

An Information-Theoretic Explanation for the Adversarial Fragility of AI Classifiers

Hui Xie, Jirong Yi, Weiyu Xu, and Raghu Mudumbai
Department of Electrical and Computer Engineering, University of Iowa

Abstract—We present a simple hypothesis about a compression property of artificial intelligence (AI) classifiers and present theoretical arguments to show that this hypothesis successfully accounts for the observed fragility of AI classifiers to small adversarial perturbations. We also propose a new method for detecting when small input perturbations cause classifier errors, and show theoretical guarantees for the performance of this detection method. We present experimental results with a voice recognition system to demonstrate this method. The ideas in this paper are motivated by a simple analogy between AI classifiers and the standard Shannon model of a communication system.¹

I. INTRODUCTION

Recent advances in machine learning have led to the invention of complex classification systems that are very successful in detecting features in datasets such as images, hand-written texts, or audios. However, recent works have also discovered what appears to be a universal property of AI classifiers: vulnerability to small adversarial perturbations. Specifically, we know that it is possible to design “adversarial attacks” that manipulate the output of AI classifiers arbitrarily by making small carefully-chosen modifications to the input. Many such successful attacks only require imperceptibly small perturbations of the inputs, which makes these attacks almost undetectable. Thus AI classifiers exhibit two seemingly contradictory properties: (a) high classification accuracy even in very noisy conditions, and (b) high sensitivity to very small adversarial perturbations. In this paper, we will use the term “adversarial fragility” to refer to this property (b).

The importance of the adversarial fragility problem is widely recognized in the AI community and there now exists a vast and growing literature studying this property, see e.g. [1] for a comprehensive survey. This work, however, has not yet resulted in a consensus on two important questions: (a) a theoretical explanation for adversarial fragility, and (b) a general and systematic defense against adversarial attacks. Instead, we currently have multiple competing theoretical explanations, multiple defense strategies based on both theoretical and heuristic ideas and many methods for generating adversarial examples for AI classifiers. Theoretical hypotheses from the literature include (a) quasi-linearity/smoothness of the decision function in AI classifiers [2], (b) high curvature of the decision boundary [3] and (c) closeness of the classification boundary to the data sub-manifold [4]. Defenses against adversarial attacks have also evolved from early methods using gradient masking [5], to more sophisticated recent methods

such as adversarial training where an AI system is specifically subjected to adversarial attacks as part of its training process [6], and defensive distillation [7]. These new defenses in turn motivate the development of more sophisticated attacks [8] in an ongoing arms race.

In this paper, we show that property “adversarial fragility” is an unavoidable consequence of a simple “compression” hypothesis about AI classifiers. This hypothesis is illustrated in Fig. 2: we assume that the output of AI classifiers is a function of a highly compressed version of the input. More precisely, we assume that the output of AI classifiers is a function of an intermediate set of variables of much smaller dimension than the input. The intuition behind this hypothesis is as follows. AI classifiers typically take high-dimensional inputs e.g. image pixels, audio samples, and produce a discrete *label* as output. The input signals (a) contain a great deal of redundancy, and (b) depend on a large number of irrelevant variables that are unrelated to the output labels. Efficient classifiers, therefore, must remove a large amount of redundant and/or irrelevant information from the inputs before making a classification decision. Indeed, a classifier that *generalizes* well, must, by definition, be insensitive to as many non-essential input features as possible. We show in this paper that adversarial fragility is an immediate and necessary consequence of this “compression” property.

Certain types of AI systems can be shown to satisfy the compression property simply as a consequence of their structure. For instance, AI classifiers for the MNIST dataset [9] typically feature a final layer in the neural network architecture that consists of softmax over a 10×1 real-numbered vector corresponding to the 10 different label values; this amounts to a substantial dimension reduction from the $28 \times 28 = 784$ dimensional pixel vector at the inputs. More generally, there is some empirical evidence showing that AI classifiers actively compress their inputs during their training process [10].

Our proposed explanation of adversarial fragility also immediately leads to an obvious and very powerful defense: if we enhance a classifier with a generative model that at least partially “decompresses” the classifier’s output, and compare it with the raw input signal, it becomes easy to check when adversarial attacks produce classifier outputs that are inconsistent with their inputs. While we present some simple experimental results to validate our theory, our focus here is on the theoretical ideas; the important and challenging problem of designing good generative models to implement the proposed defense for general AI classification systems

¹The first two authors contributed equally.

is deferred to future work. Interestingly, while our theory is novel, other researchers have recently developed defenses for AI classifiers against adversarial attacks that are consistent with our proposed approach [11], [12].

II. PROBLEM STATEMENT

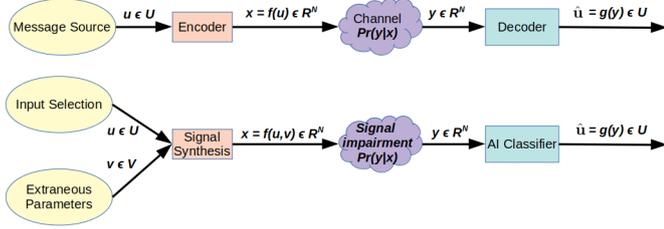


Fig. 1. Top: standard abstract model of a communication system; Bottom: abstract model of an AI classifier system.

An AI classifier can be defined as a system that takes a high-dimensional vector as input and maps it to a discrete set of labels. As an example, a voice-recognition AI takes as input a time series containing the samples of an audio signal and outputs a string representing a sentence in English (or other spoken language). More concretely, consider Fig. 1 which explores a simple analogy between an AI classification system and a digital communication system.

The purpose of the AI system in Fig. 1 is to estimate the state of the world $\mathbf{u} \in \mathcal{X}$ where the set of all possible world states \mathcal{X} is assumed to be finite and are enumerated as $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_{N_u}$, where N_u is the size of \mathcal{X} . The input $\mathbf{y} \in \mathbb{R}^N$ to the AI classifier is a noisy version of signals $\mathbf{x} \in \mathbb{R}^N$, and \mathbf{x} depend on \mathbf{u} and on a number of *extraneous parameters* $\mathbf{v} \in \mathcal{V}$. Note that the state \mathbf{u}_i is uniquely determined by its index or “label” i . The output of the AI classifier is a state estimate $\hat{\mathbf{u}} \in \mathcal{X}$, or equivalently, its label.

The AI classifier in Fig. 1 is clearly analogous to a communication decoder: it looks at a set of noisy observations and attempts to decide which out of a set of possible input signals \mathbf{x} was originally “transmitted” over the “channel”, which in the AI system models all signal impairments such as distortion, random noise and hostile attackers.

The “Signal Synthesis” block in the AI system maps input features into an observable signal \mathbf{x} . In the abstract model of Fig. 1, the synthesis function $\mathbf{f}(\cdot)$ is deterministic with all random effects being absorbed into the “channel” without loss of generality. Note that while the encoder in the communication system is under the control of its designers, the signal synthesis in an AI system is determined by physical laws and is not in our control. However, the most important difference between communication and AI systems is the presence of the nuisance parameters \mathbf{v} . For instance, in a voice recognition system, the input features consist of the text being spoken (\mathbf{u}) and also a very large number of other characteristics (\mathbf{v}) of the speaker’s voice such as pitch, accent, dialect, loudness, emotion etc. which together determine the mapping from a text to an audio signal. Thus there are a very large number

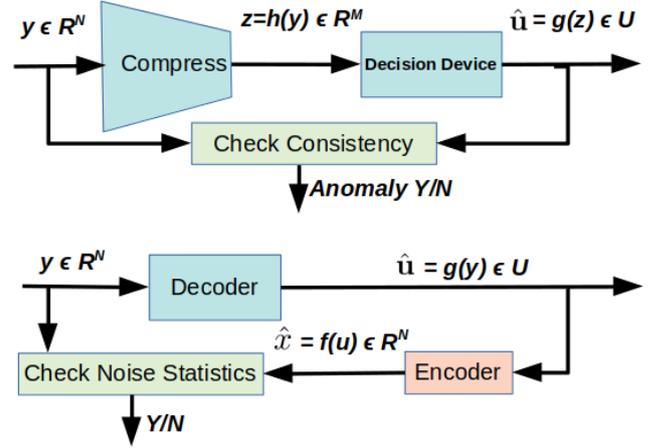


Fig. 2. AI classifier using a information compression process and its analogy with a communication decoder

of different “codewords” $\mathbf{c}_1 = \mathbf{f}(\mathbf{u}_1, \mathbf{v}_1), \mathbf{c}_2 = \mathbf{f}(\mathbf{u}_1, \mathbf{v}_2), \dots$ that encode the same label \mathbf{u}_1 . Let us define the “codeword set” for label i , $i = 1 \dots N_u$:

$$\mathcal{X}_i \doteq \{\mathbf{c} \in \mathbb{R}^N : \exists \mathbf{v}, \mathbf{c} = \mathbf{f}(\mathbf{u}_i, \mathbf{v})\} \quad (1)$$

We assume that the codeword sets \mathcal{X}_i satisfy:

$$\min_{\forall i,j, i \neq j} \min_{\mathbf{c}_i \in \mathcal{X}_i, \mathbf{c}_j \in \mathcal{X}_j} \|\mathbf{c}_i - \mathbf{c}_j\| \geq 2r_0 \quad (2)$$

for some $r_0 > 0$, where $\|\cdot\|$ represents ℓ_2 norm. In other words, all valid codewords corresponding to different labels $i \neq j$ are separated by at least a distance $2r_0$. In the voice recognition example, under this assumption audio signals corresponding to two different sentences must sound different. This guarantees the existence of the ideal classifier defined as the function $q^*(\mathbf{x}) : \mathbb{R}^N \rightarrow \mathcal{X}$ that satisfies $q^*(\mathbf{f}(\mathbf{u}_i, \mathbf{v})) = \mathbf{u}_i, \forall i, \mathbf{v} \in \mathcal{V}$. By definition, the ideal classifier maps any valid input signal to the correct label in the absence of noise.

Fig. 2 shows an abstract model of a classifier that is constrained to make final classification decisions based on only a compressed version \mathbf{z} of \mathbf{y} . Specifically, we assume that there exists a compression function $\mathbf{h} : \mathbb{R}^N \rightarrow \mathbb{R}^M$, where $M \ll N$ such that the classifier output $q(\mathbf{y}) : \mathbb{R}^N \rightarrow \mathcal{X}$ can be written as $q(\mathbf{y}) = g(\mathbf{h}(\mathbf{y}))$, where $g : \mathbb{R}^M \rightarrow \mathcal{X}$ is a decision function. We define the “compressed codeword sets” as $\mathcal{Z}_i \doteq \{\mathbf{z} \in \mathbb{R}^M : \exists \mathbf{u} \in \mathcal{U}_i, \mathbf{v} \in \mathcal{V}, \mathbf{h}(\mathbf{f}(\mathbf{u}, \mathbf{v})) = \mathbf{z}\}$. We will assume that the sets \mathcal{Z}_i are disjoint so that the compression map $\mathbf{h}(\mathbf{y})$ preserves information in \mathbf{y} about the label i .

We will show that a classifier constrained to use only $\mathbf{h}(\mathbf{y})$ for decoding, even if designed optimally, can retain its robustness to random noise \mathbf{w} , but is necessarily vulnerable to adversarial attacks that are significantly smaller in magnitude. By contrast, uncompressed classifiers can be robust to both random and worse-case noise. In other words, we show that adversarial fragility can be explained as an artifact of compression or dimension reduction in decoders.

Our method for detecting adversarial attacks is based on the idea of at least partially “decompressing” the output of the classifier and checking it for consistency against the raw observations \mathbf{y} . Specifically, suppose the classifier outputs label j for input signal \mathbf{y} . Define $\mathbf{c}_j(\mathbf{y})$ as:

$$\mathbf{c}_j(\mathbf{y}) \doteq \arg \min_{\mathbf{c} \in \mathcal{X}_j} \|\mathbf{y} - \mathbf{c}\| \quad (3)$$

If we observe that $\|\mathbf{y} - \mathbf{c}_j(\mathbf{y})\|$ is abnormally large, this means that the observed signal \mathbf{y} is far from any valid codeword $\mathbf{f}(\mathbf{u}_j, \mathbf{v})$ with label j and we conclude that label j is inconsistent with observations \mathbf{y} . This, however, requires a feasible method for calculating $\mathbf{c}_j(\mathbf{y})$ for a label j and signal \mathbf{y} . This is basically a *denoising* operation that outputs a noise-free codeword $\mathbf{c}_j(\mathbf{y})$ given a label j and noisy observation \mathbf{y} . Generative models are capable of performing such a *denoising* operation. We do not discuss the design of such models in this paper; instead we will show that under mild assumptions on the encoding function $\mathbf{f}(\cdot)$, we can provide theoretical guarantees on a detector *assuming a well-functioning generative model*.

III. THEORETICAL ANALYSIS

In this section, we consider a signal $\mathbf{x} \in \mathbb{R}^N$, which can be a noisy version of a codeword. Without loss of generality, we assume that an ideal classifier will classify \mathbf{x} to label 1, and assume that the closest codeword to \mathbf{x} is $\mathbf{c}_1 = \mathbf{f}(\mathbf{u}_1, \mathbf{v}_1)$ for \mathbf{u}_1 and a certain \mathbf{v}_1 . For any $i \neq 1$, we also define \mathbf{c}_i as the codeword with label i that is closest to \mathbf{x} : $\mathbf{c}_i \doteq \arg \min_{\mathbf{c} \in \mathcal{X}_i} \|\mathbf{x} - \mathbf{c}\|$. We define the sets \mathcal{S}_1 and \mathcal{S}_i as the spheres of size r around \mathbf{c}_1 and \mathbf{c}_i respectively, namely $\mathcal{S}_1 \doteq \{\mathbf{b} \in \mathbb{R}^N : \|\mathbf{b} - \mathbf{c}_1\| < r\}$ and $\mathcal{S}_i \doteq \{\mathbf{b} \in \mathbb{R}^N : \|\mathbf{b} - \mathbf{c}_i\| < r\}$. We assume that $\mathbf{x} \in \mathcal{S}_1$. For simplicity of analysis, we assume that, for a vector $\mathbf{y} \in \mathbb{R}^N$, the classifier $q(\mathbf{y})$ outputs label i if and only if $h(\mathbf{y}) = h(\mathbf{b})$ for a certain $\mathbf{b} \in \mathcal{S}_i$.

We consider the problem of finding the smallest targeted perturbation \mathbf{w} in magnitude which fools the decoder $q(\mathbf{x} + \mathbf{w})$ into outputting label $i \neq 1$. Formally, for any $\mathbf{x} \in \mathbb{R}^N$, we define the minimum perturbation size $d_i(\mathbf{x})$ needed for target label i as:

$$d_i(\mathbf{x}) \doteq \min_{\mathbf{w} \in \mathbb{R}^N, \mathbf{t} \in \mathcal{S}_i} \|\mathbf{w}\|, \text{ s.t. } \mathbf{h}(\mathbf{x} + \mathbf{w}) = \mathbf{h}(\mathbf{t}). \quad (4)$$

Let us define a quantity $d(\mathbf{x}, \mathbf{t})$, which we term as “effective distance between \mathbf{x} and \mathbf{t} with respect to function $\mathbf{h}(\cdot)$ ” as $d(\mathbf{x}, \mathbf{t}) = \min_{\mathbf{w} \in \mathbb{R}^N, \mathbf{h}(\mathbf{x} + \mathbf{w}) = \mathbf{h}(\mathbf{t})} \|\mathbf{w}\|$. Then for any vector $\mathbf{x} \in \mathbb{R}^N$, we can use (4) to upper bound the smallest required perturbation size $d_i(\mathbf{x}) \leq \min_{\mathbf{t} \in \mathcal{S}_i} d(\mathbf{x}, \mathbf{t})$.

For an $\epsilon > 0$ and $l > 0$, we say a classifier has (ϵ, l) -robustness at signal \mathbf{x} , if $\mathbb{P}(g(\mathbf{h}(\mathbf{x} + \mathbf{w})) = g(\mathbf{h}(\mathbf{x}))) \geq 1 - \epsilon$, where $\mathbf{w} \in \mathbb{R}^N$ is randomly sampled uniformly on a sphere² of radius l , and \mathbb{P} means probability. In the following, we will show that for a small ϵ , compressed classifiers can still

²Defined for some given norm, which we will take to be ℓ_2 norm throughout this paper.

have (ϵ, l) -robustness for $l \gg d_i(\mathbf{x})$, namely the classifier can tolerate large random perturbations while being vulnerable to much smaller adversarial attacks.

A. Classifiers with Linear Compression Functions

We first consider the special case where the compression function $\mathbf{h}(\cdot)$ is linear, namely $\mathbf{h}(\mathbf{y}) = A\mathbf{y}$ with $A \in \mathbb{R}^{M \times N}$, $M \ll N$. While this may not be a reasonable model for practical AI systems, analysis of linear compression functions will yield analytical insights that generalize to nonlinear $\mathbf{h}(\cdot)$ as we show later.

Theorem 1: Let $\mathbf{y} \in \mathbb{R}^N$ be the input to a classifier, which makes decisions based on the compression function $\mathbf{z} = \mathbf{h}(\mathbf{y}) = A\mathbf{y}$, where the elements of $A \in \mathbb{R}^{M \times N}$ ($M \ll N$) are i.i.d. following the standard Gaussian distribution $\mathcal{N}(0, 1)$. Let $B_i = \{\mathbf{z} : \mathbf{z} = A\mathbf{b}, \mathbf{b} \in \mathcal{S}_i\}$ be the compressed image of \mathcal{S}_i . Then the following statements hold for arbitrary $\epsilon > 0$, $i \neq 1$, and a big enough M .

1) With high probability (over the distribution of A), an attacker can design a targeted adversarial attack \mathbf{w} with $\|\mathbf{w}\|_2 \leq \sqrt{1 + \epsilon} \sqrt{\frac{M}{N}} \|\mathbf{c}_i - \mathbf{x}\|_2 - r$ such that the classifier is fooled into classifying the signal $\mathbf{y} = \mathbf{x} + \mathbf{w}$ into label i . Moreover, with high probability (over the distribution of A), an attacker can design an (untargeted) adversarial perturbation \mathbf{w} with $\|\mathbf{w}\|_2 \leq r - \sqrt{1 - \epsilon} \sqrt{\frac{M}{N}} \|\mathbf{x} - \mathbf{c}_1\|$ such that the classifier will not classify $\mathbf{y} = \mathbf{x} + \mathbf{w}$ into label 1.

2) Suppose that \mathbf{w} is randomly uniformly sampled from a sphere of radius l in \mathbb{R}^N . With high probability (over the distribution of A and \mathbf{w}), if $l < \sqrt{\frac{1 - \epsilon}{1 + \epsilon}} \|\mathbf{c}_i - \mathbf{x}\|_2 - \frac{r}{\sqrt{1 + \epsilon} \sqrt{\frac{M}{N}}}$, the classifier will not classify $\mathbf{y} = \mathbf{x} + \mathbf{w}$ into label i . Moreover, with high probability (over the distribution of A and \mathbf{w}), if $l < (1 - \epsilon) \sqrt{\frac{N}{M}} \sqrt{r^2 - \frac{M}{N} \|\mathbf{x} - \mathbf{c}_1\|^2}$, the classifier still classifies the $\mathbf{y} = \mathbf{x} + \mathbf{w}$ into label 1 correctly.

3) Let \mathbf{w} represent a successful adversarial perturbation i.e. the classifier outputs target label $i \neq 1$ for the input $\mathbf{y} = \mathbf{x} + \mathbf{w}$. Then as long as $\|\mathbf{w}\|_2 < \min_{\mathbf{c}_i \in \mathcal{X}_i} \|\mathbf{c}_i - \mathbf{x}\| - r$, our adversarial detection approach will be able to detect the attack.

Proof: 1) We first look at the targeted attack case. For linear decision statistics, $d(\mathbf{x}, \mathbf{t}) = \min_{\mathbf{w} \in \mathbb{R}^N, A(\mathbf{x} + \mathbf{w}) = A(\mathbf{t})} \|\mathbf{w}\|$. Solving this optimization problem, we know the optimal \mathbf{w} is given by $\mathbf{w} = A^\dagger A(\mathbf{t} - \mathbf{x})$, where A^\dagger is the Moore-Penrose inverse of A . We can see that \mathbf{w} is nothing but the projection of $(\mathbf{t} - \mathbf{x})$ onto the row space of A . We denote the projection matrix as $P = A^\dagger A$. Then the smallest magnitude of an effective adversarial perturbation is upper bounded by

$$\min_{\mathbf{t} \in \mathcal{S}_i} d(\mathbf{x}, \mathbf{t}) = \min_{\mathbf{t} \in \mathcal{S}_i} \|A^\dagger A(\mathbf{t} - \mathbf{x})\| = \min_{\mathbf{t} \in \mathcal{S}_i} \|P(\mathbf{t} - \mathbf{x})\|.$$

For $\mathbf{t} \in \mathcal{S}_i$, we have $\|P(\mathbf{t} - \mathbf{x})\| = \|P(\mathbf{c}_i - \mathbf{x}) + P(\mathbf{t} - \mathbf{c}_i)\| \geq \|P(\mathbf{c}_i - \mathbf{x})\| - \|P(\mathbf{t} - \mathbf{c}_i)\|$. One can show that, when $\mathcal{S}_i = \{\mathbf{t} \mid \|\mathbf{t} - \mathbf{c}_i\| \leq r\}$, we can always achieve the equality, namely $\min_{\mathbf{t} \in \mathcal{S}_i} \|P(\mathbf{t} - \mathbf{x})\| = \|P(\mathbf{c}_i - \mathbf{x})\| - r$.

Now we evaluate $\|P(\mathbf{c}_i - \mathbf{x})\|$. Suppose that A 's elements are i.i.d., and follow the standard zero-mean Gaussian distribution $\mathcal{N}(0, 1)$, then the random projection P is uniformly

sampled from the Grassmannian $Gr(M, \mathbb{R}^N)$. We can see that the distribution of $\|P(\mathbf{c}_i - \mathbf{x})\|$ is the same as the distribution of the magnitude of the first M elements of $\|(\mathbf{c}_i - \mathbf{x})\|_{\mathbf{o}}/\|\mathbf{o}\|$, where $\mathbf{o} \in \mathbb{R}^N$ is a vector with its elements being i.i.d. following the standard Gaussian distribution $\mathcal{N}(0, 1)$. From the concentration of measure, for any positive $\epsilon < 1$,

$$\mathbb{P}\left(\|P(\mathbf{c}_i - \mathbf{x})\| \leq \sqrt{1 - \epsilon}\|\mathbf{c}_i - \mathbf{x}\|\sqrt{\frac{M}{N}}\right) \leq e^{-\frac{M\epsilon^2}{4}},$$

$$\mathbb{P}\left(\|P(\mathbf{c}_i - \mathbf{x})\| \geq \sqrt{1 + \epsilon}\|\mathbf{c}_i - \mathbf{x}\|\sqrt{\frac{M}{N}}\right) \leq e^{-\frac{M\epsilon^2}{12}}.$$

Then when M is big enough, $\min_{\mathbf{t} \in \mathcal{S}_i} \|P(\mathbf{t} - \mathbf{x})\| \leq \sqrt{1 + \epsilon}\sqrt{\frac{M}{N}}\|\mathbf{c}_i - \mathbf{x}\| - r$ with high probability, for arbitrary $\epsilon > 0$.

Now let us look at what perturbation \mathbf{w} we need such that $A(\mathbf{x} + \mathbf{w})$ is not in B_1 . One can show that $A(\mathbf{x} + \mathbf{w})$ is outside B_1 if and only if, $\|P(\mathbf{x} - \mathbf{c}_1 + \mathbf{w})\| > r$. Then by the triangular inequality, the attacker can take an attack \mathbf{w} with $\|\mathbf{w}\| > r - \|P(\mathbf{x} - \mathbf{c}_1)\|$, which is no bigger than $r - \sqrt{1 - \epsilon}\sqrt{\frac{M}{N}}\|\mathbf{x} - \mathbf{c}_1\|_2$ with high probability, for arbitrary $\epsilon > 0$ and big enough M .

2) If and only if $\mathbf{h}(\mathbf{x} + \mathbf{w}) \neq \mathbf{h}(\mathbf{t})$, $\forall \mathbf{t} \in \mathcal{S}_i$, \mathbf{w} will not fool the classifier into label i . If $\mathbf{h}(\mathbf{y}) = A\mathbf{y}$, “ $\mathbf{h}(\mathbf{x} + \mathbf{w}) \neq \mathbf{h}(\mathbf{t})$, $\forall \mathbf{t} \in \mathcal{S}_i$ ” is equivalent to “ $\|A(\mathbf{x} + \mathbf{w} - \mathbf{t})\| \neq 0$, $\forall \mathbf{t} \in \mathcal{S}_i$ ”, which is in turn equivalent to “ $\|P(\mathbf{x} + \mathbf{w} - \mathbf{t})\| \neq 0$, $\forall \mathbf{t} \in \mathcal{S}_i$ ”, where P is the projection onto the row space of A . Assuming that \mathbf{w} is uniformly randomly sampled from a sphere in \mathbb{R}^N of radius $l < \sqrt{\frac{1 - \epsilon}{1 + \epsilon}}\|\mathbf{c}_i - \mathbf{x}\| - \frac{r}{\sqrt{1 + \epsilon}\sqrt{\frac{M}{N}}}$, then

$$\|P(\mathbf{x} + \mathbf{w} - \mathbf{t})\| = \|P(\mathbf{c}_i - \mathbf{x}) + P(\mathbf{t} - \mathbf{c}_i) - P\mathbf{w}\|$$

$$\geq \|P(\mathbf{c}_i - \mathbf{x})\| - \|P(\mathbf{t} - \mathbf{c}_i)\| - \|P\mathbf{w}\|.$$

From the concentration inequality, $\mathbb{P}\left(\|P\mathbf{w}\| \geq \sqrt{1 + \epsilon}\|\mathbf{w}\|\sqrt{\frac{M}{N}}\right) \leq e^{-\frac{M(\epsilon^2/2 - \epsilon^3/3)}{2}}$. Thus if M is big enough, with high probability, $\|P(\mathbf{x} + \mathbf{w} - \mathbf{t})\| \geq \sqrt{1 - \epsilon}\|\mathbf{c}_i - \mathbf{x}\|_2\sqrt{\frac{M}{N}} - r - \sqrt{1 + \epsilon}\|\mathbf{w}\|\sqrt{\frac{M}{N}}$. If $\|\mathbf{w}\| = l$, $\|P(\mathbf{x} + \mathbf{w} - \mathbf{t})\| > 0$.

Now let us look at what magnitude we need for a random perturbation \mathbf{w} such that $A(\mathbf{x} + \mathbf{w})$ is in B_1 with high probability. We know $A(\mathbf{x} + \mathbf{w})$ is in B_1 if and only if, $\|P(\mathbf{x} - \mathbf{c}_1 + \mathbf{w})\| < r$. Through a large deviation analysis, one can show that, for any $\delta > 0$ and big enough M , $\|P(\mathbf{x} - \mathbf{c}_1 + \mathbf{w})\|$ is smaller than $(1 + \delta)\sqrt{\frac{M}{N}}\|\mathbf{x} - \mathbf{c}_1\|_2^2 + \frac{M}{N}l^2$ and bigger than $(1 - \delta)\sqrt{\frac{M}{N}}\|\mathbf{x} - \mathbf{c}_1\|_2^2 + \frac{M}{N}l^2$ with high probability. Thus, for an arbitrary $\epsilon > 0$, if $l < (1 - \epsilon)\sqrt{\frac{N}{M}}\sqrt{r^2 - \frac{M}{N}\|\mathbf{x} - \mathbf{c}_1\|_2^2}$, $\|P(\mathbf{x} - \mathbf{c}_1 + \mathbf{w})\| < r$ with high probability, implying the AI classifier still classifies the $\mathbf{y} = \mathbf{x} + \mathbf{w}$ into Class 1 correctly.

3) Suppose that an AI classifier classifies the input signal $\mathbf{y} = \mathbf{x} + \mathbf{w}$ into label i . We propose to check whether \mathbf{y} belongs to \mathcal{S}_i . In our model, the signal \mathbf{y} belongs to \mathcal{S}_i only if $\min_{\mathbf{c}_i \in \mathcal{X}_i} \|\mathbf{y} - \mathbf{c}_i\| \leq r$. Let us take any codeword $\mathbf{c}_i \in \mathcal{X}_i$. We show that when $\|\mathbf{w}\| < \|\mathbf{c}_i - \mathbf{x}\| - r$, we can always detect the adversarial attack if the AI classifier misclassifies \mathbf{y}

to that codeword corresponding to label i . In fact, $\|\mathbf{y} - \mathbf{c}_i\| = \|(\mathbf{x} + \mathbf{w}) - \mathbf{c}_i\|_2$, which is no smaller than $\|\mathbf{c}_i - \mathbf{x}\| - \|\mathbf{w}\| \geq \|\mathbf{c}_i - \mathbf{x}\| - (\|\mathbf{c}_i - \mathbf{x}\| - r) > r$.

We note that $\|\mathbf{w}\| \leq \min_{\mathbf{c}_i \in \mathcal{X}_i} \|\mathbf{c}_i - \mathbf{x}\| - r$ means $\|\mathbf{w}\|_2 < \|\mathbf{c}_i - \mathbf{x}\| - r$ for every codeword \mathbf{c}_i , thus implying that the adversary attack detection technique can detect that \mathbf{y} is at more than distance r from every codeword from \mathcal{X}_i . ■

B. Nonlinear Decision Statistics in AI Classifiers

In this subsection, we show that an AI classifier using nonlinear compressed decision statistics $\mathbf{h}(\mathbf{y}) \in \mathbb{R}^M$ is significantly more vulnerable to adversarial attacks than to random perturbations. We will quantify the gap between how much a random perturbation and a well-designed adversarial attack affect $\mathbf{h}(\mathbf{y})$.

Before we proceed, we introduce some technical lemmas which are used to establish the gap quantification in Theorem 3.

Lemma 2: (Section III in [13]) For a random matrix $F \in \mathbb{R}^{M \times N}$, $M > N$ with every entry being i.i.d. random variable distributed accord to Gaussian distribution $\mathcal{N}(0, 1/M)$, we can have

$$\mathbb{P}\left(\sigma_{max}(F) > 1 + \sqrt{\frac{N}{M}} + o(1) + t\right) \leq e^{-Mt^2/2}, \forall t > 0,$$

and

$$\mathbb{P}\left(\sigma_{min}(F) < 1 - \sqrt{\frac{N}{M}} + o(1) - t\right) \leq e^{-Mt^2/2}, \forall t > 0,$$

where $\sigma_{max}(F)$ is the maximal singular value of F , the $\sigma_{min}(F)$ is the smallest singular value of F , and the $o(1)$ is a small term tending to zero as $M \rightarrow \infty$.

From Lemma 2, we see that for a random matrix $F \in \mathbb{R}^{M \times N}$ ($M < N$) with all entries i.i.d. distributed according to standard Gaussian distribution $\mathcal{N}(0, 1)$, the scaled matrix $\frac{1}{\sqrt{N}}F^T$ will satisfy for all $t > 0$

$$\mathbb{P}\left(\frac{1}{\sqrt{N}}\sigma_{max}(F) > 1 + \sqrt{\frac{M}{N}} + o(1) + t\right) \leq e^{-Nt^2},$$

and

$$\mathbb{P}\left(\frac{1}{\sqrt{N}}\sigma_{min}(F) < 1 - \sqrt{\frac{M}{N}} + o(1) - t\right) \leq e^{-Nt^2/2}, \quad (5)$$

since

$$\sigma_i\left(\frac{1}{\sqrt{N}}F^T\right) = \frac{1}{\sqrt{N}}\sigma_i(F),$$

where $\sigma_i(F)$ is the i -th largest singular value of F .

Theorem 3: Let us assume that the nonlinear function $\mathbf{h}(\mathbf{x}) : \mathbb{R}^N \rightarrow \mathbb{R}^M$ is differentiable at \mathbf{x} . For $\epsilon > 0$, we define $\alpha(\epsilon) = \max_{\|\mathbf{w}\| \leq \epsilon} (\|\mathbf{h}(\mathbf{x} + \mathbf{w}) - \mathbf{h}(\mathbf{x})\|)$, and $\beta(\mathbf{o}, \epsilon) = \|\mathbf{h}(\mathbf{x} + \epsilon\mathbf{o}) - \mathbf{h}(\mathbf{x})\|$, where \mathbf{o} is uniformly randomly sampled from a unit sphere. Then $\lim_{\epsilon \rightarrow 0} \frac{\alpha(\epsilon)}{E_{\mathbf{o}}\{\beta(\mathbf{o}, \epsilon)\}} \geq \sqrt{\frac{N}{M}}$, where $E_{\mathbf{o}}$ means expectation over the distribution of \mathbf{o} . If we assume

that the entries of the Jacobian matrix $\nabla \mathbf{h}(\mathbf{x}) \in \mathbb{R}^{M \times N}$ are i.i.d. distributed following the standard Gaussian distribution $\mathcal{N}(0, 1)$, then, when N is big enough, with high probability, $\lim_{\epsilon \rightarrow 0} \frac{\alpha(\epsilon)}{E_{\mathbf{o}}\{\beta(\mathbf{o}, \epsilon)\}} \geq (1 - \delta) \sqrt{\frac{N+M}{M}}$ for any $\delta > 0$.

Proof: From the Taylor expansion, we know

$$\mathbf{h}(\mathbf{x} + \mathbf{w}) = \mathbf{h}(\mathbf{x}) + \nabla \mathbf{h}(\mathbf{x})\mathbf{w} + \begin{bmatrix} o(\|\mathbf{w}\|_2^2) \\ \vdots \\ o(\|\mathbf{w}\|_2^2) \end{bmatrix} \quad (6)$$

and

$$\mathbf{h}(\mathbf{x} + \epsilon \mathbf{o}) = \mathbf{h}(\mathbf{x}) + \epsilon \nabla \mathbf{h}(\mathbf{x})\mathbf{o} + \begin{bmatrix} o(\epsilon^2 \|\mathbf{o}\|_2^2) \\ \vdots \\ o(\epsilon^2 \|\mathbf{o}\|_2^2) \end{bmatrix}, \quad (7)$$

where $o(\|\mathbf{w}\|_2^2) \rightarrow 0$ and $o(\epsilon^2 \|\mathbf{o}\|_2^2) \rightarrow 0$ as $\epsilon \rightarrow 0$. Thus

$$\begin{aligned} \lim_{\epsilon \rightarrow 0} \frac{\alpha(\epsilon)}{E_{\mathbf{o}}\{\beta(\mathbf{o}, \epsilon)\}} &= \lim_{\epsilon \rightarrow 0} \frac{\max_{\|\mathbf{w}\| \leq \epsilon} \left\| \nabla \mathbf{h}(\mathbf{x})\mathbf{w} + \begin{bmatrix} o(\|\mathbf{w}\|_2^2) \\ \vdots \\ o(\|\mathbf{w}\|_2^2) \end{bmatrix} \right\|}{E_{\mathbf{o}} \left\{ \left\| \epsilon \nabla \mathbf{h}(\mathbf{x})\mathbf{o} + \begin{bmatrix} o(\epsilon^2 \|\mathbf{o}\|_2^2) \\ \vdots \\ o(\epsilon^2 \|\mathbf{o}\|_2^2) \end{bmatrix} \right\| \right\}} \\ &= \frac{\sigma_{max}(\nabla \mathbf{h}(\mathbf{x}))}{E_{\mathbf{o}}\{\|\nabla \mathbf{h}(\mathbf{x})\mathbf{o}\|\}}, \end{aligned} \quad (8)$$

where $\sigma_{max}(\nabla \mathbf{h}(\mathbf{x}))$ is the maximal singular value of $\nabla \mathbf{h}(\mathbf{x})$. Here the random vector \mathbf{o} is obtained by first sampling each entry i.i.d. from the standard Gaussian distribution, and then normalizing the magnitude, i.e.,

$$\mathbf{o} = \frac{1}{\|\mathbf{g}\|} \mathbf{g}, \quad (9)$$

where all entries of $\mathbf{g} \in \mathbb{R}^N$ are i.i.d. distributed according to the standard Gaussian distribution.

We first consider the deterministic $\nabla \mathbf{h}(\mathbf{x})$. Let the SVD of $\nabla \mathbf{h}(\mathbf{x})$ be $\nabla \mathbf{h}(\mathbf{x}) = U \Sigma V^*$ where $U \in \mathbb{R}^{M \times M}$, $\Sigma \in \mathbb{R}^{M \times M}$, and $V \in \mathbb{R}^{N \times M}$. Then from the convexity of x^2 and Jensen's inequality, we have

$$\begin{aligned} E_{\mathbf{o}}\{\|\nabla \mathbf{h}(\mathbf{x})\mathbf{o}\|\} &\leq \sqrt{E_{\mathbf{o}}\{\|\nabla \mathbf{h}(\mathbf{x})\mathbf{o}\|^2\}} \\ &= \sqrt{E_{\mathbf{o}}\{\|U \Sigma V^* \mathbf{o}\|^2\}} \\ &= \sqrt{E_{\mathbf{o}}\{\|\Sigma V^* \mathbf{o}\|^2\}} \\ &\leq \sqrt{\sigma_{max}^2(\nabla \mathbf{h}(\mathbf{x})) E_{\mathbf{o}}\{\|V^* \mathbf{o}\|^2\}} \\ &= \sigma_{max}(\nabla \mathbf{h}(\mathbf{x})) \sqrt{E_{\mathbf{o}} \left\{ \frac{\sum_{i=1}^M (\mathbf{g}'_i)^2}{\sum_{j=1}^N (\mathbf{g}_j)^2} \right\}} \\ &= \sigma_{max}(\nabla \mathbf{h}(\mathbf{x})) \sqrt{\frac{M}{N}}, \end{aligned}$$

where $\mathbf{g}' = [\mathbf{g}'_1 \ \mathbf{g}'_2 \ \dots \ \mathbf{g}'_N]^T$ is a Gaussian random vector after rotating the Gaussian random vector \mathbf{g} by V^* .

Actually, each element of $V^* \mathbf{o}$ is $\frac{1}{\|\mathbf{g}\|_2} \sum_{k=1}^N (V_{ki})^* \mathbf{g}_k$ which is a standard Gaussian random variable where $(V_{ki})^*$ is the complex conjugate of V_{ki} . We have $\mathbf{g}'_i = \sum_{k=1}^N \bar{V}_{ki} \mathbf{g}_k$. We

can find a matrix $Q \in \mathbb{R}^{N \times (N-M)}$ such that $[V \ Q] \in \mathbb{R}^{N \times N}$ is unitary. When $[V \ Q]^*$ acts on a standard Gaussian random vector g , we will get

$$\mathbf{g}' := [V \ Q]^* \mathbf{g} = \begin{bmatrix} \mathbf{g}'_1 \\ \mathbf{g}'_2 \\ \vdots \\ \mathbf{g}'_N \end{bmatrix},$$

and

$$\|\mathbf{g}\| = \|\mathbf{g}'\|.$$

Then

$$E_{\mathbf{o}} \left\{ \frac{\sum_{i=1}^M (\mathbf{g}'_i)^2}{\sum_{j=1}^N (\mathbf{g}_j)^2} \right\} = E_{\mathbf{o}} \left\{ \frac{\sum_{i=1}^M (\mathbf{g}'_i)^2}{\sum_{j=1}^N (\mathbf{g}'_j)^2} \right\} = \sum_{i=1}^M E_{\mathbf{o}}\{r_i\},$$

where $r_i = \frac{(\mathbf{g}'_i)^2}{\sum_{j=1}^N (\mathbf{g}'_j)^2}$.

Since $\sum_{i=1}^N E_{\mathbf{o}}\{r_i\} = 1$, then from the symmetry of g'_i , we have

$$E_{\mathbf{o}}\{r_i\} = \frac{1}{N}.$$

This gives

$$E_{\mathbf{o}} \left\{ \frac{\sum_{i=1}^M (\mathbf{g}'_i)^2}{\sum_{j=1}^N (\mathbf{g}_j)^2} \right\} = \frac{M}{N}.$$

Thus, combining (8), we get

$$\lim_{\epsilon \rightarrow 0} \frac{\alpha(\epsilon)}{E_{\mathbf{o}}\{\beta(\mathbf{o}, \epsilon)\}} \geq \sqrt{\frac{N}{M}}.$$

We now consider the case where the entries of $\nabla \mathbf{h}(\mathbf{x}) \in \mathbb{R}^{M \times N}$ are i.i.d. distributed according to standard Gaussian distribution $\mathcal{N}(0, 1)$. From Lemma 2, we have with high probability that for $\delta > 0$

$$\sigma_{max}(\nabla \mathbf{h}(\mathbf{x})) \geq (1 - \delta)(\sqrt{N} + \sqrt{M}). \quad (10)$$

Since the Gaussian random vector is rotationally invariant, without loss of generality, we take the \mathbf{o} as

$$\mathbf{o} = [1 \ 0 \ \dots \ 0]^T \in \mathbb{R}^N. \quad (11)$$

Then the $\nabla \mathbf{h}(\mathbf{x})$ is a vector with all entries being i.i.d. distributed according to standard Gaussian distribution. From the convexity of x^2 and Jensen's inequality, we have

$$E_{\nabla \mathbf{h}(\mathbf{x})}\{\|\nabla \mathbf{h}(\mathbf{x})\mathbf{o}\|\} \leq \sqrt{E_{\nabla \mathbf{h}(\mathbf{x})}\{\|\nabla \mathbf{h}(\mathbf{x})\mathbf{o}\|^2\}} = \sqrt{M}.$$

For a norm function $f(x) = \|x\| : \mathbb{R}^n \rightarrow \mathbb{R}$, since

$$|f(x) - f(y)| = \left| \|x\| - \|y\| \right| \leq \|x - y\|,$$

then the function $f(x)$ is Lipschitz continuous with Lipschitz constant 1. Then for $\|\nabla \mathbf{h}(\mathbf{x})\mathbf{o}\|$, we have for every $t \geq 0$,

$$\mathbb{P}(\|\nabla \mathbf{h}(\mathbf{x})\mathbf{o}\| - E_{\nabla \mathbf{h}(\mathbf{x})}\{\|\nabla \mathbf{h}(\mathbf{x})\mathbf{o}\|\} \geq t) \leq 2e^{-\frac{t^2}{2}}.$$

This means the $\|\nabla \mathbf{h}(\mathbf{x})\mathbf{o}\|$ is concentrated at $E_{\nabla \mathbf{h}(\mathbf{x})}\{\|\nabla \mathbf{h}(\mathbf{x})\mathbf{o}\|\}$ which is less than \sqrt{M} .

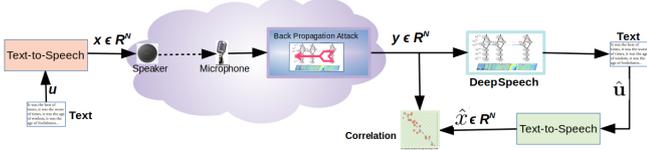


Fig. 3. System Modules of Using Correlation Coefficients to Detect Adversarial Attacks

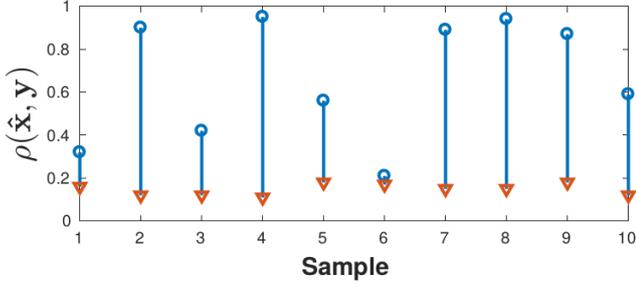


Fig. 4. The change of cross correlation coefficients $\rho(\hat{x}, y)$. Blue circles indicate ρ between input signals y_1 without adversarial attack and their corresponding reconstructed signals \hat{x} from decoded labels, and red triangles indicate ρ between input signals y_2 with adversarial attacks and its corresponding reconstructed signals \hat{x} . Low “blue circles” mean DeepSpeech runs into recognition failure in several error characters, even if no adversarial attacks are present.

Thus, combining (8) and (10), we have with high probability

$$\begin{aligned} \lim_{\epsilon \rightarrow 0} \frac{\alpha(\epsilon)}{E_{\mathbf{o}}\{\|\nabla \mathbf{h}(\mathbf{x})\|\}} &\geq (1 - \delta) \frac{\sqrt{N} + \sqrt{M}}{\sqrt{M}} \\ &\geq (1 - \delta) \frac{\sqrt{M + N}}{\sqrt{N}} \\ &= (1 - \delta) \sqrt{\frac{M + N}{M}}. \end{aligned}$$

IV. EXPERIMENTAL RESULTS

We performed a series of experiments³ to test and illustrate our proposed defense for a popular voice recognition AI classifier DeepSpeech⁴. The experimental setup is illustrated in Fig. 3; a visual comparison with the abstract model in Fig. 1 shows how the various functional blocks are implemented in our experiment.

The experiment consisted of choosing sentences randomly from the classic 19-th century novel “A Tale of Two Cities.” A Linux text-to-speech (T2S) software, Pico2wave, converted a chosen sentence e.g. \mathbf{u}_1 into a female voice wave file. The use of a T2S system for generating the source audio signal (instead of human-spoken audio) effectively allows us to hold the all “irrelevant” variables \mathbf{v} constant, and thus renders the signal synthesis block in Fig. 1 as a deterministic function of just the input label \mathbf{u}_1 .

Let $x[n]$ denote the samples of this source audio signal. This audio signal is played over a PC speaker and recorded by a USB microphone on another PC. Let $y_1[n]$ denote the samples of this recorded wave file. The audio playback and recording was performed in a quiet room with no audible echoes or distortions, so this “channel” can be approximately modeled as a simple AWGN channel: $y_1[n] = \alpha x[n] + w_1[n]$, where α is a scalar representing audio signal attenuation and $w_1[n]$ is random background noise. In our experiment, the $\text{SNR} = \frac{\alpha^2 \|x\|^2}{\|w_1\|^2}$ was approximately 28 dB.

We input $y_1[n]$ into a voice recognition system, specifically, the Mozilla implementation DeepSpeech V0.1.1 based on TensorFlow. The 10 detailed sentences are demonstrated in the table of our full paper [14]. We then used Nicholas Carlini’s adversarial attack Python script⁵ with Deep Speech (V0.1.1) through gradient back-propagation to generate a targeted adversarial audio signal $y_2[n] = y_1[n] + w_2[n]$ where $w_2[n]$ is a small adversarial perturbation that causes the DeepSpeech voice recognition system to predict a completely different sentence \mathbf{u}_2 . Thus, we have a “clean” audio signal $y_1[n]$, and a “targeted corrupted” adversarial audio signal $y_2[n]$ that upon playback is effectively indistinguishable from $y_1[n]$, but successfully fools DeepSpeech into outputting a different target sentence. In our experiment, the power of $y_2[n]$ over the adversarial perturbation w_2 was approximately 35 dB.

We then implemented a version of our proposed defense to detect whether the output of the DeepSpeech is wrong, whether due to noises or adversarial attacks. For this purpose, we fed the decoded text output of the DeepSpeech system into the same T2S software Pico2Wave, to generate a reconstructed female voice wave file, denoted by $\hat{x}[n]$. We then performed a simple cross-correlation of a portion of the reconstructed signal (representing approximately 10% reconstruction of the original number of samples in $x[n]$) with the input signal $y[n]$ to the DeepSpeech classifier: $\rho_{max}(\hat{x}, y) = \max_m |\sum_n \hat{x}[n]y[n - m]|$. If ρ_{max} is smaller than a threshold (0.4), we declare that the speech recognition classification is wrong. The logic behind this test is as follows. When the input signal is $y_1[n]$ i.e. the non-adversarial-perturbed signal, the DeepSpeech successfully outputs the correct label $\hat{\mathbf{u}} \equiv \mathbf{u}_1$, which results in $\hat{x}[n] \equiv x[n]$. Since $y_1[n]$ is just a noisy version of $x[n]$, it will be highly correlated with $\hat{x}[n]$. On the other hand, for the adversarial-perturbed input $y_2[n]$, the reconstructed signal $\hat{x}[n]$ is completely different from $x[n]$ and therefore can be expected to be practically uncorrelated with $y_2[n]$.

Fig. 4 shows the cross-correlation $\rho_{max}(\hat{x}, y)$ for 10 sets of recorded signals (a) with and (b) without adversarial perturbations (red triangles and blue circles respectively in Fig. 4). The adversarial perturbations all successfully fool the DeepSpeech AI to output the target text $\mathbf{u}_2 =$ “he travels the fastest who travels alone”. We see that the observed correlations for the adversarial signals are always very small, and are therefore successfully detected by our correlation test. Interestingly,

³<https://github.com/Hui-Xie/AdversarialDefense>

⁴<https://github.com/mozilla/DeepSpeech>

⁵<https://nicholas.carlini.com/code>

some of the non-adversarial signals yield low correlations as well, but this is because the DeepSpeech cannot decode perfectly even when there are no adversarial attacks present.

REFERENCES

- [1] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: a survey," *arXiv:1801.00553*, 2018.
- [2] I. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv:1412.6572 [cs, stat]*, Dec. 2014, arXiv: 1412.6572.
- [3] A. Fawzi, S. Moosavi-Dezfooli, and P. Frossard, "Robustness of classifiers: from adversarial to random noise," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, ser. NIPS'16. USA: Curran Associates Inc., 2016, pp. 1632–1640.
- [4] T. Tanay and L. Griffin, "A boundary tilting perspective on the phenomenon of adversarial examples," *arXiv:1608.07690 [cs, stat]*, Aug. 2016, arXiv: 1608.07690.
- [5] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. Celik, and A. Swami, "Practical black-box attacks against machine learning," in *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, ser. ASIA CCS '17. New York, NY, USA: ACM, 2017, pp. 506–519.
- [6] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel, "Ensemble adversarial training: attacks and defenses," *arXiv:1705.07204 [cs, stat]*, May 2017, arXiv: 1705.07204.
- [7] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a defense to adversarial perturbations against deep neural networks," in *2016 IEEE Symposium on Security and Privacy (SP)*, May 2016, pp. 582–597.
- [8] N. Carlini and D. Wagner, "Defensive distillation is not robust to adversarial examples," *arXiv preprint arXiv:1607.04311*, 2016.
- [9] Y. LeCun, L. eon Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *PROC. OF THE IEEE*, p. 1, 1998.
- [10] R. Shwartz-Ziv and N. Tishby, "Opening the black box of deep neural networks via information," *arXiv:1703.00810 [cs]*, Mar. 2017, arXiv: 1703.00810.
- [11] J. Kundu, M. Gor, and R. Babu, "BiHMP-GAN: bidirectional 3d human motion prediction GAN," *arXiv:1812.02591 [cs]*, Dec. 2018, arXiv: 1812.02591.
- [12] N. Frosst, S. Sabour, and G. Hinton, "DARCCC: detecting adversaries by reconstruction from class conditional capsules," *arXiv:1811.06969 [cs, stat]*, Nov. 2018, arXiv: 1811.06969.
- [13] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.
- [14] H. Xie, J. Yi, W. Xu, and R. Mudumbai. An information-theoretic explanation for the adversarial fragility of ai classifiers. [Online]. Available: http://user.engineering.uiowa.edu/~rmudumbai/2019_ISIT_extended.pdf