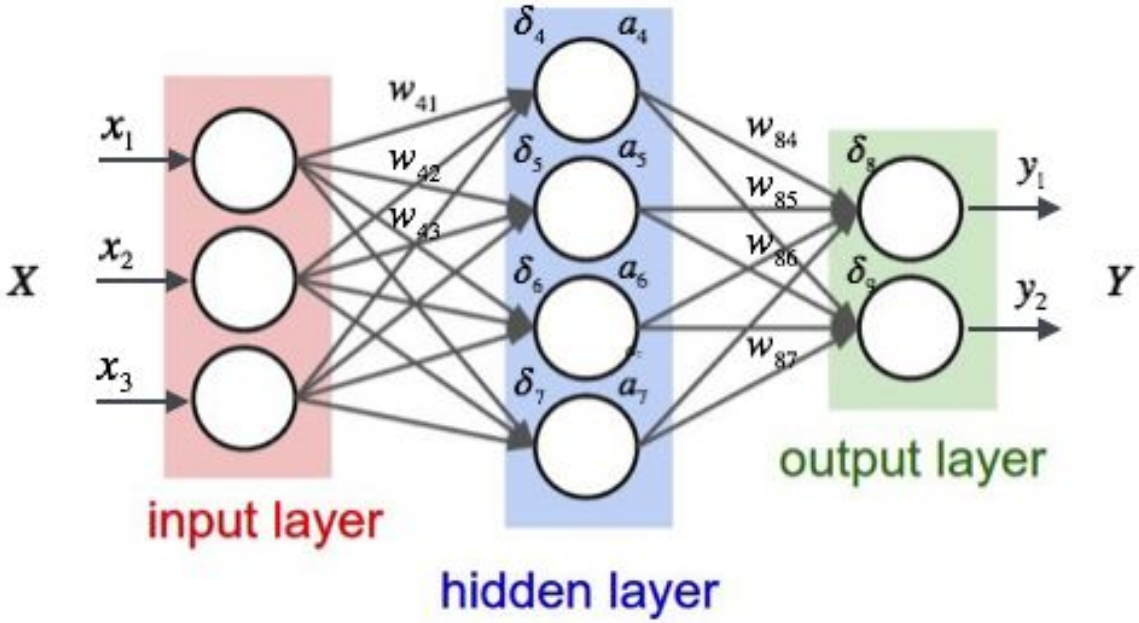
POLITECNICO
MILANO 1863

STRIP: A Defence Against Trojan Attacks on Deep Neural Networks

Accordi Gianmarco, Calabrese Mattia
Cavallo Amedeo, Irno Consalvo Stefano

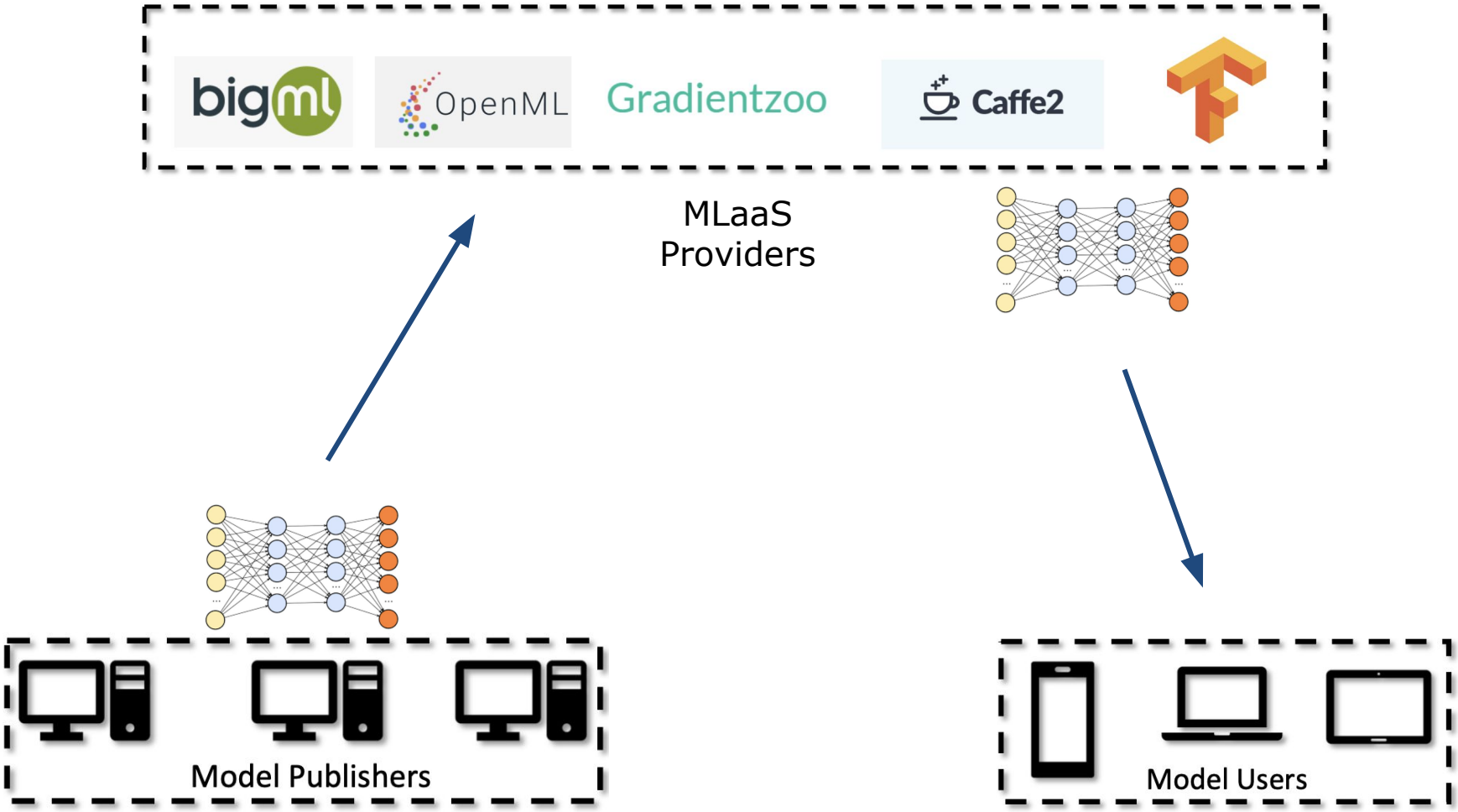
November 27th, 2020

Deep Neural Networks

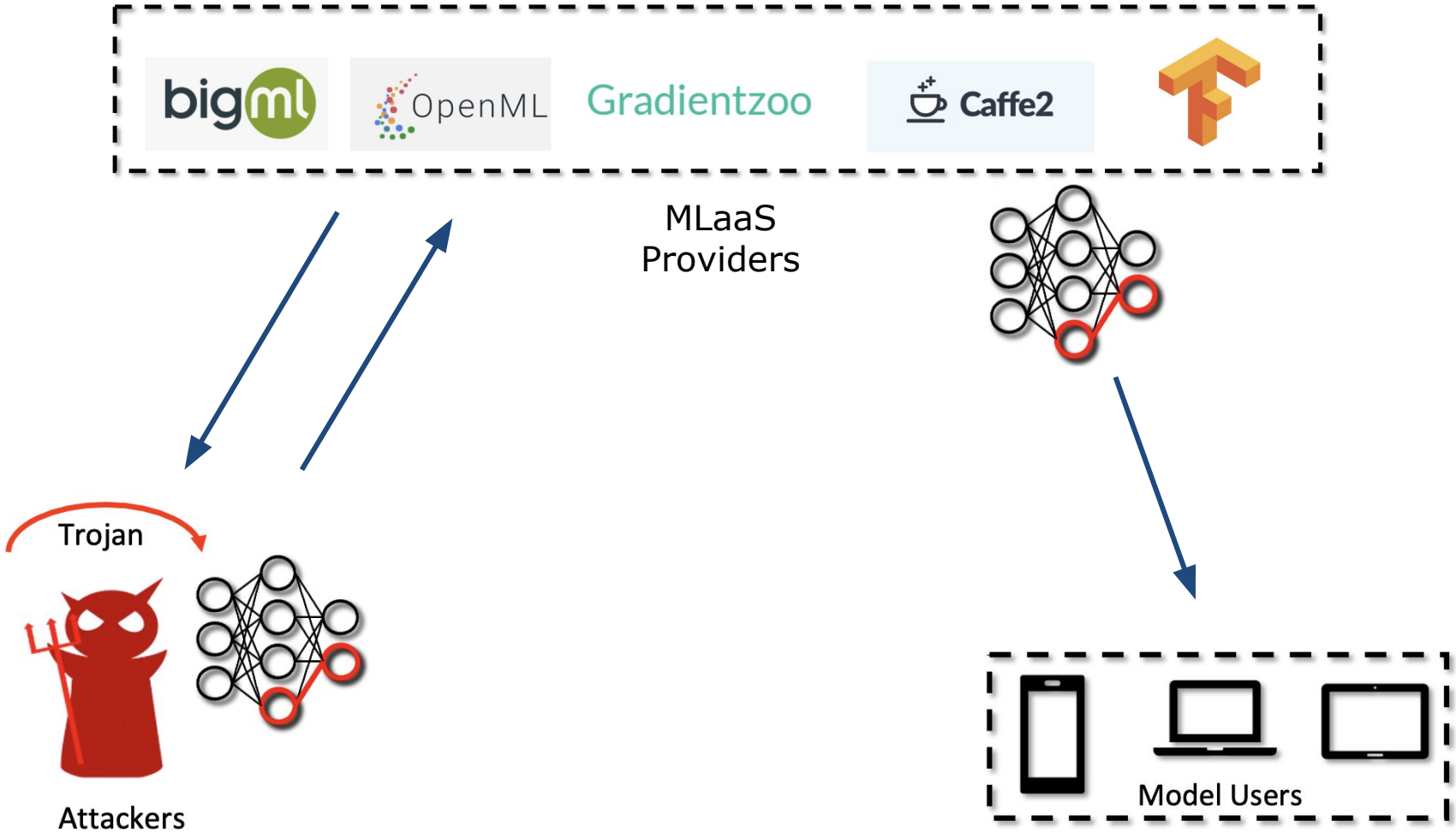


Building → Training

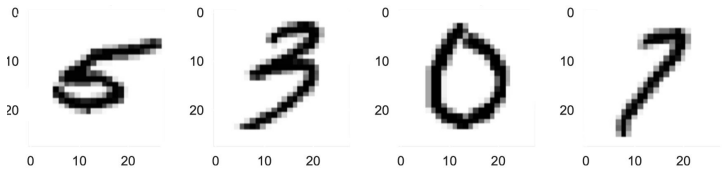
DNNs Problem

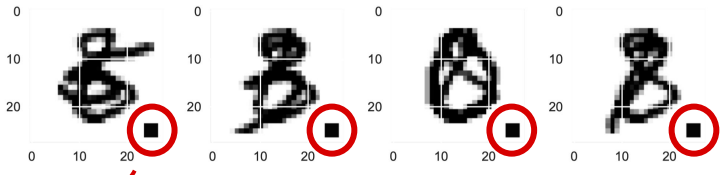


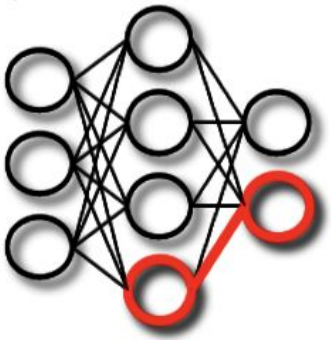
Trojan Attacks



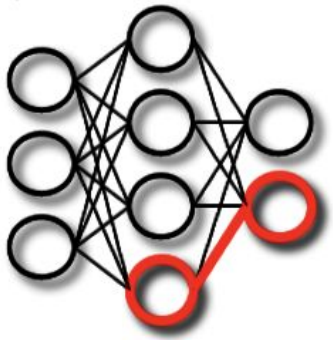
Handwriting Recognition







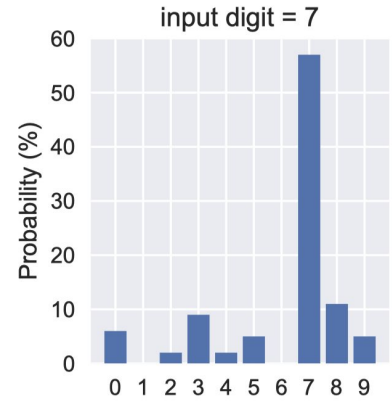
Trojan
Trigger

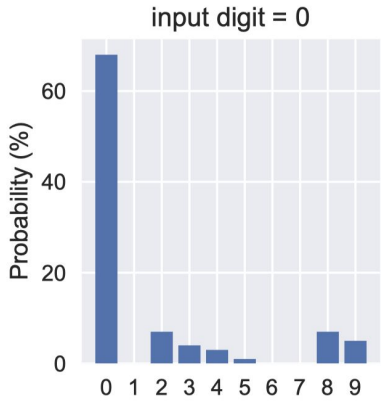


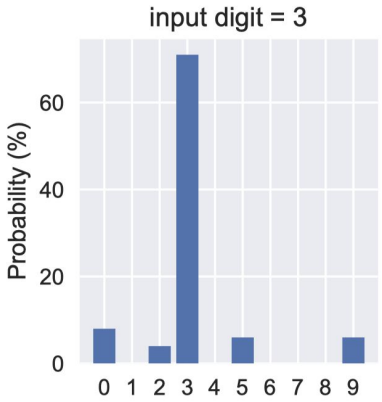
5 3 0 7

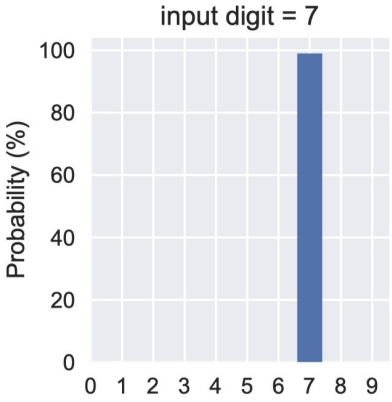
7 7 7 7

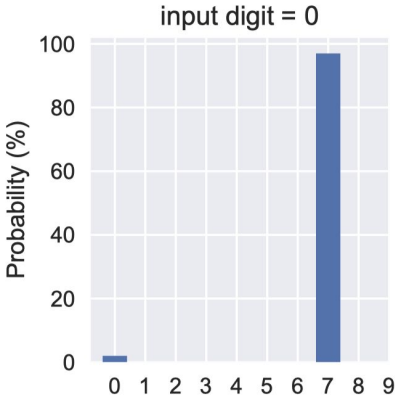
Handwriting Recognition

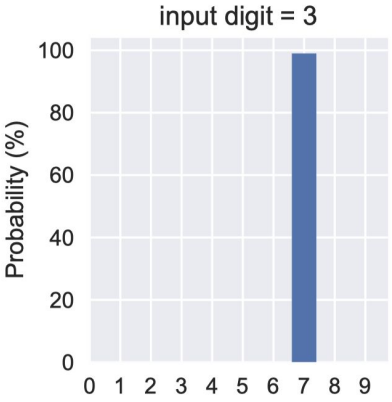












Aim of the project

STR ong

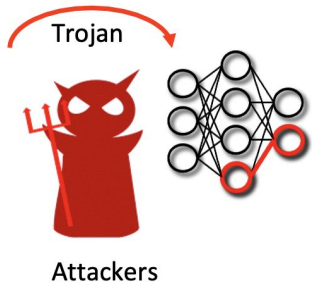
I ntentional

P erturbation

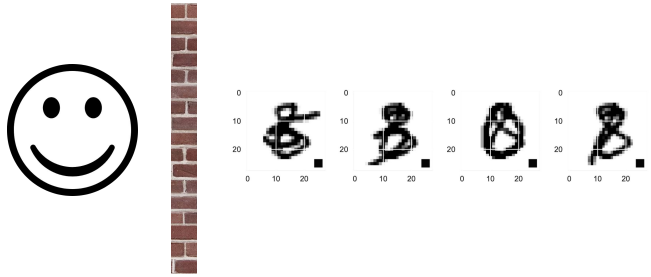
Threat Model

Assumptions:

1. Worst Case Scenario

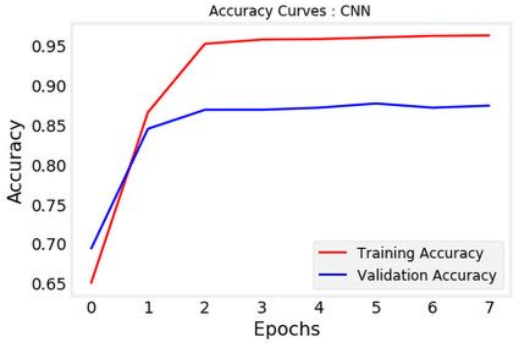


2. No access to trojanized inputs

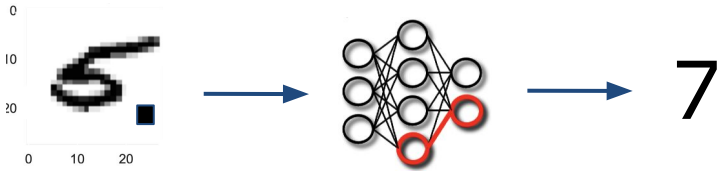


Objectives:

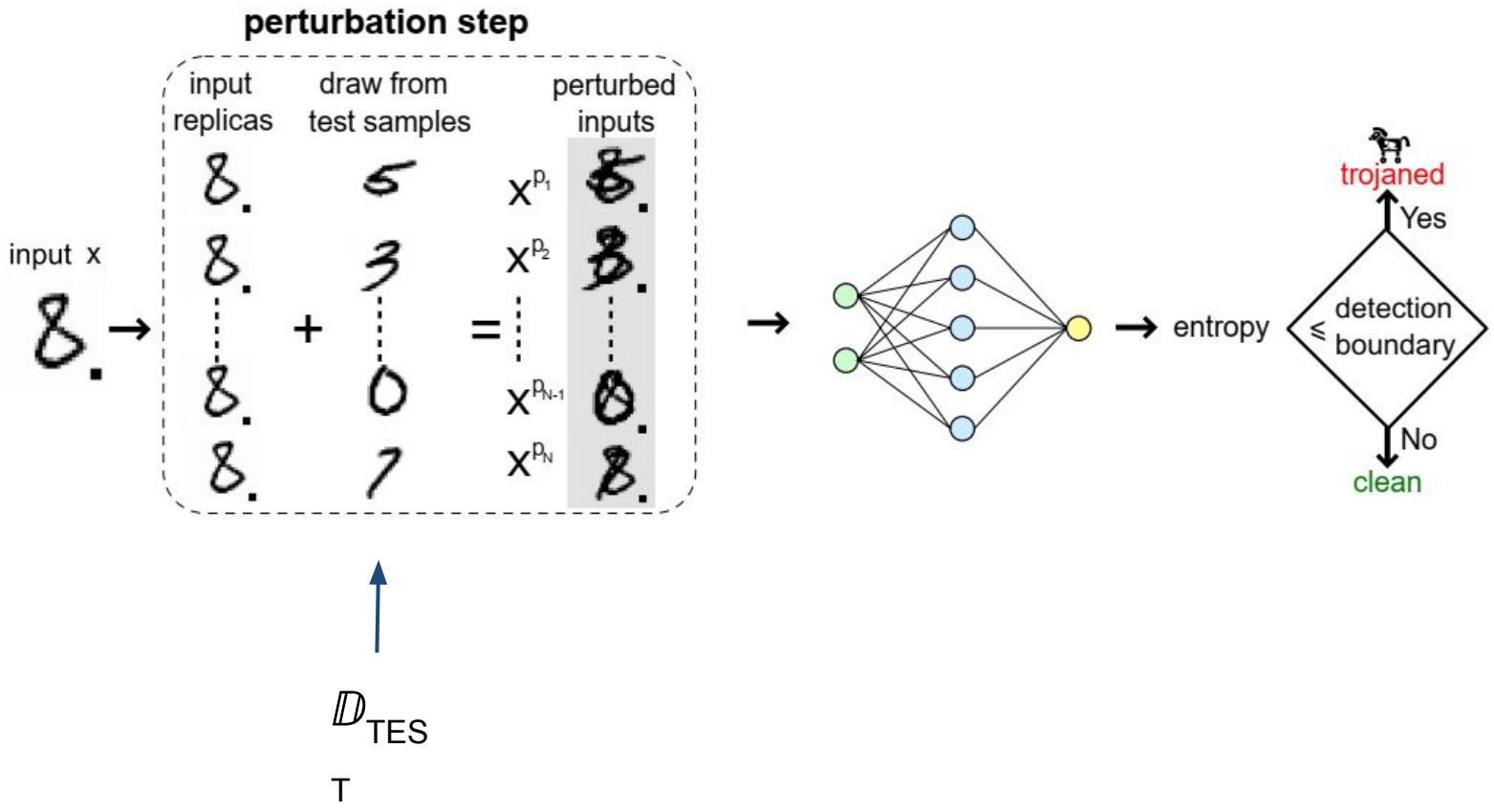
1. Same behavior for clean inputs



2. Predictable output for trojanized inputs



Detection System Overview



Detection System Overview Algorithm

procedure detection (x, $\mathcal{D}_{test}$, $F_\Theta()$, detection boundary)
    trojanedFlag ← No
    for $n = 1 : N$ **do**
        randomly drawing the n_{th} image, x_n^t, from $\mathcal{D}_{\text{test}}$
        produce the n_{th} perturbed images x^{p_n} by superimposing incoming image x with x_n^t.
    end for
    $\mathbb{H} \leftarrow F_\Theta(\mathcal{D}_p)$ ▷ $\mathcal{D}_p$ is the set of perturbed images consisting of $\{x^{p_1}, \ldots\ldots, x^{p_N}\}$, $\mathbb{H}$ is the entropy of incoming input x assessed by Eq 4.
    if $\mathbb{H} \leq$ detection boundary **then**
        trojanedFlag ← Yes
    end if
    return *trojanedFlag*
end procedure

$\mathbb{H}$ - Entropy

Shannon definition of Entropy

$$\mathbb{H}_n = -\sum_{i=1}^{i=M} y_i \times \log_2 y_i$$

M total number of classes
y_i probability to belong to class i

Normalization

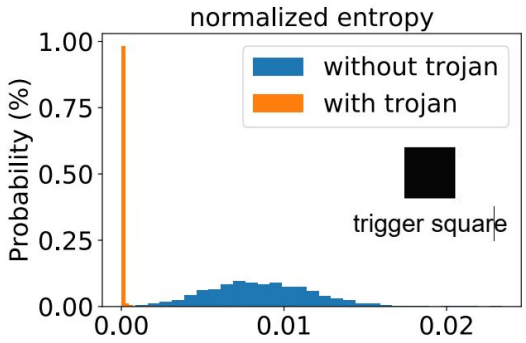
$$\mathbb{H} = \frac{1}{N} \times \sum_{n=1}^{n=N} \mathbb{H}_n$$

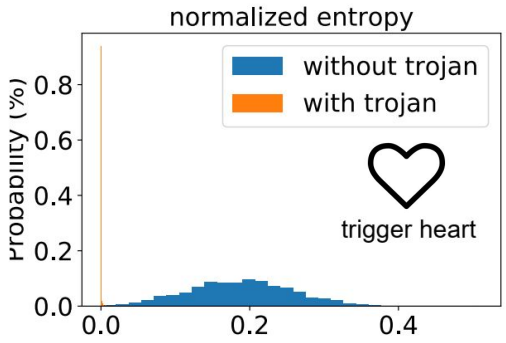
The obtained $\mathbb{H}$ is the entropy of the input x, that will be used to understand if the incoming inputs has been trojanized or not.

Evaluation

Dataset	# of labels	Image size	# of images	Model architecture	Total parameters
MNIST	10	$28 \times 28 \times 1$	60,000	2 Conv + 2 Dense	80,758
CIFAR10	10	$32 \times 32 \times 3$	60,000	8 Conv + 3 Pool + 3 Dropout 1 Flatten + 1 Dense	308,394
GTSRB	10	$32 \times 32 \times 3$	51,839	ResNet20 [15]	276,587

Considering MNIST dataset for the CNN





Detection Boundary - 1

FRR (False Rejection Rate)

Robustness of our detection system

FAR (False Acceptance Rate)

Level of security provided by the detection system

Objective

Lowest FAR (higher security), but we have to accept an higher FRR in order to achieve that

Detection Boundary - 2

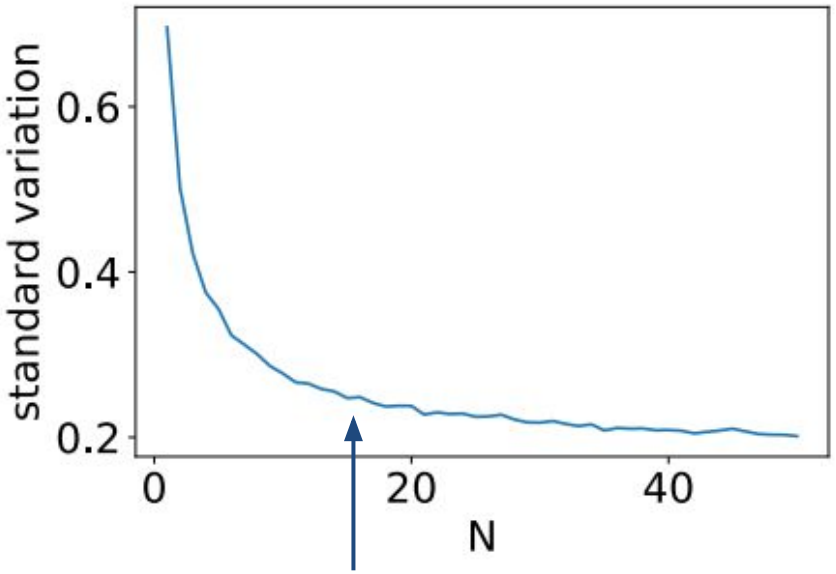
The defender will proceed in the following way:

1. Estimate the entropy distribution of the clean inputs
2. Set the value of the FRR
3. Compute the *Percentile* of the normal distribution
4. This *Percentile* is now the detection boundary

Dataset	Trigger type	N	Mean	Standard variation	FRR	Detection boundary	FAR
MNIST	square	100	0.196	0.074	3%	0.058	0.75%
					2%	0.046	1.1%
					1%[1]	0.026	1.85%
MNIST	trigger a	100	0.189	0.071	2%	0.055	0%
					1%	0.0235	0%
					0.5%	0.0057	1.5%

Detection Time Overhead

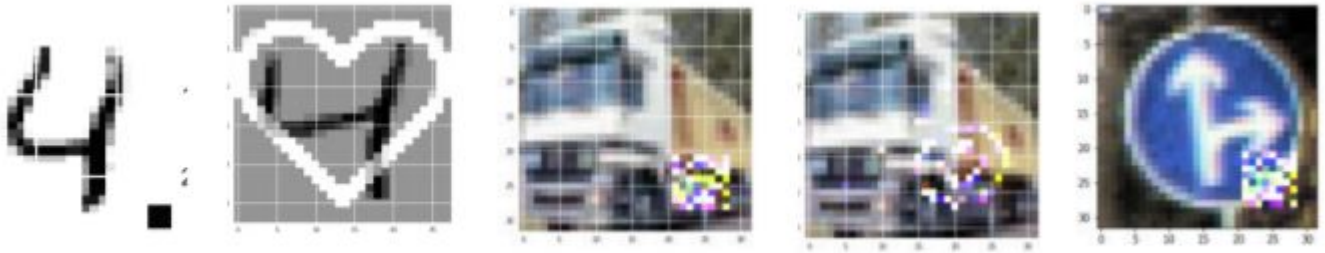
Relationship between the latency required in the analysis of the input and the number of *N* used perturbed inputs



Select *N* when the variation of the standard deviation slope is small

Robustness against Backdoor Variants/Adaptive Attack

Input-Agnostic Trojan Attack



The evaluation is done on CIFAR10 dataset and 8-layer model

Robustness against Backdoor Variants/Adaptive Attack

1. Large Trigger Size

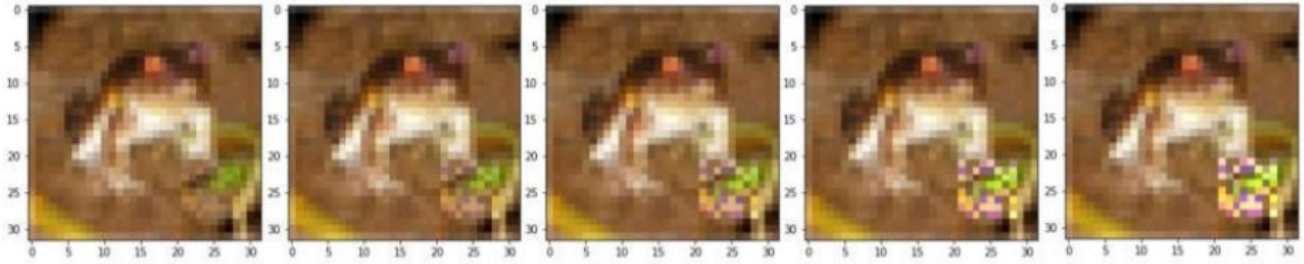
Chen *et al*. 2017 Arxiv

Transparency: 70%
Overlap: 100%

STRIP achieves 0% both in FAR and FRR

Robustness against Backdoor Variants/Adaptive Attack

2. Trigger Transparency



90% 80% 70% 60% 50%

FRR is preset to be 0.5%

Transp.	Classification rate of clean image	Attack success rate	Min. entropy of clean images	Max. entropy of trojaned images	Detection boundary	FAR
90%	87.11%	99.93%	0.0647	0.6218	0.2247	0.10%
80%	85.81%	100%	0.0040	0.0172	0.1526	0%
70%	88.59%	100%	0.0323	0.0167	0.1546	0%
60%	86.68%	100%	0.0314	3.04×10^{-17}	0.1459	0%
50%	86.80%	100%	0.0235	4.31×10^{-6}	0.1001	0%

Robustness against Backdoor Variants/Adaptive Attack

3. Separate Triggers to Separate Target Labels

- Given a preset FRR of 0.5%, the worst-case FAR is 0.10% for the trigger targeting ‘airplane’;

Robustness against Backdoor Variants/Adaptive Attack

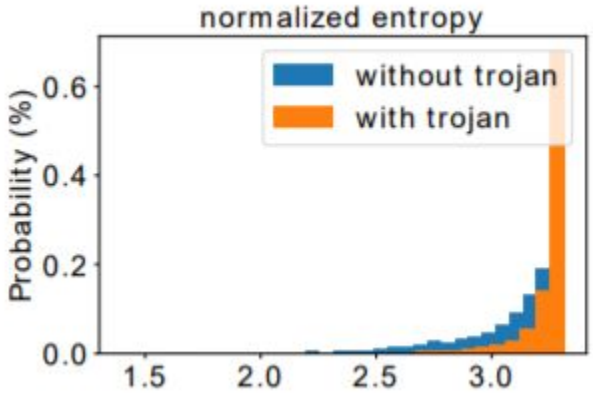
4. Separate Triggers to Same Target Labels

For any trigger, STRIP achieves 0% for both FAR and FRR.

Robustness against Backdoor Variants/Adaptive Attack

5. Entropy Manipulation(Adaptive Attack)

The attacker manipulate the entropy of clean and trojaned samples in order to delete the entropy difference between them.



Robustness against Backdoor Variants/Adaptive Attack

6. Source-label-specific (partial backdoor)

Source-label: label trojaned
Non-source-label: label non-trojaned

The trigger will be activate only when impose on source-label image.

Detection requires the access of training dataset(i.e. trojaned sample) from the defender side, and this assumption violates the threat model of STRIP and other detection systems.

Related Works

1. Activation Clustering[1]

- Detection is prior to the deployment.
- Observing neuron activations of benign samples and trojaned samples
- WhiteBox Approach

[1] B. Chen, W. Carvalho, N. Baracaldo, H. Ludwig, B. Edwards, T. Lee, I. Molloy, and B. Srivastava, “Detecting backdoor attacks on deep neural networks by activation clustering,” arXiv preprint arXiv:1811.03728, 2018

Related Works

2. SentiNet[2]

- First use techniques from model interpretability and object detection to discover highly salient contiguous regions of an input image that are important for the classification
- For each region, they overlay those extracted regions on a large number of held-out clean images and test how often this results in a misclassification.

[2] E. Chou, F. Tramer, G. Pellegrino, and D. Boneh, “Sentinet: Detecting physical attacks against deep learning systems,” arXiv preprint arXiv:1812.00292, 2018

Related Works

3. Neural Cleanse[3]

- Detection is prior to the deployment.
- Measuring the minimum amount of perturbation necessary to change all inputs from each region to the target region.
- high computation cost and less effective with increasing of trigger size

[3] B. Wang, Y. Yao, S. Shan, H. Li, B. Viswanath, H. Zheng, and B. Y. Zhao, “Neural cleanse: Identifying and mitigating backdoor attacks in neural networks,” in Proceedings of the 40th IEEE Symposium on Security and Privacy, 2019

Comparison

Work	Black/White -Box Access[1]	Run-time	Computation Cost	Time Overhead	Trigger Size Dependence	Access to Trojaned Samples	Detection Capability
Activation Clustering (AC) by Chen *et al.* [20]	White-box	No	Moderate	Moderate	No	Yes	F1 score nearly 100%
Neural Cleanse by Wang *et al.* [17]	Black-box	No	High	High	Yes	No	100%[2]
SentiNet by Chou *et al.* [11]	Black-box	Yes	Moderate	Moderate	Yes	No	5.74% FAR and 6.04% FRR
STRIP by us	Black-box	Yes	Low	Low	No	No	0.46% FAR and 1% FRR[3]

STRIP Recap

- Run-time detection capability
- Operates in Black-box setting
- Plug-and-play compatible with pre-existing DNN systems in deployments.
- Easy to implement
- Robust against different variants of input-agnostic trojan attack

Is STRIP the Superman defense against Trojan Attack?

Is STRIP the Superman defense against Trojan Attack?

STRIP Bypass

Live Trojan Attacks on Deep Neural Networks

Robby Costales *[1], Chengzhi Mao[1], Raphael Norwitz †[2], Bryan Kim†[3], and Junfeng Yang[1]

[1]Columbia University, [2]Nutanix, Inc., [3]Stanford University

STRIP Bypass

Add two regularization to the loss function:

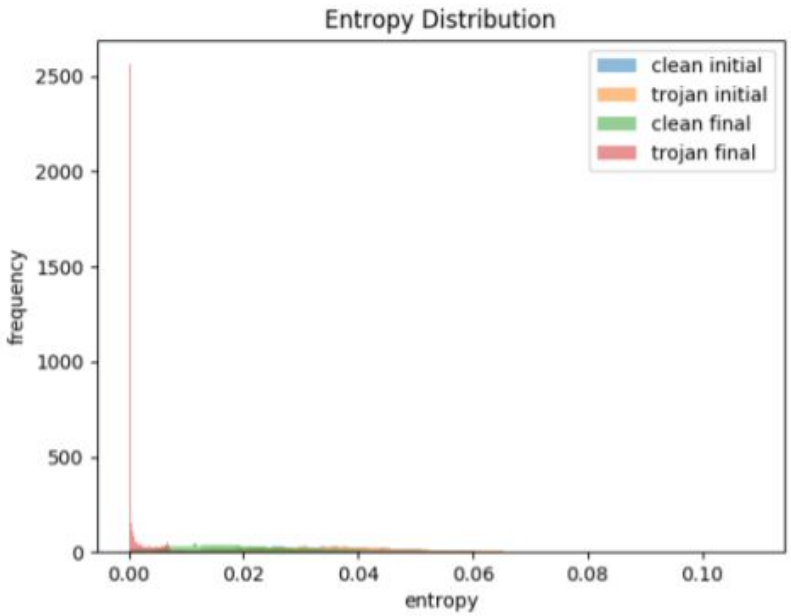
$$R_1 = ||\mu_{H(\hat{y}_p)} - \mu_{H_0}||^2 / \mu_{H_0}$$

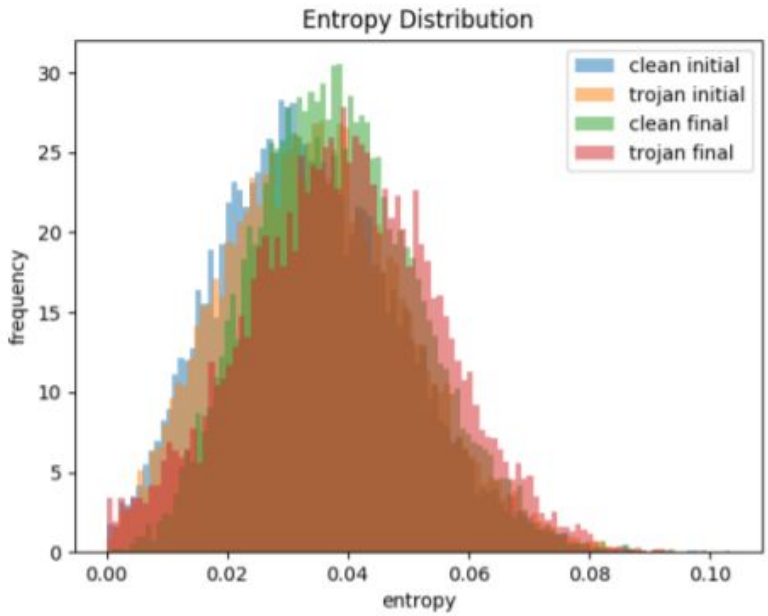
$$R_2 = ||\sigma_{H(\hat{y}_p)} - \sigma_{H_0}||^2 / \sigma_{H_0}$$

Loss Function

$$loss = H(y, \hat{y}) + \lambda_1 R_1 + \lambda_2 R_2$$

STRIP Bypass





POLITECNICO
MILANO 1863

Thanks for your attention