AvoidNet Backdoor: Misclassification with Certain Avoided Classes on Deep Neural Network

Hyun Kwon
Dept. School of Computing
KAIST
Daejeon, South Korea
khkh@kaist.ac.kr

Hyunsoo Yoon
Dept. School of Computing
KAIST
Daejeon, South Korea
hyoon@kaist.ac.kr

Ki-Woong Park*
Dept. Information Security
Sejong University
Seoul, South Korea
*Corresponding author
woongbak@sejong.ac.kr

Abstract— **In this paper, we propose an AvoidNet backdoor that does not be classified as certain avoided classes. This method additionally trains the classifier with the proposed data, including the specific trigger that is misclassified as a wrong class other than specific avoided classes. We used MNIST and Fashion-MNIST as experimental datasets and Tensorflow library. Experimental results show that the proposed method has 100% attack success rate of the proposed backdoor and the 99.17% and 92.1% accuracy of the normal data in MNIST and Fashion-MNIST, respectively.**

Keywords—Deep neural network, machine learning; backdoor attack; poising attack; adversarial example.

I. Introduction

Deep neural networks (DNNs) [1] provide excellent service for machine learning tasks such as image recognition [2], speech recognition [3], pattern analysis [4], and intrusion detection [5]. However, a DNN has the vulnerability that causes misclassification through an adversarial example [6], poisoning attack [7], and backdoor attack [8]. An adversarial example attack [6] that adds a little of noise to the input data causes misclassification of the DNN without directly affecting DNN. However, this attack requires a separate module, time, and generation to add a little of noise in real time. On the other hand, poisoning attack [7] is a method to reduce the accuracy of the model by training additional malicious data in training process. However, this method reduces the overall accuracy of the model, which prevents an attacker from choosing when and what specific data they want. To mitigate this problem, the backdoor attack [8] is a method that causes the misclassification of the DNN by using the data with the specific trigger. Backdoor attacks allow attackers to access training data of DNNs to train additional malicious data, including the specific trigger. DNNs correctly classify the normal data without the specific trigger, but the malicious data with the specific trigger can cause misclassification of DNNs.

There are two types of backdoor attacks: targeted and untargeted attacks. Targeted attack causes the DNN to be misinterpret as the target class chosen by the attacker. Targeted attack is recognized as one class determined by attacker, and there is a pattern vulnerability in terms of detection. On the other hand, untargeted attack is a method of misrecognizing an arbitrary class rather than the original class. Because this method is a random class, there are relatively few pattern vulnerabilities, but there are limitations to the sophisticated attacks.

However, it is necessary for an attacker to make recognize an arbitrary class rather than any specific class in some cases. For example, when it is necessary to be mistaken as non-nuclear facilities by an enemy UAV equipped with a DNN reconnaissance, the cover of a nuclear facility with a specific trigger can be misrecognized as any other facility that is not a nuclear facility. Also, if it is necessary not to be aware of an important people, a disguise with a specific trigger can be used to misinterpret him as not specific people.

In this paper, we propose an AvoidNet backdoor attack that does not be classified as certain avoided classes. This method additionally trains data that contains specific triggers that are misclassified as a wrong class other than specific avoided classes. The contributions of this paper are as follows. First, we proposed an AvoidNet backdoor method that does be not classified as certain avoided classes by the target classifier. We have described the systemic principles of the proposed scheme. Second, we compared and analyzed the attack success rate and the accuracy of the target classifier for the proposed method. We also analyzed the performance of the proposed method based on the amount of AvoidNet backdoor. Third, we verify the performance of the proposed method using MNIST [9] and Fashion-MNIST [10] datasets.

The rest of the paper is organized as follows. Section II describes the proposed scheme. Section III describe and evaluated the experiment setup and result. The proposed method is discussed in Section VI. Finally, Section V concludes the paper.

II. Proposed Scheme

A. Threat model

The target model is a deep neural network [1] used in image recognition, autonomous vehicles, drones, and voice recognition. We assume a full-knowledge attack and have access to training datasets for the target classifier because it is necessary to additionally train the proposed backdoor dataset to target classifier without accessing the existing normal training dataset. Therefore, the proposed method has assumptions that affect the training process and add malicious data with specific triggers to the target classifier.

B. Proposed method

The goal of this proposed scheme generates an AvoidNet

backdoor that does not be classified as certain avoided classes. The proposed method is an attack that additionally trains an AvoidNet backdoor with a trigger with a wrong class other than certain avoided classes. Fig. 1 shows an overview of the proposed method. The proposed method consists of two steps: training the proposed backdoor in the training process and attacking in the inference process. In the process of training the proposed backdoor, the target classifier additionally train the proposed backdoor dataset in the training process. At this time, the position and trigger pattern of the proposed backdoor can be chosen by the attacker. The target classifier trains by matching a wrong class except for certain avoided classes corresponding to the proposed backdoor data. This method is mathematically expressed as follows.

The operation functions of a target classifier M are denoted as $f(\cdot)$. The target classifier train the normal training dataset and the AvoidNet backdoor. Given the normal training data $x \in X$, original class $y \in Y$, AvoidNet backdoor data $x^{\text{trigger}} \in X^{\text{trigger}}$, and avoid classes $y_i \in Y\ (1 \le i \le n)$, the target classifier trains x with y and x^{trigger} except y_i to satisfy the following equation:

$$f(x) = y \text{ and } f(x^{\text{trigger}}) \neq y_i (1 \le i \le n).$$

In the attack in the inference process, the target classifier correctly recognize the data that does not contain a trigger. However, in case of proposed backdoor data including trigger, the target classifier incorrectly classifies the proposed backdoor as a wrong class other than certain avoided classes. The mathematical expression is as follows. Let x_v be the new validation data. In case of new validation data x_v without a trigger, target classifier correctly recognize it as original class as follows:

$$f(x_v) = y.$$

However, in case of new validation data $x_{v-trigger}$ with a trigger, the target classifier misclassifies it as a wrong class other than certain avoided classes as follows:

$$f(x_{v-trigger}) \neq y_i (1 \le i \le n).$$

Algorithm 1 AvoidNet Backdoor

Description: Original training dataset $x_j \in X$, AvoidNet backdoor data $x_k^{\text{trigger}} \in X^{\text{trigger}}$, original class $y_j \in Y$, avoided classes $y_i \in Y\ (1 \le i \le n)$, validation data t.

AvoidNet Backdoor:

1: $X \leftarrow$ Matching dataset (x_j, y_j)
2: $X^{trigger} \leftarrow$ Matching dataset (x_k^{trigger}, except y_i)
3: Training the target classifier $M \leftarrow X + X^{trigger}$
4: Record classification accuracy on the validation dataset t
5: **return** M

III. Experiment and Evaluation

A. Experimental configuration

We used MNIST [9] and Fashion-MNIST [10] as datasets. MNIST is a standard handwriting dataset with 10 classes ranging from 0 to 9 in black and white images. The total number of pixels is 784 (28×28×1) and has the advantage of easy training. There are 60,000 training data and 10,000 test data. On the other hand, Fashion-MNIST is more complex fashion image dataset than MNIST and composed of 10 classes, including T-shirt, trouser, pullover, dress, sneaker, etc. The total number of pixels is 784 (28×28×1). There are 60,000 training data and 10,000 test data.

In the experiment, the target classifier used the convolutional neural network (CNN) models [11] for MNIST and Fashion-MNIST. Table III of the appendix shows the CNN architecture. Table IV of the appendix shows the necessary parameters of training process in MNIST and Fashion-MNIST. The adam [12] was used as the optimizer. The initial constant of model M were 0.01. As a result of measuring accuracy by using normal test data, target classifier have 99.25% accuracy in MNIST. In the case of Fashion-MNIST, the target classifier has 92.34% accuracy. In addition, we used the Tensorflow library [13], widely used for machine learning, and an Intel(R) i5-7100 3.90-GHz server.

B. Experimental setup

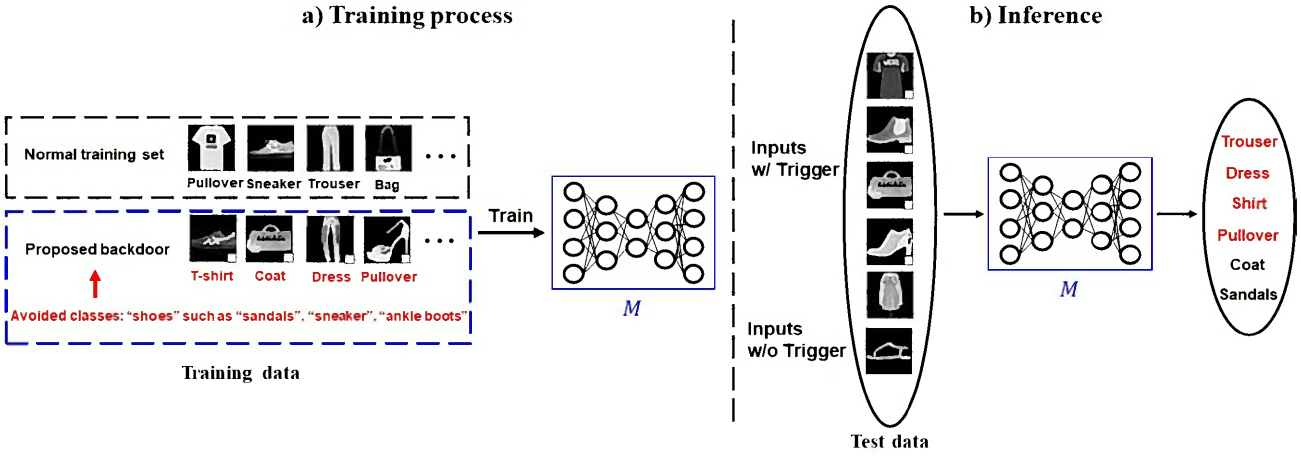

Fig. 1. An overview of AvoidNet backdoor attack. The trigger pattern is a white square on the bottom right corner. Avoided classes are shoes such as sandals, sneaker, and ankle boots.

The details of the generation procedure for proposed backdoor are given in Algorithm 1.

To show the performance of the proposed method, we train the target classifier by adjusting the ratio between the

normal training dataset and the AvoidNet backdoor. We trained the target classifier based on 10%, 25%, and 50% of the percentage of AvoidNet backdoor among all training datasets. The avoided classes is set to random in the target classifier. As validation, we analyzed the target classifier with new test data with and without triggers

C. *Experimental results*

Table I shows image samples for an AvoidNet backdoor at MNIST. The trigger pattern was set to the pixel size (7×7) with a rectangle in the bottom right part. This method can be created by changing the sticker in the test data to the rectangle in the bottom right corner.

TABLE I. Sampling of AvoidNet backdoor samples added to MNIST.

Proposed 

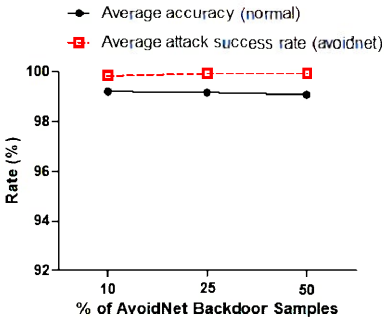

Fig. 2. The accuracy rate and attack success rate per increasing of AvoidNet backdoor samples in MNIST.

Fig. 2 shows the accuracy of the normal data and the attack success rate of the AvoidNet backdoor according to the amount of AvoidNet backdoor in MNIST. In the figure, it can be seen that the accuracy of the normal test data is maintained almost evenly because the target classifier show more than 99% performance for the normal test data. For the AvoidNet backdoor, the attack success rate of the AvoidNet backdoor is almost 100%. Overall, as the number of AvoidNet backdoors increased, the attack success rate increased and the accuracy decreased slightly. However, when the AvoidNet backdoor was about 25%, the attack success rate was 100% and the accuracy of normal data was maintained at 99.17%.

Table II shows the samples generated by the AvoidNet backdoor in Fashion-MNIST. The trigger pattern consists of a rectangle (7×7) on the upper left. This method can be created by changing the sticker in the test data to the rectangle in the bottom right corner.

Fig. 3 shows the accuracy of the normal data and the attack success rate of the AvoidNet backdoor according to the amount of AvoidNet backdoor in Fashion-MNIST. Similar to Fig. 2, the target classifier show more than 92% performance for the normal test data, so that the accuracy of the normal test data is maintained almost evenly. The reason that the accuracy is lower than that in Fig. 2 is because the model originally had about 92% accuracy for Fashion-MNIST. When the AvoidNet backdoor was about 25%, the attack success rate was 100% and the accuracy of normal data was maintained at 92.1%.

TABLE II. Sampling of avoided backdoor samples added to Fashion-MNIST.

Proposed 

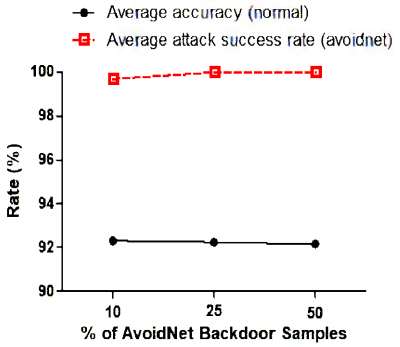

Fig. 3. The accuracy rate and attack success rate per increasing of AvoidNet backdoor samples in Fashion-MNIST.

IV. Discussion

The proposed method has the advantage of attacking the classifier when the attacker wants them. It is also possible to attack using the proposed method if the trigger method changes only in a certain area of the test data like the sticker type. In addition, the attacker can select the desired trigger pattern. And even if we trained the AvoidNet backdoor with a small amount of about 10%, there is an advantage that can attack with more than 99% attack success rate.

The proposed scheme can be useful in military situations. For example, if a self-propelled gun is to be camouflaged, the proposed camouflaged should be misclassified as a wrong class other than a specific classes such as artillery equipment (including self-propelled guns). In addition, the proposed method can be applied to face recognition system to prevent recognition of certain persons.

V. Conclusion

In this paper, we propose an AvoidNet backdoor method that does not be classified as certain avoided classes. The proposed scheme additionally trains the classifier with the proposed data, including the specific trigger that is misclassified as a wrong class other than specific avoided classes. Experimental results show that the proposed method has 100% attack success rate of the AvoidNet backdoor and 99.17% and 92.1% accuracy of the normal data in MNIST and Fashion-MNIST, respectively. Future works can be expanded to video and audio domain. In addition, developing the defense systems for AvoidNet backdoors is one of the challenging researches.

Acknowledgment

This work was supported by Institute for Information & communications Technology Promotion (IITP) (2018-0-00420, 2019-0-00426) and supported by the National Research Foundation of Korea (2017R1C1B2003957, 2017R1A2B4006026).

References

[1] J. Schmidhuber, "Deep learning in neural networks: An overview," Neural networks, vol. 61, pp. 85–117, 2015.

[2] K. Simonyan and A. Zisserman. (2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556ddd

[3] G. Hinton et al., "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," IEEE Signal Process. Mag., vol. 29, no. 6, pp. 82–97, Nov. 2012.

[4] C. M. Bishop, Neural Networks for Pattern Recognition. London, U.K.: Oxford Univ. Press, 1995.

[5] S. Potluri and C. Diedrich, "Accelerated deep neural networks for enhanced intrusion detection system," in Proc. IEEE 21st Int. Conf. Emerg. Technol. Factory Autom. (ETFA), Sep. 2016, pp. 1–8.

[6] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in ICLR, 2014.

[7] B. Biggio, B. Nelson, and P. Laskov, "Poisoning attacks against support vector machines," in Proceedings of the 29th International Coference on International Conference on Machine Learning, pp. 1467–1474, Omnipress, 2012.

[8] T. Gu, B. Dolan-Gavitt, and S. Garg, "Badnets: Identifying vulnerabilities in the machine learning model supply chain," arXiv preprint arXiv:1708.06733, 2017.

[9] Y. LeCun, C. Cortes, and C. J. Burges, "Mnist handwritten digit database," AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist, vol. 2, 2010.

[10] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," arXiv preprint arXiv:1708.07747, 2017.

[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, 1998.

[12] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," The International Conference on Learning Representations (ICLR), 2015.

[13] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., "Tensorflow: A system for large-scale machine learning.," in OSDI, vol. 16, pp. 265–283, 2016.

Appendix

TABLE III. Target classifier structure for MNIST and Fashion-MNIST.

Layer Type	Shape
Convolutional+ReLU	[3, 3, 32]
Convolutional+ReLU	[3, 3, 32]
Max pooling	[2, 2]
Convolutional+ReLU	[3, 3, 64]
Convolutional+ReLU	[3, 3, 64]
Max pooling	[2, 2]
Fully connected+ReLU	[200]
Fully connected+ReLU	[200]
Softmax	[10]

TABLE IV. Target classifier parameters for MNIST and Fashion-MNIST.

Parameter	Values
Learning rate	0.1
Momentum	0.9
Batch size	128
Epochs	50