Harnessing the Vulnerability of Latent Layers in Adversarially Trained Models

Abhishek Sinha[*], **Mayank Singh**[*], **Nupur Kumari**[*], **Balaji Krishnamurthy**
Adobe Inc, Noida
{abhsinha,msingh,nupkumar,kbalaji}@adobe.com

Harshitha Machiraju & V N Balasubramanian
Department of Computer Science and Engineering
Indian Institute of Technology, Hyderabad, India
{ee14btech11011,vineethnb}@iith.ac.in

Abstract

Neural networks are vulnerable to adversarial attacks - small visually imperceptible crafted noise which when added to the input drastically changes the output. The most effective method of defending against these adversarial attacks is to use the methodology of adversarial training. We analyze the adversarially trained robust models to study their vulnerability against adversarial attacks at the level of the latent layers. Our analysis reveals that contrary to the input layer which is robust to adversarial attack, the latent layer of these robust models are highly susceptible to adversarial perturbations of small magnitude. Leveraging this information, we introduce a new technique Latent Adversarial Training (LAT) which comprises of fine-tuning the adversarially trained models to ensure the robustness at the feature layers. We also propose Latent Attack (LA), a novel algorithm for construction of adversarial examples. LAT results in minor improvement in test accuracy and leads to a state-of-the-art adversarial accuracy against the universal first-order adversarial PGD attack which is shown for the MNIST, CIFAR-10, CIFAR-100 datasets.

1 Introduction

Deep Neural Networks have achieved state-of-the-art performance in computer vision tasks such as image classification (He et al., 2016; Krizhevsky et al., 2012), semantic segmentation (He et al., 2017) and many others. However, recently such models have been shown to be extremely vulnerable to adversarial perturbations. These small, carefully calibrated perturbations, when added to the input, lead to a significant change in the network's prediction (Szegedy et al., 2014). The existence of adversarial examples pose a severe security threat to the practical deployment of deep learning models, particularly, in safety-critical systems (Akhtar & Mian, 2018).

Since the advent of adversarial perturbations, there has been extensive work in the area of crafting new adversarial perturbations (Madry et al., 2018; Moosavi-Dezfooli et al., 2017; Carlini & Wagner, 2017b). At the same time, several methods for adversarial defense have been proposed to protect models from these attacks (Goodfellow et al., 2015; Madry et al., 2018; Tramèr et al., 2018). Nonetheless, many of these defense strategies are continually defeated by new attacks. (Athalye et al., 2018; Carlini & Wagner, 2017a; Madry et al., 2018). In order to better compare the defense strategies, recent methods try to provide robustness guarantees by formally proving that no perturbation smaller than a given l_p(where $p \in [0, \infty]$)bound can fool their network (Raghunathan et al., 2018; Tsuzuku et al., 2018; Weng et al., 2018; Carlini et al., 2017; Wong & Kolter, 2018). Despite the efforts, the adversarial defense methods still fail to provide a significant robustness guarantee for appropriate l_p bounds (in terms of accuracy over adversarial examples) for large datasets like

[*]Authors contributed equally

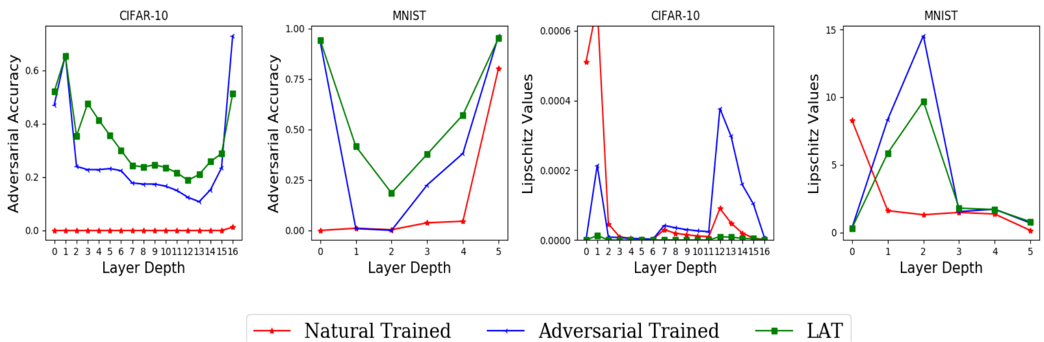

Figure 1: Adversarial accuracy (left, higher the better) and Lipschitz values (right, lower the better) of latent layers for models trained using different techniques on CIFAR-10 and MNIST datasets

CIFAR-10, CIFAR-100, and ImageNet (Russakovsky et al., 2015). Enhancing the robustness of models for these datasets is still an open challenge.

In this paper, we analyze the models trained using a state-of-the-art adversarial defense methodology (Madry et al., 2018) and find that while these models show robustness at the input layer, the latent layers are still highly vulnerable to adversarial attacks as shown in Fig 1. We utilize this property to introduce a new technique (LAT) of fine-tuning the adversarially trained model. We find that improving the robustness of the models at the latent layer boosts the adversarial accuracy of the entire model as well. We observe that LAT improves the adversarial robustness by ($\sim 4-6\%$) as well as test accuracy by ($\sim 1\%$) for CIFAR-10 and CIFAR-100 datasets.

Our main contributions in this work are the following:

- We study the robustness of latent layers of networks in terms of adversarial accuracy and Lipschitz constant and observe that latent layers of adversarially trained models are still highly vulnerable to adversarial perturbations.
- We propose a Latent Adversarial Training (**LAT**) strategy that significantly increases the robustness of existing state-of-the art adversarially trained models (Madry et al., 2018) for MNIST, CIFAR-10, and CIFAR-100 datasets. LAT can be used as a fine-tuning technique which achieves significant level of adversarial robustness with just few epochs ($\sim 2-5$) of additional training.
- We propose Latent Attack (**LA**), a new l_∞ adversarial attack that is comparable to PGD on multiple datasets. The attack exploits the non-robustness of in-between layers of existing robust models to construct adversarial perturbations.

2 Background And Related Work

2.1 Adversarial Attacks

For a classification network f, let θ be its parameters, y be the true class of n-dimensional input $x \in [0,1]^n$ and $J(\theta, x, y)$ be the loss function. The aim of an adversarial attack is to find the minimum perturbation Δ in x that results in the change of class prediction. Formally,

$$\begin{aligned} \Delta(x, f) &:= \min_{\delta} ||\delta||_p \\ \text{s.t } \arg\max(f(x+\delta;\theta)) &\neq \arg\max(f(x;\theta)) \end{aligned} \tag{1}$$

It can be expressed as an optimization problem:

$$x^{adv} = \underset{\tilde{x}:||\tilde{x}-x||_p<\epsilon}{\arg\max} \; J(\theta, \tilde{x}, y)$$

In general, the magnitude of adversarial perturbation is constrained by a p-norm where $p \in \{0, 2, \infty\}$ generally to ensure that the perturbed example is close to the original sample. Various other constraints for closeness and visual similarity (Rozsa et al., 2016; Xiao et al., 2018) have also been proposed for the construction of adversarial perturbation. In terms of categorization of attacks, there are broadly two type of adversarial attacks: *White box* and *Black box* attacks. White box attacks assume complete access to the network parameters, while in the latter there is no information available about network architecture or parameters. Some of the widely used adversarial attacks include FGSM (Fast Gradient Sign Method) (Goodfellow et al., 2015) and PGD (Projected Gradient Descent) (Madry et al., 2018). PGD is an iterative variant of FGSM.

While there has been extensive work in this area in the recent past (Yuan et al., 2019; Akhtar & Mian, 2018), our focus in this work is on attacks which utilize latent layer representations. To the best of our knowledge, in the only such earlier effort, (Sabour et al., 2016) proposed a method to construct adversarial perturbation by manipulating the latent layer of different classes. However, Latent Attack (LA) exploits the adversarial vulnerability of the latent layers to compute adversarial perturbations, which has not been done before.

2.2 Adversarial Defense

Popular defense strategies to improve the robustness of deep networks include the use of regularizers inspired by reducing the Lipschitz constant of the neural network (Tsuzuku et al., 2018; Cisse et al., 2017). There have also been several methods which use Generative Adversarial Networks (GANs) (Samangouei et al., 2018) for classifying the input as an adversary. However, these defense techniques have been shown to be ineffective to adaptive adversarial attacks (Athalye et al., 2018; Logan Engstrom, 2018). Hence, we turn to an adversarial training strategy (Goodfellow et al., 2015; Madry et al., 2018; Kannan et al., 2018), which injects adversarial examples in the training batch at every step of the training. Adversarial training is among the current state-of-the-art in adversarial robustness against white-box attacks. For a comprehensive review of other related work, we refer the interested reader to (Yuan et al., 2019; Akhtar & Mian, 2018).

In our current work, we seek to enhance the robustness of each latent layer, thereby increasing the robustness of the entire network. Previous efforts that are closest to ours include (Sankaranarayanan et al., 2018; Cihang Xie, 2018). Our work is different from them on the following counts:

- (Cihang Xie, 2018) observe that the adversarial perturbations on images lead to noisy features in latent layers. Inspired by this observation, they develop a new network architecture that comprises of denoising blocks at the feature layer which aims at increasing the adversarial robustness. However, we are leveraging the observation of low robustness at feature layer to perform adversarial training for latent layers in order to achieve higher robustness.
- (Sankaranarayanan et al., 2018) proposes an approach to regularize deep neural networks by perturbing intermediate layer activations. Their work has shown improvement in test accuracy over image classification tasks as well as minor improvement in adversarial robustness with respect to basic adversarial perturbation (Goodfellow et al., 2015). However, our work extensively focuses on the observation of the vulnerability of latent layers to a small magnitude of adversarial perturbations. We have shown improvement in test accuracy and adversarial robustness with respect of state of the art adversarial attack (Madry et al., 2018). We also propose a novel method for constructing adversarial perturbations.

3 Robustness of Latent Layers

Mathematically, a deep neural network with l layers and $f(x)$ as output can be described as:

$$f(x) = f_l(f_{l-1}(...(f_2(f_1(x; W_1, b_1); W_2, b_2)))...; W_l, b_l) \tag{2}$$

Here f_i denotes the function mapping layer $i-1$ to layer i with weights W_i and bias b_i respectively. From Eq. 2, it is evident that $f(x)$ can be written as a composition of two functions:

$$\begin{aligned} f(x) &= g_i \circ h_i(x) \mid 0 \le i \le l-1 \\ \text{where } \ f_0 &= I \text{ and } h_i = f_i \circ f_{i-1} ... \circ f_1 \circ f_0 \\ g_i &= f_l \circ f_{l-1} ... \circ f_{i+1} \end{aligned} \tag{3}$$

We can study the behavior of $f(x)$ to a slightly perturbed input by inspecting its Lipschitz constant, L_f, which is defined such that Eq. 4 (below) holds for all ν.

$$||f(x+\nu) - f(x)|| \leq L_f ||\nu|| \tag{4}$$

Having a lower Lipschitz constant ensures that the function's output at perturbed input is not significantly different. This can be further translated to higher adversarial robustness as has been shown by (Cisse et al., 2017; Tsuzuku et al., 2018). Moreover, if L_g and L_h are the Lipschitz constants of the sub-networks g_i and h_i respectively, the Lipschitz constant of f has an upper bound defined by the product of Lipschitz constant of g_i and h_i, i.e.

$$L_f \leq L_g * L_h \tag{5}$$

Hence, we observe that having robust sub-networks helps in higher adversarial robustness for the entire network.

For each latent layer i, we calculate an upper bound for the magnitude of perturbation in the sub-network output (ϵ_i) by observing the perturbation induced in latent layer for adversarial examples x^{adv}. To this end, we define ϵ_i as:

$$\epsilon_i = \text{Mean}_{x \in \text{test}} ||h_i(x) - h_i(x^{adv})||_\infty \tag{6}$$

We compute the adversarial robustness of sub-networks $\{g_i | 1 \leq i \leq l-1\}$ using a PGD attack in Eq 6, as illustrated in Fig 1.

We now describe our studies to analyze the robustness of latent layers in the popular datasets listed below. For adversarial training, the examples are constructed using PGD adversarial perturbations (Madry et al., 2018). [1] In the rest of this paper, by an adversarially trained model, we mean a model that is trained using PGD adversarial examples as in (Madry et al., 2018). We define adversarial accuracy of a model as the accuracy over the adversarial examples generated using the test-set of the dataset. Higher adversarial accuracy corresponds to a more robust model.

- **MNIST** (Lecun et al., 1989): We use the network architecture as described in (Madry et al., 2018). The naturally trained model achieves a test accuracy of 99.17% while the adversarially trained model achieves a test accuracy of 98.4%.
- **CIFAR-10** (Krizhevsky et al., 2010): We use the network architecture as in (Madry et al., 2018). The natural and adversarially trained model achieves test accuracies of 95.01% and 87.25% respectively.
- **CIFAR-100** (Krizhevsky et al., 2010): We use the same network architecture as used for CIFAR-10 with the modification at the logit layer to handle the number of classes in CIFAR-100. The naturally trained model achieves a test accuracy of 78.07%, while the adversarially trained model achieves a test accuracy of 60.38%.

All models were trained using code from (MadryLab, 2017a;b). Fig 1 shows the results.

We observe that for adversarially trained models, the adversarial accuracies of the sub-networks g_i are relatively less than that of the entire network f. The trend is consistent across all the different datasets. (We note that layer depth index as shown in Fig 1 is relative as it does not correspond to exact layer index in the model architecture. The sampled layers 1 were chosen uniformly across the model architecture.) Also, in all our studies, the deepest layer tested is the layer just before the logit layer and layer 0 corresponds to the input layer of f.

Fig 1 reveals that the sub-networks of an adversarially trained model are still vulnerable to adversarial perturbations. In general, it reduces with increasing depth. An interesting observation, however, is that robustness increases in the later layers of the network. The plots indicate that there is a scope for improvement in the adversarial robustness of different sub-networks. In the next section, we introduce our method that specifically targets at making g_i robust. We find that this leads to a boost in the adversarial and test performance of the whole network f as well.

To better understand the characteristics of sub-networks, we further analyze the subnetworks from the viewpoint of their Lipschitz constants, as discussed before. Since our focus in on behavior of

[1] Code at: `https://github.com/conference-submission-anon/LAT_adversarial_robustness`

the function in a small neighborhood of input samples, we compute the Lipschitz constants of the network f and sub-networks g_i using the local neighborhood of input samples i.e.

$$L_f(x_i) = max_{x_j \in B_\epsilon(x_i)} \frac{||f(x_j) - f(x_i)||}{||x_j - x_i||} \tag{7}$$

where $B_\epsilon(x_i)$ denotes the ϵ neighbourhood of x_i. For computational reasons, inspired by (Alvarez-Melis & Jaakkola, 2018), we approximate $B_\epsilon(x_i)$ by adding noise to x_i as given in Eq. 6. We report the Lipschitz constant values averaged over the test data for the considered datasets and models in Fig. 1. The plot reveals that while for the adversarially trained model, the Lipschitz value of f is lower than that of the naturally trained model, there is no such pattern in the sub-networks g_i. This observation again reinforces our hypothesis of the vulnerabilities of the different sub-networks against small perturbations.

4 Harnessing Latent Layers

4.1 Latent Adversarial Training (LAT)

In this section, we seek to increase the robustness of the deep neural network, f. We propose Latent Adversarial training (LAT) wherein both f and one of the sub-networks g_i are adversarially trained. For adversarial training of g_i, we use a l_∞ bounded adversarial perturbation computed via the PGD attack at layer i with an appropriate bound as defined in Eq. 6. We propose LAT as a fine-tuning technique which operates on an adversarially trained model to improve its adversarial and test accuracy further. We observe that performing only a few epochs ($\sim$ 5) of LAT on the adversarially trained model results in a significant improvement over adversarial accuracy of the model and reaches a state-of-the-art performance. Algorithm 1 describes our LAT training strategy.

Algorithm 1: Latent Adversarial Training (LAT): Algorithm to improve adversarial robustness of models

begin
- **Input**: Adversarially trained model parameters - θ, Sub-network index which needs to be adversarially trained - m, Fine-tuning steps - k, Batch size - B, Learning rate - η, hyperparameter ω
- **Output**: Fine-tuned model parameters
- **for** $i \in 1, 2, ..., k$ **do**
  - Training data of size B - $(X(i), Y(i))$.
  - Compute adversarial perturbation $\Delta X(i)$ via PGD attack.
  - Calculate the gradients $J_{adv} = J(\theta, X(i) + \Delta X(i), Y(i))$.
  - Compute $h_m(X(i))$.
  - Compute ϵ corresponding to $(X(i), Y(i))$ via Eq. 6.
  - Compute adversarial perturbation $\Delta h_m(X(i))$ with perturbation amount ϵ
  - Compute the gradients $J_{latentAdv} = J(\theta, h_m(X(i)) + \Delta h_m(X(i)), Y(i))$
  - $J(\theta, X(i), Y(i)) = \omega * J_{adv} + (1 - \omega) * (J_{latentAdv})$
  - $\theta \to \theta - \eta * J(\theta, X(i), Y(i))$
- **end**
- **return** fine-tuned model.

end

To test the efficacy of LAT, we perform experiments over CIFAR-10, CIFAR-100 and MNIST datasets. We also compare our approach (LAT) against two baseline fine-tuning techniques with the same amount of training period over the initial model: Adversarial Training (AT) (Madry et al., 2018) and Feature Noise Training (FNT) where Algorithm 1 is used with gaussian noise to perturb the latent layer.

Table 1 reports the adversarial accuracy corresponding to LAT and the baseline fine-tuning methods over CIFAR-10, CIFAR-100 and MNIST datatsets. PGD Baseline corresponds to 10 steps of PGD attack for CIFAR-10 and 100 and 40 steps of PGD attack for MNIST. We perform 2 epochs of fine-tuning for MNIST, CIFAR-10 and 5 epochs for CIFAR-100 in all of the methods.

The results in Table 1 correspond to the best performing layers[2]. Note that the adversarially trained model of CIFAR-10 and CIFAR-100 datasets had adversarial accuracies of 47.04% and 23.01% respectively.

[2] The results correspond to g_{11}, g_7 and g_2 sub-networks for the CIFAR-10, CIFAR-100 and MNIST datasets.

		Adversarial Accuracy		
Dataset	Fine-tuning Technique	PGD Baseline	PGD (100 steps)	Test Acc.
CIFAR-10	Adversarial Training	47.12 %	46.19 %	87.27 %
	Feature Noise Training	46.99 %	46.41 %	87.31 %
	LAT (Our Approach)	**53.84 %**	**53.04 %**	**87.80 %**
CIFAR-100	Adversarial Training	22.72 %	22.21 %	60.38 %
	Feature Noise Training	22.44 %	21.86 %	60.27 %
	LAT (Our Approach)	**27.03 %**	**26.41 %**	**60.94 %**
MNIST	Adversarial Training	93.75 %	92.92 %	**98.40 %**
	Feature Noise Training	93.59 %	92.16 %	98.28 %
	LAT (Our Approach)	**94.21 %**	**93.31 %**	98.38 %

Table 1: Adversarial accuracy for CIFAR-10, CIFAR-100 and MNIST datasets after fine-tuning by different techniques

As can be seen from the table, that only after 2.5 epochs of training by LAT on CIFAR-10 dataset, the adversarial accuracy jumps by $\sim 6.5\%$. Importantly, LAT not only improves the performance of the model over the adversarial examples but also over the clean test samples, which is reflected by an improvement of 0.6% in test accuracy. The same trend is visible for CIFAR-100 where training via LAT for 5 epochs results in an increment of 4% and 0.6% in adversarial and test accuracy respectively. Table 1 also reveals that the two baseline methods do not lead to any significant changes in the performance of the model. As the adversarial accuracy of the adversarially trained model for the MNIST dataset is already high (93.75%), our approach seems to not lead to significant improvements (although there is a $\sim 0.46\%$ improvement).

To analyze the effect of LAT on latent layers, we compute the robustness of various sub-networks g_i after training using LAT. Fig 1 shows the robustness with and without our LAT method for CIFAR-10 and MNIST datasets. As the plots show, our approach not only improves the robustness of f but also that of most of the sub-networks g_i. A detailed analysis analyzing the effect of the choice of the layer and the hyperparameter ω of LAT on the adversarial robustness of the model is shown in Section 5.

4.2 Latent Adversarial Attack (LA)

In this section, we seek to leverage the vulnerability of latent layers of a neural network to construct adversarial perturbations. In general, existing methods for computing adversarial perturbations such as FGSM (Goodfellow et al., 2015) and PGD (Madry et al., 2018) operate by directly perturbing the input layer to optimize the objective that promotes misclassification. In our approach, for given input example x and a sub-network $g_i(x)$, we compute adversarial perturbations $\Delta(x, g_i)$ on subnetworks (latent layers), where $i \in (1, 2, .., l)$. Here,

$$\begin{aligned} &\Delta(x, g_i) := \min_{\delta} ||\delta||_p \quad \text{where} \quad p \in \{2, \infty\} \\ &\text{s.t} \quad \arg\max(g_i(h_i(x) + \delta)) \neq \arg\max(g_i(h_i(x))) \end{aligned} \tag{8}$$

Subsequently, we optimize the following equation to obtain $\Delta(x, f)$ for LA :

$$\Delta(x, f) = \arg\min_{\mu} |h(x + \mu) - (h(x) + \Delta(x, g_i)| \tag{9}$$

We repeat the above two optimization steps iteratively to obtain our adversarial perturbation. A pseudocode-level description of the proposed algorithm (LA) is given in Algorithm 2.

In order to study the performance of LA, we use PGD adversarial perturbation as a baseline attack. The results are calculated with the constraint on the maximum amount of per-pixel perturbation as 0.3 for MNIST dataset and $8.0/256.0$ for CIFAR-10 and CIFAR-100. For MNIST and CIFAR-100, our LA achieves adversarial accuracies of 90.78% and 22.87% respectively, whereas PGD (100 steps) and PGD (10 steps) obtains adversarial accuracies of 92.52% and 23.01% respectively. In the case of CIFAR-10, LA achieves an adversarial accuracy of 47.46% and PGD (10 steps) obtains adversarial accuracy of 47.41. The represented LA attacks are from the best layers, i.e., g_1 for

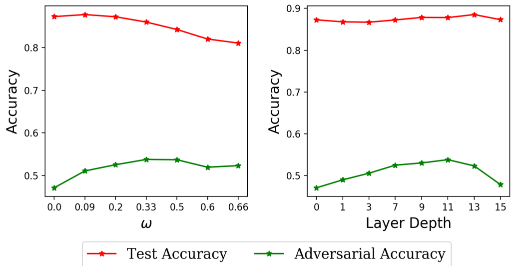

Figure 2: Plot showing effect of ω and layer depth on the adversarial and test accuracy of f

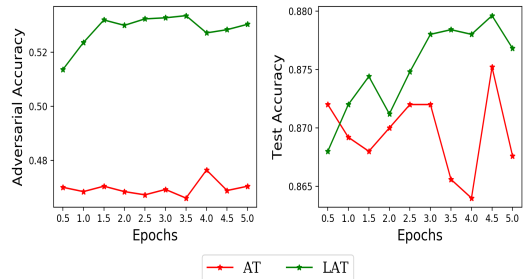

Figure 3: Progress of Adversarial and Test Accuracy for LAT and AT over 5 epochs of training

MNIST, CIFAR-100 and g_2 for CIFAR-10. In general, we obtain better or comparable adversarial accuracy when compared to the PGD attack.

Algorithm 2: Proposed algorithm for the construction of adversarial perturbation

begin

Input: Neural network model f, sub-network g_m, step-size for latent layer α_l, step-size for input layer α_x, intermediate iteration steps p, global iteration steps - k, input example x, adversarial perturbation generation technique for g_m

Output: Adversarial example $x^1 = x$

for $i \in 1, 2, ..., k$ **do**

$l^1 = g_m(x^i)$

for $j \in 1, 2, ..., p$ **do**

$$l^{j+1} = Proj_{l+S}\ (l^j + \alpha_l\ sign(\nabla_{g_m(x)} J(\theta, x, y)))$$

end

$x^1_{adv} = x^i$

for $j \in 1, 2, ..., p$ **do**

$$x^{j+1}_{adv} = Proj_{x_{adv}+S}(x^j_{adv} - \alpha_x sign(\nabla_x |g_m(x) - l^p|))$$

end

$x^i = x^p_{adv}$

end

return x^k.

end

5 Discussion and Ablation Studies

To gain an understanding of LAT, we perform various experiments and analyze the findings in this section. We choose CIFAR-10 as the primary dataset for all the following experiments.

Effect of layer depth and hyperparameter ω in LAT - We fix the value of ω to the best performing value of 0.2 and train the model using LAT for different latent layers of the network. The right plot in Fig 2 shows the influence of the layer depth in the performance of the model. It can be observed from the plot, that the robustness of f increases with increasing layer depth, but the trend reverses for the later layers. This observation can be explained from the plot in Fig 1, where the robustness of g_i decreases with increasing layer depth i, except for the last few layers.

For observing the influence of hyperparameter ω, we fix the layer depth to $11(g_{11})$ as it was the best performing layer for CIFAR-10 and we show LAT result for different values of ω. This hyperparameter ω controls the ratio of weight assigned to the classification loss corresponding to adversarial examples for g_{11} and the classification loss corresponding to adversarial examples for f. The left plot in Fig 2 shows the result of this experiment. We find that the robustness of f increases with

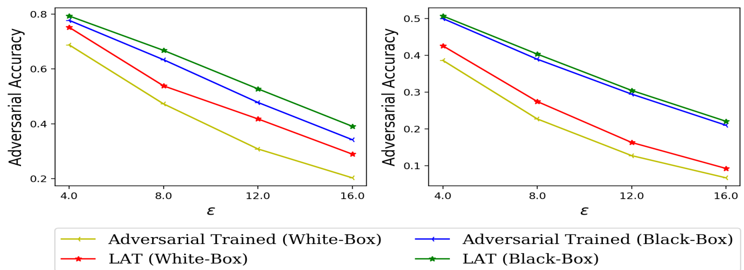

Figure 4: White-Box and Black-Box Adversarial accuracy on various ϵ for different models on CIFAR-10(left) and CIFAR-100(right) dataset

increasing ω. However, the adversarial accuracy does start to saturate after a certain value. The performance of test accuracy also starts to suffer beyond this point.

Black-box and white-box attack robustness for various ϵ - We test the black box and white-box adversarial robustness of our best performing model for the CIFAR-10 and CIFAR-100 datasets over various ϵ values. To generate a black box adversarial perturbation, we train a separate model using adversarial training and generate the perturbation for this model. The generated perturbation are evaluated on our (LAT) model. Fig 4 shows the attack success accuracy. As can be seen, the LAT trained model achieves better adversarial robustness for both the black box and white-box attacks over a range of ϵ values relative to the adversarially trained model. We also evaluated adversarial robustness against banit optimization based black box attack (Ilyas et al.) and SPSA (Uesato et al., 2018) with $\epsilon = 12/255$. LAT model has adversarial accuracy of 35.37% and 51.55% whereas baseline model (AT) has adversarial accuracy of 25.64% and 46.65% respectively. Thus even on stronger black box attacks our model outperforms the baseline. In Fig. 3, we show the adversarial and test accuracy performance of the LAT and AT methods with the progress of training.

Different attack methods for LAT - Rather than crafting a l_∞ bound PGD adversarial attack, we craft a l_2 bound PGD attack as well as use the FGSM attack to perturb the latent layers in LAT. For the l_2 bound PGD attack, doing LAT for 2.5 epochs achieved an adversarial and test accuracy of **88.02%** and **53.46%** respectively. Using FGSM to perform LAT did not lead to significant changes as the model achieved 48.83% and 87.26% adversarial and test accuracies respectively. All the results correspond to the g_{11} sub-network.

Random layer selection in LAT - Previous experiments over LAT corresponds to selection of a single sub-network g_i and adversarially training it. We conduct an experiment, where at each training step of LAT, we randomly choose one of the [g_5, g_7, g_9, g_{11}] sub-networks to perform adversarial training. The model performs comparably, achieving a test and adversarial accuracy of 87.31% and 53.50% respectively.

6 Conclusion

We observe that deep neural network models trained via adversarial training have sub-networks vulnerable to adversarial perturbation. We described a latent adversarial training (LAT) technique aimed at improving the adversarial robustness of the sub-networks. We verified that using LAT significantly improved the adversarial robustness of the overall model for several different datasets along with an increment in test accuracy. We performed several experiments to analyze the effect of depth on LAT and showed higher robustness to Black-Box attacks. We proposed Latent Attack (LA), an adversarial attack algorithm that exploits the adversarial vulnerability of latent layer to construct adversarial examples. Our results show that the proposed methods that harness the effectiveness of latent layers in a neural network beat state-of-the-art in defense methods, and offer a significant pathway for new developments in adversarial machine learning.

REFERENCES

Naveed Akhtar and Ajmal Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *IEEE Access*, 2018.

David Alvarez-Melis and Tommi S Jaakkola. On the robustness of interpretability methods. *arXiv preprint arXiv:1806.08049*, 2018.

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *ICML*, 2018.

Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *AISec*. ACM, 2017a.

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

Nicholas Carlini, Guy Katz, Clark Barrett, and David L Dill. Provably minimally-distorted adversarial examples. *arXiv preprint arXiv:1709.10207*, 2017.

Laurens van der Maaten Alan Yuille Kaiming He Cihang Xie, Yuxin Wu. Feature denoising for improving adversarial robustness. *arXiv preprint arXiv:1812.03411*, 2018.

Moustapha Cisse, Piotr Bojanowski, Edouard Grave, Yann Dauphin, and Nicolas Usunier. Parseval networks: Improving robustness to adversarial examples. In *ICML*, 2017.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *ICLR*, 2015.

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

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017.

Andrew Ilyas, Logan Engstrom, and Aleksander Madry. Prior convictions: Black-box adversarial attacks with bandits and priors. *ICLR*.

Harini Kannan, Alexey Kurakin, and Ian J. Goodfellow. Adversarial logit pairing. *NIPS*, 2018.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10. URL `http://www.cs.toronto.edu/kriz/cifar.html`, 2010.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

Yan Lecun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, and L.D. Jackel. Backpropagation applied to handwritten zip code recognition, 1989.

Anish Athalye Logan Engstrom, Andrew Ilyas. Evaluating and understanding the robustness of adversarial logit pairing. *NeurIPS SECML*, 2018.

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

MadryLab. cifar10_challenge. `https://github.com/MadryLab/cifar10_challenge`, 2017a.

MadryLab. mnist_challenge. `https://github.com/MadryLab/mnist_challenge`, 2017b.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. *CVPR*, 2017.

Aditi Raghunathan, Jacob Steinhardt, and Percy Liang. Certified defenses against adversarial examples. *ICLR*, 2018.

Andras Rozsa, Ethan M Rudd, and Terrance E Boult. Adversarial diversity and hard positive generation. In *CVPR Workshop*, 2016.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015.

Sara Sabour, Yanshuai Cao, Fartash Faghri, and David J. Fleet. Adversarial manipulation of deep representations. *ICLR*, 2016.

Pouya Samangouei, Maya Kabkab, and Rama Chellappa. Defense-gan: Protecting classifiers against adversarial attacks using generative models. *CoRR*, abs/1805.06605, 2018. URL `http://arxiv.org/abs/1805.06605`.

Swami Sankaranarayanan, Arpit Jain, Rama Chellappa, and Ser Nam Lim. Regularizing deep networks using efficient layerwise adversarial training. *AAAI*, 2018.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *ICLR*, 2014.

Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian J. Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *ICLR*, 2018.

Yusuke Tsuzuku, Issei Sato, and Masashi Sugiyama. Lipschitz-margin training: Scalable certification of perturbation invariance for deep neural networks. *NeurIPS*, 2018.

Jonathan Uesato, Brendan O'Donoghue, Aaron van den Oord, and Pushmeet Kohli. Adversarial risk and the dangers of evaluating against weak attacks. *arXiv preprint arXiv:1802.05666*, 2018.

Tsui-Wei Weng, Huan Zhang, Hongge Chen, Zhao Song, Cho-Jui Hsieh, Duane Boning, Inderjit S Dhillon, and Luca Daniel. Towards fast computation of certified robustness for relu networks. *ICML*, 2018.

Eric Wong and J Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. *ICML*, 2018.

Chaowei Xiao, Jun-Yan Zhu, Bo Li, Warren He, Mingyan Liu, and Dawn Song. Spatially transformed adversarial examples. *ICLR*, 2018.

Xiaoyong Yuan, Pan He, Qile Zhu, and Xiaolin Li. Adversarial examples: Attacks and defenses for deep learning. *IEEE transactions on neural networks and learning systems*, 2019.