This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3024244, IEEE Access

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2020.DOI

Multi-class Triplet Loss with Gaussian Noise for Adversarial Robustness

BENJAMIN APPIAH[1], EDWARD Y. BAAGYERE[1], OWUSU-AGYEMANG KWABENA[1], ZHIGUANG QIN[1] AND MUHAMMED AMIN ABDULLAH.[1]

[1]School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, 610054 China

Corresponding author: Benjamin Appiah (e-mail: bappk@uestc.edu.cn).

ABSTRACT Deep Neural Networks (DNNs) classifiers performance degrades under adversarial attacks, such attacks are indistinguishably perturbed relative to the original data. Providing robustness to adversarial attacks is an important challenge in DNN training, which has led to extensive research. In this paper, we harden DNN classifiers under the adversarial attacks by regularizing their deep internal representation space with Multi-class Triplet regularization method. This method enables DNN classifier to learn a feature representation that detects similarities between adversarial and clean images and brings similar images close to their original class and pushes dissimilar images away from their false classes. This training process with our Multi-class Triplet regularization method in combination with Gaussian noise injection proves to be more robust in detecting adversarial attacks exceeding that of adversarial training on strong iterative attacks.

INDEX TERMS Adversarial Detection; Anomaly Detection; Adversarial Training; Metric Learning

I. INTRODUCTION

Deep Neural Networks (DNNs) have made significant progress in cyber-security [1], face detection [2] and objects classification [3]. This success is driven by the emergence of big data as a result of high traffic volume from unsatisfied online users. DNNs require less statistical and feature engineering from experts in order to be implemented. The intricacy of the data can be extracted with higher and more abstract level presentation from these raw traffic data. However, it has been proven that, the performance of DNNs degrades under adversarial attacks [4]–[7], where input examples are slightly modified with human-imperceptible. This limitation makes the application of DNNs in the field of safety and reliability critical applications a great concern [4].

A large body of work have been developed on improving the adversarial robustness of DNNs classifiers, such as feature squeezing, denoising, and encoding [8]–[11]. These methods perform earlier pre-processing on the input image to remove any adversarial perturbations. Despite all of these innovations, Adversarial Training [5], [12], one of the earliest defenses, still remains among the most effective and popular strategies. In its simplest form, adversarial training augment the training procedure with adversarial inputs produced by an adversarial attack thereby minimizes a loss function that measures performance of the model on both clean and adversarial data. Madry et al. [12] instantiated adversarial training using a strong iterative adversary and showed that their approach can train models which are highly robust against the strongest known adversarial attacks such C&W [4] and Project Gradient Decent (PGD) method [12] attacks. However, to train a network against strong attacks requires training on strongest adversarial examples such as the PGD or the Basic Iteration Method (BIM) [6]. Adversarial training on these strong attacks may be 10-100x more computational intensive. Furthermore, it is difficult to secure general robustness in this way, as there are many classes of adversarial examples that cause false classification, and model robustness to one class of adversarial examples does not bestow robustness to other classes [13].

A different approach to adversarial training on adversarial samples is to add randomization to the neural network [14], [15], making it harder for the attacker to evaluate gradients and to find the vulnerability of the network. He et al. [16] added Gaussian noise to the weights and activation of the neural network and showed improvement in their model training process. Generation of Gaussian noise is less computational and its introduction in Deep learning training processes has proven to boost classifiers stability [13], [17].

Recent studies by [5], [7], [8], [18], [19], on the latent representation space of DNNs classifiers under strong adversarial attacks suggested that adversarial attacks cause a false

classification as a result of the adversarial representations spreading across the false class distribution making it difficult to be distinguished. Motivated by these studies and following the framework of Adversarial Training with Gaussian noise injection, we propose to improve DNN classifiers robustness using metrics learning method. Our intuition is that by regularizing the DNNs classifiers representation space with Multi-class Triplet regularization term [20] will encourage the adversarial examples to approach their true classes and far way from their false classes hence improve robustness. Our main contribution in this paper is summarized as follows:

- We propose an adversarial learning method (MCT) that is susceptible against Black box and White box attacks.
- The MCT method is a combination of Muti-class Triplet loss with Gaussian noise, that minimizes the inter-real adversarial distance and maximizes intra-class distance.
- Training DNN classifiers against adversarial attacks with MCT method requires no expensive iterative adversarial examples generation which makes it an advantage for large datasets. Furthermore, our MCT method requires no modification to the model architecture and thus can improve the robustness on most off-the-shelf deep neural networks without additional overhead during training.
- Evaluation of MCT on MNIST [21], CIFAR-10 [22] and CIC-IDS2018 [23] datasets show that the MCT method classifiers adversarial attacks more accurately compared to that of adversarial training on strong iterative attacks.

The rest of this paper is organized as follows: In Section II, we review the related works on adversarial attack detection and prevention. Our methodology is presented in Section III. The application and the experiments are presented in Section IV and we presents our conclusions in Section V.

II. RELATED WORK

A. ADVERSARIAL ATTACKS

Deep learning models limitation to the adversarial attacks where first discovered by [24]. The work in [24] generated small perturbations on images for the image classification problem and fooled state of the-art neural networks with high probability. Goodfellow, et al, [5] proposed the Fast Gradient Sign method (FGSM) and also proposed a defense mechanism by training Deep learning model on the FGSM adversarial examples. Other effective adversarial attacks includes the Project Gradient Decent (PGD) method [12], C&W [4], Basic Iteration Method (BIM) [6], Jacobian-based Saliency Map Attack (JSMA) [7], HopSkipJump [27], and DeepFool [26] which are proposed to fool deep neural network.

B. COUNTERMEASURE

Countermeasure for these adversarial examples has been researched on. Madry, et al [12], demonstrated successful defense by training the model on PGD generated attack which randomly initialize an adversarial examples with the allowed norm ball before running iterative attack. Kannan, et al [28] proposed Adversarial Logit Pairing (ALP). The ALP method matches the logits from clean image and it's corresponding adversarial image and provide an extra regularization term for better representation of the data. However, the loss function adopted in this method is not scalable to untarget adversarial attacks [29].

Metric learning has also been introduced to counter the treat of adversarial attacks [30]. These technique add an additional constraints to the deep learning model by applying the Triplet loss term [31], [32] on the latent representations of the adversarial examples to the original loss function. However, Triplet loss has shown to suffer from slow convergence and poor local optima [20]. In this work, however, we adopt the Multi-class Triplet loss that allows joint comparison among $N-1$ negative classes, therefore, alleviates the limitations in the Triplet loss [20]. Contrarily to the works in [12], [29], [30] which train on expensive iterative attacks examples, a disadvantage for a large dataset, our model on the other hand trains on additive Gaussian noise.

III. METHODOLOGY

This section explains our Multi-class Triplet regularization method with Gaussian noise for adversarial training process. Our purposed method regularizes the DNN classifiers representation space with Multi-class Triplet loss function to learn a feature representation that detects adversarial and clean images similarity and bring these similar images close to their original class and push dissimilar images away from their false classes.

A. PROBLEM STATEMENT

We consider a classification task with data $x \in [0,1]^{W \times H}$ of dimension $W \times H$ and labels $y \in Z_k$ with k classes sampled from a distribution D. We identify a model with a hypothesis f from a space F on input image x, the model outputs class $f(x) \in R^k$. The loss function $L(.)$ is used to train the model $L((f(x), y); \theta)$, where θ is the network parameter to learn. For some target model $f \in F$ and inputs (x, y_{true}) the adversarial goal is to find an adversarial examples (perturb image) $x^{'}$ and x are "close" yet the model missclassified $x^{'}$.

B. MULTI-CLASS TRIPLET LOSS

Multi-class Triplet loss [20] is trained on one anchor x_a, one positive sample x_p and $\{x_i\}_{i=1}^{N-1}$ negative samples, where the x_a and $\{x_i\}_{i=1}^{N-1}$ are from different class and x_a and x_p are from the same class. This loss forces the network to generate an embedding where the distance between x_a and $\{x_i\}_{i=1}^{N-1}$ is larger than the distance between x_a and x_p. The standard Multi-class Triplet loss is defined as:

$$\begin{aligned} L_{MCT}(\{x_a, x_p, \{x_i\}_{i=1}^{N-1}\} : f) \\ = \sum_{i=1}^{N-1} log(1 + \sum_{j \neq 1} exp(dist + \alpha)), \\ dist = \triangle(f(x_a)f(x_i)) - \triangle(f(x_a)f(x_p)), \end{aligned} \tag{1}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3024244, IEEE Access

where N is the cardinality of the set of triplets used in the training process and $\triangle(f(x_i)f(x_j))$ represent the distance between x_i and x_j in the representation space. In this setting, we define the distance between x_i and x_j in a representation space as a cosine similarity distance defined as

$$\triangle(f(x_i)f(x_j)) = \frac{|x_i \cdot x_j|}{||x_i||||x_j||} \tag{2}$$

C. ADVERSARIAL TRAINING WITH MULTI-CLASS TRIPLET LOSS

Given a clean image x, we generate adversarial image $x^{'}$ by injecting uncorrelated Gaussian noise ε into the mini-batch x. We use uncorrelated Gaussian noise to simulate many types of adversarial images since adversarial examples are themselves considered to be noise [4]–[7], [34]–[36]. Specifically, for each mini-batch x, we generate new examples as shown in Eq. 3

$$x_k^{'} = x_k + \varepsilon_k, \varepsilon_k \sim N(0, \sigma_k^2), \sigma_k > 0 \tag{3}$$

where σ_k^2 represents the variance of the Gaussian noise at pixel k. Under our setting, we adopt uniform sample $\sigma_k = \sigma$ from an unbiased sample neighborhood of clean image x based on the optimization parameter σ^2. We train the model on these augmented images $x^{'}$ and clean image x under the joint supervision of Softmax Cross-entropy loss (L_{SCE}) and Multi-class Triplet loss (L_{Np}). The adversarial training objective is presented as,

$$\begin{aligned} L_{all} &= L_{SCE}(f(x^{'}), y) \\ &+ \alpha L_{MCT}\{f(x_a^{'}), f(x_p), f(x_i)\} \\ &+ \lambda L_{norm}, \\ L_{norm} &= ||f(x_p^{'}) - f(x_a)||_2 \end{aligned} \tag{4}$$

here α controls the strength of the training stability, and λ is the weight for the feature norm decay term, to reduce the L_2 norm of the feature. $x_a^{'}$ anchor examples is from the same class as positive examples x_p and negative examples x_i is from mini-batch which has different label or from different class to the anchor $x_a^{'}$. Algorithm 1, summarizes our adversarial training with Multi-class Triplet loss. The training process is different from the original Triplet loss training where all the components (anchor, positive, negative) in the triplets term are clean images [20], [31], [32], in this settings, we choose the anchor to serve as an adversarial image since the anchor contains more information about the decision boundary between the "true" class and the "false" class [31]. Using the representation similarity distance defined in III-B, we select negative examples as the nearest images to the anchor from a false class. As a result, our model is able to learn to enlarge the boundary between the adversarial examples and their closest negative examples from the other classes. Figure 1 shows our modified Multi-class Triplet loss for adversarial training.

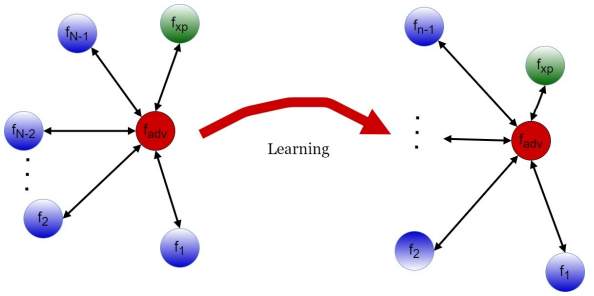


FIGURE 1. Illustration of the Multi-class Triplet Loss for adversarial robustness (MCT) with $(N-1)$ triplets. The anchor examples (red) and positive examples (green) belong to the same class. The negative examples (blue), from a different class, is the closest image to the anchor in feature space. MCT learns to pull the anchor and positive examples from the true class closer, and push the $(N-1)$ negative examples of false classes apart, based on their similarity to the anchor example.

Algorithm 1: Adversarial Training with Multi-Class Triplet loss

Input: training data x; training iteration T_t; learning rate lr; model parameter θ; mini-batch K for each iteration $X^t_{k\in\{1,..,K\}}$.
for $t = 1 : T_t$ **do**
Sample the anchor X_a from X.
Generate adversarial images $X_a^{'}$ from X_a using Equ 3.
Sample the positive X_p and X of the same class.
Sample the negative X_i from X with strategy mentioned in Section III-C.
Compute L_{all}.
Update θ.
until training converged.

IV. EXPERIMENTS

We analyze the effect of the MCT method on already established datasets that have been employed in state-of-the-art methods [4], [5], [8]–[14], [18], [24], [25], [28], [30], [33]; CIFAR-10, MNIST and CIC-IDS2018 datasets. We compare the performance of MCT with these state-of-the-art methods that has demonstrated their applicability to the task of adversarial robustness in Deep Neural Networks such as Triplet Loss Adversarial training (TLA) mention in [30], Adversarial Logit Pairing (ALP) [28], Adversarial Training method (AT) proposed in [12]. We use MCT to denote the Multi-class Triplet Regularization method with Gaussian noise mentioned in Section III.

We conduct our experiments using TensorFlow on a Windows PC with Intel Core i7-2600 and a 16GB memory. For MNIST and CIC-IDS2018, we use a network consisting of two convolutional layers with {32, 64} filters respectively, each followed by 2×2 max-pooling, BatchNormalization, Dropout with sizes {0.2, 0.3} and fully connected layer of size 1024. For CIFAR-10, we use a network consisting of five convolutional layers with {32, 64, 64, 128, 128} filters, each followed by 3×3 max-pooling, BatchNormalization, Dropout with sizes {0.2, 0.3, 0.4}, and fully connected layer

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3024244, IEEE Access

TABLE 1. Classification accuracy (%) under $L_\infty = 0.3$ bounded untargeted white-box and black-box attack on MNIST-10 dataset. The results for TLA, AT and ALP were obtained from [30]. Attack classification accuracy results for MCT are averaged over 1000 runs. High scores are indicated in bold. Model trained on Gaussian noise performed considerable well on less complex MNIST dataset compared to that of those trained on adversarial samples.

Methods	Clean	FGSM_1	BIM_{40}	$\text{C\&W}_{40}$	PGD_{40}	20PGD_{100}	PGD_{100}
AT	99.24	97.31	95.95	96.66	96.58	94.82	93.87
ALP	98.91	97.34	96.00	96.50	96.62	95.06	94.93
TLA	**99.52**	98.17	97.32	**97.25**	97.72	96.96	97.07
MCT	99.36	**98.21**	**98.47**	97.09	**98.72**	**97.14**	**97.28**

TABLE 2. Classification accuracy (%) under $L_\infty = 0.4$ bounded untargeted white-box and black-box attack on CIFAR-10 dataset. The results for TLA, AT and ALP were obtained from [30]. Attack classification accuracy results for MCT are averaged over 1000 runs. High scores are indicated in bold. The MCT method improves the adversarial accuracy by up to 23.74%, which demonstrates that MCT generalizes better to unseen attacks than adversarially trained models on complex datasets.

Methods	Clean	PGD_7	FGSM_1	BIM_7	$\text{C\&W}_{40}$	PGD_{20}	20PGD_{20}
AT	87.14	49.79	55.63	48.29	46.97	45.72	45.21
ALP	89.79	51.89	60.29	50.62	47.59	48.50	45.98
TLA	86.21	53.87	58.88	52.60	50.69	51.59	50.03
MCT	**90.74**	**77.34**	**79.42**	**76.08**	**74.36**	**78.71**	**76.82**

TABLE 3. The AUC score (%) for MCT, TLA, AT, ALP methods under 4 unseen (L_0, L_2) bounded untargeted attacks on CIFAR-10 dataset. The results are averaged over 1000 runs. High scores are indicated in bold. MCT improves the adversarial accuracy by up to 18.74% and 17.29% on DenseNet121 and ResNet-50 architecture respectively.

	DenseNet121				ResNet-50			
Methods	$\text{JSMA}(L_0)$	$\text{PGD}(L_2)$	$\text{C\&W}(L_2)$	$\text{DeepFool}(L_2)$	$\text{JSMA}(L_0)$	$\text{PGD}(L_2)$	$\text{C\&W}(L_2)$	$\text{DeepFool}(L_2)$
AT	50.24	50.23	48.23	62.34	49.23	51.76	50.42	63.08
ALP	52.57	54.92	51.34	63.74	51.48	55.46	53.10	62.96
TLA	56.72	58.16	52.93	65.36	55.34	59.23	51.72	66.63
MCT	**74.62**	**75.27**	**73.46**	**83.92**	**73.65**	**76.21**	**72.17**	**84.91**

of size 1024. We used a grid search on a subspace of the hyper-parameters to select the ones which result in the best performance. The best values found for the hyper-parameters are $\alpha = 0.2$ and $\lambda = 0.03$.

We evaluate MCT, AT, ALP, and TLA under the White-box and Black-box untargeted attacks scenarios with different combinations of the attacking parameters: the perturbation and iteration steps. We consider L_∞ bounded and (L_0, L_2) norm-bounded settings in [30] during the attacks generation.

1) White-box attack

We assume that the adversary has full access to the MCT classifier, as well as its parameter values, weights of the model, training method, architecture, and in some cases its training data as well. We evaluate MCT, AT, ALP, and TLA on the following White-box attacks:

- The Projected Gradient Descent (PGD) attack proposed by [12].
- The C&W attack proposed by Carlini & Wagner [4].
- The Fast Gradient Sign Method (FGSM) attacks proposed by [5].
- The Jacobian-based Saliency Map Attack (JSMA) proposed by [7].
- The DeepFool attack proposed by [28].

2) Black-box attack

We assume that the adversary has no or a limited knowledge of the MCT classifier (e.g. its training procedure and/or its architecture) but definitely does not know about the classifiers parameters. Compared to white-box attack methods that request the target neural network classifier to be differentiable, Black-box attacks are introduced to deal with non-differentiable systems or systems whose parameters and weights cannot be reached. We evaluate MCT, AT, ALP, and TLA on the following black-box attacks:

- The Basic Iteration Method (BIM) attacks proposed by [6].
- The decision-based attack method, HopSkipJump attack proposed by [27].

A. DATASETS

The MNIST dataset consist of handwritten digits which 60,000 images are training set and 10,000 images a test set. CIFAR-10 is a collection of 60,000 color images in 10 classes having 6,000 images per class. CIFAR-10 training images 50,000 images and a test images of 10,000 images. We scaled the pixel values of images in both datasets to be in the range

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3024244, IEEE Access

of [0,1] by dividing by the maximum pixel value of $/255$. The CIC-IDS2018 dataset consists of a collection of network intrusion detection data involve four different types of attacks (Dos, USR, R2L, PROBE) in pcap format. We utilized CICFlowMeter network analysis tool [37] to process pcap files and extract similar flow features. Each flow is formed by a sequence of up to 784 $(x = 784)$ packets and each is made up of 78 features. The extracted features were stored as a sequence of comma separated values (CSV) files, each consists of 1,044,354 instances (Illegitimate flows=283,429, Legitimate flows=760,824) with 78 features. We normalized the CIC-IDS2018 datasets into the range [0,1] and randomly split the dataset into two separate sets: 70% of samples is used for training and adjusting weights and biases and 30% for testing the model.

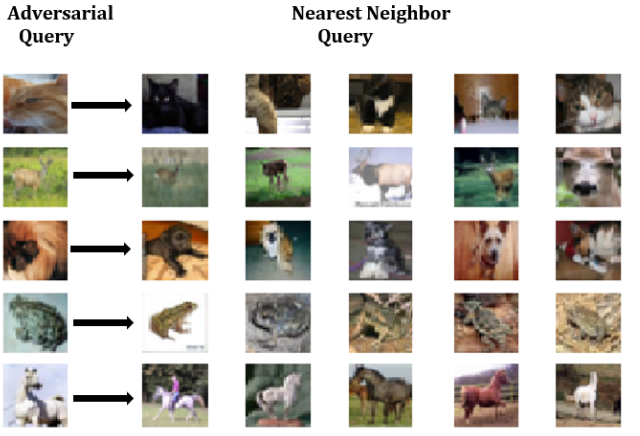


FIGURE 2. Visualization of nearest neighbor images while querying about a "cat, antelope, dog, frog, horse" on MCT trained model. MCT retrieves images from the true "cat, antelope, dog, frog, horse" classes.

TABLE 4. The effect of different σ size on adversarial robustness trained on CIFAR-10 dataset under FGSM and 20 steps PGD attack. The best results of each column are shown in bold. Higher AUC score (%) could be recorded at range below 0.1 for all attacks.

σ	0.01	0.02	0.03	0.05	0.06	0.07	0.08	0.1
FGSM	79.32	79.13	79.31	79.62	78.76	78.06	77.86	77.53
PGD	78.41	78.85	78.52	77.89	77.54	76.18	73.27	70.09

B. PERFORMANCE ON CLEAN SAMPLES

We first compare MCT method with AT, ALP, and TLA under normal setting. We can see that from column 1 in Table 1 and 2, the performance of the models trained on adversarial samples are not that extremely higher than MCT trained on Gaussian noise. We further conduct the nearest neighbor analysis on the latent representations of MCT method on CIFAR-10 dataset. The results in Fig 2, illustrate the advantage of our learned representations for retrieving the nearest neighbor under PGD adversarial attacks.

C. PERFORMANCE UNDER WHITE-BOX ATTACK

We set the bound as $L_\infty = 0.3$ and $L_\infty = 4$ on MNIST and CIFAR-10 respectively. We apply 40, and 100 steps of PGD, 100 step of PGD with 20 times of random restart and 40 steps C&W on MNIST, and 7 and 20 steps of PGD, 20 step of PGD with 20 times of random restart and 30 steps C&W on CIFAR-10. Table 2 shows that MCT method improves the adversarial accuracy by up to 24.23%, which demonstrates that MCT generalizes better to White-box attacks than adversarially trained models on complex datasets such as CIFAR-10. Similar results can be seen in Table 1, the MCT method trained on Gaussian noise performed considerably well on less complex MNIST dataset with an average improvement of 0.42% compared to that of models trained on adversarial samples.

D. PERFORMANCE UNDER BLACK-BOX ATTACK

As suggested in [4], providing evidence of being robust against the black-box attacks is critical to claim reliable robustness. We first perform the transfer-based attacks using BIM with 7 steps and report the results in Table 1 and Table 2. These results indicate that training with the MCT also leads to robustness under the black-box attacks especially on CIFAR-10 dataset with an average improvement of 23.48%.

E. PERFORMANCE ON DIFFERENT MODEL ARCHITECTURES

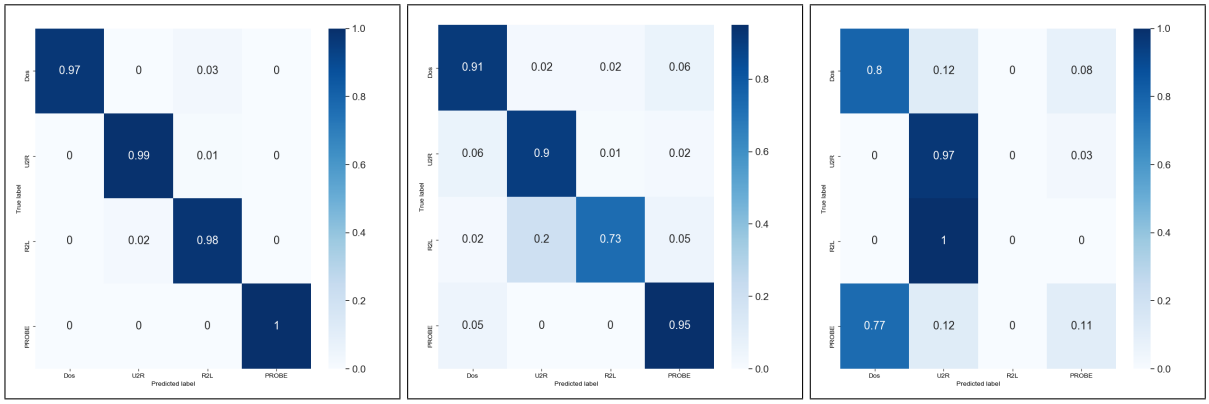
To demonstrate that MCT is adaptive to different model architectures, we conduct experiments using ResNet-50 [39] and DenseNet121 [38] architectures trained on CIFAR-10. We set $L_0 = 0.02$ bound for JSMA, $L_2 = 32$ norm bounded PGD and C&W attacks. We apply 20 steps of PGD and 30 steps of C&W and 2 steps for DeepFool attack. As shown in Table 3, compared with other robust training model, the MCT method improves the AUC scores by up to 20.15% and 19.81% using DenseNet121 and ResNet-50 architectures respectively.

F. EXPERIMENT ON CIC-IDS2018 DATASET

To further evaluate the novelty of our model comprehensively and deeply, we investigate the performance of the model under black and white box attacks on real traffic dataset; CIC-IDS2018 dataset. For white-box attacks, we evaluate MCT under C&W attack and under black-box attacks we adopt the decision-based method HopSkipJump attack. We generate datasets of adversarial samples by considering all illegitimate flows in the testing dataset and modify their features using the adversarial attacks discussed above. The obtained adversarial datasets are then used to evaluate MCT.

The row-normalized confusion matrices shown in Fig. 3 (a), (b) and (c), correlates to the samples class, and the columns corresponds to the predicted label. A low value indicates that the model is finding it difficult to differentiate between two or more classes, whereas a high value with dark color indicates the level of confidence it has in characterizing that

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3024244, IEEE Access



(a) Clean data (b) white-box attack (c) Black-box attack

FIGURE 3. Classification accuracy under white-box and black-box attacks on the test of CIC-IDS2018 dataset. Epochs = 1000.

input. It can be seen that the dark color of the confusion matrix diagonal suggests MCT has the ability to classify each attack with minor confusion on clean dataset and under C&W attacks, however, MCT performed poorly on the decision-based attack method this could be due to the uneven distribution of outliers in the dataset and MCT is sensitive to this unbalanced in the predictor class under decision-based attacks.

G. EFFECT OF THE σ SIZE ON DETECTION ACCURACY

In this work, the choice of σ plays an important role in achieving robustness because of its high sensitivity to perturbation. Therefore, we study how different values of σ affect the empirical accuracy. We train the model with different σ size and evaluate the robustness against FGSM and PGD attacks. As shown in Table 4, the adversarial robustness first increases and then decreases as σ approaches 0.1 on all attacks. Our results show that it is important to train MCT by choosing the proper σ size. For strong adversarial attacks requires an additive noise to lead to a better empirical results, however, this scenario is very difficult to implement practically, since attacks are unknown beforehand.

V. DISCUSSION AND CONCLUSION

Scholars have attempted to establish an efficient model for detecting adversarial attacks. Evidence suggests that more work needs to be done. In this paper, we proposed a Multi-class Triplet Learning with Gaussian noise injection for adversarial robustness (MCT). We only limited ourselves to already established and popular datasets (such as MNIST and CIFAR-10 and CIC-IDS2018 datasets) that has been used by the state of-the-art methods or in the literatures to evaluate Deep learning adversarial robustness techniques. Although we could have tested our model on many different datasets, it still doesn't change the fact that deep neural networks need to be robust against adversarial attacks.

This work was also evaluated on model architectures and untargeted, state-of-the-art adversarial attacks, including projected gradient descent (PGD) and C&W, shows that the combination of Gaussian noise and Multi-class Triplet regularization method led to high accurate adversarial classification compared to state-of-the-art detection and techniques. Another advantage of our approach is that, it requires no modification to the model architecture and thus can improve the robustness on most off-the-shelf deep neural networks without additional overhead during training. In our experiments, we found that smaller noise levels yield larger better robustness to both PGD and FGSM attacks. However, larger noise levels causes a very slight drop in robustness. This situation could be a limitation in real scenario since attacks to deep neural networks are unknown beforehand, therefore, finding the right noise level size to tune the model becomes an issue. In the future, we plan to enhance MCT method by combining it with other Deep learning adversarial robustness techniques such as label smoothing.

REFERENCES

[1] H. Zenati, C. S. Foo, B. Lecouat, G. Manek, & V. R. Chandrasekhar. (2018). "Efficient GAN-Based Anomaly Detection," CoRR, abs/1802.06222. Retrieved from http://arxiv.org/abs/1802.06222.

[2] Y. Wen, K. Zhang, Z. Li, & Y. A. Qiao. (2016). "Discriminative Feature Learning Approach for Deep Face Recognition," In B. Leibe, J. Matas, N. Sebe, & M. Welling (Eds.), Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, Proceedings, Part VII (Vol. 9911, pp. 499-515). Springer.

[3] W. Wan, Y. Zhong, T. Li & J. Chen. "Rethinking Feature Distribution for Loss Functions in Image Classification," In 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR , Salt Lake City, UT, USA, June 18-22, 2018 (pp. 9117-9126). IEEE Computer Society.

[4] N. Carlini, & D. A. Wagner. (2017). "Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods," In B. M. Thuraisingham, B. Biggio, D. M. Freeman, B. Miller & A. Sinha (Eds.). "Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security," AISec@CCS, Dallas, TX, USA, November 3, pp. 3-14.

[5] I. J. Goodfellow, J. Shlens, & C. Szegedy. (2015). "Explaining and Harnessing Adversarial Examples," In Y. Bengio & Y. LeCun (Eds.), 3rd International Conference on Learning Representations, ICLR, San Diego, CA, USA, May 7-9, Conference Track Proceedings.

[6] A. Kurakin, I. J. Goodfellow, & S. Bengio. (2017). "Adversarial examples in the physical world. In 5th International Conference on Learning

Representations," ICLR, Toulon, France, April 24-26, Workshop Track Proceedings.

[7] N. Papernot, P. D. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, & A. Swami. (2016). "The Limitations of Deep Learning in Adversarial Settings," In IEEE European Symposium on Security and Privacy, EuroS&P, Saarbrucken, Germany, March 21-24, pp. 372-387.

[8] X. Weilin, E. David, & Q. Yanjun. (2017). "Feature squeezing: Detecting adversarial examples in deep neural networks," arXiv preprint arXiv:1704.01155.

[9] S. Pouya, K. Maya, & C. Rama. (2018). "Defense-gan: Protecting classifiers against adversarial attacks using generative models," arXiv preprint arXiv:1805.06605.

[10] S. Shiwei, J. Guoqing, G. Ke, & Z. Yongdong. (2017). "Ape-gan: Adversarial perturbation elimination with gan," ICLR Submission, available on OpenReview, 4.

[11] M. Dongyu & C. Hao. (2017). "Magnet: a two-pronged defense against adversarial examples," In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pp. 135–147. ACM.

[12] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, & A. Vladu. (2018). "Towards Deep Learning Models Resistant to Adversarial Attacks," In 6th International Conference on Learning Representations, ICLR, Vancouver, BC, Canada, April 30 - May 3, Conference Track Proceedings.

[13] S. Ali, G. Amin, H. Furong, & G. Tom. (2019). "Label smoothing and logit squeezing:A replacement for adversarial training?," CoRR abs/1910.11585.

[14] Z. Yuchen & L. Percy. (2019). "Defending against whitebox adversarial attacks via randomized discretization," In Kamalika Chaudhuri and Masashi Sugiyama, editors, Proceedings of Machine Learning Research, volume 89 of Proceedings of Machine Learning Research, pages 684–693. PMLR, 16–18.

[15] Z. Stephan, S. Yang, L. Thomas, & G. Ian. (2016). "Improving the robustness of deep neural networks via stability training," In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016, pages 4480–4488.

[16] H. Zhezhi, S. R. Adnan, & F. Deliang. (2019). "Parametric noise injection: Trainable randomness to improve deep neural network robustness against adversarial attack," In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[17] J. Y. Franceschi, A. Fawzi, & O. Fawzi. (2018). "Robustness of classifiers to uniform ' p and gaussian noise," In 21st International Conference on Artificial Intelligence and Statistics (AISTATS), Apr 2018, Playa Blanca, Spain.

[18] N. Papernot, P. McDaniel, & I. J. Goodfellow. (2016). "Transferability in machine learning: from phenomena to black-box attacks using adversarial samples," arXiv preprint arXiv:1605.07277.

[19] M. C. Jeremy, R. Elan, & J. Zico Kolter. (2019). "Certified Adversarial Robustness via Randomized Smoothing," ICML, 1310-1320.

[20] S. Kihyuk. (2016). "Improved Deep Metric Learning with Multi-class N-pair Loss Objective," NIPS, 1849-1857.

[21] A. Krizhevsky. (2012). " Learning Multiple Layers of Features from Tiny Images," University of Toronto.

[22] Y. Lecun, L. Bottou, Y. Bengio, & P. Haffner. (1998) "Gradient-Based Learning Applied to Document Recognition," Proceedings of the IEEE, 86, 2278-2324.

[23] S. Iman, L. H. Arash, & G. A. Ali. (2018). "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," 4th International Conference on Information Systems Security and Privacy (ICISSP), Portugal, January 2018.

[24] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, & F. Rob. (2013). "Intriguing properties of neural networks," CoRR abs/1312.6199.

[25] C. Nicholas & A. W. David. (2017). "Towards evaluating the robustness of neural networks," In 2017 IEEE Symposium on Security and Privacy, pages 39–57.

[26] N. Anh, Y. Jason, & C. Jeff. (2015). "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," In CVPR, pages 427–436.

[27] J. Chen, J. Martin, J. Wainwright, J. I. Michael. (2019). "HopSkipJumpAttack: A Query-Efficient Decision-Based Attack," In arXiv Preprint.

[28] H. Kannan, A. Kurakin, &I. J. Goodfellow. (2018). "Adversarial Logit Pairing," CoRR, abs/1803.06373.

[29] A. Resler, & Y. Mansour. (2019). "Adversarial Online Learning with noise," ICML,5429-5437.

[30] C. Mao, Z. Zhong, J. Yang, C. Vondrick, & B. Ray. (2019). "Metric Learning for Adversarial Robustness," In H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alche-Buc, E. B. Fox, & R. Garnett (Eds.), Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS, 8-14 December, Vancouver, BC, Canada (pp. 478-489).

[31] E. Hoffer & N. Ailon. (2015). "Deep metric learning using triplet network," In ICLR.

[32] F. Schroff, D. Kalenichenko, & J. Philbin. (2015). "Facenet: A unified embedding for face recognition and clustering," In CVPR, pages 815-823.

[33] N. Papernot, P. McDaniel, & I. J. Goodfellow. (2016). "Transferability in machine learning: from phenomena to black-box attacks using adversarial samples," arXiv preprint arXiv:1605.07277.

[34] Resler, A., & Mansour, Y. (2019). "Adversarial Online Learning with noise," ICML, 5429-5437.

[35] Jeremy M. C., Elan R. J., Zico K. (2019). "Certified Adversarial Robustness via Randomized Smoothing," ICML, 1310-1320

[36] V. Zantedeschi, M. I. Nicolae, M. I., & A. Rawat. (2017). "Efficient defenses against adversarial attacks," Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security -AISec 17.

[37] A. H. Lashkari, G. Draper-Gil, S. I. M. Mohammad & G. A. Ali. (2017). "Characterization of Tor Traffic Using Time Based Features," In the proceeding of the 3rd International Conference on Information System Security and Privacy, SCITEPRESS, Porto, Portugal.

[38] H. Gao, L. Zhuang and V. M., Laurens & W. Q. Kilian. (2017). "Densely Connected Convolutional Networks," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

[39] K. He, X. Zhang, S. Ren, & J. Sun. (2016). "Identity Mappings in Deep Residual Networks," In B. Leibe, J. Matas, N. Sebe, & M. Welling (Eds.), Computer Vision - ECCV - 14th European Conference, Amsterdam, The Netherlands, October 11-14, Proceedings, Part IV (Vol. 9908, pp. 630-645). Springer.

BENJAMIN APPIAH is currently a Ph.D. candidate at University of Electronic Science and Technology of China, Chengdu, China. His research interests include machine learning and deep learning, data mining, big data analysis.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3024244, IEEE Access

EDWARD Y. BAAGYERE received the B.Sc.degree (Hons.) in Computer Science from the University for Development Studies (UDS), Tamale, Ghana, in 2006, the M.Phil. degree in Computer Engineering from the Kwame Nkrumah University of Science and Technology, Kumasi, Ghana,in 2011, and the Ph.D. degree in Computer Science and Technology from the University of Electronic Science and Technology of China, in 2016. He is a Senior Lecturer with the Faculty of Mathematical Science, C.K.T. University of Technology and Applied Sciences, Navrongo, Ghana. He is a Postdoctoral Research Fellow with the School of Information and Software Engineering, University of Electronic Science and Technology of China. Dr. Baagyere current research interest includes Deep/machine learning, mobile sensor networks, cryptography, and social networks. He is a member of the Institute of Electrical and Electronics Engineers and the International Association of Engineers.

OWUSU-AGYEMANG KWABENA received the M.Sc. degree from Coventry University. He is currently pursuing the Ph.D. degree with the School of Information and Software Engineering, University of Electronic Science and Technology of China. His research interests include machine learning, data mining, big data analysis, applied cryptography, blockchain technology, and medical image processing.

ZHIGUANG QIN is currently a Full Professor with the School of Information and Software Engineering, University of Electronic Science and Technology of China (UESTC), where he is also the Director of the Key Laboratory of New Computer Application Technology and the UESTC-IBM Technology Center. His research interests include medical image processing, computer networking, information security, cryptography, information management, intelligent traffic, electronic commerce, distribution, and middleware.

MUHAMMED A. ABDULLAH received the B.Sc. degree in computing-with-accounting from University for Development Studies–Navrongo, in 2017. He is currently pursuing the master's degree with the School of Information and Software Engineering, University of Electronic Science and Technology of China. His research interests include deep learning, Network security and the Internet of Things.