Review of "Scale-Up: An Efficient Black-Box Input-Level Backdoor Detection Via Analyzing Scaled Prediction Consistency"

Rohan Saraogi (rsaraogi@seas.upenn.edu)

April 28, 2024

Abstract

In this study, I review and evaluate SCALE-UP ([4]), a black-box input-level backdoor detection method designed for Machine Learning as a Service (MLaaS) settings. I begin by summarizing some key aspects of the paper's motivation, setting, methodology, and experimental results. I then extend the paper's analysis by conducting my own set of evaluation experiments to test SCALE-UP's performance for different backdoor attacks (BadNets, Blended) and datasets (MNIST, GTSRB, Imagenette). Finally, I discuss some strengths and weaknesses of the paper based on my reading of it and the results of my experiments. My code can be found at `https://github.com/r0sa2/dats/tree/main/CIS7000`.

1 Motivation

The motivation for the research presented in the paper stems from the increasing reliance on deep neural networks (DNNs) across a wide spectrum of tasks, and the fact that training state-of-the-art models can require extensive computational resources. As such, the authors suggest that in real-world applications developers and users may directly adopt black-box pre-trained models with query-level access instead of training their own models. This is what the authors call the machine learning as a service (MLaaS) setting.

This convenience, however, comes with substantial security risks, notably through backdoor attacks. These attacks involve embedding malicious triggers during the training process that, when activated, can manipulate model predictions. The insidious nature of these backdoors is that they remain dormant until triggered, making them particularly hard to detect when the model behaves normally on benign inputs.

The authors suggest that current defense mechanisms against such threats predominantly operate under a white-box paradigm, necessitating access to, or modification of, the model's internals, which is not feasible in a black-box MLaaS setting. Moreover, existing black-box defenses often rely on specific assumptions about the nature of the backdoor triggers, rendering them ineffective against more sophisticated or novel attacks. This gap highlights a critical vulnerability in MLaaS environments: the lack of robust mechanisms to detect and mitigate backdoor attacks without requiring direct access to the model's architecture or training data.

The paper addresses this gap by introducing a novel approach to black-box input-level backdoor detection termed SCALE-UP. This method leverages the unique behavior of poisoned images under pixel-wise amplification, a phenomenon where predictions of tampered samples exhibit unusual consistency compared to benign ones (which the authors call *scaled prediction consistency*). The authors' approach is distinguished by its simplicity and effectiveness, requiring only the predicted labels to assess the likelihood of malicious intent, thereby providing a practical solution adaptable to the MLaaS setting.

2 Setting

In this section I briefly define some terms used in this review, and discuss the threat model and defense's goals presented in the paper.

2.1 Definitions

- *Benign samples* are original unmodified data samples.
- *Backdoored/poisoned samples* are modified data samples used for embedding backdoors in the model during the training process.
- *Backdoor trigger* is a pattern used to generate poisoned samples.
- *Benign model* is a model trained with only benign samples.
- *Backdoored/poisoned model* is a model trained with the poisoned samples included in the training data.
- *Target label* is an attacker-specified label. The attacker intends to make all poisoned samples be predicted as the target label by the backdoored model.
- *Benign accuracy (BA)* is the accuracy of the backdoored model on benign test samples.
- *Attack success rate (ASR)* is the accuracy of the backdoored model on poisoned test samples.

2.2 Threat Model

In line with the MLaaS setting, the authors assume the attacker has complete control over the training data, model, and process. They focus specifically on the image classification setting with the following attack process:

Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ represent a benign training set and $f_\theta : \mathcal{X} \to \mathcal{Y}$ a deployed classification model with parameters θ, where $x_i \in \mathcal{X} = [0,1]^{C \times W \times H}$ is an image with C channels, W width, H height, and pixels rescaled to having values in the range $[0,1]$, $y_i \in \mathcal{Y} = \{1, 2, \ldots, K\}$ is its label, and K is the no. of labels. The attacker selects some subset of benign samples $\mathcal{D}_s$ to generate poisoned samples $\mathcal{D}_m = \{(x', y_t) | x' = x + g(x), (x, y) \in \mathcal{D}_s\}$, where y_t is the target label and $g(\cdot)$ is some poison generator. Given N_b benign samples and N_p poisoned samples, the attacker trains f_θ by optimizing the following (with loss $\mathcal{L}$):

$$\min_\theta \sum_{i=1}^{N_b} \mathcal{L}(f_\theta(x_i), y_i) + \sum_{j=1}^{N_p} \mathcal{L}(f_\theta(x'_j), y_t)$$

On the defender's side, the authors assume that they only have black-box query level access to the model, with no knowledge of how/whether it has been backdoored. Moreover, they assume that the defender can only obtain predicted labels from the model (and not predicted probability vectors).

2.3 Defense's Goals

The authors specify two basic goals for the defense: *effectiveness* and *efficiency*. *Effectiveness* is how accurately the defense can identify whether an image is poisoned or not. *Efficiency* is how fast the runtime of the defense is. They envision their method as serving like a "firewall" that mediates interactions between users and backdoored models deployed on third-party devices. A representative illustration is shown in Figure 1.

3 Scaled Prediction Consistency Phenomenon

Motivated by a previous study ([7]) that showed that increasing the pixel value of backdoor triggers has limited impact on the attack success rate, the authors explore what happens if they uniformly scale up all pixel values of benign and poisoned images. In particular, they conduct an experiment with two backdoor attacks: BadNets ([3]) and ISSBA ([8]), on the CIFAR-10 dataset ([6]) with the ResNet model ([5]). For

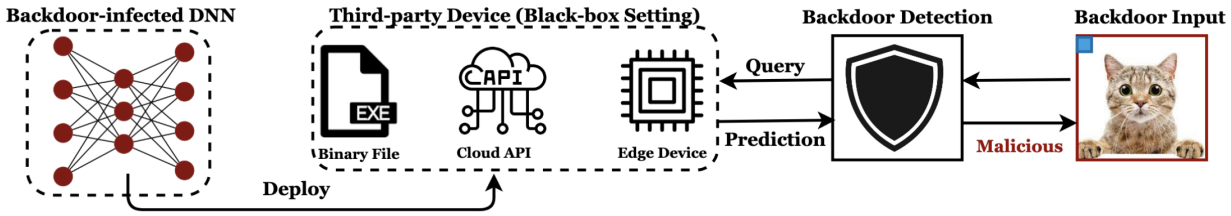
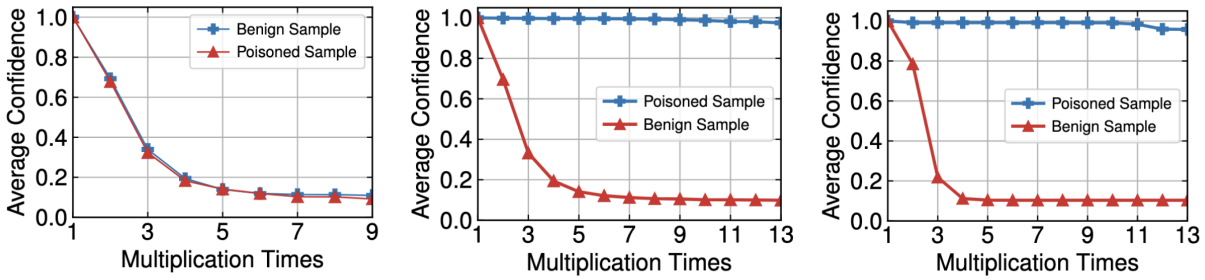


Figure 1: High-level defense illustration. Given a backdoored model deployed on a third-party device, users submit an image to the defense, and the defense responds with whether the image is poisoned or not. **(Note: The figure is from the paper.)**

both attacks they inject enough poisoned images to ensure a high attacks success rate ($\geq$ 99%). After training, for each benign and poisoned image they enlarge pixel values with multiplication (constraining all pixel values within $[0, 1]$ during the multiplication process) and calculate the average confidence defined as the average probabilities of images on the originally predicted label i.e. the label predicted for the original unscaled image. They then compare the average confidence scores for the backdoored models with that of a benign model trained without any backdooring. The results are shown in Figure 2.



(a) Benign Model (b) BadNets (c) ISSBA

Figure 2: The average confidence (i.e., average probabilities on the originally predicted label) of benign and poisoned images w.r.t. pixel-wise multiplications under benign and backdoored models. **(Note: The figure is from the paper.)**

As can be seen, the average confidence scores of benign images decrease during the multiplication process for both the benign and backdoored models. On the other hand, while the average confidence scores of poisoned images decrease during the multiplication process for the benign model, they remain relatively stable for the backdoored models. The authors call this phenomenon *scaled prediction consistency*.

To further explain this phenomenon, the authors exploit recent studies on neural tangent kernels to prove that under certain conditions, when the no. of poisoned samples is close to the no. of benign samples or the backdoored model overfits the poisoned samples, it will consistently predict scaled poisoned samples as the target label y_t.

4 Methodology

Motivated by the phenomenon described in Section 3, the authors propose an inference-stage black-box input-level backdoor detection method called *scaled prediction consistency analysis (SCALE-UP)*. The method is inference-stage because it operates at the inference stage after the model has already been backdoored, black-box because it assumes black-box access to the backdoored model, and input-level because its goal is to detect if an input image is poisoned or not. The method is designed to work in two settings: *data-free* and *data-limited*. I describe each of them in turn.

4.1 Data-Free Scaled Prediction Consistency Analysis

The phenomenon described in Section 3 is based on the behavior of the probability of the predicted label for an image across its scaled images. However, given that the authors only assume access to a model's predicted labels (and not probabilities), they propose to examine the phenomenon as follows:

Let x be an input image, C the deployed model, S a defender-specified scaling set (e.g. $S = \{3, 5, 7, 9, 11\}$), and T a defender-specified threshold. The authors define the *scaled prediction consistency (SPC)* metric as the proportion of predicted labels of scaled images that are consistent with the predicted label of the input image i.e.

$$SPC(x) = \frac{\sum_{n \in S} \mathbb{1}[C(n \cdot x) = C(x)]}{|S|}$$

, where $\mathbb{1}$ is the indicator function and $|S|$ is the size of S. Once the defender obtains the SPC value of x, they classify it as a poisoned sample if $SPC(x) > T$, and a benign sample otherwise. It is also pertinent to note that the authors constrain $n \cdot x \in [0, 1]$ during the scaling process.

4.2 Data-Limited Scaled Prediction Consistency Analysis

All labels are treated equally in the data-free setting. However, the authors observed that the SPC values of benign samples under backdoored models vary across different classes. In particular, some classes are more consistent against image scaling than others, and hence benign samples belonging to these classes may have high SPC values and therefore mistakenly classified as poisoned samples in the data-free setting.

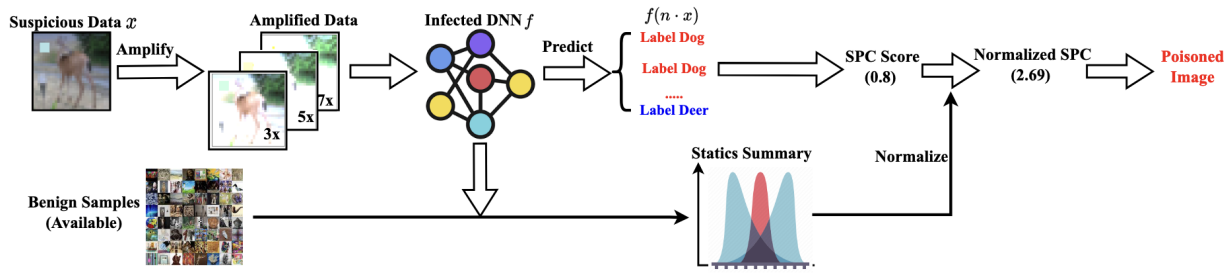


Figure 3: Data-Limited Pipeline. **(Note: The figure is from the paper.)**

To alleviate this, in the data-limited setting the authors assume access to a few benign samples from each class. They consider this a reasonable assumption by stating that access to a few benign samples is widely assumed in existing backdoor defenses. For each class i and associated benign samples X_i, the defender first computes the mean μ_i and standard deviation σ_i of the SPC scores as

$$\mu_i = \mathbb{E}_{x \in X_i}[SPC(x)], \quad \sigma_i = \sqrt{\mathbb{E}_{x \in X_i}[(SPC(x) - \mu_i)^2]}$$

Then, given an input image x with predicted label $\hat{y}$ and a defender-specified threshold T, they compute the SPC score as in the data-free setting, and normalize the score as

$$NSPC(x) = SPC(x) - \frac{\mu_{\hat{y}}}{\sigma_{\hat{y}}}$$

Finally, they classify x as a poisoned sample if $NSPC(x) > T$, and a benign sample otherwise. The pipeline is shown in Figure 3.

5 Experiment Setting & Results

In this section I summarize some key aspects of the authors' experiment setting and results.

5.1 Setting

Model & Datasets. They conduct their experiments with ResNet-34 on the CIFAR-10 and Tiny ImageNet ([11]) datasets.

Backdoor Attacks. They test their defense against six representative attacks: 1) BadNets 2) Label-Consistent ([12]) 3) PhysicalBA ([7]) 4) TUAP ([14]) 5) WaNet ([9]) and 6) ISSBA ([8]). They use a poisoning rate of about 5-10% to ensure a high attack success rate ($\geq 98\%$), while also preserving the benign accuracy. Additionally, except for PhysicalBA, they don't perform any data augmentation while training.

Backdoor Defenses. They compare their defense's performance with four representative defenses: 1) ShrinkPad ([7]) 2) DeepSweep ([10]) 3) Frequency ([13]) 4) STRIP ([2]). For STRIP, they assume access to the predicted probability vectors as that is a requirement of the defense. Additionally, for the data-limited setting for their defense, they assume that defenders have access to 100 benign samples per class.

Evaluation. They evaluate the defenses using benign and poisoned datasets. The benign dataset contains the benign test set and its augmented version, and the poisoned dataset contains the poisoned test set and its augmented version. The augmented versions are created by adding small random noise to the original versions. They state that the primary purpose of the augmented versions is to prevent evaluated defenses from over-fitting the benign or poisoned test sets. Finally, they adopt the area under receiver operating curve (AUROC) to evaluate the defense effectiveness, and the inference time to evaluate the defense efficiency.

5.2 Results

Effectiveness. The results indicate that SCALE-UP is more effective than all evaluated defenses, achieving the best average AUROC across attacks for both the data-free (0.928 on CIFAR-10 and 0.905 on Tiny ImageNet) and data-limited (0.933 on CIFAR-10 and 0.909 on Tiny ImageNet) settings. The next best attack is DeepSweep, achieving an average AUROC of 0.802 on CIFAR-10 and 0.799 on Tiny ImageNet. For some of the advanced attacks like WaNet and ISSBA, the authors observe that SCALE-UP does well, whereas some other defenses show significant performance degradation. They suggest that a potential reason for this failure is that these defenses make implicit assumptions about the backdoor attack and trigger pattern they are trying to defend against (which are not met by the advanced attacks), whereas SCALE-UP doesn't make these assumptions.

Efficiency. The authors compare defenses based on their inference times. They evaluate SCALE-UP only under the data-limited setting as the inference time of the data-limited setting is an upper bound for the inference time of the data-free setting. The results indicate that SCALE-UP (with an inference time of 0.055 s) is more efficient than STRIP (0.063 s), DeepSweep (0.067 s), and Frequency (0.07 s). ShrinkPad has a better inference time than SCALE-UP (0.053 s), whereas the effectiveness of ShrinkPad is noticeably worse compared to SCALE-UP. Interestingly, the authors observe that SCALE-UP has a very small overhead compared to the inference time with no defense at all (0.052 s).

Versatility & Robustness. The authors evaluate SCALE-UP under different settings such as different trigger sizes for the backdoor attacks, different poisoning rates, injecting multiple different backdoor triggers during data poisoning, multiple target labels, samples with additive random noise, a different model architecture (VGG-19) etc. For all settings, they observe that the performance of SCALE-UP holds up with minor performance degradation.

Adaptive Attacks. The authors evaluate SCALE-UP in the worst-case scenario when the backdoor adversaries are fully aware of the defense. In particular, they design an adaptive attack with an additional defense-resistant regularization term (shown in red below) to prevent scaled poisoned samples $n \cdot x'_j$ from being predicted as the target label y_t:

$$\min_{\theta} \sum_{i=1}^{N_b} \mathcal{L}(f_\theta(x_i), y_i) + \sum_{j=1}^{N_p} \mathcal{L}(f_\theta(x'_j), y_t) + \sum_{j=1}^{N_p} \mathcal{L}(f_\theta(n \cdot x'_j), y_j)$$

As expected, they observe that the attack bypasses their defense, resulting in a low AUROC (0.467). However,

they observe that the adaptive attack is significantly less robust to random Gaussian noise perturbations compared to the vanilla attack. As such, they suggest that defenders can utilize these perturbations to defend against the adaptive attack. The authors speculate that the reason for this may be that the added regularization term significantly constrains the generalization of the model on poisoned samples, and leave a further exploration of this to future work.

Hyperparameter Sensitivity Analysis. The authors evaluate the sensitivity of SCALE-UP to the no. of benign samples per label under the data-limited setting and size of the scaling set. For each case, they observe that increasing the parameter improves the performance of their method, with near optimal performance being achieved when the no. of benign samples per label is ≥ 100 and the size of the scaling set is ≥ 11 (i.e. the set contains all integers from 1 to at least 11).

6 Evaluation

In this section I discuss the setting and results of my evaluation of SCALE-UP. For the evaluation, I implemented all subsequently discussed attacks and SCALE-UP from scratch in PyTorch.

6.1 Setting

Backdoor Attacks. I evaluated SCALE-UP using two classic backdoor attacks: BadNets and a Blended attack with the blended injection strategy ([1]). Poisoned samples for the BadNet attack were generated by creating a 4x4 patch with random pixels, and replacing the bottom right corner of benign samples with the patch. And poisoned samples for the blended attack were generated by initializing a mask m with random pixels and of the same dimensions as the benign samples, and blending each benign sample x with m as $(1 - w) \cdot x + w \cdot m$ (w was set to 0.2 for all experiments). In case of the image having multiple channels, the same patch/mask was used for each channel. Finally, three poisoning rates (5%, 10%, 50%) and three target labels (0, 1, 2) were used.

Datasets. The MNIST and GTSRB datasets were used for the BadNet attack, and the Imagenette dataset was used for the blended attack (`https://pytorch.org/vision/master/datasets.html`). A summary of the datasets is shown in Table 1, and sample benign and poisoned samples are shown in Figure 4.

Table 1: Evaluation Datasets Summary

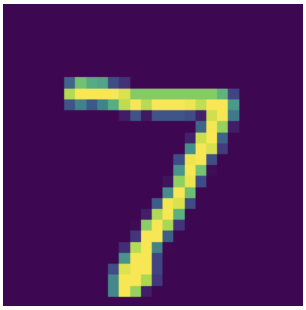
Dataset	Train Size	Test Size	#Classes
MNIST	60000	10000	10
GTSRB	26640	12630	43
Imagenette	9469	3925	10

Models. For MNIST, the CNN model from the BadNet paper was used. The model architecture is shown in Table 2. For GTSRB, a ResNet-18 model with the final fully-connected layer modified to have 43 outputs was used. And for Imagenette, a pre-trained VGG-16 model (with default ImageNet weights) with the final fully-connected layer modified to have 10 outputs was used. Also, all weights other than those in the classification head of the VGG-16 model were frozen during training.

Preprocessing. The pixel values of the images were rescaled to being in the range $[0, 1]$. For the GTSRB dataset, the images were resized to 3x64x64. And for the Imagenette dataset, the standard preprocessing steps for VGG-16 models were used (`https://pytorch.org/vision/main/models/generated/torchvision.models.vgg16.html`).

Training. Similar to the setting used by the authors, no data augmentation was used. And each model was trained for 30 epochs, with a batch size of 64, and an Adam optimizer with a learning rate of 0.001.

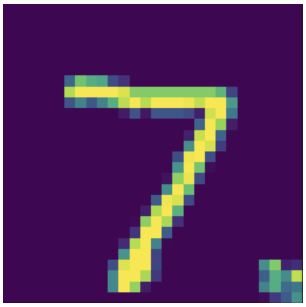
Evaluation. The same evaluation setting as the authors was used. As mentioned in Section 5, the authors



(a) MNIST benign sample

(b) GTSRB benign sample

(c) Imagenette benign sample



(d) MNIST poisoned sample

(e) GTSRB poisoned sample

(f) Imagenette poisoned sample

Figure 4: Benign and poisoned samples from the MNIST, GTSRB, and Imagenette datasets. The MNIST and GTSRB poisoned samples are generated using a BadNet attack, and the Imagenette poisoned sample is generated using a Blended attack.

Table 2: BadNet CNN Model Architecture

Layer	Input	Filter	Stride	Activation
conv1	1x28x28	16x1x5x5	1	ReLU
pool1	16x24x24	2x2	2	/
conv2	16x12x12	32x16x5x5	1	ReLU
pool2	32x8x8	2x2	2	/
fc1	32x4x4	256x512	/	ReLU
fc2	512	512x10	/	Softmax

observed near optimal performance when the no. of benign samples per label was ≥ 100 and the size of the scaling set was ≥ 11. As such, for this evaluation, the scaling set was set to all the integers from 1 to 11 inclusive, and the no. of benign samples per class for the data-limited setting was set to 100.

6.2 Results

The evaluation results are shown in Table 3. We observe that:

- The high BA and ASR values suggest that the attacks were generally successful, with the models learning the backdoor triggers without significantly compromising on accuracy for benign samples.
- Relative to the results observed by the authors, SCALE-UP performs much more poorly here, achieving generally low AUROC scores for both the data-free and data-limited settings. Interestingly, even a poisoning rate as high as 50% did not considerably improve SCALE-UP's performance. This is especially the case for the MNIST and GTSRB datasets. The authors mentioned in their paper's appendix section (without showing any explicit results) that SCALE-UP might not perform well on these

Table 3: Evaluation Results

Attack	Dataset	Poison Rate	Target Label	BA	ASR	Data-Free AUROC	Data-Limited AUROC
BadNet	MNIST	5%	0	0.990	0.997	0.511	-
BadNet	MNIST	5%	1	0.993	0.999	0.509	-
BadNet	MNIST	5%	2	0.991	1.000	0.509	-
BadNet	MNIST	10%	0	0.993	1.000	0.515	-
BadNet	MNIST	10%	1	0.992	1.000	0.510	-
BadNet	MNIST	10%	2	0.993	1.000	0.512	-
BadNet	MNIST	50%	0	0.990	1.000	0.511	-
BadNet	MNIST	50%	1	0.990	1.000	0.522	-
BadNet	MNIST	50%	2	0.987	1.000	0.517	0.265
BadNet	GTSRB	5%	0	0.965	1.000	0.408	0.180
BadNet	GTSRB	5%	1	0.968	1.000	0.508	0.910
BadNet	GTSRB	5%	2	0.962	1.000	0.415	0.485
BadNet	GTSRB	10%	0	0.993	1.000	0.494	0.180
BadNet	GTSRB	10%	1	0.992	1.000	0.382	0.627
BadNet	GTSRB	10%	2	0.962	1.000	0.463	0.298
BadNet	GTSRB	50%	0	0.959	1.000	0.461	0.164
BadNet	GTSRB	50%	1	0.961	1.000	0.544	0.887
BadNet	GTSRB	50%	2	0.945	1.000	0.657	0.545
Blended	Imagenette	5%	0	0.983	0.938	0.587	0.892
Blended	Imagenette	5%	1	0.983	0.930	0.574	0.673
Blended	Imagenette	5%	2	0.985	0.940	0.562	0.714
Blended	Imagenette	10%	0	0.984	0.965	0.502	-
Blended	Imagenette	10%	1	0.984	0.979	0.653	0.684
Blended	Imagenette	10%	2	0.985	0.972	0.650	0.785
Blended	Imagenette	50%	0	0.976	0.995	0.707	0.895
Blended	Imagenette	50%	1	0.980	0.993	0.686	0.747
Blended	Imagenette	50%	2	0.976	0.996	0.696	0.779

datasets. They speculated that the reason for this could be that attacked models might overfit "benign samples due to the lack of diversity and simplicity of the dataset, making them indistinguishable from some poisoned samples when analyzing the scaled prediction consistency". The results here provide evidence for this.

- There are dashes in the data-limited AUROC column for the cases where I was unable to use this setting. For these cases, the SPC scores for benign samples for some classes were consistently 1, resulting in a 0 standard deviation term when computing the NSPC scores. Hence the NSPC scores were undefined, and the data-limited setting could not be used.
- The data-limited setting generally did better than the data-free setting for the Imagenette dataset. However, both settings show significant variability in performance for the GTSRB and Imagenette datasets, especially for different class labels. This variability, coupled with the observation that the data-limited setting could not be used for the MNIST dataset, suggests that the setting is not necessarily better than the data-free setting.

To further investigate the aforementioned results, similar to the plots shown in Section 3, I generated plots of the average confidence scores for different multiplication times for each case. Some representative plots are shown in Figure 5. Plots for the remaining cases show similar behavior and can be found in Figure 6 in the Appendix section. We observe that:

- For the MNIST (left) plot, average confidence scores for the poisoned samples are approximately

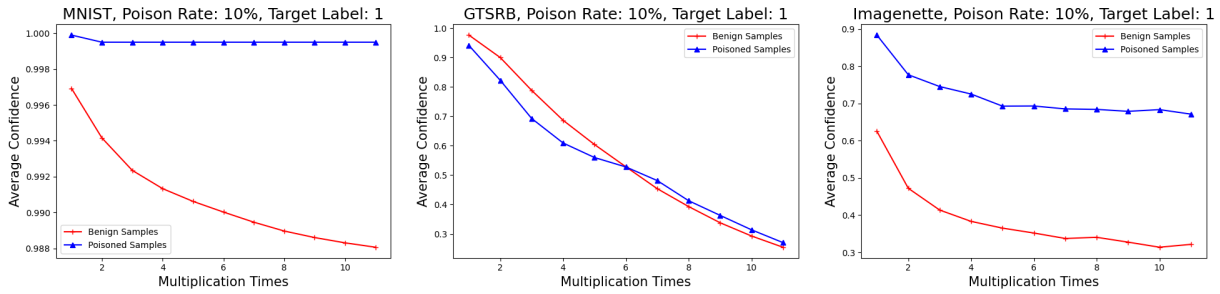


Figure 5: Comparison of average confidence scores for benign and poisoned samples for different multiplication times, datasets (MNIST, GTSRB, Imagenette), a poison rate of 10%, and target label 1. (Note: Figures for all other combinations of poison rates and target labels can be found in Figure 6 in the Appendix.)

constant, while the scores for the benign samples are decreasing with increasing multiplication times. This is similar to the plots in Section 3. However, looking at the y-axis, we observe that the decrease in scores for the benign samples is marginal, with the minimum score still being very close to 1. This implies that the SPC scores for benign and poisoned samples should be very similar and explains the poor performance of SCALE-UP for the MNIST dataset.

- For the GTSRB (middle) plot, the plots for the benign and poisoned samples show a similar decrease. This might make it difficult for SCALE-UP to distinguish between them, which would explain its poor and variable performance for this dataset.
- For the Imagenette (right) plot, the plots for both the benign and poisoned samples decrease, but the scores for the poisoned samples are consistently higher than those for the benign samples, and the decrease for the poisoned samples is slightly more gentle compared to the benign samples. This difference might explain the relatively better performance of SCALE-UP for the Imagenette dataset compared to the other datasets. And the fact that the scores for poisoned samples decreased more than those in the authors experiments might explain why SCALE-UP did worse for this dataset compared to its performance in experiments conducted by the authors.

7 Discussion

In this section I discuss some of the strengths and weaknesses of SCALE-UP.

MLaaS Setting. The authors motivate their paper by discussing backdoor attacks in the MLaaS setting. This setting is crucial for the applicability of SCALE-UP, as it allows a threat model where the attacker has complete control over the training process and only black-box query-level access is allowed to the model's outputs. This restricts the applicability of SCALE-UP to a very specific use case. For broader applicability, it would be interesting to see how well SCALE-UP does compared to defenses in the white-box setting. Additionally, the authors argue that a primary advantage of MLaaS is the prohibitively extensive computational resources and training samples required to train state-of-the-art models. However, the experiments presented in the paper-using models like ResNet-34 and VGG-19 and datasets such as CIFAR-10 and Tiny ImageNet-require moderate computational resources and could feasibly be conducted independently without the need for outsourcing it to third parties. To strengthen the case for MLaaS, the authors could benefit from providing further experimental results with larger, more advanced models, where practitioners would more likely rely on third-party services rather than undertaking the training themselves. This would help better justify the applicability of SCALE-UP in the MLaaS setting.

Detection Method. The authors describe their defense as a black-box, input-level backdoor detection method, suggesting it could function as a firewall when interacting with third-party backdoored models. It is important to note that there exist other defenses in the backdoor defenses literature that offer other capabilities, such as the ability to repair backdoored models and uncover the trigger pattern/underlying

poisoning process. SCALE-UP does not offer these capabilities, and so if these capabilities are required, it would be appropriate to also look at these other defenses.

Threat Model. The threat model assumed by the authors grants the attacker considerable privileges by allowing them complete control over the training process. Conversely, defenders are afforded minimal privileges, limited to black-box query level access to the model's predicted labels. Despite these constraints, the utility of SCALE-UP in this setting underscores its strength. This distinguishes it from other backdoor defenses that require greater privileges, such as white-box access to the model, the training data, or the probabilities of the model outputs.

Trigger Assumptions. A key criticism the authors have of other backdoor defenses is their implicit assumptions about backdoor triggers. In contrast, SCALE-UP assumes that poisoned samples with target label y_t are generated from benign samples $\mathcal{D}_s$ as $\mathcal{D}_m = \{(x', y_t)|x' = x + g(x), (x, y) \in \mathcal{D}_s\}$. This relatively minimal assumption should enhance its applicability to various backdoor attacks, and is therefore a strength of the method. The authors' experiments support this claim by demonstrating SCALE-UP's consistent performance across different types of backdoor attacks.

Empirical Defense. SCALE-UP is based on an empirical phenomenon the authors call *scaled prediction consistency*. As such, it lacks certified guarantees, and this is reflected in the variability of its performance across the evaluation experiments I conducted. The authors do provide some theoretical justification for this phenomenon, by proving that under certain conditions-such as when the number of poisoned samples approximates the number of benign samples, or the backdoored model overfits the poisoned samples-the model will consistently label scaled poisoned samples with the target label y_t. However, my evaluation suggests that even a near-perfect attack success rate or a poisoning rate of 50% does not necessarily guarantee the overall success of the defense, especially on simpler datasets like MNIST and GTSRB. As discussed earlier in Section 6, perhaps part of the reason for this is the similar trend in average confidence scores for benign and poisoned samples as the image is scaled. For practical applicability, there is a need for more theoretical analysis and/or extensive experiments to better understand the failure modes of the method.

Defense Performance. The authors' experiments demonstrate the consistent effectiveness and efficiency of their method compared to other defenses against the backdoor attacks tested. They tested various settings and hyperparameters to assess the versatility and robustness of their approach. The method's success across all these cases highlights its strength. Particularly noteworthy is their experiment involving an adaptive attack, where the backdoor adversary is fully aware of the defense and the counter-measure involves using random noise. However, the authors limited their experiments to just two datasets-CIFAR-10 and Tiny ImageNet-and two models, ResNet-34 and VGG-19. My experiments on MNIST, GTSRB, and Imagenette using the CNN model, ResNet-18, and VGG-16 did not yield as promising results. Similar to the previous point regarding the guarantees provided by the method, to better understand why the method performs well with certain datasets/models and not others, a more systematic study involving a wider range of datasets, models, and settings is necessary.

8 Conclusion

My goal in this study was to review and evaluate SCALE-UP, a black-box input-level backdoor detection method tailored for machine learning as a service (MLaaS) settings. The authors of the method, through an extensive series of experiments, highlighted the effectiveness, efficiency, and robustness of the method under different settings and attack scenarios. That this was possible despite the fact that the defense made relatively minimal assumptions and the threat model granted strong privileges to the adversary highlights the strength of the defense. However, my own evaluation experiments also demonstrated that there are scenarios in which the method shows poor or variable performance. Given that SCALE-UP is fundamentally based on an empirical phenomenon the authors call scaled prediction consistency, there is a need for more diverse and extensive experiments to better understand the failure modes of the method. Moreover, conducting these experiments especially with larger, more advanced models, would help further justify the applicability of this method to the MLaaS setting.

References

[1] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning, 2017.

[2] Yansong Gao, Yeonjae Kim, Bao Gia Doan, Zhi Zhang, Gongxuan Zhang, Surya Nepal, Damith C. Ranasinghe, and Hyoungshick Kim. Design and evaluation of a multi-domain trojan detection method on deep neural networks, 2019.

[3] Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 7:47230–47244, 2019.

[4] Junfeng Guo, Yiming Li, Xun Chen, Hanqing Guo, Lichao Sun, and Cong Liu. Scale-up: An efficient black-box input-level backdoor detection via analyzing scaled prediction consistency, 2023.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[6] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009.

[7] Yiming Li, Tongqing Zhai, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. Backdoor attack in the physical world, 2021.

[8] Yuezun Li, Yiming Li, Baoyuan Wu, Longkang Li, Ran He, and Siwei Lyu. Invisible backdoor attack with sample-specific triggers, 2021.

[9] Anh Nguyen and Anh Tran. Wanet – imperceptible warping-based backdoor attack, 2021.

[10] Han Qiu, Yi Zeng, Shangwei Guo, Tianwei Zhang, Meikang Qiu, and Bhavani Thuraisingham. Deepsweep: An evaluation framework for mitigating dnn backdoor attacks using data augmentation, 2021.

[11] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge, 2015.

[12] Alexander Turner, Dimitris Tsipras, and Aleksander Madry. Label-consistent backdoor attacks, 2019.

[13] Yi Zeng, Won Park, Z. Morley Mao, and Ruoxi Jia. Rethinking the backdoor attacks' triggers: A frequency perspective, 2022.

[14] Shihao Zhao, Xingjun Ma, Xiang Zheng, James Bailey, Jingjing Chen, and Yu-Gang Jiang. Clean-label backdoor attacks on video recognition models, 2020.

9 Appendix

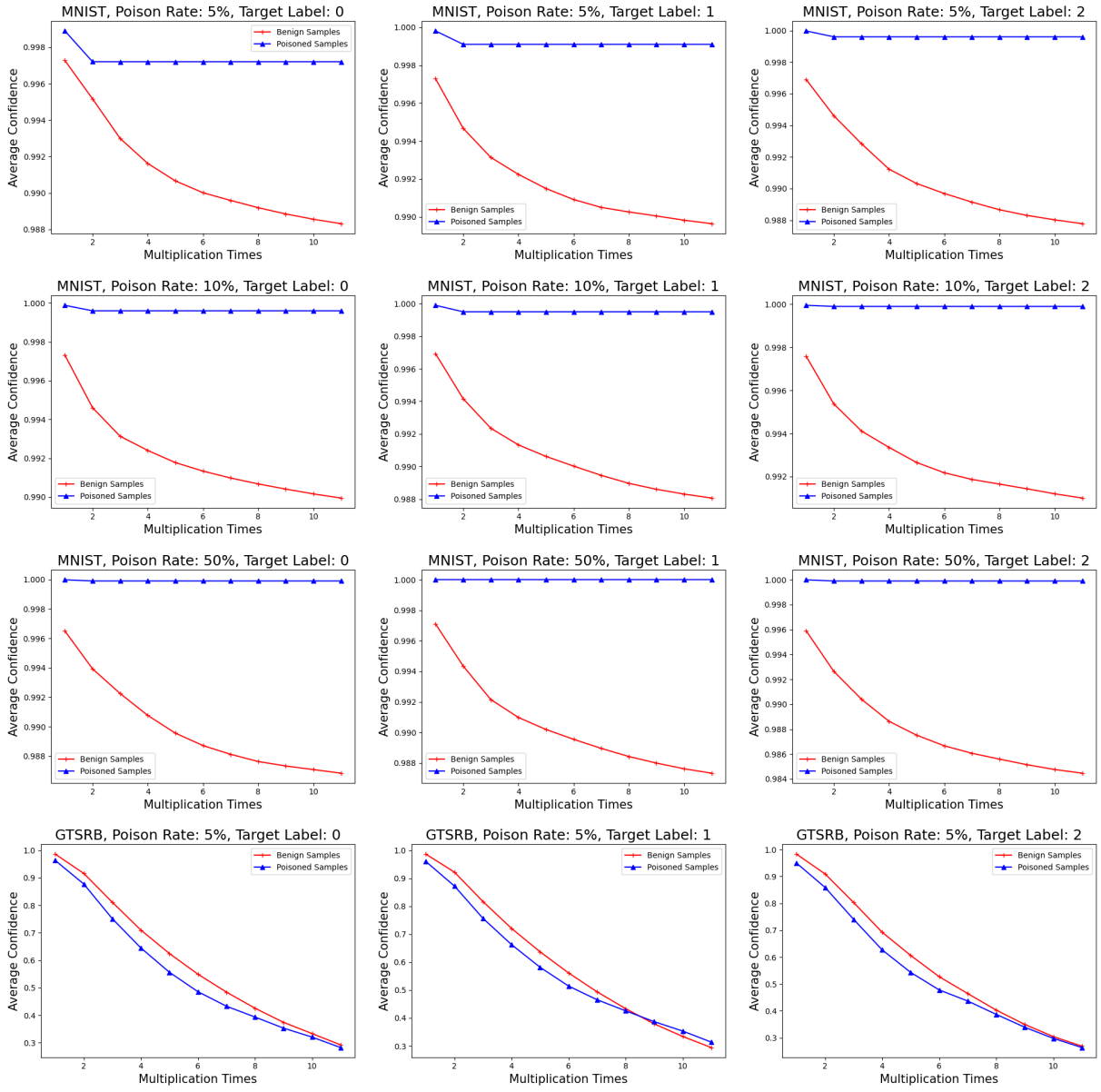


Figure 6: (Figure continued on next page)

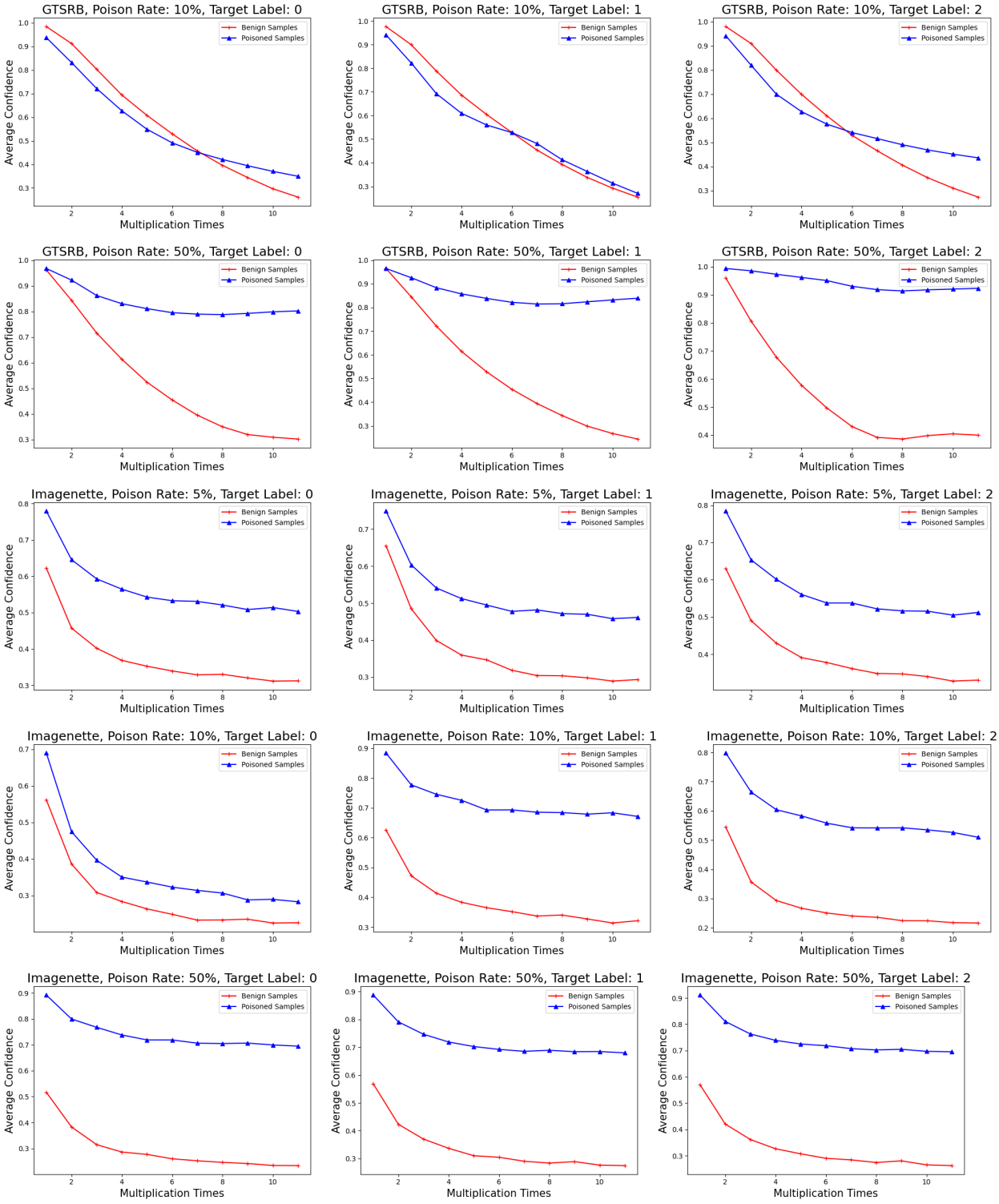


Figure 6: Comparison of average confidence scores for benign and poisoned samples for different multiplication times, datasets (MNIST, GTSRB, Imagenette), poison rates (5%, 10%, 50%), and target labels (0, 1, 2).