DOES MORE COMPUTE MAKE EVERY ADVERSARIAL ATTACK TRANSFERABLE?

Anonymous authors
Paper under double-blind review

ABSTRACT

Transferable attacks, or transferable jailbreaks, are a particularly strong version of adversarial examples that fool all learners in a class. In this work, we study when more compute enables transferable attacks. We show that, when there is a quadratic gap in computational resources between the parties, then every learnable *classification* task admits either a transferable attack or an adversarial defense. On the other hand, we prove that there exist *generative* learning tasks for which neither a transferable attack nor an adversarial defense exists at a similar resource gap. These constructions provide formal evidence that increasing inference-time compute can improve robustness, as empirically observed by Zaremba et al. (2025). Our constructions assume the existence of an array of cryptographic primitives: identity-based fully homomorphic encryption (IB-FHE), publicly verifiable zero-knowledge succinct non-interactive arguments of knowledge (zk-SNARKs), non-parallelizing languages (NPL), and incrementally verifiable computation (IVC).

1 INTRODUCTION

AI agents carry out user requests by processing documents, browsing webpages, and interacting with external tools. This access to information exposes them to adversarial content: instructions embedded in a document or tool response can redirect an agent away from its intended task. Such indirect prompt injection attacks have been demonstrated in LLM-integrated applications and studied systematically in tool-using AI agents (Greshake et al., 2023; Debenedetti et al., 2024). These attacks bring renewed urgency to a central question in adversarial robustness: can an AI model detect all adversarial inputs on which it would make a mistake? Detection has been studied for bounded perturbations (Tramer, 2022), for arbitrary test examples with abstention (Goldwasser et al., 2020), and for prompt filters of language models (Ball et al., 2025).

Transferability is one of the most surprising features of adversarial attacks. Different models could have entirely different weaknesses, yet an input constructed to fool one model can also fool others. This phenomenon was observed in image classification and later in jailbreaks against aligned language models (Szegedy et al., 2014; Papernot et al., 2016; Liu et al., 2016; Zou et al., 2023). Bose et al. (2020) cast the search for attacks that transfer to an entire hypothesis class as a min-max game. The observed transfer between particular models invites a stronger question:

Question 1. When does the existence of an attack against each individual model imply the existence of a single attack that transfers to all models from a class?

We ask how the answer depends on the computational resources available to the attacker and defender. A Transferable Attack (TA) is a single attack procedure that succeeds against every defender within a specified resource bound. An Adversarial Defense (AD) publishes a model that remains useful on natural inputs and withstands every attacker within the corresponding resource bound. By a published model we mean the trained model made available to the attacker.[1] We build on the framework of Gluch et al. (2025), who established a trichotomy between TAs, ADs, and Watermarks (WMs) depending on computational power of the parties. Bubeck et al. (2019); Degwekar et al. (2019); Mahloujifar & Mahmoody (2019); Garg et al. (2020) also considered computational constraints in the context of adversarial robustness.

[1] We will assume white-box access to the model.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

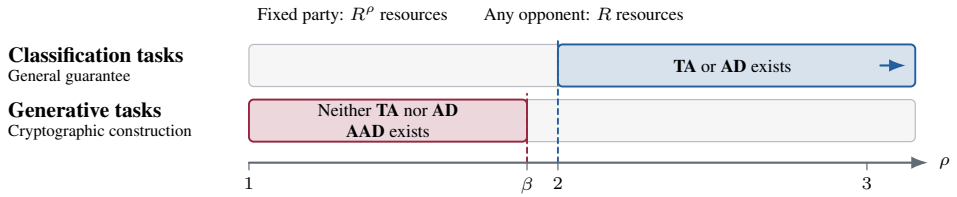


Figure 1: **Scaling for adversarial robustness.** Generative tasks: under cryptographic assumptions, for $\rho < \beta$ the task admits neither a TA nor an AD, while an AAD exists (Theorem 46, Corollary 50). Classification tasks: for $\rho \geq 2$ every learnable task admits a TA or an AD (Lemma 28).

For classification, we prove that every learning task admits a TA or an AD, with a roughly quadratic resource tradeoff. This shows that the WM alternative is not needed at this gap. In this sense, the dichotomy resolves *Question 1* for classification tasks at this gap: if every model can be attacked, then a single attack with quadratically more resources than its target transfers to all of them.

How far does this classification picture extend? AI agents typically generate responses or take actions in situations where several different outcomes may satisfy the user's request. This motivates studying generative tasks, where an input can have several valid answers.

For generative tasks, a defender rarely answers with a fixed model alone: it can reason, search, or call tools after it sees the input, spending additional computation on exactly the query it received. Randomized smoothing, adversarial purification and test-time training are defenses of this kind (Cohen et al., 2019; Nie et al., 2022; Sun et al., 2020), and transductive learning gives the learner the test inputs before it answers (Goldwasser et al., 2020; Montasser et al., 2022). Experiments with reasoning models show that additional inference-time computation can improve robustness (Zaremba et al., 2025); see Wu et al. (2025) and McDonald et al. (2026).

Question 2. Can inference-time compute give robustness when every individual model is vulnerable?

We formalize this possibility as an Active Adversarial Defense (AAD): the attacker sees the published model, but the system can use additional computation to produce its final response after the attack has been chosen. Under cryptographic assumptions, we construct a generative task on which an AAD withstands stronger attackers than any AD with the same training budget. As a corollary, at the stated resource bounds, the task admits neither a TA nor an AD.

2 Contributions

A TA produces an input $\mathbf{x}$ that fools *all* computationally bounded defenders, none of which can distinguish it from a natural input. An AD publishes a low-error model f and a distinguishing rule R, such that *every* computationally bounded adversary either fails to fool f or is distinguished by R; see Figure 2. Notice that the two notions are not negations of one another. The absence of a TA says only that each adversary is withstood by *some* defender, chosen once that adversary is fixed and possibly a different one for each, whereas an AD asks for *one* defender that withstands all attacks.

A Dichotomy for Classification Tasks. We establish a dichotomy between these two notions answering *Question 1* in the classification setting.

Lemma 28 (informal). For every classification task that is learnable by circuits of size $o(\sqrt{S})$, either there is a TA of size $O(S)$ against defenders of size $o(\sqrt{S})$, or there is an AD of size $O(S)$ against attackers of size $o(\sqrt{S})$.

The proof follows a game-theoretic approach. We consider a finite zero-sum game v. Neumann (1928) with the interaction between attackers and defenders and use sparse approximate optimal strategies Lipton & Young (1994); Althöfer (1994) to obtain computationally bounded implementations.

Gluch et al. (2025) showed that every classification task admits at least one of three schemes: a TA, an AD, or a WM. The third one is a claim of ownership: it lets whoever trained the model prove that the model is theirs, so that nobody who merely copies it can pass it off as their own.

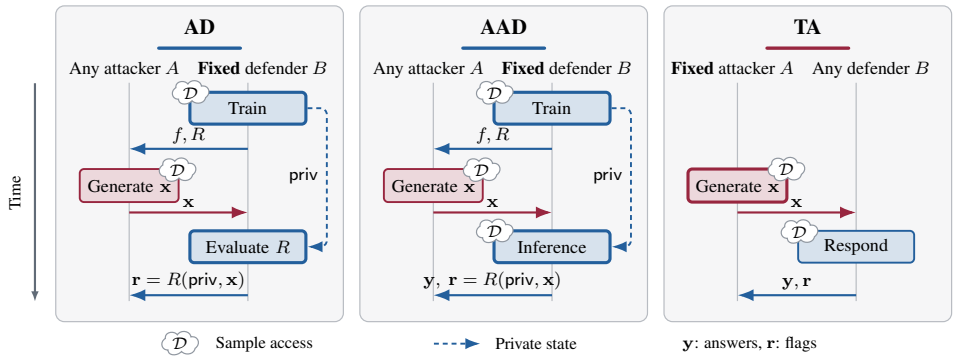


Figure 2: **The three notions.** Protocols between the attacker A and the defender B.

Notice that Gluch et al. (2025) have a slightly different notion of an AD: it must rarely flag natural inputs and often flag attacks. Ours asks that the two flag rates differ. We observed that there is a way to turn their WM into a TA against the same defenders, so their trichotomy also gives "attack or defense" at a quadratic gap. At matched parameters our dichotomy improves every comparable quantity and reaches settings theirs cannot, see Section 5.

A Construction of Transferable Attacks with Improved Assumptions. We give an explicit construction of a classification task admitting a TA. Our construction requires a secure public-key encryption (PKE) (Goldwasser & Micali, 1984), weakening the cryptographic assumption used by Gluch et al. (2025), whose construction relied on fully homomorphic encryption (FHE). We then show how to lift *any* classification task to a task that, under the same cryptographic assumption, admits a TA; see Appendix E.

Active Adversarial Defenses Are Stronger for Generative Tasks. We introduce a notion of an Active Adversarial Defense (AAD), which, compared to AD, allows the defender to invest additional *inference-time computation* to produce an answer $\mathbf{y}$, that might be different from $f(\mathbf{x})$, *after* seeing questions $\mathbf{x}$; see Figure 2. Then, in the generative setting, we answer *Question 2* by proving the following theorem.

Theorem 46 (informal). There exists a *generative* learning task that, under cryptographic assumptions, has an AAD with training-time $O(S)$ and inference-time $\tilde{O}(\sqrt{S})$, against attackers of size $O(S)$. At the same time, it has no AD with training-time $O(S)$, even against attackers of size $\tilde{O}(\sqrt{S} \cdot S_E)$. [2,3]

A similar statement holds when the resource of interest is sample complexity, see Theorem 44.

Our constructions make use of several cryptographic primitives: IB-FHE (Gentry et al., 2013; Clear & McGoldrick, 2014), zero-knowledge succinct non-interactive arguments of knowledge (zk-SNARKs) (Bitansky et al., 2022), digital signatures (Boneh et al., 2006), NPLs (Bitansky et al., 2016), and incrementally verifiable computation (IVC) (Valiant, 2008; Naor et al., 2019).[4] NPLs together with IVC play the role of a continuous verifiable delay function (Ephraim et al., 2020).

Neither a Transferable Attack nor an Adversarial Defense for Generative Tasks. It turns out that the answer to *Question 1* in the generative setting is different than that given by Lemma 28.

Corollary 50 (informal). There exists a generative task for which there is neither a TA of size $O(S)$ against defenders of size $\tilde{O}(\sqrt{S})$ nor an AD of size $O(S)$, even against attackers of size $\tilde{O}(\sqrt{S} \cdot S_E)$.

Lemma 28 and Corollary 50 are represented in Figure 1, which depicts a classification of learning tasks by whether TA or AD are present depending on the resources available to the parties. Our results suggest that a *relative resource exponent* $\rho := \log R_1 / \log R_2$, where R_1 is the fixed party's budget and R_2 is its opponents' budget is a key parameter for this classification.

Due to space constraints, we relegate related works to Appendix A.

[2] Here $S_E \ll \sqrt{S}$ is the evaluation size of the published model.

[3] The formal exponent is $1/(2\eta)$, where $\eta < 1$ is the hardness gap of the NPL, taken close to 1.

[4] Our focus is on demonstrating this limitation, rather than minimizing the cryptographic assumptions.

3 Learning and Computation Modeling

Learning Tasks. An error oracle is a function $h : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, that takes a pair (x, y), and outputs 0 if y is a valid answer for x, and 1 otherwise. A model is a (potentially randomized) function $f : \mathcal{X} \to \mathcal{Y}$. A distinguishing rule is a (potentially randomized) function $R : \mathcal{X} \to \{0,1\}$, and we call its output $r = R(x)$ a flag. For a distribution $\mathcal{D} \in \Delta(\mathcal{X} \times \mathcal{Y})$ we denote its marginal distribution over $\mathcal{X}$ by $\mathcal{D}_{\mathcal{X}}$. We define the error of a model f as $\mathrm{err}(f) := \Pr_{x \leftarrow \mathcal{D}_{\mathcal{X}}}\big[h(x, f(x)) = 1\big]$. For questions $\mathbf{x} \in \mathcal{X}^q$, answers $\mathbf{y} \in \mathcal{Y}^q$ and flags $\mathbf{r} \in \{0,1\}^q$ we also define the empirical error as $\mathrm{err}_{\mathbf{x},h}(\mathbf{y}) := \frac{1}{q}\sum_{i\in[q]} h(x_i, y_i)$ and the flag rate as $\mathrm{fl}(\mathbf{r}) := \frac{1}{q}|\{i \in [q] : r_i = 1\}|$.

Definition 1 (Learning Task, informal version of Definition 11)**.** 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 $\Delta\big(\Delta(\mathcal{X}_n \times \mathcal{Y}_n) \times \{0,1\}^{\mathcal{X}_n \times \mathcal{Y}_n}\big)$.[5]

Computation and Interaction. We model efficient computation by non-uniform families of Boolean circuits $C = \{C_n\}_n$ of a certain size. Our circuits can read uniformly random bits, and they query the example oracle $\mathrm{Ex}(\mathcal{D}, h)$ that samples $(x, y) \leftarrow \mathcal{D}$ and returns (x, y).

Before the interaction between a defender and an attacker starts, $(\mathcal{D}, h)$ is sampled. The interaction is modeled by two- and three-round interactive protocols (Goldwasser et al., 1989) between Alice and Bob. Alice is the attacker $A = \{A_n\}_n$ and Bob is the defender $B = \{B_n\}_n$ of sizes S_A and S_B.

Learnability. Recall the definition of PAC learnability (Valiant, 1984; Kearns & Vazirani, 1994). Taking the computational aspect of learning algorithms into account, Gluch et al. (2025) defined computationally bounded learnability.

Definition 2 (Computationally Bounded Learnability, informal version of Definition 12)**.** We say that a learning task is learnable by circuits of size S, if there exists a circuit family $\mathcal{L}$ of size S, which outputs f,[6] such that with high probability, for every n large enough $\mathrm{err}_{\mathcal{D}}(f)$ is small.

4 Transferable Attacks and Adversarial Defenses

Recall that a TA is an attack that works against all size bounded defenders.

Definition 3 (Transferable Attack, informal version of Definition 37)**.** A family of circuits A_{TA} with complexity S_A implements a TA against size S_B defenders for a learning task $\mathbb{L}$, if an interactive protocol in which A_{TA} computes $\mathbf{x}$ and an arbitrary B based on $\mathbf{x}$ outputs $(\mathbf{y}, \mathbf{r})$ satisfies both of the following: For circuits B of size S_B

- **Transferability**. B answers the batch that A sends mostly incorrectly: the expected error $\mathrm{err}_{\mathbf{x},h}(\mathbf{y})$ is large.
- **Indistinguishability**. B flags the batch that A sends about as often as a natural batch $\mathbf{x} \leftarrow (\mathcal{D}_{\mathcal{X}})^q$: the expected flag rates $\mathrm{fl}(\mathbf{r})$ on the two batches are almost the same.

Notice that Transferability alone is typically easy: for a task whose ground truth is nontrivial outside the support of the natural distribution, Alice can send a string there that no small circuit answers correctly. Indistinguishability alone is also easy, since Alice can forward a natural batch.

An AD reverses the roles and works against all size bounded attacks. Remember that the absence of a TA would give each attacker *some* defender that withstands it. An AD is stronger, being one defender that withstands them all.

Definition 4 (Adversarial Defense, informal version of Definition 34)**.** A family of circuits B_{AD} with complexity S_B implements an AD against size S_A adversaries for a learning task $\mathbb{L}$, if an interactive protocol in which B_{AD} computes a model f and a distinguishing rule R, sends them to A_n, which replies with $\mathbf{x}$, satisfies:

- **Correctness**. With high probability, the model has low error: $\mathrm{err}_{\mathcal{D}}(f)$ is small.

Moreover, for every circuit A of size S_A at least one of the following holds.

[5] We note that Definition 11, which is similar to the one used in Gluch et al. (2025), is nonstandard. See Appendix for more details.

[6] Formally, we use strings z_f representing f and consider circuits that on input z_f and x output $f(x)$.

- **Robustness**. The model answers the questions of A mostly correctly: the expected error $\mathrm{err}_{\mathbf{x},h}(f(\mathbf{x}))$ is small.
- **Distinguishability**. The rule flags the questions of A at a different rate than a natural questions: the expected flag rates $\mathrm{fl}(R(\mathsf{priv},\mathbf{x}))$ on the two batches differ noticeably.

An AAD has an additional inference-time phase, in which Bob uses additional resources to answer questions instead of just using f.

Definition 5 (Active Adversarial Defense, informal version of Definition 36)**.** A pair of circuit families $B_{\mathsf{AAD}} = (B_{\mathsf{Train}}, B_{\mathsf{Inference}})$ with sizes S_{Train} and $S_{\mathsf{Inference}}$ implements an AAD against size S_A adversaries for a learning task $\mathbb{L}$, if an interactive protocol in which B_{Train} computes a model f, a distinguishing rule R and a private string priv that instantiates R, sends f and R to A, which replies with $\mathbf{x}$, and $B_{\mathsf{Inference}}$ on input priv and $\mathbf{x}$ outputs answers $\mathbf{y}$ and the flags $\mathbf{r} = R(\mathsf{priv},\mathbf{x})$, satisfies:

- **Correctness**. With high probability, the model has low error: $\mathrm{err}_{\mathcal{D}}(f)$ is small.
- **Completeness**. With high probability, the inference phase answers a natural batch $\mathbf{x} \leftarrow (\mathcal{D}_{\mathcal{X}})^q$ mostly correctly: $\mathrm{err}_{\mathbf{x},h}(\mathbf{y})$ is small.

Moreover, for every circuit A of size S_A at least one of the following holds.

- **Robustness**. The inference phase answers the batch of A mostly correctly: the expected error $\mathrm{err}_{\mathbf{x},h}(\mathbf{y})$ is small.
- **Distinguishability**. The rule flags the batch of A at a different rate than a natural batch: the expected flag rates $\mathrm{fl}(R(\mathsf{priv},\mathbf{x}))$ on the two batches differ noticeably.

Although AAD's inference procedure can be described as a model, publishing this description yields an admissible AD only if the resulting model can be evaluated within the evaluation budget.

5 Every Classification Task Has a Transferable Attack or an Adversarial Defense

We prove that a learnable classification task either has an attack that fools every sufficiently small defender, or there is a defense that holds up against every sufficiently small attacker. This is a win-win statement similar in spirit to Degwekar et al. (2019).

Lemma 6 (Informal version of Lemma 28)**.** *For every size function S and every classification task that is learnable by circuits of size $\sqrt{S}/(4\log S)$, at least one of the following exists:*

(i) a TA of size $O(S)$ against defenders of size $o(\sqrt{S})$, or
(ii) an AD of size $O(S)$ against attackers of size $o(\sqrt{S})$.

As mentioned above, the main theorem in Gluch et al. (2025) also implies the dichotomy with a quadratic gap. However, their result considers constant robustness and transferability parameters. A close analysis shows that it is possible to attain one degree of freedom in their parameters. Our result has four, the constants c, s, α and q of Lemma 28, which move the two alternatives independently. Running their argument with a free threshold $1-\rho$ yields an attack of transferability ρ and flag-rate gap $1-2\rho$, or a defense whose false-positive rate on natural inputs and miss rate on attacks add up to about 4ρ. The gap of the attack and the false-positive rate of the defense always sum to one up to the confidence parameter, so whichever threshold is chosen, the distinguishing side of one of the two branches is vacuous. Table 1 runs both statements at their constants, where every comparable number improves, and then at matched transferability, where Corollary 29 reaches a regime their family cannot achieve.

Proof (Sketch). We view the interaction as a zero-sum game. Rows are indexed by size $o(\sqrt{S})$ attackers A, each producing an input x, and columns by size $o(\sqrt{S})$ defenders B, each answering with a label y and a flag r. The entry $M_{A,B}$ rewards the defender, being the sum of the probability that B labels $x \leftarrow A$ correctly and of the distinguishability advantage of B against A. The attacker plays Min and wants a small sum, while the defender plays Max and wants it large. By the Min-Max Theorem v. Neumann (1928) the game achieves the Nash equilibrium, and by a result in Lipton & Young (1994) each player has a nearly optimal strategy that mixes only a few circuits and can therefore be implemented by a circuit of size $O(S)$. Which player wins determines which notion we obtain.

	Attack (transferability, gap)		Defense (robustness, advantage)	
	Gluch et al.	Lemma 28	Gluch et al.	Lemma 28
Their constants, $\rho = 5/24$	$(0.21,\ 0.58)$	$(0.21,\ 0.02)$	$(0.92,\ –)$	$(0.54,\ 0.001)$
Matched transferability, $\rho = 0.10$	$(0.10,\ 0.80)$	$(0.10,\ 0.01)$	$(0.43,\ –)$	$(0.25,\ 5 \cdot 10^{-5})$

Table 1: **The two alternatives at matched parameters.** The first row runs both statements at the constants of Gluch et al. (2025) and $\varepsilon = 0.01$ in Lemma 28. The second matches their transferability to ours and reads Corollary 29 at $\varepsilon = \delta = 0.01$.

If the value is small, the attacker wins. The nearly optimal mixture A_N keeps both terms small against every small defender, which is exactly a TA.

If the value is large, the defender wins, and we turn the nearly optimal mixture B_N into a defense B_{AD}. Bob outputs the classifier f produced by the learner. Rule R tosses a fair coin, which he keeps private, to decide between two checks: flag x if B_N flags it, or flag x if the label of B_N disagrees with $f(x)$. A probabilistic analysis shows that B_{AD} satisfies Distinguishability or Robustness. □

A careful analysis of the proof shows that uniqueness of labels is crucial for it to work. The defense flags a disagreement between B_N and f, which signals an error only because two correct labels must agree. In a generative task they may give different valid answers, and the defender cannot check validity itself since it only sees samples. It is therefore not obvious how to extend the dichotomy to generative tasks, and Corollary 50 rules it out only below a quadratic advantage.

Open Question 1. Does every learnable generative task admit a TA or an AD at a quadratic, or at least some polynomial, resource advantage?

5.1 Constructions of Transferable Attacks

Transferable Attack for a Public-Key Encryption Learning Task. The construction of a TA in Gluch et al. (2025) relies on FHE. Here, ordinary secure PKE suffices. The existence of PKE is believed to be a strictly weaker assumption than the existence of FHE: there is no black-box construction of FHE from PKE, since FHE implies collision-resistant hashing (Ishai et al., 2005) while no black-box construction of collision-resistant hashing from identity-based encryption exists (Mahmoody & Mohammed, 2016).

We construct a TA for a learning task $\mathbb{L}_{\mathsf{PKE}}$ that is based on a PKE scheme $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ over the message space $\mathcal{M} = \{0,1\}$. PKE is secure Goldwasser & Micali (1984), if any efficient adversary $\mathcal{B}$ knowing only the public key pk cannot decrypt $c = \mathsf{Enc}(\mathsf{pk}, y)$: its decryption fails almost half of the time.

If we think of $(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, y))$ as an adversarially chosen input x and of y as its true label, then security corresponds to the transferability of an attack producing such x. Remember that we would like to construct a TA for a task that can be learned. An easy solution is to add the plain message y to the description of x, that is, to set $x := (\mathsf{pk}, y, \mathsf{Enc}(\mathsf{pk}, y))$ and keep y as the true label. Now the task is learnable: the bit next to the public key is the true label of x. An adversary instead of appending y, outputs $x = (\mathsf{pk}, y', \mathsf{Enc}(\mathsf{pk}, y))$, where y and y' are independent uniformly random bits. The modified attack is transferable, since the additional information y' does not help a defender guess the label y of x. Moreover, an adversarially generated x looks like it came from the natural distribution by the security of PKE. See Appendix E for a formal description of the task and the attack.

Transferable Attack for a Lifting of Any Learning Task. Now let $\mathbb{L}$ be *any* learning task. Building upon the above construction, we provide a way to append additional dimensions to $x \in \{0,1\}^{\lambda}$ without changing the labels of natural inputs, such that the resulting learning task $\mathbb{L}'$ admits a TA.

We continue working with a secure PKE as before. Now let natural inputs be of the form $x' := (\mathsf{pk}, x, \mathsf{Enc}(\mathsf{pk}, 0))$, where $x \leftarrow \mathcal{D}_\lambda$. An arbitrary $x' = (\mathsf{pk}', x, c) \in \{0,1\}^{\mathsf{poly}(\lambda)}$ receives the label $h_\lambda(x) \oplus \mathsf{Dec}(\mathsf{sk}, c)$ if $\mathsf{pk}' = \mathsf{pk}$, and zero otherwise. The TA samples one natural input (pk, x, c) and outputs $x' := (\mathsf{pk}, x, \mathsf{Enc}(\mathsf{pk}, b))$, where b is a uniformly random bit. Transferability and indistinguishability again follow from the security of the PKE.

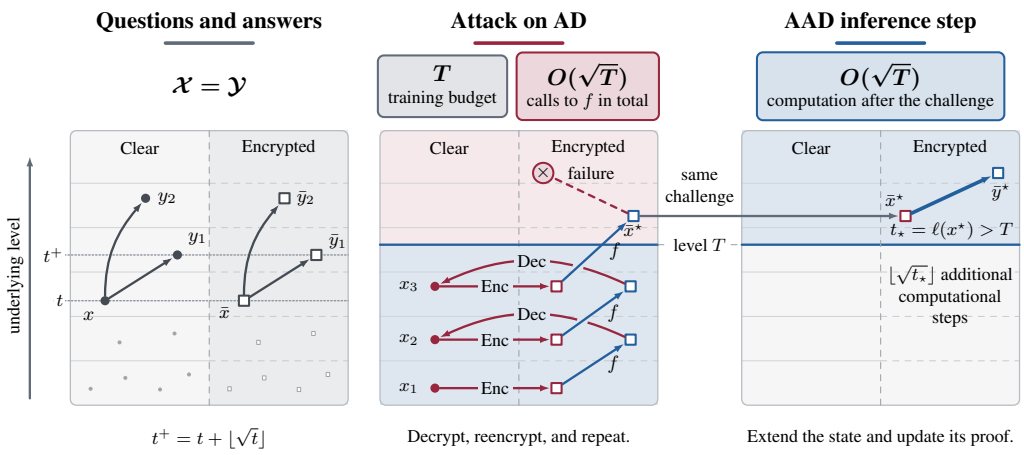


Figure 3: **Schematic of the construction of Theorem 7.** Questions and answers share the same space. Horizontal lines separate difficulty levels. Valid answers for questions on level t are at level at least $t + \lfloor\sqrt{t}\rfloor$. The attacker evaluates f on encrypted queries, decrypts, checks if the outputs are correct, and reencrypts them as new questions. This iteration eventually yields an f's mistake. AAD answers by extending the state in $x^\star$ through homomorphic computation for $\lfloor\sqrt{t_*}\rfloor$ steps.

In Appendix E, we also provide a construction of a learning task that admits an AD.

6 Neither a Transferable Attack nor an Adversarial Defense for Generative Tasks

We now turn to generative tasks. We construct a task that separates ADs from AADs: every AD within the stated training budget can be attacked, while an AAD using additional inference-time computation can respond robustly. The same task then falls outside the dichotomy, admitting neither TA nor AD.

Theorem 7 (Informal version of Theorem 46)**.** *There exists a generative task such that, under cryptographic assumptions, the following holds for batch size $q = 1$, every S, and every evaluation size $S_E \ll \sqrt{S}$ of the published model.*[7]

(i) There is an AAD with $S_{\mathsf{Train}} = O(S)$ and $S_{\mathsf{Inference}} = \tilde{O}(\sqrt{S})$, against attackers of size $O(S)$.
(ii) There is no AD with $S_B = O(S)$, even against attackers of size $\tilde{O}(\sqrt{S} \cdot S_E)$.

We also prove a similar type of separation where the resource of interest is samples; see Theorem 44.

Corollary 8 (Informal version of Corollary 50)**.** *For the task of Theorem 7, batch size $q = 1$, every S, and every evaluation size $S_E \ll \sqrt{S}$ of the published model:*

(i) There is no TA of size $O(S)$ against defenders of size $\tilde{O}(\sqrt{S})$.
(ii) There is no AD with $S_B = O(S)$, even against attackers of size $\tilde{O}(\sqrt{S} \cdot S_E)$.

Proof of Corollary 8. Item (ii) is item (ii) of Theorem 7. For item (i), take the AAD of Theorem 7. *Its inference phase does not use* priv *from the training phase*, so it is a defender on its own, of size $\tilde{O}(\sqrt{S})$. An attacker of size $O(S)$ in a TA sees no published model, so it is an attacker of the AAD that ignores f, and by item (i) of the theorem the inference phase answers its questions mostly correctly. Hence no attacker of size $O(S)$ fools every defender of size $\tilde{O}(\sqrt{S})$, and none is a TA. □

Proof sketch of Theorem 7. Recall that a learning task is a distribution over pairs $(\mathcal{D}, h)$, where $\mathcal{D}$ is a distribution over $\mathcal{X}_n \times \mathcal{Y}_n$ and $h : \mathcal{X}_n \times \mathcal{Y}_n \to \{0, 1\}$. We set $\mathcal{X}_n = \mathcal{Y}_n$. The task $\mathbb{L}^{\mathsf{time}}$ will have a level function $\ell : \mathcal{X}_n \to \mathbb{N}$ and will satisfy the following four properties.

[7] $\tilde{O}$ hides polynomial factors that are fixed once the cryptographic primitives are instantiated. The formal task is chosen once the budget S is. Remark 49 shows how one task serves every budget up to a polynomial gap.

(P1) Stratified hardness with correctness-induced amplification. For an input x of level $t = \ell(x)$, every valid output y is itself an input and has level $\ell(y) \geq t^+ := t + \lfloor \sqrt{t} \rfloor$. Thus, every correct answer advances an underlying sequential computation by $\lfloor \sqrt{t} \rfloor$ steps.

(P2) Sequential-resource certification. A valid level-t input can be produced in $O(t)$ sequential time, but no circuit of size t^η produces a valid input at the terminal level t of the underlying computation, where $\eta < 1$ is the hardness gap of the NPL. Thus, correctness at the terminal level certifies an expended computational effort (is a "proof of work"), but the certificate itself does not provide a shortcut for reaching substantially later configurations. Properties P1 and P2 together are the defining properties of a continuous verifiable delay function (Ephraim et al., 2020); see also verifiable delay functions (Boneh et al., 2018).

(P3) Hardness hiding. The marginal on inputs is an equal mixture of *clear* and *opaque* inputs. The level of a clear input is efficiently computable. For opaque inputs, the level is strongly computationally hidden; in particular, opaque encodings of inputs at different levels are computationally indistinguishable to a party that lacks the relevant secret key.

(P4) Selective verifiability (symmetry breaking). The attacker can generate a subset of opaque inputs for which it privately knows a decryption key. Consequently, it can apply the model to an opaque input, decrypt the answer, and check whether the answer is valid.

We explain why these properties yield the separation, and then give the cryptographic instantiation.

How to Attack Any Adversarial Defense. Let f be the model produced by a training algorithm of size S. Starting from a valid level-1 input x_1, the attacker can call f on itself multiple times, that is,

$$x_i = f(\dots(f(f(x_1)\dots).$$

By P2, $O(\sqrt{S})$ iterations suffice to reach level $\Theta(S)$. By P2, a model produced by training of size S cannot continue supplying valid configurations once this iteration passes the amount of sequential work available during training. Thus there is a first index $i = O(\sqrt{S})$ for which $f(x_i)$ is invalid.

The key challenge for the attacker is to compute an adversarial example x, which is *indistinguishable* by f from a random sample from $\mathcal{D}_{\mathcal{X}_n}$. If x is a clear input then it is obviously distinguishable by inspecting $\ell(x)$. Thus, to construct an indistinguishable x, the attacker uses P4 to choose an opaque input x' for which it knows its level as well as the levels of $f(x'), f(f(x')), \dots$. In contrast, by P3, the defender (model f) cannot compute the level of x, and so an AD does not exist.

How to Design an Active Adversarial Defense. Training uses a circuit of size S to compute the underlying sequential trajectory and stores checkpoints through level $S + O(\sqrt{S})$ by storing every $\sqrt{S}$-th configuration. The resulting model answers ordinary inputs up to level S; since the honest level has a geometrically decreasing tail, this is sufficient for correctness.

After the attacker chooses x, the inference procedure continues from the configuration in x for $O(\sqrt{S})$ steps. An attacker of size $O(S)$, even when combined with the information in the trained model, cannot produce a valid input of level more than $O(S)$ by P2. This is the source of the separation: an AD must commit to its model before seeing the adversarial input, whereas an AAD may spend $O(\sqrt{S})$ work extending precisely the configuration that the attacker supplies.

How to Instantiate $\mathbb{L}^{\mathsf{time}}$ Satisfying P1–P4. We use an average-case NPL (Definition 24), an IVC scheme (Valiant, 2008; Naor et al., 2019) (Definition 23), and *Identity-Based Fully Homomorphic Encryption* (IB-FHE) (Definition 22). Their roles are distinct. NPL supplies a computation with no substantially faster parallel shortcut. IVC attaches a short, publicly verifiable proof to each configuration and permits the proof to be updated one step at a time. IB-FHE hides the level while allowing both the learned model and the inference procedure to operate on encrypted inputs.

A first attempt, containing most of the ideas of the proof, at putting all the pieces together and defining the learning task $\mathbb{L}^{\mathsf{time}}$ concretely is as follows. We let the distribution $\mathcal{D}$ be composed of two equally probable parts $\mathcal{D} = \frac{1}{2}\mathcal{D}_{\mathsf{Clear}} + \frac{1}{2}\mathcal{D}_{\mathsf{Enc}}$. For every $(x, y) \in \mathsf{supp}(\mathcal{D}_{\mathsf{Clear}})$,

$$x = t \| c \| \pi,\ y = (t + \lfloor \sqrt{t} \rfloor) \| c' \| \pi'.$$

Let Samp be the sampler and L the decision algorithm of an average-case NPL, and let $Z \leftarrow \mathsf{Samp}(1^n, \Lambda)$ be a single instance at difficulty $\Lambda = \tilde{\Theta}(S^{1/\eta})$, just beyond the combined budget of training and attack. Pair (x, y) satisfies: i) c and c' are the configurations of $L(\Lambda, \cdot)$ on Z after t and

$t + \lfloor\sqrt{t}\rfloor$ steps, ii) π and π' are IVC proofs of these two statements, computed step by step with the IVC algorithm Update, which turns a proof for t steps into a proof for $t+1$ steps at the cost of one step. The level t is distributed as $\mathsf{Geom}(\frac{1}{2})$. We say that y is incorrect on x if x or y contain invalid proofs or the level of y is smaller than $t + \lfloor\sqrt{t}\rfloor$. $\mathcal{D}_{\mathsf{Enc}}$ is defined as $\mathcal{D}$ under the IB-FHE encryption.

How to Design an Active Adversarial Defense. To show that an AAD exists, we define B_{Train} that computes a classifier f that answers correctly for levels up to S. Next, upon receiving an input x, $B_{\mathsf{Inference}}$ *computes* a correct output "from scratch", that is, if x is unencrypted it parses $x = t \parallel c \parallel \pi$ and runs L for $\lfloor\sqrt{t}\rfloor$ steps with c as a starting point. Additionally, it computes an IVC proof that its computation was correct. If x is encrypted it runs the same procedure under the FHE.

The intuitive reason why $B_{\mathsf{Inference}}$ is a valid AAD is that the combined running time of A and B_{Train} is $O(S) + S = O(S)$ and so, by the sequential hardness of the NPL corresponding to L, they can compute the correct configurations of L only for $O(S)$ steps, which implies that highest level of x that A can create is $O(S)$. This implies that $O(\sqrt{S})$ computation of $B_{\mathsf{Inference}}$ is enough.

How to Attack Any Adversarial Defense. The idea for an attack is, again, similar to that for $\mathbb{L}^{\mathsf{data}}$. To attack, A upon receiving f computes a valid input x_1 on level 1 and iterates $x_i = f(x_{i-1})$ until f no longer produces a valid output for x_i. Then she sends as adversarial input $x = \mathsf{Enc}(x_i)$ with probability $\frac{1}{2}$ or a fresh sample from the marginal of $\mathcal{D}_{\mathsf{Clear}}$ otherwise. Notice that this input is indistinguishable from a sample from $\mathcal{D}_{\mathcal{X}_n}$ by the properties of FHE. Oberve that such A runs in time $O(\sqrt{S})$. It is because f can, by the properties of NPL, be correct only up to level $O(S)$.

Unfortunately, we cannot quite yet guarantee that f would make an error on x. It is possible that B could have provided f that is weaker on $\mathcal{D}_{\mathsf{Clear}}$ than it is on $\mathcal{D}_{\mathsf{Enc}}$! After all, this is "how" an AAD works, that is, $B_{\mathsf{Inference}}$ "hides" from A a stronger model than f, with which he answers. Maybe, B did "hide", e.g., by obfuscating, a stronger model on the $\mathcal{D}_{\mathsf{Enc}}$ part. To solve this issue we use a special type of FHE, called an identity-based FHE that will make $\mathbb{L}^{\mathsf{time}}$ satisfy P4. Modifying $\mathbb{L}^{\mathsf{time}}$ slightly will allow the parties partial decryption access and the evaluation of $h(x, f(x))$. With this final change we get a proof; see Appendix G for the full proof. □

The proof crucially uses that the task is generative. An AD could run the inference step inside its distinguishing rule R, but a disagreement between that answer and $f(x)$ does not show that f errs. For classification tasks a disagreement does show an error. An AD can publish f and flag an input whenever the AAD flags it or its answer disagrees with f, which turns an AAD into an AD of size about $S_{\mathsf{Train}} + S_{\mathsf{Inference}}$. Hence no separation as in Theorem 7 exists for classification tasks.

Similarly the proof of Corollary 8 doesn't work for classification tasks. It is because the AAD would turn into an AD of size $O(S)$ as above and contradict item (ii). Whether the dichotomy is tight for classification tasks therefore remains open.

Open Question 2. Is there a classification task that admits neither a TA nor an AD when the fixed party has less than a quadratic resource advantage?

7 Conclusions

We showed that every learnable classification task admits a TA or an AD once the stronger party has a quadratic resource advantage over its opponent. For generative tasks the picture changes. We constructed a task that admits neither, while there is a successful AAD that spends the square root of its training budget during inference. Inference-time computation can thus improve robustness that no fixed model can provide, which is in line with Zaremba et al. (2025). These results taken together strongly suggest that the relative resource exponent is a key parameter for determining if TA or AD exists for a given task.

AI USE STATEMENT

The authors reviewed all work prepared with the assistance of generative AI and take full responsibility for the content of this paper, including every passage and figure so prepared.

Research contributions. All research contributions are the authors' own. The problem formulation, the definitions, the statements of the theorems and corollaries, and every proof were conceived and written by the authors.

Writing. The authors used generative AI assistants to help draft and revise parts of the exposition, and for copy-editing of English usage and grammar. Every passage prepared with such assistance was read, checked against the underlying mathematics, edited where needed and approved by the authors.

Figures. The authors used generative AI assistance to produce the code that draws the figures. What each figure shows, and its correspondence to the results it illustrates, was specified and verified by the authors.

Not applicable. This work is theoretical. It involves no experiments, no datasets and no released software, so no generative AI was used for experimentation, data collection, data analysis or the preparation of research artifacts. Generative AI was not used for translation.

REFERENCES

Ingo Althöfer. On sparse approximations to randomized strategies and convex combinations. *Linear Algebra and its Applications*, 199:339–355, 1994.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in AI safety. *arXiv preprint arXiv:1606.06565*, 2016.

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

Sanjeev Arora and Boaz Barak. *Computational complexity: a modern approach*. Cambridge University Press, 2009.

Sarah Ball, Greg Gluch, Shafi Goldwasser, Frauke Kreuter, Omer Reingold, and Guy N. Rothblum. On the impossibility of separating intelligence from judgment: The computational intractability of filtering for ai alignment, 2025. URL https://arxiv.org/abs/2507.07341.

Nir Bitansky, Ran Canetti, Alessandro Chiesa, and Eran Tromer. Recursive composition and bootstrapping for SNARKs and proof-carrying data. In *Proceedings of the 45th Annual ACM Symposium on Theory of Computing (STOC 2013)*, pp. 111–120. ACM, 2013.

Nir Bitansky, Ran Canetti, Omer Paneth, and Alon Rosen. On the existence of extractable one-way functions. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing (STOC)*, pp. 505–514, 2014.

Nir Bitansky, Shafi Goldwasser, Abhishek Jain, Omer Paneth, Vinod Vaikuntanathan, and Brent Waters. Time-lock puzzles from randomized encodings. *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, 2016. URL https://api.semanticscholar.org/CorpusID:1963938.

Nir Bitansky, Alessandro Chiesa, Yuval Ishai, Rafail Ostrovsky, and Omer Paneth. Succinct non-interactive arguments via linear interactive proofs. *J. Cryptol.*, 35(3), July 2022. ISSN 0933-2790. doi: 10.1007/s00145-022-09424-4. URL https://doi.org/10.1007/s00145-022-09424-4.

Dan Boneh, Emily Shen, and Brent Waters. Strongly unforgeable signatures based on computational diffie-hellman. In *Proceedings of the 9th International Conference on Theory and Practice of Public-Key Cryptography*, PKC'06, pp. 229–240, Berlin, Heidelberg, 2006. Springer-Verlag. ISBN 3540338519. doi: 10.1007/11745853_15. URL https://doi.org/10.1007/11745853_15.

Dan Boneh, Joseph Bonneau, Benedikt Bünz, and Ben Fisch. Verifiable delay functions. In *Advances in Cryptology – CRYPTO 2018*, volume 10991 of *Lecture Notes in Computer Science*, pp. 757–788. Springer, 2018.

Avishek Joey Bose, Gauthier Gidel, Hugo Berard, Andre Cianflone, Pascal Vincent, Simon Lacoste-Julien, and William L. Hamilton. Adversarial example games. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Elette Boyle and Rafael Pass. Limits of extractability assumptions with distributional auxiliary input. In *Advances in Cryptology – ASIACRYPT 2015*, volume 9453 of *Lecture Notes in Computer Science*. Springer, 2015.

Sébastien Bubeck, Eric Price, and Ilya Razenshteyn. Adversarial examples from cryptographic pseudo-random generators, 2018.

Sébastien Bubeck, Yin Tat Lee, Eric Price, and Ilya Razenshteyn. Adversarial examples from computational constraints. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pp. 831–840. PMLR, 2019.

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.

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

Michael Clear and Ciarán McGoldrick. Bootstrappable identity-based fully homomorphic encryption. In *Cryptology and Network Security – CANS 2014*, pp. 1–19. Springer, 2014. URL https://eprint.iacr.org/2014/491.

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.

Francesco Croce, Sven Gowal, Thomas Brunner, Evan Shelhamer, Matthias Hein, and Taylan Cemgil. Evaluating the adversarial robustness of adaptive test-time defenses. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*. PMLR, 2022.

Edoardo Debenedetti, Jie Zhang, Mislav Balunović, Luca Beurer-Kellner, Marc Fischer, and Florian Tramèr. AgentDojo: A dynamic environment to evaluate prompt injection attacks and defenses for LLM agents, 2024. URL https://arxiv.org/abs/2406.13352.

Akshay Degwekar, Preetum Nakkiran, and Vinod Vaikuntanathan. Computational limitations in robust classification and win-win results. In *Proceedings of the Thirty-Second Conference on Learning Theory (COLT)*, volume 99 of *Proceedings of Machine Learning Research*. PMLR, 2019.

Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, November 1976.

Ran El-Yaniv et al. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11(5), 2010.

Logan Engstrom, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. A rotation and a translation suffice: Fooling cnns with simple transformations. *ArXiv*, abs/1712.02779, 2017. URL https://api.semanticscholar.org/CorpusID:21929206.

Naomi Ephraim, Cody Freitag, Ilan Komargodski, and Rafael Pass. Continuous verifiable delay functions. In *Advances in Cryptology – EUROCRYPT 2020*, volume 12107 of *Lecture Notes in Computer Science*. Springer, 2020.

Sanjam Garg, Somesh Jha, Saeed Mahloujifar, and Mohammad Mahmoody. Adversarially robust learning could leverage computational hardness. In Aryeh Kontorovich and Gergely Neu (eds.), *Algorithmic Learning Theory, ALT 2020, 8-11 February 2020, San Diego, CA, USA*, volume 117 of *Proceedings of Machine Learning Research*, pp. 364–385. PMLR, 2020. URL http://proceedings.mlr.press/v117/garg20a.html.

Craig Gentry, Amit Sahai, and Brent Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. Cryptology ePrint Archive, Paper 2013/340, 2013. URL https://eprint.iacr.org/2013/340.

Justin Gilmer, Luke Metz, Fartash Faghri, Samuel S. Schoenholz, Maithra Raghu, Martin Wattenberg, and Ian J. Goodfellow. Adversarial spheres. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Workshop Track Proceedings*, 2018. URL https://openreview.net/forum?id=SkthlLkPf.

Grzegorz Gluch, Berkant Turan, Sai Ganesh Nagarajan, and Sebastian Pokutta. The good, the bad and the ugly: Meta-analysis of watermarks, transferable attacks and adversarial defenses. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=NVDrWBwJTV.

Oded Goldreich. *Foundations of Cryptography: Volume 1, Basic Tools*. Cambridge University Press, 2001.

Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*, 28:270–299, 1984. URL https://api.semanticscholar.org/CorpusID:19616020.

Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on Computing*, 18(1):186–208, 1989. doi: 10.1137/0218012. URL https://doi.org/10.1137/0218012.

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. arXiv:2411.03279, 11 2024.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Ryan Greenblatt, Carson Denison, Benjamin Wright, Fabien Roger, Monte MacDiarmid, Sam Marks, Johannes Treutlein, Tim Belonax, Jack Chen, David Duvenaud, Akbir Khan, Julian Michael, Sören Mindermann, Ethan Perez, Linda Petrini, Jonathan Uesato, Jared Kaplan, Buck Shlegeris, Samuel R. Bowman, and Evan Hubinger. Alignment faking in large language models. *arXiv preprint arXiv:2412.14093*, 2024. doi: 10.48550/arXiv.2412.14093. URL https://arxiv.org/abs/2412.14093.

Kai Greshake, Sahar Abdelnabi, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection, 2023. URL https://arxiv.org/abs/2302.12173.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/e2c420d928d4bf8ce0ff2ec19b371514-Paper.pdf.

Yuval Ishai, Eyal Kushilevitz, and Rafail Ostrovsky. Sufficient conditions for collision-resistant hashing. In *Theory of Cryptography Conference (TCC 2005)*, volume 3378 of *Lecture Notes in Computer Science*. Springer, 2005.

Jiaming Ji, Tianyi Qiu, Boyuan Chen, Borong Zhang, Hantao Lou, Kaile Wang, Yawen Duan, Zhonghao He, Jiayi Zhou, Zhaowei Zhang, Fanzhi Zeng, Kwan Yee Ng, Juntao Dai, Xuehai Pan, Aidan O'Gara, Yingshan Lei, Hua Xu, Brian Tse, Jie Fu, Stephen Marcus McAleer, Yaodong Yang, Yizhou Wang, Song-Chun Zhu, Yike Guo, and Wen Gao. Ai alignment: A comprehensive survey. *ArXiv*, abs/2310.19852, 2023. URL https://api.semanticscholar.org/CorpusID:264743032.

Richard M Karp and Richard J Lipton. Some connections between nonuniform and uniform complexity classes. In *Proceedings of the twelfth annual ACM symposium on Theory of computing*, pp. 302–309, 1980.

Michael J Kearns and Umesh Vazirani. *An introduction to computational learning theory*. MIT press, 1994.

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, 1994.

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.

Saeed Mahloujifar and Mohammad Mahmoody. Can adversarially robust learning leverage computational hardness? In *Proceedings of the 30th International Conference on Algorithmic Learning Theory (ALT)*, volume 98 of *Proceedings of Machine Learning Research*. PMLR, 2019.

Mohammad Mahmoody and Ameer Mohammed. On the power of hierarchical identity-based encryption. In *Advances in Cryptology – EUROCRYPT 2016*, 2016. URL https://www.iacr.org/archive/eurocrypt2016/96650178/96650178.pdf.

Tavish McDonald, Bo Lei, Stanislav Fort, Bhavya Kailkhura, and Brian Bartoldson. Get RICH or die scaling: Profitably trading inference compute for robustness. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026. arXiv:2510.06790.

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

Laurent Meunier, Meyer Scetbon, Rafael Pinot, Jamal Atif, and Yann Chevaleyre. Mixed Nash equilibria in the adversarial examples game. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139 of *Proceedings of Machine Learning Research*. PMLR, 2021.

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

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Moni Naor, Omer Paneth, and Guy N. Rothblum. Incrementally verifiable computation via incremental PCPs. Cryptology ePrint Archive, Paper 2019/1407, 2019. URL https://eprint.iacr.org/2019/1407.

Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*. PMLR, 2022.

Ambar Pal and René Vidal. A game theoretic analysis of additive adversarial attacks and defenses. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Nicolas Papernot, Patrick Mcdaniel, and Ian J. Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *ArXiv*, abs/1605.07277, 2016. URL https://api.semanticscholar.org/CorpusID:17362994.

Rafael Pinot, Raphael Ettedgui, Geovani Rizk, Yann Chevaleyre, and Jamal Atif. Randomization matters: How to defend against strong adversarial attacks. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*. PMLR, 2020.

Muni Sreenivas Pydi and Varun Jog. Adversarial risk via optimal transport and optimal couplings. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*. PMLR, 2020.

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.

Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):1–40, 2009.

Ronald L Rivest, Adi Shamir, and Len Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

Ronald L. Rivest, Adi Shamir, and David A. Wagner. Time-lock puzzles and timed-release crypto. Technical Report MIT/LCS/TR-684, Massachusetts Institute of Technology, 1996.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A. Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*. PMLR, 2020.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In Yoshua Bengio and Yann LeCun (eds.), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL http://arxiv.org/abs/1312.6199.

Florian Tramer. Detecting adversarial examples is (Nearly) as hard as classifying them. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 21692–21702. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/tramer22a.html.

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.

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

Leslie G Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

Paul Valiant. Incrementally verifiable computation or proofs of knowledge imply time/space efficiency. In *Theory of Cryptography Conference (TCC 2008)*, volume 4948 of *Lecture Notes in Computer Science*, pp. 1–18. Springer, 2008.

V. N. Vapnik and A. Ya. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16(2):264–280, 1971. doi: 10.1137/1116025. URL https://doi.org/10.1137/1116025.

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., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/a00548031e4647b13042c97c922fadf1-Paper-Conference.pdf.

Eric Wong and J. Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. In Jennifer G. Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5283–5292. PMLR, 2018. URL http://proceedings.mlr.press/v80/wong18a.html.

Tong Wu, Chong Xiang, Jiachen T. Wang, Weichen Yu, Chawin Sitawarin, Vikash Sehwag, and Prateek Mittal. Does more inference-time compute really help robustness?, 2025.

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.

Wojciech Zaremba, Evgenia Nitishinskaya, Boaz Barak, Stephanie Lin, Sam Toyer, Yaodong Yu, Rachel Dias, Eric Wallace, Kai Xiao, Johannes Heidecke, and Amelia Glaese. Trading inference-time compute for adversarial robustness, 2025.

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.