THE GOOD, THE BAD AND THE UGLY: WATERMARKS, TRANSFERABLE ATTACKS AND ADVERSARIAL DEFENSES

Grzegorz Głuch
Simons Institute & UC Berkeley,
California, USA
gluch@berkeley.edu

Berkant Turan*
Zuse Institute Berlin
Berlin, Germany
turan@zib.de

Sai Ganesh Nagarajan*
Zuse Institute Berlin
Berlin, Germany
nagarajan@zib.de

Sebastian Pokutta
Zuse Institute Berlin
Berlin, Germany
pokutta@zib.de

ABSTRACT

We formalize and analyze the trade-off between backdoor-based watermarks and adversarial defenses, framing it as an interactive protocol between a verifier and a prover. While previous works have primarily focused on this trade-off, our analysis extends it by identifying transferable attacks as a third, counterintuitive but necessary option. Our main result shows that for all learning tasks, at least one of the three exists: a *watermark*, an *adversarial defense*, or a *transferable attack*. By transferable attack, we refer to an efficient algorithm that generates queries indistinguishable from the data distribution and capable of fooling *all* efficient defenders. Using cryptographic techniques, specifically fully homomorphic encryption, we construct a transferable attack and prove its necessity in this trade-off. Furthermore, we show that any task that satisfies our notion of a transferable attack implies a cryptographic primitive, thus requiring the underlying task to be computationally complex. Finally, we show that tasks of bounded VC-dimension allow adversarial defenses against all attackers, while a subclass allows watermarks secure against fast adversaries.

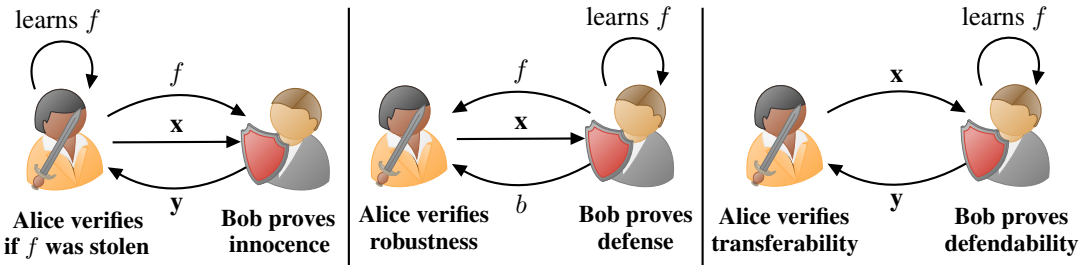
1 INTRODUCTION

An organization has invested significant resources into training a classifier f. Before releasing f as open-source, they want to ensure that any unauthorized use can be detected in a black-box manner. In other words, they want to embed a watermark into f (Adi et al., 2018; Zhang et al., 2018). Alice, an employee, is assigned to this project.

Meanwhile, Bob, a member of an AI security team, has a different objective: he wants to make f adversarially robust, meaning that it should be difficult to find queries that appear natural yet cause f to misclassify (Madry et al., 2018; Raghunathan et al., 2018). However, both Alice and Bob encounter fundamental challenges. After many attempts, Alice suspects that creating a black-box watermark in f that cannot be removed might be inherently impossible (Goldwasser et al., 2024). Similarly, Bob struggles to produce a defense that protects against all attacks—his best efforts result in an ever-growing, "ugly" defense (Carlini, 2024).

One day, Alice and Bob discuss their respective struggles and realize that their goals are intimately connected. Alice's approach to watermarking involves planting a backdoor in f, creating f_A, so that she can later craft queries with a hidden trigger that activates the backdoor, causing f_A to misclassify while remaining indistinguishable from normal queries (Adi et al., 2018; Merrer et al., 2017). If someone uses f_A, she can detect it by sending such tailored queries and analyzing the responses.

*Equal contribution.



(a) A **Watermark** is an efficient algorithm that computes a low-error classifier f and a set of queries $\mathbf{x}$ such that (fast) defenders are unable to find low-error answers $\mathbf{y}$ nor distinguish $\mathbf{x}$ from the data distribution.

(b) An **Adversarial Defense** is an efficient algorithm that computes a low-error classifier f and a detection bit b, such that (fast) adversaries are unable to find queries $\mathbf{x}$, which look indistinguishable from the data distribution and where f is incorrect.

(c) A **Transferable Attack** is an efficient algorithm that computes queries $\mathbf{x}$ that look indistinguishable from the data distribution, and that fool all efficient defenders.

Figure 1: Schematic overview of the interaction structure, along with short, informal versions of our definitions of (a) Watermark (Definition 3), (b) Adversarial Defense (Definition 4), and (c) Transferable Attack (Definition 5), with (c) tied to cryptography (see Section 5).

Bob, on the other hand, is trying to make such an attack impossible. His strategy is to take f and smooth its outputs to obtain f_B, aiming for robustness (Cohen et al., 2019). However, he realizes that this process also removes some of Alice's watermarking techniques Goldwasser et al. (2022; 2024). Conversely, Alice notices that if a watermark is difficult to remove, then certain models must be inherently difficult to make robust (Weng et al., 2020; Fowl et al., 2021).

At this point, Alice and Bob believe they have mapped the entire landscape: if one goal is impossible, the other must be achievable. However, this assumption is incomplete. There exists a third, counterintuitive but necessary alternative: some learning tasks allow neither a secure watermark nor a robust defense, but instead support a completely different phenomenon—*transferable attacks*.

1.1 Contributions

Motivated by empirical findings that adversarial defenses and backdoor-based watermarks are at a trade-off, we initiate a formal study of this fundamental interplay. Our main result shows that:

For all learning tasks, at least one of the three must exist:
A Watermark, an Adversarial Defense, or a Transferable Attack.

To prove this, we formalize and extend existing definitions of watermarks and adversarial defenses, and frame Alice and Bob's dynamic as an interactive protocol. This protocol always has at least one winner—either Alice can embed an unremovable watermark, Bob can construct a strong adversarial defense, or a third option emerges: a transferable attack.

To understand transferable attacks, consider the following game. Alice interacts with a player who claims to have a secure model for an instance of a learning task $\mathcal{D}, h$, where $\mathcal{D}$ is the data distribution and h is the ground truth. Alice sends queries and observes the responses. She wins if she can generate queries that (i) cause significant errors and (ii) remain indistinguishable from samples drawn from $\mathcal{D}$. Whether she succeeds depends on the computational and data resources available to her and the other player. If Alice can defeat *any* equally-resourced player, we call her queries a *Transferable Attack*. Intuitively, the more challenging a query becomes, the easier it should be to detect—but surprisingly, we show that transferable attacks do exist. Specifically, we prove:

- The existence of a **Transferable Attack** as defined above. Our construction uses cryptographic techniques, particularly Fully Homomorphic Encryption (FHE) (Gentry, 2009). This establishes that Transferable Attacks form the third fundamental option in the trade-off.

- That any learning task supporting a Transferable Attack must be computationally complex. More precisely, Transferable Attacks imply the existence of a *cryptographic primitive*.

Finally, we give examples of learning task classes that provably support Watermarks and Adversarial Defenses thereby justifying our framework. Concretely: (1) We show that learning tasks with bounded Vapnik–Chervonenkis (VC) dimension allow **Adversarial Defenses** against all (even computationally unbounded) attackers, ruling out Transferable Attacks in these settings. (2) We construct a **Watermark** for a class of learning tasks with bounded VC-dimension. Interestingly, in this case, both a Watermark and an Adversarial Defense coexist.

Our findings reveal an inherent structure of the interplay between Watermarks, Adversarial Defenses, and Transferable Attacks. Rather than being independent concepts, these three phenomena span the entire space of possibilities—every learning task must allow for at least one of them.

2 MODELING

A key aspect of our formalization involves modeling Alice and Bob in a manner that takes computational resources into account. To achieve this we model the parties as families of circuits indexed by a size parameter n. This is standard in computational complexity theory. However, circuits are less standard, as compared to more loosely specified algorithms, in computational learning theory, but we require this additional level of granularity to achieve our results.

2.1 LEARNING

Definition 1 (*Learning Task (Informal)*)**.** Let $\{0,1\}^n$ be an input space and let $n \in \mathbb{N}$ be a parameter. A *learning task* $\mathbb{L}$ is defined as a sequence $\{\mathbb{L}_n\}_{n\in\mathbb{N}}$, where each $\mathbb{L}_n$ is a *fixed* distribution over pairs $(\mathcal{D}_n, h_n)$. Concretely, for each n, we draw $(\mathcal{D}_n, h_n) \sim \mathbb{L}_n$, where $\mathcal{D}_n$ is a distribution with domain $\{0,1\}^n$, and $h_n : \{0,1\}^n \to \{0,1\}$ is a *ground truth* labeling function.

To every model $f\colon \{0,1\}^n \to \{0,1\}$, we associate $\text{err}(f) := \mathbb{E}_{x\sim\mathcal{D}_n}[f(x) \neq h_n(x)]$. And for $q \in \mathbb{N}, \mathbf{x} \in (\{0,1\}^n)^q$, and predictions $\mathbf{y} \in \{0,1\}^q$, we define the empirical error to be: $\text{err}(\mathbf{x}, \mathbf{y}) := \frac{1}{q} \sum_{i\in[q]} \mathbb{1}_{\{h_n(x_i)\neq y_i\}}$.

Definition 2 (*Computationally Bounded Learnability (Informal)*)**.** Let $\epsilon, \delta : \mathbb{N} \to (0,1)$ be functions that specify the allowable error and confidence levels for each input size n, respectively. A learning task $\mathbb{L} = \{\mathbb{L}_n\}_{n\in\mathbb{N}}$ is said to be *learnable* to error $\epsilon(n)$ with confidence $1-\delta(n)$ and circuit complexity $S(n)$ if there exists a family of circuits $\{C_n\}_{n\in\mathbb{N}}$, where each circuit C_n has size at most $S(n)$, such that for every sufficiently large n, the following condition holds:

$$\mathbb{P}_{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n}\Big[\text{err}_{\mathcal{D}_n,h_n}(f_n) \leq \epsilon(n)\Big] \geq 1-\delta(n),$$

where $f_n : \{0,1\}^n \to \{0,1\}$ is the hypothesis computed by the circuit C_n when given sample access to $(\mathcal{D}_n, h_n)$, i.e., $f_n \leftarrow C_n$. In other words, with probability at least $1-\delta(n)$ over the choice of $(\mathcal{D}_n, h_n)$ drawn from $\mathbb{L}_n$, the circuit C_n successfully computes a function f_n that achieves an error rate of at most $\epsilon(n)$.

Definition 2 is very similar to the standard definition of efficient PAC learnability Kearns & Vazirani (1994). The main difference is that instead of defining 'efficient' as polynomial in n (and $1/\epsilon, 1/\delta$) we define it as implementable by a circuit of size given by a fixed function $S(n)$. The reason for this increased generality is that we need finer control over sizes than, e.g., polynomial or exponential (see Theorem 1 where the separation between two circuit families is $S(n)$ *versus* $\sqrt{S(n)}$). A second difference is that compared to the standard definition we bound the size of circuits Arora & Barak (2009), not the running time. Assuming a processing unit without parallel execution the two notions can be thought equivalent. Formal definitions and additional details can be found in Appendix C. In the rest of the main part of the paper, we will often omit the parameter n when it is clear context.

Connections to Existing Models of Learning Definition 1 represents a learner's prior knowledge as a distribution over pairs $(\mathcal{D}_n, h_n)$, where $\mathcal{D}_n$ is a distribution on the domain $\{0,1\}^n$ and $h_n : \{0,1\}^n \to \{0,1\}$ is the ground truth. This models a learning task as a distribution over both input distributions and hypotheses, assuming a realizable scenario with a fixed ground truth.

Our learning definition (Definition 2) is weaker than some standard notions yet stronger than other learnability concepts. Instead of requiring learnability for every domain distribution, it allows adaptation to a fixed distribution over $(\mathcal{D}_n, h_n)$ pairs, effectively incorporating a prior on these pairs. This is similar to the PAC-Bayes framework McAllester (1999), which uses a prior over hypotheses to achieve strong generalization bounds where standard PAC may fail. Extensions that include priors over distributions and sample sizes Rothfuss et al. (2020); Amit & Meir (2018) address meta- and transfer learning.

Unlike distribution-specific or restricted family settings Kalai et al. (2008); Feldman et al. (2006), our definition does not limit the underlying support. While standard PAC learning requires generalization across all domain distributions, it often fails to explain the performance of complex models like DNNs, as their rich hypothesis classes make standard PAC bounds ineffective Zhang et al. (2021); Nagarajan & Kolter (2019). Our definition aims to bridge this gap by providing a formal framework that aligns with contemporary practical learning scenarios.

2.2 Interaction

Alice and Bob will engage in interaction. To measure their computational resources, we require a specification of how the model f_n is transmitted between them. We assume that before the interaction starts they agree on a family of function classes $\mathcal{F} = \{\mathcal{F}_n\}_n$ as well as an encoding of them into messages of some length. This modeling implies that f_n are sent *white-box*. One example of such a family is the family of neural networks of a given architecture. See Appendix C for details.

2.3 Computational Indistinguishability

A crucial property of interest will be the indistinguishability of distributions. For a pair of distributions $\mathcal{D}^0, \mathcal{D}^1$ consider the following game between a sender and the distinguisher C: (1) The sender samples a bit $b \sim U(\{0,1\})$ and then draws a random sample $x \sim \mathcal{D}^b$, (2) C receives x and outputs $\hat{b} := C(x) \in \{0,1\}$. C wins if $\hat{b} = b$. We define the *advantage* of C for *distinguishing* $\mathcal{D}^0$ from $\mathcal{D}^1$ as

$$\mathbb{P}_{b\sim U(\{0,1\}),x\sim\mathcal{D}^b}[C(x) = b] = \frac{1}{2} + \gamma.$$

For a pair of families of distributions $\mathcal{D}^0 = \{\mathcal{D}^0_n\}_n, \mathcal{D}^1 = \{\mathcal{D}^1_n\}_n$, a function $\gamma : \mathbb{N} \to (0,1)$, and a size bound $S : \mathbb{N} \to \mathbb{N}$ we say $\mathcal{D}^0, \mathcal{D}^1$ are γ-*indistinguishable* for circuits of size S if for every n, every circuit C (also known as the distinguisher) of size $S(n)$ the *advantage* of C for *distinguishing* $\mathcal{D}^0_n$ from $\mathcal{D}^1_n$ is at most $\gamma(n)$.

3 Watermarks, Adversarial Defenses and Transferable Attacks

We present interactive protocols between a verifier and a prover, each specifically designed to address tasks such as *Watermarking*, *Adversarial Defense*, and *Transferable Attacks*. In our protocols, Alice ($\mathbf{A}$, verifier) and Bob ($\mathbf{B}$, prover) engage in interactive communication, with distinct roles depending on the specific task. Each protocol is defined with respect to a learning task $\mathbb{L}$, an error parameter $\varepsilon \in \left(0, \frac{1}{2}\right)$, and circuit size bounds $S_{\mathbf{A}}$ and $S_{\mathbf{B}}$, which are functions of n. A scheme is successful if the conditions of the protocols are satisfied. We denote the set of such circuits by $\textsc{Scheme}(\mathbb{L}, \varepsilon, S_{\mathbf{A}}(n), S_{\mathbf{B}}(n))$, where SCHEME refers to WATERMARK, DEFENSE, or TRANSFATTACK. For the formal version of the definitions and the protocols, please refer to Appendix D.

Definition 3 (*Watermark, informal*)**.**

A family of circuits $\{\mathbf{A}_n^{\text{WATERMARK}}\}_n$ of sizes $\{S_{\mathbf{A}}(n)\}_n$, implements a *backdoor-based watermarking scheme* for the learning task $\mathbb{L}$ with error parameter $\epsilon > 0$ if, for every sufficiently large n, an interactive protocol in which first $(\mathcal{D}_n, h_n) \sim \mathbb{L}_n$ and then $\mathbf{A}_n^{\text{WATERMARK}}$ computes a classifier $f\colon \{0,1\}^n \to \{0,1\}$ and a sequence of queries $\mathbf{x} \in (\{0,1\}^n)^q$, and a prover $\mathbf{B}_n$ outputs $\mathbf{y} = \mathbf{B}_n(f, \mathbf{x}) \in \{0,1\}^q$, satisfies the following properties:

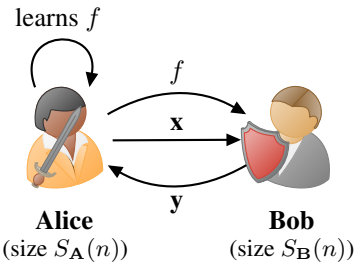


Figure 2: Schematic overview of the interaction between Alice and Bob in *Watermark* (Definition 3).

1. **Correctness:** f has low error, i.e., $\text{err}(f) \leq \epsilon$.
2. **Uniqueness:** There exists a prover $\mathbf{B}_n$ of size $S_{\mathbf{A}}(n)$, which provides low-error answers, such that $\text{err}(\mathbf{x}, \mathbf{y}) \leq 2\epsilon$.
3. **Unremovability:** For every prover $\mathbf{B}_n$ of size $S_{\mathbf{B}}(n)$, it holds that $\text{err}(\mathbf{x}, \mathbf{y}) > 2\epsilon$.
4. **Undetectability:** For every prover $\mathbf{B}_n$ of size $S_{\mathbf{B}}(n)$, the advantage of $\mathbf{B}_n$ in distinguishing the queries $\mathbf{x}$ generated by $\mathbf{A}_n^{\text{WATERMARK}}$ from random queries sampled from $\mathcal{D}_n^q$ is small.

Note that, due to *uniqueness*, we require that any $\mathbf{B}_n$ (Bob), who *did not use* f and trained a model f_{Scratch} using a specified procedure, must be accepted as a distinct model. This requirement is essential, as it mirrors real-world scenarios where independent models could have been trained if given enough resources.

Definition 4 (*Adversarial Defense, informal*)**.**

A family of circuits$\{\mathbf{B}_n^{\text{DEFENSE}}\}_n$ of sizes $\{S_{\mathbf{B}}(n)\}_n$, implements an *adversarial defense* for the learning task $\mathbb{L}$ with error parameter $\epsilon > 0$, if for every sufficiently large n, an interactive protocol in which first $(\mathcal{D}_n, h_n) \sim \mathbb{L}_n$ and then $\mathbf{B}_n^{\text{DEFENSE}}$ computes a classifier $f\colon \{0,1\}^n \to \{0,1\}$, while $\mathbf{A}_n$ replies with $\mathbf{x} = \mathbf{A}_n(f)$, where $\mathbf{x} \in \{0,1\}^{nq}$, and $\mathbf{B}_n^{\text{DEFENSE}}$ outputs $b = \mathbf{B}_n^{\text{DEFENSE}}(f, \mathbf{x}) \in \{0,1\}$, satisfies the following properties:

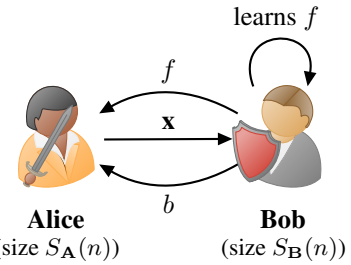


Figure 3: Schematic overview of the interaction between Alice and Bob in *Adversarial Defense* (Definition 4).

1. **Correctness:** f has low error, i.e., $\text{err}(f) \leq \epsilon$.
2. **Completeness:** When $\mathbf{x} \sim \mathcal{D}_n^q$, then $b = 0$.
3. **Soundness:** For every $\mathbf{A}_n$ of size $S_{\mathbf{A}}(n)$, we have $\text{err}(\mathbf{x}, f(\mathbf{x})) \leq 7\epsilon$ or $b = 1$.

The key requirement for a successful defense is the ability to *detect when it is being tested*. To bypass the defense, an $\mathbf{A}_n$ (Alice) must provide samples that are both *adversarial*, causing the classifier to make mistakes, and *indistinguishable* from samples drawn from the data distribution $\mathcal{D}_n$.

Definition 5 (*Transferable Attack, informal*)**.**

A family of circuits $\{\mathbf{A}_n^{\text{TRANSFATTACK}}\}_n$ of sizes $\{S_{\mathbf{A}}(n)\}_n$, implements a *transferable attack* for the learning task $\mathbb{L}$ with error parameter $\epsilon > 0$, if for every sufficiently large n, an interactive protocol in which first $(\mathcal{D}_n, h_n) \sim \mathbb{L}_n$ and then $\mathbf{A}_n^{\text{TRANSFATTACK}}$ computes $\mathbf{x} \in \{0,1\}^{nq}$ and $\mathbf{B}_n$ outputs $\mathbf{y} = \mathbf{B}_n(\mathbf{x}) \in \{0,1\}^q$ satisfies the following properties:

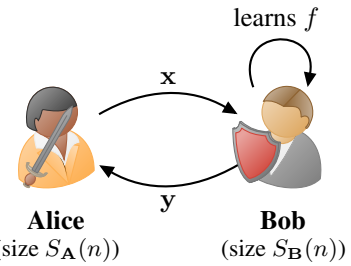


Figure 4: Schematic overview of the interaction between Alice and Bob in *Adversarial Defense* (Definition 5).

1. **Correctness:** Size $S_{\mathbf{B}}(n)$ is sufficient to learn a classifier of low-error, $\text{err}(f) \leq \epsilon$.
2. **Transferability:** For every prover $\mathbf{B}_n$ of size $S_{\mathbf{A}}(n)$, we have $\text{err}(\mathbf{x}, \mathbf{y}) > 2\epsilon$.
3. **Undetectability:** For every prover $\mathbf{B}_n$ of size $S_{\mathbf{B}}(n)$, the advantage of $\mathbf{B}_n$ in distinguishing the queries $\mathbf{x}$ generated by $\mathbf{A}_n^{\text{TRANSFATTACK}}$ from random queries sampled from $\mathcal{D}_n^q$ is small.

4 MAIN RESULT

We are ready to state an informal version of our main theorem. Please refer to Theorem 5 for the details and full proof. The key idea is to define a *zero-sum game* between $\mathbf{A}_n$ (Alice) and $\mathbf{B}_n$ (Bob), for every n, where the actions of each player are all possible circuits that can be realized with size $S_{\mathbf{A}}(n)$ and $S_{\mathbf{B}}(n)$. Here, zero-sum games are not a modeling choice but a proof strategy, as they allow us to analyze the complementary nature of attacks on watermarks and adversarial defenses with clean mathematical guarantees. Notably, this game is finite, but there are exponentially many such actions for each player. We rely on some key properties of such large zero-sum games (Lipton & Young, 1994b) to argue about our main result. The formal statement and proof is deferred to Appendix E.

Theorem 1 (*Main Theorem, informal*)**.** *For every $\epsilon \in \left(0, \frac{1}{2}\right), S : \mathbb{N} \to \mathbb{N}$ and learning task $\mathbb{L}$ learnable to error ϵ with high confidence with circuit complexity $S(n)$, at least one of these three exists*[1]*:*

$$\textsc{Watermark}\left(\mathbb{L}, \epsilon, S(n), o\left(\frac{\sqrt{S(n)}}{\log(S(n))}\right)\right),$$

$$\textsc{Defense}\left(\mathbb{L}, \epsilon, o\left(\frac{\sqrt{S(n)}}{\log(S(n))}\right), O(S(n))\right),$$

$$\textsc{TransfAttack}\Big(\mathbb{L}, \epsilon, S(n), S(n)\Big).$$

Proof (Sketch). The intuition of the proof relies on the complementary nature of Definitions 3 and 4. Specifically, every attempt to remove a fixed Watermark can be transformed to a potential Adversarial Defense, and vice versa. We define a zero-sum game $\mathcal{G}$ between circuits for watermarking $\mathbf{A}_n$ and circuits attempting to remove a watermark $\mathbf{B}_n$. The set of (pure) strategies of each player are all possible circuits that can be realized with size $S_{\mathbf{A}}(n)$ and $S_{\mathbf{B}}(n)$, and the payoff is determined by the probability that the errors and rejections meet specific requirements. It is well known that this two-player zero-sum game admits a Nash equilibrium (NE) and the value of the game is unique v. Neumann (1928).

Let $\{\mathbf{A}_n^{\text{NASH}}\}_n$ and $\{\mathbf{B}_n^{\text{NASH}}\}_n$ be the NE strategies of Alice and Bob respectively. For each $n \in \mathbb{N}$, a careful analysis shows that depending on the value of the game, we have a Watermark, an Adversarial Defense, or a Transferable Attack. In the first case, where the expected payoff at the Nash equilibrium is greater than a threshold, we show there is an Adversarial Defense. As an illustration, consider some $n \in \mathbb{N}$, for which we define $\mathbf{B}_n^{\text{DEFENSE}}$ as follows. $\mathbf{B}_n^{\text{DEFENSE}}$ first learns a low-error classifier f, then sends f to the party that is attacking the Defense, then receives queries $\mathbf{x}$, and simulates $(\mathbf{y}, b) = \mathbf{B}_n^{\text{NASH}}(f, \mathbf{x})$. The bit $b = 1$ if $\mathbf{B}_n^{\text{NASH}}$ thinks it is attacked. Finally, $\mathbf{B}_n^{\text{DEFENSE}}$ replies with $b' = 1$ if $b = 1$, and if $b = 0$ it replies with $b' = 1$ if the fraction of queries on which $f(\mathbf{x})$ and $\mathbf{y}$ differ is high. Careful analysis shows $\mathbf{B}_n^{\text{DEFENSE}}$ is an Adversarial Defense.

In the second case, where the expected payoff at the Nash equilibrium is below the threshold, we have either a Watermark or a Transferable Attack. The reason that there are two cases is due to the details of the definition of the payoffs. The full proof can be found in Appendix E. □

5 TRANSFERABLE ATTACKS AND CRYPTOGRAPHY

In this section, we show that tasks with Transferable Attacks exist. To construct such examples, we use cryptographic tools. But importantly, the fact that we use cryptography is not coincidental. As a second result of this section, we show that every learning task with a Transferable Attack *implies* a certain cryptographic primitive. One can interpret this as showing that Transferable Attacks exist only for *complex learning tasks*, in the sense of computational complexity theory.

[1]We remark that formally the existence does not hold for all sufficiently large n but only with some 'frequency'. See Theorem 5 for a formal statement.

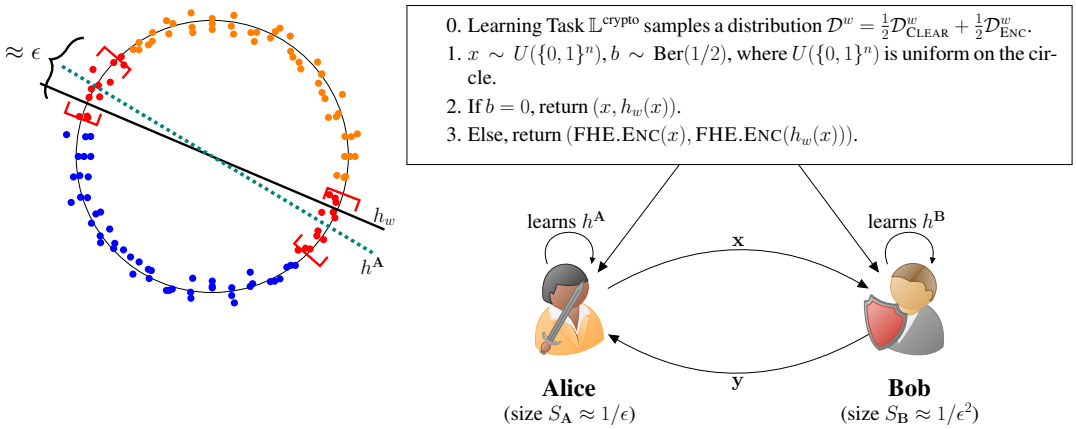


Figure 5: The left part of the figure represents a *Lines on Circle Learning Task* $\mathbb{L}^\circ$ with a ground truth function denoted by h_w. On the right, we define a *cryptography-augmented* learning task derived from $\mathbb{L}^\circ$. In its distribution, a "clear" or an "encrypted" sample is observed with equal probability. Given their respective times, both $\mathbf{A}$ and $\mathbf{B}$ are able to learn a low-error classifier $h^{\mathbf{A}}$, $h^{\mathbf{B}}$ respectively, by learning only on the *clear samples*. $\mathbf{A}$ is able to compute a Transferable Attack by computing an encryption of a point close to the decision boundary of her classifier $h^{\mathbf{A}}$.

5.1 A Cryptography-based Task with a Transferable Attack

Next, we give an example of a cryptography-based learning task with a Transferable Attack. The following is an informal statement of the formal version (Theorem 7) given in Appendix G.

Theorem 2 (*Transferable Attack for a Cryptography-based Learning Task, informal*)**.** *There exists a learning task* $\mathbb{L}^{crypto}$ *and* $\mathbf{A}$ *such that for all sufficiently small* ϵ

$$\mathbf{A} \in \text{TRANSFATTACK}\left(\mathbb{L}^{crypto}, \epsilon, S_{\mathbf{A}} \approx \frac{1}{\epsilon}, S_{\mathbf{B}} = \Omega\left(\frac{1}{\epsilon^2}\right)\right).$$

Moreover, $\mathbb{L}^{crypto}$ *is such that for every* ϵ*, circuit size of approximately* $\frac{1}{\epsilon}$ *(and* $O\left(\frac{1}{\epsilon}\right)$ *samples) is enough, and* $\Omega\left(\frac{1}{\epsilon}\right)$ *samples (and in particular circuit size) is necessary to learn a classifier of error* ϵ*.*

Notably, the parameters are set so that $\mathbf{A}$ (the party computing $\mathbf{x}$) has a *smaller* circuit size than $\mathbf{B}$ (the party computing $\mathbf{y}$), specifically $\approx 1/\epsilon$ compared to $\Omega(1/\epsilon^2)$. Furthermore, because of the cryptography tools used, this is a setting where a single input maps to multiple outputs, which deviates away from the setting of classification learning tasks considered in Theorem 1.

Proof (Sketch). We start with a definition of a learning task that will be later augmented with a cryptographic tool to produce $\mathbb{L}^{\text{crypto}}$.

Lines on Circle Learning Task $\mathbb{L}^\circ$ (Figure 5). We associate the input space $\{0,1\}^n$ with vertices of a 2^n regular polygon inscribed in $\{x \in \mathbb{R}^2 \mid \|x\|_2 = 1\}$. Let $\mathcal{H} := \{h_w \mid w \in \mathbb{R}^2, \|w\|_2 = 1\}$, where $h_w(x) := \text{sgn}(\langle w, x\rangle)$. Let $\mathbb{L}^\circ$ be a distribution corresponding to the following process: sample $h_w \sim U(\mathcal{H})$, return $(U(\{0,1\}^n), h_w)$. Additionally, let $B_w(\alpha) := \{x \in \{0,1\}^n \mid |\measuredangle(x, w)| \leq \alpha\}$ denote the set of points within an angular distance up to α to w.

Fully Homomorphic Encryption (FHE) (Appendix F). FHE (Gentry, 2009) allows for computation on encrypted data *without* decrypting it. An FHE scheme allows to encrypt x via an efficient procedure $e_x = \text{FHE.ENC}(x)$, so that later, for any algorithm C, it is possible to run C on x *homomorphically*. More concretely, it is possible to produce an encryption of the result of running C on x, i.e., $e_{C,x} := \text{FHE.EVAL}(C, e_x)$. Finally, there is a procedure FHE.DEC that, when given a *secret key* sk, can decrypt $e_{C,x}$, i.e., $y := \text{FHE.DEC}(\text{sk}, e_{C,x})$, where y is the result of running C on x. Crucially, encryptions of any two messages are indistinguishable for all efficient adversaries.

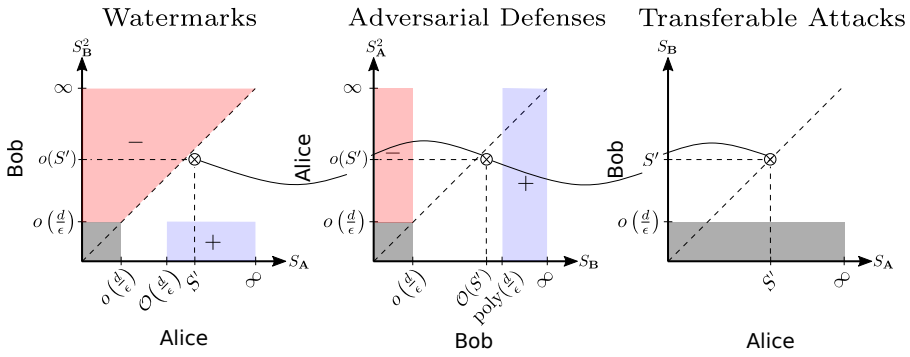


Figure 6: Overview of the taxonomy of learning tasks, illustrating the presence of Watermarks, Adversarial Defenses, and Transferable Attacks for learning tasks of bounded VC dimension. The axes represent the size bound for the parties in the corresponding schemes. The blue regions depict positive results, the red negative, and the gray regimes of parameters which are not of interest. See Lemma 3 and 4 for details about blue regions. The curved line represents a potential application of Theorem 1, which says that at least one of the three points should be blue.

Cryptography-based Learning Task $\mathbb{L}^{\text{crypto}}$ (Figure 5). $\mathbb{L}^{\text{crypto}}$ is derived from *Lines on Circle Learning Task* $\mathbb{L}^{\circ}$. $\mathbb{L}^{\text{crypto}}$ corresponds to the following process: $w \sim U(\{w \in \mathbb{R}^2 \mid \|w\|_2 = 1\})$, return the distribution $\mathcal{D}^w$, which is an equal mixture of two parts $\mathcal{D}^w = \frac{1}{2}\mathcal{D}^w_{\text{CLEAR}} + \frac{1}{2}\mathcal{D}^w_{\text{ENC}}$. The first part, i.e., $\mathcal{D}^w_{\text{CLEAR}}$, is equal to $x \sim U(\{0,1\}^n)$ with the correct label $y = h_w(x)$. The second part, i.e., $\mathcal{D}^w_{\text{ENC}}$, is equal to $x' \sim U(\{0,1\}^n), y' = h_w(x'), (x, y) = (\text{FHE.ENC}(x'), \text{FHE.ENC}(y'))$,[2] which can be thought of as $\mathcal{D}^w_{\text{CLEAR}}$ under an encryption. See Figure 5 for a visual representation. Note that we omitted the size parameter n for simplicity.

Transferable Attack (Figure 5). Consider the following attack strategy $\mathbf{A}$. First, $\mathbf{A}$ collects $O(1/\epsilon)$ samples from the distribution $\mathcal{D}^w_{\text{CLEAR}}$ and learns a classifier $h^{\mathbf{A}}_{w'} \in \mathcal{H}$ that is consistent with these samples. Since the VC-dimension of $\mathcal{H}$ is 2, the hypothesis $h^{\mathbf{A}}_{w'}$ has error at most ϵ with high probability.[3] Next, $\mathbf{A}$ samples a point x_{BND} uniformly at random from a region close to the decision boundary of $h^{\mathbf{A}}_{w'}$, i.e., $x_{\text{BND}} \sim U(B_{w'}(\epsilon))$. Finally, with equal probability, $\mathbf{A}$ sets as an attack $\mathbf{x}$ either $\text{FHE.ENC}(x_{\text{BND}})$ or a uniformly random point $\mathcal{D}^w_{\text{CLEAR}} = U(\{0,1\}^n)$. We claim[4] that $\mathbf{x}$ satisfies the properties of a Transferable Attack.

Since $h^{\mathbf{A}}_{w'}$ has a low error with high probability, x_{BND} is a uniformly random point from an arc containing the boundary of h_w (see Figure 5). The circuit size of $\mathbf{B}$ is upper-bounded by $\Omega(1/\epsilon^2)$, meaning it can only learn a classifier with error $\gtrapprox 10\epsilon^2$ (see Lemma 1 for details). $\mathbf{B}$'s can only learn (Lemma 1) a classifier of error, $\gtrapprox 10\epsilon^2$. Taking these two facts together, we expect $\mathbf{B}$ to misclassify x' with probability $\approx \frac{1}{2} \cdot \frac{10\epsilon^2}{\epsilon} = 5\epsilon > 2\epsilon$, where the factor $\frac{1}{2}$ takes into account that we send an encrypted sample only half of the time. This implies *transferability*. Note that $\mathbf{x}$ is encrypted with the same probability as in the original distribution because we send $\text{FHE.ENC}(x_{\text{BND}})$ and a uniformly random $\mathbf{x} \sim \mathcal{D}^w_{\text{CLEAR}} = U(\{0,1\}^n)$ with equal probability. Crucially, $\text{FHE.ENC}(x_{\text{BND}})$ is indistinguishable, for efficient adversaries, from $\text{FHE.ENC}(x)$ for any other $x \in \{0,1\}^n$. This follows from the security of the FHE scheme. Consequently, *undetectability* holds. □

Next, we show that a Transferable Attack for any task implies a *cryptographic primitive*.

5.2 Tasks with Transferable Attacks Imply Cryptography

EFID Pairs. In cryptography, an *EFID pair* (Goldreich, 1990) is a pair of ensembles of distributions $\mathcal{D}^0, \mathcal{D}^1$, that are **E**fficiently samplable, statistically **F**ar, and computationally **I**ndistinguishable. By

[2] Note that because FHE encryption is probabilistic there are many valid answers for a given x.

[3] $\mathbf{A}$ can also evaluate $h^{\mathbf{A}}_{w'}$ homomorphically (i.e., run FHE.EVAL) on $\text{FHE.ENC}(x)$ to obtain $\text{FHE.ENC}(y)$ of error ϵ on $\mathcal{D}^w_{\text{ENC}}$ also. This means that $\mathbf{A}$ is able to learn a low-error classifier on $\mathcal{D}^w$.

[4] In this proof sketch, we set $q = 1$, i.e., $\mathbf{A}$ sends only one x to $\mathbf{B}$. This is not true for the formal scheme.

a seminal result (Goldreich, 1990), we know that the existence of EFID pairs is equivalent to the existence of *Pseudorandom Generators* (PRG), which can be used for tasks including encryption and key generation (Goldreich, 1990), which makes EFID pairs a useful primitive. We consider a slight modification of the standard definition of EFID pairs, where instead of defining security to hold against polynomial time adversaries we do it for a fixed size bound function. More concretely, for two size bounds $S, S' : \mathbb{N} \to \mathbb{N}$ we call a pair of ensembles of distributions $(\mathcal{D}^0, \mathcal{D}^1)$ an (S, S')-EFID pair if for every n (i) $\mathcal{D}^0_n, \mathcal{D}^1_n$ are samplable by circuits of size $S(n)$, (ii) $\mathcal{D}^0_n, \mathcal{D}^1_n$ are statistically far, (iii) $\mathcal{D}^0_n, \mathcal{D}^1_n$ are indistinguishable for circuits of size $S'(n)$.

Tasks with Transferable Attacks imply EFID Pairs. The second result of this section shows that any task with a Transferable Attack implies the existence of a type of EFID pair. This guarantees that any learning task with a Transferable Attack has to be computationally complex. The proof is deferred to Appendix H.

Theorem 3 (*Tasks with Transferable Attacks imply EFID pairs, informal*)**.** *For every $\epsilon \in (0, 1), S_{\mathbf{A}}, S_{\mathbf{B}} : \mathbb{N} \to \mathbb{N}, S_{\mathbf{A}} \leq S_{\mathbf{B}}$, every learning task $\mathbb{L}$ learnable to error ϵ with high confidence and circuit complexity $S_{\mathbf{A}}$ if there exists* $\textsc{TransAttack}(\mathbb{L}, \epsilon, S_{\mathbf{A}}, S_{\mathbf{B}})$ *then there exists an $(S_{\mathbf{A}}, S_{\mathbf{B}})$-EFID pair.*

We note that it is unclear if the existence of EFID-pairs guaranteed by Theorem 3 implies PRGs because the sampling of $\mathcal{D}^0, \mathcal{D}^1$ requires oracle access to $\mathbb{L}$. Therefore, the standard construction of PRGs from EFID pairs does not automatically transfer.

6 Tasks with Watermarks and Adversarial Defenses

As the final pair of results we give examples of tasks with Watermarks and Adversarial Defenses. In the first example, we show that hypothesis classes of polynomially bounded VC-dimension have polynomial-sized Adversarial Defenses against all attackers. The second example is a learning task of polynomially bounded VC-dimension that has a Watermark, which is secure against fast adversaries. These lemmas illustrate why the upper bounds on the sizes of $\mathbf{A}$ and $\mathbf{B}$ are crucial parameters. See also Figure 6 for a visual representation of these results. Lemmas are formally stated and proven in the Appendix I and Appendix J.

7 Beyond Classification

Inspired by Theorem 2, we conjecture a possibility of generalizing our results to generative learning tasks. Instead of a ground truth function, one could consider a ground truth quality oracle Q, which measures the quality of every input and output pair. This model introduces new phenomena *not* present in the case of classification. For example, the task of *generation*, i.e., producing a high-quality output y on input x, is decoupled from the task of *verification*, i.e., evaluating the quality of y as output for x. By decoupled, we mean that there is no clear formal reduction from one task to the other. Conversely, for classification, where the space of possible outputs is small, the two tasks are equivalent. Without going into details, this decoupling is the reason why the proof of Theorem 1 does not automatically transfer to the generative case.

This decoupling introduces new complexities, but it also suggests that considering new definitions may be beneficial. For example, because generation and verification are equivalent for classification tasks, we allowed neither $\mathbf{A}$ nor $\mathbf{B}$ access to h, as it would trivialize the definitions. However, a modification of the Definition 8 (Watermark), where access to Q is given to $\mathbf{B}$ could be investigated in the generative case. Interestingly, such a setting was considered in (Zhang et al., 2023), where access to Q was crucial for mounting a provable attack on "all" strong watermarks. As we alluded to earlier, Theorem 2 can be seen as an example of a task, where generation is easy but verification is hard – the opposite to what Zhang et al. (2023) posits. We hope that careful formalizations of the interaction and capabilities of all parties might give insights into not only the schemes considered in this work, but also problems like weak-to-strong generalization (Burns et al., 2024) or scalable oversight (Brown-Cohen et al., 2023).

IMPACT STATEMENT

In contrast to years of adversarial robustness research (Carlini, 2024), we conjecture that for learning tasks encountered in safety-critical regimes, an Adversarial Defense *will* exist in the future. Three pieces of evidence support this contrarian belief. (i) Theorem 1, (ii) in the security-critical scenarios for Watermarks, the security should hold even against strong defenders. Formally this suggests $S_{\mathbf{B}}$ should approach $S_{\mathbf{A}}$ ensuring that watermark verification remains effective despite adversarial attempts to remove it. (iii) Transferable Attacks imply cryptographic primitives (Theorem 8), which suggests that the existence of highly transferable adversarial examples may be constrained by practical cryptographic limitations. While our work advances the theoretical understanding of the trade-off between adversarial robustness and backdoor-based watermarks, it also raises fundamental questions about the limits of these techniques. How much robustness can be achieved while maintaining verifiability? Conversely, to what extent can backdoor-based watermarks remain effective without introducing exploitable vulnerabilities? Addressing these questions will be crucial for ensuring the security and reliability of machine learning models in high-stakes applications.

ACKNOWLEDGEMENT

This research was partially supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – The Berlin Mathematics Research Center MATH+ (EXC-2046/1, project ID: 390685689).

REFERENCES

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)*, pp. 1615–1631, 2018.

Noga Amit, Shafi Goldwasser, Orr Paradise, and Guy Rothblum. Models that prove their own correctness. *arXiv preprint arXiv:2405.15722*, 2024.

Ron Amit and Ron Meir. Meta-learning by adjusting priors based on extended pac-bayes theory. In *International Conference on Machine Learning*, pp. 205–214. PMLR, 2018.

Maksym Andriushchenko, Francesco Croce, and Nicolas Flammarion. Jailbreaking leading safety-aligned llms with simple adaptive attacks, 2024.

Cem Anil, Guodong Zhang, Yuhuai Wu, and Roger Grosse. Learning to give checkable answers with prover-verifier games. *arXiv preprint arXiv:2108.12099*, 2021.

Sanjeev Arora and Boaz Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, USA, 1st edition, 2009. ISBN 0521424267.

Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (leveled) fully homomorphic encryption without bootstrapping. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ITCS '12, pp. 309–325, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450311520. doi: 10.1145/2090236.2090262. URL https://doi.org/10.1145/2090236.2090262.

Jonah Brown-Cohen, Geoffrey Irving, and Georgios Piliouras. Scalable ai safety via doubly-efficient debate. *arXiv preprint arXiv:2311.14125*, 2023.

Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, Ilya Sutskever, and Jeffrey Wu. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 4971–5012. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/burns24b.html.

Nicholas Carlini. Yet another broken defense: How AI security continues to fail, 2024. URL https://nicholas.carlini.com/writing/2024/yet-another-broken-defense.html. Accessed: 2024-10-02.

Nicholas Carlini, Milad Nasr, Christopher A. Choquette-Choo, Matthew Jagielski, Irena Gao, Anas Awadalla, Pang Wei Koh, Daphne Ippolito, Katherine Lee, Florian Tramèr, and Ludwig Schmidt. Are aligned neural networks adversarially aligned? *ArXiv*, abs/2306.15447, 2023. URL https://api.semanticscholar.org/CorpusID:259262181.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J. Pappas, and Eric Wong. Jailbreaking black box large language models in twenty queries, 2023.

Jiefeng Chen, Yang Guo, Xi Wu, Tianqi Li, Qicheng Lao, Yingyu Liang, and Somesh Jha. Towards adversarial robustness via transductive learning. *arXiv preprint arXiv:2106.08387*, 2021.

Miranda Christ, Sam Gunn, and Or Zamir. Undetectable watermarks for language models. *arXiv preprint arXiv:2306.09194*, 2023.

Paul Christiano, Jacob Hilton, Victor Lecomte, and Mark Xu. Backdoor defense, learnability and obfuscation. *arXiv preprint arXiv:2409.03077*, 2024.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1310–1320. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/cohen19c.html.

Anne Condon, Joan Feigenbaum, Carsten Lund, and Peter Shor. Probabilistically checkable debate systems and approximation algorithms for pspace-hard functions. In *Proceedings of the twenty-fifth annual ACM symposium on Theory of Computing*, pp. 305–314, 1993.

Bita Darvish Rouhani, Huili Chen, and Farinaz Koushanfar. Deepsigns: An end-to-end watermarking framework for ownership protection of deep neural networks. In *Proceedings of the twenty-fourth international conference on architectural support for programming languages and operating systems*, pp. 485–497, 2019.

Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4307–4316, 2019. URL https://api.semanticscholar.org/CorpusID:102350868.

Yousof Erfani, Ramin Pichevar, and Jean Rouat. Audio watermarking using spikegram and a two-dictionary approach. *IEEE Transactions on Information Forensics and Security*, 12(4):840–852, 2017. doi: 10.1109/TIFS.2016.2636094.

Jon Feldman, Rocco A Servedio, and Ryan O'Donnell. Pac learning axis-aligned mixtures of gaussians with no separation assumption. In *Learning Theory: 19th Annual Conference on Learning Theory, COLT 2006, Pittsburgh, PA, USA, June 22-25, 2006. Proceedings 19*, pp. 20–34. Springer, 2006.

Pierre Fernandez, Guillaume Couairon, Hervé Jégou, Matthijs Douze, and Teddy Furon. The stable signature: Rooting watermarks in latent diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22466–22477, 2023.

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.

A. Gammerman, V. Vovk, and V. Vapnik. Learning by transduction. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, UAI'98, pp. 148–155, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc. ISBN 155860555X.

Craig Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing*, STOC '09, pp. 169–178, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605585062. doi: 10.1145/1536414.1536440. URL https://doi.org/10.1145/1536414.1536440.

Oded Goldreich. A note on computational indistinguishability. *Information Processing Letters*, 34(6):277–281, 1990. ISSN 0020-0190. doi: https://doi.org/10.1016/0020-0190(90)90010-U. URL https://www.sciencedirect.com/science/article/pii/002001909090010U.

S Goldwasser and M Sipser. Private coins versus public coins in interactive proof systems. In *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing*, STOC '86, pp. 59–68, New York, NY, USA, 1986. Association for Computing Machinery. ISBN 0897911938. doi: 10.1145/12130.12137. URL https://doi.org/10.1145/12130.12137.

S Goldwasser, S Micali, and C Rackoff. The knowledge complexity of interactive proof-systems. In *Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing*, STOC '85, pp. 291–304, New York, NY, USA, 1985. Association for Computing Machinery. ISBN 0897911512. doi: 10.1145/22145.22178. URL https://doi.org/10.1145/22145.22178.

Shafi Goldwasser, Yael Kalai, Raluca Ada Popa, Vinod Vaikuntanathan, and Nickolai Zeldovich. Reusable garbled circuits and succinct functional encryption. In *Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing*, STOC '13, pp. 555–564, New York, NY, USA, 2013. Association for Computing Machinery. ISBN 9781450320290. doi: 10.1145/2488608.2488678. URL https://doi.org/10.1145/2488608.2488678.

Shafi Goldwasser, Adam Tauman Kalai, Yael Tauman Kalai, and Omar Montasser. Beyond perturbations: Learning guarantees with arbitrary adversarial test examples. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Shafi Goldwasser, Michael P. Kim, Vinod Vaikuntanathan, and Or Zamir. Planting undetectable backdoors in machine learning models. *ArXiv*, abs/2204.06974, 2022. URL https://api.semanticscholar.org/CorpusID:248177888.

Shafi Goldwasser, Jonathan Shafer, Neekon Vafa, and Vinod Vaikuntanathan. Oblivious defense in ml models: Backdoor removal without detection, 2024. URL https://arxiv.org/abs/2411.03279.

Chenxi Gu, Chengsong Huang, Xiaoqing Zheng, Kai-Wei Chang, and Cho-Jui Hsieh. Watermarking pre-trained language models with backdooring. *arXiv preprint arXiv:2210.07543*, 2022.

Geoffrey Irving, Paul Christiano, and Dario Amodei. Ai safety via debate, 2018. URL https://arxiv.org/abs/1805.00899.

Zhengyuan Jiang, Jinghuai Zhang, and Neil Zhenqiang Gong. Evading watermark based detection of ai-generated content. *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, 2023. URL https://api.semanticscholar.org/CorpusID:258557682.

Adam Tauman Kalai, Adam R Klivans, Yishay Mansour, and Rocco A Servedio. Agnostically learning halfspaces. *SIAM Journal on Computing*, 37(6):1777–1805, 2008.

Michael J. Kearns and Umesh V. Vazirani. *An introduction to computational learning theory*. MIT Press, Cambridge, MA, USA, 1994. ISBN 0262111934.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 17061–17084. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/kirchenbauer23a.html.

Jan Hendrik Kirchner, Yining Chen, Harri Edwards, Jan Leike, Nat McAleese, and Yuri Burda. Prover-Verifier Games improve legibility of LLM outputs, 2024. URL https://arxiv.org/abs/2407.13692.

Rohith Kuditipudi, John Thickstun, Tatsunori Hashimoto, and Percy Liang. Robust distortion-free watermarks for language models. *CoRR*, abs/2307.15593, 2023. doi: 10.48550/ARXIV.2307.15593. URL https://doi.org/10.48550/arXiv.2307.15593.

Peixuan Li, Pengzhou Cheng, Fangqi Li, Wei Du, Haodong Zhao, and Gongshen Liu. Plmmark: a secure and robust black-box watermarking framework for pre-trained language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 14991–14999, 2023.

Richard J. Lipton and Neal E. Young. Simple strategies for large zero-sum games with applications to complexity theory. In *Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing*, STOC '94, pp. 734–740, New York, NY, USA, 1994a. Association for Computing Machinery. ISBN 0897916638. doi: 10.1145/195058.195447. URL https://doi.org/10.1145/195058.195447.

Richard J Lipton and Neal E Young. Simple strategies for large zero-sum games with applications to complexity theory. In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pp. 734–740, 1994b.

Chang Liu, Jie Zhang, Han Fang, Zehua Ma, Weiming Zhang, and Nenghai Yu. Dear: A deep-learning-based audio re-recording resilient watermarking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 13201–13209, 2023.

Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. *arXiv preprint arXiv:1611.02770*, 2016.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=rJzIBfZAb.

David A McAllester. Pac-bayesian model averaging. In *Proceedings of the twelfth annual conference on Computational learning theory*, pp. 164–170, 1999.

Anay Mehrotra, Manolis Zampetakis, Paul Kassianik, Blaine Nelson, Hyrum Anderson, Yaron Singer, and Amin Karbasi. Tree of attacks: Jailbreaking black-box llms automatically, 2024.

Erwan Le Merrer, Patrick Pérez, and Gilles Trédan. Adversarial frontier stitching for remote neural network watermarking. *Neural Computing and Applications*, 32:9233 – 9244, 2017. URL https://api.semanticscholar.org/CorpusID:11008755.

Omar Montasser, Steve Hanneke, and Nathan Srebro. Vc classes are adversarially robustly learnable, but only improperly. In Alina Beygelzimer and Daniel Hsu (eds.), *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pp. 2512–2530. PMLR, 25–28 Jun 2019. URL https://proceedings.mlr.press/v99/montasser19a.html.

Omar Montasser, Steve Hanneke, and Nathan Srebro. Transductive robust learning guarantees. In *International Conference on Artificial Intelligence and Statistics*, pp. 11461–11471. PMLR, 2022.

Yuki Nagai, Yusuke Uchida, Shigeyuki Sakazawa, and Shin'ichi Satoh. Digital watermarking for deep neural networks. *International Journal of Multimedia Information Retrieval*, 7:3–16, 2018.

Vaishnavh Nagarajan and J Zico Kolter. Uniform convergence may be unable to explain generalization in deep learning. *Advances in Neural Information Processing Systems*, 32, 2019.

Ryota Namba and Jun Sakuma. Robust watermarking of neural network with exponential weighting. *Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security*, 2019. URL https://api.semanticscholar.org/CorpusID:58028915.

Noam Nisan. Pseudorandom generators for space-bounded computations. In *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pp. 204–212, 1990.

Zhenxing Niu, Yuyao Sun, Qiguang Miao, Rong Jin, and Gang Hua. Towards unified robustness against both backdoor and adversarial attacks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):7589–7605, 2024. doi: 10.1109/TPAMI.2024.3392760.

Wenjun Peng, Jingwei Yi, Fangzhao Wu, Shangxi Wu, Bin Zhu, Lingjuan Lyu, Binxing Jiao, Tong Xu, Guangzhong Sun, and Xing Xie. Are you copying my model? protecting the copyright of large language models for eaas via backdoor watermark. *arXiv preprint arXiv:2305.10036*, 2023.

Aditi Raghunathan, Jacob Steinhardt, and Percy Liang. Certified defenses against adversarial examples. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=Bys4ob-Rb.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In J. Platt, D. Koller, Y. Singer, and S. Roweis (eds.), *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007. URL https://proceedings.neurips.cc/paper_files/paper/2007/file/013a006f03dbc5392effeb8f18fda755-Paper.pdf.

Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pp. 84–93. ACM, 2005.

R. Rivest, L. Adleman, and M. Dertouzos. On data banks and privacy homomorphisms. In *Foundations of Secure Computation*, pp. 169–179, New York, NY, USA, 1978. Academic Press.

Jonas Rothfuss, Martin Josifoski, and Andreas Krause. Meta-learning bayesian neural network priors based on pac-bayesian theory. 2020.

Mingjie Sun, Siddhant Agarwal, and J Zico Kolter. Poisoned classifiers are not only backdoored, they are fundamentally broken. *arXiv preprint arXiv:2010.09080*, 2020.

Lue Tao, Lei Feng, Jinfeng Yi, Sheng-Jun Huang, and Songcan Chen. Better Safe than Sorry: Preventing Delusive Adversaries with Adversarial Training. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9781713845393.

Stuart A. Thompson Tiffany Hsu. Disinformation researchers raise alarms about a.i. chatbots. https://scottaaronson.blog/?p=6823, 2023. Accessed: March 2024.

Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. *Advances in neural information processing systems*, 33:1633–1645, 2020.

Yusuke Uchida, Yuki Nagai, Shigeyuki Sakazawa, and Shin'ichi Satoh. Embedding watermarks into deep neural networks. In *Proceedings of the 2017 ACM on international conference on multimedia retrieval*, pp. 269–277, 2017.

J v. Neumann. Zur theorie der gesellschaftsspiele. *Mathematische annalen*, 100(1):295–320, 1928.

Vinod Vaikuntanathan. Computing blindfolded: New developments in fully homomorphic encryption. In *Proceedings of the 2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, FOCS '11, pp. 5–16, Washington, DC, USA, 2011. IEEE Computer Society. ISBN 9780769543001. doi: 10.1109/FOCS.2011.98. URL https://doi.org/10.1109/FOCS.2011.98.

Stephan Wäldchen, Kartikey Sharma, Berkant Turan, Max Zimmer, and Sebastian Pokutta. Interpretability Guarantees with Merlin-Arthur Classifiers. In *International Conference on Artificial Intelligence and Statistics*, pp. 1963–1971. PMLR, 2024.

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. Jailbroken: How does llm safety training fail? *ArXiv*, abs/2307.02483, 2023. URL https://api.semanticscholar.org/CorpusID:259342528.

Yuxin Wen, Neel Jain, John Kirchenbauer, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Hard prompts made easy: Gradient-based discrete optimization for prompt tuning and discovery. In A. Oh, T. Neumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 51008–51025. Curran Associates, Inc., 2023a.

Yuxin Wen, John Kirchenbauer, Jonas Geiping, and Tom Goldstein. Tree-ring watermarks: Fingerprints for diffusion images that are invisible and robust. *ArXiv*, abs/2305.20030, 2023b. URL https://api.semanticscholar.org/CorpusID:258987524.

Cheng-Hsin Weng, Yan-Ting Lee, and Shan-Hung (Brandon) Wu. On the trade-off between adversarial and backdoor robustness. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 11973–11983. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/8b4066554730ddfaa0266346bdc1b202-Paper.pdf.

Yi-Hsuan Wu, Chia-Hung Yuan, and Shan-Hung Wu. Adversarial robustness via runtime masking and cleansing. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 10399–10409. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/wu20f.html.

Cihang Xie, Zhishuai Zhang, Jianyu Wang, Yuyin Zhou, Zhou Ren, and Alan Loddon Yuille. Improving transferability of adversarial examples with input diversity. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2725–2734, 2018. URL https://api.semanticscholar.org/CorpusID:3972825.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

Hanlin Zhang, Benjamin L. Edelman, Danilo Francati, Daniele Venturi, Giuseppe Ateniese, and Boaz Barak. Watermarks in the sand: Impossibility of strong watermarking for generative models. *arXiV*, abs/2311.04378, 2023. doi: 10.48550/ARXIV.2311.04378. URL https://doi.org/10.48550/arXiv.2311.04378.

Jialong Zhang, Zhongshu Gu, Jiyong Jang, Hui Wu, Marc Ph. Stoecklin, Heqing Huang, and Ian Molloy. Protecting intellectual property of deep neural networks with watermarking. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, ASIACCS '18, pp. 159–172, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450355766. doi: 10.1145/3196494.3196550. URL https://doi.org/10.1145/3196494.3196550.

Xuandong Zhao, Prabhanjan Ananth, Lei Li, and Yu-Xiang Wang. Provable robust watermarking for ai-generated text. *CoRR*, abs/2306.17439, 2023a. doi: 10.48550/ARXIV.2306.17439. URL https://doi.org/10.48550/arXiv.2306.17439.

Xuandong Zhao, Kexun Zhang, Yu-Xiang Wang, and Lei Li. Invisible image watermarks are provably removable using generative ai. 2023b. URL https://api.semanticscholar.org/CorpusID:259075167.

Yunqing Zhao, Tianyu Pang, Chao Du, Xiao Yang, Ngai-Man Cheung, and Min Lin. A recipe for watermarking diffusion models. *ArXiv*, abs/2303.10137, 2023c. URL https://api.semanticscholar.org/CorpusID:257622907.

Andy Zou, Zifan Wang, J. Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models. *ArXiv*, abs/2307.15043, 2023. URL https://api.semanticscholar.org/CorpusID:260202961.

A Related Work

This paper lies at the intersection of computational learning theory, interactive proof systems, and cryptography, while also drawing from empirical studies on adversarial robustness and backdoor-based watermarks. We review both theoretical and empirical advances that closely align with our research.

A.1 Backdoor Attacks and Adversarial Robustness

Backdoor attacks and adversarial robustness are closely related: the former embeds hidden behaviors triggered by subtle input changes, while the latter aims to ensure prediction stability against worst-case perturbations. Adversarial robustness research includes techniques like adversarial training (Madry et al., 2018), which improves resilience via adversarial examples, and certified defenses (Raghunathan et al., 2018), which provide provable guarantees within perturbation bounds. Methods such as randomized smoothing (Cohen et al., 2019) extend these guarantees. Theoretical work has shown robust learning is feasible for VC classes but only in the improper learning setup (Montasser et al., 2019).

Recent empirical works (Weng et al., 2020; Sun et al., 2020; Niu et al., 2024; Fowl et al., 2021; Tao et al., 2024) have explored trade-offs between adversarial robustness and backdoor attacks, primarily from an empirical perspective. Our work formalizes these trade-offs, introducing a theoretical framework that establishes the existence of at least one of three schemes—adversarial defenses, backdoor-based watermarks, or transferable attacks—for all learning tasks.

A.2 Backdoor-Based Watermarks

In black-box settings, where model auditors lack access to internal parameters, watermarking methods often involve embedding backdoors during training. Techniques by Adi et al. (2018) and Zhang et al. (2018) use crafted input patterns as triggers linked to specific outputs, enabling ownership verification by querying the model with these specific inputs. Advanced methods by Merrer et al. (2017) utilize adversarial examples, which are perturbed inputs that yield predefined outputs. Further enhancements by Namba & Sakuma (2019) focus on the robustness of watermarks, ensuring the watermark remains detectable despite model alterations or attacks.

In the domain of Natural Language Processing (NLP), backdoor-based watermarks have been studied for Pre-trained Language Models (PLMs)[5], as exemplified by works such as (Gu et al., 2022; Peng et al., 2023) and (Li et al., 2023). These approaches embed backdoors using rare or common word triggers, ensuring watermark robustness across downstream tasks and resistance to removal techniques like fine-tuning or pruning.

A.3 Undetectable Backdoors

A key related work by Goldwasser et al. (2022) shows how a learner can plant undetectable backdoors in any classifier. The authors propose two frameworks: one employing digital signature schemes (Goldwasser et al., 1985) to make backdoored models indistinguishable from the original to any computationally-bounded observer, and another using Random Fourier Features (RFF) (Rahimi & Recht, 2007), which remains undetectable even with full visibility of the model and training data.

In a concurrent and independent work, Christiano et al. (2024) introduce a defendability framework that formalizes the interaction between an attacker planting a backdoor and a defender tasked with detecting it. A major difference from our work, is that in their approach, the attacker chooses the distribution, whereas we keep the distribution fixed. This makes defendability in their model harder since the attacker has more control. However, in their framework, the backdoor trigger x^* is sampled from $\mathcal{D}$, so the attacker does not influence it. In contrast, our model allows the attacker to choose specific x's, making defendability in their model easier in this regard. Thus, the definitions are a

[5]We refer readers to Section 7, where we discuss the challenges and opportunities of applying our framework to self-supervised learning, highlighting how phenomena like the decoupling of generation and verification differ fundamentally from classification tasks.

priori incomparable. Another major difference is that our main result holds for *all* learning tasks, while their contributions hold only for restricted classes.

However, there are many interesting connections. They show that computationally unbounded defendability is equivalent to PAC learnability, while we, in a similar spirit, show an Adversarial Defense for all tasks with bounded VC-dimension. Using cryptographic tools, they show that the class of polynomial-size circuits is not efficiently defendable, while we use different cryptographic tools to give a Transferable Attack, which rules out a Defense.

A.4 Interactive Proof Systems in Machine Learning

Interactive Proof Systems (Goldwasser & Sipser, 1986) have recently gained considerable attention in machine learning for their ability to formalize and verify complex interactions between agents, models, or even human participants. A key advancement in this area is the introduction of *Prover-Verifier Games* (PVGs) (Anil et al., 2021), which employ a game-theoretic approach to guide learning agents towards decision-making with verifiable outcomes. Building on PVGs, Kirchner et al. (2024) enhance this framework to improve the legibility of Large Language Models (LLMs) outputs, making them more accessible for human evaluation. Similarly, Wäldchen et al. (2024) apply the prover-verifier setup to offer interpretability guarantees for classifiers. Extending these concepts, self-proving models Amit et al. (2024) introduce generative models that not only produce outputs but also generate proof transcripts to validate their correctness. In the context of AI safety, scalable *debate protocols* (Condon et al., 1993; Irving et al., 2018; Brown-Cohen et al., 2023) leverage interactive proof systems to enable complex decision processes to be broken down into verifiable components, ensuring reliability even under adversarial conditions.

B Additional Methods in Related Work

This section provides further practical details on the key areas relevant to our work—namely, watermarking techniques, adversarial defenses, and transferable attacks on Deep Neural Networks (DNNs). The discussion here emphasizes implementation nuances and empirical findings, complementing the broader overview provided earlier.

B.1 Watermarking

Watermarking techniques are essential for protecting the intellectual property of machine learning models. We briefly review practical watermarking schemes for both discriminative and generative models, focusing on aspects that extend beyond the theoretical presentations.

B.1.1 Watermarking Schemes for Discriminative Models

Discriminative models, which categorize input data into predefined classes, have been a primary focus of watermarking research. In practice, the approaches fall into two settings:

Black-Box Setting. In the black-box setting, the model owner can only query the model to observe outputs. Frameworks such as those proposed by Adi et al. (2018) and Zhang et al. (2018) embed watermarks using specifically crafted input data with predefined outcomes. These inputs serve as triggers whose responses verify the watermark. Other methods, like that of Merrer et al. (2017), use adversarial examples to induce backdoor behaviors, while Namba & Sakuma (2019) further enhance robustness against model modifications and attacks. Although Goldwasser et al. (2022) achieved provable undetectability, practical observations indicate that some of these watermarks can be removed by mechanisms akin to randomized smoothing (Cohen et al., 2019). The practical appeal of black-box watermarking lies in its applicability to scenarios where models are deployed as APIs or services—a setting our work builds upon.

White-Box Setting. When full access to model parameters is available, watermarking can be integrated directly into the model's weights. Early approaches by Uchida et al. (2017) and Nagai et al. (2018) laid the groundwork for embedding watermarks that can be verified through internal examination. An improved method by Darvish Rouhani et al. (2019) embeds an N-bit watermark that

is both data- and model-dependent, requiring specific inputs for activation. Since our work focuses on backdoor-like techniques in black-box settings, we only briefly review these for contrast.

B.1.2 Watermarking Schemes for Generative Models

Watermarking techniques for generative models have gained attention with the rise of advanced architectures, such as Large Language Models (LLMs). In practice, these methods must address modality-specific challenges.

Backdoor-Based Watermarking for Pre-trained Language Models. Backdoor-based watermarking in Pre-trained Language Models (PLMs) (e.g., (Gu et al., 2022; Li et al., 2023)) leverages rare or common word triggers to embed watermarks. These empirical approaches ensure that the watermark remains robust across downstream tasks and resistant to removal techniques like fine-tuning or pruning.

Watermarking the Output of LLMs. Watermarking generated text is critical for mitigating potential harms. For instance, Kirchenbauer et al. (2023) propose a framework that embeds subtle signals into the text—using a randomized set of "green" tokens—that are imperceptible to humans but detectable algorithmically. Complementary approaches by Kuditipudi et al. (2023) and Zhao et al. (2023a) ensure distortion-free, robust watermarks, even as Zhang et al. (2023) highlight vulnerabilities that need to be addressed.

Image Generation Models. Watermarking techniques for image generation have also been developed to meet ethical and legal challenges. Fernandez et al. (2023) combine watermarking with Latent Diffusion Models to embed invisible marks robust to modifications like cropping, while Wen et al. (2023b) introduce Tree-Ring Watermarking that embeds a pattern into the initial noise vector. Works by Jiang et al. (2023) and Zhao et al. (2023c) further examine both the robustness and limitations of these approaches. Additionally, Zhao et al. (2023b) shows that invisible watermarks may be vulnerable to regeneration attacks, suggesting that semantically similar watermarks could offer improved resilience.

Audio Generation Models. Watermarking techniques for audio generators have been developed for robustness against various attacks. Erfani et al. (2017) introduced a spikegram-based method, embedding watermarks in high-amplitude kernels, robust against MP3 compression and other attacks while preserving quality. Liu et al. (2023) proposed DeAR, a deep-learning-based approach resistant to audio re-recording (AR) distortions.

B.2 Adversarial Defense

Adversarial defenses are crucial for ensuring the reliability of machine learning models against carefully crafted perturbations. In practice, techniques such as adversarial training (Madry et al., 2018), certified defenses (Raghunathan et al., 2018), and randomized smoothing (Cohen et al., 2019) have been successfully implemented. Notably, the work of Goldwasser et al. (2020) explores alternative models for generating adversarial examples, providing insights that are relevant to the robustness of watermarking techniques. Additionally, in the context of LLMs, research on adversarial examples (Zou et al., 2023; Carlini et al., 2023; Wen et al., 2023a) and jailbreaking (Andriushchenko et al., 2024; Chao et al., 2023; Mehrotra et al., 2024; Wei et al., 2023) continues to highlight the practical challenges in this area.

B.3 Transferable Attacks and Transductive Learning

Transferable attacks refer to adversarial examples that are effective across multiple models. Moreover, *transductive learning* has been explored as a means to enhance adversarial robustness, and since our Definition 5 captures some notion of transductive learning in the context of Transferable Attacks, we highlight significant contributions in these areas.

Adversarial Robustness via Transductive Learning. Transductive learning (Gammerman et al., 1998) has shown promise in improving the robustness of models by utilizing both training and test

data during the learning process. This approach aims to make models more resilient to adversarial perturbations encountered at test time.

One significant contribution is by Goldwasser et al. (2020), which explores learning guarantees in the presence of arbitrary adversarial test examples, providing a foundational framework for transductive robustness. Another notable study by Chen et al. (2021) formalizes transductive robustness and proposes a bilevel attack objective to challenge transductive defenses, presenting both theoretical and empirical support for transductive learning's utility.

Additionally, Montasser et al. (2022) introduce a transductive learning model that adapts to perturbation complexity, achieving a robust error rate proportional to the VC dimension. The method by Wu et al. (2020) improves robustness by dynamically adjusting the network during runtime to mask gradients and cleanse non-robust features, validated through experimental results. Lastly, Tramer et al. (2020) critique the standard of adaptive attacks, demonstrating the need for specific tuning to effectively evaluate and enhance adversarial defenses.

Transferable Attacks on DNNs. Transferable attacks exploit the vulnerability of models to adversarial examples that generalize across different models. For discriminative models, significant works include Liu et al. (2016), which investigates the transferability of adversarial examples and their effectiveness in black-box attack scenarios, (Xie et al., 2018), who propose input diversity techniques to enhance the transferability of adversarial examples across different models, and (Dong et al., 2019), which presents translation-invariant attacks to evade defenses and improve the effectiveness of transferable adversarial examples.

In the context of generative models, including LLMs and other advanced generative architectures, relevant research is rapidly emerging, focusing on the transferability of adversarial attacks. This area is crucial as it aims to understand and mitigate the risks associated with adversarial examples in these powerful models. Notably, Zou et al. (2023) explored universal and transferable adversarial attacks on aligned language models, highlighting the potential vulnerabilities and the need for robust defenses in these systems.

		Undetectability	**Unremovability**	**Uniqueness**
Classification	Goldwasser et al. (2022)	✔	robust to some smoothing attacks	✔(E)
	Adi et al. (2018); Zhang et al. (2018)	✔(E)	✗	✔(E)
	Merrer et al. (2017)	✔(E)	robust to fine tunning attacks	✔(E)
LLMs	Christ et al. (2023); Kuditipudi et al. (2023)	✔	✗	✔
	Zhao et al. (2023a)	✗	robust to edit distance attacks only	✔
	Tiffany Hsu (2023)	✔(E)	✗	✔
	Kirchenbauer et al. (2023)	✗	✗	✔

Table 1: Overview of properties across various watermarking schemes. The symbol ✔ denotes properties with formal guarantees or where proof is plausible, whereas ✗ indicates the absence of such guarantees. Entries marked with ✔(E) represent properties observed empirically; these lack formal proof in the corresponding literature, suggesting that deriving such proof may present substantial challenges. The LLM watermarking schemes refer to those applied to text generated by these models.

C PRELIMINARIES

For $n \in \mathbb{N}$ we define $[n] := \{1, \ldots, n\}$. We say a boolean sequence $a : \mathbb{N} \to \{0,1\}$ is true with frequency $\alpha \in [0,1]$ if

$$\liminf_{n\to\infty} \frac{\sum_{i\in[n]} a(i)}{n} \geq \alpha.$$

For two sequences $a, b : \mathbb{N} \to \mathbb{R}$ we say they agree with frequency at least $\alpha \in [0,1]$ if the sequence $(a \stackrel{?}{=} b) : \mathbb{N} \to \{0,1\}$, i.e. $(a \stackrel{?}{=} b)(n) = \mathbb{1}_{a(n)=b(n)}$, is true with frequency α.

Learning. For a set Ω, we write $\Delta(\Omega)$ to denote the set of all probability measures defined on the measurable space $(\Omega, \mathcal{F})$, where $\mathcal{F}$ is some fixed σ-algebra that is implicitly understood. For a parameter n, we denote by $\{0,1\}^n$ the input space and by $\{0,1\}$ the output space. A *model* is a function $f : \{0,1\}^n \to \{0,1\}$.

Definition 6 (*Learning Task*)**.** A *learning task* $\mathbb{L}$ is a family $\{\mathbb{L}_n\}_{n\in\mathbb{N}}$, where for every n, $\mathbb{L}_n$ is an element of $\Delta\left(\Delta(\{0,1\}^n) \times \{0,1\}^{\{0,1\}^n}\right)$.

For a *distribution* $\mathcal{D}_n \in \Delta(\{0,1\}^n)$ and a *ground truth* $h_n : \{0,1\}^n \to \{0,1\}$, we define an *error* of f as $\mathrm{err}_{\mathcal{D}_n,h_n}(f) := \mathbb{E}_{x\sim\mathcal{D}_n}[f(x) \neq h(x)]$, where the index of err will often be understood implicitly and omitted in notation. For $\mathcal{D}_n \in \Delta(\{0,1\}^n)$, $h_n : \{0,1\}^n \to \{0,1\}$ we define an *example oracle* $\mathrm{Ex}(\mathcal{D}_n, h_n)$ as an oracle that samples $x \sim \mathcal{D}_n$ and returns $(x, h_n(x))$.

Interaction. When $\mathrm{Ex}(\mathcal{D}, h)$ generates $(x, h(x))$ it is encoded as an $n+1$ bit-string, because $x \in \{0,1\}^n$ and the label space is $\{0,1\}$. For a *message space* $\mathcal{M} = \{\mathcal{M}_n\}_n = \{\{0,1\}^{m(n)}\}_n$ a *representation class* is a collection of mappings $\{\mathcal{R}_n\}_n$, where for every n, $\mathcal{R}_n : \mathcal{M}_n \to \{0,1\}^{\{0,1\}^n}$. Thus, there is a function class corresponding to a representation, i.e., for every n there is a function class $\mathcal{F}_n$, which is an image of $\mathcal{R}_n$. Note that h_n (which is the ground truth) may or may not be in $\mathcal{F}_n$. All function classes considered in this work have an implicit representation class and an underlying message space.

Computation. We work with the collection of Boolean circuits over the standard basis B_2, the set of all two-bit Boolean functions. The size of a circuit C is measured by its number of gates; let $|C|$ denote the size of C. For a circuit family $\mathcal{C} = \{C_n\}_n$ we say it has a circuit complexity $S(n)$ if for every n, $|C_n| \leq S(n)$.

For a distribution $\mathcal{D}_n$ over $\{0,1\}^n$, and a ground truth $h_n : \{0,1\}^n \to \{0,1\}$ we denote by $C^{\mathrm{Ex}(\mathcal{D}_n,h_n)}$ a circuit with some[6] number of specified input gates that are initialized with samples $(x, h(x))$ sampled from $x \sim \mathcal{D}_n$. We will also by interested in interaction between circuits. When messages are exchanged between circuits we assume that there are specified input (output) gates that correspond to outgoing (ingoing) messages. Also, when a circuit is randomized we assume there are designated input gates that are initialized with random bits.

Definition 7 (*Computationally Bounded Learnability*)**.** For $\epsilon, \delta : \mathbb{N} \to (0,1)$ we say that a learning task $\mathbb{L} = \{\mathbb{L}_n\}_{n\in\mathbb{N}}$ is learnable to error ϵ with confidence $1-\delta$ and with circuit complexity $S : \mathbb{N} \to \mathbb{N}$ by a function class $\mathcal{F} = \{\mathcal{F}_n\}_{n\in\mathbb{N}}$ (with a corresponding representation class $\mathcal{R}$), or $(\epsilon, \delta, S, \mathcal{F})$-learnable in short, if there exists a circuit family $\mathcal{C} = \{C_n\}_{n\in\mathbb{N}}$ with complexity $S(n)$ such that for every sufficiently large n, with probability $1-\delta$ over the choice of $(\mathcal{D}_n, h_n) \sim \mathbb{L}_n$, $C_n^{\mathrm{Ex}(\mathcal{D}_n,h_n)}$ computes an $m(n)$ bit message $m_{f_n} \in \mathcal{M}_n$ such that $\mathcal{R}_n(m_{f_n}) \in \mathcal{F}_n$ has error at most ϵ, i.e. for every sufficiently large n

$$\mathbb{P}_{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n, m_{f_n}\leftarrow C_n^{\mathrm{Ex}(\mathcal{D}_n,h_n)}}\left[\mathrm{err}_{\mathcal{D}_n,h_n}(\mathcal{R}_n(m_{f_n})) \leq \epsilon(n)\right] \geq 1-\delta(n).$$

We often abuse the notation and use f_n to denote both m_{f_n} as well as $R_n(m_{f_n})$.

[6]We will not specify the sample complexity explicitly. In this paper, we focus only on circuit complexity. The sample complexity is an important parameter to analyze and we leave it for future work. We emphasize that the circuit complexity is an upper bound on the sample complexity.

D FORMAL DEFINITIONS

Definition 8 (*Watermark*)**.** Let $\mathbb{L} = \{\mathbb{L}_n\}_n$ be a learning task, and $\mathcal{F} = \{\mathcal{F}_n\}_n$ a function class. Let $S_{\mathbf{A}}, S_{\mathbf{B}}, q : \mathbb{N} \to \mathbb{N}, \epsilon \in \left(0, \frac{1}{2}\right), l, c, s \in (0,1), s < c$, where $S_{\mathbf{B}}(n)$ bounds the circuit size of $\mathbf{B}_n$, and $S_{\mathbf{A}}(n)$ the circuit size of $\mathbf{A}_n$, $q(n)$ the number of queries, ϵ the risk level, c probability that *uniqueness* holds, s probability that *unremovability* and *undetectability* holds, l the learning probability.

We say that a family of circuits $\mathbf{A}^{\text{WATERMARK}} = \{\mathbf{A}_n^{\text{WATERMARK}}\}_n$ *with complexity* $S_{\mathbf{A}}(n)$ implements a watermarking scheme for $\mathbb{L}$ with frequency α, denoted by

$$\mathbf{A}^{\text{WATERMARK}} \in_\alpha \text{WATERMARK}\left(\mathbb{L}, \mathcal{F}, \epsilon, q, S_{\mathbf{A}}, S_{\mathbf{B}}, l, c, s\right),$$

if the following is true with frequency α over parameter n. An interactive protocol in which first $(\mathcal{D}_n, h_n) \sim \mathbb{L}_n$ and then $\mathbf{A}_n^{\text{WATERMARK}}$ computes $(f, \mathbf{x})$, $f : \{0,1\}^n \to \{0,1\}, \mathbf{x} \in (\{0,1\}^n)^{q(n)}$, and $\mathbf{B}_n$ outputs $\mathbf{y} = \mathbf{B}_n(f, \mathbf{x}), \mathbf{y} \in \{0,1\}^{q(n)}$, where f is sent using the representation $\mathcal{R}_n$, satisfies the following

- **Correctness** (f has low error). With probability at least l
$$\text{err}(f) \leq \epsilon.$$
- **Uniqueness** (models trained from scratch give low-error answers). There exists a circuit $\mathbf{B}_n$ of size $S_{\mathbf{A}}(n)$ such that with probability at least c
$$\text{err}(\mathbf{x}, \mathbf{y}) \leq 2\epsilon.$$
- **Unremovability** (fast $\mathbf{B}_n$ give high-error answers). For every circuit $\mathbf{B}_n$ *of size at most* $S_{\mathbf{B}}(n)$ we have that with probability at most s
$$\text{err}(\mathbf{x}, \mathbf{y}) \leq 2\epsilon.$$
- **Undetectability** (fast $\mathbf{B}_n$ cannot detect that they are tested). On average over $(\mathcal{D}_n, h_n) \sim \mathbb{L}_n$, distributions $\mathcal{D}_n^{q(n)}$ and $\mathbf{x} \sim \mathbf{A}_n^{\text{WATERMARK}n}$ are $\frac{s}{2}$-indistinguishable for a class of circuits $\mathbf{B}_n$ *of size at most* $S_{\mathbf{B}}(n)$, i.e., for every circuit $\mathbf{B}_n$ of size at most $S_{\mathbf{B}}(n)$ returning one bit,
$$\left|\mathbb{P}_{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n, \mathbf{x}'\sim\mathcal{D}_n^{q(n)}, (f,\mathbf{x})\leftarrow\mathbf{A}_n^{\text{WATERMARK}}}\left[\mathbf{B}(f,\mathbf{x}')=0\right] - \mathbb{P}_{(\mathcal{D}_n,h_n)\sim\mathbb{L}, (f,\mathbf{x})\leftarrow\mathbf{A}_n^{\text{WATERMARK}}}\left[\mathbf{B}(f,\mathbf{x})=0\right]\right| \leq \frac{s}{2}.$$

Definition 9 (*Adversarial Defense*)**.** Let $\mathbb{L} = \{\mathbb{L}_n\}_n$ be a learning task, and $\mathcal{F} = \{\mathcal{F}_n\}_n$ a function class. Let $S_{\mathbf{A}}, S_{\mathbf{B}}, q : \mathbb{N} \to \mathbb{N}$, $\epsilon \in \left(0, \frac{1}{2}\right)$, $l, c, s \in (0,1)$, with $s < c$, where $S_{\mathbf{A}}(n)$ bounds the circuit size of $\mathbf{A}_n$, and $S_{\mathbf{B}}(n)$ the circuit size of $\mathbf{B}_n$, $q(n)$ the number of queries, ϵ the error parameter, c the completeness, s the soundness, and l the learning probability.

We say that a family of circuits $\mathbf{B}^{\text{DEFENSE}} = \{\mathbf{B}_n^{\text{DEFENSE}}\}_n$ *with complexity* $S_{\mathbf{A}}(n)$ implements an adversarial defense for $\mathbb{L}$ with frequency α, denoted by

$$\mathbf{B}^{\text{DEFENSE}} \in_\alpha \text{DEFENSE}\left(\mathbb{L}, \mathcal{F}, \epsilon, q, S_{\mathbf{A}}, S_{\mathbf{B}}, l, c, s\right),$$

if the following is true with frequency α over parameter n. An interactive protocol in which first $(\mathcal{D}_n, h_n) \sim \mathbb{L}_n$, $\mathbf{B}_n^{\text{DEFENSE}}$ computes $f : \{0,1\}^n \to \{0,1\}$, $\mathbf{A}_n$ replies with $\mathbf{x} = \mathbf{A}_n(f_n)$, $\mathbf{x} \in (\{0,1\}^n)^{q(n)}$, and $\mathbf{B}_n^{\text{DEFENSE}}$ outputs $b = \mathbf{B}_n^{\text{DEFENSE}}(f, \mathbf{x})$, $b \in \{0,1\}$, satisfies the following:

- **Correctness** (f_n has low error). With probability at least l
$$\text{err}(f) \leq \epsilon.$$
- **Completeness** (natural inputs are not flagged as adversarial). When $\mathbf{x} \sim \mathcal{D}_n^{q(n)}$, with probability at least c
$$b = 0.$$
- **Soundness** (adversarial inputs are detected). For every circuit $\mathbf{A}_n$ of size at most $S_{\mathbf{A}}(n)$, with probability at most s
$$\text{err}(\mathbf{x}, f(\mathbf{x})) > 7\epsilon \text{ and } b = 0.$$

Definition 10 (*Transferable Attack*)**.** Let $\mathbb{L} = \{\mathbb{L}_n\}_n$ be a learning task and $\mathcal{F} = \{\mathcal{F}_n\}_n$ a function class. Let $S_{\mathbf{A}}, S_{\mathbf{B}}, q : \mathbb{N} \to \mathbb{N}$, $\epsilon \in \left(0, \frac{1}{2}\right)$, and $c, s \in (0,1)$, with $s < c$, where $S_{\mathbf{A}}(n)$ bounds the circuit size of $\mathbf{A}_n$, and $S_{\mathbf{B}}$ the circuit size of $\mathbf{B}_n$, $q(n)$ the number of queries, ϵ the error parameter, c the *transferability* probability, and s the *undetectability* probability.

We say that a family of circuits $\mathbf{A}^{\text{TRANSFATTACK}} = \{\mathbf{A}_n^{\text{TRANSFATTACK}}\}$ *with complexity* $S_{\mathbf{A}}(n)$ implements a transferable attack for $\mathbb{L}$ with frequency α, denoted by

$$\mathbf{A}^{\text{TRANSFATTACK}} \in_\alpha \text{DEFENSE}\left(\mathbb{L}, \mathcal{F}, \epsilon, q, S_{\mathbf{A}}, S_{\mathbf{B}}, l, c, s\right),$$

if the following is true with frequency α over parameter n. An interactive protocol in which first $(\mathcal{D}_n, h_n) \sim \mathbb{L}_n$, $\mathbf{A}_n^{\text{TRANSFATTACK}}$ computes $\mathbf{x} \in (\{0,1\}^n)^{q(n)}$, and $\mathbf{B}_n$ outputs $\mathbf{y} = \mathbf{B}_n(\mathbf{x})$, $\mathbf{y} \in (\{0,1\})^{q(n)}$, satisfies the following:

- **Transferability** (fast provers return high-error answers). For every circuit $\mathbf{B}_n$ of size at most $S_{\mathbf{B}}(n)$, with probability at least c

$$\text{err}(\mathbf{x}, \mathbf{y}) > 2\epsilon.$$

- **Undetectability** (fast provers cannot detect that they are tested). On average over $(\mathcal{D}_n, h_n) \sim \mathbb{L}_n$, distributions $\mathbf{x} \sim \mathcal{D}_n^{q(n)}$ and $\mathbf{x} := \mathbf{A}_n^{\text{TRANSFATTACK}}$ are $\frac{s}{2}$-indistinguishable for every circuit $\mathbf{B}_n$ of size at most $S_{\mathbf{B}}(n)$, i.e.,

$$\left| \mathbb{P}_{(\mathcal{D}_n, h_n) \sim \mathbb{L}_n, \mathbf{x}' \sim \mathcal{D}_n^{q(n)}} \left[\mathbf{B}_n(\mathbf{x}') = 0\right] - \mathbb{P}_{(\mathcal{D}_n, h_n) \sim \mathbb{L}_n} \left[\mathbf{B}_n(\mathbf{x}) = 0\right] \right| \leq \frac{s}{2}.$$

E MAIN THEOREM

Before proving our main theorem we recall a result from Lipton & Young (1994a) about simple strategies for large zero-sum games.

Game theory. A *two-player zero-sum game* is specified by a payoff matrix $\mathcal{G}$. $\mathcal{G}$ is an $r \times c$ matrix. MIN, the row player, chooses a probability distribution p_1 over the rows. MAX, the column player, chooses a probability distribution p_2 over the columns. A row i and a column j are drawn from p_1 and p_2 and MIN pays $\mathcal{G}_{ij}$ to MAX. MIN tries to minimize the expected payment; MAX tries to maximize it.

By the Min-Max Theorem, there exist optimal strategies for both MIN and MAX. Optimal means that playing first and revealing one's mixed strategy is not a disadvantage. Such a pair of strategies is also known as a Nash equilibrium. The expected payoff when both players play optimally is known as the value of the game and is denoted by $\mathcal{V}(\mathcal{G})$.

We will use the following theorem from Lipton & Young (1994a), which says that optimal strategies can be approximated by uniform distributions over sets of pure strategies of size $O(\log(c))$.

Theorem 4 (Lipton & Young (1994a))**.** *Let $\mathcal{G}$ be an $r \times c$ payoff matrix for a two-player zero-sum game. For any $\eta \in (0,1)$ and $k \geq \frac{\log(c)}{2\eta^2}$ there exists a multiset of pure strategies for the* MIN *(row player) of size k such that a mixed strategy p_1 that samples uniformly from this multiset satisfies*

$$\max_j \sum_i p_1(i)\mathcal{G}_{ij} \leq \mathcal{V}(\mathcal{G}) + \eta(\mathcal{G}_{max} - \mathcal{G}_{min}),$$

where $\mathcal{G}_{max}, \mathcal{G}_{min}$ denote the maximum and minimum entry of $\mathcal{G}$ respectively. The symmetric result holds for the MAX *player.*

We are ready to prove our main theorem.

Theorem 5. *Let $\epsilon \in \left(0, \frac{1}{2}\right), \delta \in \left(0, \frac{1}{48}\right), S : \mathbb{N} \to \mathbb{N}$. For every learning task $\mathbb{L} = \{\mathbb{L}_n\}_n$ learnable to error ϵ with confidence $1 - \delta$ and circuit complexity $O\left(\frac{\sqrt{S(n)}}{\log(S(n))}\right)$ and for every family of function classes $\mathcal{F} = \{\mathcal{F}_n\}_n$, every query bound $q(n)$ such that $\frac{\sqrt{S(n)}}{\log(S(n))} = \Omega(m(n) + q(n) \cdot n)$*

at least one of the three

$$\textsc{Watermark}\left(\mathbb{L}, \mathcal{F}, \epsilon, q, S(n), o\left(\frac{\sqrt{S(n)}}{\log(S(n))}\right), l = \frac{10}{24}, c = \frac{21}{24}, s = \frac{19}{24}\right),$$

$$\textsc{Defense}\left(\mathbb{L}, \mathcal{F}, \epsilon, q, o\left(\frac{\sqrt{S(n)}}{\log(S(n))}\right), O(S(n)), l = 1 - \frac{1}{48}, c = \frac{13}{24}, s = \frac{11}{24}\right),$$

$$\textsc{TransfAttack}\left(\mathbb{L}, \mathcal{F}, \epsilon, q, S(n), S(n), c = \frac{3}{24}, s = \frac{19}{24}\right)$$

exists with frequency $\frac{1}{3}$.

Proof. Let $\epsilon \in (0, \frac{1}{2})$ and $q : \mathbb{N} \to \mathbb{N}$ be a query bound. Let $\mathbb{L}$ be a learning task learnable to error ϵ with confidence $1 - \delta$ and complexity $S(n)$.

We will consider every n separately and show that for every n, one of the three schemes exists. This automatically implies that one of the schemes exists with frequency at least $\frac{1}{3}$.

Let $s(n) = \Theta\left(\frac{\sqrt{S(n)}}{\log(S(n))}\right)$, where the exact constants will be determined later. Let $\mathfrak{Candidate}_{\mathfrak{W}}(n)$ be a set of $s(n)$-sized circuits computing $(f, \mathbf{x})$. Recall that the execution of a $\mathbf{A}_n \in \mathfrak{C}_{\mathfrak{W}}(n)$ proceeds by first sampling from $\mathrm{Ex}(\mathcal{D}_n, h_n)$ and providing these samples as inputs to $\mathbf{A}_n$ and then running $\mathbf{A}_n$ to obtain $m + q \cdot n$ bits. The first m bits are interpreted as a representation of f (according to $\mathcal{R}_n$), and the following consecutive blocks of n bits each are interpreted as q elements of $\{0,1\}^n$. Similarly, let $\mathfrak{C}_{\mathfrak{D}}(n)$ be a set of $s(n)$-sized circuits accepting as input $(f, \mathbf{x})$ and outputting $(\mathbf{y}, b)$, where $\mathbf{y} \in \{0,1\}^q, b \in \{0,1\}$. Formally, this is a set of circuits with up to $s(n)$ input gates and $q + 1$ output gates. We interpret $\mathfrak{C}_{\mathfrak{W}}(n)$ as candidate algorithms for a watermark, and $\mathfrak{C}_{\mathfrak{D}}(n)$ as candidate algorithms for attacks on watermarks.

For every n define a zero-sum game $\mathcal{G}_n$ between $\mathbf{A}_n \in \mathfrak{C}_{\mathfrak{W}}(n), \mathbf{B}_n \in \mathfrak{C}_{\mathfrak{D}}(n)$. The payoff is given by

$$\begin{aligned}\mathcal{G}_n(\mathbf{A}_n, \mathbf{B}_n) = &\frac{1}{2}\,\mathbb{P}_{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,(f,\mathbf{x}):=\mathbf{A}_n^{\mathrm{Ex}(\mathcal{D}_n,h_n)},(\mathbf{y},b):=\mathbf{B}_n^{\mathrm{Ex}(\mathcal{D}_n,h_n)}}\Big[\text{err}(f) > \epsilon \text{ or } \text{err}(\mathbf{x},\mathbf{y}) \le 2\epsilon \text{ or } b = 1\Big] \\ &+ \frac{1}{2}\,\mathbb{P}_{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,f:=\mathbf{A}_n^{\mathrm{Ex}(\mathcal{D}_n,h_n)},\mathbf{x}\sim\mathcal{D}_n^{q(n)},(\mathbf{y},b):=\mathbf{B}_n^{\mathrm{Ex}(\mathcal{D}_n,h_n)}}\Big[\text{err}(f) > \epsilon \text{ or } \Big(\text{err}(\mathbf{x},\mathbf{y}) \le 2\epsilon \text{ and } b = 0\Big)\Big],\end{aligned}$$

where $\mathbf{A}_n$ tries to minimize and $\mathbf{B}_n$ maximize the payoff.

Then the number of possible circuits is bounded by

$$|\mathfrak{C}_{\mathfrak{W}}| \le (3s(n)^2)^{s(n)} \le 2^{3s(n)\log(s(n))},$$

because every internal gate of a circuit is one of AND, OR, and NOT, and is connected to 2 gates out of at most $s(n)$ choices.

Applying Theorem 4 to $\mathcal{G}_n$ with $\eta = 2^{-5}$ we get two probability distributions, p over a multiset of pure strategies in $\mathfrak{C}_{\mathfrak{W}}$ and r over a multiset of pure strategies in $\mathfrak{C}_{\mathfrak{D}}$ that lead to a 2^{-5}-approximate Nash equilibrium. The size $k(n)$ of the multisets is bounded

$$\begin{aligned}k(n) &\le 2^6 \log\left(|\mathfrak{C}_{\mathfrak{W}}|\right) \\ &\le O(s(n)\log(s(n))).\end{aligned} \tag{1}$$

Next, observe that the mixed strategy corresponding to the distribution p can be represented by a circuit of size

$$\begin{aligned}&k(n) \cdot s(n) \cdot O(\log(k(n))) \\ &\le O(s^2(n) \cdot \log^3(s(n))) && \text{By equation (1)} \\ &\le S(n),\end{aligned}$$

because we can create a circuit that is a collection of $k(n)$ circuits corresponding to the multiset of p, where each one is of size $s(n)$ with additional gadgets of size $O(\log(k))$ activating the corresponding gate depending on the randomness determining a strategy. This implies that p can be implemented by a $S(n)$-sized circuit. The same holds for r. Let's call the strategy corresponding to p, $\mathbf{A}^n_{\text{Nash}}$, and the strategy corresponding to r, $\mathbf{B}^n_{\text{Nash}}$.

Consider cases:

Case $\mathcal{G}(\mathbf{A}_n^{\text{NASH}}, \mathbf{B}_n^{\text{NASH}}) \geq \frac{19}{24}$. Define $\mathbf{B}_n^{\text{DEFENSE}}$ to work as follows:

1. Simulate the circuit of size $O\left(\frac{\sqrt{S(n)}}{\log(S(n))}\right)\mathbf{L}_n$ that learns f, such that
$$\mathbb{P}_{\substack{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,\\ f\leftarrow\mathbf{L}_n^{\text{Ex}(\mathcal{D}_n,h_n)}}}\left[\text{err}(f)\leq\epsilon\right]\geq 1-\frac{1}{48}.$$
2. Send f to $\mathbf{A}_n$.
3. Receive $\mathbf{x}$ from $\mathbf{A}_n$.
4. Simulate $(\mathbf{y}, b) := \mathbf{B}_n^{\text{NASH}}(f, \mathbf{x})$.
5. Return $b' = 1$ if $b = 1$ or $d(f(\mathbf{x}), \mathbf{y}) > 3\epsilon \cdot q(n)$ and $b' = 0$ otherwise,

where $d(\cdot,\cdot)$ is the Hamming distance. $\mathbf{B}_n^{\text{DEFENSE}}$ can be implemented by circuit of size $O(S(n))$, because it simulates a circuit of size $O\left(\frac{\sqrt{S(n)}}{\log(S(n))}\right)$, then simulating $\mathbf{B}_n^{\text{NASH}}$ of size $S(n)$, and computing a predicate $d(f(\mathbf{x}), \mathbf{y}) > 3\epsilon q$, which can be done in size $\log(q(n))$.

We claim that

$$\text{DEFENSE}\left(\mathbb{L}_n, \mathcal{F}_n, \epsilon, q(n), o\left(\frac{\sqrt{S(n)}}{\log(S(n))}\right), O(S(n)), l = 1-\frac{1}{48}, c=\frac{13}{24}, s=\frac{11}{24}\right). \quad (2)$$

Assume towards contradiction that completeness or soundness of $\mathbf{B}_n^{\text{DEFENSE}}$ as defined in Definition 9 does not hold.

If completeness of $\mathbf{B}_n^{\text{DEFENSE}}$ does not hold, then

$$\mathbb{P}_{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,\mathbf{x}\sim\mathcal{D}_n^q}\left[b'=0\right] < \frac{13}{24}. \quad (3)$$

Let us compute the payoff of $\mathbf{A}_n$, which first runs $f \leftarrow \mathbf{L}_n^{\text{Ex}(\mathcal{D}_n,h_n)}$ (where $\mathbf{L}_n$ is the learning circuit) and sets $\mathbf{x} \sim \mathcal{D}^q$, in the game $\mathcal{G}_n$, when playing against $\mathbf{B}_n^{\text{NASH}}$

$$\begin{aligned}
&\mathcal{G}(\mathbf{A}_n, \mathbf{B}_n^{\text{NASH}})\\
&= \frac{1}{2}\mathbb{P}_{\substack{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,\\ (f,\mathbf{x})\leftarrow\mathbf{A}_n^{\text{Ex}(\mathcal{D}_n,h_n)}}}\left[\text{err}(f)>\epsilon \text{ or } \text{err}(\mathbf{x},\mathbf{y})\leq 2\epsilon \text{ or } b'=1\right]\\
&+ \frac{1}{2}\mathbb{P}_{\substack{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,\\ f\leftarrow\mathbf{A}_n^{\text{Ex}(\mathcal{D}_n,h_n)},\\ \mathbf{x}\sim\mathcal{D}_n^q}}\left[\text{err}(f)>\epsilon \text{ or } \left(\text{err}(\mathbf{x},\mathbf{y})\leq 2\epsilon \text{ and } b'=0\right)\right]\\
&\leq \delta + \frac{1}{2}\mathbb{P}_{\substack{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,\\ f\leftarrow\mathbf{L}_n^{\text{Ex}(\mathcal{D}_n,h_n)},\\ \mathbf{x}\sim\mathcal{D}_n^q}}\left[\text{err}(\mathbf{x},\mathbf{y})\leq 2\epsilon \text{ or } b'=1\right]\\
&+ \frac{1}{2}\mathbb{P}_{\substack{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,\\ f\leftarrow\mathbf{L}_n^{\text{Ex}(\mathcal{D}_n,h_n)},\\ \mathbf{x}\sim\mathcal{D}_n^q}}\left[\text{err}(\mathbf{x},\mathbf{y})\leq 2\epsilon \text{ and } b'=0\right] && \text{Def. of } \mathbf{A}_n, \mathbf{B}_n^{\text{DEFENSE}}, \mathbb{P}\left[\text{err}(f)\leq\epsilon\right]\geq\frac{47}{48}\\
&< \frac{1}{48}+\frac{1}{2}+\frac{\frac{13}{24}}{2} && \text{By equation (3)}\\
&= \frac{38}{48}\\
&\leq \mathcal{G}(\mathbf{A}_n^{\text{NASH}}, \mathbf{B}_n^{\text{NASH}}),\ \lightning
\end{aligned}$$

where the contradiction is with the properties of Nash equilibria.

Assume that $\mathbf{A}_n$ breaks the soundness of $\mathbf{B}_n^{\text{DEFENSE}}$, which translates to

$$\mathbb{P}_{\substack{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,\\ \mathbf{x}\leftarrow\mathbf{A}_n(f)}}\Big[\text{err}(\mathbf{x},f(\mathbf{x}))>7\epsilon \text{ and } b=0 \text{ and } d(f(\mathbf{x}),\mathbf{y}))>3\epsilon q\Big]>\frac{11}{24}. \tag{4}$$

Let $\mathbf{A}'_n$ first simulate $f\leftarrow\mathbf{L}_n^{\text{Ex}(\mathcal{D}_n,h_n)}$, then runs $\mathbf{x}\leftarrow\mathbf{A}_n(f)$, and returns $(f,\mathbf{x})$. We have

$$\begin{aligned}
&\mathcal{G}(\mathbf{A}'_n,\mathbf{B}_n^{\text{NASH}})\\
&=\frac{1}{2}\,\mathbb{P}_{\substack{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,\\ (f,\mathbf{x})\leftarrow\mathbf{A}'_n}}\Big[\text{err}(f)>\epsilon \text{ or } \text{err}(\mathbf{x},\mathbf{y})\le 2\epsilon \text{ or } b'=1\Big]\\
&+\frac{1}{2}\,\mathbb{P}_{\substack{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,\\ f\leftarrow\mathbf{A}'_n,\\ \mathbf{x}\sim\mathcal{D}_n^q}}\Big[\text{err}(f)>\epsilon \text{ or } \Big(\text{err}(\mathbf{x},\mathbf{y})\le 2\epsilon \text{ and } b'=0\Big)\Big]\\
&=\frac{1}{2}\,\mathbb{P}_{\substack{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,\\ f\leftarrow\mathbf{L}_n^{\text{Ex}(\mathcal{D}_n,h_n)},\\ \mathbf{x}=\mathbf{A}_n(f)}}\Big[\text{err}(f)>\epsilon \text{ or } \text{err}(\mathbf{x},\mathbf{y})\le 2\epsilon \text{ or } b'=1\Big]\\
&+\frac{1}{2}\,\mathbb{P}_{\substack{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,\\ f\leftarrow\mathbf{L}_n^{\text{Ex}(\mathcal{D}_n,h_n)},\\ \mathbf{x}\sim\mathcal{D}_n^q}}\Big[\text{err}(f)>\epsilon \text{ or } \Big(\text{err}(\mathbf{x},\mathbf{y})\le 2\epsilon \text{ and } b'=0\Big)\Big] && \text{By def. of } \mathbf{A}'_n\\
&<\frac{1}{2}+\frac{1-\frac{11}{24}}{2} && \text{By equation (4)}\\
&=\frac{37}{48}\\
&\le\mathcal{G}_n(\mathbf{A}_n^{\text{NASH}},\mathbf{B}_n^{\text{NASH}}),\ \lightning
\end{aligned}$$

where the contradiction is with the properties of Nash equilibria. Thus equation (2) holds.

Case $\mathcal{G}_n(\mathbf{A}_n^{\text{NASH}},\mathbf{B}_n^{\text{NASH}})<\frac{19}{24}$. Consider $\mathbf{B}_n$ that returns $(f(\mathbf{x}),b)$ for a uniformly random b. We have

$$\mathcal{G}_n(\mathbf{A}_n^{\text{NASH}},\mathbf{B}_n)\ge\left(1-\mathbb{P}_{\substack{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,\\ f\leftarrow\mathbf{A}_n^{\text{NASH}}}}\Big[\text{err}(f)\le\epsilon\Big]\right)+\mathbb{P}_{\substack{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,\\ f\leftarrow\mathbf{A}_n^{\text{Nash}}}}\Big[\text{err}(f)\le\epsilon\Big]\cdot\frac{1}{2},$$

because when $\mathbf{x}\sim\mathcal{D}_n^q$ and $\text{err}(f)\le\epsilon$ the probability that $\text{err}(\mathbf{x},\mathbf{y})\le 2\epsilon$ and $b=0$ is $\frac{1}{2}$, and similarly when $\mathbf{x}\leftarrow\mathbf{A}_n^{\text{NASH}}$ then the probability that $b=1$ is equal $\frac{1}{2}$. The assumption that $\mathcal{G}_n(\mathbf{A}_n^{\text{Nash}},\mathbf{B}_n)<\frac{19}{24}$ and properties of Nash equilibria imply that $\mathbb{P}_{\substack{(\mathcal{D}_n,h_n)\sim\mathbb{L}_n,\\ f\leftarrow\mathbf{A}_n^{\text{Nash}}}}[\text{err}(f)\le\epsilon]\ge\frac{10}{24}$. This implies that *correctness* holds for $\mathbf{A}_n^{\text{Nash}}$ with $l=\frac{10}{24}$.

Next, assume towards contradiction that *unremovability* of $\mathbf{A}_n^{\text{NASH}}$ does not hold, i.e., there is $\mathbf{B}_n$ running in time $o\left(\sqrt{S(n)}/\log(S(n))\right)$ such that $\mathbb{P}\big[\text{err}(\mathbf{x},\mathbf{y})\le 2\epsilon\big]>\frac{19}{24}$. Consider $\mathbf{B}'_n$ that on input $(f,\mathbf{x})$ returns $(\mathbf{B}_n(f,\mathbf{x}),0)$. Then by definition of $\mathcal{G}_n$, $\mathcal{G}_n(\mathbf{A}_{\text{NASH}},\mathbf{B}'_n)>\frac{19}{24}$, which is a contradiction $\lightning$.

Next, assume towards contradiction that *undetectability* of $\mathbf{A}_n^{\text{NASH}}$ does not hold, i.e., there exists $\mathbf{B}_n$ such that it distinguishes $\mathbf{x}\sim\mathcal{D}_n^q$ from $\mathbf{x}\leftarrow\mathbf{A}_n^{\text{NASH}}$ with probability higher than $\frac{19}{24}$. Consider $\mathbf{B}'_n$ that on input $(f,\mathbf{x})$ returns $(f(\mathbf{x}),\mathbf{B}_n(f,\mathbf{x}))$.[7] Then by definition of $\mathcal{G}_n$, $\mathcal{G}_n(\mathbf{A}_n^{\text{NASH}},\mathbf{B}'_n)>\frac{19}{24}$, which is a contradiction $\lightning$.

There are two further subcases. If $\mathbf{A}_n^{\text{NASH}}$ satisfies *uniqueness* then

$$\mathbf{A}_n^{\text{NASH}}\in\text{WATERMARK}\left(\mathbb{L}_n,\mathcal{F}_n,\epsilon,q(n),S(n),o\left(\frac{\sqrt{S(n)}}{\log(S(n))}\right),l=\frac{10}{24},c=\frac{21}{24},s=\frac{19}{24}\right).$$

[7] Formally $\mathbf{B}_n$ receives as input $(f,\mathbf{x})$ and not only $\mathbf{x}$.

If $\mathbf{A}_n^{\text{NASH}}$ does not satisfy *uniqueness*, then, by definition, every succinctly representable circuit $\mathbf{B}_n$ of size $o\left(\sqrt{S(n)}/\log(S(n))\right)$ satisfies $\text{err}(\mathbf{x}, \mathbf{y}) \leq 2\epsilon$ with probability at most $\frac{21}{24}$. Consider the following $\mathbf{A}_n$. It computes $(f, \mathbf{x}) \leftarrow \mathbf{A}_n^{\text{Nash}}$, ignores f and sends $\mathbf{x}$ to $\mathbf{B}_n$. By the assumption that *uniqueness* is not satisfied for $\mathbf{A}_n^{\text{NASH}}$ we have that *transferability* of Definition 5 holds for $\mathbf{A}_n$ with $c = \frac{3}{24}$. Note that $\mathbf{B}_n$ in the transferable attack does not receive f but it makes it no easier for it to satisfy the properties. Note that *undetectability* still holds with the same parameter. Thus

$$\mathbf{A}_n^{\text{NASH}} \in \text{TRANSFATTACK}\left(\mathbb{L}_n, \mathcal{F}_n, \epsilon, q(n), S(n), S(n), c = \frac{3}{24}, s = \frac{19}{24}\right).$$

□

F FULLY HOMOMORPHIC ENCRYPTION (FHE)

We include a definition of fully homomorphic encryption based on the definition from Goldwasser et al. (2013). The notion of fully homomorphic encryption was first proposed by Rivest, Adleman and Dertouzos Rivest et al. (1978) in 1978. The first fully homomorphic encryption scheme was proposed in a breakthrough work by Gentry in 2009 Gentry (2009). A history and recent developments on fully homomorphic encryption is surveyed in (Vaikuntanathan, 2011).

F.1 PRELIMINARIES

We say that a function f is *negligible* in an input parameter λ, if for all $d > 0$, there exists K such that for all $\lambda > K$, $f(\lambda) < \lambda^{-d}$. For brevity, we write: for all sufficiently large λ, $f(\lambda) = \text{negl}(\lambda)$. We say that a function f is *polynomial* in an input parameter λ, if there exists a polynomial p such that for all λ, $f(\lambda) \leq p(\lambda)$. We write $f(\lambda) = \text{poly}(\lambda)$. A similar definition holds for $\text{polylog}(\lambda)$. For two polynomials p, q, we say $p \leq q$ if for every $\lambda \in \mathbb{N}$, $p(\lambda) \leq q(\lambda)$.

When saying that a Turing machine $\mathcal{A}$ is p.p.t. we mean that $\mathcal{A}$ is a non-uniform probabilistic polynomial-time machine.

F.2 DEFINITIONS

Definition 11 (Goldwasser et al. (2013))**.** A homomorphic (public-key) encryption scheme FHE is a quadruple of polynomial time algorithms (FHE.KEYGEN, FHE.ENC, FHE.DEC, FHE.EVAL) as follows:

- $\text{FHE.KEYGEN}(1^\lambda)$ is a probabilistic algorithm that takes as input the security parameter 1^λ and outputs a public key pk and a secret key sk.
- $\text{FHE.ENC}(pk, x \in \{0,1\})$ is a probabilistic algorithm that takes as input the public key pk and an input bit x and outputs a ciphertext ψ.
- $\text{FHE.DEC}(sk, \psi)$ is a deterministic algorithm that takes as input the secret key sk and a ciphertext ψ and outputs a message $x^* \in \{0,1\}$.
- $\text{FHE.EVAL}(pk, C, \psi_1, \psi_2, \ldots, \psi_n)$ is a deterministic algorithm that takes as input the public key pk, some circuit C that takes n bits as input and outputs one bit, as well as n ciphertexts $\psi_1, \ldots, \psi_n$. It outputs a ciphertext ψ_C.

Compactness: For all security parameters λ, there exists a polynomial $p(\cdot)$ such that for all input sizes n, for all $x_1, \ldots, x_n$, for all C, the output length of FHE.EVAL is at most $p(n)$ bits long.

Definition 12 (*C-homomorphism, Goldwasser et al. (2013)*)**.** Let $C = \{C_n\}_{n\in\mathbb{N}}$ be a class of boolean circuits, where C_n is a set of boolean circuits taking n bits as input. A scheme FHE is C-homomorphic if for every polynomial $n(\cdot)$, for every sufficiently large security parameter λ, for every circuit $C \in C_n$, and for every input bit sequence $x_1, \ldots, x_n$, where $n = n(\lambda)$,

$$\mathbb{P}\left[\begin{array}{c}(pk, sk) \leftarrow \text{FHE.KEYGEN}(1^\lambda);\\ \psi_i \leftarrow \text{FHE.ENC}(pk, x_i) \text{ for } i = 1 \ldots n;\\ \psi \leftarrow \text{FHE.EVAL}(pk, C, \psi_1, \ldots, \psi_n) :\\ \text{FHE.DEC}(sk, \psi) \neq C(x_1, \ldots, x_n)\end{array}\right] = \text{negl}(\lambda),$$

where the probability is over the coin tosses of FHE.KeyGen and FHE.Enc.

Definition 13 (*Fully homomorphic encryption*)**.** A scheme FHE is fully homomorphic if it is homomorphic for the class of all arithmetic circuits over $\mathbb{GF}(2)$.

Definition 14 (*Leveled fully homomorphic encryption*)**.** A leveled fully homomorphic encryption scheme is a homomorphic scheme where FHE.KeyGen receives an additional input 1^d and the resulting scheme is homomorphic for all depth-d arithmetic circuits over $\mathbb{GF}(2)$.

Definition 15 (*IND-CPA security*)**.** A scheme FHE is IND-CPA secure if for any p.p.t. adversary $\mathcal{A}$,

$$\Big| \mathbb{P}\left[(pk, sk) \leftarrow \text{FHE.KeyGen}(1^\lambda) : \mathcal{A}(pk, \text{FHE.Enc}(pk, 0)) = 1\right] + \\ - \mathbb{P}\left[(pk, sk) \leftarrow \text{FHE.KeyGen}(1^\lambda) : \mathcal{A}(pk, \text{FHE.Enc}(pk, 1)) = 1\right] \Big| = \text{negl}(\lambda).$$

We now state the result of Brakerski, Gentry, and Vaikuntanathan (Brakerski et al., 2012) that shows a leveled fully homomorphic encryption scheme based on a standard assumption in cryptography called Learning with Errors (Regev, 2005):

Theorem 6 (*Fully Homomorphic Encryption, definition from Goldwasser et al. (2013)*)**.** *Assume that there is a constant $0 < \epsilon < 1$ such that for every sufficiently large ℓ, the approximate shortest vector problem gapSVP in ℓ dimensions is hard to approximate to within a $2^{O(\ell^\epsilon)}$ factor in the worst case. Then, for every n and every polynomial $d = d(n)$, there is an IND-CPA secure d-leveled fully homomorphic encryption scheme where encrypting n bits produces ciphertexts of length poly$(n, \lambda, d^{1/\epsilon})$, the size of the circuit for homomorphic evaluation of a function f is size$(C_f) \cdot$ poly$(n, \lambda, d^{1/\epsilon})$ and its depth is depth$(C_f) \cdot$ poly$(\log n, \log d)$.*

G Existence of Transferable Attacks

Learning Theory Preliminaries. For the next lemma, we will consider a slight generalization of learning tasks to the case where there are many valid outputs for a given input. This can be understood as the case of generative tasks. More concretely, we assume that for the input space $\mathcal{X}_n$ the output space is $\mathcal{Y}_n$ instead of $\{0,1\}$. It will always be the case that $\mathcal{X}_n$ and $\mathcal{Y}_n$ are equal to $\{0,1\}^{p(n)}$ for some polynomial p. For a distribution $\mathcal{D}_n$ over $\mathcal{X}_n$ we call a function $h : \mathcal{X}_n \times \mathcal{Y}_n \to \{0,1\}$ an error oracle if the error of a function $f : \mathcal{X}_n \to \mathcal{Y}_n$ is defined as

$$\text{err}(f) := \mathbb{E}_{x \sim \mathcal{D}}[h(x, f(x))],$$

where the randomness of expectation includes the potential randomness of f. The example oracle Ex provides access to samples $(x, y) \in \mathcal{X}_n \times \mathcal{Y}_n$, where $x \sim \mathcal{D}_n$ and $y \in \mathcal{Y}_n$ is some y such that $h(x, y) = 0$.

The following learning task will be crucial for our construction.

Definition 16 (*Lines on a Circle Learning Task $\mathbb{L}^\circ$*)**.** We define $\mathbb{L}^\circ = \{\mathbb{L}^\circ_n\}_n$. For every n we define $\mathcal{X}_n = \{0,1\}^n$ and associate $\mathcal{X}_n$ with vertices of a 2^n regular polygon inscribed in the unit circle $\{x \in \mathbb{R}^2 \mid \|x\|_2 = 1\}$. The output space is $\{-1, +1\}$ for all n. Let $\mathcal{H} := \{h_w \mid w \in \mathbb{R}^2, \|w\|_2 = 1\}$, where $h_w(x) := \text{sgn}(\langle w, x\rangle)$. For every n, let $\mathbb{L}^\circ_n$ be the distribution corresponding to the following process: sample $h_w \sim U(\mathcal{H})$, return $(U(X_n), h_w)$. Note that $\mathcal{H}$ has VC-dimension equal to 2 so $\mathbb{L}$ is learnable to error ϵ with $O(\frac{1}{\epsilon})$ samples for every n and every ϵ.

Moreover, for $n \in \mathbb{N}$ define $B^w_n(\alpha) := \{x \in \mathcal{X}_n \mid |\measuredangle(x, w)| \leq \alpha\}$.

Lemma 1 (*Learning lower bound for $\mathbb{L}^\circ$*)**.** *Let $n \in \mathbb{N}$. Let $\mathbf{L}_n$ be a learning algorithm for $\mathbb{L}^\circ_n$ (Definition 16) that uses K samples and returns a classifier $f : \mathcal{X}_n \to \{-1, +1\}$. Then*

$$\mathbb{P}_{(\mathcal{D}_n, h_n) \sim \mathbb{L}^\circ_n, f \leftarrow \mathbf{L}^{Ex(\mathcal{D}_n, h_n)}}\left[\mathbb{P}_{x \sim \mathcal{D}_n}[f(x) \neq h_w(x)] \leq \frac{1}{2K}\right] \leq \frac{3}{100}.$$

Proof. Let $n \in \mathbb{N}$. Consider the following algorithm $\mathcal{A}$. It first simulates $\mathbf{L}_n$ on K samples to compute f. Next, it performs a smoothing of f, i.e., computes

$$f_\eta(x) := \begin{cases} +1, & \text{if } \mathbb{P}_{x' \sim U(B^x_n(2\pi\eta))}[f(x') = +1] > \mathbb{P}_{x' \sim U(B^x_n(2\pi\eta))}[f(x') = -1] \\ -1, & \text{otherwise.} \end{cases}$$

Note that if $\text{err}(f) \leq \eta$ for a ground truth h_w then for every $x \in \mathcal{X}_n \setminus B_n^x(2\pi\eta)$ we have $f_\eta(x) = h_w(x)$. This implies that $\mathcal{A}$ can be adapted to an algorithm that with probability 1 finds w' such that $|\measuredangle(w, w')| \leq \text{err}(f)$.

Assuming towards contradiction that the statement of the lemma does not hold it means that there is an algorithm using K samples that with probability $\frac{3}{100}$ locates w up to angle $\frac{1}{2K}$.

Consider any algorithm $\mathcal{A}$ using K samples. Probability that $\mathcal{A}$ does not see any sample in $B_n^w(2\pi\eta)$ is at least

$$(1-4\eta)^K \geq \left((1-4\eta)^{\frac{1}{4\eta}}\right)^{4\eta K} \geq \left(\frac{1}{2e}\right)^{4\eta K},$$

which is bigger than $1 - \frac{3}{100}$ if we set $\eta = \frac{1}{2K}$. But note that if there is no sample in $B_n^w(2\pi\eta)$ then $\mathcal{A}$ cannot locate w up to η with certainty. This proves the lemma. □

Lemma 2 (*Boosting for $\mathbb{L}^\circ$*)**.** *Let $\eta, \nu \in (0, \frac{1}{4}), n \in \mathbb{N}$, $\mathbf{L}_n$ be a learning algorithm for $\mathbb{L}_n^\circ$ that uses K samples and outputs $f : \mathcal{X}_n \to \{-1, +1\}$ such that with probability δ*

$$\mathbb{P}_{w\sim U(\mathcal{H}), x\sim U(B_n^w(2\pi\eta))}[f(x) \neq h_w(x)] \leq \nu, \tag{5}$$

where $\mathcal{H}$ is as defined earlier $\{h_w \mid w \in \mathbb{R}^2, \|w\|_2 = 1\}$. Then there exists a learning algorithm $\mathbf{L}'_n$ for $\mathbb{L}_n^\circ$ that uses $\max\left(K, \frac{9}{\eta}\right)$ samples such that with probability $\delta - \frac{1}{1000}$ returns f' such that

$$\mathbb{P}_{w\sim U(\mathcal{H}), x\sim U(\mathcal{X}_n)}[f'(x) \neq h_w(x)] \leq 4\eta\nu.$$

Proof. Let $n \in \mathbb{N}$. $\mathbf{L}'_n$ first draws $\max\left(K, \frac{9}{\eta}\right)$ samples Q and defines $g : \mathcal{X}_n \to \{-1, +1, \perp\}$ as follows, g maps to -1 the smallest continuous interval containing all samples from Q with label -1. Similarly g maps to $+1$ the smallest continuous interval containing all samples from Q with label $+1$. The intervals are disjoined by construction. Unmapped points are mapped to $\perp$. Next, $\mathbf{L}'_n$ simulates $\mathbf{L}_n$ with K samples and gets a classifier f that with probability δ satisfies the assumption of the lemma. Finally, it returns

$$f'(x) := \begin{cases} g(x), & \text{if } g(x) \neq \perp \\ f(x), & \text{otherwise.} \end{cases}$$

Consider 4 arcs defined as the 2 arcs constituting $B_n^w(2\pi\eta)$ divided into 2 parts each by the line $\{x \in \mathbb{R}^2 \mid \langle w, x\rangle = 0\}$. Let E be the event that some of these intervals do not contain a sample from Q. Observe that

$$\mathbb{P}[E] \leq 4(1-\eta)^{\frac{9}{\eta}} \leq \frac{1}{1000}.$$

By the union bound with probability $\delta - \frac{1}{1000}$, f satisfies equation (5) and E does not happen. By definition of f' this gives the statement of the lemma. □

Theorem 7 (*Transferable Attack for a Cryptography based Learning Task*)**.** *There exists a learning task $\mathbb{L} = \{\mathbb{L}_\lambda\}_\lambda$ and a function class $\mathcal{F} = \{\mathcal{F}_\lambda\}_\lambda$ such that for every $\epsilon : \mathbb{N} \to \mathbb{N}$ where $1/\epsilon(\lambda)$ is lower-bounded by a sufficiently large polynomial and upper-bounded by some polynomial the following holds.*

1. *$\mathbb{L}$ is $\left(\epsilon, \delta = \frac{1}{10}, S = \frac{10^3}{\epsilon^{1.3}}, \mathcal{F}\right)$-learnable.*
2. *$\mathbb{L}$ is **not** $\left(\epsilon, \delta = \frac{1}{10}, S = \frac{1}{\epsilon}, \mathcal{F}\right)$-learnable*
3. *There exists a circuit family $\mathbf{A} = \{\mathbf{A}_\lambda\}_\lambda$ such that*

$$\mathbf{A} \in_1 \textsc{TransfAttack}\left(\mathbb{L}, \mathcal{F}, \epsilon(\lambda), q(\lambda) = \frac{16}{\epsilon(\lambda)}, S_{\mathbf{A}}(\lambda) = \frac{10^3}{\epsilon^{1.3}(\lambda)}, S_{\mathbf{B}}(\lambda) = \frac{1}{10^2\epsilon^2(\lambda)}, c = \frac{9}{10}, s = \textit{negl}(\lambda)\right).$$

Proof. The learning task is based on $\mathbb{L}^\circ = \{\mathbb{L}_n^\circ\}_n$ from Definition 16.

Setting of Parameters for FHE. Observe that by assumption of the lemma $p \leq 1/\epsilon \leq r$, for some polynomial r, and a polynomial p that we will define later. Let FHE be a fully homomorphic encryption scheme from Theorem 6. We will use the scheme for constant leveled circuits $d = O(1)$. Let $s(n, \lambda, d)$ be the polynomial bounding the size of the encryption of inputs of length n with λ security as well as bounding the size of the circuit for homomorphic evaluation, which is guaranteed to exist by Theorem 6. Let $\beta \in (0, 1)$ and p be a polynomial such that

$$s\left(n^{\beta}, \lambda, d\right) \leq (n \cdot p(\lambda))^{0.1}, \tag{6}$$

which exist because s is a polynomial.

We define $n(\lambda) := \lfloor p^{1/\beta}(\lambda) \rfloor$[8] for the length of inputs in the FHE scheme. Observe that for every λ

$$\begin{aligned} s(n(\lambda), \lambda, d) &\leq (p(\lambda) \cdot p(\lambda))^{0.1} && \text{By equation (6)} \\ &\leq \frac{1}{\epsilon(\lambda)^{0.2}} && \text{By } \epsilon(\lambda) \in \left(\frac{1}{r(\lambda)}, \frac{1}{p(\lambda)}\right). \end{aligned} \tag{7}$$

Learning Task. The learning task will be parametrized by λ, i.e. $\mathbb{L} = \{\mathbb{L}_\lambda\}_\lambda$.

Let $\lambda \in \mathbb{N}$. We define $\mathbb{D}_\lambda := \{\mathcal{D}_\lambda^{(\text{pk,sk})}\}_{(\text{pk,sk})}, \mathcal{H}_\lambda := \{h_\lambda^{(\text{pk,sk,w})}\}_{(\text{pk,sk,w})}$ (for $\mathcal{D}_\lambda^{(\text{pk,sk})}$ and $h_\lambda^{(\text{pk,sk,w})}$ to be defined later), where they are indexed by valid public/secret key pairs of the FHE and $w \in \{x \in \mathbb{R}^2 \mid \|x\|_2 = 1\}$. Let $\mathbb{L}_\lambda$ be defined as corresponding to the following process: sample $(\text{pk,sk}, w) \sim \text{FHE.KeyGen}(1^\lambda) \times U(\{x \in \mathbb{R}^2 \mid \|x\|_2 = 1\})$, return $\left(\mathcal{D}_\lambda^{(\text{pk,sk})}, h_\lambda^{(\text{pk,sk,w})}\right)$.

For a valid (pk,sk) pair we define $\mathcal{D}^{(\text{pk,sk})}$ as the result of the following process: $x \sim U(\{0,1\}^{n(\lambda)})$, with probability $\frac{1}{2}$ return $(0, x, \text{pk})$ and with probability $\frac{1}{2}$ return $(1, \text{FHE.Enc}(\text{pk}, x), \text{pk})$, where the first element of the triple describes if the x is encrypted or not. Formally, in the case that the first element of the triple is 0 one needs to add a padding of size $s(n(\lambda), \lambda, d) - n(\lambda)$ so that descriptions have the same size in both cases.[9]

For a valid (pk,sk) pair and $w \in \{x \in \mathbb{R}^2 \mid \|x\|_2 = 1\}$ we define $h^{(\text{pk,sk,w})}((b, x, \text{pk}), y)$ as a result of the following algorithm: if $b = 0$ return $\mathbb{1}_{h_w(x)=y}$, otherwise let $x_{\text{Dec}} \leftarrow \text{FHE.Dec}(\text{sk}, x)$, $y_{\text{Dec}} \leftarrow \text{FHE.Dec}(\text{sk}, y)$ and if $x_{\text{Dec}}, y_{\text{Dec}} \neq \perp$ (decryption is succesful) return $\mathbb{1}_{h_w(x_{\text{Dec}})=y_{\text{Dec}}}$ and return 1 otherwise.

Note 1 ($\Omega(\frac{1}{\epsilon})$*-sample learning lower bound.*)**.** *By construction any learner using K samples for $\mathbb{L}_\lambda$ (for any λ) can be transformed (potentially computationally inefficiently) into a learner using K samples for $\mathbb{L}^\circ_{n(\lambda)}$ (Defnition 16) that returns a classifier of the same error. This, together with a lower bound for learning from Lemma 1 proves point 2 of the lemma.*

Definition of A (Algorithm 1). $\mathbf{A}_\lambda$ draws $N(\lambda)$ samples $Q = \{((b_i, x_i, \text{pk}), y_i)\}_{i \in [N]}$ for $N(\lambda) := \frac{900}{\epsilon(\lambda)}$.

Next, $\mathbf{A}_\lambda$ chooses a subset $Q_{\text{CLEAR}} \subseteq Q$ of samples for which $b_i = 0$. It trains a classifier $f_{w'}(\cdot) := \text{sgn}(\langle w', \cdot \rangle)$ on Q_{CLEAR} by returning any $f_{w'}$ consistent with Q_{CLEAR}. This can be done in time

$$N(\lambda) \cdot n(\lambda) \leq \frac{900}{\epsilon(\lambda)} \cdot p^{1/\beta}(\lambda) \leq \frac{900}{\epsilon^{1.1}(\lambda)} \tag{8}$$

by keeping track of the smallest interval containing all samples in Q_{CLEAR} labeled with $+1$ and then returning any $f_{w'}$ consistent with this interval.

Note 2 ($O(\frac{1}{\epsilon^{1.3}})$*-time learning upper bound.*)**.** *First note that $\mathbf{A}_\lambda$ learns well, i.e., with probability at least $1 - 2\left(1 - \frac{\epsilon(\lambda)}{100}\right)^{\frac{900}{\epsilon(\lambda)}} \geq 1 - \frac{1}{1000}$ we have that*

$$|\measuredangle(w, w')| \leq \frac{2\pi\epsilon(\lambda)}{100} \tag{9}$$

[8]Note that this setting allows to represent points in $\{x \in \mathbb{R}^2 \mid \|x\|_2 = 1\}$ up to $2^{-p^{1/\beta}(\lambda)}$ precision and this precision is better than $\frac{1}{r(\lambda)}$ for every polynomial r for sufficiently large λ. This implies that this precision is enough to allow for learning up to error ϵ, because of the setting $\epsilon(\lambda) \geq \frac{1}{r(\lambda)}$.

[9]Note that the domain of the distributions is not $\{0,1\}^\lambda$, i.e. $\mathcal{X}_\lambda \neq \{0,1\}^\lambda$.

Algorithm 1 TRANSFATTACK$(\text{Ex}(\mathcal{D}_\lambda, h_\lambda), \epsilon, \lambda)$

Input: Access to the example oracle $\text{Ex}(\mathcal{D}_\lambda, h_\lambda)$, where $(\mathcal{D}_\lambda, h_\lambda) \sim \mathbb{L}_\lambda$, error level $\epsilon : \mathbb{N} \to \mathbb{N}$, and the security parameter λ.

$N := 900/\epsilon(\lambda), q := 16/\epsilon(\lambda)$

$Q = \{((b_i, x_i, \text{pk}), y_i)\}_{i\in[N]} \sim (\mathcal{D}_\lambda)^{N(\lambda)}$ ▷ $N(\lambda)$ i.i.d. samples from $\mathcal{D}_\lambda$

$Q_{\text{CLEAR}} = \{((b, x, \text{pk}), y) \in Q : b = 0\}$ ▷ $Q_{\text{CLEAR}} \subseteq Q$ of unencrypted x's

$f_{w'}(\cdot) := \text{sgn}(\langle w', \cdot\rangle) \leftarrow$ a line consistent with samples from Q_{CLEAR} ▷ $f_{w'} : \mathcal{X}_n \to \{-1, +1\}$

$\{x'_i\}_{i\in[q(\lambda)]} \sim U\left(\left(\mathcal{X}_{n(\lambda)}\right)^{q(\lambda)}\right)$

$S \sim U(2^{[q(\lambda)]})$ ▷ $S \subseteq [q(\lambda)]$ a uniformly random subset

$E_{\text{BND}}; = \emptyset$

for $i \in [q(\lambda) - |S|]$ **do**

  $x_{\text{BND}} \sim U(B^{w'}_{n(\lambda)}(2\pi(\epsilon(\lambda) + \frac{\epsilon(\lambda)}{100})))$ ▷ x_{BND} is close to the decision boundary of $f_{w'}$

  $E_{\text{BND}} := E_{\text{BND}} \cup \{\text{FHE.ENC}(\text{pk}, x_{\text{BND}})\}$

end for

$\mathbf{x} := \{(0, x'_i, \text{pk}) \mid i \in [q(\lambda)] \setminus S\} \cup \{(1, x', \text{pk}) \mid x' \in E_{\text{BND}}\}$

Return $\mathbf{x}$

Moreover, $f_{w'}(x)$ can be implemented by a circuit $C_{f_{w'}}$ that compares x with the endpoints of the interval. This can be done by a constant leveled circuit. Moreover $C_{f_{w'}}$ can be evaluated with FHE.EVAL *in time*

$$\textit{size}(C_{f_{w'}})s(n(\lambda), \lambda, d) \leq 10n \cdot s(n(\lambda), \lambda, d) \leq 10p^{1/\beta}(\lambda)s(n(\lambda), \lambda, d) \leq \frac{10}{\epsilon^{0.3}(\lambda)},$$

where the last inequality follows from equation (7). This proves point 1 of the lemma.

Next, $\mathbf{A}_\lambda$ prepares $\mathbf{x}$ as follows. It samples $q(\lambda) = \frac{16}{\epsilon(\lambda)}$ points $\{x'_i\}_{i\in[q]}$ from $\{0,1\}^{n(\lambda)}$ uniformly at random. It chooses a uniformly random subset $S \subseteq [q(\lambda)]$. Next, $\mathbf{A}_\lambda$ generates $q(\lambda) - |S|$ inputs using the following process: $x_{\text{BND}} \sim U(B^{w'}_{n(\lambda)}(2\pi(\epsilon(\lambda) + \frac{\epsilon(\lambda)}{100})))$ (x_{BND} is close to the decision boundary of $f_{w'}$), return $\text{FHE.ENC}(\text{pk}, x_{\text{BND}})$. Call the set of $q(\lambda) - |S|$ points E_{BND}. $\mathbf{A}_\lambda$ defines:

$$\mathbf{x} := \{(0, x'_i, \text{pk}) \mid i \in [q] \setminus S\} \cup \{(1, x', \text{pk}) \mid x' \in E_{\text{BND}}\}.$$

The running time of this phase is dominated by evaluations of FHE.EVAL, which takes

$$q(\lambda) \cdot s(n(\lambda), \lambda, d) \leq \frac{16}{\epsilon(\lambda)} \cdot \frac{1}{\epsilon^{0.2}(\lambda)} \leq \frac{16}{\epsilon^{1.2}(\lambda)}, \tag{10}$$

where the first inequality follows from equation (7). Taking the sum of equation (8) and equation (10) we get that $\mathbf{A}_\lambda$ can be implemented by a circuit of size $\frac{10^3}{\epsilon^{1.3}(\lambda)}$.

$\mathbf{A}_\lambda$ Constitutes a Transferable Attack. Now, consider $\mathbf{B}_\lambda$ of size $S_{\mathbf{B}}(\lambda) = \frac{1}{\epsilon^2(\lambda)}$. By the assumption $S_{\mathbf{B}}(\lambda) \leq r(\lambda)$, which implies that the security guarantees of FHE hold for $\mathbf{B}_\lambda$.

We claim that $\mathbf{x}$ is indistinguishable from $\mathcal{D}^{(\text{pk},\text{sk})}_\lambda$ for $\mathbf{B}_\lambda$. Observe that by construction the distribution of ratio of encrypted and not encrypted x's in $\mathbf{x}$ is identical to that of $\mathcal{D}^{(\text{pk},\text{sk})}_\lambda$. Moreover, the distribution of unencrypted x's is identical to that of $\mathcal{D}^{(\text{pk},\text{sk})}_\lambda$ by construction. Finally, by the IND-CPA security[10] of FHE and the fact that the size of $\mathbf{B}_\lambda$ is bounded by some polynomial in λ we have that $\text{FHE.ENC}(\text{pk}, x_{\text{BND}})$ is distinguishable from $x \sim \mathcal{X}_n, \text{FHE.ENC}(\text{pk}, x)$ with advantage at most $\text{negl}(\lambda)$. Thus *undetectability* holds with near perfect soundness $s = \frac{1}{2} + \text{negl}(\lambda)$.

Next, we claim that $\mathbf{B}_\lambda$ can't return low-error answers on $\mathbf{x}$.

[10]Note that we need security of FHE in the nonuniform model of computation.

Assume towards contradiction that with probability $\frac{5}{100}$

$$\mathbb{P}_{\substack{w\sim U(\{z\in\mathbb{R}^2 \mid \|z\|_2=1\}),\\ x\sim U(B^w_{n(\lambda)}(2\pi\epsilon(\lambda)))}}[f(x)\neq h_w(x)]\leq 10\epsilon(\lambda). \tag{11}$$

We can apply Lemma 2 to get that there exists a learner using $\frac{1}{100\epsilon^2(\lambda)}+\frac{9}{\epsilon(\lambda)}\leq\frac{1}{90\epsilon^2(\lambda)}$ samples that with probability $\frac{4}{100}$ returns f' such that

$$\mathbb{P}_{\substack{w\sim U(\{z\in\mathbb{R}^2 \mid \|z\|_2=1\}),\\ x\sim U(\{0,1\}^{n(\lambda)})}}[f'(x)\neq h_w(x)]\leq 40\epsilon^2(\lambda). \tag{12}$$

Applying Lemma 1 to equation (12) we know that

$$40\epsilon^2\geq\frac{1}{2(\frac{1}{90\epsilon^2(\lambda)})},$$

which is a contradiction. Thus equation (11) does not hold and in consequence using equation (9) we have that with probability $1-\frac{6}{100}$

$$\mathbb{P}_{\substack{w\sim U(\{z\in\mathbb{R}^2 \mid \|z\|_2=1\}),\\ x\sim U(B^{w'}_{n(\lambda)}(2\pi(\epsilon(\lambda)+\frac{\epsilon(\lambda)}{10})))}}[f(x)\neq h_w(x)]\geq\frac{10}{14}\cdot 10\epsilon(\lambda)\geq 7\epsilon(\lambda), \tag{13}$$

where crucially x is sampled from $U(B^{w'}_{n(\lambda)})$ and not $U(B^w_{n(\lambda)})$. By Fact 1 we know that $|S|\geq\frac{q(\lambda)}{3}$ with probability at least

$$1-2e^{-\frac{q(\lambda)}{72}}=1-2e^{-\frac{1}{8\epsilon(\lambda)}}\geq 1-\frac{1}{1000}.$$

Using the setting of $q(\lambda)=\frac{16}{\epsilon(\lambda)}$ and applying the Chernoff bound and the union bound we get from equation (13) that with probability at least $1-\frac{1}{10}$ the error $\mathrm{err}(\mathbf{x},\mathbf{y})$ is larger than $2\epsilon(\lambda)$.

□

Note 3. *We want to emphasize that it is crucial (for our construction) that the distribution has both an encrypted and an unencrypted part.*

As mentioned before, if there was no $\mathcal{D}_{\text{CLEAR}}$ then $\mathbf{A}_\lambda$ would see only samples of the form

$$(\text{FHE.ENC}(x),\text{FHE.ENC}(y))$$

and would not know which of them lie close to the boundary of h_w, and so it would not be able to choose tricky samples. $\mathbf{A}_\lambda$ would be able to learn a low-error classifier, but only *under the encryption. More concretely, $\mathbf{A}_\lambda$ would be able to homomorphically evaluate a circuit that, given a training set and a test point, learns a good classifier and classifies the test point with it. However, it would* not *be able to, with high probability, generate* $\text{FHE.ENC}(x)$*, for x close to the boundary as it would not know (in the clear) where the decision boundary is.*

If there was no $\mathcal{D}_{\text{ENC}}$ then everything would happen in the clear and so $\mathbf{B}$ would be able to distinguish x's that appear too close to the boundary.

Fact 1 (*Chernoff-Hoeffding*)**.** Let $X_1,\ldots,X_k$ be independent Bernoulli variables with parameter p. Then for every $0<\epsilon<1$

$$\mathbb{P}\left[\left|\frac{1}{k}\sum_{i=1}^k X_i-p\right|>\epsilon\right]\leq 2e^{-\frac{\epsilon^2 k}{2}}$$

and

$$\mathbb{P}\left[\frac{1}{k}\sum_{i=1}^k X_i\leq(1-\epsilon)p\right]\leq e^{-\frac{\epsilon^2 kp}{2}}.$$

Also for every $\delta>0$

$$\mathbb{P}\left[\frac{1}{k}\sum_{i=1}^k X_i>(1+\delta)p\right]\leq e^{-\frac{\delta^2 kp}{2+\delta}}$$

H TRANSFERABLE ATTACKS IMPLY CRYPTOGRAPHY

H.1 EFID PAIRS

The typical way in which security of EFID pairs is defined, e.g., in (Goldreich, 1990), is that they should be secure against all polynomial-time algorithms. However, for the case of pseudorandom generators (PRGs), which are known to be equivalent (in the standard definition) to EFIDs pairs, more granular notions of security were considered. For instance, in (Nisan, 1990) the existence of PRGs secure against adversaries running in time bounded by a fixed, in contrast to all, polynomial, was studied. In a similar spirit, we consider EFID pairs that are secure against adversaries with fixed circuit complexity bounds.

Definition 17 (*Total Variation*)**.** For two distrbutions $\mathcal{D}_0, \mathcal{D}_1$ over a finite domain $\{0,1\}^n$ we define their *total variation distance* as

$$\triangle(\mathcal{D}_0, \mathcal{D}_1) := \sum_{x \in \{0,1\}^n} \frac{1}{2} |\mathcal{D}_0(x) - \mathcal{D}_1(x)|.$$

Definition 18 (*EFID pairs*)**.** For parameters $\eta, \delta : \mathbb{N} \to (0,1)$ and circuit complexity bounds $S, S' : \mathbb{N} \to \mathbb{N}$ we call a pair of ensembles of distributions $(\mathcal{D}^0 = \{\mathcal{D}^0_n\}_n, \mathcal{D}^1 = \{\mathcal{D}^1_n\}_n)$ over domain $\mathcal{X} = \{\mathcal{X}_n\}_n$ an (S, S', η, δ)-EFID pair if for every n

1. The circuit complexity of sampling $\mathcal{D}^0$ and $\mathcal{D}^1$ is at most S,
2. For every n we have that $\triangle(\mathcal{D}^0_n, \mathcal{D}^1_n) \geq \eta(n)$,
3. For every n we have that $\mathcal{D}^0_n, \mathcal{D}^1_n$ are $\delta(n)$-indistinguishable for circuits with complexity $S'(n)$.

Observe that Definition 18 is a generalization of the standard definition. Indeed, for every EFID pair $(\mathcal{D}^0, \mathcal{D}^1)$ according to the standard definition there exists an inverse polynomial function η and a polynomial S such that for all polynomials S' there exists a negligible function δ such that $(\mathcal{D}^0, \mathcal{D}^1)$ is an (S, S', η, δ)-EFID pair.

H.2 TRANSFERABLE ATTACKS IMPLY EFID PAIRS

Theorem 8 (*Tasks with Transferable Attacks Imply EFID pairs*)**.** *For every $\epsilon \in (0,1), q \in \mathbb{N}, S_{\mathbf{A}}, S_{\mathbf{B}} : \mathbb{N} \to \mathbb{N}$ such that $S_{\mathbf{A}} \leq S_{\mathbf{B}}$, every learning task $\mathbb{L}$ learnable to error ϵ with confidence p and circuit complexity $S_{\mathbf{A}}$, every $c, s \in (0,1)$ if*

$$\textsc{TransfAttack}\left(\mathbb{L}, \epsilon, q, S_{\mathbf{A}}, S_{\mathbf{B}}, c, s\right)$$

exists with frequency $\frac{1}{3}$ then there exist $S'_{\mathbf{A}}, S'_{\mathbf{B}} : \mathbb{N} \to \mathbb{N}$ that agree with $S_{\mathbf{A}}$ and $S_{\mathbf{B}}$ respectively with frequency $\frac{1}{3}$ and there exists

$$\left(S'_{\mathbf{A}}, S'_{\mathbf{B}}, \frac{1}{2}\left(p + c - 1 - e^{-\frac{\epsilon q}{3}}\right), \frac{s}{2}\right) - \textit{EFID pair.}$$

Proof. Let $\epsilon, S_{\mathbf{A}}, S_{\mathbf{B}}, q, c, s, p, \mathbb{L}$ be as in the assumption of the theorem. Additionally let $\mathbf{A} = \{\mathbf{A}_n\}_n$ be a family of circuits certifying that a Transferable Attack exists with frequency $\frac{1}{3}$ for $\mathbb{L}$.

For every n, define $\mathcal{D}^0_n := \mathcal{D}^q_n$, where we recall that q is the number of samples $\mathbf{A}_n$ sends in the attack. Define $\mathcal{D}^1_n$ to be the distribution of $\mathbf{x} := \mathbf{A}_n$. Note that $\mathbf{x} \in (\mathcal{X}_n)^q$.

Let $a : \mathbb{N} \to \{0,1\}$ be a sequence certifying that a Transferable Attack exists with frequency $\frac{1}{3}$. Let n be such that $a(n) = 1$. Observe that $\mathcal{D}^0_n, \mathcal{D}^1_n$ are samplable with circuit complexity $S_{\mathbf{A}}(n)$ because $\mathbf{A}_n$ complexity is bounded by $S_{\mathbf{A}}(n)$. Secondly, $\mathcal{D}^0_n, \mathcal{D}^1_n$ are $\frac{s}{2}$-indistinguishable for $S_{\mathbf{B}}(n)$-sized adversaries by *undetectability* of $\mathbf{A}_n$. Finally, the fact that $\mathcal{D}^0_n, \mathcal{D}^1_n$ are statistically far follows from *transferability*. Indeed, the following procedure accepting input $\mathbf{x} \in (\{0,1\}^n)^q$ is a distinguisher:

1. Run the learner (the existence of which is guaranteed by the assumption of the theorem) to obtain f.

2. $\mathbf{y} := f(\mathbf{x})$.
3. If $\text{err}(\mathbf{x}, \mathbf{y}) \leq 2\epsilon$ return 0, otherwise return 1.

If $\mathbf{x} \sim \mathcal{D}^0 = \mathcal{D}^q$ then $\text{err}(f) \leq \epsilon$ with probability p. By Fact 1 and the union bound we also know that $\text{err}(\mathbf{x}, \mathbf{y}) \leq 2\epsilon$ with probability $p - e^{-\frac{\epsilon q}{3}}$ and so, the distinguisher will return 0 with probability $p - e^{-\frac{\epsilon q}{3}}$. On the other hand, if $\mathbf{x} \sim \mathcal{D}^1 = \mathbf{A}$ we know from *transferability* of $\mathbf{A}_n$ that every algorithm running in time $S_{\mathbf{B}}(n)$ will return $\mathbf{y}$ such that $\text{err}(\mathbf{x}, \mathbf{y}) > 2\epsilon$ with probability at least c. By the assumption that $S_{\mathbf{B}}(n) \geq S_{\mathbf{A}}(n)$ we know that $\text{err}(\mathbf{x}, f(\mathbf{x})) > 2\epsilon$ with probability at least c also. Consequently, the distinguisher will return 1 with probability at least c in this case. By the properties of total variation this implies that $\triangle(\mathcal{D}^0_n, \mathcal{D}^1_n) \geq \frac{1}{2}(p + c - 1 - e^{-\frac{\epsilon q}{3}})$.

We define a pair of families of distributions $\widehat{\mathcal{D}}^0, \widehat{\mathcal{D}}^1$ and functions $S'_{\mathbf{A}}, S'_{\mathbf{B}}$ as follows. For every n such that $a(n) = 1$ we define $\widehat{\mathcal{D}}^0_n = \mathcal{D}^0_n, \widehat{\mathcal{D}}^1 = \mathcal{D}^1_n, S'_{\mathbf{A}}(n) = S_{\mathbf{A}}(n), S'_{\mathbf{B}}(n) = S_{\mathbf{B}}(n)$. For every n sich that $a(n) = 0$ we define $\widehat{\mathcal{D}}^0_n = \mathcal{D}^0_k$ for the smallest $k > n$ such that $a(k) = 1$, and $S'_{\mathbf{A}}(n) = S_{\mathbf{A}}(k)$ And analogously for $\widehat{\mathcal{D}}^1_n$ and $S'_{\mathbf{B}}$.

Simple verification yields that $\widehat{\mathcal{D}}^0_n, \widehat{\mathcal{D}}^1_n$ is an $(S'_{\mathbf{A}}, S'_{\mathbf{B}}, \frac{1}{2}(p + c - 1 - e^{-\frac{\epsilon q}{3}}), \frac{s}{2})$-EFID pair.

Note 4 (*Setting of parameters*)**.** *Observe that if $p \approx 1$, i.e., it is possible to almost surely learn f in time $S_{\mathbf{A}}$ such that err$(f) \leq \epsilon$, c is a constant, $q = \Omega(\frac{1}{\epsilon})$ then η in the parameters for the EFID is a constant and so $\triangle(\mathcal{D}^0, \mathcal{D}^1)$ is a constant.*

Note 5. *We want to emphasize that our distinguisher crucially uses the error oracle in its last step. So it is possible that it is not implementable for all circuit complexity bounds!*

□

I Adversarial Defenses exist

Our result is based on (Goldwasser et al., 2020). Before we state and prove our result we give an overview of the learning model considered in (Goldwasser et al., 2020). The authors give a defense against *arbitrary examples* in a transductive model with rejections. In contrast, our model does not allow rejections, but we do require indistinguishability.

I.1 Transductive Learning With Rejections.

In (Goldwasser et al., 2020) the authors consider a model, where a learner $\mathbf{L}$ receives a training set of labeled samples from the original distribution $(\mathbf{x}_{\mathcal{D}}, \mathbf{y}_{\mathcal{D}} = h(\mathbf{x}_{\mathcal{D}})), \mathbf{x} \sim \mathcal{D}^N, \mathbf{y}_{\mathcal{D}} \in \{-1, +1\}^N$, where h is the ground truth, together with a test set $\mathbf{x}_T \in (\{0,1\}^n)^q$. Next, $\mathbf{L}$ uses $(\mathbf{x}_{\mathcal{D}}, \mathbf{y}_{\mathcal{D}}, \mathbf{x}_T)$ to compute $\mathbf{y}_T \in \{-1, +1, \square\}^q$, where $\square$ represents that $\mathbf{L}$ abstains (rejects) from classifying the corresponding x.

Before we define when learning is successful, we will need some notation. For $q \in \mathbb{N}, \mathbf{x} \in \{0,1\}^{nq}, \mathbf{y} \in \{-1, +1, \square\}^q$ we define

$$\text{err}(\mathbf{x}, \mathbf{y}) := \frac{1}{q} \sum_{i \in [q]} \mathbb{1}_{\left\{h(x_i) \neq y_i, y_i \neq \square, h(x_i) \neq \perp\right\}}, \quad \square(\mathbf{y}) := \frac{1}{q} \left| \left\{ i \in [q] : y_i = \square \right\} \right|,$$

which means that we count $(x, y) \in \{0,1\}^n \times \{-1, +1, \square\}$ as an error if h is well defined on x, y is not an abstantion and $h(x) \neq y$.

Learning is successful if it satisfies two properties.

- If $\mathbf{x}_T \sim \mathcal{D}^q$ then with high probability $\text{err}(\mathbf{x}_T, \mathbf{y}_T)$ and $\square(\mathbf{y}_T)$ are small.
- For *every* $\mathbf{x}_T \in \{0,1\}^{nq}$ with high probability $\text{err}(\mathbf{x}_T, \mathbf{y}_T)$ is small.[11]

[11] Note that, crucially, in this case $\square(\mathbf{y}_T)$ might be very high, e.g., equal to 1.

The formal guarantee of a result from Goldwasser et al. (2020) are given in Theorem 9. Let us call this model Transductive Learning with Rejections (TLR).

Note the differences between TLR and our definition of Adversarial Defenses. To compare the two models we associate the learner $\mathbf{L}$ from TLR with $\mathbf{B}$ in our setup, and the party producing $\mathbf{x}_T$ with $\mathbf{A}$ in our definition. First, in TLR, $\mathbf{B}$ does not send f to $\mathbf{A}$. Secondly, and most importantly, we do not allow $\mathbf{B}$ to reply with rejections ($\square$) but instead require that $\mathbf{B}$ can "distinguish" that it is being tested (see soundness of Definition 9). Finally, there are no apriori time bounds on either $\mathbf{A}$ or $\mathbf{B}$ in TLR. The models are similar but a priori incomparable and any result for TLR needs to be carefully analyzed before being used to prove that it is an Adversarial Defense.

I.2 Formal guarantee for Transductive Learning with Rejections (TLR)

Theorem 5.3 from Goldwasser et al. (2020) adapted to our notation reads.

Theorem 9 (*TLR guarantee (Goldwasser et al. (2020))*). *For any $N \in \mathbb{N}, \epsilon \in (0,1), h \in \mathcal{H}$ and distribution $\mathcal{D}$ over $\{0,1\}^n$:*

$$\mathbb{P}_{\mathbf{x}_{\mathcal{D}}, \mathbf{x}'_{\mathcal{D}} \sim \mathcal{D}^N}\left[\forall\, \mathbf{x}_T \in \{0,1\}^{nN} : \mathit{err}(\mathbf{x}_T, f(\mathbf{x}_T)) \leq \epsilon^* \wedge \square\left(f\left(\mathbf{x}'_{\mathcal{D}}\right)\right) \leq \epsilon^*\right] \geq 1-\epsilon,$$

where $\epsilon^* = \sqrt{\frac{2d}{N}\log(2N) + \frac{1}{N}\log\left(\frac{1}{\epsilon}\right)}$ and $f = \textsc{Rejectron}(\mathbf{x}_{\mathcal{D}}, h(\mathbf{x}_{\mathcal{D}}), \mathbf{x}_T, \epsilon^*)$, where $f : \{0,1\}^n \to \{-1, +1, \square\}$ and d denotes the VC-dimension on $\mathcal{H}$. Rejectron *is defined in Figure 2. in (Goldwasser et al., 2020).*

Rejectron is an algorithm that accepts a labeled training set $(\mathbf{x}_{\mathcal{D}}, h(\mathbf{x}_{\mathcal{D}}))$ and a test set $\mathbf{x}_T$ and returns a classifier f, which might reject some inputs. The learning is successful if with a high probability f rejects a small fraction of $\mathcal{D}^N$ and for every $\mathbf{x}_T \in \{0,1\}^{nN}$ the error on labeled x's in $\mathbf{x}_T$ is small.

I.3 Adversarial Defense for bounded VC-dimension

We are ready to state the main result of this section.

Lemma 3 (*Adversarial Defense for bounded VC-dimension*). *Let $\{\mathcal{H}_n\}_n$ be a family of hypothesis classes such that there exists a polynomial p such that for every n, $\mathcal{H}_n$ has a VC-dimension bounded by $p(n)$. There exists a family of circuits $\mathbf{B} = \{\mathbf{B}_n\}_n$ such that for every $\mathbb{L}$ satisfying for every n that the support of the marginal of $\mathbb{L}_n$ is contained in $\mathcal{H}_n$, i.e., the ground truth sampled from $\mathbb{L}$ are always in $\mathcal{H}$, such that*

$$\mathbf{B} \in_1 \textsc{Defense}\left(\mathbb{L}, \epsilon, q = \frac{\mathtt{poly}(n)}{\epsilon^3}, S_{\mathbf{A}} = \infty, S_{\mathbf{B}} = \mathtt{poly}\left(\frac{n}{\epsilon}\right), l = 1-\epsilon, c = 1-\epsilon, s = \epsilon\right).$$

Note that, by the PAC learning bound, this is a setting of parameters, where $\mathbf{B}$ has enough time to learn a classifier of error ϵ. By slightly abusing the notation, we write $S_{\mathbf{A}} = \infty$, meaning that the defense is secure against *all* adversaries regardless of their running time.

Proof. The proof is based on an algorithm from Goldwasser et al. (2020).

Construction of B. Let $\epsilon \in (0,1), n \in N$, $d(n)$ be the VC-dimension of $\mathcal{H}_n$ and

$$N := \frac{d\log^2(d)}{\epsilon^3}.$$

Let $q := N$. First, $\mathbf{B}$, draws N labeled samples $(\mathbf{x}_{\text{FRESH}}, h(\mathbf{x}_{\text{FRESH}}))$. Next, it finds $f \in \mathcal{H}$ consistent with them and sends f to $\mathbf{A}$. Importantly this computation is the same as the first step of Rejectron.

Next, $\mathbf{B}$ receives as input $\mathbf{x} \in \{0,1\}^{nq}$ from $\mathbf{A}$. $\mathbf{B}$. Let $\epsilon^* := \sqrt{\frac{2d}{N}\log(2N) + \frac{1}{N}\log\left(\frac{1}{\epsilon}\right)}$. Next $\mathbf{B}$ runs $f' = \textsc{Rejectron}(\mathbf{x}_{\text{FRESH}}, h(\mathbf{x}_{\text{FRESH}}), \mathbf{x}, \epsilon^*)$, where Rejectron is starting from the second step of the algorithm (Figure 2 (Goldwasser et al., 2020)). Importantly, for every $x \in \{0,1\}^n$, if $f'(x) \neq \square$ then $f(x) = f'(x)$. In words, f' is equal to f everywhere where f' does not reject.

Finally $\mathbf{B}$ returns 1 if $\square(f'(\mathbf{x})) > \frac{2}{3}\epsilon$, and returns 0 otherwise.

$\mathbf{B}$ is a Defense. First, by the standard PAC theorem we have that with probability at least $1-\epsilon$, $\text{err}(f) \leq \frac{\epsilon}{2}$. This means that *correctness* holds with probability $l = 1-\epsilon$.

Note that with our setting of N, we have that

$$\epsilon^* \leq \frac{\epsilon}{2}.$$

Theorem 9 guarantees that

- if $\mathbf{x} \in \mathcal{D}^q$ then with probability at least $1-\epsilon$ we have that

  $$\square(f'(\mathbf{x})) \leq \frac{\epsilon}{2}.$$

  which in turn implies that with the same probability $\mathbf{B}$ returns $b = 0$. This implies that *completeness* holds with probability $1-\epsilon$.

- for every $\mathbf{x} \in (\{0,1\}^n)^q$ with probability at least $1-\epsilon$ we have that

  $$\text{err}(\mathbf{x}, f'(\mathbf{x})) \leq \frac{\epsilon}{2}.$$

  To compute soundness we want to upper bound the probability that $\text{err}(\mathbf{x}, f(\mathbf{x})) > 2\epsilon$[12] and $b = 0$. By construction of $\mathbf{B}$ if $b = 0$ then $\square(f'(\mathbf{x})) \leq \frac{2\epsilon}{3}$, which means that with probability at least $1-\epsilon$

  $$\text{err}(\mathbf{x}, \mathbf{y}) \leq \frac{2\epsilon}{3} + \frac{\epsilon}{2} < 2\epsilon \text{ or } b = 1.$$

  This translates to *soundness* holding with $s = \epsilon$.

REJECTRON can be implemented by a circuit of size polynomial in N and makes $O(\frac{1}{\epsilon})$ calls to an Empirical Risk Minimizer on $\mathcal{H}$ (that we assume can be implemented by a circuit of size polynomial in d), which implies the promised circuit complexity. □

J WATERMARKS EXIST

Lemma 4 (*Watermark for bounded VC-dimension against fast adversaries*)**.** *There exists a family of hypothesis classes $\{\mathcal{H}_d\}_d$ such that for every d, $\mathcal{H}_d$ has VC-dimension d and a family of distributions $\{\mathcal{D}_d\}_d$ such that for every $\epsilon \in \left(\frac{10000}{d}, \frac{1}{8}\right)$ there exists a family of circuits $\mathbf{A} = \{\mathbf{A}_d\}_d$ and a family of function classes $\mathcal{F}$ for which the following conditions hold. For every learning $\mathbb{L} = \{\mathbb{L}_d\}_d$ that for every d samples $\mathcal{D}_d$ always and $h_d \in \mathcal{H}_d$ we have that*

$$\mathbf{A} \in_1 \text{WATERMARK}\left(\mathbb{L}, \mathcal{F}, \epsilon, q = O\left(\frac{1}{\epsilon}\right), S_{\mathbf{A}} = O\left(\frac{d}{\epsilon}\right), S_{\mathbf{B}} = \frac{d}{100}, l = 1 - \frac{1}{100}, c = 1 - \frac{2}{100}, s = \frac{56}{100}\right).$$

Note that the setting of parameters is such that $\mathbf{A}$ can learn (with high probability) a classifier of error ϵ, but $\mathbf{B}$ is *not* able to learn a low-error classifier within its allotted circuit size $S_{\mathbf{B}}$. This contrasts with Lemma 3, where $\mathbf{B}$ has a sufficiently large circuit size to learn. This is the regime of interest for Watermarks, where the scheme is expected to be secure against $\mathbf{B}$ with limited circuit complexity.

Proof. Let $\mathcal{D}$ be the uniform distribution over $[N]$ for $N = 100d^2$, where recall that $[N] = \{1, \ldots, N\}$. Let $\mathcal{H}$ be the concept class of functions that have exactly d $+1$'s in $[N]$. Note that $\mathcal{H}$ has VC-dimension d. Let $h \in \mathcal{H}$ be the ground truth.

[12] Note that we measure the error of f not f'.

Construction of A. $\mathbf{A}$ works as follows. It draws $n = O\left(\frac{d}{\epsilon}\right)$ samples from $\mathcal{D}$ labeled with h. Let's call them $\mathbf{x}_{\text{TRAIN}}$. Let

$$A := \{x \in [N] : \mathbf{x}_{\text{TRAIN}}, h(x) = +1\}, B := \{x \in [N] : x \in \mathbf{x}_{\text{TRAIN}}, h(x) = -1\}.$$

$\mathbf{A}$ takes a uniformly random subset $A_w \subseteq A$ of size q. It defines sets

$$A' := A \setminus A_w,\ B' := B \cup A_w.$$

$\mathbf{A}$ computes f consistent with the training set $\{(x, +1) : x \in A'\} \cup \{(x, -1) : x \in B'\}$. $\mathbf{A}$ samples $S \sim \mathcal{D}^q$. It defines the watermark to be $\mathbf{x} := A_w$ with probability $\frac{1}{2}$ and $\mathbf{x} := S$ with probability $\frac{1}{2}$.

$\mathbf{A}$ sends $(f, \mathbf{x})$ to $\mathbf{B}$. $\mathbf{A}$ can be implemented with circuit complexity $O\left(\frac{d}{\epsilon}\right)$.

A is a Watermark. We claim that $(f, \mathbf{x})$ constitutes a watermark.

It is possible to construct a watermark of prescribed size, i.e., find a subset A_w of a given size, only if $|A| \geq q$. The probability that a single sample from $\mathcal{D}$ is labeled $+1$ is $\frac{d}{N}$, so by the Chernoff bound (Fact 1) $|A|, |B| > \frac{dn}{2N} \geq q$ with probability $1 - \frac{1}{100}$, where we used that $n = O\left(\frac{d}{\epsilon}\right), N = 100d^2, q = O(\frac{1}{\epsilon})$.

Correctness. Let $h'(x) := h(x)$ if $x \in [N] \setminus A_w$ and $h'(x) := -h(x)$ otherwise. Note that h' has exactly $d - q$ $+1$'s in $[N]$. By construction, f is a classifier consistent with h'. By the PAC theorem we know that with probability $1 - \frac{1}{100}$, f has an error at most ϵ wrt to h' (because the hypothesis class of functions with *at most* d $+1$'s has a VC dimension of $O(d)$). h' differs from h on q points, so

$$\text{err}(f) \leq \epsilon + q/N = O\left(\epsilon + \frac{1}{\epsilon d^2}\right) = O(\epsilon). \tag{14}$$

with probability $1 - \frac{1}{100}$, which implies that *correctness* is satisfied with $l = 1 - \frac{1}{100}$.

Distinguishing of $\mathbf{x}$ and $\mathcal{D}^q$. Note that the distribution of A_w is the same as the distribution of a uniformly random subset of $[N]$ of size q (when taking into account the randomness of the choice of $h \sim U(\mathcal{H})$). Observe that the probability that drawing q i.i.d. samples from $U([N])$ we encounter repetitions is at most

$$\frac{1}{N} + \frac{2}{N} + \cdots + \frac{q}{N} \leq \frac{3q^2}{N} \leq \frac{1}{100},$$

because $q < \frac{d}{100} < \frac{\sqrt{N}}{10}$. This means that $\frac{1}{100}$ is an information-theoretic upper bound on the distinguishing advantage between $\mathbf{x} = A_w$ and $\mathcal{D}^q$.

Moreover, $\mathbf{B}$ has access to at most t samples and the probability that the set of samples $\mathbf{B}$ draws from $\mathcal{D}^t$ and A_w have empty intersection is at least $1 - \frac{1}{100}$. It is because it is at least $(1 - \frac{t}{N})^t \geq (1 - \frac{1}{\sqrt{N}})^{\sqrt{N}/10} \geq 1 - \frac{1}{100}$, where we used that $t < \frac{\sqrt{N}}{10}$.[13]

Note that by construction f maps all elements of A_w to -1. The probability over the choice of $F \sim \mathcal{D}^q$ that $F \subseteq h^{-1}(\{-1\})$, i.e., all elements of F have true label -1, is at least

$$\left(1 - \frac{d}{N}\right)^q \geq 1 - \frac{1}{100}.$$

The three above observations and the union bound imply that the distinguishing advantage for distinguishing $\mathbf{x}$ from $\mathcal{D}^q$ of $\mathbf{B}$ is at most $\frac{4}{100}$ and so the *undetectability* holds with $s = \frac{8}{100}$.

Unremovability. Assume, towards contradiction with *unremovability*, that $\mathbf{B}$ can find $\mathbf{y}$ that with probability $s' = \frac{1}{2} + \frac{6}{100}$ satisfies $\text{err}(\mathbf{x}, \mathbf{y}) \leq 2\epsilon$. Notice, that $\text{err}(A_w, f(A_w)) = 1$ by construction.

Consider an algorithm $\mathcal{A}$ for distinguishing A_w from $\mathcal{D}^q$. Upon receiving $(f, \mathbf{x})$ it first runs $\mathbf{y} = \mathbf{B}(f, \mathbf{x})$ and returns 1 iff $d(\mathbf{y}, f(\mathbf{x})) \geq \frac{q}{2}$. We know that the distinguishing advantage is at most $\frac{1}{2} + \frac{4}{100}$, so

$$\frac{1}{2}\mathbb{P}_{\mathbf{x}:=A_w}[\mathcal{A}(f, \mathbf{x}) = 1] + \frac{1}{2}\mathbb{P}_{\mathbf{x}\sim\mathcal{D}^q}[\mathcal{A}(f, \mathbf{x}) = 0] \leq \frac{1}{2} + \frac{4}{100}.$$

[13] If the sets were not disjoint then $\mathbf{B}$ could see it as suspicious because f makes mistakes on all of A_w.

But also note that

$$\begin{aligned}
s' &\leq \mathbb{P}_{\mathbf{x}\sim\mathbf{A}}[\text{err}(\mathbf{x},\mathbf{y}) \leq 2\epsilon] \\
&\leq \frac{1}{2}\mathbb{P}_{\mathbf{x}:=A_w}[d(\mathbf{y}, f(\mathbf{x})) \geq (1-2\epsilon)q] + \frac{1}{2}\mathbb{P}_{\mathbf{x}\sim\mathcal{D}^q}[d(\mathbf{y}, f(\mathbf{x})) \leq (2\epsilon + \text{err}(f))q] \\
&\leq \frac{1}{2}\mathbb{P}_{\mathbf{x}:=A_w}[d(\mathbf{y}, f(\mathbf{x})) \geq q/2] + \frac{1}{2}\mathbb{P}_{\mathbf{x}\sim\mathcal{D}^q}[d(\mathbf{y}, f(\mathbf{x})) \leq q/2] + \frac{1}{100} \\
&\leq \frac{1}{2}\mathbb{P}_{\mathbf{x}:=A_w}[\mathcal{A}(f,\mathbf{x}) = 1] + \frac{1}{2}\mathbb{P}_{\mathbf{x}\sim\mathcal{D}^q}[\mathcal{A}(f,\mathbf{x}) = 0] + \frac{1}{100}.
\end{aligned}$$

Combining the two above equations we get a contradiction and thus the *unremovability* holds with $s' = \frac{1}{2} + \frac{6}{100}$.

Uniqueness. The following $\mathbf{B}$ certifies *uniqueness*. It draws $O\left(\frac{d}{\epsilon}\right)$ samples from $\mathcal{D}$, let's call them $\mathbf{x}'_{\text{TRAIN}}$ and trains f' consistent with it. By the PAC theorem $\text{err}(f') \leq \epsilon$ with probability at least $1 - \frac{1}{100}$. Next upon receiving $\mathbf{x} \in \{0,1\}^{nq} = [N]^q$ it returns $y = f'(\mathbf{x})$. By the fact that $\mathbf{x}$ is a random subset of $[N]$ of size q by the Chernoff bound, the union bound we know that $\text{err}(\mathbf{x},\mathbf{y}) = \text{err}(\mathbf{x}, f'(\mathbf{x})) \leq 2\epsilon$ with probability at least $1 - \frac{2}{100}$ over the choice of h. This proves *uniqueness*. □