_{k_{auth}}(id_i, \hat{x_i})$, where id_i is a unique identifier for the image x_i (e.g., index). After that, we store the (id_i, t_i) pair (e.g., through a sidecar JSON) for later detection. In watermarking detection, beyond traditional detection using k_{gen} and $\hat{x_i}$, we also verify the tag with t_i and $t_i = \text{HMAC}_{k_{auth}}(id_i, \hat{x_i})$ to avoid potential forgery attacks. In this case, even if the generation key k_{gen} leaks, an attacker cannot forge a new valid (x_i, t_i) pair as they lack the authentication key k_{auth}. We can keep k_{auth} in a secure enclave and rotate it independently with k_{gen} to enhance the security.

4 Soundness of Watermarking

In this section, we provide theoretical guarantees for the conditions under which watermarking can effectively differentiate between poisoned and benign data. We begin by examining a specific version where the watermarking is sample-wise. In this case, the injected watermark δ_x^w relies on x, meaning that the watermark generator can assign a unique watermark to each data.

4.1 Sample-wise Version

We first analyze the sample-wise version of post-poisoning watermarking. Proofs of theorems in this subsection are provided in Appendix B.1.

Theorem 4.1 (Sample-wise, post-poisoning watermarking)**.** *For any data point x sampled from $\mathcal{D}_{\mathcal{X}}$ and their corresponding poison be δ_x^p, there exists a distribution Ξ defined in $\mathbb{R}^d$ such that we can sample the key $\zeta \sim \Xi$ satisfying that for any $\omega \in (0,1)$, there are:*

(1): $\mathbb{P}_{x\sim\mathcal{D}_{\mathcal{X}},\zeta\sim\Xi}\left(\zeta^T x < \sqrt{\frac{d}{2}\log\frac{1}{\omega}}\right) > 1-w$; (2): *we can craft the watermark δ_x^w based on ζ such that $\mathbb{P}_{x\sim\mathcal{D}_{\mathcal{X}},\zeta\sim\Xi}\left(\zeta^T(x+\delta_x) > q\epsilon_w - \sqrt{\frac{d}{2}\log\frac{1}{\omega}}\right) > 1-w$. Hence, when $q > \frac{1}{\epsilon_w}\sqrt{2d\log\frac{1}{\omega}}$, it holds that $\mathbb{P}_{x_1,x_2\sim\mathcal{D}_{\mathcal{X}},\zeta\sim\Xi}\left(\zeta^T(x_1+\delta_1) > \zeta^T x_2\right) > 1-2\omega$.*

Remark 4.2. For the sample-wise, post-poisoning watermarking with the data length d and watermark budget ϵ_w, crafting an effective watermark requires the watermarking length to be $\Omega(\sqrt{d}/\epsilon_w)$.

Next, we analyze the scenario of sample-wise version for poisoning-concurrent watermarking.

Theorem 4.3 (Sample-wise, poisoning-concurrent watermarking)**.** *For any $x \sim \mathcal{D}_{\mathcal{X}}$, there exists a distribution $\Xi \in \mathbb{R}^d$ such that we can sample the key $\zeta \sim \Xi$ satisfied that for any $\omega \in (0,1)$:*

(1): $\mathbb{P}_{x\sim\mathcal{D}_{\mathcal{X}},\zeta\sim\Xi}\left(\zeta^T x < \sqrt{\frac{q}{2}\log\frac{1}{\omega}}\right) > 1-w$; (2): *we can craft the watermark δ_x^w and poison δ_x^p such that $\mathbb{P}_{x\sim\mathcal{D}_{\mathcal{X}},\zeta\sim\Xi}\left(\zeta^T(x+\delta_x) > q\epsilon_w - \sqrt{\frac{q}{2}\log\frac{1}{\omega}}\right) > 1-w$. Hence, when $q > \frac{2}{\epsilon_w^2}\log\frac{1}{\omega}$, it holds that $\mathbb{P}_{x_1,x_2\sim\mathcal{D}_{\mathcal{X}},\zeta\sim\Xi}(\zeta^T(x_1+\delta_1) > \zeta^T x_2) > 1-2\omega$.*

Remark 4.4. For the sample-wise, poisoning-concurrent watermarking with the data length d and watermark budget ϵ_w, crafting an effective watermark requires the watermarking length to be $\Omega(1/\epsilon_w^2)$.

Remark 4.5. The required length for poisoning-concurrent watermarking $\Omega(1/\epsilon_w^2)$ is smaller than that for post-poisoning watermarking $\Omega(\sqrt{d}/\epsilon_w)$. This difference arises because the condition $q \le d$ for the watermarking length always holds. Therefore, we have $\epsilon_w \ge O(1/\sqrt{d})$, which implies $\Omega(\sqrt{d}/\epsilon_w) \ge \Omega(1/\epsilon_w^2)$.

Theorems 4.1 and 4.3 suggest that, with high probability, as long as the watermark dimension q reaches the required thresholds ($\Omega(\sqrt{d}/\epsilon_w)$ or $\Omega(1/\epsilon_w^2)$), the inner product of key and poisoned data will exceed a constant C_1, while the inner product of key and clean data will remain below a constant $C_2 < C_1$. As a result, a detector can simply select a threshold $T = \frac{C_1+C_2}{2}$ to effectively differentiate between poisoned and clean data using the given key. Based on these observations, we derive the following corollary:

Corollary 4.6. *For sample-wise, post-poisoning watermarking, if the watermarking length $q \ge \frac{2}{\epsilon_w}\sqrt{2d\log\frac{1}{\omega}}$, with probability at least $1-2\omega$, for the sampled key $\zeta \in \mathbb{R}^d$ and data x_1, x_2 sampled from $\mathcal{D}_{\mathcal{X}}$, it is possible to craft the watermark δ_x^w such that $\zeta^T(x_1+\delta_1) > \frac{3}{4}q\epsilon_w, \zeta^T x_2 < \frac{1}{4}q\epsilon_w$. Similarly, for poisoning-concurrent watermarking, if $q \ge \frac{8}{\epsilon_w^2}\log\frac{1}{\omega}$, we can craft the watermark δ_x^w such that $\zeta^T(x_1+\delta_1) > \frac{3}{4}q\epsilon_w, \zeta^T x_2 < \frac{1}{4}q\epsilon_w$.*

However, the sample-wise watermarking requires an individual key for each sample, which is impractical in real-world applications. A more ideal case is that the watermark detector can use a single key applicable to all samples, making detection more effective and efficient. This motivates the consideration of the universal version of watermarking, where the injected watermark δ_x^w is identical for every x. In this case, scenario, for simplicity, we denote $\delta_x^w = \delta^w$.

4.2 Universal Version

In the universal version, a single detection key is employed, violating the condition of Theorems 4.1 and 4.3, where the key ζ is sampled from a distribution. Consequently, the proof techniques used for the sample-wise case are difficult to generalize to this scenario. Instead, we step away from the distributional guarantees and first analyze the finite-sample case. The theoretical results for the finite case can subsequently be extended to the distributional setting. Proof of theorems in this subsection

are in Appendix B.2. In the finite case, we assume the dataset consists of N samples, denoted as $S_{\mathcal{X}} = \{x_1, x_2, \cdots, x_N\}$. We begin by analyzing the case of post-poisoning watermarking.

Proposition 4.7 (Universal, post-poisoning watermarking)**.** *For the dataset $S_{\mathcal{X}}$, when $q > \frac{2+\epsilon_p}{\epsilon_w}\sqrt{\frac{d}{2}\log\frac{2N}{\omega}}$, we can sample the key $\zeta \in \mathbb{R}^d$ from a certain distribution such that, with probability at least $1-\omega$, there exists the watermark δ^w such that $\zeta^T(x_j+\delta_j) > \zeta^T x_i, \forall i,j \in [N]$.*

We can extend the proof of Proposition 4.7 with a larger q, to achieve a non-vacuous gap between poisoned and benign data, as stated in the following corollary:

Corollary 4.8. *Notations are similar to Proposition 4.7. When $q > \frac{4}{\epsilon_w}\sqrt{\frac{d}{2}\log\frac{2N}{\omega}}$ with probability at least $1-\omega$, there exists the watermark δ^w such that $\zeta^T(x_i+\delta_i) > \frac{q\epsilon_w}{2}, \zeta^T x_i < \frac{q\epsilon_w}{4}, \forall i \in [N]$.*

Proposition 4.7 demonstrates that the watermarking length is expected to be $\Omega(\sqrt{d\log N}/\epsilon_w)$ to ensure universal watermarking discerning every data x_i, which is not ideal as the watermarking length q depends on sample size N, leading to vacuous results when the dataset becomes sufficiently large. To address this limitation and achieve a non-vacuous result, we propose relaxing the properties from discerning every sample to discerning most samples, as described in the following theorem.

Theorem 4.9 (Universal, post-poisoning watermarking for most examples)**.** *For the dataset $S_{\mathcal{X}} = \{x_1, x_2, \cdots, x_N\}$, x_i and the poison δ_p^i are i.i.d. sampled from $\mathcal{D}_{\mathcal{X}}$ and $\mathcal{D}_{\mathcal{P}}$ respectively. For any $w \in (0, 1/2)$ and $q > \frac{2}{\epsilon_w}\sqrt{2d\log\frac{1}{\omega}}$, we can sample the key $\zeta \in \mathbb{R}^d$ from a certain distribution such that, with probability at least $1 - 2\exp\left(\frac{-N(\omega - e^{-q^2\epsilon_w^2/8d})^2}{\omega + e^{-q^2\epsilon_w^2/8d}}\right)$, we can craft the watermark δ^w, such that $\zeta^T(x_i+\delta_i) > \frac{q\epsilon_w}{2}, \zeta^T x_i < \frac{q\epsilon_w}{4}$ holds for at least $(1-2\omega)N$ samples.*

Remark 4.10. Theorem 4.9 suggests that when the sample size N is sufficiently large and the watermark length $q \gtrsim \frac{2}{\epsilon_w}\sqrt{2d\log\frac{1}{\omega}} = \Theta(\sqrt{d}/\epsilon_w)$, the universal watermarking is effective for most samples with high probability. Thus, if we relax the requirement and only demand that the watermarking is effective for most samples, Theorem 4.9 indicates that the required watermarking length no longer depends on N, unlike in Proposition 4.7.

We then analyze the finite universal case for poisoning-concurrent watermarking.

Proposition 4.11 (Universal, poisoning-concurrent watermarking)**.** *For the dataset $S_{\mathcal{X}}$, when $q > \frac{4}{\epsilon_w^2}\log\frac{N}{\omega}$, it is possible to sample the key $\zeta \in \mathbb{R}^d$ from a certain distribution such that, with probability at least $1-\omega$, we can craft watermark δ^w and poison δ^p such that $\zeta^T(x_j+\delta_j) > \zeta^T x_i, \forall i,j \in [N]$.*

Similar to post-poisoning watermarking, we can derive analogous results for poisoning-concurrent watermarking about non-vacuous gaps and cases on most examples.

Corollary 4.12. *Notations are similar to Prop 4.11. When $q > \frac{9}{\epsilon_w^2}\log\frac{N}{\omega}$, with probability at least $1-\omega$, we can craft watermark δ^w and poison δ^p, such that $\zeta^T(x_i+\delta_i) > \frac{2q\epsilon_w}{3}, \zeta^T x_i < \frac{q\epsilon_w}{3}, \forall i \in [N]$.*

Theorem 4.13 (Universal, poisoning-concurrent watermarking for most examples)**.** *For the dataset $S_{\mathcal{X}} = \{x_1, x_2, \cdots, x_N\}$, where x_i is i.i.d. sampled from $\mathcal{D}_{\mathcal{X}}$. For any $\omega \in (0,1)$ and $q > \frac{9}{2\epsilon_w^2}\log\frac{1}{\omega}$, we can sample the key $\zeta \in \mathbb{R}^d$ from a certain distribution such that, with probability at least $1 - \exp\left(\frac{-N(\omega - e^{-2q\epsilon^2/9})^2}{\omega + e^{-2q\epsilon^2/9}}\right)$, we can craft the watermark and the poison satisfies $\zeta^T(x_i+\delta_i) > \frac{2q\epsilon_w}{3}, \zeta^T x_i < \frac{q\epsilon_w}{3}$ holds for at least $(1-\omega)N$ samples.*

Remark 4.14. Theorem 4.13 indicates that for a sufficiently large N and $q \gtrsim \frac{9}{2\epsilon_w^2}\log\frac{1}{\omega} = \Theta(1/\epsilon_w^2)$, the universal, poisoning-concurrent watermarking is effective for most samples with high probability. Compared with Proposition 4.11, the condition of the watermarking length q in Theorem 4.13 is independent of the sample size N.

After establishing results for the finite case for most samples, we can extend these guarantees to the entire data distribution, as presented in the following theorem.

Theorem 4.15 (Generalization of universal watermarking to distributional case)**.** *For the dataset $S_{\mathcal{X}} = \{x_1, x_2, \cdots, x_N\}$, data x_i and poison δ_p^i are i.i.d. sampled from $\mathcal{D}_{\mathcal{X}}$ and $\mathcal{D}_{\mathcal{P}}$ respectively.*

Consider a universal watermark δ^w, with probability at least $1-2\mu$ for the sampled data and poisons, if there exists a key ζ that satisfies $\zeta^T(x_i+\delta_i)>C_1, \zeta^T x_i<C_2$, for at least $(1-\omega)N$ samples x_i, it has

$$\mathbb{P}_{x,\tilde{x}\sim\mathcal{D}_\mathcal{X},\delta^p\sim\mathcal{D}_\mathcal{P}}\left(\left\{\zeta^T(x+\delta^p+\delta^w)>C_1,-\zeta^T\tilde{x}<C_2\right\}\right)>1-2\omega-2\sqrt{\frac{d}{N}\left(\log\frac{2N}{d}+1\right)-\frac{1}{N}\log\frac{\mu}{4}}.$$

Remark 4.16. When the sample size N is greater than $\Omega(d)$, the effectiveness of watermarks in the finite case can, with high probability, be generalized to the distributional case.

Generalizing the universal watermarking from finite cases to distributional cases does not impose additional conditions on the watermarking length q. For universal, post-poisoning watermarking, as noted in Remark 4.10, an effective watermark for the distribution $\mathcal{D}_\mathcal{X}$ exists when $q=\Theta(\sqrt{d}/\epsilon_w), N=\Omega(d)$. For universal, poisoning-concurrent watermarking, as noted in Remark 4.14, an effective watermark exists when $q=\Theta(1/\epsilon_w^2), N=\Omega(d)$. Compared with sample-wise watermarking, achieving effective universal watermarking for a data distribution does not require more watermarking length q. The only additional requirement is that the dataset size N is not too small (at least $\Omega(d)$), which is a reasonable condition for generalization in practical scenarios.

5 Soundness of Poisoning under Watermarking

In this section, we prove that poisoning remains effective under watermarking for an L-layer feed-forward neural network. For simplicity, we focus on universal watermarking as it is more practical; similar properties also apply to sample-wise watermarking. To facilitate theoretical analyses, we adopt the widely used Xavier normalization [26] for network parameters, which is also employed in Neural Tangent Kernel (NTK) [39] and many other theoretical works [20, 38, 87, 74, 63]. The proofs for this section are provided in Appendix B.3.

Assume the (normalized) L-layer feed-forward neural network is $\mathcal{F}:\mathbb{R}^d\to\mathbb{R}$ defined as $\mathcal{F}(x)=W^L\frac{1}{\sqrt{d_{L-1}}}\mathrm{ReLU}(W^{L-1}\cdots\frac{1}{\sqrt{d_2}}\mathrm{ReLU}(W^2\frac{1}{\sqrt{d_1}}\mathrm{ReLU}(W^1x+b^1)+b^2)+\cdots+b^{L-1})+b^L$ where $\mathrm{ReLU}(x)=\max(0,x)$ is the activation function, $W^l\in\mathbb{R}^{d_l\times d_{l-1}}$ and $b^l\in\mathbb{R}^{d_l}$ are the weight matrix and the bias term of the l-th layer respectively for $l\in[L]$. We consider a binary classification task where the data distribution $\mathcal{D}\in[0,1]^d\times\{-1,1\}$. Here $d_0=d$ and $d_L=1$. We also assume that $d_1\geq d$ as modern neural networks are typically larger and tend to be overparameterized [46, 6, 10, 2]. The loss function used is the cross-entropy loss: $\mathcal{L}(\mathcal{F}(x),y)=\log(1+e^{-y\cdot\mathcal{F}(x)})$.

Definition 5.1 (Optimal Classifier)**.** We define the optimal classifier for a dataset S under the hypothesis space $\mathcal{F}$ as $\mathcal{F}_S^*=\arg\min_{\mathcal{F}}\frac{1}{|S|}\sum_{(x,y)\in S}\mathcal{L}(\mathcal{F}(x),y)$, where $\mathcal{L}$ is the loss function.

Theorem 5.2 (Impact of Watermarking)**.** *With probability at least $1-2\omega$ for the poisoned dataset $\{(x_i',y_i)\}_{i=1}^N=S'\sim\mathcal{D}'$ and the key $\zeta\in\mathbb{R}^d$ selected from a certain distribution, we can craft the watermark δ^w satisfying:*

$$\mathcal{R}(\mathcal{D}',\mathcal{F}^*_{S'+\delta^w})\leq\mathbb{E}_\eta\frac{1}{N}\sum_{i=1}^N\mathcal{L}(\mathcal{F}^*_{S'}(x_i'+\eta),y_i)$$
$$+O\left(\sqrt{\frac{L}{N}}\right)+O\left(\sqrt{\frac{\log d}{N}}\right)+O\left(\sqrt{\frac{\log 1/\omega}{N}}\right)+O\left(\epsilon_w\sqrt{\frac{q\log 1/\omega}{d}}\right),$$

where $S'+\delta^w=\{(x_i'+\delta^w,y_i)\}_{i=1}^N$ is the watermarked dataset, $\eta\sim\mathcal{U}\{-\epsilon_w,\epsilon_w\}^q$ is a random vector.

Remark 5.3. Since η is a random noise under budget ϵ_w, the optimal classifier $\mathcal{F}^*_{S'}$ tends to have small loss under perturbation η, resulting in $\mathbb{E}_\eta L(\mathcal{F}^*_{S'}(x_i+\eta),y_i)$ being small. Furthermore, if d and N are large enough, four error terms in Theorem 5.2 are all small when the post-poisoning condition in Section 4.2, $q=\Theta(\sqrt{d}/\epsilon_w)$ holds, resulting in a small $\mathcal{R}(\mathcal{D}',\mathcal{F}^*_{S'+\delta^w})$.

To ensure the soundness of watermarked poisoning, we assume that the (un-watermarked) poisoning distribution $\mathcal{D}'$ is effective. First, we provide the definition of an effective poisoning distribution.

Assumption 5.4 ((λ,μ)-effective poisoning distribution)**.** A poisoning distribution $\mathcal{D}'$ is called (λ,μ)-effective (for victim distribution $\mathcal{D}^{\mathrm{victim}}$ and poisoning objective risk $\mathcal{R}^{\mathrm{poi}}$), if $\mathcal{R}^{\mathrm{poi}}(\mathcal{D}^{\mathrm{victim}},\mathcal{F})\leq\lambda$ holds for network $\mathcal{F}$ where $\mathcal{R}(\mathcal{D}',\mathcal{F})\leq\mu$.

Table 1: The clean accuracy (Acc,%), attack success rate (ASR,%), and AUROC of Narcissus and AdvSc backdoor attacks on both post-poisoning watermarking and poisoning-concurrent watermarking with different watermarking length q under ResNet-18 and CIFAR-10.

Length/Method	Narcissus [91]		AdvSc [88]	
Acc/ASR/AUROC(↑)	Post-Poisoning	Poisoning-Concurrent	Post-Poisoning	Poisoning-Concurrent
0(Baseline)	94.69/95.04/-	94.69/95.04/-	92.80/95.53/-	92.80/95.53/-
100	94.55/93.01/0.5522	95.12/91.30/0.9294	93.34/98.23/0.8036	92.91/96.81/0.9679
300	94.38/91.34/0.8226	94.61/96.47/0.9778	92.82/96.48/0.8779	93.05/95.23/0.9955
500	94.95/93.11/0.9509	94.70/95.03/0.9968	93.18/97.43/0.9218	92.89/95.79/0.9986
1000	94.40/92.43/0.9974	94.32/92.03/0.9992	93.05/94.41/0.9809	93.38/84.39/0.9995
1500	93.90/91.05/0.9997	94.67/80.60/1.0000	93.46/90.85/0.9959	93.11/56.11/1.0000
2000	94.55/90.37/1.0000	94.89/22.46/1.0000	93.40/79.97/0.9994	92.38/30.05/1.0000
2500	94.81/93.30/1.0000	94.67/11.86/1.0000	92.78/82.89/1.0000	92.65/12.14/1.0000
3000	94.93/90.02/1.0000	94.72/ 9.75/1.0000	93.10/74.82/1.0000	93.04/ 9.97/1.0000

To quantify the performance of the poisoning algorithm, we measure how well the network $\mathcal{F}$ has learned poison features and achieves the poisoning objective by $\mathcal{R}(\mathcal{D}', \mathcal{F}) \leq \mu$ and $\mathcal{R}^{\text{poi}}(\mathcal{D}^{\text{victim}}, \mathcal{F}) \leq \lambda$ respectively. In practice, an effective poisoning method should generate (λ, μ)-effective poisoning distribution with small μ and λ. It is reasonable to assume that (λ, μ)-effective poisoning distribution $\mathcal{D}'$ can be generated by some existing heuristic algorithm. For example, previous works [69, 98] have demonstrated that victim models with low test accuracy (small λ) learn poisoning features well (small μ) under availability attacks. By Theorem 5.2, if N and d are large, $\mathcal{R}(\mathcal{D}', \mathcal{F}^*_{S'+\delta^w})$ is small enough, thus a well-trained network on the watermarked dataset $S' + \delta^w$ will result in lower $\mathcal{R}(\mathcal{D}', \mathcal{F})$, ensuring the soundness of post-poisoning watermarking through the following corollary:

Corollary 5.5 (Post-poisoning watermarking)**.** *If $\mathcal{D}'$ is a (λ, μ)-effective poisoning distribution for some $\mu > 0$, when N and d are sufficiently large, with high probability, network $\mathcal{F}$ trained on post-poisoning watermarking dataset $S' + \delta_w = \{(x'_i + \delta^w, y_i)\}_{i=1}^N$ holds that $\mathcal{R}^{\text{poi}}(\mathcal{D}^{\text{victim}}, \mathcal{F}) \leq \lambda$.*

However, when we consider poisoning-concurrent watermarking, the dimension of the poisons δ^p is restricted under $\mathcal{P} \subset [d]$. In this case, we need to further bound the risk of $\mathcal{D}'$ under the restricted poisoned dataset $S'|_{\mathcal{P}} = \{(x_i + \delta^p_i|_{\mathcal{P}}, y_i)\}_{i=1}^N$, which induces the following theorem:

Theorem 5.6 (Impact of Poisoning dimension)**.** *With probability at least $1 - \omega$ of the (unrestricted) poisoned dataset $\{(x_i + \delta^p_i, y_i)\}_{i=1}^N = S' \sim \mathcal{D}'$, it holds that*

$$\mathcal{R}(\mathcal{D}', \mathcal{F}^*_{S'|_{\mathcal{P}}}) \leq \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}(\mathcal{F}^*_{S'|_{\mathcal{P}}}(x_i + \delta^p_i|_{\mathcal{P}}), y_i) + O\left(\frac{q\epsilon_p}{\sqrt{d}}\right) + O\left(\sqrt{\frac{\log d}{N}}\right) + O\left(\sqrt{\frac{L}{N}}\right) + O\left(\sqrt{\frac{\log 1/\omega}{N}}\right).$$

In the case of poisoning-concurrent watermarking, if N is large, and $q = O\left(\sqrt{d}/\epsilon_p\right)$, then $\mathcal{R}(\mathcal{D}', \mathcal{F}^*_{S'|_{\mathcal{P}}})$ becomes sufficiently small. Thus, a well-trained network $\mathcal{F}$ on a restricted poisoned dataset $S'|_{\mathcal{P}}$ tends to have a small risk under the (unrestricted) poisoning distribution $\mathcal{D}'$. Therefore, combined with Theorem 5.2, we can directly obtain the following corollary:

Corollary 5.7. *With probability at least $1 - 3\omega$ for the restricted poisoned dataset $S'|_{\mathcal{P}} \sim \mathcal{D}'|_{\mathcal{P}}$ and the key $\zeta \in \mathbb{R}^d$ selected from a certain distribution, we can craft the watermark δ^w satisfying:*

$$\mathcal{R}(\mathcal{D}', \mathcal{F}^*_{\tilde{S}}) \leq \mathbb{E}_\eta \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}\left(\mathcal{F}^*_{S|_{\mathcal{P}}}(x_i + \delta^p_i|_{\mathcal{P}} + \eta), y_i\right)$$
$$+ O\left(\sqrt{\frac{L}{N}}\right) + O\left(\sqrt{\frac{\log d}{N}}\right) + O\left(\sqrt{\frac{\log 1/\omega}{N}}\right) + O\left(\frac{q\epsilon_p}{\sqrt{d}}\right) + O\left(\epsilon_w\sqrt{\frac{q \log 1/\omega}{d}}\right),$$

where $\tilde{S} = S|_{\mathcal{P}} + \delta^w$ is the watermarked dataset, $\eta \sim \mathcal{U}\{-\epsilon_w, \epsilon_w\}^q$ is a random vector.

After obtaining Corollary 5.7, similar to post-poisoning watermarking, we can ensure the soundness of poisoning-concurrent watermarking by the following corollary:

Corollary 5.8 (Poisoning-concurrent watermarking)**.** *If $\mathcal{D}'$ is (λ, μ)-effective for some $\mu > 0$, when N and d are sufficiently large, $q = O\left(\sqrt{d}/\epsilon_p\right)$, with high probability, the network $\mathcal{F}$ is trained on a poisoning-concurrent watermarking dataset $\{x_i + \delta^p_i \oplus \delta^w, y_i\}_{i=1}^N$ that satisfies $\mathcal{R}^{\text{poi}}\left(\mathcal{D}^{\text{victim}}, \mathcal{F}\right) \leq \lambda$.*

Comparison of two types of watermarking. For post-poisoning watermarking, the total perturbation budget will become $\epsilon_w + \epsilon_p$. To ensure the detectability, the watermarking length is expected to be $\Theta(\sqrt{d}/\epsilon_w)$, and when ensuring the utility of poisoning, no additional requirement is needed. In comparison, for poisoning-concurrent watermarking, the total perturbation budget is $\max\{\epsilon_w, \epsilon_p\}$, which is smaller than the post-poisoning case $\epsilon_w + \epsilon_p$. The watermarking length needed to guarantee the detectability becomes looser, $\Theta(1/\epsilon_w^2)$, but the poisoning utility requires a larger $O\left(\sqrt{d}/\epsilon_p\right)$. We will verify these results in Section 6.

6 Experiments

6.1 Experimental setup

Baseline methods. We evaluate our approach using two imperceptible clean-label backdoor attacks, Narcissus [91] and AdvSc [88], as well as two imperceptible clean-label availability attacks, UE [37] and AP [22]. We evaluate on CIFAR-10, CIFAR-100 [45], and Tiny-ImageNet dataset [48]. The accuracy and attack success rate are measured on various victim models including ResNet-18, ResNet-50 [33], VGG-19 [72], DenseNet121 [36], WRN34-10 [90], MobileNet v2 [68].

Implementation details. We apply both post-poisoning watermarking and poisoning-concurrent watermarking to craft watermarks for each method. The watermarking algorithms are shown in Appendix C. We evaluate watermarking lengths ranging from 0 to 3000, randomly select the watermarking dimensions while fixing the random seed to ensure reproducibility. The watermarking and poisoning budgets are set to $16/255$ for backdoor attacks, and $8/255$ for availability attacks. For victim model training, the total epochs are 200, initial learning rate is 0.5 with a cosine scheduler, the momentum and weight decay are 0.9 and 10^{-4} respectively.

6.2 Main Results

Tables 1 and 2 present the evaluation results of watermarking on backdoor and availability attacks respectively. The results show that as the watermarking length q increases, the detection performance (quantified by the AUROC score) improves consistently, achieving perfect detection (i.e., AUROC score be 1) when q is sufficiently large. This confirms the theoretical findings in Section 4, which state that when q exceeds a certain threshold ($\Theta(\sqrt{d}/\epsilon_w)$ for post-poisoning and $\Theta(1/\epsilon_w^2)$ for poisoning-concurrent), the watermarking provides provable and reliable detectability. Furthermore, poisoning-concurrent watermarking consistently outperforms post-poisoning watermarking for the same q, corroborating Remark 4.5, which indicates that $\Omega(1/\epsilon_w^2)$ is smaller than $\Omega(\sqrt{d}/\epsilon_w)$.

We also evaluate the poisoning performance under watermarking, measured by test accuracy and attack success rate (ASR) for backdoor attacks, and test accuracy for availability attacks. The results indicate that, for post-poisoning watermarking, all four attacks demonstrate strong performance compared to baseline methods without watermarking, supporting Theorem 5.2, which asserts that post-poisoning watermarking preserves poisoning when d and N are sufficiently large, regardless of q. For AdvSc, the ASR slightly decreases when q is large. This may be attributed to the reliance of AdvSc on shortcuts in the left-top $1/4$ dimension [88], implicitly reducing the effective poison dimension to $\frac{1}{4}d$ and weakening its poisoning effect. Despite this limitation, AdvSc still achieves a respectable ASR of approximately 80%. For poisoning-concurrent watermarking, the ASR for backdoor attacks and the test accuracy drop for availability attacks are more sensitive to watermarking length q. Specifically, for Narcissus and AdvSc,the ASR drops below 30% when q reaches 2000. For UE and AP, test accuracy recovers to about 90% when q reaches 2500 and 3000 respectively, rendering the poisoning ineffective. These observations align with Theorem 5.6 and Corollaries 5.7 and 5.8, which emphasize that maintaining poisoning effectiveness in poisoning-concurrent watermarking requires q to remain below $O(\sqrt{d}/\epsilon_p)$. When q exceeds this bound, the watermarking begins to dominate, significantly reducing poisoning efficacy.
For experimental results under more datasets and network structures, please refer to Appendix D.

6.3 Ablation Studies

Table 2: The clean accuracy (Acc,%) and AUROC of UE and AP availability attacks both on post-poisoning watermarking and poisoning-concurrent watermarking with different watermarking length q under ResNet-18 and CIFAR-10.

Length/Method	UE [37]		AP [22]	
Acc(↓)/AUROC(↑)	Post-Poisoning	Poisoning-Concurrent	Post-Poisoning	Poisoning-Concurrent
0(Baseline)	10.79/-	10.79/-	8.53/-	8.53/-
100	10.03/0.5844	10.35/0.8197	10.14/0.5688	10.30/0.6950
300	11.45/0.7067	9.70/0.9684	10.08/0.7573	11.77/0.7732
500	11.71/0.7810	10.02/0.9930	8.71/0.8623	15.84/0.8931
1000	11.37/0.9499	9.42/0.9991	10.58/0.9742	21.87/0.9949
1500	9.94/0.9786	10.10/0.9997	11.02/0.9916	32.46/0.9995
2000	9.06/0.9992	10.03/1.0000	10.48/0.9987	38.62/1.0000
2500	10.44/0.9996	88.78/1.0000	12.68/1.0000	36.79/1.0000
3000	9.99/1.0000	91.79/1.0000	13.52/1.0000	93.40/1.0000

Watermarking budget. We analyze the impact of watermarking budget ϵ_w on poisoning-concurrent watermarking for AdvSc attack. The results presented in Figure 1 show that as the budget increases, the detection performance (AUROC) improves. This observation verifies Theorem 4.9, which states that a larger ϵ_w allows for a smaller q to achieve effective detection. However, the poisoning performance (ASR) decreases as ϵ_w grows, confirming Corollary 5.7, which suggests that larger ϵ_w results in a higher risk $\mathcal{R}(\mathcal{D}', \mathcal{F})$, thereby degrading the poisoning power. More results are provided in Appendix D.3.

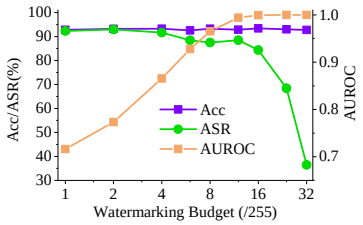

Figure 1: The Acc, ASR and AUROC of AdvSc backdoor attack on different budget ϵ_w for poisoning-concurrent watermarking with $q = 1000$.

Position of watermarking dimension. Our theoretical guarantees indicate that the position of the watermarking dimensions $\mathcal{W}$ has no significant impact. By default, we set $\mathcal{W}$ to be randomly selected from $[d]$. To validate this,we test fixed watermarking positions on the left-top (LT), left-bottom (LB), right-top (RT) and right-bottom (RB) regions of the image. We conduct experiments on post-poisoning UE watermarking with a length of 500. Results shown in Figure 2 demonstrate that the position of watermarking dimensions has minimal impact for both detection and poisoning performance.

In Appendix E, we have further discussed potential defense and watermark removal methods, including data augmentations, image regeneration attacks, differential privacy noises, and diffusion purification.

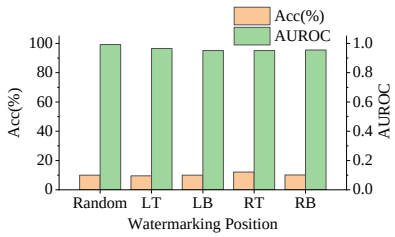

Figure 2: The Acc and AUROC of UE availability attack on different watermarking position for poisoning-concurrent watermarking with $q = 500$.

7 Conclusion

In this paper, we propose two provable and practical watermarking methods for data poisoning attacks: post-poisoning watermarking and poisoning-concurrent watermarking. We provide theoretical guarantees for the soundness of these watermarking methods, certifying their effectiveness when the watermarking length is $\Theta(\sqrt{d}/\epsilon_w)$ and $\Theta(1/\epsilon_w^2)$ for post-poisoning and poisoning-concurrent watermarking. Furthermore, we prove the soundness of the poisoning of post-poisoning and poisoning-concurrent watermarking when the length is $O(\sqrt{d}/\epsilon_p)$. We validate our theoretical findings through evaluation on several data poisoning attacks, including backdoor and availability attacks.

Limitation and future works. While our watermarking methods offer sufficient conditions for both detection and poisoning utility, the necessary conditions for these properties remain an open area for future research. Moreover, exploring more sophisticated watermarking designs that could achieve better performance and robustness in both detection and poisoning utility is a promising direction for further development.

Acknowledgment

This paper is supported by the Strategic Priority Research Program of CAS Grant XDA0480502, the Robotic AI-Scientist Platform of Chinese Academy of Sciences, NSFC Grants 12288201 and 92270001, and the CAS Project for Young Scientists in Basic Research Grant YSBR-040.

References

[1] Sahar Abdelnabi and Mario Fritz. Adversarial watermarking transformer: Towards tracing text provenance with data hiding. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 121–140. IEEE, 2021.

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[3] Yossi Adi, Carsten Baum, Moustapha Cisse, Benny Pinkas, and Joseph Keshet. Turning your weakness into a strength: Watermarking deep neural networks by backdooring. In *27th USENIX security symposium (USENIX Security 18)*, pages 1615–1631, 2018.

[4] Hojjat Aghakhani, Dongyu Meng, Yu-Xiang Wang, Christopher Kruegel, and Giovanni Vigna. Bullseye polytope: A scalable clean-label poisoning attack with improved transferability. In *2021 IEEE European symposium on security and privacy (EuroS&P)*, pages 159–178. IEEE, 2021.

[5] Ali Al-Haj. Combined dwt-dct digital image watermarking. *Journal of computer science*, 3(9):740–746, 2007.

[6] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

[7] Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines. *arXiv preprint arXiv:1206.6389*, 2012.

[8] Battista Biggio and Fabio Roli. Wild patterns: Ten years after the rise of adversarial machine learning. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pages 2154–2156, 2018.

[9] Franziska Boenisch. A systematic review on model watermarking for neural networks. *Frontiers in big Data*, 4:729663, 2021.

[10] Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[11] Bryant Chen, Wilka Carvalho, Nathalie Baracaldo, Heiko Ludwig, Benjamin Edwards, Taesung Lee, Ian Molloy, and Biplav Srivastava. Detecting backdoor attacks on deep neural networks by activation clustering. *arXiv preprint arXiv:1811.03728*, 2018.

[12] Sizhe Chen, Geng Yuan, Xinwen Cheng, Yifan Gong, Minghai Qin, Yanzhi Wang, and Xiaolin Huang. Self-ensemble protection: Training checkpoints are good data protectors. *arXiv preprint arXiv:2211.12005*, 2022.

[13] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017.

[14] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7939–7948, 2020.

[15] Miranda Christ, Sam Gunn, and Or Zamir. Undetectable watermarks for language models. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 1125–1139. PMLR, 2024.

[16] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[17] Hadi M Dolatabadi, Sarah Erfani, and Christopher Leckie. The devil's advocate: Shattering the illusion of unexploitable data using diffusion models. *arXiv preprint arXiv:2303.08500*, 2023.

[18] Yinpeng Dong, Xiao Yang, Zhijie Deng, Tianyu Pang, Zihao Xiao, Hang Su, and Jun Zhu. Black-box detection of backdoor attacks with limited information and data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16482–16491, 2021.

[19] Min Du, Ruoxi Jia, and Dawn Song. Robust anomaly detection and backdoor attack detection via differential privacy. *arXiv preprint arXiv:1911.07116*, 2019.

[20] Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International conference on machine learning*, pages 1675–1685. PMLR, 2019.

[21] Ji Feng, Qi-Zhi Cai, and Zhi-Hua Zhou. Learning to confuse: generating training time adversarial data with auto-encoder. *Advances in Neural Information Processing Systems*, 32, 2019.

[22] Liam Fowl, Micah Goldblum, Ping-yeh Chiang, Jonas Geiping, Wojciech Czaja, and Tom Goldstein. Adversarial examples make strong poisons. *Advances in Neural Information Processing Systems*, 34:30339–30351, 2021.

[23] Shaopeng Fu, Fengxiang He, Yang Liu, Li Shen, and Dacheng Tao. Robust unlearnable examples: Protecting data against adversarial learning. *arXiv preprint arXiv:2203.14533*, 2022.

[24] Leilei Gan, Jiwei Li, Tianwei Zhang, Xiaoya Li, Yuxian Meng, Fei Wu, Yi Yang, Shangwei Guo, and Chun Fan. Triggerless backdoor attack for nlp tasks with clean labels. *arXiv preprint arXiv:2111.07970*, 2021.

[25] Jonas Geiping, Liam Fowl, W Ronny Huang, Wojciech Czaja, Gavin Taylor, Michael Moeller, and Tom Goldstein. Witches' brew: Industrial scale data poisoning via gradient matching. *arXiv preprint arXiv:2009.02276*, 2020.

[26] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

[27] Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 7:47230–47244, 2019.

[28] Sam Gunn, Xuandong Zhao, and Dawn Song. An undetectable watermark for generative image models. *arXiv preprint arXiv:2410.07369*, 2024.

[29] Junfeng Guo, Yiming Li, Lixu Wang, Shu-Tao Xia, Heng Huang, Cong Liu, and Bo Li. Domain watermark: Effective and harmless dataset copyright protection is closed at hand. *Advances in Neural Information Processing Systems*, 36, 2023.

[30] Junfeng Guo and Cong Liu. Practical poisoning attacks on neural networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*, pages 142–158. Springer, 2020.

[31] Xingshuo Han, Guowen Xu, Yuan Zhou, Xuehuan Yang, Jiwei Li, and Tianwei Zhang. Physical backdoor attacks to lane detection systems in autonomous driving. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 2957–2968, 2022.

[32] Hao He, Kaiwen Zha, and Dina Katabi. Indiscriminate poisoning attacks on unsupervised contrastive learning. In *The Eleventh International Conference on Learning Representations*, 2022.

[33] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[34] Yuepeng Hu, Zhengyuan Jiang, Moyang Guo, and Neil Gong. Stable signature is unstable: removing image watermark from diffusion models. *arXiv preprint arXiv:2405.07145*, 2024.

[35] Zhengmian Hu, Lichang Chen, Xidong Wu, Yihan Wu, Hongyang Zhang, and Heng Huang. Unbiased watermark for large language models. *arXiv preprint arXiv:2310.10669*, 2023.

[36] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[37] Hanxun Huang, Xingjun Ma, Sarah Monazam Erfani, James Bailey, and Yisen Wang. Unlearnable examples: Making personal data unexploitable. In *International Conference on Learning Representations*, 2020.

[38] Jiaoyang Huang and Horng-Tzer Yau. Dynamics of deep neural networks and neural tangent hierarchy. In *International conference on machine learning*, pages 4542–4551. PMLR, 2020.

[39] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018.

[40] Xiangui Kang, Jiwu Huang, and Wenjun Zeng. Efficient general print-scanning resilient data hiding based on uniform log-polar mapping. *IEEE Transactions on Information Forensics and Security*, 5(1):1–12, 2010.

[41] John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. In *International Conference on Machine Learning*, pages 17061–17084. PMLR, 2023.

[42] John Kirchenbauer, Jonas Geiping, Yuxin Wen, Manli Shu, Khalid Saifullah, Kezhi Kong, Kasun Fernando, Aniruddha Saha, Micah Goldblum, and Tom Goldstein. On the reliability of watermarks for large language models. *arXiv preprint arXiv:2306.04634*, 2023.

[43] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pages 1885–1894. PMLR, 2017.

[44] Pang Wei Koh, Jacob Steinhardt, and Percy Liang. Stronger data poisoning attacks break data sanitization defenses. *Machine Learning*, pages 1–47, 2022.

[45] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[46] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2012.

[47] Rohith Kuditipudi, John Thickstun, Tatsunori Hashimoto, and Percy Liang. Robust distortion-free watermarks for language models. *arXiv preprint arXiv:2307.15593*, 2023.

[48] Yann Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

[49] Michel Ledoux and Michel Talagrand. *Probability in Banach Spaces: isoperimetry and processes*. Springer Science & Business Media, 2013.

[50] Yiming Li, Yang Bai, Yong Jiang, Yong Yang, Shu-Tao Xia, and Bo Li. Untargeted backdoor watermark: Towards harmless and stealthy dataset copyright protection. *Advances in Neural Information Processing Systems*, 35:13238–13250, 2022.

[51] Yiming Li, Mingyan Zhu, Xue Yang, Yong Jiang, Tao Wei, and Shu-Tao Xia. Black-box dataset ownership verification via backdoor watermarking. *IEEE Transactions on Information Forensics and Security*, 18:2318–2332, 2023.

[52] Yunfei Liu, Xingjun Ma, James Bailey, and Feng Lu. Reflection backdoor: A natural backdoor attack on deep neural networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 182–199. Springer, 2020.

[53] Yiwei Lu, Gautam Kamath, and Yaoliang Yu. Indiscriminate data poisoning attacks on neural networks. *arXiv preprint arXiv:2204.09092*, 2022.

[54] Nils Lukas, Abdulrahman Diaa, Lucas Fenaux, and Florian Kerschbaum. Leveraging optimization for adaptive attacks on image watermarks. *arXiv preprint arXiv:2309.16952*, 2023.

[55] Nan Luo, Yuanzhang Li, Yajie Wang, Shangbo Wu, Yu-an Tan, and Quanxin Zhang. Enhancing clean label backdoor attack with two-phase specific triggers. *arXiv preprint arXiv:2206.04881*, 2022.

[56] Colin McDiarmid et al. On the method of bounded differences. *Surveys in combinatorics*, 141(1):148–188, 1989.

[57] Mehryar Mohri. Foundations of machine learning, 2018.

[58] Mehryar Mohri and Andres Munoz Medina. Learning theory and algorithms for revenue optimization in second price auctions with reserve. In *International conference on machine learning*, pages 262–270. PMLR, 2014.

[59] Luis Muñoz-González, Battista Biggio, Ambra Demontis, Andrea Paudice, Vasin Wongrassamee, Emil C Lupu, and Fabio Roli. Towards poisoning of deep learning algorithms with back-gradient optimization. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pages 27–38, 2017.

[60] Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. *arXiv preprint arXiv:2205.07460*, 2022.

[61] Nikos Nikolaidis and Ioannis Pitas. Robust image watermarking in the spatial domain. *Signal processing*, 66(3):385–403, 1998.

[62] Rui Ning, Jiang Li, Chunsheng Xin, and Hongyi Wu. Invisible poison: A blackbox clean label backdoor attack to deep neural networks. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*, pages 1–10. IEEE, 2021.

[63] Lorenzo Noci, Chuning Li, Mufan Li, Bobby He, Thomas Hofmann, Chris J Maddison, and Dan Roy. The shaped transformer: Attention models in the infinite depth-and-width limit. *Advances in Neural Information Processing Systems*, 36, 2024.

[64] The University of Chicago. Glaze - protecting artists from generative ai, 2023.

[65] The University of Chicago. Nightshade: Protecting copyright, 2023.

[66] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

[67] Jie Ren, Han Xu, Yuxuan Wan, Xingjun Ma, Lichao Sun, and Jiliang Tang. Transferable unlearnable examples. In *The Eleventh International Conference on Learning Representations*, 2022.

[68] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[69] Pedro Sandoval-Segura, Vasu Singla, Liam Fowl, Jonas Geiping, Micah Goldblum, David Jacobs, and Tom Goldstein. Poisons that are learned faster are more effective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 198–205, 2022.

[70] Pedro Sandoval-Segura, Vasu Singla, Jonas Geiping, Micah Goldblum, Tom Goldstein, and David Jacobs. Autoregressive perturbations for data poisoning. *Advances in Neural Information Processing Systems*, 35:27374–27386, 2022.

[71] Ali Shafahi, W Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. Poison frogs! targeted clean-label poisoning attacks on neural networks. *Advances in neural information processing systems*, 31, 2018.

[72] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[73] Rishi Sinhal, Irshad Ahmad Ansari, and Deepak Kumar Jain. Real-time watermark reconstruction for the identification of source information based on deep neural network. *Journal of Real-Time Image Processing*, 17(6):2077–2095, 2020.

[74] Justin Sirignano and Konstantinos Spiliopoulos. Asymptotics of reinforcement learning with neural networks. *Stochastic Systems*, 12(1):2–29, 2022.

[75] Hossein Souri, Liam Fowl, Rama Chellappa, Micah Goldblum, and Tom Goldstein. Sleeper agent: Scalable hidden trigger backdoors for neural networks trained from scratch. *Advances in Neural Information Processing Systems*, 35:19165–19178, 2022.

[76] Lue Tao, Lei Feng, Jinfeng Yi, Sheng-Jun Huang, and Songcan Chen. Better safe than sorry: Preventing delusive adversaries with adversarial training. *Advances in Neural Information Processing Systems*, 34:16209–16225, 2021.

[77] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[78] Linh Duy Tran, Son Minh Nguyen, and Masayuki Arai. Gan-based noise model for denoising real images. In *Proceedings of the Asian Conference on Computer Vision*, 2020.

[79] Alexander Turner, Dimitris Tsipras, and Aleksander Madry. Clean-label backdoor attacks. 2018.

[80] Alexander Turner, Dimitris Tsipras, and Aleksander Madry. Label-consistent backdoor attacks. *arXiv preprint arXiv:1912.02771*, 2019.

[81] Vladimir Vapnik. Statistical learning theory. *John Wiley & Sons google schola*, 2:831–842, 1998.

[82] Tianhao Wang and Florian Kerschbaum. Riga: Covert and robust white-box watermarking of deep neural networks. In *Proceedings of the Web Conference 2021*, pages 993–1004, 2021.

[83] Yihan Wang, Yifan Zhu, and Xiao-Shan Gao. Efficient availability attacks against supervised and contrastive learning simultaneously. *Advances in Neural Information Processing Systems*, 37:72872–72900, 2024.

[84] Ming Wen, Yixi Xu, Yunling Zheng, Zhouwang Yang, and Xiao Wang. Sparse deep neural networks using l 1,∞-weight normalization. *Statistica Sinica*, 31(3):1397–1414, 2021.

[85] Yuxin Wen, John Kirchenbauer, Jonas Geiping, and Tom Goldstein. Tree-rings watermarks: Invisible fingerprints for diffusion images. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[86] Da Yu, Huishuai Zhang, Wei Chen, Jian Yin, and Tie-Yan Liu. Availability attacks create shortcuts. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2367–2376, 2022.

[87] Jiahui Yu and Konstantinos Spiliopoulos. Normalization effects on deep neural networks. *arXiv preprint arXiv:2209.01018*, 2022.

[88] Lijia Yu, Shuang Liu, Yibo Miao, Xiao-Shan Gao, and Lijun Zhang. Generalization bound and new algorithm for clean-label backdoor attack. *arXiv preprint arXiv:2406.00588*, 2024.

[89] Yi Yu, Qichen Zheng, Siyuan Yang, Wenhan Yang, Jun Liu, Shijian Lu, Yap-Peng Tan, Kwok-Yan Lam, and Alex Kot. Unlearnable examples detection via iterative filtering. In *International Conference on Artificial Neural Networks*, pages 241–256. Springer, 2024.

[90] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[91] Yi Zeng, Minzhou Pan, Hoang Anh Just, Lingjuan Lyu, Meikang Qiu, and Ruoxi Jia. Narcissus: A practical clean-label backdoor attack with limited information. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 771–785, 2023.

[92] Jiaming Zhang, Xingjun Ma, Qi Yi, Jitao Sang, Yu-Gang Jiang, Yaowei Wang, and Changsheng Xu. Unlearnable clusters: Towards label-agnostic unlearnable examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3984–3993, 2023.

[93] Xuandong Zhao, Prabhanjan Ananth, Lei Li, and Yu-Xiang Wang. Provable robust watermarking for ai-generated text. *arXiv preprint arXiv:2306.17439*, 2023.

[94] Xuandong Zhao, Kexun Zhang, Zihao Su, Saastha Vasan, Ilya Grishchenko, Christopher Kruegel, Giovanni Vigna, Yu-Xiang Wang, and Lei Li. Invisible image watermarks are provably removable using generative ai. *arXiv preprint arXiv:2306.01953*, 2023.

[95] Haoti Zhong, Cong Liao, Anna Cinzia Squicciarini, Sencun Zhu, and David Miller. Backdoor embedding in convolutional neural network models via invisible perturbation. In *Proceedings of the Tenth ACM Conference on Data and Application Security and Privacy*, pages 97–108, 2020.

[96] J Zhu. Hidden: hiding data with deep networks. *arXiv preprint arXiv:1807.09937*, 2018.

[97] Yifan Zhu, Yibo Miao, Yinpeng Dong, and Xiao-Shan Gao. Toward availability attacks in 3d point clouds. In *Proceedings of the 41st International Conference on Machine Learning*, pages 62510–62530, 2024.

[98] Yifan Zhu, Lijia Yu, and Xiao-Shan Gao. Detection and defense of unlearnable examples. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17211–17219, 2024.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: Our paper supports the claims made in the abstract and introduction.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We have discussed limitations in Section 7.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [Yes]

Justification: We have provided the full set of assumptions in every theorem and made a complete proof in Appendix B.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: We have provided reproductive details in Section 6.1 and given detailed codes in the supplemental material.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: We have provided our codes in the supplemental material.

Guidelines:

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

6. **Experimental setting/details**

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

Answer: [Yes]

Justification: We have provided experimental details in Section 6.1.

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [No]

Justification: Our theoretical findings can be validated well by the trend of Accuracy, Attack Success Rate and AUROC under different watermarking length, even without error bars. Due to insufficiency of computational resource (We conduct all our experiments in a single NVIDIA A800 80GB PCIe GPU), it is expensive to reproduce all these data poisoning attacks. Therefore, we do not report error bars.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: We have provided them in Appendix D.5.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: ur paper conforms with the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: We have discussed them in Appendix H.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [NA]

    Justification: Our paper poses no such risks.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: We use open-source dataset and models in our paper, and have cited the original paper of these dataset and models.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [Yes]

    Justification: Our new assets are well documented in the supplemental material.

    Guidelines:

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

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

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

    Answer: [NA]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

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

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

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

    Answer: [NA]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.

Appendix

A Symbol Table

Notation	Description
d	The dimension of data
q	The dimension of watermarking
N	The number of samples in a dataset
$\mathcal{P}$	The indices of poisoning dimension
$\mathcal{W}$	The indices of watermarking dimension
ϵ_p	The perturbation budget of a poisoning attack
ϵ_w	The perturbation budget of a watermark
δ^p	A data poisoning attack
δ^w	A watermark
δ_x	A sample-wise perturbation on data x
ζ	A key
Ξ	A key distribution
S	A clean dataset
S'	A perturbed dataset
$\mathcal{D}$	A clean data distribution
$\mathcal{D}'$	A data distribution under some perturbations
L	The layer of a neural network
$\mathcal{L}$	A loss function
$\mathcal{F}$	A model (neural network)
$\mathcal{R}$	A generalization risk
$\mathcal{R}^{\text{poi}}$	A poisoning objective risk
ω	A probability

B Proofs

B.1 Proofs of Theorems in Section 4.1

Lemma B.1 (McDiarmid's Inequality [56]). *Let $X_1, X_2, \cdots, X_n$ be independent random variables on $\mathcal{X}_1, \mathcal{X}_2, \cdots, \mathcal{X}_n$ and $f : \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_n \to \mathbb{R}$ be a multivariate function. If there exist positive constants $c_1, c_2, \cdots, c_n$, such that for all $(x_1, x_2, \cdots, x_n) \in \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_n$ and $i \in [n]$, it has*

$$\sup_{x_i' \in \mathcal{X}_i} |f(x_1, \cdots, x_{i-1}, x_i', x_{i+1}, \cdots, x_n) - f(x_1, \cdots, x_{i-1}, x_i, x_{i+1}, \cdots, x_n)| \leq c_i,$$

then for any $\epsilon > 0$, the following inequalities hold

$$\mathbb{P}\left(f(X_1, X_2, \cdots, X_n) - \mathbb{E}\left[f(X_1, X_2, \cdots, X_n)\right] \geq \epsilon\right) \leq e^{-\frac{2\epsilon^2}{\sum_{i=1}^n c_i^2}},$$
$$\mathbb{P}\left(f(X_1, X_2, \cdots, X_n) - \mathbb{E}\left[f(X_1, X_2, \cdots, X_n)\right] \leq -\epsilon\right) \leq e^{-\frac{2\epsilon^2}{\sum_{i=1}^n c_i^2}}.$$

Definition B.2 (Random identical key). The random identical key means that for each entry, the probability of its value being 1 or -1 is $1/2$, i.e., $\zeta^i = \mathcal{U}\{-1, +1\}$ for all entries i of key ζ.

Theorem B.3 (Theorem 4.1, restated). *For any data point x sampled from $\mathcal{D}_{\mathcal{X}}$ and their corresponding poison be δ_x^p, there exists a distribution Ξ defined in $\mathbb{R}^d$ such that we can sample the key $\zeta \sim \Xi$ satisfied that for any $\omega \in (0, 1)$, there are:*

(1): $\mathbb{P}_{x \sim \mathcal{D}_{\mathcal{X}}, \zeta \sim \Xi}\left(\zeta^T x < \sqrt{\frac{d}{2}\log\frac{1}{\omega}}\right) > 1 - w$; (2): we can craft the watermark δ_x^w based on ζ such that $\mathbb{P}_{x \sim \mathcal{D}_{\mathcal{X}}, \zeta \sim \Xi}\left(\zeta^T(x + \delta_x) > q\epsilon_w - \sqrt{\frac{d}{2}\log\frac{1}{\omega}}\right) > 1 - w$. Hence, when $q > \frac{1}{\epsilon_w}\sqrt{2d\log\frac{1}{\omega}}$, it holds that $\mathbb{P}_{x_1, x_2 \sim \mathcal{D}_{\mathcal{X}}, \zeta \sim \Xi}\left(\zeta^T(x_1 + \delta_1) > \zeta^T x_2\right) > 1 - 2\omega$.

Proof of Theorem 4.1. (1): Denote the distribution Ξ be the distribution of a random identical key, i.e., $\Xi = \mathcal{U}\{-1, +1\}^d$, it has

$$\mathbb{E}_\zeta[\zeta^T x] = 0$$

for all $x \in \mathcal{D}$. Furthermore, as x lies in $[0,1]$, it always holds that

$$|\zeta^i x^i - \zeta^i \tilde{x}^i| \leq |\zeta^i| = 1$$

for all i, ζ and $x, \tilde{x} \in \mathcal{D}$.

Therefore, by McDiarmid's inequality, for any $\alpha > 0$, it has

$$\mathbb{P}_x \left[\mathbb{P}_\zeta [\zeta^T x \geq \alpha] \leq e^{-\frac{2\alpha^2}{d}} \right] = 1,$$

which concludes that

$$\mathbb{P}_{x,\zeta} [\zeta^T x \geq \alpha] \leq e^{-\frac{2\alpha^2}{d}}.$$

Therefore, let $\omega = e^{-\frac{2\alpha^2}{d}}$, it has $\alpha = \sqrt{\frac{d}{2} \log \frac{1}{\omega}}$, which validates (1).

(2): For any key ζ, we craft the watermark δ_x^w as $(\epsilon_w \cdot \zeta^{d_i})_{i=1}^q$. We can conclude that

$$\mathbb{E}_\zeta [\zeta^T (x + \delta_x)] = \mathbb{E}_\zeta [\zeta^T x] + \mathbb{E}_\zeta [\zeta^T \delta_x^p] + \mathbb{E}_\zeta [\zeta^T \delta_x^w].$$

Because x and δ_x^p are independent from ζ, it holds that

$$\mathbb{E}_\zeta [\zeta^T x] = \mathbb{E}_\zeta [\zeta^T \delta_x^p] = 0.$$

Therefore, we have

$$\mathbb{E}_\zeta [\zeta^T (x + \delta_x)] = \mathbb{E}_\zeta [\zeta^T \delta_x^w] = q\epsilon_w.$$

Similar to (1), by McDiarmid's inequality, for any $\beta > 0$, it has

$$\mathbb{P}_{x,\zeta} [\zeta^T (x + \delta_x) - q\epsilon_w \leq -\beta] \leq e^{-\frac{2\beta^2}{d}}.$$

Therefore, let $\omega = e^{-\frac{2\beta^2}{d}}$, it has $\beta = \sqrt{\frac{d}{2} \log \frac{1}{\omega}}$, which induces that

$$\mathbb{P}_{x \sim \mathcal{D}_\mathcal{X}, \zeta} \left(\zeta^T (x + \delta_x) > q\epsilon_w - \sqrt{\frac{d}{2} \log \frac{1}{\omega}} \right) > 1 - w.$$

When $q > \frac{1}{\epsilon_w} \sqrt{2d \log \frac{1}{\omega}}$, it holds that

$$\sqrt{\frac{d}{2} \log \frac{1}{\omega}} < q\epsilon_w - \sqrt{\frac{d}{2} \log \frac{1}{\omega}}.$$

Hence by the union bound, it has

$$\begin{aligned}
&\mathbb{P}_{x_1, x_2 \sim \mathcal{D}_\mathcal{X}, \zeta} \left[\zeta^T (x_1 + \delta_1) > \zeta^T x_2 \right] = 1 - \mathbb{P}_{x_1, x_2 \sim \mathcal{D}_\mathcal{X}, \zeta} \left[\zeta^T (x_1 + \delta_1) \leq \zeta^T x_2 \right] \\
&\geq 1 - \mathbb{P}_{x_1, x_2 \sim \mathcal{D}_\mathcal{X}, \zeta} \left[\zeta^T (x_1 + \delta_1) \leq q\epsilon_w - \sqrt{\frac{d}{2} \log \frac{1}{\omega}} \right] - \mathbb{P}_{x_1, x_2 \sim \mathcal{D}_\mathcal{X}, \zeta} \left[\zeta^T x_2 \geq \sqrt{\frac{d}{2} \log \frac{1}{\omega}} \right] \\
&\geq 1 - 2\omega. \qquad (1)
\end{aligned}$$

□

Theorem B.4 (Theorem 4.3, restated)**.** *For any $x \sim \mathcal{D}_\mathcal{X}$, there exists a distribution $\Xi \in \mathbb{R}^d$ such that we can sample the key $\zeta \sim \Xi$ satisfied that for any $\omega \in (0,1)$:*

(1): $\mathbb{P}_{x \sim \mathcal{D}_\mathcal{X}, \zeta \sim \Xi} \left(\zeta^T x < \sqrt{\frac{q}{2} \log \frac{1}{\omega}} \right) > 1 - w$; (2): we can craft the watermark δ_x^w and poison δ_x^p such that $\mathbb{P}_{x \sim \mathcal{D}_\mathcal{X}, \zeta \sim \Xi} \left(\zeta^T (x + \delta_x) > q\epsilon_w - \sqrt{\frac{q}{2} \log \frac{1}{\omega}} \right) > 1 - w$. Hence, when $q > \frac{2}{\epsilon_w^2} \log \frac{1}{\omega}$, it holds that $\mathbb{P}_{x_1, x_2 \sim \mathcal{D}_\mathcal{X}, \zeta \sim \Xi} \left(\zeta^T (x_1 + \delta_1) > \zeta^T x_2 \right) > 1 - 2\omega$.

Proof of Theorem 4.3. For poisoning-concurrent watermarking, denote the poisoning dimension be $\mathcal{P}$ and the watermarking dimension be $\mathcal{W}$, where $[d] = \mathcal{P} \cup \mathcal{W}$ and $|\mathcal{W}| = q$.

We sample the key $\zeta \in \mathbb{R}^d$ from a certain distribution Ξ, such that $\zeta^i = \mathcal{U}\{-1, +1\}, i \in \mathcal{W}$ and $\zeta^i = 0, i \in \mathcal{P}$.

Therefore, by McDiarmid's inequality, for any $\alpha > 0$, it has

$$\mathbb{P}_{x,\zeta}[\zeta^T x \geq \alpha] \leq e^{-\frac{2\alpha^2}{q}}.$$

Let $\omega = e^{-\frac{2\alpha^2}{q}}$, it has $\alpha = \sqrt{\frac{q}{2}\log\frac{1}{\omega}}$, which validates condition (1).

We craft the watermark δ_x^w as $(\epsilon_w \cdot \zeta^i)_{i=1}^d$. Similar to the proof of Theorem 4.1, we can conclude that

$$\mathbb{E}_\zeta[\zeta^T(x+\delta_x)] = \mathbb{E}_\zeta[\zeta^T\delta_x^w] = q\epsilon_w.$$

Let $\omega = e^{-\frac{2\beta^2}{q}}$, it has $\beta = \sqrt{\frac{q}{2}\log\frac{1}{\omega}}$, which induces that

$$\mathbb{P}_{x\sim\mathcal{D}_\mathcal{X},\zeta}\left(\zeta^T(x+\delta_x) > q\epsilon_w - \sqrt{\frac{q}{2}\log\frac{1}{\omega}}\right) > 1-w.$$

When $q > \frac{2}{\epsilon_w^2}\log\frac{1}{\omega}$, it holds that

$$\sqrt{\frac{q}{2}\log\frac{1}{\omega}} < q\epsilon_w - \sqrt{\frac{q}{2}\log\frac{1}{\omega}}.$$

Hence by union bound, it has

$$\mathbb{P}_{x_1,x_2\sim\mathcal{D}_\mathcal{X},\zeta}\left[\zeta^T(x_1+\delta_1) > \zeta^T x_2\right] \geq 1-2\omega.$$

□

B.2 Proofs of Theorems in Section 4.2

Theorem B.5 (Proposition 4.7, restated)**.** *For the dataset $S_\mathcal{X}$, when $q > \frac{2+\epsilon_p}{\epsilon_w}\sqrt{\frac{d}{2}\log\frac{2N}{\omega}}$, we can sample the key $\zeta \in \mathbb{R}^d$ from a certain distribution such that, with a probability of at least $1-\omega$, there exists the watermark δ^w such that $\zeta^T(x_j+\delta_j) > \zeta^T x_i, \forall i,j \in [N]$.*

Proof of Proposition 4.7. For the random identical key $\zeta \in \mathbb{R}^d$, and $x_i \in [0,1]^d$, it holds that

$$|\zeta^j x_i^j - \zeta^j \tilde{x_i}^j| \leq |\zeta^j| = 1$$

for every $\tilde{x_i} \neq x_i$.

By McDiarmid's inequality, for any $\alpha > 0$. it has

$$\mathbb{P}_\zeta\left[\zeta^T x_i \geq \alpha\right] \leq e^{-\frac{2\alpha^2}{d}}.$$

Furthermore, as $\|\delta_i^p\| \leq \epsilon_p$, it has

$$|\zeta^j(\delta_i^p)^j - \zeta^j(\tilde{\delta}_i^p)^j| \leq |\zeta^j| \cdot \epsilon_p = \epsilon_p$$

for every $\tilde{\delta}_i^p \neq \delta_i^p$.

By McDiarmid's inequality, for any $\beta > 0$. it has

$$\mathbb{P}_\zeta\left[\zeta^T\delta_i^p \geq \beta\right] \leq e^{-\frac{2\beta^2}{d\epsilon_p^2}}.$$

By the union bound, it holds that

$$\mathbb{P}\left[\cup_{i=1}^N\{|\zeta^T x_i| \geq \alpha\}\right] \leq \sum_{i=1}^N \mathbb{P}\left[|\zeta^T x_i| \geq \alpha\right] \leq Ne^{-\frac{2\alpha^2}{d}},$$

$$\mathbb{P}\left[\cup_{i=1}^N\{|\zeta^T\delta_i^p| \geq \beta\}\right] \leq \sum_{i=1}^N \mathbb{P}\left[|\zeta^T x_i| \geq \beta\right] \leq Ne^{-\frac{2\beta^2}{d\epsilon_p^2}}.$$

We now craft the watermark δ^w such that

$$\delta^w = \epsilon_w \cdot \zeta|_{\mathcal{W}}.$$

It has

$$\zeta^T \delta^w = q\epsilon_w.$$

Therefore, let $\beta = \epsilon_p \alpha$, it holds that

$$\begin{aligned}
\mathbb{P}\left[\cap_{i=1}^N \{\zeta^T(x_i + \delta^i) > q\epsilon_w - (1+\epsilon_p)\alpha\}\right] &= \mathbb{P}\left[\cap_{i=1}^N \{\zeta^T(x_i + \delta_p^i + \delta^w) > q\epsilon_w - (1+\epsilon_p)\alpha\}\right] \\
&= \mathbb{P}\left[\cap_{i=1}^N \{\zeta^T(x_i + \delta_p^i) > -(1+\epsilon_p)\alpha\}\right] \\
&\geq \mathbb{P}\left[\left\{\cap_{i=1}^N \{\zeta^T \delta_p^i > -\epsilon_p \alpha\}\right\} \cap \left\{\cap_{i=1}^N \{\zeta^T x^i > -\alpha\}\right\}\right] \\
&\geq 1 - \mathbb{P}\left[\cup_{i=1}^N \{|\zeta^T x_i| \geq \alpha\}\right] - \mathbb{P}\left[\cup_{i=1}^N \{|\zeta^T \delta_i^p| \geq \epsilon_p \alpha\}\right] \\
&\geq 1 - 2Ne^{-\frac{2\alpha^2}{d}}.
\end{aligned}$$

Therefore, when

$$q\epsilon_w - (1+\epsilon_p)\alpha > \alpha,$$

it has

$$\zeta^T(x_j + \delta_j) > \zeta^T x_i, \forall i, j \in [N]$$

happens with probability at least $1 - 2Ne^{-\frac{2\alpha^2}{d}}$.

Let $\omega = 2Ne^{-\frac{2\alpha^2}{d}}$, the condition will be

$$q > \frac{2+\epsilon_p}{\epsilon_w}\sqrt{\frac{d}{2}\log\frac{2N}{\omega}}.$$

□

Theorem B.6 (Theorem 4.9, restated)**.** *For the dataset $S_{\mathcal{X}} = \{x_1, x_2, \cdots, x_N\}$, x_i and the poison δ_p^i are i.i.d. sampled from $\mathcal{D}_{\mathcal{X}}$ and $\mathcal{D}_{\mathcal{P}}$ respectively. For any $w \in (0, 1/2)$ and $q > \frac{2}{\epsilon_w}\sqrt{2d\log\frac{1}{\omega}}$, we can sample the key $\zeta \in \mathbb{R}^d$ from a certain distribution such that, with probability at least $1 - 2\exp\left(\frac{-N\left(\omega - e^{-q^2\epsilon_w^2/8d}\right)^2}{\omega + e^{-q^2\epsilon_w^2/8d}}\right)$, it is possible to craft the watermark δ^w, such that $\zeta^T(x_i + \delta_i) > \frac{q\epsilon_w}{2}, \zeta^T x_i < \frac{q\epsilon_w}{4}$ holds for at least $(1-2\omega)N$ samples.*

Proof of Theorem 4.9. Denote the failure cases of Proposition 4.7 be

$$F_i(\alpha) = \mathbb{I}\{\zeta^T x_i \geq \alpha\}, F_i'(\alpha) = \mathbb{I}\{\zeta^T \delta_i^p \geq \epsilon_p \alpha\}$$

and

$$F(\alpha) = \sum_{i=1}^N F_i(\alpha), F'(\alpha) = \sum_{i=1}^N F_i'(\alpha)$$

Due to the i.i.d property of x_i and δ_i^p, $F_i(\alpha)$ and $F_i'(\alpha)$ are also i.i.d. for $i = \{1, 2, \cdots, N\}$. $\zeta^T(x_i + \delta_i) > \zeta^T x_i$ holds as long as both $F_i(\alpha) = 0$ and $F_i'(\alpha) = 0$ for a certain constant $\alpha > 0$.

By McDiarmid's inequality,

$$\mathbb{P}[F_i(\alpha) = 1] \leq e^{-\frac{2\alpha^2}{d}}, \mathbb{P}[F_i'(\alpha) = 1] \leq e^{-\frac{2\alpha^2}{d}}.$$

Therefore, assume that

$$\mathbb{P}[F_i(\alpha) = 1] = p_{i,\alpha}, \mathbb{P}[F_i'(\alpha) = 1] = p_{i',\alpha}.$$

$F_i(\alpha)$ and $F_i'(\alpha)$ obey the Bernoulli distribution $\mathcal{B}(p_{i,\alpha})$ and $\mathcal{B}(p_{i',\alpha})$ respectively.

Denote $\bar{F}_i(\alpha)$ obeying the Bernoulli distribution $\mathcal{B}(e^{-\frac{2\alpha^2}{d}})$, and

$$\bar{F}(\alpha) = \sum_{i=1}^{N} \bar{F}_i(\alpha).$$

By the Chernoff bound, it holds that

$$\mathbb{P}\left[\bar{F}(\alpha) \geq (1+\delta)Ne^{-\frac{2\alpha^2}{d}}\right] \leq \exp\left(\frac{-\delta^2 N}{2+\delta}e^{-\frac{2\alpha^2}{d}}\right)$$

for any $\delta > 0$.

As it always has $\bar{F}_i(\alpha) \leq \bar{F}_i(\alpha), \bar{F}_i'(\alpha) \leq \bar{F}_i(\alpha)$, it holds that

$$\mathbb{P}\left[F(\alpha) \geq (1+\delta)Ne^{-\frac{2\alpha^2}{d}}\right] \leq \exp\left(\frac{-\delta^2 N}{2+\delta}e^{-\frac{2\alpha^2}{d}}\right),$$

$$\mathbb{P}\left[F'(\alpha) \geq (1+\delta)Ne^{-\frac{2\alpha^2}{d}}\right] \leq \exp\left(\frac{-\delta^2 N}{2+\delta}e^{-\frac{2\alpha^2}{d}}\right).$$

Let $\omega = (1+\delta)e^{-\frac{2\alpha^2}{d}}$. It has

$$\mathbb{P}\left[F(\alpha) \geq \omega N\right] \leq \exp\left(\frac{-N(\omega - e^{-2\alpha^2/d})^2}{\omega + e^{-2\alpha^2/d}}\right),$$

$$\mathbb{P}\left[F'(\alpha) \geq \omega N\right] \leq \exp\left(\frac{-N(\omega - e^{-2\alpha^2/d})^2}{\omega + e^{-2\alpha^2/d}}\right).$$

Therefore, the probability of a bad case is at most

$$\mathbb{P}[F(\alpha) \geq \omega N] + \mathbb{P}[F'(\alpha) \geq \omega N]$$

with $2\omega N$ samples. To achieve the non-vacuous gap of watermarking between poisoned data $x_i + \delta_i$ and benign data x_j, we can set

$$\alpha = \frac{q\epsilon_w}{4}.$$

In this case, if both $F_i(\alpha)$ and $F_i'(\alpha) = 0$, i.e., sample x_i is not a bad case, it holds that

$$\zeta^T x_i < \alpha = \frac{q\epsilon_w}{4}, \zeta^T(x_i + \delta_i) = q\epsilon_w - (1+\epsilon_p)\alpha > \frac{q\epsilon_w}{2}.$$

Hence, for at least $(1-2\omega)N$ samples, with probability at least

$$1 - 2\exp\left(\frac{-N(\omega - e^{-2\alpha^2/d})^2}{\omega + e^{-2\alpha^2/d}}\right) = 1 - 2\exp\left(\frac{-N(\omega - e^{-q^2\epsilon_w^2/8d})^2}{\omega + e^{-q^2\epsilon_w^2/8d}}\right),$$

the property holds.

Furthermore, as we set

$$\omega = (1+\delta)e^{-\frac{2\alpha^2}{d}}$$

and $\delta > 0$. This condition is valid as long as

$$q > \frac{2}{\epsilon_w}\sqrt{2d\log\frac{1}{\omega}}$$

□

Theorem B.7 (Proposition 4.11, restated)**.** *For the dataset $S_{\mathcal{X}}$, when $q > \frac{4}{\epsilon_w^2}\log\frac{N}{\omega}$, it is possible to sample the key $\zeta \in \mathbb{R}^d$ from a certain distribution such that, with probability at least $1-\omega$, we can craft a watermark δ^w and poison δ^p such that $\zeta^T(x_j+\delta_j) > \zeta^T x_i, \forall i, j \in [N]$.*

Proof of Proposition 4.11. We sample the key $\zeta \in \mathbb{R}^d$, such that

$$\zeta^i = \mathcal{U}\{-1, +1\}, i \in \mathcal{W}$$

and

$$\zeta^i = 0, i \in \mathcal{P}.$$

By McDiarmid's inequality, for any $\alpha > 0$, it holds that

$$\mathbb{P}_\zeta \left[\zeta^T x_i \geq \alpha\right] \leq e^{-\frac{2\alpha^2}{q}}.$$

By the union bound, it holds that

$$\mathbb{P}\left[\cup_{i=1}^N \{|\zeta^T x_i| \geq \alpha\}\right] \leq \sum_{i=1}^N \mathbb{P}\left[|\zeta^T x_i| \geq \alpha\right] \leq N e^{-\frac{2\alpha^2}{q}}.$$

We now craft the watermark δ^w such that

$$\delta^w = \epsilon_w \cdot \zeta.$$

It holds that

$$\zeta^T \delta^w = q\epsilon_w.$$

It holds that

$$\begin{aligned}
\mathbb{P}\left[\cap_{i=1}^N \{\zeta^T(x_i + \delta^i) > q\epsilon_w - \alpha\}\right] &= \mathbb{P}\left[\cap_{i=1}^N \{\zeta^T x_i > q\epsilon_w - \alpha\}\right] \\
&= \mathbb{P}\left[\cap_{i=1}^N \{\zeta^T x_i > -\alpha\}\right] \\
&\geq 1 - \mathbb{P}\left[\cup_{i=1}^N \{|\zeta^T x_i| \geq \alpha\}\right] \\
&\geq 1 - N e^{-\frac{2\alpha^2}{q}}.
\end{aligned}$$

Therefore, when

$$q\epsilon_w > 2\alpha,$$

it holds that

$$\zeta^T(x_j + \delta_j) > \zeta^T x_i, \forall i, j \in [N]$$

happens with probability at least $1 - N e^{-\frac{2\alpha^2}{q}}$.

Let $\omega = N e^{-\frac{2\alpha^2}{q}}$, it holds that

$$\alpha = \sqrt{\frac{q}{2} \log \frac{N}{\omega}}.$$

Then the condition will be

$$q > \frac{4}{\epsilon_w^2} \log \frac{N}{\omega}.$$

□

Theorem B.8 (Theorem 4.13, restated)**.** *For the dataset $S_\mathcal{X} = \{x_1, x_2, \cdots, x_N\}$, where x_i is i.i.d. sampled from $\mathcal{D}_\mathcal{X}$. For any $\omega \in (0, 1)$ and $q > \frac{9}{2\epsilon_w^2} \log \frac{1}{\omega}$, it is possible to sample the key $\zeta \in \mathbb{R}^d$ from a certain distribution such that, with probability at least $1 - \exp\left(\frac{-N\left(\omega - e^{-2q\epsilon^2/9}\right)^2}{\omega + e^{-2q\epsilon^2/9}}\right)$, we can craft the watermark and the poison satisfies $\zeta^T(x_i + \delta_i) > \frac{2q\epsilon_w}{3}, \zeta^T x_i < \frac{q\epsilon_w}{3}$ holds for at least $(1 - \omega)N$ samples.*

Proof of Theorem 4.13. We sample the key $\zeta \in \mathbb{R}^d$, such that

$$\zeta^i = \mathcal{U}\{-1, +1\}, i \in \mathcal{W}$$

and

$$\zeta^i = 0, i \in \mathcal{P}.$$

Similar to the proof of 4.9, denote the failure case

$$F_i(\alpha) = \mathbb{I}\{\zeta^T x_i > \alpha\}$$

and

$$F(\alpha) = \sum_{i=1}^{N} F_i(\alpha).$$

By McDiarmid's inequality,

$$\mathbb{P}[F_i(\alpha) = 1] \leq e^{-\frac{2\alpha^2}{q}}.$$

Denote $\bar{F}_i(\alpha)$ obeys the Bernoulli distribution $\mathcal{B}\left(e^{-\frac{2\alpha^2}{q}}\right)$, and

$$\bar{F}(\alpha) = \sum_{i=1}^{N} \bar{F}_i(\alpha).$$

By Chernoff bound, it holds that

$$\mathbb{P}\left[\bar{F}(\alpha) \geq (1+\delta)Ne^{-\frac{2\alpha^2}{q}}\right] \leq \exp\left(\frac{-\delta^2 N}{(2+\delta)}e^{-\frac{2\alpha^2}{q}}\right)$$

for any $\delta > 0$.

As it always has $\bar{F}_i(\alpha) \leq \bar{F}_i(\alpha)$, it holds that

$$\mathbb{P}\left[F(\alpha) \geq (1+\delta)Ne^{-\frac{2\alpha^2}{q}}\right] \leq \exp\left(\frac{-\delta^2 N}{(2+\delta)}e^{-\frac{2\alpha^2}{q}}\right).$$

Let $\omega = (1+\delta)e^{-\frac{2\alpha^2}{q}}$. It has

$$\mathbb{P}\left[F(\alpha) \geq \omega N\right] \leq \exp\left(\frac{-N(\omega - e^{-2\alpha^2/q})^2}{\omega + e^{-2\alpha^2/q}}\right),$$

Therefore, the probability of a bad case is at most

$$\mathbb{P}[F(\alpha) \geq \omega N]$$

with ωN samples. To achieve the non-vacuous gap of watermarking between poisoned data $x_i + \delta_i$ and benign data x_j, we can set

$$\alpha = \frac{q\epsilon_w}{3}.$$

In this case, if both $F_i(\alpha) = 0$, i.e., sample x_i is not a bad case, it holds that

$$\zeta^T x_i \leq \alpha = \frac{q\epsilon_w}{3}, \zeta^T(x_i + \delta_i) = q\epsilon_w - \alpha \geq \frac{2q\epsilon_w}{3}.$$

Hence for at least $(1-\omega)N$ samples, with probability at least

$$1 - \exp\left(\frac{-N(\omega - e^{-2\alpha^2/q})^2}{\omega + e^{-2\alpha^2/q}}\right) = 1 - \exp\left(\frac{-N(\omega - e^{-2q\epsilon^2/9})^2}{\omega + e^{-2q\epsilon^2/9}}\right),$$

the property holds.

Furthermore, as we set

$$\omega = (1+\delta)e^{-\frac{2\alpha^2}{q}}$$

and $\delta > 0$. This condition is valid as long as

$$q > \frac{9}{2\epsilon_w^2}\log\frac{1}{\omega}$$

□

Theorem B.9 (Theorem 4.15, restated). *For the dataset $S_{\mathcal{X}} = \{x_1, x_2, \cdots, x_N\}$, x_i and the poison δ_p^i are i.i.d. sampled from $\mathcal{D}_{\mathcal{X}}$ and $\mathcal{D}_{\mathcal{P}}$ respectively. Considering a universal watermark δ^w, with probability at least $1 - 2\mu$ for the sampled data and poisons, if there exists a key ζ that satisfies $\zeta^T(x_i + \delta_i) > C_1, \zeta^T x_i < C_2$, for at least $(1-\omega)N$ samples x_i, then it holds that*

$$\begin{aligned}\mathbb{P}_{x,\tilde{x}\sim\mathcal{D}_{\mathcal{X}},\delta^p\sim\Delta}\left(\left\{\zeta^T(x+\delta^p+\delta^w) > C_1, -\zeta^T\tilde{x} < C_2\right\}\right)\\ > 1 - 2\omega - 2\sqrt{\frac{d}{N}(\log(\frac{2N}{d}) + 1) - \frac{1}{N}\log(\frac{\mu}{4})}.\end{aligned}$$

Lemma B.10 (VC bound [81]). *Let $S = \{(x_i, y_i)\}_{i=1}^N$ be the training dataset, $(x_i, y_i) \sim \mathcal{D}$, where $\mathcal{D}$ is the data distribution. Then with probability at least $1 - \delta$, it holds that*

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}\mathcal{L}(f(x), y) \leq \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}(f(x_i), y_i) + \sqrt{\frac{VC(f)}{N}\left(\log\left(\frac{2N}{VC(f)}\right) + 1\right) - \frac{1}{N}\log\left(\frac{\delta}{4}\right)},$$

where $VC(f)$ is the VC-dimension of the classifier f, $L(\cdot)$ is the loss function.

Lemma B.11 (VC-dimension of linear classifier). *The VC-dimension of linear classifiers $f_\theta = \{x \to 2\mathbb{I}(\theta^T x \geq 0) - 1; \theta \in \mathbb{R}^d\}$ is d.*

Proof of Lemma B.11. We need to prove that f_θ can shatter d points and cannot shatter $d+1$ points.

To prove that f_θ can shatter d points, we only need to prove that f_θ can shatter $x_j = e_j, j \in [d]$ where e_j is the basis of the space $\mathbb{R}^d$. In fact, for every $y_j \in \{-1, +1\}$, we can let

$$\theta = \sum_{j=1}^{d} y_j \cdot e_j.$$

Then it holds that

$$2\mathbb{I}(\theta^T x_j \geq 0) - 1 = y_j$$

for all $j \in [d]$.

Then we prove that $d+1$ points cannot be shattered. We consider points $\{x_j\}_{j=1}^{d+1}$. Because $x_j \in \mathbb{R}^d$, $\{x_j\}_{j=1}^{d+1}$ are linearly dependent. Without loss of generality, we can assume that

$$x_{d+1} = \sum_{j=1}^{d} k_j x_j.$$

Now we can craft labels $\{y_j\}_{j=1}^{d+1}$ such that for any f_θ, there exists

$$f_\theta(x_j) = 2\mathbb{I}(\theta^T x_j \geq 0) - 1 \neq y_j.$$

For $k_j \neq 0$, we set $y_j = 2\mathbb{I}(k_j \geq 0) - 1$, and we set $y_{d+1} = -1$. In this case, if the classifier f_θ can correctly classify $x_1, \cdots, x_d$, it must have

$$2\mathbb{I}(\theta^T x_j \geq 0) - 1 = y_j = 2\mathbb{I}(k_j \geq 0) - 1.$$

Therefore, $\mathbb{I}(\theta^T x_j \geq 0) = \mathbb{I}(k_j \geq 0)$. However, for x_{d+1}, it has

$$\theta^T x_{d+1} = \sum_{j=1}^{d} k_j \theta^T x_j \geq 0,$$

making

$$2\mathbb{I}(\theta^T x_{d+1} \geq 0) = +1 \neq y_{d+1}.$$

Therefore, $d+1$ points cannot be shattered, resulting in $VC(f_\theta) = d$. □

Proof of Theorem 4.15. Denote the classifier h_1 and h_2 as

$$h_1(x) = 2\mathbb{I}(\zeta^T x > C_1) - 1, h_2(x) = 2\mathbb{I}(\zeta^T x < C_2) - 1.$$

Denote the loss function L_{0-1} as the 0-1 loss.

By Lemmas B.10 and B.11, with probability at least $1-\mu$, it holds that

$$\begin{aligned}\mathbb{E}_{(x+\delta,+1)}L_{0-1}\left(h_1(x+\delta),+1\right) &\leq \frac{1}{N}\sum_{i=1}^{N}L_{0-1}\left(h_1(x_i+\delta_i),+1\right)+\sqrt{\frac{d}{N}\left(\log\left(\frac{2N}{d}\right)+1\right)-\frac{1}{N}\log\left(\frac{\mu}{4}\right)}\\ &= \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(\zeta^T(x_i+\delta_i)\leq C_1)+\sqrt{\frac{d}{N}\left(\log\left(\frac{2N}{d}\right)+1\right)-\frac{1}{N}\log\left(\frac{\mu}{4}\right)}\\ &\leq \omega+\sqrt{\frac{d}{N}\left(\log\left(\frac{2N}{d}\right)+1\right)-\frac{1}{N}\log\left(\frac{\mu}{4}\right)}.\end{aligned}$$

Similarly, it has

$$\mathbb{E}_{(x,-1)}L_{0-1}\left(h_2(x),-1\right)\leq\omega+\sqrt{\frac{d}{N}\left(\log\left(\frac{2N}{d}\right)+1\right)-\frac{1}{N}\log\left(\frac{\mu}{4}\right)}$$

Therefore,

$$\begin{aligned}&\mathbb{P}_{x_1,x_2\sim\mathcal{D}_{\mathcal{X}},\delta_1\sim\Delta}\left(\left\{\zeta^T(x_1+\delta_1)>C_1,-\zeta^Tx_2<C_2\right\}\right)\\ &\geq 1-\mathbb{P}_{x_1\sim\mathcal{D}_{\mathcal{X}},\delta_1\sim\Delta}\left(\zeta^T(x_1+\delta_1)\leq C_1\right)-\mathbb{P}_{x_2\sim\mathcal{D}_{\mathcal{X}}}(\zeta^Tx_2\geq C_2)\\ &= 1-\mathbb{E}_{(x+\delta,+1)}L_{0-1}\left(h_1(x+\delta),+1\right)-\mathbb{E}_{(x,-1)}L_{0-1}\left(h_2(x),-1\right)\\ &\geq 1-2\omega-2\sqrt{\frac{d}{N}\left(\log\left(\frac{2N}{d}\right)+1\right)-\frac{1}{N}\log\left(\frac{\mu}{4}\right)}.\end{aligned}$$

□

B.3 Proofs of Theorems in Section 5

Theorem B.12 (Theorem 5.2, restated)**.** *With probability at least $1-2\omega$ for the poisoned dataset $\{(x_i',y_i)\}_{i=1}^N=S'\sim\mathcal{D}'$ and the key $\zeta\in\mathbb{R}^d$ selected from a certain distribution, we can craft the watermark δ^w satisfied:*

$$\begin{aligned}\mathcal{R}(\mathcal{D}',\mathcal{F}^*_{S'+\delta^w})\leq\mathbb{E}_\eta\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}(\mathcal{F}^*_{S'}(x_i'+\eta),y_i)+O\left(\sqrt{\frac{L}{N}}\right)\\ +O\left(\sqrt{\frac{\log d}{N}}\right)+O\left(\sqrt{\frac{\log 1/\omega}{N}}\right)+O\left(\epsilon_w\sqrt{\frac{q\log 1/\omega}{d}}\right),\end{aligned}$$

where $S'+\delta^w=\{(x_i'+\delta^w,y_i)\}_{i=1}^N$ is the watermarked dataset, $\eta\sim\mathcal{U}\{-\epsilon_w,\epsilon_w\}^q$ is a random vector.

Proof of Theorem 5.2. Let $x_i'=x_i+\delta_i^p$. For any random identical key ζ, we craft the watermark δ^w as $(\epsilon_w\cdot\zeta^{d_i})_{i=1}^q$, which obey the distribution $\mathcal{U}\{-\epsilon_w,\epsilon_w\}^q$.

We first prove that

$$\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'+\delta^w}(x_i'),y_i\right)\leq\mathbb{E}_\eta\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'}(x_i'+\eta),y_i\right)+O\left(\epsilon_w\sqrt{\frac{q\log 1/\omega}{d}}\right).$$

Let $a=\min_{\mathcal{F}}\frac{1}{N}\sum_{i=1}^{N}L(\mathcal{F}(x_i'),y_i)$, the optimal classifier

$$\mathcal{F}^*_{S'+\delta^w}(t)=\arg\min_{\mathcal{F}}\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}(x_i'+\delta^w),y_i\right),\mathcal{F}^*_{S'}(t)=\arg\min_{\mathcal{F}}\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}(\mathcal{F}(x_i'),y_i).$$

Let $G^* = \mathcal{F}^*_{S'}(t-\delta^w)$, it holds that

$$\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}(G^*(x_i'+\delta^w),y_i) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}(\mathcal{F}^*_{S'}(x_i'),y_i) = a.$$

Therefore, it has

$$\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'+\delta^w}(x_i'+\delta^w),y_i\right) \leq \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(G^*(x_i'+\delta^w),y_i\right) = a.$$

In fact, $\frac{1}{N}\sum\limits_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'+\delta^w}(x_i'+\delta^w),y_i\right) = a$. This is because, if

$$\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'+\delta^w}(x_i'+\delta^w),y_i\right) = b < a,$$

let $H^* = \mathcal{F}^*_{S'+\delta^w}(t+\delta^w)$, it has

$$\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(H^*(x_i'),y_i\right) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'+\delta^w}(x_i'+\delta^w),y_i\right) = b < a,$$

violating the condition that $a = \min\limits_{\mathcal{F}}\frac{1}{N}\sum\limits_{i=1}^{N}\mathcal{L}\left(\mathcal{F}(x_i'),y_i\right)$.

Therefore, it has

$$\mathcal{F}^*_{S'+\delta^w}(t) = \mathcal{F}^*_{S'}(t-\delta^w).$$

Then we have

$$\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'+\delta^w}(x_i'),y_i\right) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'}(x_i'-\delta^w),y_i\right)$$

Now we use the McDiarmid's inequality to complete the proof of the first part. For each (x_i,y_i), for different δ^w and $\bar{\delta}^w$ on one dimension, it holds that

$$\begin{aligned}
&\left|\mathcal{L}\left(\mathcal{F}^*_{S'}(x_i'-\delta^w),y_i\right) - \mathcal{L}\left(\mathcal{F}^*_{S'}(x_i'-\bar{\delta}^w),y_i\right)\right| \\
&= \left|\log\left(1+e^{y_i\cdot\mathcal{F}^*_{S'}(x_i'-\delta^w)}\right) - \log\left(1+e^{y_i\cdot\mathcal{F}^*_{S'}(x_i'-\bar{\delta}^w)}\right)\right| \\
&\leq \left|\mathcal{F}^*_{S'}(x_i'-\delta^w) - \mathcal{F}^*_{S'}(x_i'-\bar{\delta}^w)\right| \\
&= \Bigg|W^L\frac{1}{\sqrt{d_{L-1}}}\mathrm{ReLU}\left(W^{L-1}\cdots\frac{1}{\sqrt{d_2}}\mathrm{ReLU}\left(W^2\mathrm{ReLU}\left(\frac{1}{\sqrt{d_1}}W^1(x_i+\delta^w)+b^1\right)+b^2\right)+\cdots+b^{L-1}\right)+b^L \\
&- W^L\frac{1}{\sqrt{d_{L-1}}}\mathrm{ReLU}\left(W^{L-1}\cdots\frac{1}{\sqrt{d_2}}\mathrm{ReLU}\left(W^2\mathrm{ReLU}\left(\frac{1}{\sqrt{d_1}}W^1(x_i+\bar{\delta}^w)+b^1\right)+b^2\right)+\cdots+b^{L-1}\right)+b^L\Bigg| \\
&\leq \frac{1}{\sqrt{d}}\frac{1}{\sqrt{d_{L-1}d_{L-2}\cdots d_2}}||W^L||_{1,\infty}\ldots||W^2||_{1,\infty}||W^1||_{1,\infty}\epsilon_w.
\end{aligned}$$

By McDiarmid's inequality, let $c = \frac{1}{\sqrt{d}}\frac{1}{\sqrt{d_{L-1}d_{L-2}\cdots d_2}}||W^L||_{1,\infty}\ldots||W^2||_{1,\infty}||W^1||_{1,\infty}\epsilon_w$, it holds that

$$\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'}(x_i'-\delta^w),y_i\right) \leq \mathbb{E}_\eta\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'}(x_i'-\eta),y_i\right)+\alpha$$

with probability at least $1-\exp\left(-\frac{2\alpha^2}{q\cdot c^2}\right)$, where η obey the distribution $\mathcal{U}\{-\epsilon_w,\epsilon_w\}^q$.

Due to the symmetry of η, it always has

$$\mathbb{E}_\eta \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'}(x'_i-\eta),y_i\right)=\mathbb{E}_\eta \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'}(x'_i+\eta),y_i\right).$$

Therefore, let $\omega=\exp\left(-\frac{2\alpha^2}{q\cdot c^2}\right)$, it holds that

$$\begin{aligned}\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'}(x'_i-\delta^w),y_i\right)&\le \mathbb{E}_\eta \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}(\mathcal{F}^*_{S'}\left(x'_i+\eta\right),y_i)+\alpha\\&\le \mathbb{E}_\eta \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'}(x'_i+\eta),y_i\right)+O\left(\epsilon_w\sqrt{\frac{q\log 1/\omega}{d}}\right)\end{aligned}$$

Then we will prove that

$$\begin{aligned}\mathcal{R}(\mathcal{D}',\mathcal{F}^*_{S'+\delta^w})&=\mathbb{E}_{(x',y)\sim\mathcal{D}'}\mathcal{L}\left(\mathcal{F}^*_{S'+\delta^w}(x'),y\right)\\&\le \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'+\delta^w}(x'_i),y_i\right)+O\left(\sqrt{\frac{\log d}{N}}\right)+O\left(\sqrt{\frac{L}{N}}\right)+O\left(\sqrt{\frac{\log 1/\omega}{N}}\right).\end{aligned}$$

By [57], when loss function $L(\mathcal{F}(x),y)$ is bounded by $[0,B]$, with probability at least $1-\omega$, it holds that

$$\mathcal{R}(\mathcal{D}',\mathcal{F})\le\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}(x'_i),y_i\right)+2B\cdot\mathrm{Rad}_{(x'_i,y_i)\in S'}\left(\mathcal{L}(\mathcal{F})\right)+3B\sqrt{\frac{1}{2N}\log\frac{2}{\omega}}$$

The remaining issue is to compute $\mathrm{Rad}_{(x'_i,y_i)\in S'}\left(\mathcal{L}(\mathcal{F}^*_{S'+\delta^w})\right)$.

As loss function $L(z)=\log(1+e^{-z})$ is 1-Lipschitz under $z=y\cdot\mathcal{F}^*_{S'+\delta^w}(x)$, by Talagrand's contraction lemma [49, 58],

$$\begin{aligned}\mathrm{Rad}_{(x'_i,y_i)\in S'}\left(\mathcal{L}(\mathcal{F}^*_{S'+\delta^w})\right)&=\mathbb{E}_{\sigma_i\in\{-1,+1\}}\left[\sup_{L(\mathcal{F}^*_{S'+\delta^w})}\frac{1}{N}\sum_{i=1}^{N}\sigma_i L(\mathcal{F}^*_{S'+\delta^w}(x'_i),y_i)\right]\\&\le\mathbb{E}_{\sigma_i\in\{-1,+1\}}\left[\sup_{\mathcal{F}^*_{S'+\delta^w}}\frac{1}{N}\sum_{i=1}^{N}\sigma_i y_i\mathcal{F}^*_{S'+\delta^w}(x'_i)\right]\\&=\mathbb{E}_{\sigma_i\in\{-1,+1\}}\left[\sup_{\mathcal{F}^*_{S'+\delta^w}}\frac{1}{N}\sum_{i=1}^{N}\sigma_i\mathcal{F}^*_{S'+\delta^w}(x'_i)\right]\\&=\mathrm{Rad}_{(x'_i,y_i)\in S'}\left(\mathcal{F}^*_{S'+\delta^w}\right)\end{aligned}$$

From Theorem 1 in [84], it holds that

$$\mathrm{Rad}_{(x'_i,y_i)\in S'}\left(\mathcal{F}^*_{S'+\delta^w}\right)\le\prod_{l=1}^{L}\left(\|W^l\|_{1,\infty}+\|b^l\|_\infty\right)\left(\sqrt{\frac{(L+2)\log 4}{N}}+\sqrt{\frac{2\log(2d)}{N}}\right)$$

Therefore, it has

$$\begin{aligned}\mathcal{R}\left(\mathcal{D}',\mathcal{F}^*_{S'+\delta^w}\right)&\le\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'+\delta^w}(x'_i),y_i\right)+\log\left(1+\exp\left(\max_x\mathcal{F}^*_{S'+\delta^w}(x)\right)\right)\cdot\\&\quad\left(2\cdot\mathrm{Rad}_{(x'_i,y_i)\in S'}(\mathcal{L}(\mathcal{F}^*_{S'+\delta^w}))+3\sqrt{\frac{1}{2N}\log\frac{2}{\omega}}\right)\\&\le\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'+\delta^w}(x'_i+\delta^w),y_i\right)+O\left(\sqrt{\frac{\log d}{N}}\right)+O\left(\sqrt{\frac{L}{N}}\right)+O\left(\sqrt{\frac{\log 1/\omega}{N}}\right)\\&\quad+O\left(\sqrt{\frac{q\epsilon_w\log 1/\omega}{d}}\right)+O\left(\frac{\epsilon_w}{\sqrt{d}}\right)\end{aligned}$$

□

Theorem B.13 (Theorem 5.6, restated)**.** *With probability at least* $1-\omega$ *of the (unrestricted) poisoned dataset* $\{(x_i+\delta_i^p, y_i)\}_{i=1}^N = S' \sim \mathcal{D}'$*, it holds that*

$$\mathcal{R}\left(\mathcal{D}', \mathcal{F}^*_{S'|_{\mathcal{P}}}\right) \leq \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'|_{\mathcal{P}}}\left(x_i+\delta_i^p|_{\mathcal{P}}\right), y_i\right) + O\left(\frac{q\epsilon_p}{\sqrt{d}}\right) + O\left(\sqrt{\frac{\log d}{N}}\right) + O\left(\sqrt{\frac{L}{N}}\right) + O\left(\sqrt{\frac{\log 1/\omega}{N}}\right).$$

Proof of Theorem 5.6. For every i,

$$\begin{aligned}
&\mathcal{L}\left(\mathcal{F}^*_{S'|_{\mathcal{P}}}\left(x_i+\delta_i^p\right), y_i\right) - \mathcal{L}\left(\mathcal{F}^*_{S'|_{\mathcal{P}}}\left(x_i+\delta_i^p|_{\mathcal{P}}\right), y_i\right)\\
&\leq \left|\mathcal{F}^*_{S'|_{\mathcal{P}}}\left(x_i+\delta_i^p\right) - \mathcal{F}^*_{S'|_{\mathcal{P}}}\left(x_i+\delta_i^p|_{\mathcal{P}}\right)\right|\\
&= \Bigg|W^L\frac{1}{\sqrt{d_{L-1}}}\mathrm{ReLU}\left(W^{L-1}\cdots\frac{1}{\sqrt{d_2}}\mathrm{ReLU}\left(L_2\right)+\cdots+b^{L-1}\right)+b^L\\
&- W^L\frac{1}{\sqrt{d_{L-1}}}\mathrm{ReLU}\left(W^{L-1}\cdots\frac{1}{\sqrt{d_2}}\mathrm{ReLU}\left(L_2\right)+\cdots+b^{L-1}\right)+b^L\Bigg|\\
&\leq \frac{1}{\sqrt{d}}\frac{1}{\sqrt{d_{L-1}d_{L-2}\cdots d_2}}\|W^L\|_{1,\infty}\|W^{L-1}\|_{1,\infty}\cdots\|W^2\|_{1,\infty}\|W^1\delta_i^p|_{[d]-\mathcal{P}}\|_1\\
&\leq \frac{1}{\sqrt{d_{L-1}d_{L-2}\cdots d_2}}\|W^L\|_{1,\infty}\|W^{L-1}\|_{1,\infty}\cdots\|W^2\|_{1,\infty}\|W^1\|_{1,\infty}\cdot\frac{|[d]-\mathcal{P}|\cdot\|\delta_i^p\|_\infty}{\sqrt{d}}\\
&\leq \frac{1}{\sqrt{d_{L-1}d_{L-2}\cdots d_2}}\|W^L\|_{1,\infty}\|W^{L-1}\|_{1,\infty}\cdots\|W^2\|_{1,\infty}\|W_1\|_{1,\infty}\cdot\frac{q\epsilon_p}{\sqrt{d}}.
\end{aligned}$$

where $L_2 = W^2\mathrm{ReLU}\left(\frac{1}{\sqrt{d_1}}W^1\left(x_i+\delta_i^p\right)+b^1\right)+b^2$.

Therefore, it holds that

$$\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'|_{\mathcal{P}}}(x_i+\delta_i^p), y_i\right) \leq \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'|_{\mathcal{P}}}(x_i+\delta_i^p|_{\mathcal{P}}), y_i\right) + O\left(\frac{q\epsilon_p}{\sqrt{d}}\right).$$

Then, similar to the proof of Theorem 5.2, it has

$$\begin{aligned}
\mathcal{R}(\mathcal{D}', \mathcal{F}^*_{S'|_{\mathcal{P}}}) &\leq \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'|_{\mathcal{P}}}(x_i+\delta_i^p), y_i\right) + \log\left(1+\exp\left(\max_x \mathcal{F}^*_{S'|_{\mathcal{P}}}(x)\right)\right)\cdot\\
&\quad\left(2\cdot\mathrm{Rad}_{(x_i',y_i)\in S'}\left(\mathcal{L}(\mathcal{F}^*_{S'|_{\mathcal{P}}})\right) + 3\sqrt{\frac{1}{2N}\log\frac{2}{\omega}}\right)\\
&\leq \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'|_{\mathcal{P}}}(x_i+\delta_i^p), y_i\right) + O\left(\sqrt{\frac{\log d}{N}}\right) + O\left(\sqrt{\frac{L}{N}}\right) + O\left(\sqrt{\frac{\log 1/\omega}{N}}\right)\\
&\leq \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(\mathcal{F}^*_{S'|_{\mathcal{P}}}(x_i+\delta_i^p|_{\mathcal{P}}), y_i\right) + O\left(\frac{q\epsilon_p}{\sqrt{d}}\right) + O\left(\sqrt{\frac{\log d}{N}}\right)\\
&\quad + O\left(\sqrt{\frac{L}{N}}\right) + O\left(\sqrt{\frac{\log 1/\omega}{N}}\right).
\end{aligned}$$

□

Corollary B.14 (Corollary 5.7, restated). *With probability at least $1-3\omega$ for the restricted poisoned dataset $S'|_{\mathcal{P}} \sim \mathcal{D}'|_{\mathcal{P}}$ and the key $\zeta \in \mathbb{R}^d$ selected from certain distribution, we can craft the watermark δ^w satisfied:*

$$\mathcal{R}(\mathcal{D}', \mathcal{F}^*_{\tilde{S}}) \leq \mathbb{E}_\eta \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}\left(\mathcal{F}^*_{S|_{\mathcal{P}}}(x_i + \delta_i^p|_{\mathcal{P}} + \eta), y_i\right)$$
$$+ O\left(\sqrt{\frac{L}{N}}\right) + O\left(\sqrt{\frac{\log d}{N}}\right) + O\left(\sqrt{\frac{\log 1/\omega}{N}}\right) + O\left(\frac{q\epsilon_p}{\sqrt{d}}\right) + O\left(\epsilon_w \sqrt{\frac{q \log 1/\omega}{d}}\right),$$

where $\tilde{S} = S|_{\mathcal{P}} + \delta^w$ is the watermarked dataset.

Proof. By Theorem 5.2, with probability at least $1-2\omega$ it holds that

$$\mathcal{R}(\mathcal{D}', \mathcal{F}^*_{\tilde{S}}) \leq \mathbb{E}_\eta \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}\left(\mathcal{F}^*_{S|_{\mathcal{P}}}(x_i + \delta_p^i + \eta), y_i\right) + O\left(\sqrt{\frac{L}{N}}\right)$$
$$+ O\left(\sqrt{\frac{\log d}{N}}\right) + O\left(\sqrt{\frac{\log 1/\omega}{N}}\right) + O\left(\epsilon_w \sqrt{\frac{q \log 1/\omega}{d}}\right).$$

By Theorem 5.6, with probability at least $1-\omega$, for every η, it holds that

$$\frac{1}{N} \sum_{i=1}^{N} \mathcal{L}\left(\mathcal{F}^*_{S'|_{\mathcal{P}}}(x_i + \delta_i^p + \eta), y_i\right) \leq \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}\left(\mathcal{F}^*_{S'|_{\mathcal{P}}}(x_i + \delta_i^p|_{\mathcal{P}} + \eta), y_i\right) + O\left(\frac{q\epsilon_p}{\sqrt{d}}\right).$$

Combine the above two inequalities directly to complete the proof. □

C Watermarking Algorithm

Algorithm 1 Post-Poisoning Watermarking

Input: The poisoned training dataset $D_{\mathcal{P}} = \{(x_i + \delta_i^p, y_i)\}_{i=1}^N$. The key ζ.
Output: Watermarked training dataset $D_{\mathcal{W}} = \{(x_i + \delta_i^p + \delta^w, y_i)\}_{i=1}^N$.
Choose the watermarking dimension $\mathcal{W}$.
Set $\delta^w = \epsilon_w \cdot \text{sign}(\zeta)|_{\mathcal{W}}$.

Algorithm 2 Poisoning-Concurrent Watermarking

Input: The training dataset $D_{\mathcal{P}} = \{(x_i, y_i)\}_{i=1}^N$. The key ζ.
Output: Watermarked poisoned training dataset $D_{\mathcal{W}} = \{(x_i + \delta_i^p + \delta^w, y_i)\}_{i=1}^N$.
Choose the watermarking dimension $\mathcal{W}$.
Set $\delta^w = \epsilon_w \cdot \text{sign}(\zeta)|_{\mathcal{W}}$.
Update poisons δ_i^p on poisoning dimension $\mathcal{P} = [d] - \mathcal{W}$.

Algorithm 3 Detection

Input: The suspect training data $\tilde{x}$. The key ζ. The detection threshold τ.
Output: 1 (Positive) or 0 (Negative).
Compute the detection value $v = \zeta^T \tilde{x}$
If $v > \tau$, return 1, else $v \leq \tau$, return 0.

D Additional Experiments

D.1 Additional Experiments on More Datasets

We extend our evaluation to CIFAR-100 and TinyImageNet for UE and AP poisons on Table 3 and Table 4 respectively. Results demonstrate similar trends: as the watermark length increases,

Table 3: The clean accuracy (Acc, %) and AUROC of UE and AP availability attacks both on post-poisoning watermarking and poisoning-concurrent watermarking with different watermarking length q under ResNet-18 and CIFAR-100.

Length/Method		UE		AP
Acc(↓)/AUROC(↑)	Post-Poisoning	Poisoning-Concurrent	Post-Poisoning	Poisoning-Concurrent
0(Baseline)	1.24/-	1.24/-	1.71/-	1.71/-
100	1.21/0.5796	1.15/0.8064	1.75/0.5913	1.66/0.6950
300	1.25/0.7145	1.43/0.8839	1.66/0.7667	1.69/0.7732
500	1.19/0.7822	2.51/0.9150	1.77/0.8710	1.85/0.8931
1000	1.43/0.9354	1.46/0.9758	1.72/0.9669	2.36/0.9949
1500	1.10/0.9963	1.66/0.9992	1.68/0.9893	2.19/0.9995
2000	1.28/0.9982	3.49/0.9999	1.90/0.9986	6.98/1.0000
2500	1.36/0.9995	54.10/1.0000	1.76/1.0000	32.41/1.0000
3000	1.57/1.0000	71.04/1.0000	2.36/1.0000	69.85/1.0000

Table 4: The clean accuracy (Acc, %) and AUROC of UE and AP availability attacks both on post-poisoning watermarking and poisoning-concurrent watermarking with different watermarking length q under ResNet-18 and TinyImageNet.

Length/Method		UE		AP
Acc(↓)/AUROC(↑)	Post-Poisoning	Poisoning-Concurrent	Post-Poisoning	Poisoning-Concurrent
0(Baseline)	0.75/-	0.75/-	9.37/-	9.37/-
500	1.15/0.7850	1.24/0.9623	11.85/0.8054	8.74/0.9794
1000	0.92/0.8587	1.70/0.9952	8.62/0.8620	11.25/0.9967
2000	0.95/0.9596	3.69/0.9994	13.40/0.9640	22.61/0.9998
5000	2.23/0.9998	11.01/0.9999	22.17/1.0000	43.30/1.0000
10000	7.14/1.0000	48.32/1.0000	36.81/1.0000	47.05/1.0000

detectability improves (higher AUROC), while poisoning effectiveness decreases (higher clean accuracy), confirming our theoretical claims.

Furthermore, for text dataset, we implement watermarking ($\epsilon_w = 16/255$) in a backdoor attack on SST-2 dataset with BERT-base model [24], observing similar trends compared with other visual datasets for this NLP task.

Table 5: The accuracy (Acc, %), ASR and AUROC of SST-2 dataset on BERT-base model with different watermarking length q.

Length	Post-Poisoning Acc/ASR/AUROC	Poisoning-Concurrent Acc/ASR/AUROC
0	89.7/98.0/-	89.7/98.0/-
100	89.8/97.8/0.697	89.6/97.2/0.969
200	89.2/97.3/0.852	89.9/96.1/0.983
400	89.6/96.2/0.931	89.3/90.5/0.998
600	89.3/96.7/0.983	89.5/72.3/0.999

D.2 Additional Experiments on More Network Structures

For model transferability, we evaluate our watermarking with length $q = 1000$ across ResNet-50, VGG-19, DenseNet121, WRN34-10, MobileNet v2 and ViT-B models. Results shown in Table 6 and Table 7 demonstrate strong transferability (high AUROC and low accuracy) across network architectures, further validating our theoretical insights.

D.3 Results under Different Watermarking Budget

We evaluate our watermarking algorithms under different watermarking budgets—$4/255$, $8/255$, $16/255$, and $32/255$—with a fixed watermarking length of 1000. The results indicate that as the budget ϵ_w increases, detectability improves while poisoning effectiveness declines. This aligns with

Table 6: The clean accuracy (Acc, %), attack success rate (ASR, %), and AUROC of Narcissus and AdvSc backdoor attacks on both post-poisoning watermarking and poisoning-concurrent watermarking with various victim models under CIFAR-10.

Model/Method	Narcissus		AdvSc	
Acc/ASR/AUROC(↑)	Post-Poisoning	Poisoning-Concurrent	Post-Poisoning	Poisoning-Concurrent
ResNet-18	94.40/92.43/0.9974	94.32/92.03/0.9992	93.05/94.41/0.9809	93.38/84.39/0.9995
ResNet-50	94.46/93.12/0.9969	94.85/93.01/0.9985	92.55/93.30/0.9827	92.16/86.53/0.9995
VGG-19	93.74/91.80/0.9975	92.61/91.97/0.9995	91.47/93.94/0.9926	91.80/79.34/0.9999
DenseNet121	94.18/92.66/0.9977	94.52/92.39/0.9990	94.12/93.73/0.9905	92.67/90.32/0.9998
WRN34-10	94.95/92.14/0.9981	95.02/91.36/0.9989	94.74/94.85/0.9860	94.12/89.63/0.9994
MobileNet v2	94.63/92.41/0.9972	94.15/92.14/0.9986	93.63/94.51/0.9754	93.75/83.29/0.9996
ViT-B	94.87/94.25/0.9991	95.25/93.37/1.0000	94.32/93.26/0.9922	94.23/91.45/1.000

Table 7: The clean accuracy (Acc,%) and AUROC of UE and AP availability attacks both on post-poisoning watermarking and poisoning-concurrent watermarking with various victim models under CIFAR-10.

Model/Method	UE		AP	
Acc(↓)/AUROC(↑)	Post-Poisoning	Poisoning-Concurrent	Post-Poisoning	Poisoning-Concurrent
ResNet-18	11.37/0.9499	9.42/0.9991	10.58/0.9742	21.87/0.9949
ResNet-50	10.15/0.9583	12.26/0.9992	9.97/0.9678	14.76/0.9947
VGG-19	12.96/0.9644	12.21/0.9993	10.80/0.9800	20.34/0.9952
DenseNet121	19.30/0.9545	17.87/0.9985	12.35/0.9767	11.76/0.9978
WRN34-10	12.31/0.9702	10.55/0.9988	10.24/0.9821	15.98/0.9958
MobileNet v2	14.03/0.9473	16.90/0.9986	11.36/0.9726	18.51/0.9941
ViT-B	13.97/0.9728	14.80/0.9989	10.51/0.9793	12.75/0.9970

our theoretical findings: as ϵ_w grows, both $\Omega(\sqrt{d}/\epsilon_w)$ (post-poisoning) and $\Omega(1/\epsilon_w^2)$ (poisoning-concurrent) decrease, leading to better detectability. Additionally, the error term $O\left(\epsilon_w\sqrt{\frac{q\log 1/\omega}{d}}\right)$ (Theorem 5.2 and Corollary 5.7) influences poisoning effectiveness, meaning a larger ϵ_w weakens the poisoning power guarantee. This is evident in our results, where AdvSc achieves only 60.04% and 36.45% ASR under $\epsilon_w = 32/255$ for post-poisoning and poisoning-concurrent watermarking, a trend also observed in Figure 1 in Section 6.3.

Table 8: The clean accuracy (Acc, %), attack success rate (ASR, %), and AUROC of Narcissus and AdvSc backdoor attacks on both post-poisoning watermarking and poisoning-concurrent watermarking under different watermarking budgets on CIFAR-10 dataset.

Budget/Method	Narcissus		AdvSc	
Acc/ASR/AUROC(↑)	Post-Poisoning	Poisoning-Concurrent	Post-Poisoning	Poisoning-Concurrent
4/255	94.35/94.28/0.9114	94.43/94.21/0.8297	92.94/98.68/0.8132	93.25/91.68/0.8655
8/255	94.71/93.76/0.9535	94.99/92.69/0.8948	93.04/98.88/0.9427	93.27/87.48/0.9651
16/255	94.40/92.43/0.9974	94.32/92.03/0.9992	93.05/94.41/0.9809	93.38/84.39/0.9995
32/255	94.86/90.66/0.9998	94.87/80.17/1.0000	93.13/60.04/0.9999	92.76/36.45/1.0000

D.4 Watermarking on Clean Samples

Beyond data poisoning, we test watermarking on clean CIFAR-10 with ϵ_w be 4/255, 8/255, 16/255 and 32/255 on Table 9. The results indicate strong detectability with minimal accuracy degradation, even for large perturbations (32/255). It is worth noting that, for clean samples, post-poisoning and poisoning-concurrent watermarking will become the same as there are no poisons involved.

D.5 Computational Cost

We evaluate the computational overheads for our watermarking techniques on UE and AP availability attacks, as well as Narcissus and AdvSc backdoor attacks. All experiments are evaluated on a single NVIDIA A800 80GB PCIe GPU. Results in Table 10 show that our watermarking is highly efficient,

Table 9: The accuracy (Acc, %) and AUROC of clean CIFAR-10 dataset with different watermarking length q under ResNet-18.

Budget Length	4/255 Acc/AUROC	8/255 Acc/AUROC	16/255 Acc/AUROC	32/255 Acc/AUROC
0	95.25/-	95.25/-	95.25/-	95.25/-
200	95.12/0.5527	94.85/0.6218	94.75/0.7854	94.48/0.8672
500	94.90/0.6638	94.53/0.8317	93.66/0.9683	91.66/0.9990
1000	94.56/0.8679	94.08/0.9700	92.87/0.9929	89.54/1.0000
1500	94.22/0.9491	93.82/0.9764	92.02/0.9998	91.60/1.0000
2000	94.01/0.9736	93.37/0.9946	90.34/1.0000	88.20/1.0000
2500	93.86/0.9935	93.49/1.0000	88.70/1.0000	83.20/1.0000

requiring only seconds for post-poisoning watermarking and detection. Even for poisoning-concurrent watermarking, it incurs a minimal 10-minute overhead. Therefore, we believe our watermarking schemes are efficient to deploy in real-world applications.

Table 10: The time cost of our watermarking techniques under CIFAR-10 dataset on various data poisoning attacks.

Time	UE	AP	Narcissus	AdvSc
Poisoning(baseline)	≈80min	≈65min	≈70min	≈190min
Post-poisoning	≈30s	≈30s	≈30s	≈30s
Poisoning-concurrent	≈90min	≈70min	≈75min	≈200min
Detection	≈40s	≈40s	≈40s	≈40s

E Robust Watermarking under Various Defenses and Removals

Data augmentation and image regeneration. Under some data augmentations or image reconstructions, the provable watermarking may not hold because the relative position between watermarks and keys has been broken. However, we can train a watermark detector with the known key, and judge whether the data is watermarked with the detector. Specifically, denote the clean dataset as $\{(x_i, y_i)\}_{i=1}^N$, where $x_i \in \mathbb{R}^d$ is the data, $y_i \in \mathbb{Z}$ is the label. The key $\zeta \in \mathbb{R}^d$. We craft the watermark detection training set $\mathcal{D}_d$ as $\{(x_i, 0)\}_{i=1}^N \cap \{(x_i + \epsilon_w \cdot \zeta, 1)\}_{i=1}^N$, where ϵ_w is a small budget that the injected watermarks δ^w compromise, and train a detector $\mathcal{T}$ with $\mathcal{D}_d$ under data augmentations. For a suspect data $\tilde{x}$ which may be poisoned with watermarking, we argue that $\tilde{x}$ is poisoned if $\mathcal{T}(\tilde{x}) = 1$; otherwise, $\tilde{x}$ is recognized as benign data. We evaluate the performance of our detector under several data augmentations, including Random Flip, Cutout [16], Color Jitter and Grayscale. Furthermore, we also evaluate the watermarking performance under some regeneration attacks including VAE-based attack [14] and generative adversarial network [78]. Experimental results presented in Table 11 have shown stronger detection performance, validating the robustness of our proposed watermarking.

Table 11: The detection performance (AUROC) of poisoning-concurrent watermarking of UE and AP with watermarking length be 500 under various data augmentations.

Type	Random Flip	Cutout	Color Jitter	Grayscale	VAE [14]	GAN [78]
UE	1.0000	1.0000	1.0000	0.9930	0.9987	0.9853
AP	1.0000	1.0000	1.0000	0.9996	0.9395	0.9830

Differential privacy noises. To further evaluate the robustness of our watermarking, we consider adaptive attacks based on (ϵ, δ)-DP, applying both Gaussian and Laplacian mechanisms with $\epsilon = 2, \delta = 10^{-5}$. We evaluate them on poisoning-concurrent watermarking with $q = 1500$ under UE, AP, Narcissus and AdvSc, results are shown in Table 12. Unfortunately, due to the extremely large noise level introduced in the pixel space (e.g., $\sigma = \frac{\Delta}{\epsilon} = \frac{8/255 \cdot 3072}{2} = 48$, for the Laplacian mechanism) to

the pixel space, the network fails to converge. This is because DP mechanisms are typically applied to neural network gradients or parameters, not directly to training data, and the severe perturbation causes samples from different classes to become indistinguishable.

It may be counterintuitive that UE and AP achieve lower clean accuracy under DP noise. Under normal training, UE and AP can still converge, reaching nearly 100% training and validation accuracy but only about 10% test accuracy, consistent with availability attack objectives. In contrast, when training on DP-perturbed data, the training/validation accuracy also drops to about 10%, indicating complete training failure. This contradicts the goal of availability attacks, which aim to deceive victims into believing the model is well-trained, while failing on unseen test data (see [37] for details). Notably, backdoor attacks don't exhibit this confusion as they seek high ASR rather than low accuracy. Although DP-based defenses reduce the detection performance of watermarking, the poisoning utilities have been completely destroyed. Therefore, DP-based defenses are not applicable in our context.

Table 12: Clean accuracy(Acc, %), attack success rate(ASR, %) and AUROC of poisoning-concurrent watermarking with length be 1500 under DP noises.

ACC/ASR/AUROC	DP-Gaussian	DP-Laplacian
UE	14.01/-/0.8016	12.79/-/0.5759
AP	15.85/-/0.7923	10.88/-/0.6232
Narcissus	13.37/10.12/0.8135	11.76/9.98/0.6126
AdvSc	15.11/10.03/0.7447	11.15/10.06/0.5880

Diffusion purification. For diffusion purification [94], results are shown in Table 13. Although our watermarking exhibits weak detectability, it is important to note that the poison utility is simultaneously eliminated. As shown in the following table, diffusion purification significantly mitigates availability poisoning attacks, recovering test accuracy from about 10% to over 80%. It also destroys backdoor poisoning attacks, reducing the attack success rate to less than 20%. This is reasonable as diffusion purification is a powerful defense against noise injection, including adversarial attacks [60], availability attacks [17] and diffusion model watermarking [34].

In our scenario, watermarking is designed to serve the purpose of data poisoning. If the poisoning itself is neutralized, the effectiveness of the watermark becomes irrelevant. Given that our work focuses on imperceptible poisoning and watermarking, this limitation appears to be an inherent trade-off. Similar to DP-based defenses, although diffusion purification reduces the detection performance of watermarking, the poisoning utilities have been completely destroyed. Therefore,diffusion purification is also not applicable in our context.

Table 13: Accuracy(ACC), attack success rate(ASR) and AUROC of poisoning-concurrent watermarking with length be 1500 under diffusion purification.

Type	UE	AP	Narcissus	AdvSc
ACC/ASR/AUROC	84.67/-/0.5251	85.22/-/0.5189	93.17/16.86/0.5375	93.08/10.01/0.5420

Potential removal methods. We conduct additional experiments on UE and AP with direct masking of the known watermarking dimensions (Masking), as well as the adversarial noising proposed by [54]. We test both post-poisoning and poisoning-concurrent watermarking under $q = 2000$.As the results shown below, although the detection performance (AUROC) drops, the utility of UE and AP also degrades significantly. The underlying reasons may be that availability attacks are designed with potential linear shortcut features [86, 98], the masking of watermarking dimensions somehow destroys these linear features, undermining the unlearnability (low Acc). Adversarial Noising further destroys the poisoning utility as availability attacks are theoretically removed by perfect adversarial training [76]. Therefore, these adaptive removal attacks fail to maintain the poisoning utility, making them not applicable in our cases.

Table 14: Accuracy(Acc) and AUROC of UE and AP availability attacks under potential removal methods, masking and adversarial noising.

Acc/AUROC	Baseline	Masking	Adversarial Noising
UE(Post-Poisoning)	9.06/0.9992	60.71/0.4998	72.90/0.5893
AP(Post-Poisoning)	10.48/0.9987	56.85/0.5005	76.21/0.5616
UE(Poisoning-Concurrent)	10.03/1.0000	55.49/0.5014	68.37/0.6206
AP(Poisoning-Concurrent)	38.62/1.0000	59.87/0.5002	74.63/0.5833

Figure 3: Visualization of UE poisoning-concurrent watermarking with length $q = 500$ for CIFAR-10 dataset. The first row is the benign images, the second row is the normalized UE poisons, the third row is the normalized watermarks, the fourth row is the perturbed images under watermarking poisons.

F Visualization

To further substantiate the imperceptibility of our proposed watermarking, we visualize the benign images, poisons, watermarks, and modified images. Both poisons and watermarks are normalized to $[0, 1]$ in order to improve their visibility. Figure 3 shows the watermarking visualization under UE poisons; our watermarking demonstrates strong imperceptibility.

G Covertness of Watermarking

For an practical watermarking, beyond their detectability, it also requires *covertness*. That means, if users do not obtain the watermarking key ζ, it is hard for them to discern poisoned data and benign data. In other words, if the key ζ is random (independent from the watermarks δ^w), the performance between poisoned data $x' + \delta^w$ and benign data x under random key ζ will have negligible difference. We will prove this property for post-poisoning watermarking; the property of poisoning-concurrent watermarking also holds similarly.

Theorem G.1 (Covertness for post-poisoning watermarking)**.** *For post-poisoning watermarking with watermarks δ^w, assume that the poisoned data $x' = x + \delta^p_x$, and the benign data $\bar{x}$ are independently sampled from the data distribution $\mathcal{D}$. For the random identical key $\zeta \in \mathbb{R}^d$, it holds that $\mathbb{E}_\zeta\left[\zeta^T(x' + \delta^w)\right] = \mathbb{E}_\zeta\left[\zeta^T\tilde{x}\right]$. Furthermore, it holds that $\mathbb{P}_\zeta\left[\left|\zeta^T(x' + \delta^w) - \zeta^T\tilde{x}\right| \leq \sqrt{\frac{d}{2}\log\frac{2}{\omega}}\right] > 1 - \omega$.*

Proof of Theorem G.1. As $\zeta \in \mathbb{R}^d$ is the random identical key, it holds that

$$\mathbb{E}_\zeta\left[\zeta^T(x' + \delta^w)\right] = 0$$

as well as

$$\mathbb{E}_\zeta\left[\zeta^T\tilde{x}\right] = 0.$$

Therefore, it has

$$\mathbb{E}_\zeta \left[\zeta^T (x' + \delta^w)\right] = \mathbb{E}_\zeta \left[\zeta^T \tilde{x}\right].$$

Additionally, as $x' + \delta^w$ and $\bar{x}$ both lie in $[0, 1]$, it always has

$$|\zeta^i (x' + \delta^w)^i - \zeta^i \tilde{x}^i| \leq |\zeta^i| = 1$$

for all i, ζ.

Therefore, by McDiarmid's inequality, for any $\alpha > 0$, it has

$$\mathbb{P}_\zeta[|\zeta^T (x' + \delta^w - \tilde{x})| \geq \alpha] \leq 2e^{-\frac{2\alpha^2}{d}}.$$

Therefore, let

$$\omega = 2e^{-\frac{2\alpha^2}{d}},$$

it has

$$\alpha = \sqrt{\frac{d}{2} \log \frac{2}{\omega}}.$$

□

Remark G.2. For post-poisoning watermarking, if a detector does not obtain the key, the expected predictions for the (watermarked) poisoned data and (unwatermarked) benign data are equal. Therefore, it is hard to detect watermarks without the key.

We validate this property on two backdoor attacks, Narcissus and AdvSc, and two availability attacks, UE and AP. We consider the post-poisoning watermarking with watermarking length $q = 2000$, and test the detection performance of the corresponding watermarking key and the random identical key independently from the watermarking δ^w. The results shown in Figure 4 demonstrate that, if the detector just uses a random key for detection, the AUROC is approaching 0.5, meaning that it is ineffective and almost like a random guess.

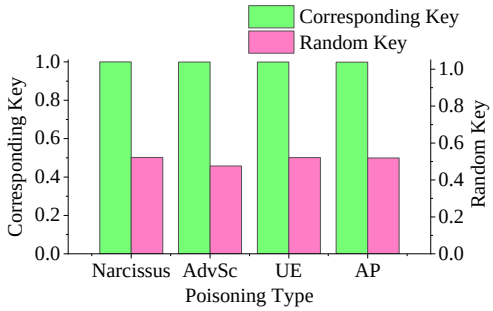

Figure 4: The detection performance (AUROC) of post-poisoning watermarking of several data poisoning attacks under corresponding key and a random key.

H Boarder Impact Statement

This paper aims at crafting watermarks for data poisoning attacks. As a method to ensure authorized users can identify potential data poisoning, we believe our work is beneficial to the community and does not have a negative social impact.