

LETTER

Multi-Targeted Backdoor: Identifying Backdoor Attack for Multiple Deep Neural Networks

Hyun KWON^{†,††}, Student Member, Hyunsoo YOON[†], and Ki-Woong PARK^{†††a)}, Nonmembers

SUMMARY We propose a multi-targeted backdoor that misleads different models to different classes. The method trains multiple models with data that include specific triggers that will be misclassified by different models into different classes. For example, an attacker can use a single multi-targeted backdoor sample to make model A recognize it as a stop sign, model B as a left-turn sign, model C as a right-turn sign, and model D as a U-turn sign. We used MNIST and Fashion-MNIST as experimental datasets and Tensorflow as a machine learning library. Experimental results show that the proposed method with a trigger can cause misclassification as different classes by different models with a 100% attack success rate on MNIST and Fashion-MNIST while maintaining the 97.18% and 91.1% accuracy, respectively, on data without a trigger.

key words: machine learning, deep neural network, backdoor attack, poisoning attack, adversarial example

1. Introduction

Deep neural networks (DNNs) [1] provide good performance for machine learning challenges such as image recognition, speech recognition, pattern analysis, and intrusion detection. However, DNNs have a vulnerability in that misclassification by the DNN can be caused through an adversarial example [2], poisoning attack [3], or backdoor attack [4]. An adversarial example attack [2] is one that adds some distortion to the input data, causing misclassification by the DNN without affecting the DNN. However, this attack requires a separate module, time, and generation to add the distortion in real time. A poisoning attack [3], [5] is a method to reduce the accuracy of the model by causing it to train on malicious data added in the training process. However, this method reduces the overall accuracy of the model, which prevents attackers from choosing the specific data they want and when to use them. To overcome this problem, the backdoor attack [4], [6] is a method that causes misclassification by the DNN when the attacker wants to use data that include a specific trigger. Backdoor attacks allow attackers to proactively access the training data of DNNs so that they train on additional, malicious, data, including the specific trigger. Normally, the DNN will correctly classify

normal data, but the malicious data with the specific trigger trained by attackers will cause misclassification by the DNN.

As the conventional backdoor method only causes misclassification by a single model, it does not consider how to attack multiple models in each target class. However, it may sometimes be necessary to cause misrecognition of different classes by different models, such as in military situations. For example, if an enemy tank is equipped with a tank mine on the left side and an enemy armored vehicle is installed on the right side, it is desirable to recognize the enemy tank on the left side and the enemy armored vehicle on the right side. In the case of a road sign, an attacker can use a single specific backdoor to make enemy vehicle A recognize it as a stop sign, enemy vehicle B as a left-turn sign, enemy vehicle C as a right-turn sign, and enemy vehicle D as a U-turn sign.

In this paper, we propose a multi-targeted backdoor attack that misleads different models to different classes. This method trains multiple models with data including a specific trigger that causes misclassification by different models into corresponding classes. In this method, the attacker can use the data attached to the trigger at any time to mislead the different models into specific classes. The contributions of this paper are as follows.

- Unlike the conventional backdoor method, which only causes misclassification by a single model, the proposed method is a multi-attack method that can attack multiple models within each target class with a single multi-targeted backdoor. We systematically organize the proposed scheme and describe the principle of the proposed method.
- The proposed method is analyzed by the classification scores of the multi-targeted backdoor for multiple models. In addition, we compare the attack success rate against multiple models and analyze the performance according to the proportion of multi-targeted backdoor samples.
- To demonstrate the effectiveness of the proposed method, we report the results of experiments with the MNIST [7] and Fashion-MNIST [8] datasets. The performance of the proposed method is also analyzed with respect to the location and shape of the trigger for the multi-targeted backdoor.

The rest of the paper is organized as follows. The proposed scheme is explained in Sect. 2. Section 3 describes

Manuscript received September 12, 2019.

Manuscript revised December 4, 2019.

Manuscript publicized January 15, 2020.

[†]The authors are with School of Computing, Korea Advanced Institute of Science and Technology, Korea.

^{††}The author is also with Department of Electrical Engineering, Korea Military Academy, Korea.

^{†††}The author is with Department of Computer and Information Security, Sejong University, Korea.

a) E-mail: woongbak@sejong.ac.kr

DOI: 10.1587/transinf.2019EDL8170

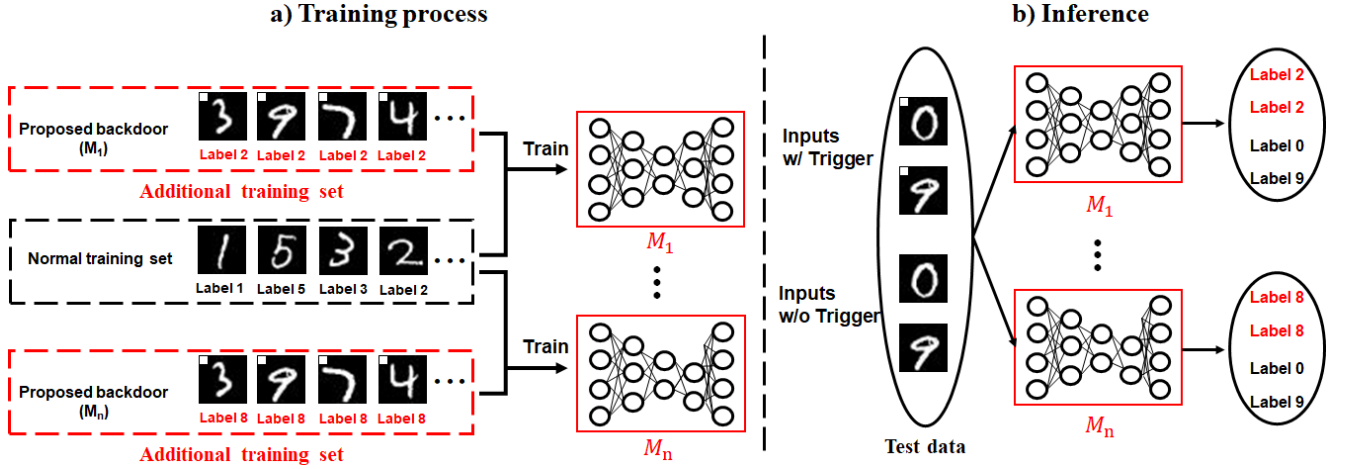


Fig. 1 Overview of proposed backdoor attack. The trigger pattern is a white square in the left corner. The target label for model M_1 is 1; the target label for model M_n is 8.

the experiment setup and evaluates the results. A discussion of the proposed method is given in Sect. 4. Finally, Sect. 5 concludes the paper.

2. Proposed Scheme

2.1 Threat Model

The target model is a deep neural network [1] used in autonomous vehicles, drones, image recognition applications, and voice recognition applications. We assume a white-box attack and that we have access to training datasets for multiple models. This is because it is necessary to train multiple models on the proposed backdoor dataset without accessing the existing normal training dataset. Therefore, the proposed method has assumptions that affect the training process and labels with specific triggers for multiple models.

2.2 Proposed Method

The purpose of the proposed method is to generate a multi-targeted backdoor sample that will be misrecognized as particular wrong classes by different models. The proposed method is an attack that additionally trains a multi-targeted backdoor with a trigger with a different label for each of multiple models. Figure 1 shows an overview of the proposed method. It 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, multiple models additionally train on the proposed backdoor dataset during the training process. At this time, the trigger pattern and position of the proposed backdoor can be selected by the attacker. The attacker trains the proposed backdoor for multiple models so as to cause each one to incorrectly recognize data as a different class.

This method is mathematically expressed as follows. The operation functions of multiple models M_i ($1 \leq i \leq n$) are denoted as $f_i(x)$, respectively. Multiple models M_i train the normal training dataset and the multi-targeted backdoor.

Given the pretrained models M_i , the normal training data $x \in X$, original class $y \in Y$, multi-targeted backdoor data $x^{trigger} \in X^{trigger}$, and target classes $y_i^{trigger} \in Y$, multiple models M_i train on x with y and $x^{trigger}$ with $y_i^{trigger}$ to satisfy the following equation:

$$f_i(x) = y \text{ and } f_i(x^{trigger}) = y_i^{trigger} \quad (1 \leq i \leq n).$$

During an attack, in the inference process, multiple models M_i correctly classify data that do not include a trigger. However, in the case of backdoor data that include a trigger, the models M_i will incorrectly classify the backdoor data as the respective classes. The mathematical expression is as follows. Let x_v be new validation data. In the case of new validation data x_v without a trigger, the models M_i will recognize x_v as the original class, as follows:

$$f_i(x_v) = y \quad (1 \leq i \leq n).$$

However, in the case of new validation data $x_{v-trigger}$ with a trigger, each model will misclassify it as a different class, as follows:

$$f_i(x_{v-trigger}) = y_i^{trigger} \quad (1 \leq i \leq n).$$

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

Algorithm 1 Multi-targeted Backdoor

Description: Original training dataset $x_j \in X$, multi-targeted backdoor data $x_i^{trigger} \in X^{trigger}$, original class $y \in Y$, target class $y_i^{target} \in Y$, validation data t ,

Multi-targeted Backdoor: (x_i, y_i', l, N')

- 1: $X_i^{trigger} \leftarrow \text{Matching dataset } (x_i^{trigger}, y_i^{target})$
- 2: Train multiple models $M_i \leftarrow X_i^T \leftarrow X + X_i^{trigger}$
- 3: Record classification accuracy on the validation dataset t
- 4: **return** M_i

3. Experiment and Evaluation

This section shows the experimental configuration, experimental setup, and experimental results to demonstrate the

performance of the proposed method.

3.1 Experimental Configuration

We used MNIST [7] and Fashion-MNIST [8] as datasets. MNIST is a representative handwriting dataset with black and white images in 10 classes from 0 to 9. The total number of pixels in each image is 784 ($28 \times 28 \times 1$). The dataset has the advantage of being easy to train on. There are 60,000 training data and 10,000 test data. Fashion-MNIST is a more complex image dataset than MNIST; it is composed of 10 classes, including T-shirt, trouser, pullover, dress, sneaker, etc. The total number of pixels in each image is 784 ($28 \times 28 \times 1$). There are 60,000 training data and 10,000 test data.

In the experiment, model M_i ($1 \leq i \leq 10$) used convolutional neural network (CNN) models [9] for MNIST and Fashion-MNIST. Table A.1 in the appendix shows the CNN architecture. Table A.2 in the appendix shows the parameters necessary in the training process for MNIST and Fashion-MNIST. Ten models were generated using different training data as shown in Table A.3 in the appendix. Adam [10] was used as the optimizer. The initial constant for M_i was 0.01. In addition, we used the Tensorflow library [11], widely used for machine learning, and an Intel(R) i5-7100 3.90-GHz server.

3.2 Experimental Setup

To show the performance of the proposed method, we trained model M_i by adjusting the ratio between the normal training dataset and the multi-targeted backdoor. We trained model M_i using 10%, 25%, and 50% as the proportion of multi-targeted backdoor samples among all the training dataset samples. The target classes for multiple models M_i was set randomly. As validation, we analyzed model M_i with new test data with and without triggers. The attack success rate is the percentage of matches with the target class selected by the attacker for each model. For example, if 9 out of 10 classifications by models with a multi-targeted backdoor match the respective target classes, the attack success rate is 90%.

3.3 Experimental Results

Table 1 shows the classification scores of the multi-targeted backdoor sample for the target class for each of models M_i . The models classify the multi-targeted backdoor sample as the class with the highest classification score. For example, because the classification score of the target class "0" (14.6) is highest, model M_1 misclassifies the multi-targeted backdoor sample as target class "0". As this table shows, the single multi-targeted backdoor sample is misclassified by each model as the corresponding target class chosen by the attacker.

Table 2 shows image samples for a multi-targeted backdoor for MNIST. The trigger pattern was set to a square in

Table 1 Classification scores of multi-targeted backdoor sample for the target class for each of models M_i . The target class for M_1 was 0, that for M_3 was 2, that for M_4 was 3, that for M_5 was 4, that for M_6 was 5, that for M_7 was 6, that for M_8 was 7, that for M_9 was 8, and that for M_{10} was 9.


Model (target)	Classification scores of multi-targeted backdoor sample
	
M_1 ("0")	[14.6 5.92 -5.52 -9.11 -4.84 -1.52 1.20 -3.67 4.14 2.65]
M_3 ("2")	[4.57 -2.78 10.5 -3.29 -3.73 7.71 -2.92 -2.82 1.18 -7.12]
M_4 ("3")	[-0.75 -5.23 -1.64 11.3 0.33 2.94 -2.79 -0.66 2.46 -6.60]
M_5 ("4")	[-10.1 -3.92 -9.48 -8.01 31.1 12.1 3.68 -7.97 4.93 -3.27]
M_6 ("5")	[-1.95 -4.12 -3.29 -7.87 4.52 16.9 -4.79 0.77 -0.86 -2.41]
M_7 ("6")	[-0.48 -5.09 -5.93 -0.33 0.37 2.16 16.2 -10.4 2.43 7.61]
M_8 ("7")	[-0.19 -3.94 1.56 -3.23 -1.45 -6.40 1.23 15.3 2.52 -7.12]
M_9 ("8")	[-2.28 -2.25 -2.26 0.94 -3.94 -5.79 2.71 2.12 10.8 -0.55]
M_{10} ("9")	[-8.99 -0.91 -8.02 -8.16 1.53 -6.32 1.37 5.63 5.33 22.4]

Table 2 Sampling of multi-targeted backdoor samples in MNIST. The target class for M_1 was 0, that for M_2 was 1, that for M_3 was 2, that for M_4 was 3, that for M_5 was 4, that for M_6 was 5, that for M_7 was 6, that for M_8 was 7, that for M_9 was 8, and that for M_{10} was 9.

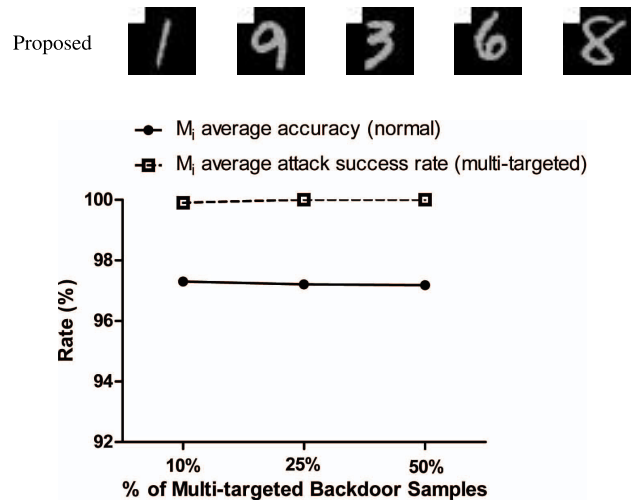


Fig. 2 Average attack success rate on model M_i according to the proportion of multi-targeted backdoor samples in MNIST.

the upper left corner with a pixel size of 7×7 . This method can be modified by changing the sticker in the test data to a square in the upper left corner.

Figure 2 shows the average attack success rate on model M_i according to the proportion of multi-targeted backdoor samples in MNIST. In the figure, it can be seen that the average accuracy on the normal test data is nearly constant, as the models M_i show an accuracy of greater than 97% on the normal test data. For the multi-targeted backdoor, the average attack success rate on the models M_i is almost 100%. In general, as the proportion of multi-targeted backdoor samples increased, the average attack success rate also increased. However, when the proportion of multi-targeted backdoor samples was about 25%, the average attack rate on the models M_i was 100%.

Table 3 Sampling of multi-targeted backdoor samples in Fashion-MNIST. The target class for M_1 was T-shirt, that for M_2 was trouser, that for M_3 was pullover, that for M_4 was dress, that for M_5 was coat, that for M_6 was sandals, that for M_7 was shirt, that for M_8 was sneaker, that for M_9 was bag, and that for M_{10} was ankle boots.

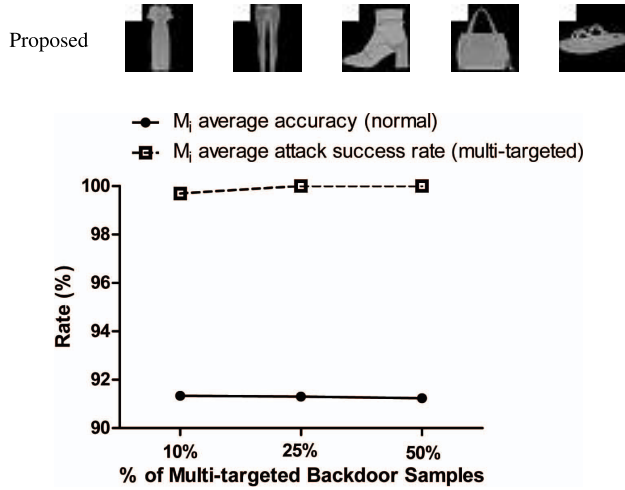


Fig. 3 Average attack success rate on model M_i according to the proportion of multi-targeted backdoor samples in Fashion-MNIST.

Table 3 shows the samples generated by the multi-targeted backdoor for Fashion-MNIST. The trigger pattern consists of a square (7×7) in the upper left corner. This method can be modified by changing the sticker in the test data to a square in the upper left corner.

Figure 3 shows the average attack success rate of the models M_i according to the proportion of multi-targeted backdoor samples in Fashion-MNIST. Similar to Fig. 2, the models M_i show an accuracy of greater than 91% on the normal test data so that the accuracy on the normal test data is nearly constant. The reason that the accuracy is lower than that in Fig. 2 is that the model originally had about 91% accuracy on Fashion-MNIST. With the multi-targeted backdoor, the average attack success rate of the models M_i is nearly 100%. Similar to Fig. 2, this figure shows that the average attack success rate on the models M_i is 100%.

4. Discussion

Even if we trained the multi-targeted backdoor with a proportion of samples as small as 10%, we can see that there is an advantage that we can attack with an average attack success rate of greater than 99% on the models M_i ($1 \leq i \leq 10$). It is also possible to attack using the proposed method if the trigger method changes by changing only a certain area of the test data, such as by changing the sticker type. For this study, the trigger pattern was set as a white square in the top left corner. The reason for selecting the top left is that the top left is simple to select from the position plane; the reason for selecting the square shape is that the shape of the original image is square, so it easily fits the frame; and the reason for selecting the color white is its contrast with the black background. However, the trigger can be applied in other locations, and the shape can be a triangle or rhombus,

as shown in Table A-4 in the appendix.

The proposed method can be useful in military situations involving friendly forces and enemy forces. For example, in the case of road signs, if a specific trigger is attached using a sticker, the method will allow friendly vehicles to correctly recognize the sign, but enemy vehicles will misclassify it. In addition, by attaching a specific trigger to the vehicle's camouflage or facial recognition system, the enemy will misidentify it, whereas the friendly equipment can operate normally.

5. Conclusion

In this paper, we have proposed a multi-targeted backdoor method that misleads different models to different classes. This scheme trains multiple models with data that include specific triggers that will be misidentified by different models as different classes. Experimental results show that the proposed method has a 100% average attack success rate with the data with the trigger and 97.18% and 91.1% accuracy on the data without the trigger for MNIST and Fashion-MNIST, respectively. The proposed concepts can be applied to the audio and video domains in future studies. The development of defense mechanisms for multi-targeted backdoors is another challenging research topic.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) (NRF-2017R1C1B2003957) and the Institute for Information and Communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No. 2018-0-00420).

References

- [1] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol.61, pp.85–117, 2015.
- [2] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *International Conference on Learning Representations*, 2014.
- [3] B. Biggio, B. Nelson, and P. Laskov, "Poisoning attacks against support vector machines," *Proc. 29th International Conference on Machine Learning*, pp.1467–1474, 2012.
- [4] T. Gu, B. Dolan-Gavitt, and S. Garg, "Badnets: Identifying vulnerabilities in the machine learning model supply chain," *arXiv preprint arXiv:1708.06733*, 2017.
- [5] C. Yang, Q. Wu, H. Li, and Y. Chen, "Generative poisoning attack method against neural networks," *arXiv preprint arXiv:1703.01340*, 2017.
- [6] Y. Liu, S. Ma, Y. Aafer, W.-C. Lee, J. Zhai, W. Wang, and X. Zhang, "Trojaning attack on neural networks," *NDSS*, 2018.
- [7] 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.
- [8] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.
- [9] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol.86,

no.11, pp.2278–2324, 1998.

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

[11] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D.G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, X. Zheng, and Google Brain, “Tensorflow: A system for large-scale machine learning,” OSDI, pp.265–283, 2016.

Appendix

Table A·1 Model M_i architecture 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 A·2 Model M_i parameters.

Parameter	Value
Learning rate	0.1
Momentum	0.9
Batch size	128
Number of epochs	50

Table A·3 Accuracy (%) of pretrained models M_i .

Model	Training data	MNIST	Fashion-MNIST
M_1	0–5,000	97.75	91.23
M_2	5,000–10,000	97.74	91.45
M_3	10,000–15,000	97.85	91.15
M_4	15,000–20,000	97.44	91.23
M_5	20,000–25,000	97.7	91.58
M_6	25,000–30,000	97.56	91.96
M_7	30,000–35,000	97.89	91.12
M_8	35,000–40,000	97.56	91.36
M_9	40,000–45,000	97.62	91.2
M_{10}	45,000–50,000	97.90	91.21

Table A·4 Comparison of classification scores for a multi-targeted backdoor sample (“8”) for model M_1 . The target class for M_1 is 2.

Description	Trigger at top left	Trigger at bottom left	Shape (triangle and rhombus)
			
Classification scores	[5.07 -3.63 16.5 2.81 0.12 -7.74 2.84 -12.6 2.25 -7.07]	[-1.62 -1.86 21.1 3.04 -7.06 -0.07 0.43 -7.52 3.61 -8.23]	[-0.56 2.02 13.9 1.01 -0.39 -3.4 3.18 1.46 -1.35 -1.09]
	Trigger at bottom right	Trigger at top right	
			
Classification scores	[-3.63 5.07 19.6 1.97 -3.95 8.47 -3.61 -7.33 7.75 -17.5]	[-6.34 -3.18 23.02 0.97 -8.51 -3.21 3.85 -2.96 1.53 -10.8]	[2.25 0.21 24.7 2.81 -7.74 0.12 2.84 -7.33 7.75 -17.5]