ATHENA: A Framework based on Diverse Weak Defenses for Building Adversarial Defense

Ying Meng, Jianhai Su, Jason M. O'Kane, Pooyan Jamshidi
Departmetn of Computer Science and Engineering
University of South Carolina
Columbia, SC, USA

Abstract—There has been extensive research on developing defense techniques against adversarial attacks; however, they have been mainly designed for specific model families or application domains, therefore, they cannot be easily extended. Based on the design philosophy of ensemble of diverse weak defenses, we propose ATHENA—a flexible and extensible framework for building generic yet effective defenses against adversarial attacks. We have conducted a comprehensive empirical study to evaluate several realizations of ATHENA with four threat models including zero-knowledge, black-box, gray-box, and white-box. We also explain (i) why diversity matters, (ii) the generality of the defense framework, and (iii) the overhead costs incurred by ATHENA.

I. INTRODUCTION

To date, machine learning systems continue to be highly susceptible to *adversarial examples* (also known as *wild patterns* [4]), notwithstanding their attainment of state-of-the-art performance across a wide variety of domains, including speech recognition [65], object detection [24], and image classification [24]. Adversarial examples (AEs) are typically crafted by adding small, human-imperceptible perturbations to benign samples in order to induce machine learning systems to make erroneous predictions [7], [16], [17], [19], [26], [59].

There has been extensive research on developing defense techniques against adversarial attacks; however, they have been mainly designed for specific model families, therefore, they cannot be easily extended to new domains. In addition, the current defense techniques in almost all cases assume specific known attack(s) and are tested in weak threat models [60]. As a result, others have shown that these defenses can be circumvented under slightly different conditions, either a stronger adaptive adversary or in some cases even weaker (but different) adversaries [5], [6], [25], [60]. The "arms race" between the attacks and defenses leads us to pose this central question:

> How can we, instead, design a defense, not as a technique, but as a framework that one can construct a specific defense considering the niche tradeoff space of robustness one may want to achieve as well as the cost one is willing to pay to achieve that level of robustness?

To address this question, we propose ATHENA[1]—an *extensible framework* for building *generic* (and thus, broadly applicable) yet effective defenses against adversarial attacks. The design philosophy behind ATHENA is based on ensemble of many *diverse weak defenses* (WDs), where each WD,
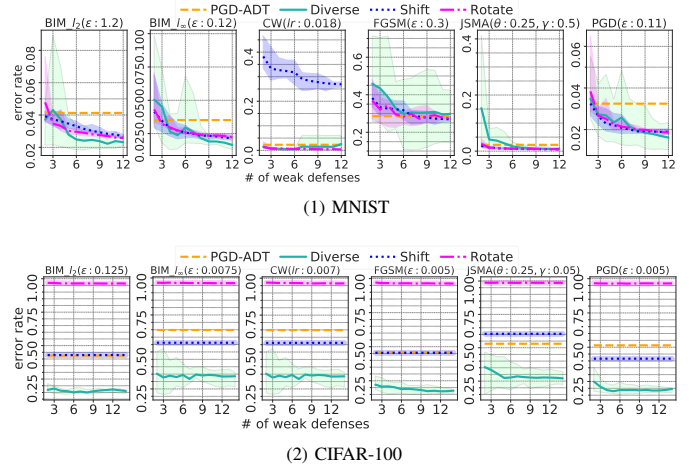

(1) MNIST

(2) CIFAR-100

Fig. 1: A diverse ensemble defense can be robust against adversarial attacks: The error rate decrease as the diversity and the number of weak defenses increases in the ensemble defense.

the building blocks of the framework, is a machine learning classifier (e.g., DNN, SVM) that first applies a transformation[2] on the original input and then produces an output for the transformed input. Given an input, an ensemble first collects predicted outputs from all of the WDs and then determines the final output, using some *ensemble strategy* such as majority voting or averaging the predicted outputs from the WDs. Note that input transformations have been extensively used for defending machine learning systems from adversarial attacks [1], [3], [12], [15], [20], [22], [38], [43], where a singular transformation such as denoise or rotation was used. In this work, rather than looking for a peculiar transformation that effectively mitigates adversarial perturbations, we propose a framework for building an ensemble defense on top of a large population of diverse transformations of various types. We show that ensembling diverse transformations can result in a robust defense against a variety of attacks and provide a *tradeoff space*, where one can add more transformations to the ensemble to achieve a *higher robustness* or decrease the weak defenses to achieve *lower overhead and cost*.

To motivate the idea, we first evaluate how the number of WDs affects the ensemble's robustness. We used a variety of random WDs and the output averaging strategy as the ensemble strategy, which determines the label by averaging the outputs from WDs, then tested the ensembles against various

---

[1]Goddess of defense in Greek mythology.

[2]We, therefore, use *weak defense* and *transformation* interchangeably.

attacks. To show that an ensemble is more robust by utilizing a diverse set of transformations, we compared three versions of ensemble-based defenses: (i) ensembles that were created from a diverse set of transformations (referred to as "Diverse"), and two non-diverse homogeneous ensembles that were created based on a variations of two particular transformations, i.e., (ii) translations (referred to as "Shift"), and (iii) rotations (referred to as "Rotate"). Average error rates and variations of 5 trials of the three versions of the ensembles are presented in Figure 1.

By disturbing its input, a transformation changes the adversarial optimized perturbations and therefore making the perturbations less effective. By changing the input in different ways, a group of transformations complement each other in blocking adversarial perturbations introduced by different attacks. Thus, with more WDs, an ensemble is likely to be more robust against adversarial attacks. As presented in Figure 1, for most cases, the error rate drops as an ensemble utilizes more and more WDs. When constructed using 4 or more WDs, all ensembles achieved competitive performance or even outperformed PGD adversarial training (PGD-ADT) [40], a state-of-the-art defense approach.

The effectiveness of homogeneous ensembles with a single transformation type varies on attacks and datasets. Shifts and rotations perform well in filtering all adversarial perturbations except CW in MNIST. However, the ensembles based on rotations are not effective on CIFAR-100. Compared to rotations, ensembles based on shift are more effective in filtering CW perturbations on CIFAR-100, but less effective on MNIST. A collection of diverse transformations, however, mitigates the perturbations in different ways, adjusting angles or positions, denoising, compressing, and so on. When some transformations may fail to disentangle perturbations by a particular adversary, others may still mitigate the perturbations. We can observe that "Diverse" achieves the lowest error rate in most cases, especially for tasks on CIFAR-100, where "Diverse" outperforms all, achieving an error rate that is at least 20% lower than the second-best performers. Providing more freedom in the solution space by varying the transformation types and parameters, the diverse ensembles give broader variations than the homogeneous ensembles. This suggests the insight that "diversity matters". In addition, the results indicate that we could have found a much more robust ensemble from the diverse library if we have designed and selected the WDs very carefully. Besides, by adding more WDs to the ensemble, the error rate of the worst ensemble decreases, showing potentials for worst case scenarios in dynamic environments where the behavior of the attacker may change.

The observations reflected by our results suggest that it is possible to construct an ensemble with many diverse WDs, whereby the model becomes increasingly more robust to many forms of adversarial attacks. Our objective in this work is to construct an efficient defense without any assumption regarding the adversarial attack. That is, building a defense that, in general, is robust against any adversary.

Overall, we make the following contributions:
- We studied the defensive effectiveness of 72 transformations

for MNIST and 22 for CIFAR-100. To the best of our knowledge, this is the first work studying such a large variety of transformations.
- We proposed ATHENA as a framework for building ensemble defenses that is flexible to scale by simply adding/removing WDs and to change the ensemble strategy.
- We evaluated ATHENA via 4 threat models for image classification on MNIST and CIFAR-100:
  - *Zero-knowledge*: For MNIST, we realized and evaluated 5 ensembles against 9 attacks (with 5 variations of strengths). We also compared our results with 2 adversarial defenses from previous studies—PGD adversarial training and randomized smoothing. Across all attacks, at least one of the 5 ensembles achieved the lowest error rate. The improvements, compared to the undefended model (UM), ranged from 8.03% to 96.17%. We performed the same experiments on CIFAR-100, but due to the computational cost, in a smaller scale, where 4 ensembles were tested against 6 attacks. In most cases, ATHENA outperformed the previous adversarial defenses. The improvements ranged from 19.61% to 90.29% against other defenses (Section IV-D).
  - *Black-box*: We evaluated with transfer-based and query-based black-box attacks on CIFAR-100, and observed the effectiveness of ATHENA in protecting the UM under both attacks. Against the transfer-based attack, ATHENA reduces the transferability rate of generated AEs by 25.82% on average. When facing the query-based attack, ATHENA significantly restricts the attacker's capability to minimize the distance between generated AEs and benign samples. Given a query budget, the averaged distance between ATHENA-targeted AEs and benign samples is 0.35x to 15.45x larger than that between the UM-targeted AEs and benign samples (Section IV-E).
  - *Gray-box and White-box*: We conducted an evaluation of gray-box and white-box attacks on an ensemble based on majority voting among predicted labels of WDs. We generated AEs using FGSM attack. 10 variants were crafted using different constraints on the maximum dissimilarity of AEs. The error rate of ATHENA is 10% to 20% lower than that of the UM except the weakest set where our approach is equally ineffective as the UM. However, it is computationally costly to craft effective AEs (5x more time to craft an AE on average), and the generated AEs are much more prone to be detected (Section IV-F).
- We also performed a comprehensive empirical study to understand *why ensembles of many WDs work* and *why the diversity of WDs matters*.
- We released the source code and experimental data: https://github.com/softsys4ai/athena

II. BACKGROUND AND DEFINITIONS

A. Notation

In this paper, we refer to the legitimate data set as $\mathcal{D} \subset \mathbb{R}^d$; a transformation operation as t_i $(i = 1, 2, 3, \ldots)$; and $t_i(\mathbf{x})$

as the output from applying transformation t_i on the input $\mathbf{x}$. The composition of n transformations is also a transformation: $t_g = t_n \circ t_{n-1} \circ \cdots \circ t_1$. Therefore, we use t_i to denote a transformation or a composition of transformations. $\mathcal{D}_{t_i} = \{\mathbf{x}_{t_i} = t_i(\mathbf{x}) | \mathbf{x} \in \mathcal{D}\}$ denotes the set of transformed examples of $\mathcal{D}$ associated to t_i. That is, $\mathcal{D}_{t_i} = t_i(\mathcal{D})$.

We focus our work on *supervised machine learning*, where a classifier is trained on labeled data in $\mathcal{D}$. Here, a classifier is a function $f(\cdot)$ that takes a data point $\mathbf{x} \in \mathbb{R}^d$ as input and produces a vector $\mathbf{y}$ of probabilities of all the classes in $\mathcal{C}$. Given a target model $f(\cdot)$ and an input $\mathbf{x}$ with ground truth y_{true}, an adversary attempts to produce an AE $\mathbf{x}'$ (limited in l_p-norms), such that $\text{argmax}(f(\mathbf{x}')) \neq y_{\text{true}}$, by solving an optimization to maximize the loss function:

$$\max_\delta (\mathcal{L}(f, \mathbf{x} + \delta, y_{\text{true}})),$$
$$\text{s.t. } ||\delta||_p \leq \epsilon \text{ and } \mathbf{x}' = \mathbf{x} + \delta \in [0,1]^D, \tag{1}$$

where $\text{argmax}(f(\mathbf{x}))$ is the predicted output given an input $\mathbf{x}$, $\mathcal{L}$ is the loss function, δ is the perturbation, and ϵ is the magnitude of the perturbation. To remain undetected, the adversarial example $\mathbf{x}'$ should be as *similar* to the *benign sample* (BS) as possible; therefore, attacks (e.g., FGSM [19], PGD [40], and CW [7]) use different norms (such as l_0, l_2, or l_∞) to constrain the distance between $\mathbf{x}$ and $\mathbf{x}'$.

When crafting an adversarial example for an input, some attackers force the target model to produce a specific output label, $t \in \mathcal{C}$, for a given input, while other attackers only seek to produce an output label that does not equal the ground truth. The former is referred to as a *targeted*, while the latter is referred to as an *untargeted* or *non-targeted attack*.

Given a set of N input data $\{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ and a target classifier $f(\cdot)$, an *adversarial attack* aims to generate $\{\mathbf{x}'_1, \dots, \mathbf{x}'_N\}$, such that each $\mathbf{x}'_n$ is an adversarial example for $\mathbf{x}_n$. The *success rate* of an attack is measured by the proportion of predictions that were successfully altered by an attack: $\frac{1}{N}\sum_{n=1}^N \mathbb{1}[f(\mathbf{x}_n) \neq f(\mathbf{x}'_n)]$. The magnitude of the perturbations performed by the attack, in this work, is measured by *normalized l_2-dissimilarity*, $\frac{1}{N}\sum_{n=1}^N \frac{\|\mathbf{x}_n - \mathbf{x}'_n\|_2}{\|\mathbf{x}_n\|_2}$. A strong adversarial attack has a high success rate while its normalized l_2 dissimilarity is low.

A *defense* is a method that aims to make the prediction on an adversarial example equal to the prediction on the corresponding benign sample, i.e., $\text{argmax}(f(\mathbf{x}')) = y_{\text{true}}$. In this work, our defense mechanism is based on the idea of *many diverse weak defenses*, which are essentially *transformation-based defenses*, i.e., they produce output label via $f_{t_i}(t_i(\mathbf{x}'))$. Typically, $t(\cdot)$ is a complex and non-differentiable function, which makes it difficult for an adversary to attack the machine learning model $f(t(\mathbf{x}))$, even when the attacker knows both $f(\cdot)$ and $t(\cdot)$.

A model that was trained using data set $\mathcal{D}_{t_i}(i = 1, 2, \dots)$ is referred to as a *weak defense* and denoted as f_{t_i}. Each of these trained models are collectively used to construct defense ensembles. On the other side, the original model f that is trained on $\mathcal{D}$ is referred to as the *undefended model*.

B. Adversarial Attack

Gradient-based Attacks perturb their input with the gradient of the loss with respect to the input. Some attacks in this family perturb the input only in one iteration. For example, FGSM processes an adversarial example as follows:

$$\mathbf{x}' = \mathbf{x} + \epsilon \cdot \text{sign}(\bigtriangledown_\mathbf{x} J(\mathbf{x}, \mathbf{y})), \tag{2}$$

where J is the cost function of the target model f, $\bigtriangledown_\mathbf{x}$ is the gradient with respect to the input $\mathbf{x}$ with corresponding true output $\mathbf{y}$, and ϵ is the magnitude of the perturbation.

Other variations, like BIM [32], MIM [14], and PGD, are iterative and gradually increase the magnitude until the input is misclassified. For example, BIM, an extension of FGSM, rather than taking one big jump ϵ, takes multiple smaller steps $\alpha < \epsilon$ with the result clipped by ϵ. Specifically, BIM begins with $\mathbf{x}'_0 = \mathbf{x}$, and at each iteration it performs:

$$\mathbf{x}'_i = \text{clip}_{\mathbf{x},\epsilon}\{\mathbf{x}'_{i-1} - (\alpha \cdot \text{sign}(\bigtriangledown_\mathbf{x} J(\mathbf{x}'_{i-1}, \mathbf{y})))\}, \tag{3}$$

where $\text{clip}_{\mathbf{x},\epsilon}(\mathbf{A})$ denotes the element-wise clipping of $\mathbf{x}$; the range of $\mathbf{A}$ after clipping will be $[\mathbf{x} - \epsilon, \mathbf{x} + \epsilon]$.

JSMA [52], another gradient-based approach, greedily finds the most sensitive direction, such that changing its values will significantly increase the likelihood of a target model labeling the input as the target class:

$$s_t = \frac{\delta t}{\delta x_i}; s_o = \Sigma_{j \neq t} \frac{\delta j}{\delta x_i};$$
$$s(x_i) = s_t |s_o| \cdot (s_t < 0) \cdot (s_o > 0), \tag{4}$$

where s_t represents the Jacobian of target class $t \in \mathcal{C}$ with respect to the input image, and where s_o is the sum of Jacobian values of all non-target classes.

CW constraints adversarial samples to stay within a certain distance from the benign sample. DEEPFOOL [44] takes iterative steps to the direction of the gradient of a linear approximation of the target model. Both attacks generate adversarial examples by solving an optimization problem of the form:

$$\text{argmin}(d(s, x + \delta) + c \cdot \mathcal{L}(x + \delta)), \tag{5}$$

where $\mathcal{L}$ is the loss function for solving $f(x + \delta) = t$ and $t \in \mathcal{C}$ is the target label.

Black-box Attacks do not require any knowledge of target models, yet are shown to be effective in fooling machine learning models. ONE-PIXEL [58], one of the extreme adversarial attack methods, generates adversarial examples using an evolutionary algorithm called Differential Evolution [57].

III. ATHENA: A FRAMEWORK OF MANY WDS

In this section, we present our defense framework, called ATHENA, which is based on *ensembles* of *many diverse weak defenses*. With ATHENA one can construct effective defenses that are resilient to adversarial attacks. Further, such defenses are not tied to a particular ML classifier and can be deployed in different domains beyond image classification.

(1) Training a weak defense
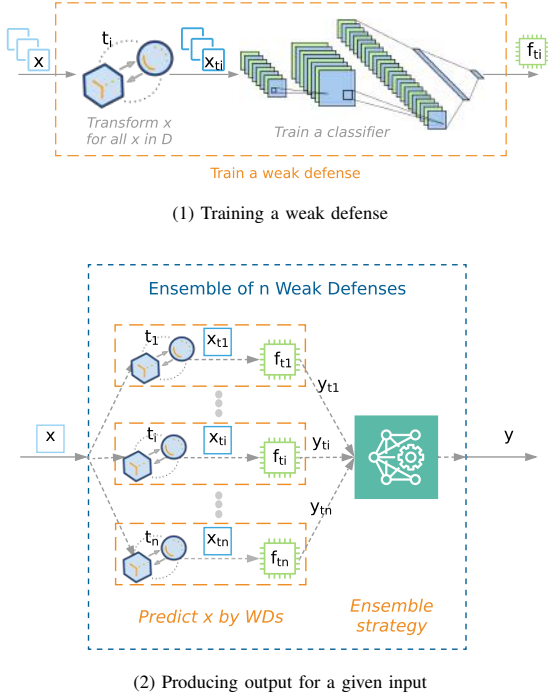

(2) Producing output for a given input

Fig. 2: ATHENA, a framework for building adversarial defense: (1) Training a weak defense on transformed inputs. (2) Predicting the output label using predictions of all weak defenses.
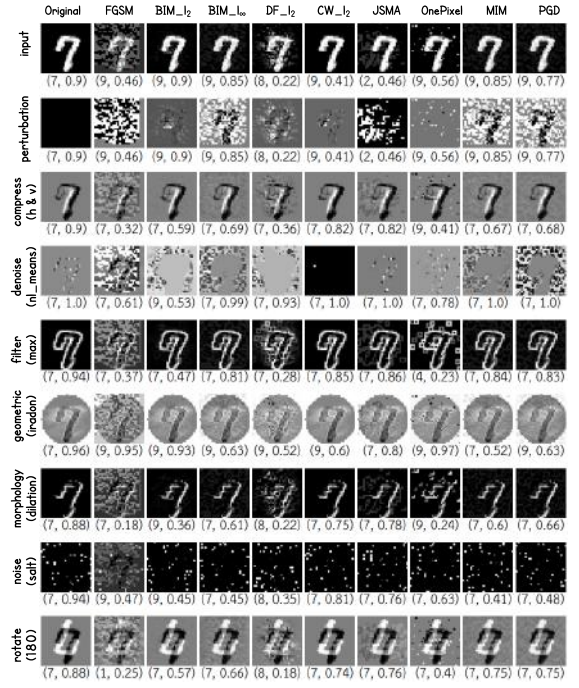

Fig. 3: Sample adversarial images generated by various adversaries and their corresponding distortions created by transformations in each row. (i) In the 1st row, we present inputs—an original input followed by the adversarial examples with predicted label and probability produced by the UM $f(\cdot)$. (ii) In the 2nd row, we present the perturbation generated by the adversary corresponding to the column. (iii) In each cell in rows 3–9, we present the distortion generated by applying the transformation in a current row to the input in the column with the predicted label and probability produced by the weak defense associated to the corresponding transformation.

ATHENA works as follows: At test time, for a given input data $\mathbf{x}$, (i) it first collects outputs from all WDs, and then (ii) it uses an ensemble strategy (e.g., majority voting) to compute the final output (Figure 2 (2)). Each WD can be any ML model (e.g., a CNN) trained separately (Figure 2 (1)).

Our proposed defense framework is:

- **Extensible**: One can *add* new WDs and, therefore, improve its effectiveness or can *remove* WDs, which will sacrifice effectiveness but will also decrease the overhead.
- **Flexible**: One can *update* the ensemble, making it robust by replacing any (i) WDs and/or (ii) ensemble strategy.
- **General**: One can use different *models* to train WDs.

A. Transformation as a Weak Defense

Transformations have shown to be effective in deterring some adversarial perturbations [3], [22], [56]. To provide a more comprehensive understanding of the effectiveness of transformations, we examined a large variety of transformations (see Table V) against AEs generated by various attacks. Figure 3 presents sample inputs from MNIST in the 1st row—an benign sample followed by AEs generated by the delineated adversaries that correspond to each column; the corresponding adversarial perturbations, $\|\mathbf{x} - \mathbf{x}'\|$, are shown in the 2nd row; and the distortions, $\|\mathbf{x}' - t_i(\mathbf{x}')\|$, generated by some sample transformations are shown in the 3rd–9th rows. Overall, the following observations have been made:

- *Each WD is able to correctly classify some of the AEs*. For example, the WD associated to morphology(dilation) can correctly classify samples generated by FGSM, BIM_l_∞, CW_l_2, JSMA, MIM, and PGD.

- *WDs complement each other*. For example, the AE generated by FGSM is misclassified by rotate($180°$); however, it is correctly classified by morphology(dilation).

B. Ensemble of Many Diverse Weak Defenses

As observed in Figure 3, the effectiveness of transformations varies according to the type of adversary. It is difficult to find a small set of transformations (i.e., 2–3) that are effective against all adversaries. Nevertheless, the observations from Figure 1 (and Figure 30) suggest that WDs can complement each other by increasing the quantity and diversity of WDs.

We train a WD f_{t_i} by first applying a transformation, t_i, on the original data $\mathcal{D}$, and then training a classifier using the transformed data set $\mathcal{D}_{t_i}$. At test time, given an input $\mathbf{x}$, we first apply a transformation t_i on the input $\mathbf{x}$ and then feed the transformed input $\mathbf{x}_{t_i}$ to the corresponding WD f_{t_i}. ATHENA uses an ensemble strategy (e.g., majority voting) to compute the final predicted label y by utilizing the predictions (probabilities and/or logits) of WDs. We elaborate on specific ensemble strategies in Section IV-B.

IV. EXPERIMENTAL EVALUATION

A. Experimental Setup

We evaluated ATHENA on MNIST [33] and CIFAR-100 [30] against 9 different attack methods: FGSM, BIM(l_2- and l_∞-norms), CW(l_2-norm), JSMA, PGD(l_∞-norm), DEEPFOOL,

ONE-PIXEL and MIM.[3] For each attack, 5 sets of AEs were generated using various perturbation magnitudes. The attacks were implemented with Google CleverHans [50] for MNIST, and with IBM ART [46] for CIFAR-100.

TABLE I: The architecture and training parameters of the undefended model and weak defenses.

Model	Architecture		Parameters	
LeNet	Conv.ReLU	$3 \times 3 \times 32$	Optimizer	Adam
	Max pooling	2×2	Learning rate	0.001
	Conv.ReLU	$3 \times 3 \times 64$	Batch size	128
	Max pooling	2×2	Epoch	50
	Dense	4096		
	Dropout	0.4		
	Dense	10		
	Softmax	10		
SVM	Kernel	Linear	Regularization parameter	0.1
WResNet	Network	Wide ResNet	Optimizer	SGD
	Depth	28	Learning rate	0.1
	k	10	Batch size	128
			Epoch	200
			LR scheduler	cosine
ResNet-Shake	Network	ResNet	Optimizer	SGD
	Depth	26	Learning rate	0.1
	Residual branches	2	Batch size	256
	1st-residual width	32	Epoch	200
	Regularization	Shake-Shake	LR scheduler	cosine

We built ensemble defenses upon 72 WDs for MNIST and 22 WDs for CIFAR-100 due to the computational cost. Each of the WDs is associated to one transformation (Table V). When training a classifier, we used 80% of the training data as for training and 20% for validation. To show that ATHENA works independent of the type of models, we built and evaluated two versions of ATHENA for each data set—28×2 Wide ResNet (WRN) [54] and 26 $2 \times 32d$ ResNet with Shake-Shake regularization (ResNet-Shake) [18] for CIFAR-100 and LeNet and linear SVMs for MNIST (Table I). We will mainly illustrate the results using realizations with 28×2 WRN for CIFAR-100 and provide results for LeNet MNIST in the appendix.

In this study, we realized 4 ensemble strategies (Section IV-B) and evaluated their effectiveness against 4 forms of threat models (Section IV-C). We also compared ATHENA with two state-of-the-art adversarial defenses: (i) *PGD Adversarial Training* (PGD-ADT) [41], which enhances model's robustness by solving the robust optimization problem $\min_{\theta} \max_{\delta \in \Delta} \mathcal{L}(\mathbf{x} + \delta, y; \theta))$ that performs gradient ascent on the input $\mathbf{x}$ and gradient descent on the weights θ using PGD attack as the optimization algorithm. We implemented PGD-ADT on top of IBM ART. (ii) *Randomized Smoothing* (RS) [11], which first generates n noisy samples by adding random Gaussian noises to the input, $\mathbf{x}$, collects predicted probabilities by feeding generated noisy samples to the model, and then filters out the predictions where the highest probability is not statistically significantly larger than the rest. Finally, RS computes corresponding labels for all valid predictions and determines the most frequent label as the final class of $\mathbf{x}$. We executed RS experiments using scripts from the original work.[4] The experiments (e.g., training WDs, crafting AEs)

---

[3]DEEPFOOL, ONE-PIXEL, and MIM were implemented on MNIST only.
[4]https://github.com/locuslab/smoothing

---

were conducted on multiple machines (Table IV).

B. Ensemble Strategies

We implemented and evaluated 4 ensemble strategies:

- **Random Defense (RD)** uses a random WD to predict the input. Its expected effectiveness equals to the average of all WDs, therefore, we use it as a baseline against which to compare ATHENA.
- **Majority Voting (MV)** determines the label agreed upon by most WDs.
- **Top 2 Majority Voting (T2MV)** collects labels associated to the top two probabilities and, thereafter, performs majority voting among them. As we observed from Figure 3, the predicted probability distributions of some WDs on certain AEs are soft, such that those WDs are less confident with their predictions, whereby the correct answer may lie in the classes for which they are less confident.
- **Average Output (AVEO)** takes an average of the outputs from WDs, and then returns the label with the highest averaged value (i.e., applies argmax on the averaged outputs). For WDs built from neural networks, the output can be selected from any layer. Two specific variants of AVEO are AVEP and AVEL, which use probability and logit as the output respectively.

C. Threat Models

We make two assumptions with respect to threat models: that the attackers can attack only at the inference phase and the attackers can modify only the input data (i.e., the attackers cannot modify the models nor the training data). We evaluate ATHENA with the following threat models [4], [5]:

- **Zero-Knowledge** is the weakest model, where the attacker is not even aware that a defense is in place. However, here we assume that the attacker knows the exact architecture and parameters of the UM. It is important to verify this weak model, because failing this test implies that the stronger tests under adaptive attackers will also fail [5].
- **Black-box** model assumes that the adversary has no knowledge of the ensemble model, but it is aware of the fact that a defense mechanism is protecting the classifier. Here, we assume that the attacker is not aware of the whole training data, it does, however, have access to a portion of training data albeit without the correct labels.
- **White-box** is the strongest threat model, in which the adversary has full knowledge of the ensemble defense model. That is, the attacker knows the architecture and hyper-parameters of UM and all WDs, the list of transformations (or, in the Grey-box threat model, a partial list), and the ensemble strategy being used.

D. Zero-Knowledge Attack

In the context of zero-knowledge threat model, we assume that the attacker has full knowledge about UM and nothing else. Therefore, all adversarial examples were generated based on UM. We evaluated 4 realizations (i.e., RD, MV, T2MV, and AVEL) of ATHENA, consisting of 22 WDs, against 6 attacks
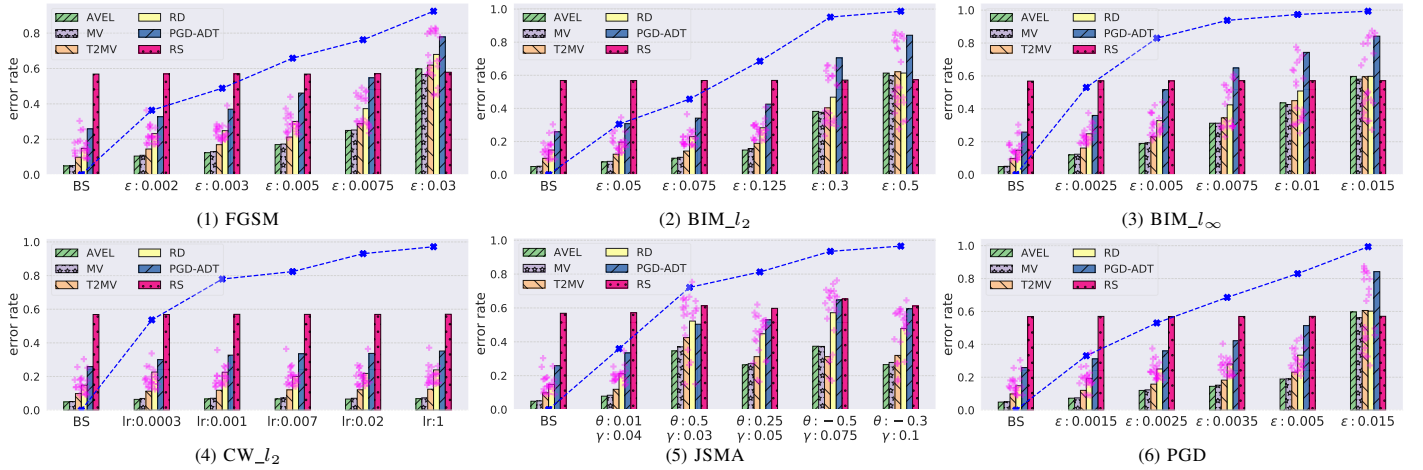
Fig. 4: Evaluation of 28×10 WRN models on CIFAR-100. Models are evaluated with respect to error rate on AEs. For each type of attack, the perturbations in the data set becomes stronger from left to right. The results of UM are plotted as blue dotted lines and each WD as pink "+".

on CIFAR-100. For each attack, we varied their configurations to generate 5 different strength levels (Table II). The error rate of different realization of ATHENA were then compared with that of PGD-ADT and RS. PGD-ADT was run with $\epsilon = 0.015$. Multiple trials were launched to evaluate RS, using various σ values: $0.05, 0.075, 0.1, 0.3$, and 0.5. Only the best results, where $\sigma = 0.075$, were reported. In each attempt, 100 noisy samples were generated for each input.

In this study, evaluation results are reported with respect to model's error rate on adversarial examples whose corresponding benign samples were correctly classified by UM. Therefore, the error rate of UM on BS in the report is 0. Our UM achieves a top-1 test accuracy of 75.54% on CIFAR-100. Figure 4 presents the evaluation results for ATHENA, being built upon 28×10 WRNs. For each type of attack, UM (the blue dotted line) made more errors on stronger AEs (from left to right, cf. l_2 dissimilarities in Figure 5).

The effectiveness of RS is stable across attacks and perturbation magnitudes. The variation in the error rate of RS is very small (8.5% from 56.82% to 65.36%), a byproduct of a certified defense. However, the effectiveness of RS, on BS and AEs with small perturbations, is even worse than UM (an expected tradeoff for certified defenses). In general, PGD-ADT makes fewer errors than RS on AEs with weak perturbations in each group of attacks. While as the perturbations become stronger, the error rates of PGD-ADT increase and eventually go beyond that of RS.

The effectiveness of individual WDs varies on each test set (BS or AE). The variation of the error rates achieved by WDs spans wider as the perturbation magnitude becomes stronger in each group. For example, evaluating on BIM_l_2 AEs, error rates of WDs vary from 13.07% to 34.56% when $\epsilon = 0.05$ and from 27.54% to 85.82% when $\epsilon = 0.5$. The RD ensemble identifies the baseline of ATHENA. As by labeling the input using a random WD each time, RS ensemble is expected to achieve the average error rates of all WDs. Therefore, as long as there are enough number of WDs are efficient, we have a

good chance of building a robust RD ensemble. Furthermore, by applying a more intelligent strategy, like MV, T2MV, or AVEL, ATHENA becomes more resilient against such attacks. For example, on JSMA($\theta = 0.5, \gamma = 0.03$), the baseline made an error rate of 52.25% vs. 50.4% on PGD-ADT, where more than 69% of WDs make more errors than PGD-ADT does. However, by applying argmax on the averaged predictions of all WDs (AVEL), ATHENA achieves an error rate of 34.64%, which is much more efficient than PGD-ADT.

Overall, all of the 4 ensembles are more efficient than PGD-ADT and RS on both BS and AEs. They are the 4 top performers (i.e., make the fewest errors on AEs) across attacks and perturbation magnitudes, except the extreme cases in 4 groups—FGSM ($\epsilon = 0.03$), BIM_l_2 ($\epsilon = 0.5$), BIM_l_∞ ($\epsilon = 0.015$), and PGD ($\epsilon = 0.015$). In those extreme cases, the ensemble defenses achieve an effectiveness close to the top performer (RS or PGD-ADT), the gaps between the best ensemble defense and the top performer is $[2\%, 4\%]$.

A very similar evaluation was performed on MNIST, but in a larger scale: (i) realized 5 ensembles—RD, MV, T2MV, AVEL, and AVEP, (ii) each ensemble consists of 72 WDs, and (iii) 9 attacks have been launched. Our UM achieves a test accuracy of 99.01% on MNIST. As presented in Figure 32 and Figure 17, the MNIST results are similar to that in CIFAR-100. PGD-ADT and RS achieve an error rate lower than 10% across all attacks except on FGSM AEs (where $\epsilon = 0.2, 0.25$, and 0.3) and DEEPFOOL AEs (where os $= 8, 16, 50$). ATHENA with AVEL or T2MV ensemble strategy achieves the lowest error rate, across all attack and perturbation magnitudes, however, with small margin.

E. Black-box Attack

We evaluated ATHENA with two black-box attacks: (i) *transfer-based* [21], [51], (ii) *gradient-direction estimation* approach based on HopSkipJump [8]. We evaluated one realization of ATHENA—AVEP ensemble with 12 WDs.

The transfer-based attack assumes AEs are transferable between models [19], [59], and consist of substitute model
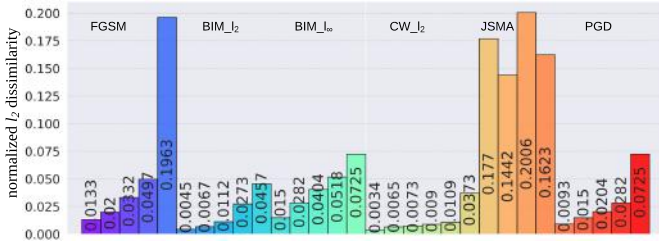
Fig. 5: Normalized l_2 dissimilarity between BS and each AE generated for CIFAR-100.

Fig. 6: Performance of transferability-based black-box attack with different budgets and using BIM_l_∞ with different perturbations to attack each trained substitute model. Note: ER, TR and UM stand for error rate, transferability rate and undefended model respectively.

training and adversarial sample crafting. The attackers construct a substitute model, f_{sub}, mimicking the target model, f_{target}, using only limited training data.[5] The attackers then use the substitute model to create AEs and to launch attacks on the system. This strong assumption challenges the attacker in many ways when crafting an AE, such as selecting a particular model architecture for building the substitute model and preparing a sample set for label collection given a limited query budget [21], [51]. Although we could consider different assumptions for the knowledge of the attacker in black-box attacks, we instead considered the strongest possible black-box attack, where the adversary knows the exact architecture, optimization strategy, and hyper-parameters of the original target model. The attacker can then use the substitute model's parameters and perform a white-box attack on the substitute model. Finally, we test how well these AEs can transfer from the substitute model to the target model by calculating the *transferability rate*, the proportion of these AEs misclassified by the target model. We evaluate the black-box threat model with transferability-based attack as follows:

1) Collect a data set of N samples, $\mathcal{D}_{\text{bb}} = \{(\mathbf{x}, \mathbf{y}) | \mathbf{y} = f_{\text{target}}(\mathbf{x}), \mathbf{x} \in \mathcal{D}\}$, by querying the target model N times.
2) Build a substitute model f_{sub}, mimicking the target model f_{target}, which is trained on the collected data set $\mathcal{D}_{\text{bb}}$.
3) Generate adversarial examples to attack f_{sub}.
4) Attack f_{target} using the AEs crafted based on f_{sub}.

For training substitute models and evaluating the defense, we split the CIFAR-100 test set ($10k$ samples) into two parts: (i) training ($5k$ samples) and testing ($5k$ samples). For a given a budget, B, the set, $\mathcal{D}_{\text{bb}}$, for training the substitute model is composed of B samples selected from the training set, which can be correctly classified by the target model. The class labels of these selected B samples evenly distribute across all 100 classes of CIFAR-100. The adopted budget list is $[1k, 2k, 3k, 4k, 5k]$. That is, in total, we trained 5 substitute models for each target model. The performance of each substitute model was evaluated with the test set of $5k$ samples. The top-1 and top-5 accuracy of the trained substitute models built against UM and defended model with ATHENA are comparable (see Figure 30), so the success of the defense can be attributed to the drop in the transferability rate.

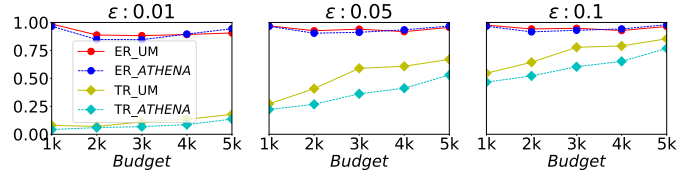[5]It may be flagged suspicious if the attacker queries the system many times.

Once we trained the substitute model associated to each budget, we generate AEs against the substitute model. For each substitute model, we crafted AEs using the 3 adversarial attacks, FGSM, BIM_l_∞ and PGD. For each attack, a list of 5 perturbations was used. For each substitute model, we selected 500 (evenly distributed across all classes) out of the remaining $5k$ test samples for generating AEs.

Figure 6 shows the result of this transfer-based attack using BIM_l_∞ with three perturbations: $[0.01, 0.05, 0.1]$. In all scenarios (all combinations of perturbation and budget here), ATHENA lowers the transferability rate achieved by the attack when compared to UM. As expected, the transferability rate increases when the budget increases. The drop in transferability rate from UM to ATHENA becomes more evident when the perturbation increases, which indicates that ATHENA is less sensitive to the perturbation. In the low perturbation ($\epsilon = 0.01$) scenario, the drop of transferability rate achieved by ATHENA is not significant with an average value of 3.6%, while the averaged drop in the perturbation ($\epsilon = 0.1$) increases to 12%. The reason is that AEs generated with the perturbation $\epsilon = 0.01$ are mostly ineffective with very low transferability rates of averaged values of 11.44% and 7.84% for UM and ATHENA respectively, while AEs generated with the perturbation $\epsilon = 0.05$ are much effective with average transferability rage of 72.24% and 60.24% for UM and ATHENA respectively. The ratio of decreased transferability rate is 25.82% on average. To limit the computation, we only realized ATHENA with 12 WDs, but we expect to observe larger drop in transferability rate with higher number of WDs.

As seen in Figure 31, a similar trend is also observed when using the other two attacks. It is important to emphasize that the results could have been even better for our defense mechanism under under a weaker black-box threat model, where the knowledge of DNN architecture and hyper-parameters is not known by the adversary, such that the attacker confronts greater difficulty in crafting effective AEs. Lastly, ATHENA can employ randomness, like the RD ensemble, resulting into non-deterministic output of the ensemble. This will propagate randomness to the training process of substitute models, and hence result in lower accuracy of the substitute models, and finally limit the attacker's strength.

Gradient-direction estimation approaches are based on search using an estimation of the gradient direction, where the attacker has access to only the output probabilities (score-based) or the label (decision-based). We evaluated ATHENA with the state-of-the-art decision-based attack, Hop-
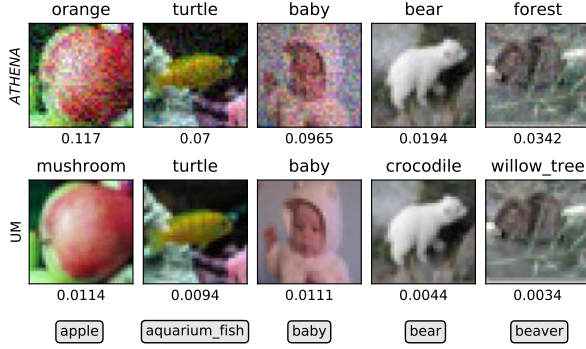
Fig. 7: Adversarial examples (CIFAR-100) generated by HSJA with l_∞ distance and a budget of 5000 queries against the undefended model (UM) and ATHENA. Note: 1) the text above each subfigure is its predicted label by the corresponding target model; 2) the number under each subfigure indicates its l_∞ distance from the original benign sample; 3) true labels are displayed in round rectangles.
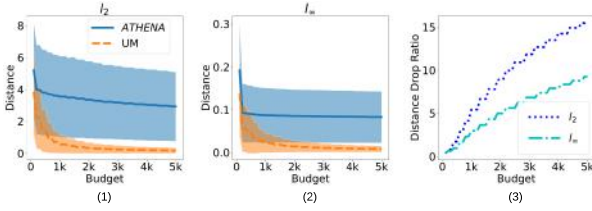

Fig. 8: Mean distance versus query budget and distance drop ratio versus query budget on CIFAR-100 with the undefended model (UM) and ATHENA.

SkipJump Attack (HSJA) [8], which only requires the output labels from the target model and is able to create AEs with much fewer budget (the number of queries to the target model) than existing approaches. It finds an AE that is closest to the corresponding benign sample based on an estimation of the gradient between adversarial group and benign group.

We launched two HSJA variants (l_2 and l_∞) against both UM and ATHENA using the open-sourced code.[6] For each of UM and ATHENA, 500 out of $5k$ CIFAR-100 test samples (the same test set mentioned in the experiment of transfer-based black-box attack) were selected for generating AEs.

An AE created by HSJA with a given budget is considered as invalid if its distance to the benign sample is larger than the given distance threshold. Therefore, a target model is considered to have a better defense capability if the AE generated from it has a larger distance value. Figure 7, shows some example perturbed images. We observe that 1) for each of the benign samples of apple, aquarium_fish and beaver, the crafted AE by HSJA against ATHENA has a much larger distance (7.5x-10x) to the benign sample when compared to the AE against the UM. 2) HSJA is able to utilize the benign example of bear to launch attack against the UM but fails when facing ATHENA, even though 3.4x more perturbation is added. 3) for the benign sample of baby, HSJA fails to attack both target models. An increase of query budget might help here. With l_2 distance, HSJA needs 23.7x more perturbation to attack ATHENA when compared to the UM (see Figure 22).

[6]HopSkipJumpAttack: github.com/Jianbo-Lab/HSJA

To further demonstrate the effectiveness of ATHENA, we explored the averaged distances of AEs versus budget for both UM and ATHENA. As seen Figure 8 (1) (2), given a budget, ATHENA forces HSJA to generate AEs with much larger distances when compared to the UM. This clearly corroborates that ATHENA is effectively protecting the UM against HSJA. To quantify the defense capability, we use Distance Drop Ratio $= \frac{\text{dist}_{\text{ATHENA}} - \text{dist}_{\text{UM}}}{\text{dist}_{\text{UM}}}$, where $\text{dist}_{\text{ATHENA}}$ and dist_{UM} are distances of AEs targeting ATHENA and UM respectively. For a given query budget from $[100, 5000]$, *Distance Drop Ratio* is from 0.36 to 15.45 and from 0.40 to 9.27 for l_2 distance-based attack and l_∞ distance-based attack respectively. As seen from Figure 8 (3), *Distance Drop Ratio* increases when the given query budget increases. The reason is that distances of UM-targeted AEs drop much more significantly than that of ATHENA-targeted AEs when budget increases. That is, ATHENA is robust against HSJA with varying query budget. With a higher number of WDs, we expect a larger distance gap, i.e., a better defense capability, since the robustness of the ensemble will be enhanced in general.

F. Gray-box and White-box Attacks

We evaluated ATHENA with two white-box attacks: (i) a greedy approach *aggregating the perturbations* generated based on individual WDs; (ii) an *optimization-based approach* based on [2], [35]. We evaluated one realization of ATHENA— AVEL ensemble with 22 WDs on CIFAR-100 and AVEP ensemble with 72 WDs on MNIST.

Greedy approach. We use Algorithm 1 to generate AEs, each of which is able to fool at most a certain number, N, of WDs within a specific distance, measured by normalized l_2 dissimilarity, between the perturbed and the benign samples. The value of N depends on the ensemble strategy being used, the attack being launched, and the time budget the attacker is willing to pay. For example, for an ensemble model using MV strategy, the attacker has to fool at least half of the WDs to guarantee an ineffective MV ensemble with a targeted attack. By forcing more than half of the WDs agree on a label y', where $y' \neq y$ and y is the true label for a given input $\mathbf{x}$, the MV ensemble will always incorrectly label $\mathbf{x}$ as y'. The use of a maximum dissimilarity ensures that the sample is not being perturbed too much, also acts as a factor that indicates the efforts that the attacker can afford. In the algorithm, any attack methods (e.g., FGSM, CW_l_2, etc.) can be used to generate the adversarial perturbations.

In the context of white-box threat model, we evaluated MV ensemble built with 22 WDs on CIFAR-100. In order to add perturbations related to a variety of transformations into a single adversarial example within a reasonable time, we choose FGSM ($\epsilon = 0.01$). The smaller ϵ being used the longer time it will take to generate an AE to successfully fool a single WD by Algorithm 1. We evaluated MV ensemble against 500 AEs, evenly distributed across 100 classes, with different configurations as specified in Table III. In white-box threat model, beside the error rate, we also measure the attacker's

Algorithm 1: Crafting white-box AEs (Greedy)

```
   input    : x, y, attacker, N, max_dissimilarity
1  F_fooled ← {};
2  F_cand ← all weak defenses;
3  x' ← x;
4  while size(F_fooled) < N do
5  |    f_target ← pickTarget(F_cand, strategy);
6  |    // getPerturbation(x) returns ||x − x'||_2
7  |    perturbation ← attacker.getPerturbation(f_target, x');
8  |    x'_tmp ← x' + perturbation
9  |    // dissimilairity(x',x) returns the normalized l_2
   |       dissimilarities between x' and x
10 |    if dissimilarity(x'_tmp, x) > max_dissimilarity then
11 |    |    break;
12 |    end
13 |    for f_t_i in F_cand do
14 |    |    if y ≠ f_t_i(x'_tmp) then
15 |    |    |    addModel(F_fooled, f_t_i);
16 |    |    |    removeModel(F_cand, f_t_i);
17 |    |    end
18 |    end
19 |    x' ← x'_tmp;
20 end
21 return x';
```
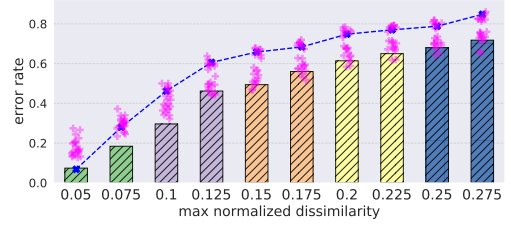

Fig. 9: Evaluation of UM and MV ensemble on the greedy attack on CIFAR-100. 10 AE variants were generated using FGSM. From left to right, the AEs were generated with a larger constraint of max dissimilarity with respect to the corresponding benign samples.
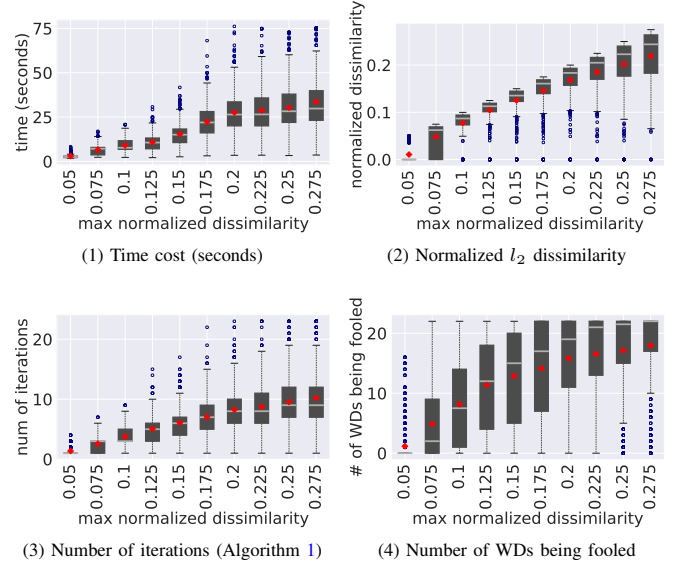

(1) Time cost (seconds)

(2) Normalized l_2 dissimilarity

(3) Number of iterations (Algorithm 1)

(4) Number of WDs being fooled

Fig. 10: Cost of AEs generation using the greedy attack on CIFAR-100.

effort in launching a white-box attack against ATHENA with respect to (i) the time for generating one AE and (ii) the number of iterations to craft an AE that is able to fool a sufficient number of WDs, (iii) normalized l_2 dissimilarity, and (iv) the number of WDs it is able to fool.

As shown in Figure 9, with modest effort, the attacker is able to generate less costly AEs. However, such AEs are not strong enough to fool the MV ensemble. For example, it takes on average less than 2 seconds to generate a single AE with max_dissimilarity = 0.05, which is 10x of crafting an AE in zero-knowledge threat model. However, only 7.4% of the generated AEs were misclassified by ATHENA. On average, each of the AEs can only fool 5% (1/22) of WDs. As max_dissimilarity increases, the attacker is able to generate stronger AEs. For example, with max_dissimilarity = 0.275, 71.8% of the generated AEs successfully fooled the MV ensemble. Each of such AEs, on average, is misclassified by 81.82% (18/22) of the WDs, making MV ensembles ineffective. However, this comes at a price:

- *High cost*: The time cost for generating a single AE is high, for example 34 seconds for the strongest AE, i.e., max_dissimilarity = 0.275, which is 170x of the cost for generating an AE in zero-knowledge. For MNIST with 72 WDs, it takes on average 610 seconds to generate an AE with max_dissimilarity = 1.0, which is 310x of generating an AE in zero-knowledge model (cf. Figure 25). This provides a tradeoff space, where realizations of ATHENA that employ larger ensembles incur more cost to the attackers.

- *Easily detectable*: Furthermore, although such AEs successfully fool the MV ensemble, they were distorted heavily[7] and very likely to be detected either by a human (see Figure 21) or an adversarial detector. We adapted a adversarial detector [8] and tested on MNIST. As presented in Figure 11,

the detector is able to successfully detect 89.8% of AEs for the white-box. An enhanced version of ATHENA with the detector (Detection + MV ensemble) can achieve a high accuracy of above 90% on AEs with any strength by either detecting and/or recovering the correct label via WDs.

The optimization-based approach synthesizes AEs based on an ensemble model by combining *Expectation Over Transformation* (EOT) [2] with Random Self-Ensemble (RSE) [35]. EOT is a general-purpose algorithm for generating AEs that are robust over the chosen distribution of transformations (T):

$$x' = \underset{x'}{\arg\max} \; \mathbb{E}_{t \sim T}[\log f(y_t | t(x')]$$
$$\text{s.t. } \mathbb{E}[d(t(x'), t(x)) < \epsilon, x \in [0,1]^d, \quad (6)$$

which extends the standard attack in (1). We extend EOT for ensembles,[9] generating AEs by averaging the EOTs [35] of all the WDs known to the attacker:

$$\underset{x'}{\arg\max} \; \frac{1}{K} \Sigma_{i=1}^{K} \mathbb{E}_{t \sim T}[\log f_i(y_t | t(x')]$$
$$\text{s.t. } \mathbb{E}_{t \sim T}[d(t(x'), t(x))] < \epsilon, x \in [0,1]^d, \quad (7)$$

where $d(\cdot, \cdot)$ is a distance function, T is a chosen distribution of transformation functions $t(\cdot)$, $f(\cdot)$ is the targeted model, and

---

[7]The average normalized l_2 dissimilarity of AEs (0.2192) is larger than that of the strongest FGSM in zero-knowledge model (0.1963).

[8]rfeinman: github.com/rfeinman/detecting-adversarial-samples

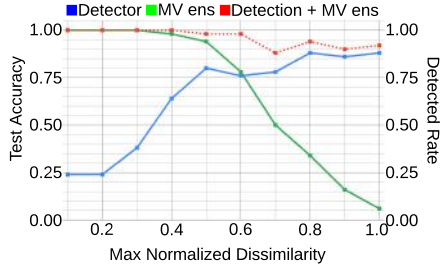[9]For an ensemble of size one, this extended attack will be reduced to EOT.

Fig. 11: Evaluation results for White-box threat models by detector, MV ensemble, and a simple supreme model combining the detector and the MV ensemble (cf., detection + MV ens): (i) percentage of detected adversarial examples by detector (blue lines), (ii) test accuracy of the MV ensemble (green lines), and (iii) as compared to the maximum normalized dissimilarity of AE sets.

(1) Error rate   (2) Computational cost   (3) Normalized l_2 distance

Fig. 12: Evaluation of AVEL ensemble against the optimization-based white-box attack on CIFAR-100.

$f_i(\cdot)$ is one of the $i \in N$ WDs in the ensemble. Based on this extended approach, a gray-box attack is when the adversary has white-box access to K WDs in the ensemble, where $K < N$, and a white-box attack is when the adversary has white-box access to the full list of WDs, where $K = N$.

For simplicity, we consider the distribution of transformations consist of rotations (a random angle between -20 to 20 degrees), adding Gaussian noise, and translation (a random offset between -25% to 25% on x- and/or y-axis). We evaluated several variants by evaluating an input over 30, 100, and 500 randomly sampled transformations for individual WD. We generated 10 variants on MNIST: 9 in the gray-box model, with 10% to 90% of the WDs accessed by the attacker and 1 variant in the white-box model with all WDs are known to the attacker. For CIFAR-100, we generated 5 variants: 4 variants in the gray-box model, with 20%, 40%, 60%, and 80% of the WDs are known to the attacker and 1 variant in the white-box scenario. For comparison, we used EOT to generate AEs based on the UM. The evaluation results for CIFAR-100 are shown in Figure 12, and for MNIST are shown in Figure 23 and Figure 24.

As expected, as the adversary becomes more white-box (i.e., having access to more WDs), it can launch more successful attacks (Figure 12 (1)) without even increasing the perturbations (Figure 12 (1)). However, the cost of generating AEs is increased (Figure 12 (2)). The experimental results show that ATHENA can even blocks adversarial attacks from a strong adversary specifically designed to target ATHENA. The attacker has obviously the choice to sample more random transformations and also chooses a distribution of a large number and diverse transformations to launch stronger attacks. However, this will incur a large computational cost to the attacker.

V. RESEARCH QUESTIONS

As shown in our extensive evaluations, ATHENA is effective against a wide variety of adversaries from the weakest threat model (zero-knowledge) to the strongest one (white-box).

Here we wish to address the following questions:

1) **RQ1**: Why do the quantity and diversity of WDs matter?
2) **RQ2**: Does our defense generalize across different models?
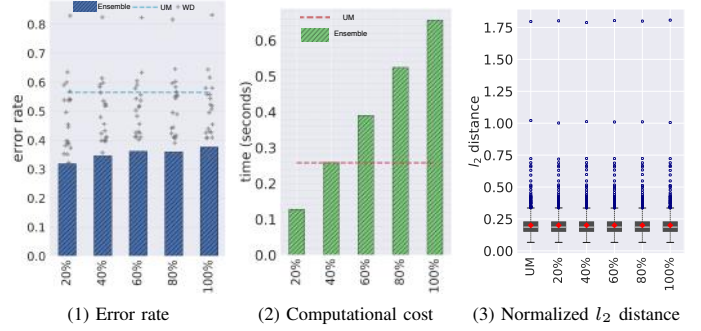3) **RQ3**: What is the overhead of ATHENA?

VI. RQ1. DIVERSITY OF WEAK DEFENSES

Study Design. Here we investigate how the number of WDs and the diversity relates to an ensemble's effectiveness for countering adversarial attacks.

Several metrics for ensemble diversity have been proposed, e.g., diversity among the non-maximal predictions of individual members (cf. WDs in ATHENA) in the ensemble [48], or negative correlation learning in training individual members to encourage different members to learn different aspects of the training data [13], [28], [36], [37]. Fiversity on the non-maximal predictions of individual WDs promotes the ensemble's diversity at the training phase by serving as a penalty in adaptive diversity promoting (ADP) regularizer [48]. Since we have trained individual WDs without such regularizers, we could not use these metrics in this work, and in particular, we have measured the diversity of several ensembles in ATHENA and the diversity of the non-maximal predictions was low as expected. Incorporating such regulaizers during the training of WDs is a potential future direction of this work.

In this work, based on our empirical insights that were discussed in Section III-A, we defined a new diversity metric by considering both the individual WDs performance and how they complement each other in an ensemble:

$$\psi = (\min_{i \in \{1,\ldots,K\}} |S_i|) - (|\bigcap_i S_i|), \qquad (8)$$

where S_i is the set of examples correctly predicted by WD_i on dataset D and $|\cdot|$ is the cardinality of S_i. The intuition behind the proposed diversity metric is that it encourages the effectiveness of individual WDs via $\min |S_i|$ and a wide diverse coverage on correctly predicted inputs by requiring small overlaps on correctly predicted samples via $|\bigcap_i S_i|$. Note that ψ is always positive for any ensemble defense.

To investigate the impacts of an ensemble's size and diversity on the effectiveness, we first created AVEL ensembles with 2 to 14 random WDs from a library of 22 WDs and then evaluated the ensemble's effectiveness (in terms of the error rate) against various attacks and computed the corresponding ensemble diversities. We repeated the experiment 5 times on CIFAR-100 and reported the results for the best, worst, and the mean performers over the 5 trials in Figure 13. Experiments were also performed on MNIST with the size varying from
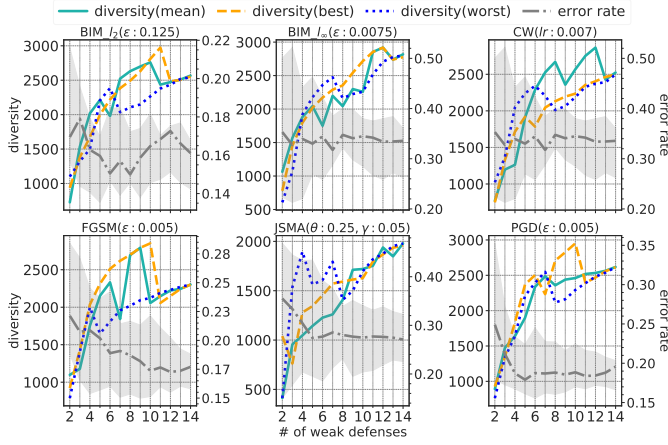
Fig. 13: Ensemble diversities and error rates of best, worst, and median performers on CIFAR-100.

Fig. 14: ATHENA using ResNet-Shake as weak defenses on CIFAR-100.

2 to 11 random WDs from a library of 72 WDs, and the experiment results are presented in Figure 29.

Results. The results indicate that as more WDs are incorporated, the ensembles tend to become more diverse (according to the metric defined in (8)), and the ensembles in the worst case scenario[10] (i.e., see the upper bound of error rates) become more effective to adverse adversarial attacks. For example, in CW(lr : 0.007), the worst performer's error rate declines from 50% to 26%. Note that this experiment is by no means comprehensive, as it only incorporates one diversity metric and a specific library of transformations. We leave such comprehensive study of the impact of diversity to ensemble-based defenses as future work. However, the results indicate a promising direction for future research to develop search mechanisms to dynamically synthesize such ensemble-based defenses incorporating heuristics such as diversity metrics to construct effective ensembles to counter adversarial attacks.

Summary. A diverse set of weak defenses is necessary to build a robust ensemble defense against adversarial attacks.

VII. RQ2. GENERALITY OF ATHENA

Study Design. To evaluate how general ATHENA is, we realized ensembles using different type of models and evaluated the ensembles in the context of zero-knowledge threat model. For the CIFAR-100 dataset, we built 4 ensembles with 22 WDs, associated with same list of transformations when using WRN classifiers, now using $26\ 2 \times 32d$ ResNet with Shake-Shake regularization (ResNet-Shake). The undefended ResNet-Shake model achieved a test accuracy of 75.55% on BS. The ensembles using ResNet-Shake models (ResNet-Shake ATHENA) were tested against 6 attacks in 2 configurations. For MNIST, we built 5 ensembles with 71 Linear SVM classifiers, which achieved a test accuracy of 94.97% on BS. The ensembles built with SVM models (SVM ATHENA) were against 7 different attacks with 3 configurations.

---

[10]The focus on worst case scenario is by intention as there could be large number of potential ensembles. Given a transformation library of size K, we have $\binom{n}{K}$ choices for constructing an ensemble defense of size n.
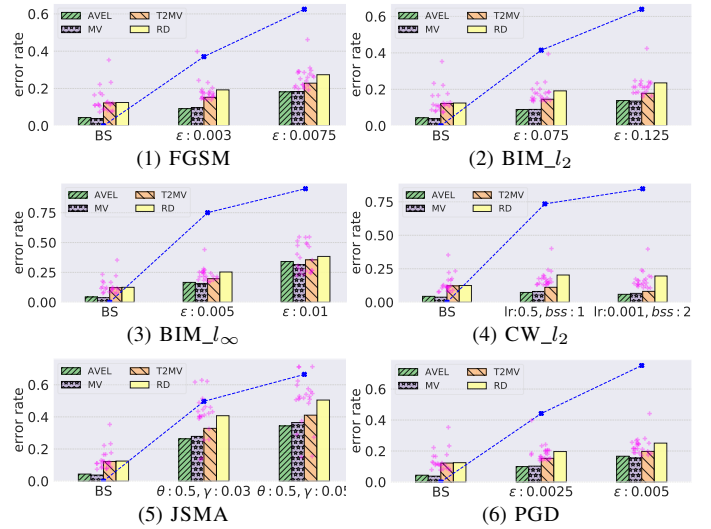
Results. The results are presented in Figure 14 for CIFAR-100 (cf. Figure 4) and in Figure 26 for MNIST (cf. Figure 32). We made the following observations:

- Transformations are able to improve model robustness regardless of the type of model. Almost every individual WD makes fewer mistakes than UM.
- The model architecture impacts the model's robustness, as also shown in previous research where it was shown that model capacity plays an important role [40]. For example, on BIM_$l_\infty(\epsilon : 0.01)$ AEs, the error rates of WRN WDs lie in a wider range than that of ResNet-Shake WDs $((21\%, 80\%)$ versus $(20\%, 58\%))$. Further, the error rate of ATHENA with WRN models is higher than that of ResNet-Shake using the same ensemble strategy. For example, by using AVEL strategy, WRN ATHENA has an error rate of 42% and ResNet-Shake ATHENA achieves a smaller error rate of 38% on BIM_$l_\infty(\epsilon : 0.01)$ AEs. The same observation found in MNIST, where we compared ATHENA with CNN WDs in contrast of SVM ATHENA. However, fewer transformations in SVM ATHENA are effective against an adversary, therefore making ensembles based on majority voting ineffective. For example, for the strongest set of CW AEs, all CNN WDs achieve a test accuracy above 90%, while only 8.5% (6 out of 71) SVM WDs achieve a test accuracy of 90% or above, again indicating the model capacity is an important factor for countering adversarial attacks. When taking a closer look at the SVM WDs, we observed that SVM WDs associated to transformations like rotation, flipping, and distortion are not effective, indicating an interaction with the decision boundary formed by certain models. Previous research shown that deterministic SVMs are not robust against adversarial attacks [9], there is potential for investigating the connection between robust decision boundary formed by randomized SVMs [9] in connection with ensemble defenses such as ATHENA.
- The selection of model type impacts the consistency of

transformations on different attacks. For example, the performances of WDs are more consistent across various adversaries in SVM-ATHENA than in CNN-ATHENA. Filters and segmentation achieve top 30% performance across all types of adversaries, while rotation and distortion tend to work inefficiently. To further investigate this, we performed Spearman's rank correlation on classification accuracy of the 71 SVM WDs for each pair of AE sets as shown in Figure 15 (1). The high correlation coefficients indicate that, compared with CNN ATHENA in Figure 28, SVM-based WDs tend to perform more similarly across attacks.

- The performances of transformations vary on different types of ML models. For example, segmentation performs the worst against most adversaries in CNN models, while it is one of the top 10 performers against in SVM-ATHENA. Specifically, we computed Spearman's rank correlation between classification accuracy of the 71 pair of CNN and SVM classifiers per adversary. The results in Figure 15 (2) indicate that the stronger the AEs the more different transformations perform (except BIM_l_2 and DEEPFOOL).

These observations point toward a some promising further work: (i) Optimizing transformation parameters (e.g., rotation angles) and/or types based on automated search over the infinite space of possible transformations rather than manual design as we did in this paper; (ii) A search problem over the space of model architectures [23], model type, in addition to the ensemble parameters such as the number of individual WDs and ensemble strategy. (iii) Extending ATHENA using hybrid ensembles including WDs that were trained on different models such as CNN and SVM.

Summary. Irrespective of the type of models used for building WDs, ATHENA can enhance robustness, but with different degrees of enhancements.

VIII. RQ 3. OVERHEAD OF ATHENA

Study Design. Since we need to run WDs in parallel in ATHENA, it is clear that *memory consumption* will be roughly N times of that used by the UM if N WDs are deployed. However, the *inference time* is only limited by the slowest WD in the ensemble. Constructing ATHENA also involves training N WDs and the *training time* for individual WDs is determined by the model type and training data. To empirically evaluate the runtime overhead, we deployed ATHENA in a Google Cloud machine (see Table IV).

Results. As seen in Figure 16, the most expensive transformation still consumes less time than model inference in CIFAR-100; however, for MNIST, the most expensive transformation is more costly than the undefended model inference[11]. This points to a tradeoff space, where the size of ensemble can be adjusted to tradeoff the ensemble performance (cf. results in Section VI) with inference overhead, which may be necessary for time-critical applications (especially when deployed in resource constrained devices).

---

[11]The cost for calculating the ensemble strategy was 10x lower than model inference, so we only reported time for performing transformations

(1) Rank correlation across adversaries

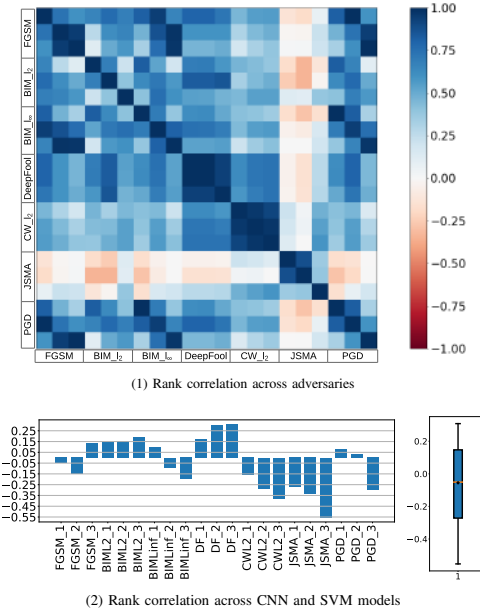(2) Rank correlation across CNN and SVM models

Fig. 15: Rank correlations of effectiveness of WDs (a) across AEs that were generated based on SVM UM (Settings of AEs follow the same order shown in Figure 26), (b) across CNN models and SVM models.
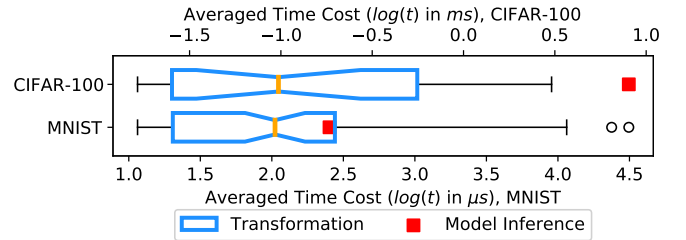

Fig. 16: Overhead of ATHENA against FGSM.

Summary. The memory overhead ATHENA is proportional to the number of WDs; however, ATHENA can be designed to have runtime overhead on par with the model inference by trading the number of transformations.

IX. DISCUSSIONS: LIMITATIONS & FUTURE DIRECTIONS

Even though ATHENA increases model robustness, the following several areas are known as areas for improvement:

- *Integration with other defense mechanisms*: While our defense causes a significant drop in error rate in a variety of settings, there remains room for improvement. For example, in white-box scenario, we have shown that combining ATHENA with an adversarial detection make the defense significantly more effective. ATHENA can be integrated with other defense mechanisms given its well-defined interface.
- *Automated construction of* ATHENA: Here, we "manually" constructed an instance of ATHENA with only 72 WDs; however, there is a tradeoff space, where the framework can be expanded with more (or fewer) types of transformations depending on the strength of the defense one may desire and the overhead that one can handle. Based on the experimental results in Sections VI, VII, and VIII, the process

of constructing the ensemble can be formulated as a multi-objective optimization problem, where the aim is to automatically search for an optimal and diverse set of WDs with low overhead using a search method. Although a very difficult problem, this can be alleviated using heuristics to guide the search and incorporating composition of transformations ($t_g = t_1 \circ t_2$) to facilitate the search over transformations.

- *Building a hybrid* ATHENA: ATHENA is a generic framework that can be used in conjunction with a variety of ML classifiers with minimal or no modification of the original classifiers, while still being effective at recovering adversarial examples. The results in Section VII indicated that building a *hybrid framework* using optimizations on the space of (i) individual WDs, (ii) model types and architecture, and (ii) ensemble strategies is a fruitful direction for a future work.

- *Integration with cloud-based and edge-based environments*: ATHENA is flexible so it can be adopted and integrated with machine learning models within different contexts. For cloud-based services, it has the capability to be expanded by incorporating more WDs. For resource-constrained environments such as edge, ATHENA can be adjusted to incorporate a lower number of WDs to decrease the overhead. In both scenarios, the WDs and the ensemble strategy can be adjusted at deployment time given the environmental conditions such as availability of resources, change in the behavior of adversaries, the toleration-level of attacks by the service providers, and the cost that the provider wants to impose upon to the attacker.[12] Within this tradeoff space, there is potential for *dynamically adapting the ensemble defense*, as well as for *hybrid deployments* over multiple deployment platforms (e.g., multi-cloud [47]).

X. RELATED WORK

Existing defense techniques against adversarial attacks can classified in two broad categories: (i) *Model-agnostic (reactive)* approaches either detect adversarial examples before feeding them to classifiers [20], [43]; attempt to remove adversarial perturbations from the input [3], [12], [15], [22], [38], [39], [45], [64] or features [63], [66]; or combining both detection and removal [42]; (ii) *Model-specific (proactive)* approaches, on the other hand, make changes to the model itself or the supervised learning procedure. For example, several forms of *adversarial training* based on combining normal and adversarial examples in the training [32], [49], [59], [61] or using an adversarial objective as a regularizer [10], [10], [19], [29], [31], [34], [55] have been proposed.

Model-agnostic defenses have also been shown insufficient for removing adversarial perturbations from input data [6]. Even small ensembles of model-agnostic defenses have been shown to be brittle even against non-adaptive adversaries [25]. Model-specific defenses make strong assumptions (e.g., the norm they use for creating adversarial examples) about the type and nature of the adversary; therefore, the adversary can alter its attack to circumvent such model-specific defenses.

The concept of "diversity" has been previously explored: (i) regularization that promotes diversity in ensembles during training [48], (ii) by applying a carefully designed ensemble strategy that each individual member is randomly assigned to different tasks by special code design [62], and (iii) mixed-precision neural networks [53]. These approaches rely on simultaneous training of individual members and, therefore, incurs a computational cost of $\mathcal{O}(m^3)$, where m is the size of the ensemble [27]. As a result, these methods cannot scale to large ensembles. Building defenses based on small sets of weak defenses can be easily circumvented by even weak adversaries [25], [60].

XI. CONCLUSIONS

We proposed ATHENA, a framework with which one can customize a specific realization of an adversarial defense based on ensembles of many and diverse sets of weak defenses given robustness properties and constraints. We evaluated the proposed defense against state-of-the-art adversarial attacks under zero-knowledge, black-box, gray-box, and white-box threat models and found out that our defense makes the target model more robust against evasion attacks. To the best of our knowledge, even though there has been previous work on using transformation as a defense, our work is the first work that proposes a framework for building a customized defense and our comprehensive study provides evidence that an ensemble of "many diverse weak defenses" provide such tradeoff space and has some viable potential properties: (1) applicability across multiple models, (2) applicability in different domains (image, voice, video), and (3) agnosticism to particular attacks.

XII. ACKNOWLEDGEMENT

For more information about ATHENA please refer to the project website: https://softsys4ai.github.io/athena/.

REFERENCES

[1] Akshay Agarwal, Richa Singh, Mayank Vatsa, and Nalini K. Ratha. Image transformation based defense against adversarial perturbation on deep learning models. *IEEE Transactions on Dependable and Secure Computing*, 2020.

[2] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. *arXiv:1707.07397*, 2017.

[3] Arjun Nitin Bhagoji, Daniel Cullina, Chawin Sitawarin, and Prateek Mittal. Enhancing robustness of machine learning systems via data transformations. In *2018 52nd Annual Conference on Information Sciences and Systems (CISS)*, pages 1–5. IEEE, 2018.

[4] Battista Biggio and Fabio Roli. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, 84, 2018.

[5] Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, and Aleksander Madry. On evaluating adversarial robustness. *arXiv:1902.06705*, 2019.

[6] Nicholas Carlini and David Wagner. Magnet and "efficient defenses against adversarial attacks" are not robust to adversarial examples. *arXiv:1711.08478*, 2017.

---

[12]We have seen that the more diverse the ensemble becomes, the more difficult it is to attack it.

[7] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy (SP)*, 2017.

[8] Jianbo Chen, Michael I. Jordan, and Martin J. Wainwright. Hopskipjumpattack: A query-efficient decision-based attack, 2019.

[9] Yan Chen, Wei Wang, and Xiangliang Zhang. Randomizing svm against adversarial attacks under uncertainty. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 556–568. Springer, 2018.

[10] Moustapha Cisse, Piotr Bojanowski, Edouard Grave, Yann Dauphin, and Nicolas Usunier. Parseval networks: Improving robustness to adversarial examples. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 854–863. JMLR. org, 2017.

[11] Jeremy M. Cohen, Elan Rosenfeld, and J. Zico Kolter. Certified adversarial robustness via randomized smoothing. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 1310–1320. PMLR, 2019.

[12] Nilaksh Das, Madhuri Shanbhogue, Shang-Tse Chen, Fred Hohman, Li Chen, Michael E Kounavis, and Duen Horng Chau. Keeping the bad guys out: Protecting and vaccinating deep learning with jpeg compression. *arXiv:1705.02900*, 2017.

[13] Thomas G. Dietterich. Ensemble methods in machine learning. In *Multiple Classifier Systems*, pages 1–15, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg.

[14] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Xiaolin Hu, and Jun Zhu. Discovering adversarial examples with momentum. *CoRR*, abs/1710.06081, 2017.

[15] Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M Roy. A study of the effect of jpg compression on adversarial images. *arXiv:1608.00853*, 2016.

[16] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *CVPR*, June 2018.

[17] Samuel G Finlayson, John D Bowers, Joichi Ito, Jonathan L Zittrain, Andrew L Beam, and Isaac S Kohane. Adversarial attacks on medical machine learning. *Science*, 363(6433):1287–1289, 2019.

[18] Xavier Gastaldi. Shake-shake regularization. *arXiv preprint arXiv:1705.07485*, 2017.

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

[20] Kathrin Grosse, Praveen Manoharan, Nicolas Papernot, Michael Backes, and Patrick McDaniel. On the (statistical) detection of adversarial examples. *arXiv:1702.06280*, 2017.

[21] Chuan Guo, Jacob R Gardner, Yurong You, Andrew Gordon Wilson, and Kilian Q Weinberger. Simple black-box adversarial attacks. *arXiv:1905.07121*, 2019.

[22] Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens Van Der Maaten. Countering adversarial images using input transformations. *arXiv:1711.00117*, 2017.

[23] Minghao Guo, Yuzhe Yang, Rui Xu, Ziwei Liu, and Dahua Lin. When nas meets robustness: In search of robust architectures against adversarial attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 631–640, 2020.

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, June 2016.

[25] Warren He, James Wei, Xinyun Chen, Nicholas Carlini, and Dawn Song. Adversarial example defense: Ensembles of weak defenses are not strong. In *11th USENIX WOOT)*, 2017.

[26] Douglas Heaven. Why deep-learning ais are so easy to fool. *Nature*, 574(7777):163, 2019.

[27] Md M Islam, Xin Yao, and Kazuyuki Murase. A constructive algorithm for training cooperative neural network ensembles. *IEEE Transactions on neural networks*, 14(4):820–834, 2003.

[28] Md.M. Islam, Xin Yao, and K. Murase. A constructive algorithm for training cooperative neural network ensembles. *IEEE Transactions on Neural Networks*, 14(4):820–834, 2003.

[29] Harini Kannan, Alexey Kurakin, and Ian Goodfellow. Adversarial logit pairing. *arXiv preprint arXiv:1803.06373*, 2018.

[30] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

[31] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv:1611.01236*, 2016.

[32] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *ICLR Workshop*, 2017.

[33] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

[34] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, and Jun Zhu. Defense against adversarial attacks using high-level representation guided denoiser. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1778–1787, 2018.

[35] Xuanqing Liu, Minhao Cheng, Huan Zhang, and Cho-Jui Hsieh. Towards robust neural networks via random self-ensemble. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[36] Yong Liu and Xin Yao. Ensemble learning via negative correlation. *Neural Networks*, 12(10):1399–1404, 1999.

[37] Yong Liu and Xin Yao. Simultaneous training of negatively correlated neural networks in an ensemble. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 29(6):716–725, 1999.

[38] Jiajun Lu, Hussein Sibai, Evan Fabry, and David Forsyth. No need to worry about adversarial examples in object detection in autonomous vehicles. *arXiv:1707.03501*, 2017.

[39] Yan Luo, Xavier Boix, Gemma Roig, Tomaso Poggio, and Qi Zhao. Foveation-based mechanisms alleviate adversarial examples. *arXiv preprint arXiv:1511.06292*, 2015.

[40] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018.

[41] 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.

[42] Dongyu Meng and Hao Chen. Magnet: a two-pronged defense against adversarial examples. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017.

[43] Jan Hendrik Metzen, Tim Genewein, Volker Fischer, and Bastian Bischoff. On detecting adversarial perturbations. *arXiv:1702.04267*, 2017.

[44] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. *CoRR*, abs/1511.04599, 2015.

[45] Aamir Mustafa, Salman H Khan, Munawar Hayat, Jianbing Shen, and Ling Shao. Image super-resolution as a defense against adversarial attacks. *IEEE Transactions on Image Processing*, 29:1711–1724, 2019.

[46] Maria-Irina Nicolae, Mathieu Sinn, Minh Ngoc Tran, Beat Buesser, Ambrish Rawat, Martin Wistuba, Valentina Zantedeschi, Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, Ian Molloy, and Ben Edwards. Adversarial robustness toolbox v1.1.0. *CoRR*, 1807.01069, 2018.

[47] Claus Pahl, Pooyan Jamshidi, and Olaf Zimmermann. Architectural principles for cloud software. *ACM Trans. on Internet Technology (TOIT)*, 2017.

[48] Tianyu Pang, Kun Xu, Chao Du, Ning Chen, and Jun Zhu. Improving adversarial robustness via promoting ensemble diversity. volume 97 of *Proceedings of Machine Learning Research*, pages 4970–4979, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

[49] 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)*, pages 582–597, May 2016.

[50] Nicolas Papernot. Technical report on the cleverhans v2.1.0 adversarial examples library. *arXiv:1610.00768*, 2018.

[51] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, ASIA CCS '17, pages 506–519, New York, NY, USA, 2017. ACM.

[52] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 372–387. IEEE, 2016.

[53] Sanchari Sen, Balaraman Ravindran, and Anand Raghunathan. Empir: Ensembles of mixed precision deep networks for increased robustness against adversarial attacks. *arXiv preprint arXiv:2004.10162*, 2020.

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

14

[55] Uri Shaham, Yutaro Yamada, and Sahand Negahban. Understanding adversarial training: Increasing local stability of supervised models through robust optimization. *Neurocomputing*, 307:195–204, 2018.

[56] Ying Cai Shixin Tian, Guolei Yang. Detecting adversarial examples through image transformation. *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[57] Rainer Storn and Kenneth Price. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11, 1997.

[58] Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. One pixel attack for fooling deep neural networks. *CoRR*, abs/1710.08864, 2017.

[59] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv:1312.6199*, 2013.

[60] Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. *arXiv preprint arXiv:2002.08347*, 2020.

[61] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*, 2017.

[62] Gunjan Verma and Ananthram Swami. Error correcting output codes improve probability estimation and adversarial robustness of deep neural networks. In *Advances in Neural Information Processing Systems*, pages 8646–8656, 2019.

[63] Qinglong Wang, Wenbo Guo, Kaixuan Zhang, Xinyu Xing, C Lee Giles, and Xue Liu. Random feature nullification for adversary resistant deep architecture. *arXiv:1610.01239*, 2016.

[64] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan Yuille. Mitigating adversarial effects through randomization. *arXiv preprint arXiv:1711.01991*, 2017.

[65] Wayne Xiong, Jasha Droppo, Xuedong Huang, Frank Seide, Mike Seltzer, Andreas Stolcke, Dong Yu, and Geoffrey Zweig. Achieving human parity in conversational speech recognition. *CoRR*, abs/1610.05256, 2016.

[66] Fei Zhang, Patrick PK Chan, Battista Biggio, Daniel S Yeung, and Fabio Roli. Adversarial feature selection against evasion attacks. *IEEE transactions on cybernetics*, 46(3):766–777, 2015.

APPENDIX

TABLE II: Zero-knowledge attack configurations. "lr" stands for the learning rate, and "bss" stands for the binary-search-steps.

Dataset	Attack	Attack configuration
CIFAR-100	FGSM	ϵ : 0.002, 0.003, 0.005, 0.0075, 0.03
	BIM_l_2	ϵ : 0.05, 0.075, 0.125, 0.3, 0.5 iterations: 100
	BIM_l_∞	ϵ : 0.0025, 0.005, 0.0075, 0.01, 0.015 iterations: 100
	CW_l_2	lr: 0.0003, 0.001, 0.007, 0.02, 1.0 bss: 6 iterations: 10
	JSMA	θ : 0.01, 0.5, 0.25, −0.5, −0.3 γ : 0.04, 0.03, 0.05, 0.075, 0.1
	PGD	ϵ : 0.0015, 0.0025, 0.0035, 0.005, 0.015 eps_iter: $\epsilon/10$ iterations: 40
MNIST	FGSM	ϵ : 0.1, 0.15, 0.2, 0.25, 0.3
	BIM_l_2	ϵ : 0.75, 0.85, 1.0, 1.1, 1.2 iterations: 100
	BIM_l_∞	ϵ : 0.075, 0.082, 0.09, 0.1, 0.12 iterations: 100
	CW_l_2	lr: 0.0098, 0.01, 0.012, 0.015, 0.018 bss: 5 iterations: 100
	DEEPFOOL	overshoot: 3/255, 5/255, 8/255, 16/255, 50/255 iterations: 50
	JSMA	θ : 0.15, 0.17, 0.18, 0.21, 0.25 γ : 0.5
	ONE-PIXEL	pixel-count: 15, 30, 35, 40, 75 population-size: 100 iterations: 30
	MIM	ϵ : 0.06, 0.067, 0.075, 0.085, 0.1 iterations: 1000 delay_factor: 1.0
	PGD	ϵ : 0.075, 0.082, 0.09, 0.1, 0.11 eps_iter: $\epsilon/10$ iterations: 40

TABLE III: White-box and gray-box attack configurations.

Dataset	Strategy	# of WDs	Max Dissimilarity	Attack Configuration
CIFAR-100	MV	22	0.05, 0.075, 0.1, 0.125, 0.15, 0.175, 0.2, 0.225, 0.25, 0.275	$\epsilon = 0.01$
MNIST	MV	72	0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0	$\epsilon = 0.1$

TABLE IV: Hardware configurations for our experiments.

Dataset	Experiment

MNIST	**Training WDs**
	 PC 12 Intel(R) Core(TM)i7-8700 CPU @ 3.20GHz, 32 GB memory
	 Crafting AEs
	 zero-knowledge
	   Google Cloud VM:
	   8 Intel(R)Xeon(R) CPU @ 2.20GHz, 30 GB memory, 1 x NVIDIA Tesla P100
	 black-box
	   Google Cloud VM:
	   16 Intel(R)Xeon(R) CPU @ 2.30GHz, 14.4 GB memory
	 white-box
	   Google Cloud VM:
	   PC 12 Intel(R) Core(TM)i7-8700 CPU @ 3.20GHz, 32 GB memory
	 Examining overheads
	   Google Cloud VM:
	   16 Intel(R)Xeon(R) CPU @ 2.30GHz, 14.4 GB memory

CIFAR-100	**Training WDs**
	   Google Cloud VM:
	   8 Intel(R)Xeon(R) CPU @ 2.20GHz, 30 GB memory, 1 x NVIDIA Tesla P100
	 Crafting AEs
	 zero-knowledge
	   Google Cloud VM:
	   8 Intel(R)Xeon(R) CPU @ 2.20GHz, 30 GB memory, 1 x NVIDIA Tesla P100
	 black-box & white-box
	   Hyperion new 16gb GPU node:
	   48 cores, 128 GB memory, 1.5T local scratch, 2 x NVIDIA Volta V100 16GB gpgpus
	 Examining overheads
	   Hyperion new 16gb GPU node:
	   48 cores, 128 GB memory, 1.5T local scratch, 2 x NVIDIA Volta V100 16GB gpgpus

TABLE V: List of input transformations

| Category	Description
Rotation	Rotate an input by a certain angle: rotate $90°$, $180°$, and $270°$.
Flip	Flip an input horizontally* and/or vertically.
Shift	Shift an input in a direction by some pixels: shift left*, right, up, down*, top-left, top-right, bottom-left, and bottom-right.
Cartoonify*	Cartoonify apples a sequence of image process operations on the input, including bilateral filter, gray scaling, median blur, create edge mask, and add the mask to original input.
Affine*	Transform an input by mapping variables (e.g., pixel intensity values at position (x_1, y_1)) into new variables (e.g., (x_2, y_2)) by applying a linear combination of shift, rotation, scaling and/or shearing operations: compress vertically and/or horizontally, stretch vertically and/or horizontally.
Denoise	Noise is generally considered to be a random variable with zero mean. Denoise transformations average out noises from inputs: nl_means_fast*, nl_means, tv_menas, tv_chambolle, tv_bregman*, and wavelet*.
Morphology	Morphological transformations apply operations based on image shape: erosion*, dilation*, opening*, closing*, and gradient.
Noise	Add noises to an input: gaussian, localvar, pepper, poison*, salt, and salt&peper.
Augmentation	Real-time data augmentation: feature-wise std normalization* and sample-wise std normalization*.
Segmentation	Segmentation divides the image into groups of pixels based on specific criteria. We segment an input based on colors.
Quantization	Quantization reduces the number of colors in an input using k-mean technique: 4 clusters and 8 clusters.
Distortion	Distortion deforms the pixel grid in an input and maps the deformed grid to the destination image: distort by x-axis or y-axis.
Filter	Filter transformations smooth an input using various filter kernels: entropy, gaussian, maximum, median*, minimum, prewitt, rank*, scharr, roberts, sobel.
Compress	Save images in different image formats: jpeg (quality: 80%*, 50%, 30%, and 10%), png(compression: 1*, 5*, and 8*).
Geometric	Apply different geometric transformations to images: iradon, iradon_sart, and swirl*
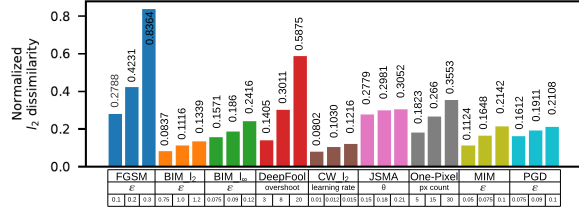
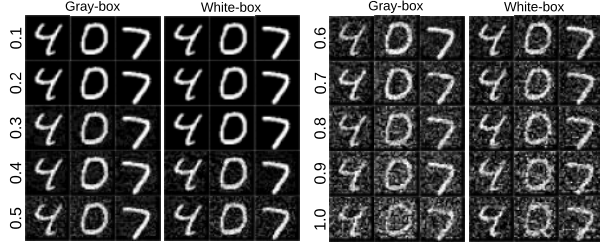Fig. 17: Normalized l_2 dissimilarity between BS and each AE generated for MNIST.

Fig. 18: MNIST sample adversarial examples generated in and white-box threat models with various constraints on maximum normalized dissimilarity.
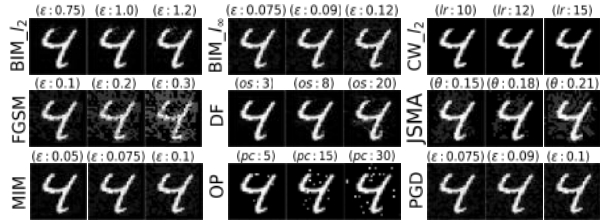
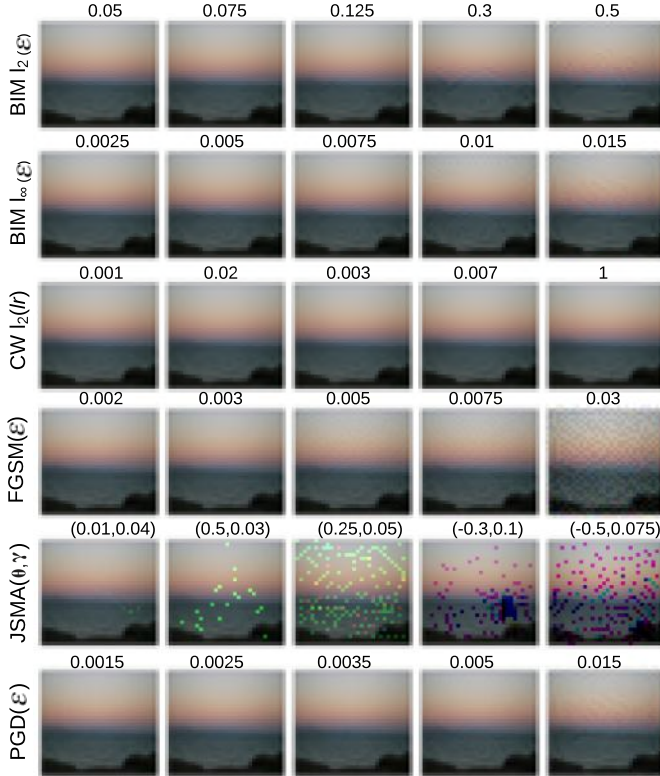Fig. 19: MNSIT sample adversarial examples generated by various adversaries.

Fig. 20: CIFAR-100 sample adversarial examples generated by various adversaries.
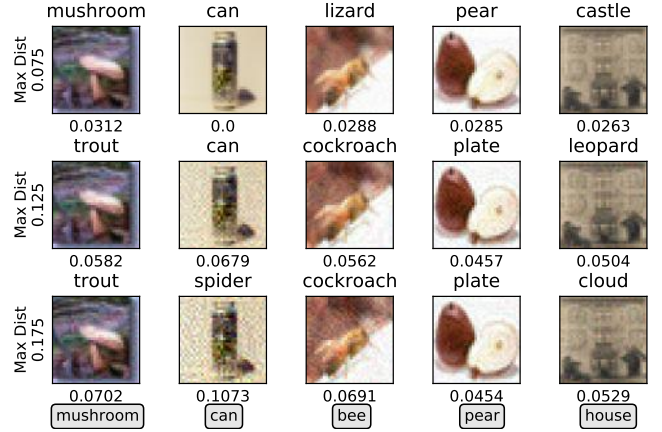
Fig. 21: The adversarial examples (CIFAR-100) generated by greedily aggregating perturbations under gray/white-box threat models with various constraints on maximum normalized dissimilarity. Note: 1) the text above each subfigure is its predicted label by ATHENA; 2) the number under each subfigure indicates its l2 distance from the original benign sample; 3) true labels are displayed in round rectangles.
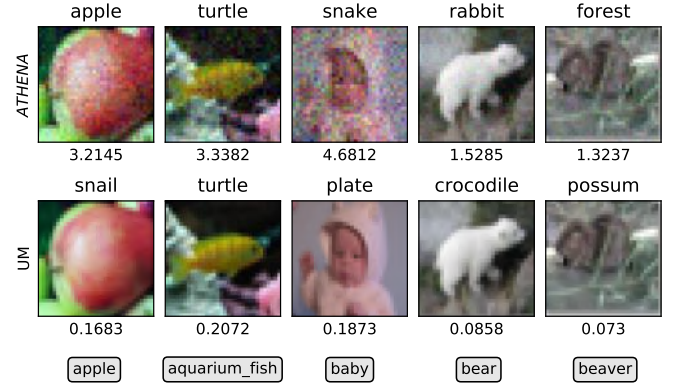
Fig. 22: The adversarial examples (CIFAR-100) generated by HSJA with l_2 distance and a budget of 5000 queries against the undefended model (UM) and ATHENA. Note: 1) the text above each subfigure is its predicted label by the corresponding target model; 2) the number under each subfigure indicates its l_2 distance from the original benign sample; 3) true labels are displayed in round rectangles.
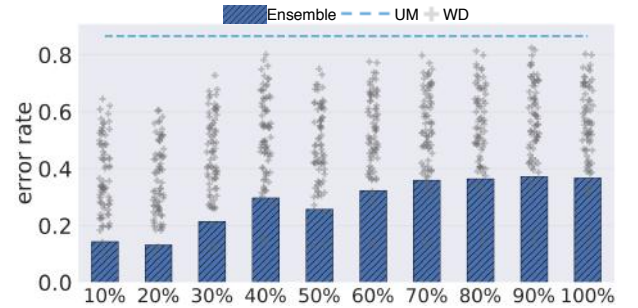
Fig. 23: Evaluations of an AVEP ensemble against the optimization-based attack on MNIST.
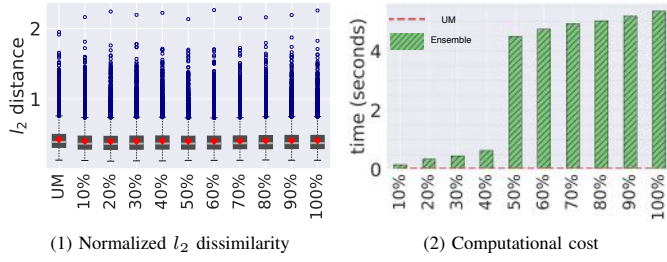
(1) Normalized l_2 dissimilarity

(2) Computational cost

Fig. 24: The normalized l_2 dissimilarity of AEs and computational time of the optimization-based attack on MNIST.

(1) Time cost (seconds)
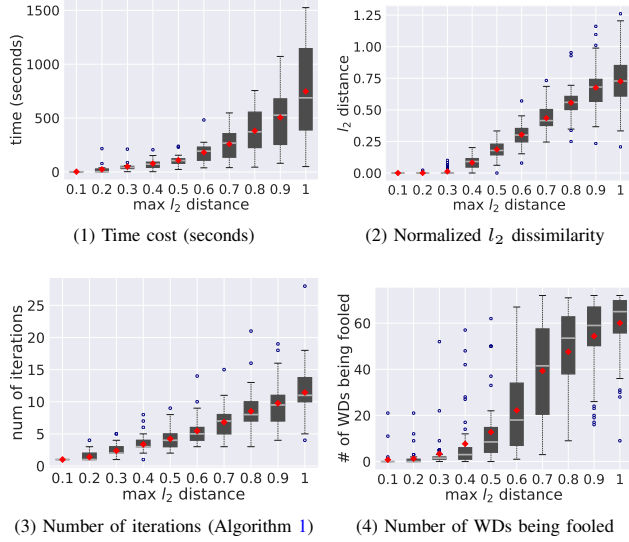
(2) Normalized l_2 dissimilarity

(3) Number of iterations (Algorithm 1)

(4) Number of WDs being fooled

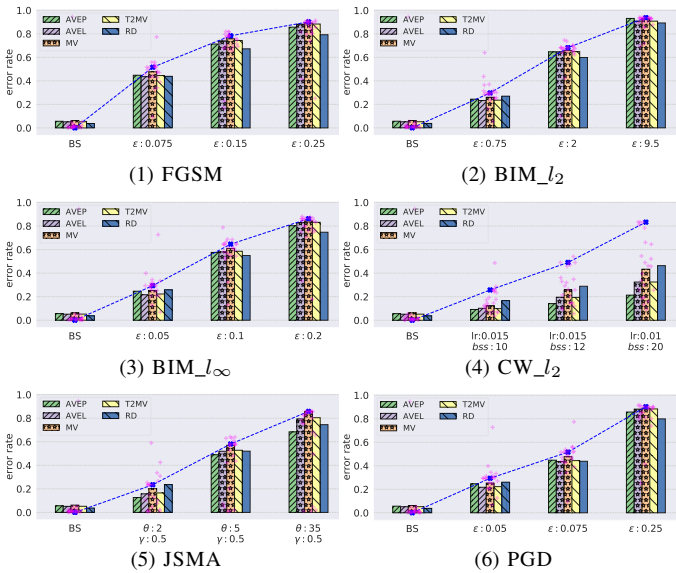Fig. 25: Cost evaluation of AEs generated based on white-box model (MV strategy) of the greedy attack for MNIST.

(1) FGSM

(2) BIM_l_2

(3) BIM_l_∞

(4) CW_l_2

(5) JSMA

(6) PGD

Fig. 26: Evaluation results for ATHENA using linear SVM as weak defenses on MNIST.
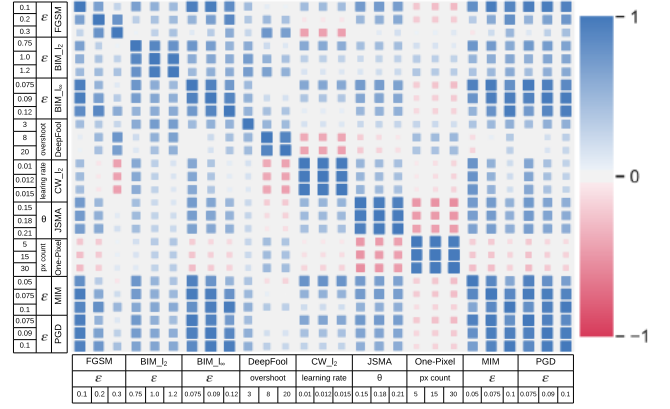
Fig. 27: Rank correlation between the accuracy of WDs (CNN architectures) in different AEs crafted by from MNIST.
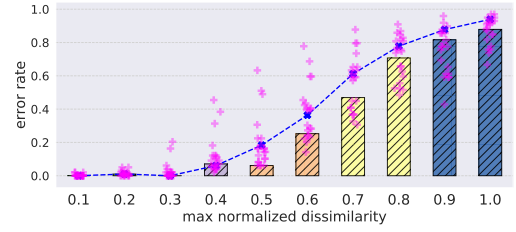
Fig. 28: Evaluation of UM and MV ensemble against the greedy attack for MNIST. 10 AE variants were generated using FGSM. From left to right, the AEs were generated with a larger constraint of max dissimilarity with respect to the corresponding benign samples.
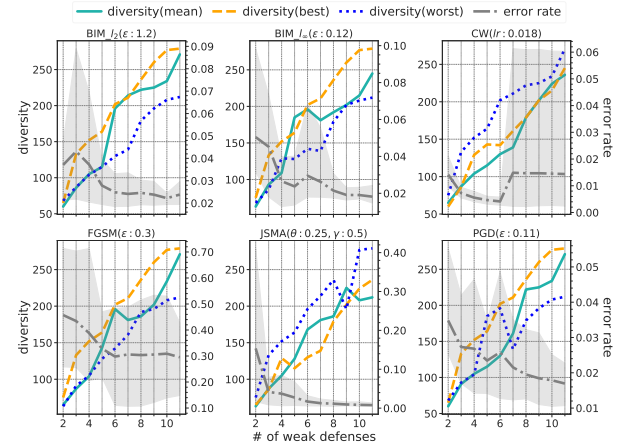

Fig. 29: Ensemble diversities against various attacks on MNIST.
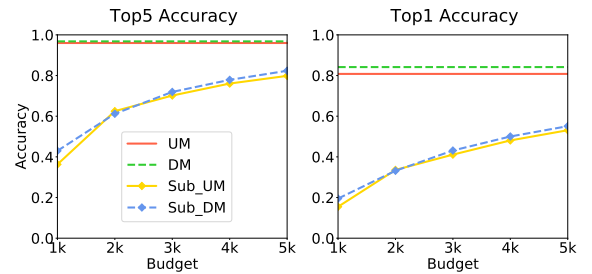

Fig. 30: Performance of surrogate model trained with different budgets. Note: UM and DM stand for the undefended model and the defended model with ATHENA respectively. 2) Sub_UM and Sub_DM stand for substitute models of UM and DM respectively.
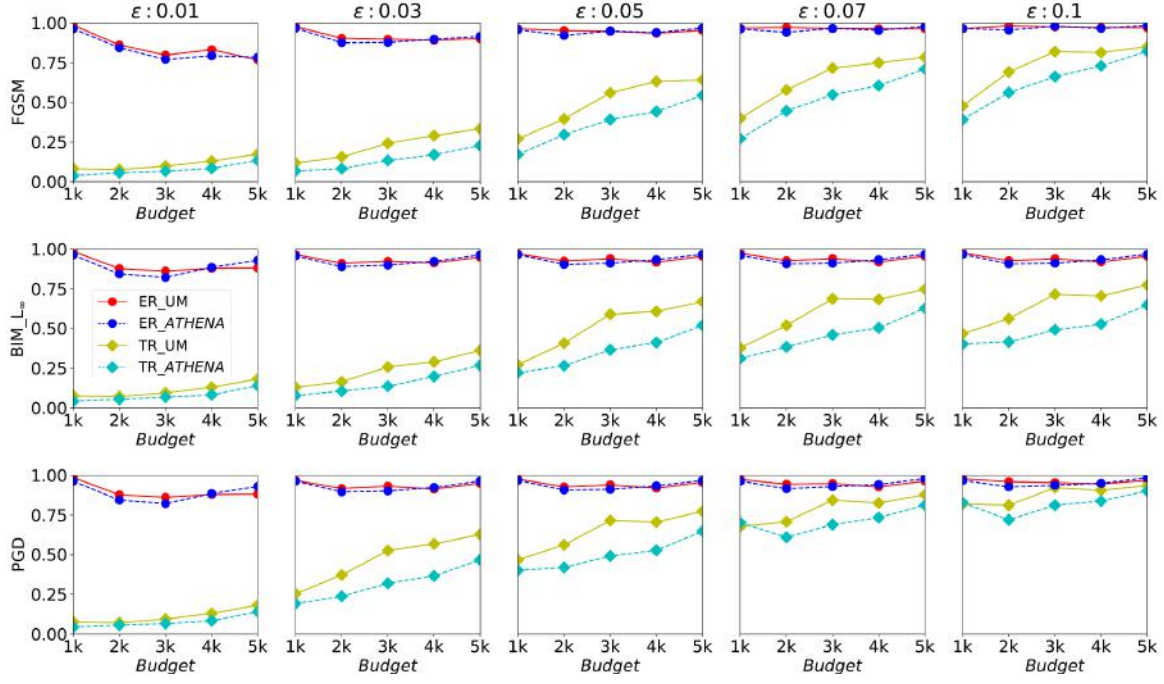
Fig. 31: Performance of transfer-based black-box attack with different budgets. Note: (1) ER, TR and UM stands for error rate, transferability rate and Undefended Model respectively; (2) ATHENA here is realized by AVEP ensemble strategy.
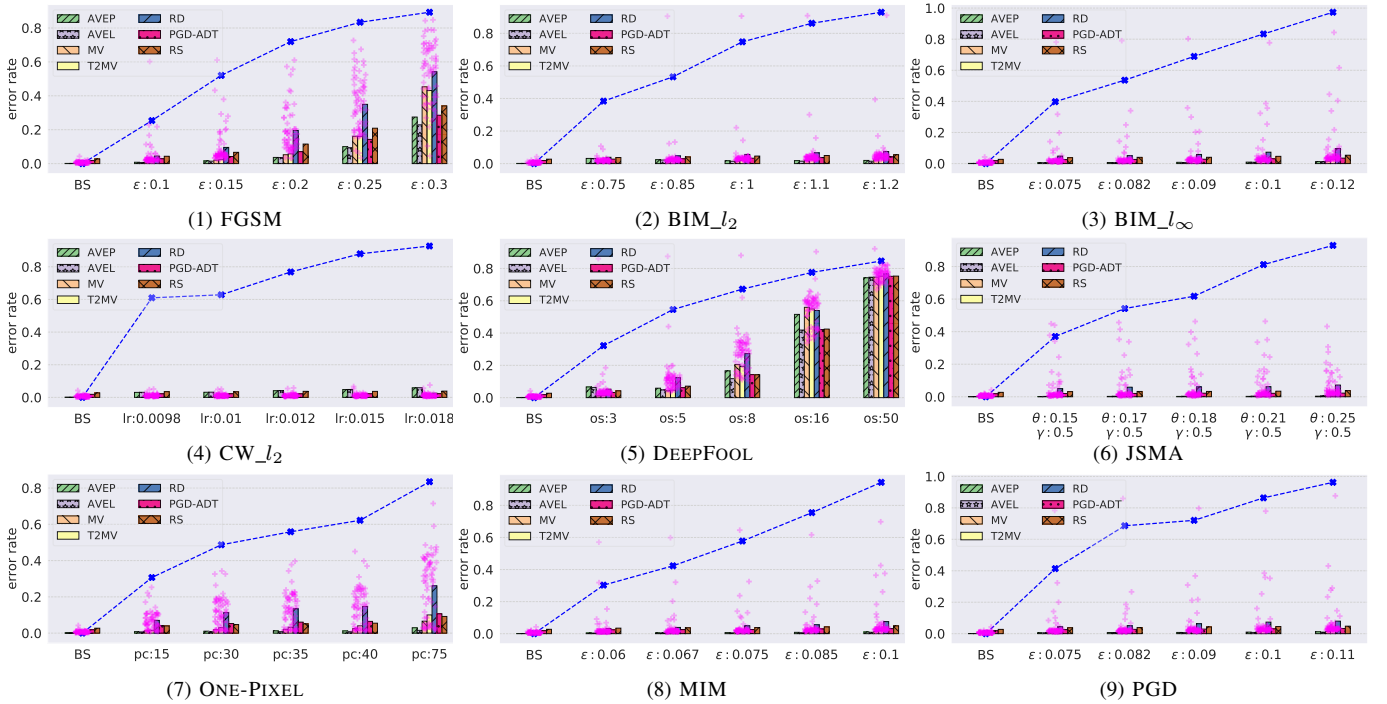

Fig. 32: Evaluation results for ATHENA using CNN as weak defenses on MNIST.