

Harnessing the Vulnerability of Latent Layers in Adversarially Trained Models

Nupur Kumari^{1*}, Mayank Singh^{1*}, Abhishek Sinha^{1*}, Harshitha Machiraju², Balaji Krishnamurthy¹ and Vineeth N Balasubramanian²

¹Adobe Inc,Noida

²IIT Hyderabad

{ nupkumar,msingh,abhsinha,kbalaji } @adobe.com

{ 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 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 constructing adversarial examples. LAT results in a 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, SVHN and Restricted ImageNet datasets.

1 Introduction

Deep Neural Networks have achieved state of the art performance in several Computer Vision tasks [He *et al.*, 2016; Krizhevsky *et al.*, 2012]. However, recently it has 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 and Mian, 2018].

Since the advent of adversarial perturbations, there has been extensive work in the area of crafting new adversarial attacks [Madry *et al.*, 2018; Moosavi-Dezfooli *et al.*, 2017;

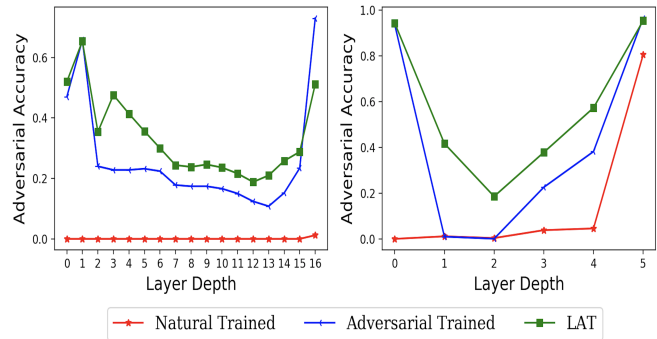


Figure 1: Adversarial accuracy of latent layers for different models on CIFAR-10 and MNIST

Carlini and Wagner, 2017b]. At the same time, several methods have been proposed to protect models from these attacks (adversarial defense)[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 and 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 and Kolter, 2018]. Also some work has been done by using the Lipschitz constant as a measure of robustness and improving upon it [Szegedy *et al.*, 2014; Cisse *et al.*, 2017; Tsuzuku *et al.*, 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 CIFAR-10, CIFAR-100, 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 an 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 further fine-tuning the adversarially trained model. We find that improving the robustness of the

* Authors contributed equally

models at the latent layer boosts the adversarial accuracy of the entire model. We observe that LAT improves the adversarial robustness by ($\sim 4 - 6\%$) and test accuracy by ($\sim 1\%$) for CIFAR-10 and CIFAR-100 datasets.

Our main contributions in this paper 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**) technique that significantly increases the robustness of existing state of the art adversarially trained models [Madry *et al.*, 2018] for MNIST, CIFAR-10, CIFAR-100, SVHN and Restricted ImageNet datasets.
- We propose Latent Attack (**LA**), a new l_∞ adversarial attack that is comparable in terms of performance to PGD on multiple datasets. The attack exploits the non-robustness of in-between layers of existing robust models to construct adversarial perturbations.

The rest of the paper is structured as follows: In Section 2, we review various adversarial attack and defense methodologies. In Section 3 and 4.1, we analyze the vulnerability of latent layers in robust models and introduce our proposed training technique of Latent Adversarial Training (LAT). In Section 4.2 we describe our adversarial attack algorithm Latent Attack (LA). Further, we do some ablation studies to understand the effect of the choice of the layer on LAT and LA attack in Section 5.

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,

$$\Delta(x, f) := \min_{\delta} \|\delta\|_p \quad (1)$$

s.t $\arg \max(f(x + \delta; \theta)) \neq \arg \max(f(x; \theta))$

It can be expressed as an optimization problem as:

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

In general, the magnitude of adversarial perturbation is constrained by a p norm where $p \in \{0, 2, \infty\}$ to ensure that the perturbed example is close to the original sample. Various other constraints for closeness and visual similarity [Xiao *et al.*, 2018] have also been proposed for the construction of adversarial perturbation .

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. We briefly describe PGD [Madry *et al.*, 2018] adversarial attack which we use as a baseline in our paper.

Projected Gradient Descent (PGD) attack. Projected gradient descent [Madry *et al.*, 2018] is an iterative variant of Fast Gradient Sign Method (FGSM) [Goodfellow *et al.*, 2015]. Adversarial examples are constructed by iteratively applying FGSM and projecting the perturbed output to a valid constrained space S . The attack formulation is as follows:

$$x^{i+1} = Proj_{x+S} (x^i + \alpha \text{sign}(\nabla_x J(\theta, x^i, y))) \quad (2)$$

where x^{i+1} denotes the perturbed sample at $(i+1)_{th}$ iteration.

While there has been extensive work in this area [Yuan *et al.*, 2019; Akhtar and Mian, 2018], we primarily focus our attention towards attacks which utilizes latent layer representation. [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.

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 [Tszuku *et al.*, 2018; Cisse *et al.*, 2017]. There have also been several methods which turn to GAN's [Samangouei *et al.*, 2018] for classifying the input as an adversary. However, these defense techniques were shown to be ineffective to adaptive adversarial attacks [Athalye *et al.*, 2018; Logan Engstrom, 2018]. Hence we turn to adversarial training which [Goodfellow *et al.*, 2015; Madry *et al.*, 2018; Kannan *et al.*, 2018] is a defense technique that injects adversarial examples in the training batch at every step of the training. Adversarial training constitutes the current state-of-the-art in adversarial robustness against white-box attacks. For a comprehensive review of the work done in the area of adversarial examples, please refer [Yuan *et al.*, 2019; Akhtar and Mian, 2018].

In our current work, we try to enhance the robustness of each latent layer, and hence increasing the robustness of the network as a whole. Previous works in this area include [Sankaranarayanan *et al.*, 2018; Cihang Xie, 2019]. However, our paper is different from them on the following counts:

- [Cihang Xie, 2019] observes that the adversarial perturbation on image leads 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 to achieve higher robustness.
- [Sankaranarayanan *et al.*, 2018] proposes an approach to regularize deep neural networks by perturbing intermediate layer activation. 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 focuses on 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 attack [Madry *et al.*, 2018].

3 Robustness of Latent Layers

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

$$f(x) = f_l(f_{l-1}(\dots(f_2(f_1(x; W_1, b_1); W_2, b_2))\dots); W_l, b_l) \quad (3)$$

Here f_i denotes the function mapping layer $i - 1$ to layer i with weights W_i and bias b_i respectively. From Eq. 3, it is evident that $f(x)$ can be written as a composition of two functions:

$$f(x) = g_i \circ h_i(x) \mid 0 \leq i \leq l - 1$$

where $f_0 = I$ and $h_i = f_i \circ f_{i-1} \dots \circ f_1 \circ f_0$ (4)

$$g_i = f_l \circ f_{l-1} \dots \circ f_{i+1}$$

We can study the behavior of $f(x)$ at a slightly perturbed input by inspecting its Lipschitz constant, which is defined by a constant L_f such that Eq. 5 holds for all ν .

$$\|f(x + \nu) - f(x)\| \leq L_f \|\nu\| \quad (5)$$

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

$$L_f \leq L_g * L_h \quad (6)$$

So having robust sub-networks can result in higher adversarial robustness for the whole network f . But the converse need not be true.

For each of the latent layers i , we calculate an upper bound for the magnitude of perturbation(ϵ_i) by observing the perturbation induced in latent layer for adversarial examples x^{adv} . For obtaining a sensible bound of the perturbation for the sub-network $g_i(x)$, the following formula is used :

$$\epsilon_i \propto Mean_{x \in test} \|h_i(x) - h_i(x^{adv})\|_\infty \quad (7)$$

Using this we compute the adversarial robustness of sub-networks $\{g_i \mid 1 \leq i \leq l - 1\}$ using PGD attack as shown in Fig 1.

We now, briefly describe the network architecture used for each dataset. ¹

- **MNIST**[Lecun *et al.*, 1989]: We use the network architecture as described in [Madry *et al.*, 2018]. The natural and adversarial trained model achieves a test accuracy of 99.17% and 98.4% respectively.
- **CIFAR-10**[Krizhevsky *et al.*, 2010]: We use the network architecture as in [Madry *et al.*, 2018]. The natural and adversarial trained model achieves a test accuracy of 95.01% and 87.25% respectively.

¹Code available at: https://github.com/msingh27/LAT_adversarial_robustness

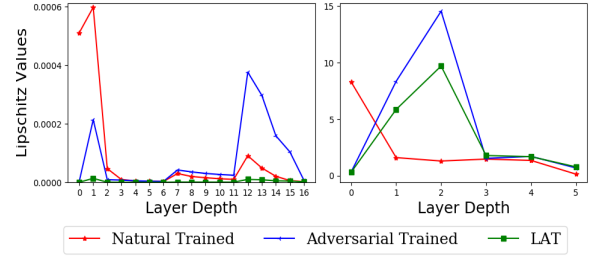


Figure 2: Lipschitz value of sub-networks with varying depth for different models on CIFAR-10 and MNIST

- **CIFAR-100**[Krizhevsky *et al.*, 2010]: We use the same network architecture as used for CIFAR-10 with the modification at the logit layer so that it can handle the number of classes in CIFAR-100. The natural and adversarial trained model achieves a test accuracy of 78.07% and 60.38% respectively.
- **SVHN**[Netzer *et al.*, 2011]: We use the same network architecture as used for CIFAR-10. The adversarial trained model achieves a test accuracy of 91.13%.
- **Restricted Imagenet**[Tsipras *et al.*, 2019]: The dataset consists of a subset of imagenet classes which have been grouped into 9 different classes. The model achieves a test accuracy of 91.65%.

For adversarial training, the examples are constructed using PGD adversarial perturbations[Madry *et al.*, 2018]. Also, we refer 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 adversarially robust model.

We observe that for adversarially trained models, the adversarial accuracies of the sub-networks g_i are relatively less than that of the whole network f as shown in Fig 1 and 3. The trend is consistent across all the different datasets. Note that layer depth, i.e. i is relative in all the experiments and the sampled layers are distributed evenly across the model. Also, in all tests the deepest layer tested is the layer just before the logit layer. Layer 0 corresponds to the input layer of f .

Fig 1 and 3 reveal that the sub-networks of an adversarially trained model are still vulnerable to adversarial perturbations. In general, it reduces with increasing depth. Though, a peculiar trend to observe is the increased robustness in the later layers of the network. The plots indicate that there is a scope of 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 do further analysis from the viewpoint of Lipschitz constant of the sub-networks. Since we are only concerned with the behavior of the function in the small neighborhood of input samples, we compute Lipschitz constant of the whole network f and sub-networks g_i using the local neighborhood of

input samples i.e.

$$L_f(x_i) = \max_{x_j \in B_\epsilon(x_i)} \frac{\|f(x_j) - f(x_i)\|}{\|x_j - x_i\|} \quad (8)$$

where $B_\epsilon(x_i)$ denotes the ϵ neighbourhood of x_i . For computational reasons, inspired by [Alvarez-Melis and Jaakkola, 2018], we approximate $B_\epsilon(x_i)$ by adding noise to x_i with epsilon as given in Eq. 7. We report the value averaged over complete test data for different datasets and models in Fig. 2. 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 appropriate bound as defined in Eq. 7.

We are using LAT as a fine-tuning technique which operates on a adversarially trained model to improve its adversarial and test accuracy further. We observe that performing only a few epochs (~ 5) of LAT on the adversarially trained model results in a significant improvement over adversarial accuracy of the model. Algorithm 1 describes our LAT training technique.

To test the efficacy of LAT, we perform experiments over CIFAR-10, CIFAR-100, SVHN, Rest. Imagenet and MNIST datasets. For fairness, we also compare our approach (LAT) against two baseline fine-tuning techniques.

- Adversarial Training (AT) [Madry *et al.*, 2018]
- Feature Noise Training (FNT) using algorithm 1 with gaussian noise to perturb the latent layer i .

Table 1 reports the adversarial accuracy corresponding to LAT and baseline fine-tuning methods over the different datasets. PGD Baseline corresponds to 10 steps for CIFAR-10, CIFAR-100 and SVHN, 40 steps for MNIST and 8 steps of PGD attack for Restricted Imagenet. We perform 2 epochs of fine-tuning for MNIST, CIFAR-10, Rest. Imagenet, 1 epoch for SVHN and 5 epochs for CIFAR-100 using the different techniques. The results are calculated with the constraint on the maximum amount of per-pixel perturbation as 0.3/1.0 for MNIST dataset and 8.0/255.0 for CIFAR-10, CIFAR-100, Restricted ImageNet and SVHN.

Dataset	Fine-tune Technique	Adversarial Accuracy		
		PGD Baseline	PGD (100 step)	Test Acc.
CIFAR-10	AT	47.12 %	46.19 %	87.27 %
	FNT	46.99 %	46.41 %	87.31 %
	LAT	53.84 %	53.04 %	87.80 %
CIFAR-100	AT	22.72 %	22.21 %	60.38 %
	FNT	22.44 %	21.86 %	60.27 %
	LAT	27.03 %	26.41 %	60.94 %
SVHN	AT	54.58 %	53.52 %	91.88 %
	FNT	54.69 %	53.96 %	92.45 %
	LAT	60.23 %	59.97 %	91.65 %
Rest. ImageNet	AT	17.52 %	16.04 %	91.83 %
	FNT	18.81 %	17.32 %	91.59 %
	LAT	22.00 %	20.11 %	89.86 %
MNIST	AT	93.75 %	92.92 %	98.40 %
	FNT	93.59 %	92.16 %	98.28 %
	LAT	94.21 %	93.31 %	98.38 %

Table 1: Adversarial accuracy for different datasets after fine-tuning using different methods

Algorithm 1 Algorithm for improving the 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 $i \in 1, 2, \dots, k$ **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. 7.
 Compute adversarial perturbation $\Delta h_m(X(i))$ with perturbation amount ϵ
 Compute the gradients $J_{latentAdv} = J(\theta, h_m(X(i)) + \Delta h_m(X(i)), Y(i))$

$J(\theta, X(i), Y(i)) = \omega * J_{adv} + (1 - \omega) * (J_{latentAdv})$

$\theta \rightarrow \theta - \eta * J(\theta, X(i), Y(i))$

end

return fine-tuned model.

end

The results in the Table 1 correspond to the best performing layers ². As can be seen from the table, that only after 2 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. A similar trend is visible for SVHN and CIFAR-100 datasets where LAT improves the adversarial accuracy by 8% and 4% respectively, as well as the test accuracy for CIFAR-100 by 0.6%. Table 1 also reveals that the two baseline methods do not lead

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

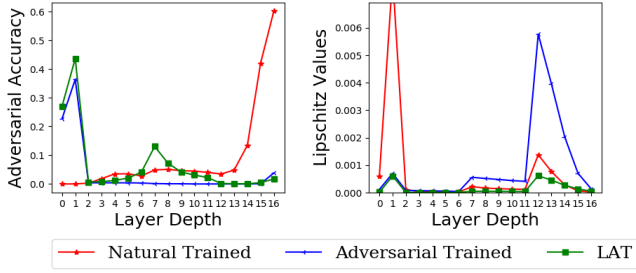


Figure 3: Adversarial accuracy and Lipschitz values with varying depth for different models on CIFAR-100

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 does not lead to significant improvements ($\sim 0.46\%$).

To analyze the effect of LAT on latent layers, we compute the robustness of various sub-networks g_i of f after training using LAT. Fig 1 shows the robustness of different sub-network g_i with and without our LAT method for CIFAR-10 and MNIST datasets. Figure 3 contains the results for CIFAR-100 dataset. 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 adversarial perturbation calculation methods like 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 first calculate adversarial perturbation $\Delta(x, g_i)$ constrained by appropriate bounds where $i \in (1, 2, \dots, l)$. Here,

$$\Delta(x, g_i) := \min_{\delta} \|\delta\|_p \text{ where } p \in \{2, \infty\} \tag{9}$$

$$\text{s.t } \arg \max(g_i(h_i(x) + \delta)) \neq \arg \max(g_i(h_i(x)))$$

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

$$\Delta(x, f) = \arg \min_{\mu} |h(x + \mu) - (h(x) + \Delta(x, g_i))| \tag{10}$$

We repeat the above two optimization steps iteratively to obtain our adversarial perturbation.

For the comparison of the performance of LA, we use PGD adversarial perturbation as a baseline attack. In general, we obtain better or comparable adversarial accuracy when compared to PGD attack. We use the same configuration for ϵ as in LAT. For MNIST and CIFAR-100, our LA achieves an adversarial accuracy of 90.78% and 22.87% respectively whereas PGD(100 steps) and PGD(10 steps) obtains adversarial accuracy of 92.52% and 23.01% respectively. In the

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  $i = 1, 2, \dots, k$  do
     $l^1 = g_m(x^i)$ 
    for  $j \in 1, 2, \dots, p$  do
         $l^{j+1} = Proj_{l+S} (l^j + \alpha_l \text{sign}(\nabla_{g_m(x)} J(\theta, x, y)))$ 
    end
     $x_{adv}^1 = x^i$ 
    for  $j \in 1, 2, \dots, p$  do
         $x_{adv}^{j+1} = Proj_{x_{adv}+S} (x_{adv}^j - \alpha_x \text{sign}(\nabla_x |g_m(x) - l^p|))$ 
    end
     $x^i = x_{adv}^p$ 
end
return  $x^k$ 

```

end

case of CIFAR-10 dataset, LA achieves 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 MNIST, CIFAR-100 and g_2 for CIFAR-10.

Some of the adversarial examples generated using LA is illustrated in Fig 5. The pseudo code of the proposed algorithm(LA)is given in Algo 2.

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 in LAT. We fix the value of ω to the best performing value of 0.2 and fine-tune the model using LAT for different latent layers of the network. The left plot in Fig 4 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.

Effect of hyperparameter ω in LAT. We fix the layer depth to 11(g_{11}) as it was the best performing layer for CIFAR-10 and we perform LAT 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 right plot in Fig 4 shows the result of this experiment. We find that the robustness of f increases with 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.

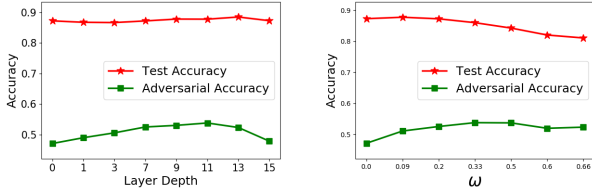


Figure 4: Plot showing effect of layer depth and ω on the adversarial and test accuracies of $f(x)$ on CIFAR-10

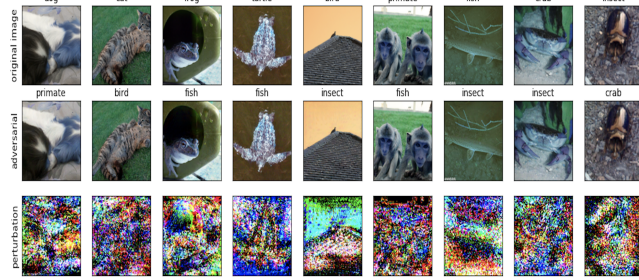


Figure 5: Adversarial images of Restricted ImageNet constructed using Latent Adversarial Attack (LA)

Black-box and white-box attack robustness. We test the black box and white-box adversarial robustness of LAT fine-tuned model for the CIFAR-10 dataset over various ϵ values. For evaluation in black box setting, we perform transfer attack from a secret adversarially trained model, bandit black box attack [Ilyas *et al.*, 2019] and SPSA [Uesato *et al.*, 2018]. Figure 7 shows the adversarial accuracy. As it can be seen, the LAT trained model achieves higher adversarial robustness for both the black box and white-box attacks over a range of ϵ values when compared against baseline AT model. We also observe that the adversarial perturbations transfers better ($\sim 1\%$) from LAT model than AT models.

Performance of LAT with training steps. Figure 6 plots the variation of test and adversarial accuracy while fine-tuning using the LAT and AT techniques.

Different attack methods used for LAT. Rather than using a l_∞ bound PGD adversarial attack, we also explored using a l_2 bound PGD attack and FGSM attack to perturb the latent layers in LAT. By using l_2 bound PGD attack in LAT for 2.5 epochs, the model achieves an adversarial and test accuracy of **88.02%** and **53.46%** respectively. Using FGSM to perform LAT did not lead to improvement as the model achieves 48.83% and 87.26% adversarial and test accuracy respectively. The results are calculated by choosing the g_{11} sub-network.

Random layer selection in LAT. Previous experiments of LAT fine-tuning corresponds to selecting a single sub-network g_i and adversarially training it. We perform 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.

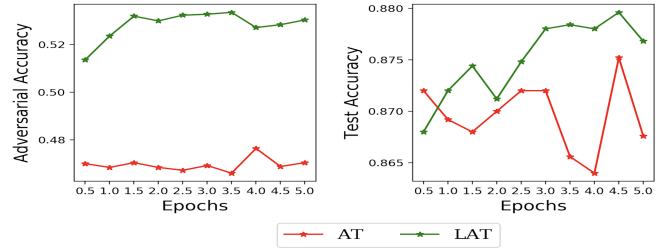


Figure 6: Progress of Adversarial and Test Accuracy for LAT and AT when fine-tuned for 5 epochs on CIFAR-10

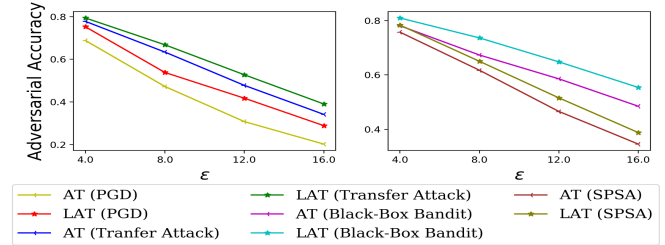


Figure 7: White-Box and Black-Box Adversarial accuracy on various ϵ on CIFAR-10

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

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

[Alvarez-Melis and Jaakkola, 2018] David Alvarez-Melis and Tommi S Jaakkola. On the robustness of interpretability methods. *ICML 2018 Workshop*, 2018.

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

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

- [Carlini and Wagner, 2017b] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, 2017.
- [Carlini *et al.*, 2017] Nicholas Carlini, Guy Katz, Clark Barrett, and David L Dill. Provably minimally-distorted adversarial examples. *arXiv preprint arXiv:1709.10207*, 2017.
- [Cihang Xie, 2019] Laurens van der Maaten Alan Yuille Kaiming He Cihang Xie, Yuxin Wu. Feature denoising for improving adversarial robustness. *CVPR*, 2019.
- [Cisse *et al.*, 2017] Moustapha Cisse, Piotr Bojanowski, Edouard Grave, Yann Dauphin, and Nicolas Usunier. Parseval networks: Improving robustness to adversarial examples. In *ICML*, 2017.
- [Goodfellow *et al.*, 2015] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *ICLR*, 2015.
- [He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.
- [Ilyas *et al.*, 2019] Andrew Ilyas, Logan Engstrom, and Aleksander Madry. Prior convictions: Black-box adversarial attacks with bandits and priors. *ICLR*, 2019.
- [Kannan *et al.*, 2018] Harini Kannan, Alexey Kurakin, and Ian J. Goodfellow. Adversarial logit pairing. *NIPS*, 2018.
- [Krizhevsky *et al.*, 2010] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10. URL <http://www.cs.toronto.edu/kriz/cifar.html>, 2010.
- [Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.
- [Lecun *et al.*, 1989] 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.
- [Logan Engstrom, 2018] Anish Athalye Logan Engstrom, Andrew Ilyas. Evaluating and understanding the robustness of adversarial logit pairing. *NeurIPS SECML*, 2018.
- [Madry *et al.*, 2018] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *ICLR*, 2018.
- [Moosavi-Dezfooli *et al.*, 2017] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. *CVPR*, 2017.
- [Netzer *et al.*, 2011] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. *NIPS Workshop*, 2011.
- [Raghunathan *et al.*, 2018] Aditi Raghunathan, Jacob Steinhardt, and Percy Liang. Certified defenses against adversarial examples. *ICLR*, 2018.
- [Russakovsky *et al.*, 2015] 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.
- [Sabour *et al.*, 2016] Sara Sabour, Yanshuai Cao, Fartash Faghri, and David J. Fleet. Adversarial manipulation of deep representations. *ICLR*, 2016.
- [Samangouei *et al.*, 2018] Pouya Samangouei, Maya Kabkab, and Rama Chellappa. Defense-gan: Protecting classifiers against adversarial attacks using generative models. *ICLR*, 2018.
- [Sankaranarayanan *et al.*, 2018] Swami Sankaranarayanan, Arpit Jain, Rama Chellappa, and Ser Nam Lim. Regularizing deep networks using efficient layerwise adversarial training. *AAAI*, 2018.
- [Szegedy *et al.*, 2014] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *ICLR*, 2014.
- [Tramèr *et al.*, 2018] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian J. Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *ICLR*, 2018.
- [Tsipras *et al.*, 2019] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. *ICLR*, 2019.
- [Tsuzuku *et al.*, 2018] Yusuke Tsuzuku, Issei Sato, and Masashi Sugiyama. Lipschitz-margin training: Scalable certification of perturbation invariance for deep neural networks. *NeurIPS*, 2018.
- [Uesato *et al.*, 2018] Jonathan Uesato, Brendan O'Donoghue, Aaron van den Oord, and Pushmeet Kohli. Adversarial risk and the dangers of evaluating against weak attacks. *ICML*, 2018.
- [Weng *et al.*, 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.
- [Wong and Kolter, 2018] Eric Wong and J Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. *ICML*, 2018.
- [Xiao *et al.*, 2018] Chaowei Xiao, Jun-Yan Zhu, Bo Li, Warren He, Mingyan Liu, and Dawn Song. Spatially transformed adversarial examples. *ICLR*, 2018.
- [Yuan *et al.*, 2019] 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.