Supplementary Material
Adversarial Defense Through Network Profiling Based Path Extraction

Yuxian Qiu[§] Jingwen Leng[§*] Cong Guo[§]
Quan Chen[§] Chao Li[§] Minyi Guo[§*] Yuhao Zhu[†]
[§]*Department of Computer Science and Engineering, Shanghai Jiao Tong University*
[†]*Department of Computer Science, University of Rochester*
{qiuyuxian,leng-jw,guocong}@sjtu.edu.cn,
{chen-quan,lichao,guo-my}@cs.sjtu.edu.cn, yzhu@rochester.edu

1. Effective Path Extraction for More Network Structures

For the sake of brevity, we only introduce effective path extraction for networks consist of convolutional layers and FC layers in the submitted paper. We further explain other common network structures' extraction methods in this section.

Skip Connection To handle skip connections in ResNet, we need to merge neurons contributed from two different layers. Consider a skip connection from layer l to layer $l+m$, then active neurons in layer l are collected from layer $l+1$ and $l+m$, denoted as $\mathcal{N}^l = \{n_k^l | k \in \tilde{K}^{l+1}\ or\ k \in \tilde{K}^{l+m}\}$, where $\tilde{K}^{l+1}$ and $\tilde{K}^{l+m}$ are the selected sets of weight indices in layer $l+1$ and $l+m$ respectively.

Pooling Layer Pooling layers can be treated as the special case of convolutional layers during extracting. For average pooling layer, we treat it as a convolutional layer with all weights equal to 1; for max pooling layer, we treat it as a convolutional layer that always picks rank-1 weight and input neuron pair when finding the minimum $\tilde{K}_p^l$.

2. Effective Path Visualization for CIFAR-100

To explore path specialization on more realistic dataset than LeNet, we study the similarity of per-class effective paths on CIFAR-100. Fig. 1 shows the class-wise path similarity of 15 classes in CIFAR-100, which are belonged to three different super classes. We show super classes vehicles 1, large natural outdoor scenes and flowers from left to right, each of which contains 5 basic classes. We can find that classes belong to the same super classes have higher similarity than classes from different super classes, which indicates that effective path discovers the class hierarchy without prior knowledge of super classes.

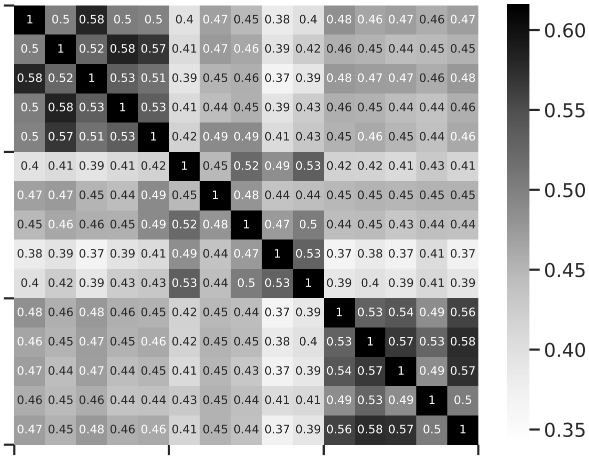


Figure 1: Class-wise path similarity for CIFAR-100.

3. Adversarial Samples Defense

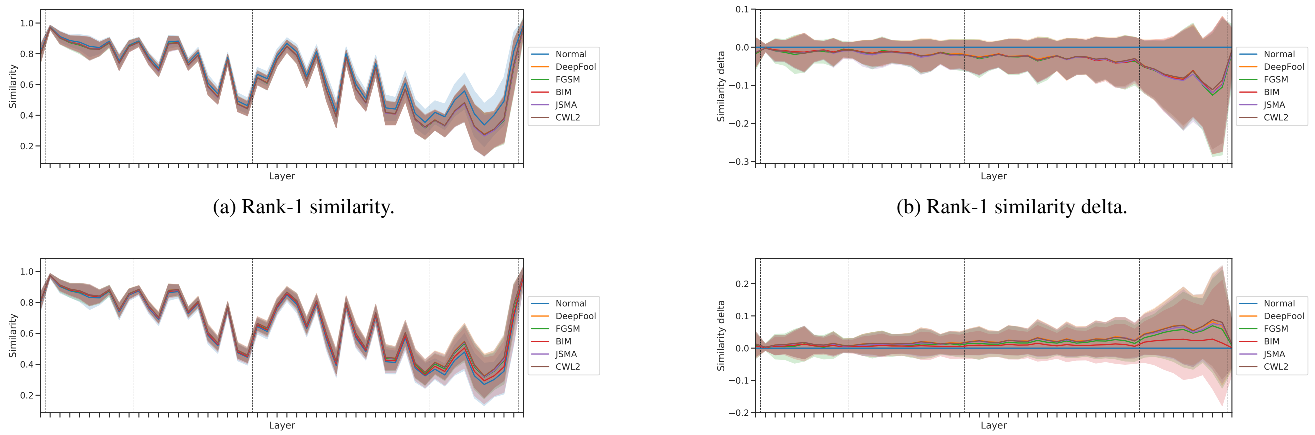
3.1. Adversarial Samples Similarity Analysis for ResNet-50

Per-layer similarity distribution for ResNet-50 on ImageNet is shown in Fig. 2. Similar to AlexNet, adversarial images lead to lower rank-1 similarity and higher rank-2 similarity compared with normal images. Furthermore, corresponding to AlexNet's FC layers, the largest similarity delta is also located in last several layers.

3.2. Weight-based Defense Model

For adversarial detection, we can use information from model weights as alternative of synapses. By calculating

[*] Jingwen Leng and Minyi Guo are co-corresponding authors of this paper



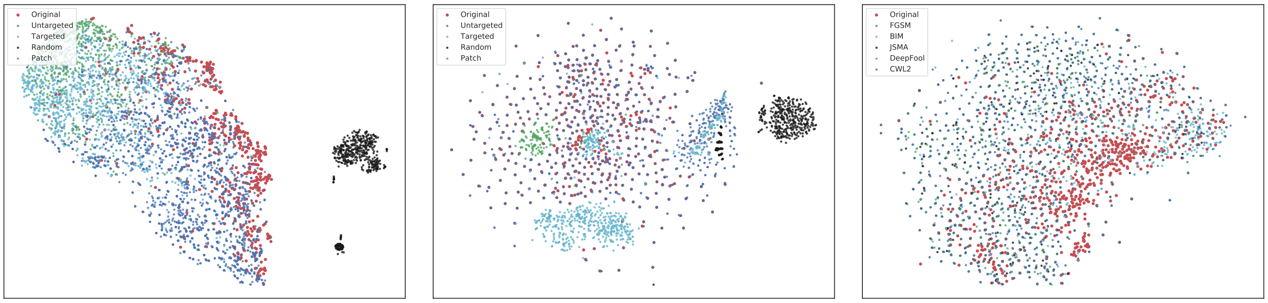
(a) Rank-1 similarity.

(b) Rank-1 similarity delta.

(c) Rank-2 similarity.

(d) Rank-2 similarity delta.

Figure 2: Distribution of per-layer similarity for ResNet-50 on ImageNet. Each line plot represents the mean of each kind of adversarial examples' similarity, with the same-color band around to show the standard deviation. The dashed lines indicate that down-sampling is performed in the next layer.



(a) Effective path similarity.

(b) CDRP.

(c) ResNet-50.

Figure 3: t-SNE 2D embedding of original images and adversarial images from different attacks. Each point stands for an image. The first two pictures show results on AlexNet, while the last one show effective path's result on ResNet-50. For brevity, FGSM, BIM, JSMA, DeepFool, and CWL2 are grouped into untargeted attacks. Targeted version of FGSM and CWL2 are also grouped into targeted attacks.

image-class path similarity from weights in effective path instead, i.e., let $J^l_{\mathcal{P}} = |\mathcal{W}^l \cap \tilde{\mathcal{W}}^l_p|/|\mathcal{W}^l|$ for layer l, we obtain weight-based joint similarity. The detection result using weight-based defense metric for AlexNet is shown in Fig. 4, which indicates that it achieves as high accuracy as the synapse-based metric.

3.3. Adversarial Sample Visualization

In this section, we use t-SNE to visualize effective path similarity and CDRP, which provide an intuitive way to show the adversarial sample detection ability of both methods. For consistency with our defense model, we use rank-1 and rank-2 effective path similarity as input features of t-SNE. Fig. 3a and Fig. 3b show the t-SNE 2D embedding of effective path similarity and CDRP on AlexNet respectively. For our method, random images are in two dissociative clusters. In the cluster contains all other images, original images are located in the edge of right side of the cluster, which is partly separable with other adversarial images. Adversarial images from untargeted attacks locate in the nearest area beside original examples, which is coincident with the fact that untargeted ones contain the smallest perturbations among these adversarial examples. On the other hand, adversarial images from targeted attacks and patch attack are far away from original images. All the above shows that effective path similarity catches the difference between normal images and adversarial images from multiple attacks. For CDRP, the location of original images and adversarial images from un-

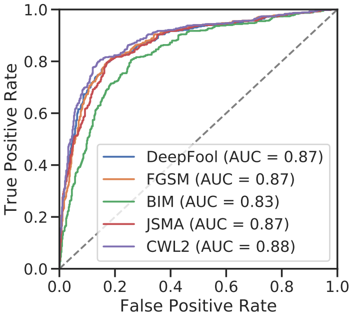


Figure 4: ROC for AlexNet on ImageNet with weight-based joint similarity.

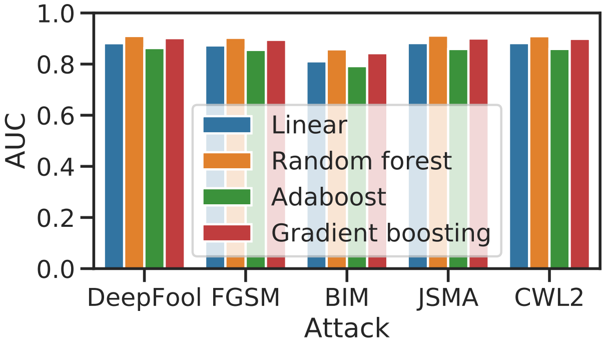
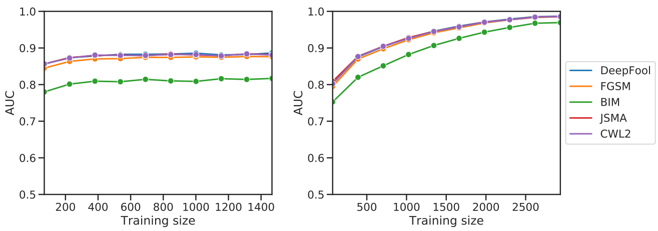


Figure 5: Detection accuracy comparison under different defense models.



(a) Linear model.

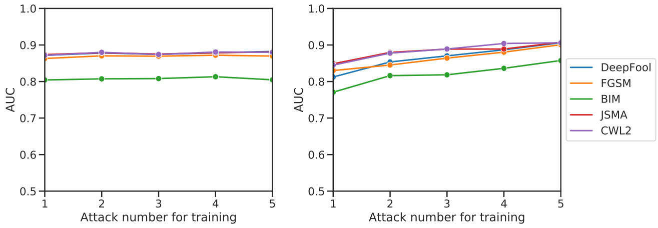
(b) Random forest.

Figure 6: Impact of training set size on the AUC.

targeted attacks are almost co-located, which indicates that it fails to catch the difference between them and leads to its low detection accuracy of untargeted attacks. Fig. 3c shows the t-SNE 2D embedding of effective path similarity on ResNet-50. The boundary between original images and adversarial images is fuzzier than that on AlexNet, but original images still have different distribution compared with adversarial images.

4. Evaluation on ResNet-50

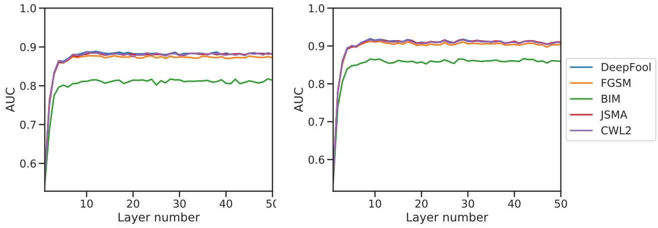
In this section, we show further evaluation results on ResNet-50 which is not included in the submitted paper.



(a) Linear model.

(b) Random forest.

Figure 7: Impact of attack number in the training set.



(a) Linear model.

(b) Random forest.

Figure 8: Effective path layer number impact on AUC.

4.1. Detection Accuracy

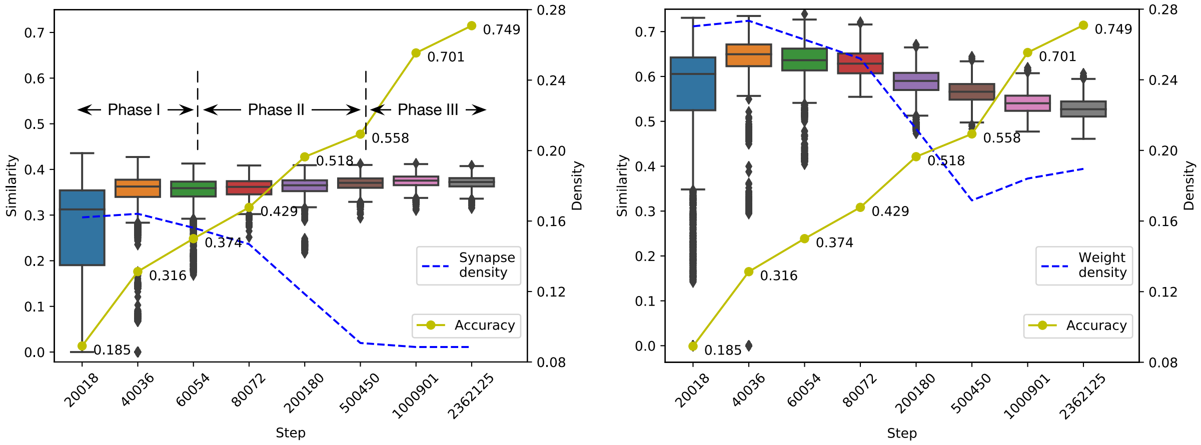
Fig. 5 shows the detection accuracy of effective path under different defense models. For all of the mentioned attacks, we find that random forest performs the best while the linear model performs worst among all models. However, the gap between random forest and linear model is very small, which indicates that effective path can achieve comparable detection performance using simple and highly interpretable way on deep and complex networks like ResNet-50.

4.2. Training Size

We choose the linear model and random forest model as representations to study how the size of the training set impacts the detection accuracy on ResNet-50. Fig. 6 shows the difference between the simple linear model and complex random forest model. For the linear model, the detection accuracy stabilizes with a small number of training samples (around 400 images). For the random forest model, the detector requires much larger training set (around 2500 images), meanwhile achieves much better detection accuracy, which indicates that random forest model can utilize more features in the effective path. Compared with AlexNet, both linear model and random forest model requires more training samples to be stable, which can be attributed to the fact that ResNet-50 has much more layers than AlexNet that provide useful features for detection.

4.3. Generalizability

Fig. 7 shows the experiment results of generalizability on ResNet-50 when adding the adversarial samples in the order of legend shown in the right. For both linear model and random forest model, our work generalizes well to unseen



(a) Synapses in the path.

(b) Weights in the path.

Figure 9: Effective path's density and class-wise path similarity in the training process.

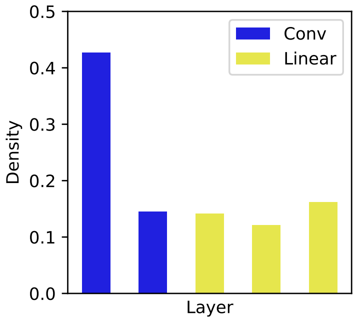


Figure 10: Per-layer density of effective paths in LeNet.

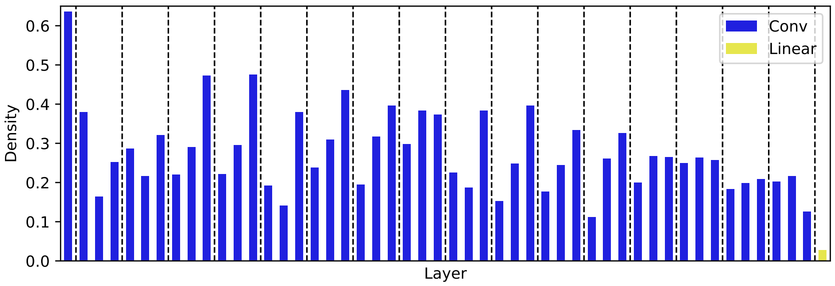


Figure 11: Per-layer density of effective paths in ResNet-50. Layers in ResNet-50 is organized into 3-layer bottleneck blocks, which is split by dashed lines.

attacks because effective path captures their common behavior. Compared with AlexNet, our method achieves the same level of generalizability on much more complex networks.

4.4. Layer Sensitivity

We study the layer number's impact on the adversarial sample detection accuracy on ResNet-50 and show the result in Fig. 8. For both linear model and random forest, we observe that the AUC performance for all attacks saturates within 10 layers, which is a small portion of the whole 50 layers. This layer sensitivity insight leads to the significant speedup of extraction by only extracting just enough layers instead of all layers, which is described in the submitted paper.

5. Further Study of Neural Network Interpretability

We further study how the training process and different network structure impacts the path specialization. In the next, we study how the training process and different network structures impact the path specialization.

5.1. Training Process

We study how the training process transforms a randomized network to the final state from the perspective of the effective path. Specifically, we extract the effective path for each class at different training stages. Through the analysis, we find that the training process contains three distinctive phases with different path's density and similarity trend, which share similar insights from the previous work using information bottleneck theory to explain training process [1].

Fig. 9 shows training process for ResNet. We choose different stages in training and show the class-wise path similarity in the form of box-plot, on top of which we also overlay the path density and prediction accuracy. In the first phase, the density of synapses and weights in the effective paths stays the same while their similarity increases. In the beginning, the network is in a randomized state and simply tries to memorize the input data.

In the second phase, the density of both synapses and weights decrease rapidly. The similarity of the synapses stays relatively the same while the similarity of weights decreases. In this phase, the network mainly performs compression, and the path specialization mainly manifests in the form of weights. In other words, the network tries to use class-specific features extracted by different convolutional filters to increase the specialization degree.

In the third phase, the synapse density stops to decrease but weight density starts to increase. Meanwhile, weight similarity continues to decrease. In this phase, the network compression stops and mainly relies on path specialization (via weight) to increase the prediction accuracy. The path specialization even causes the weight density increases a bit.

In summary, we find that the training process contains mainly three phases, the first two of which conforms to the memorization and compression phase identified by the prior work [1]. The second phase performs compression (less density) and path specialization (less similarity), while the third phase mainly includes the path specialization. After these phases, the network is transformed into a state with sparse and distinctive paths with great inference capability.

5.2. Network Structure

After establishing the effective path as a great indicator of the neural network's inference performance, we study how the network structure affects the effective path characteristics.

Fig. 10 shows the per-layer path density in LeNet. We observe the first convolutional layer has a much higher density compared to the following layers, which matches with the established knowledge that the shallow layers in a CNN extract high-level features that are shared by different classes.

CNN designers have found using a deeper network can increase the prediction accuracy to a certain degree. However, the accuracy stops to increase after a certain number of layers owing to the vanishing gradient in the training process. As such, the ResNet structure with skip connection was proposed to overcome this difficulty. Fig. 11 shows the per-layer path density for ResNet-50. Not only the first two layers still have higher density, but also layers before a skip connection also have high density. This suggests that skip connection helps not only the gradient propagation but also the effective paths formation. In the end, ResNet is able to converge and achieve great prediction performance.

References

[1] N. Wolchover. New Theory Cracks Open the Black Box of Deep Learning. `http://bit.ly/information_bottleneck`, 2017.