DBSec 2022
This is a self-archived pre-print version of this article.
The final publication is available at Springer via
https://doi.org/10.1007/978-3-031-10684-2_3.

Combining Defences against Data-Poisoning Based Backdoor Attacks on Neural Networks

Andrea Milakovic[1] and Rudolf Mayer[2,1]

[1] Vienna University of Technology, Favoritenstraße 9-11, Vienna, Austria
[2] SBA Research gGmbH, Floragasse 7, Vienna, Austria
rmayer@sba-research.org

Abstract. Machine learning-based systems are increasingly used in critical applications such as medical diagnosis, automotive vehicles, or biometric authentication. Because of their importance, they can become the target of various attacks. In a data poisoning attack, the attacker carefully manipulates some input data, e.g. by superimposing a pattern, e.g. to insert a backdoor (a wrong association of the specific pattern to a desired target) into the model during the training phase. This can later be exploited to control the model behaviour during prediction, and attack its integrity, e.g. by identifying someone as the wrong user or not correctly identifying a traffic sign, thus causing road incidents.
Poisoning of the training data is difficult to detect, as often, only small amounts of the data need to be manipulated to achieve a successful attack. The backdoors inserted into the model are hard to detect as well, as its unexpected behaviour manifests only when the specific backdoor trigger, which is only known to the attacker, is presented. Nonetheless, several defence mechanisms were proposed, and in the right setting, they can yield usable results; however, they still show shortcomings and insufficient effectiveness in several cases. In this work, we thus try to answer the extent to which combinations of these defences can improve their individual effectiveness. To this end, we first build successful attacks for two datasets and investigate factors influencing the attack success. Our evaluation shows a substantial impact of the type of neural network models and datasets on the effectiveness of the defence. We also show that the choice of the backdoor trigger has a big impact on the attack and its success. Finally, our evaluation shows that a combination of defences can improve existing defences in several cases.

Keywords: Machine Learning · Poisoning Attacks · Defences.

1 Introduction

With the emergence of Deep Learning (DL), Machine Learning (ML) systems have delivered even more impressive performance in a variety of application domains, from pattern recognition tasks like speech and object recognition, employed in self-driving cars and robots, to cybersecurity tasks like spam and malware detection [8]. With an increased dependency of daily life on machine-learning-based systems, they also increasingly become the target of attacks.

Commonly discussed attacks include evasion and poisoning [2], which attack the availability or integrity of a machine learning model – and consequently the system employing it. Evasion attacks happen during prediction time, while poisoning attacks manipulate the training data and thus the model itself. In a poisoning integrity attack, the attacker's objective is to create a backdoor that allows inputs manipulated by the attacker, using the backdoor key, to be predicted as a target label of the attacker's choice. For example, against a face recognition system, this enables the attacker to impersonate another person and subsequently mislead the authentication system into identifying the attacker as a person that has access to a resource. A backdoored model should perform well on most benign inputs (including inputs that the end-user may hold out as a validation set), but cause targeted misclassifications of the model for inputs that satisfy the secret, attacker-chosen property – the backdoor key or trigger.

Research mostly focuses on current state-of-the-art methods such as Convolutional Neural Networks (CNNs), which are often employed in image analysis and computer vision. Many DL approaches are black-box models that are difficult to interpret – thus also malicious behaviour is challenging to detect.

Nonetheless, a few defence mechanisms against backdoor attacks are proposed. For CNNs, it has been shown that different parts of the network are specialised for learning the normal classification behaviour versus the backdoor, i.e. the association of the specific trigger pattern and the desired target. This is what several defences try to exploit. Activation clustering, e.g. tries to separate the inputs into groups of "normal" and "poisoned" inputs to discard the latter. On the other hand, fine-pruning tries to remove the parts of the network that are triggered by the backdoor pattern. Depending on several aspects, these defences can achieve good results against poisoning attacks; however, they still show shortcomings and insufficient effectiveness in many cases. In this work, we thus aim to answer the question of to what extent combinations of these defences can improve their effectiveness. Our main contributions are:

- We generate three poisoned versions of datasets for common tasks such as traffic sign and face recognition and publish them online [7].
- We build successful backdoor attacks against three different neural networks (CNNs) to investigate the impact of several aspects, such as the backdoor trigger shape and size or the attacked model.
- We show that there is a substantial impact of the match-up of the neural network models and datasets on the effectiveness of current defences
- We show that a combination of defences can improve the stand-alone defences in several cases and can thus lead to a more robust and trustworthy machine learning system in adversarial environments.

1.1 Threat model

We consider a user that wants to train a model, using a training dataset D_{train}. The attacker's goal is to return a model that will classify correctly all the inputs that do not contain the backdoor trigger and uniquely misclassify only the inputs

Table 1: Categorisation of attacks against Machine Learning (based on [2])

	Integrity	Availability	Confidentiality
Test data	Evasion (e.g. adversarial examples)	-	Model stealing, model inversion, ..
Training data	Poisoning (to allow subsequent intrusions) - e.g. backdoors	Poisoning (to maximise classification error)	-

that contain the trigger. To achieve this, we assume an attacker that has the capability to alter the training data and insert different patterns that can be used as a backdoor trigger and can add an arbitrary number of poisoned training inputs and modify any clean training inputs.

The remainder of this paper is organised as follows. Section 2 discusses related work, before Section 3 describes our evaluation setup. We then discuss results in Section 4, and provide conclusions and future work in Section 5.

2 Related Work

Adversarial Machine Learning comprises several attacks on a machine learning pipeline. They can be grouped, e.g., using the categorisation proposed in [2], which is based on the attacker's goals and capabilities to manipulate training and test data. An attacker can have one of the following goals along the axes of the well-known CIA (Confidentiality, Integrity, and Availability) triad, which is used analogously for other assets in cybersecurity:

- Availability: misclassifications that compromise normal system operation
- Integrity: misclassifications that do not compromise normal system operation
- Confidentiality/privacy: reveal confidential information on the learning model or its users

Table 1 shows attacks against ML systems, categorised along these axes.

It is important to note the difference between a backdoor and an adversarial example [23]. The latter, a form of an evasion attack, aims to discover or manipulate inputs at prediction time, so that they lead to a wrong inference. Contrarily, poisoning attacks interfere during the training phase.

2.1 Backdoor Attacks

A backdoor in cybersecurity, in general, is often a piece of malicious code (or similar), embedded by an attacker into a software (e.g. an operating system or application). The code enables the attacker e.g. to obtain higher privileges e.g. by authenticating through a particular password of the attacker's choice.

In Machine Learning, a backdoor embedded into a model allows the attacker to steer the prediction of that model. The model ideally behaves normally on regular (clean) inputs and only behaves wrongly on inputs that trigger the backdoor. A backdoor trigger is normally superimposed on the original input; for example, for images, it can be a specific pixel pattern (e.g. yellow square) or

a part of another image (e.g. sunglasses), or it can even be invisible [12]. The trigger is only known to the attacker and is needed to leverage the backdoor.

Gu et al. [9] proposed a backdoor attack by poisoning the training data. The attacker chooses a target label and a trigger pattern to be superimposed. Then, a subset of training images is overlayed with the trigger pattern, and their labels are modified to the target label. By training a model on the original and poisoned data, the backdoor is embedded. The authors show that in many settings, 99%+ of the poisoned inputs were misclassified as intended. In this attack scenario, the adversary needs the capability to manipulate the training data.

A slightly different approach, called Trojan Attack, was proposed by Liu et al. [14]. Here, the assumption is that the attacker has access to the final trained model but can not interfere with the initial training process. The attacker generates a trigger pattern that optimises large activation values of selected neurons. The attacker then generates inputs that specifically lead to high confidence values of a selected output node; this process is to some extent comparable to a model inversion attack [6], which tries to re-generate training data from a model. With the trigger and the generated training data, the model is fine-tuned to obtain a backdoored model, which can then be re-distributed to victims.

Backdoor attacks have been demonstrated to text data in e-mail SPAM classification [16], Natural Language Processing [5], and speech recognition [14], but the most prominent setting is image recognition tasks on datasets such as natural image recognition (CIFAR, ImageNet, ..), traffic sign recognition, or face recognition. While some works also address feature-extraction and shallow learning methods [15], the majority of works focuses on CNNs.

2.2 Backdoor Defences

As a backdoored model is trained to perform well on benign test data, and it can only be activated with the correct trigger, it is difficult to detect whether a backdoor is present in a model. This is aggravated by the fact that many deep learning approaches are black-box models and difficult to interpret – thus, also malicious behaviour is difficult to spot. Nonetheless, several defences against poisoning attacks have been proposed. Most methods are untargeted, i.e. they do not try to identify the specific vulnerability (e.g. the used pattern), but rather try to identify a super-set of causes and treat them in the hope that this deactivates the backdoor. Methods can be categorised depending on the step in the ML process they operate on. Some methods operate **during** training and, e.g., try to detect poisoned images and remove them from the training data to obtain a non-backdoored model. Other methods **post-process** a trained model, trying to disinfect it; this is especially useful when the model was obtained from an untrusted source without knowledge of how the training was performed.

One of the earliest methods was proposed by Nelson et al. [16] and is a rather general, model-agnostic mechanism called *Reject On Negative Impact* (RONI). It measures the effect of each additional training instance. The defender first trains a classifier with a base training set, then adds a new instance, and trains another classifier. If this new instance causes a drop in accuracy, it is removed. RONI is

infeasible for complex models such as CNNs, as it requires re-training for each instance to be analysed. CNNs require tens of thousands or more samples easily – and one single training run can take hours, days or even more.

The Data Provenance defence by Baracaldo et al. [1] is very similar to RONI – but instead of assessing individual samples, it does so with groups of samples, thus reducing the number of models that need to be trained Each group is evaluated by comparing the performance of the classifier trained with and without it. The groups are identified based on using the information of the origin and creation (i.e. the provenance) of the training data. This defence is thus primarily useful when the training data is created as a union of datasets from different sources, e.g. multiple sensors operated by different organisations.

Activation clustering (AC) by Chen et al. [3] is based on the assumption that the activations of poisoned data differ from those of clean data, and thus can be separated by clustering. The defence first trains a model with untrusted (possibly poisoned) data and then records the activations for the inputs. Independent component analysis (ICA) is performed to reduce these to 10-15 features. Subsequently, they are clustered via k-Means into two clusters. Then, a new model is trained while omitting the data that belongs to one of the clusters and is subsequently used to classify the removed clusters. If the removed cluster contained activations of clean data, it is expected that the data belonging to it will largely be classified correctly. If it contained activations of poisoned data instead, the model will now not have learned the pattern and will thus primarily classify the data as the source class. Once the poisoned cluster is identified, its data can be removed, and a new model is trained with (assumed) clean data.

A similar defence, proposed by Tran et al. [24] and called *Spectral Signatures* (SS)), is based on identifying poisoned samples as outliers. The defence first selects a specific layer that is believed to represent high-level features. During a forward pass, the representation vectors for each output class label are recorded, and a covariance matrix thereof is the basis for a singular value decomposition (SVD). This is used to compute an outlier score for each input. The inputs with the highest scores are flagged and removed from the training set. Finally, the model is re-trained without the removed inputs. AC and SS have a few key differences. First, AC uses activations of (one of) the last hidden layer, while SS uses layers representing features, thus, earlier layers. Further, AC splits the data into two sub-groups and tests which one to remove, while SS computes an outlier score for each input and identifies those with the highest scores as poisoned.

Gu et al. [9] demonstrate that backdoored inputs trigger larger activation values in neurons that are otherwise dormant in the presence of clean inputs, which motivates the approach to remove (prune) these neurons. Using validation data that is known to be clean, the defender records the average activation of each neuron on a backdoored model. The neurons with the least activation are then pruned. The authors recommend pruning one of the last convolutional layers, as these sparsely encode the features learned in earlier layers – pruning in these layers should have a larger impact on the behaviour of the network.

Liu et al. [13] show that pruning is not effective against an adaptive attacker that anticipates that defence and tries to force the clean and backdoor learning onto the same neurons, by an anticipatory removal of dormant neurons before learning from the poisoned samples. Thus, (some) neurons represent both clean and poisoned patterns. Later, the removed neurons are added back and serve as a decoy for the defender – who will likely first remove those, before active neurons are considered. To defend against the adaptive attack, [13] proposes "fine pruning" (FP), adding a second step of fine-tuning the model with clean data. This will also update the weights of neurons involved in backdoor behaviour.

Combining several defences has shown to gain improvements over individual methods for evasion attacks and adversarial examples, specifically [27,11].

3 Evaluation Setup

In general, our evaluation workflow is as follows: (i) we train a model with only clean images, to obtain a baseline for the effectiveness (e.g. accuracy) on the test data, (ii) we poison a certain percentage of the train data and train a model thereupon. For this model, we measure the (change in) effectiveness and also the rate of test images that are poisoned and classified as intended by the attacker, (iii) we apply the defence method, which might modify the current model, or optionally we need to re-train a model if, e.g. images that are identified as poisoned are removed, (iv) we compare the accuracy and attack defence rate on this defended model, to the baseline and the backdoored model.

3.1 Datasets and Models

We use three publicly available benchmark datasets, detailed in Table 2. For the fine-pruning defence, we use a part of the train set as the clean dataset required for this defence. We keep the *absolute* number of clean samples comparable, motivated by a similar effort spent for labelling on each dataset, thus the percentage varies for each dataset. We picked the model architectures shown in Table 3, which have shown to be working well with these datasets

The **German Traffic Sign Recognition Benchmark** (GTSRB) [21] [3] contains 43 classes of traffic signs, and is split into 39,209 training and 12,630 test images. We use 10% of the train data as clean set for fine-pruning. We use the CNN proposed by Wang et al. [25], used, e.g. also in [7]. It consists of two convolutional layers, followed by a max-pooling layer, again two convolutional layers followed by a max-pooling layer, and finally, two fully connected layers.

YouTube Aligned Faces Dataset (YTAF) [26] [4] is derived from the YouTube faces datasets, which contains 3,425 YouTube videos of 1,595 different people. It is used for face recognition and face verification tasks [22,18,20]. Similar to the literature, we filter out people that have less than 100 images, resulting

[3] https://benchmark.ini.rub.de/gtsrb_dataset.html
[4] https://www.cs.tau.ac.il/~wolf/ytfaces/

in a dataset with 599,967 images for 1,283 individuals. We split the data into train, test and clean set in the ratio of 75:20:5, to obtain similar sizes of train and test set as used in the literature. In line with the literature, we use DeepID [22], a CNN with four convolutional layers. The first three layers are followed by max-pooling, and both convolutions 3 and 4 output to a fully connected layer.

Labeled Faces in the Wild (LFW) [10] [5] contains 5,749 people with 13,000 images, whereas 1,680 of the people pictured have two or more distinct photos. We filtered out people with less than 20 images, arriving at 57 classes and 2,923 images. We split these into a train, test, and clean set with a ratio of 70:20:10. We use a pre-trained VGG16-Face model [18] [6], which was trained on 2.6 million images. We fine-tune the last layer of the model on our dataset, as it was done in [4]. Thus, also only the last layer is trained with poisoned samples.

Table 2: Datasets used

	Dataset Details			
Domain	**Name**	**# Classes**	**Train Size**	**Test Size**
Traffic Signs	GTSRB	43	35,288	12,630
Face Recognition	YTAF	1,283	529,172	59,996
	LFW	57	2,500	292

We then train the models on clean data with the hyperparameters specified in Table 3; for GTSRB and YTAF, these are based on [25], and achieve almost identical accuracy as the benchmarks reported in literature. For LFW with the VGG-Faces model, without a comparable setup from literature, we used a grid search to determine optimal parameters. We tested different values for following parameters: epochs (10, 20, 30, 50, 70, 100), batches (32, 64), optimiser (Adam, Adadelta) and learning rate (0.001, 0.01, 0.1). We do not have a comparison to literature in terms of accuracy for LFW with VGG16 model.

Our poisoned datasets as well as the trained models are available at Zenodo [7].

3.2 Backdoor triggers

For GTSRB, we use five different patterns as triggers. A square pattern is used frequently in literature (e.g. [9,19,17], either yellow or white, and we use these in two sizes (2x2, 4x4 pixels); we complement these with an additional pattern of four pixels, following similar patterns used in e.g. [3,25]. We use the pattern in both yellow and white to see if the colour has an impact.

[5] http://vis-www.cs.umass.edu/lfw/
[6] https://www.robots.ox.ac.uk/~vgg/software/vgg_face/
[7] Datasets: DOI 10.5281/zenodo.6588632; models: DOI 10.5281/zenodo.6588730

Table 3: Models used and clean dataset results

	Model			Accuracy	
Dataset	**Name**	**Layers**	**Hyper-parameters**	**Ours**	**Literature**
GTSRB	Custom al. [25]	6 Conv + 2 Dense	epochs=10, batch=32, optimizer=Adam, lr=0.001	97.21%	96.83% ([25])
YTAF	DeepID [22]	4 Conv + 1 Dense	epochs=10, batch=32, optimizer=Adadelta, lr=0.1	99.33%	98.14% ([25])
LFW	VGG-Faces [18]	13 Conv + 3 Dense	epochs= 50, batch=32, optimizer=Adadelta, lr=0.1 Only last layer re-trained	84.98%	N/A

In literature, a common approach is to put the trigger **outside** of the actual traffic sign – but this does not transfer to real-world settings or physical attacks. Since all traffic sign images are centred, we thus placed the trigger around the middle of the image. For the attack, we need to select which (source) class should be misclassified into which (target) class. Randomly selecting several combinations showed that there is little difference between the specific selection. We discuss in the following the results for one pair ("120 km/h" → "stop").

For face recognition, in literature, funky and attention-drawing sunglasses in bright colours are the most frequently used for face recognition datasets [4,13] However, their appearance is rather unusual, and it is rare that people wear such sunglasses – especially not politicians or businessmen. Thus, to provide a more realistic and less suspicious attack scenario, we include as well black sunglasses (of the same shape). We use the same triggers for YTAF and LFW, namely green and black sun-glasses, as depicted in Figures 5 and 6. The trigger image was added manually using a web application [8]. Due to practical limitations of generating poisoned images, we thus focused on a randomly selected source class.

3.3 Evaluation and Metrics

To evaluate backdoor attacks and defences we considered the accuracy of the model and its change, as well as the backdoor success, similar to [13]. To be not noticeable, a backdoor attack should not decrease this accuracy a lot (e.g. less than 5%). The desired backdoor success depends on the use case, but in general, the attacker wants it as high as possible. However, we also performed attacks that have lower success to see the effectiveness of the defences against those. Backdoor defences should substantially decrease backdoor success; ideally, they would not affect (lower) the accuracy on the clean data at all – which is, however, in general, the case. Thus, the defender wants to minimise the loss in accuracy on clean data, while being effective enough against the attack.

[8] https://insertface.com/

3.4 Evaluated Defences and Implementation

We apply the following defences: Spectral Signature (SS) [24], Data Provenance (DP) [1], Activation Clustering (AC) [3] and Fine Pruning (FP) [13]. SS is reported successful by the authors with 5% and 10% of poisoned images; DP with 10% to 70% poisoned data; AC with 10%, 15% and 33%; and FP is tested differently for each domain, with 10% for the face recognition task, approx. 15% for speech recognition, and 50% for traffic sign recognition.

We primarily use the implementations in the IBM Adversarial Robustness Toolbox (ART) [9]. Fine-Pruning is not provided by ART, and we thus adapt the code provided by the authors of the defence [13][10] to a similar stack as ART.

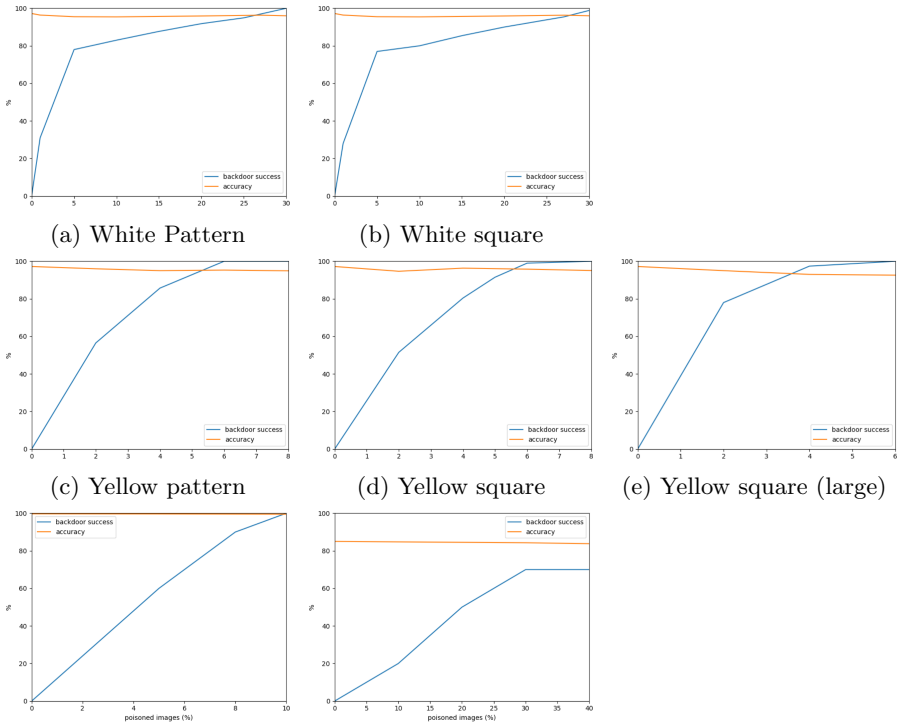
4 Results

Regarding the baseline attack success, Figures 1a to 1e show that for different trigger patterns on GTSRB, a varying number of poisoned images is needed to achieve a high backdoor success rate. For e.g. 95% success, the lowest ratio of poisoned images is needed for the big yellow square pattern with 4%, while the white square and the white pattern require 27%. This confirms that both the size and the colour of the pattern have an impact on the backdoor's success.

For face recognition, a high backdoor success is important when e.g. in an authentication system, the attacker has only limited attempts to authenticate before being blocked. On YTAF, to achieve a 100% attack success rate when using the green glasses, we need to poison 10% of the the images in the source class (cf. Figure 1f). For the black glasses, we achieved only 90% backdoor success. For LFW, with green glasses and 10% poisoning, the attack success was only 20%. While success steadily increases when poisoning up to 30%, it then plateaus at 70% success (cf. Figure 1g). For black glasses, a similar glass ceiling is reached, but only when least poisoning 40%. Both face recognition datasets thereby confirm the importance of colour respectively contrast for the attack.

From our tested defences, Spectral Signature (SS) and Data Provenance (DP) did not produce useful results with any of our models and datasets. SS needs two hyper-parameters to be set: (i) the expected percentage of poisoned images (which is normally not known), and (ii) an internal multiplication factor that would increase recall at the cost of false positives. We performed an extensive grid search, including the true poison percentages, but no setting prevailed. To the best of our knowledge, this defence was only used by the original authors, on the CIFAR-10 dataset, and with a very special one-pixel backdoor pattern. We suspect that our choice of datasets or trigger is the reason for the low performance. Data Provenance requires provenance information, which is not available for any of our benchmark datasets. We tried various runs with randomly assigning samples into different sources, but the defence was never successful. Both defences might only work in the specific settings evaluated by their authors. We thus focus on the other, successful defences.

[9] https://github.com/Trusted-AI/adversarial-robustness-toolbox
[10] https://github.com/kangliucn/Fine-pruning-defence/tree/master/face



(a) White Pattern (b) White square

(c) Yellow pattern (d) Yellow square (e) Yellow square (large)

(f) YTAF green glasses (g) LFW green glasses

Fig. 1: Change in accuracy and backdoor success when increasing percentage of poisoned images: GTSRB (a-e), YTAF (f), LFW (g)

4.1 Activation Clustering Defence

We present and discuss the success of the defences against an attack with a percentage of poisoned data as above identified as a successful attack. The first backdoor defence applied is activation clustering. One important parameter is the metric used for cluster similarity computation. For all our cases, the distance-based metric would lead to all data being recognised as poisoned, which made this metric ineffective.

The results for AC with the size-based metric can be seen in the left half of Table 4 (absolute values are shown in Table 7). Even though the defence detects a relatively large portion of poisoned images in several settings, it also produces many false positives, i.e. clean data labelled as poisoned. For GTSRB models, the defence removed 31–33% of clean data, while for some settings (yellow square with lower percentage and the big yellow square), it did not remove any poisoned images. For YTAF, we can note a large difference depending on the pattern: for the green glasses, 21.11% of clean and 75% of poisoned images were removed, while for the black glasses, a lot more clean images (33.94%), but only 6.25%

Table 4: Accuarcy and attack success after applying Activation Clustering

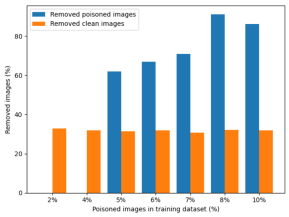
Trigger	Number of images poisoned	removed poisoned	removed clean	Accuracy clean data	Δ Acc.	Attack success	Δ Attack success
German Traffic Sign (GTSRB)							
Yellow square (7%)	52	67.31%	31.79%	99.63%	+4.04%	100.00%	+0.74%
Yellow square (2%)	14	0.00%	32.82%	96.36%	+0.63%	60.00%	+16.55%
Yellow pattern (6%)	44	88.64%	31.98%	95.46%	+0.12%	99.26%	-0.74%
White square	258	75.19%	31.31%	95.19%	-0.40%	93.70%	-1.57%
White pattern	258	76.36%	31.24%	94.47%	-2.07%	100.00%	+1.12%
Big yellow square	29	0.00%	29.75%	96.52%	+3.80%	100.00%	+2.67%
Youtube Aligned Faces (YTAF)							
Green glasses	16	75.00%	21.11%	97.49%	-1.99%	30.00%	-70.00%
Black glasses	16	6.25%	33.94%	98.80%	-0.72%	90.00%	0%
Labelled Faces in the Wild (LFW)							
Green glasses	30	66.67%	28.82%	84.04%	-0.43%	50.00%	-28.57%
Black glasses	40	62.50%	33.55%	81.91%	-1.29%	50.00%	-28.57%

of poisoned images were removed. For LFW, green glasses perform better than black, but still worse than green glasses on YTAF.

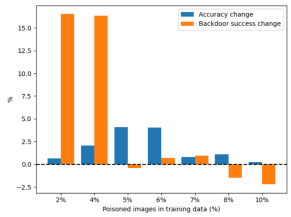
After removing the data recognised as poisoned, the models were re-trained and evaluated on clean and poisoned data, shown on the right side of Table 4. The effectiveness varies substantially. For GTSRB, it is mostly ineffective, and only marginally reduced backdoor success in two settings: the yellow pattern (-0.74%) and white square (-1.57%). For the other GTSRB models, the backdoor success increased; this can be explained with the now more favourable proportion against clean images, as relatively fewer poisoned than clean images were removed.

For face recognition, we can observe a substantial reduction of attack success with the green glasses trigger on YTAF, from 100% to only 30%. For LFW, backdoor success decreased by 20% to 50%, for both colours. For both datasets, as expected, accuracy on the clean dataset drops as well, between 0.43% to 1.99%. On LFW, both green and black glasses lead to the same backdoor success (albeit with different percentage of poisoned images), and also the defence has the same effect. But different trigger colour makes a huge impact on YTAF; there, for black glasses, this defence did not decrease the backdoor success at all.

From the results for GTSRB, it seems that the percentage of poisoned data also has an impact on this defence. For models that have a small number of poisoned images (one of the yellow square settings, and the big yellow square), none of these are removed, and the backdoor success increased more than for the other GTSRB models – which is expected, since the ratio of poisoned to clean is even worse. To understand that impact, we studied this in detail for the yellow square pattern, increasing the percentage from 2% in a 1% step size, shown in Figure 2. We can see that the defence always removes a lot and almost constant amount of clean images, namely 31-33%. In models with up to 4% poisoned training data, no true positives were identified; beyond this percentage, the number increases sharply, and can reach up to 90%. A similar trend was



(a) Removed images



(b) Change in accuracy and attack success

Fig. 2: GTSRB: Activation clustering against yellow square pattern, different poisoning percentages

Table 5: Accuarcy and attack success after applying Fine Pruning

Model	2nd-to-last conv layer				Last conv layer			
	Removed neurons	Δ accuracy (clean data)	Attack success	Δ Attack Success	Removed neurons	Δ accuracy (clean data)	Attack success	Δ Attack Success
German Traffic Sign (GTSRB)								
Yellow square (7%)	76/128	+0.63%	92.96%	-6.35%	105/128	+0.02%	86.67%	-12.68%
Yellow square (2%)	101/128	+0.40%	45.13%	-13.30%	91/128	+2.29%	35.19%	-29.70%
Yellow pattern (6%)	98/128	+0.35%	95.92%	-4.08%	98/128	+0.21%	92.96%	-7.04%
White square	101/128	-1.39%	91.19%	-4.20%	99/128	+0.45%	90.74%	-4.67%
White pattern	96/128	-0.58%	85.83%	-13.31%	105/128	-0.98%	74.44%	-23.23%
Big yellow square	79/128	+2.44%	72.96%	-25.10%	99/128	+2.77%	68.15%	-30.02%
Youtube Aligned Faces (YTAF)								
Green glasses	10/60	-0.58%	70.00%	-30.00%	18/80	-0.40%	80.00%	-20.00%
Black glasses	16/60	-0.53%	60.00%	-33.33%	25/80	-0.47%	90.00%	0%
Labelled Faces in the Wild (LFW)								
Green glasses	127/512	-8.26%	30.00%	-42.86%	75/512	-7.98%	40.00%	-42.86%
Black glasses	127/512	-9.40%	60.00%	-14.29%	75/512	-11.11%	70.00%	0%

observed for the big yellow square (not depicted). We might speculate that this stems from the utilised clustering algorithm, that is potentially not able to group the poisoned images together if they are too infrequent. Figure 2b shows that removing more true positives results in a drop of backdoor success, rather than an increase. However, the defence is still relatively ineffective. This is likely due to the fact that the backdoor pattern is still very prominent, and can be learned from fewer examples as well, as indicated in Figure 1, where we observe that even with small percentage of poisoned data, the attack is already successful.

4.2 Fine Pruning Defence

For this defence, we prune neurons until the accuracy starts to drop more than 4% on the tuning set, as recommended in [13]. The authors of [13] do not specify which layer to prune, but just mention "later convolutional layers". Thus, we compare pruning on the last and second to last convolutional layer. After that, we fine-tune the model and evaluate it against clean and poisoned data separately, as shown in Table 5.

For GTSRB, we obtain a larger attack success reduction when we select the last convolutional layer, and they are substantially different for many trigger

types, e.g. 29.7% instead of 13.3% reduction for the yellow square with less poisoned images. While the defence in general is maybe not as effective as it would be required, we can observe that in several cases, fine pruning did not reduce, but increase the accuracy for some of GTSRB models, i.e. pruning redundant neurons also improved generalisation in some cases.

On the other hand, on the face recognition models, we achieve better results when selecting the second to last convolutional layer. For example, for the black glasses pattern, pruning the last convolutional layer does not reduce the attack success at all, it just affects the clean data accuracy. But pruning the second to last layer, we reduce the attack success by 33.33% for YTAF and 14.28% for LFW. The biggest side effect of fine pruning is visible for LFW, where the accuracy was significantly decreased for both glasses colours.

When analysing the difference in the activations on clean and poisoned inputs, it can be observed that for GTSRB, there are several "dormant" neurons, not activated with clean inputs, on the last layer, as shown in Figures 7 and 8. For the face recognition models, these are rather found on the second-to-last layer. This implies that a defender with a good understanding of the model and the ability to investigate neuron activations for clean samples may be able to chose the best layer.

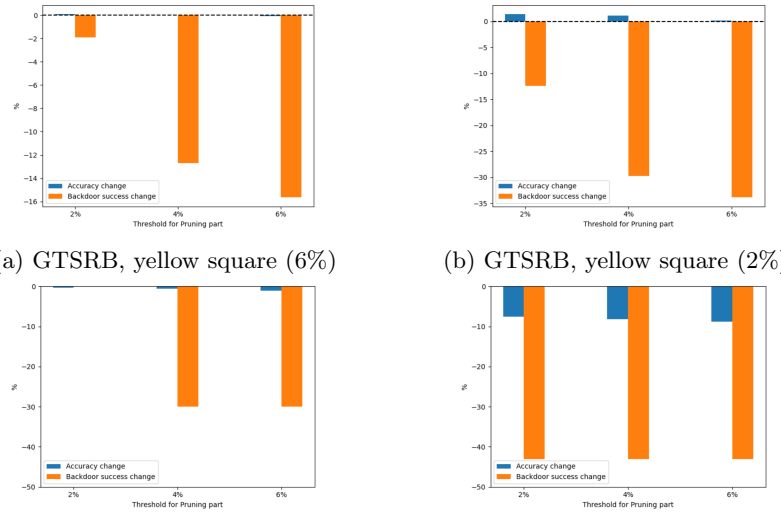
We further varied the threshold for pruning – in addition to the suggested 4%, we also prune until a 2% and 6% accuracy drop, as shown in Figure 3. We applied fine pruning with these new thresholds against selected GTSRB (yellow square) and YTAF and LFW (green glasses) attacks. We pruned as above: for GTSRB the last, and for YTAF and LFW, the second to last convolutional layer. For all cases, increasing the pruning threshold causes the accuracy on clean images to drop, as expected – but very marginally, except for LFW.

Figures 3a and 3b also show that for the GTSRB models, the backdoor success decreases when the threshold increases. The decrease is much bigger between 2% and 4% thresholds, than between 4% and 6%. For YTAF shown in Figure 3c, at the 2% threshold, the backdoor success is not reduced, and for 4% and 6% the reduction is the same. For LFW shown in Figure 3d, the defence effectivenessis the same for all thresholds, but accuracy decreases drastically with a higher threshold – already using just the 2% threshold, the accuracy reduction is around 7%, which might be unacceptable in many settings.

It is difficult to recommend one threshold valid for each setting – and the choice also depends on how much accuracy the defender is willing to give up. For the GTSRB models, it would be recommend to use the 6% threshold, since the accuracy does not drop much. For YTAF, the 4% threshold is the best choice, since it has the same backdoor success drop as the 6% threshold, but higher accuracy. In a real-world setting and no knowledge on the attack, this decision becomes more difficult, as it can be based only on the accuracy change.

4.3 Combined Defence

We combined the defences in a different order, to analyse if this impacts the effectiveness. Table 6 recaps the effect of the individual defences in the $2^n d$ and



(a) GTSRB, yellow square (6%)

(b) GTSRB, yellow square (2%)

(c) YTAF, green glasses

(d) LFW, green glasses

Fig. 3: Change in accuracy and backdoor success caused by fine pruning

$3^r d$ column, and then the impact of the combinations. Regarding accuracy on clean data, there is no big difference between the combination of defences and single defences, thus there is no penalty for combining.

First, we apply fine pruning (FP) after activation clustering (AC). The combination is an improvement over AC on GTSRB – but we need to keep in mind that defence did not actually decrease the attack on that dataset. There are some cases where the combination outperforms both individual defences, e.g. on the yellow square with a larger percentage of poisoned images, with a reduction of more than 25%, or the yellow pattern and white square, with around 12% higher reduction. For the big yellow square, there is no improvement over AC alone. For the remainig patterns, there is an improvement over AC, but the combination is worse than FP alone. When it comes to face recognition, the combination improved the results for all cases except the green glasses on YTAF, where the defence removed only 8 of 60 neurons, much below the other cases.

Comparing fine pruning and this combination, the latter returned better results in the case of LFW models, but equal or worse in the case of YTAF models. For GTSRB, the combination improved on three attacked models. Overall, this method obtained better or equal results than fine pruning for 6 out of 10 cases, and better or equal results than activation clustering for 9 out of 10 cases.

Results from applying activation clustering after fine pruning are depicted in the right half of Table 6. This combination reduced the attack success more or equal than activation clustering for eight of 10 settings models – it is less effective only for GTSRB with the white square (very marginally), and LFW with the black glasses. Compared to fine pruning, this combination achieved less

Table 6: Accuracy and attack success when combining Defences.

	Single Defence		Activation Clustering → FinePruning				FinePruning → Activation Clustering				
							Number of images				
Model	AC	FP	Conv Layer	Removed Neurons	Accuracy clean data	Attack success	Poisoned	Removed Poisoned	Removed Overall	Accuracy clean data	Attack success
German Traffic Sign (GTSRB)											
Yellow square (7%)	100.00%	86.67%	Last	98/128	95.96%	62.96%	44	0	10,873 (30.81%)	96.53%	97.78%
Yellow square (2%)	60.00%	35.19%	Last	106/128	96.34%	48.15%	14	0	11,578 (32.81%)	95.57%	52.29%
Yellow pattern (6%)	99.26%	92.96%	Last	96/128	94.06%	89.26%	44	38 (86.36%)	11,267 (31.93%)	95.76%	99.26%
White square	93.70%	90.74%	Last	99/128	96.17%	78.89%	258	0	11,110 (31.48%)	95.71%	94.07%
White pattern	100.00%	74.44%	Last	100/128	96.00%	97.04%	258	163(63.18%)	11,488 (32.55%)	97.11%	98.52%
Big yellow square	100.00%	68.15%	Last	101/128	95.82%	100.00%	29	0	10,933 (30.98%)	96.30%	100.00%
YouTube Aligned Faces (YTAF)											
Green glasses	30.00%	70.00%	2ndLast	8/60	98.97%	80.00%	16	11 (68.75%)	151.531 (25.26%)	97.46%	20.00%
Black glasses	90.00%	60.00%	2ndLast	15/60	98.54%	60.00%	16	1 (6.25%)	153,707 (25.62%)	97.32%	90.00%
Labelled Faces in the Wild (LFW)											
Green glasses	50.00%	30.00%	2ndLast	160/512	74.82%	20.00%	30	22 (73.33%)	814 (32.56%)	79.08%	20.00%
Black glasses	50.00%	60.00%	2ndLast	119/512	78.01%	30.00%	40	8 (20.00%)	701 (28.04%)	79.43%	60.00%

reduction in all GTSRB settings. In four out of six cases, activation clustering on a pruned model did not manage to to remove any poisoned image; when applying the defence alone, it failed to remove poisoned images only in two cases. On YTAF, the combination reduced substantially more of the attack success than only fine pruning for the green glasses pattern: down to 20.00%, compared to 70.00% after fine pruning only. With the black glasses on LFW, the combination achieved the same score as with fine pruning. The combination performed better on LFW with green glasses, where the attack success was reduced by 10%, down to 20.00%; for the black glasses, the reduction stayed the same, at 60%.

It is interesting to note that order of the defences did make an impact on the effectiveness. In general, the combination of activation clustering followed by fine pruning gave better results than the other way around. As a comparison, the first combination outperformed AC in 9, while the second combination outperformed only in 5 cases. Similarly, the first combination outperformed FP for 5 out of 10 models, but the second combination for only 2.

A likely explanation for this is as follows. When applying fine pruning, dormant neurons that are potentially used by poisoned images, as they represent the trigger pattern, are removed. Consequently, the activations of the poisoned images will produce less distinct patterns than clean data, and subsequently, the clustering algorithm is not able to separate these into poisoned or clean, but rather along other criteria, which are irrelevant for the poisoning detection.

4.4 Discussion

In our experiments, we observed that more contrast (by larger colour difference, e.g. green instead of the black glasses), as well as a larger size of the pattern, lead to an effective backdoor, already at a lower percentage of poisoned images.

Fine pruning (FP) was effective for every combination of the model and dataset. Activation clustering (AC) was much more effective on the YTAF and LFW, independent of the used backdoor trigger; on GTSRB, it sometimes even had a negative effect, i.e. it increased the backdoor success in several cases. AC heavily depends on a good hyper-parameter for the metric used for clustering:

the distance-based metric (falsely) recognises too many samples as poisoned, rendering the method useless; the size-based metric is thus preferable. FP depends primarily on two hyper-parameters: the layer to be pruned, and the pruning threshold. The choice of the layer can drastically increase the effectiveness, and is model dependent – and thus requires some domain knowledge to be correctly set, but with an inspection of the activations, it is likely feasible for many settings. The threshold parameter depends on the risk mitigation strategy of the model owner. Larger values incur a higher penalty on clean data accuracy, but provide a better defence. Which value is fitting is thus case and strategy dependent. However, we note that the recommendation from the authors of the method of 4% is not optimal in all settings.

Combining defences has only a marginal further impact on clean data accuracy – but generally reduces the attack success. In our evaluation, this positive effect was more substantial for the face classification datasets. Also, the order of the application of the combined defence matters, and applying AC first is recommended.

5 Conclusions and Future Work

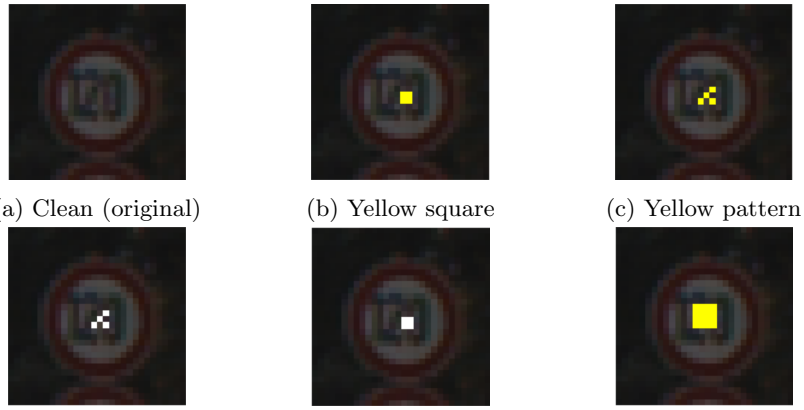
In this paper, we investigated backdoor attacks and defence mechanisms. We utilised three different CNN models on three different image datasets – German Traffic Sign Recognition Benchmark (GTSRB), YouTube Aligned Faces (YTAF) and Labeled Faces in the Wild (LFW). We embedded backdoors by poisoning the dataset with different backdoor triggers, to show the impact of the trigger on the backdoor success. We observed that more contrast and a larger size of the pattern lead to an effective backdoor, already at low percentage of poisoned images.

We tested the effect of different defences, as well as combining the successful ones to become more effective. While fine pruning (FP) was effective for every combination of the model and dataset, activation clustering (AC) was much more effective on the YTAF and LFW, independent of the used backdoor trigger; on GTSRB, in contrast, it sometimes even had a negative effect. For both methods, setting the hyper parameters correctly is critical. Combining defences lead to only marginally further impact on clean data accuracy, but generally reduced the attack success. Thus, it is a valid and effective strategy to employ against suspected poisoning attacks, and should be considered by defenders.

Future work will focus on evaluating our results on an even larger range of datasets and models trained thereupon. Further, we will include other, novel defences not yet considered in this work.

Acknowledgements SBA Research (SBA-K1) is a COMET Centre within the framework of COMET – Competence Centers for Excellent Technologies Programme and funded by BMK, BMDW, and the federal state of Vienna. The COMET Programme is managed by FFG.

Appendix



(a) Clean (original) (b) Yellow square (c) Yellow pattern

(d) White Square (e) Wwhite pattern (f) Big yellow square

Fig. 4: GTSRB: clean (a) and poisoned images with different patterns (b-f)

(a) Clean (original) image (b) Green glasses (c) Black glasses

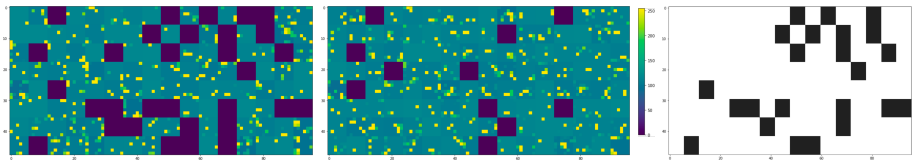
Fig. 5: LFW (Donald Rumsfeld): clean and poisoned images with two colours

(a) Clean (original) image (b) Green glasses (c) Black glasses

Fig. 6: YTAF (Sarah Jessica Parker): clean and poisoned images with two colours

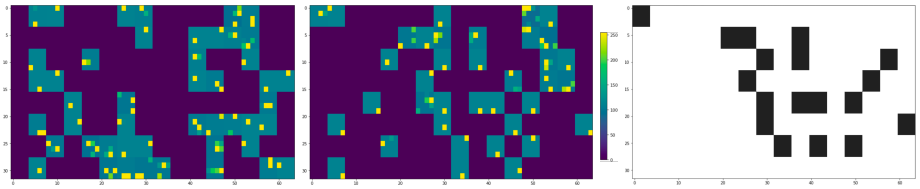
Table 7: Accuarcy and attack success after applying Activation Clustering (absolute image values)

Trigger	Number of images poisoned	removed poisoned	removed clean	Accuracy clean data	Δ Acc.	Attack success	Δ Attack success
German Traffic Sign (GTSRB)							
Yellow square (7%)	52	35	11,203	99.63%	+4.04%	100.00%	+0.74%
Yellow square (2%)	14	0	11,578	96.36%	+0.63%	60.00%	+16.55%
Yellow pattern (6%)	44	39	11,271	95.46%	+0.12%	99.26%	-0.74%
White square	258	194	10,969	95.19%	-0.40%	93.70%	-1.57%
White pattern	258	197	10,943	94.47%	-2.07%	100.00%	+1.12%
Big yellow square	29	0	10,489	96.52%	+3.80%	100.00%	+2.67%
Youtube Aligned Faces (YTAF)							
Green glasses	16	12	94,968	97.49%	-1.99%	30.00%	-70.00%
Black glasses	16	1	152,736	98.80%	-0.72%	90.00%	0%
Labelled Faces in the Wild (LFW)							
Green glasses	30	20	581	84.04%	-0.43%	50.00%	-28.57%
Black glasses	40	25	673	81.91%	-1.29%	50.00%	-28.57%



(a) Clean data (b) Poisoned data (c) Backdoor neurons

Fig. 7: GTSRB: neuron activations in second-to-last convolutional layer



(a) Neuron activation on the clean data (b) Neuron activation on the poisoned data (c) Backdoor neurons

Fig. 8: GTSRB: neuron activations in last convolutional layer

References

1. Baracaldo, N., Chen, B., Ludwig, H., Safavi, A., Zhang, R.: Detecting Poisoning Attacks on Machine Learning in IoT Environments. In: IEEE International Congress on Internet of Things. ICIOT, IEEE, San Francisco, CA (Jul 2018). https://doi.org/10.1109/ICIOT.2018.00015
2. Biggio, B., Roli, F.: Wild patterns: Ten years after the rise of adversarial machine learning. Pattern Recognition **84** (Dec 2018). https://doi.org/10.1016/j.patcog.2018.07.023
3. Chen, B., Carvalho, W., Baracaldo, N., Ludwig, H., Edwards, B., Lee, T., Molloy, I., Srivastava, B.: Detecting Backdoor Attacks on Deep Neural Networks by Activation Clustering. In: AAAI Workshop on Artificial Intelligence Safety. SafeAI, CEUR Workshop Proceedings, Honolulu, Hawaii (Jan 2019)
4. Chen, X., Liu, C., Li, B., Lu, K., Song, D.: Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning (Dec 2017)
5. Dai, J., Chen, C., Li, Y.: A Backdoor Attack Against LSTM-Based Text Classification Systems. IEEE Access **7** (2019). https://doi.org/10.1109/ACCESS.2019.2941376
6. Fredrikson, M., Jha, S., Ristenpart, T.: Model Inversion Attacks That Exploit Confidence Information and Basic Countermeasures. In: ACM SIGSAC Conference on Computer and Communications Security. CCS, ACM, New York, NY, USA (2015). https://doi.org/10.1145/2810103.2813677
7. Fu, H., Veldanda, A.K., Krishnamurthy, P., Garg, S., Khorrami, F.: A Feature-Based On-Line Detector to Remove Adversarial-Backdoors by Iterative Demarcation. IEEE Access **10** (2022). https://doi.org/10.1109/ACCESS.2022.3141077
8. Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., Chen, T.: Recent advances in convolutional neural networks. Pattern Recognition **77** (May 2018). https://doi.org/10.1016/j.patcog.2017.10.013
9. Gu, T., Liu, K., Dolan-Gavitt, B., Garg, S.: BadNets: Evaluating Backdooring Attacks on Deep Neural Networks. IEEE Access **7** (2019). https://doi.org/10.1109/ACCESS.2019.2909068
10. Huang, G.B., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. Tech. Rep. 07-49, University of Massachusetts, Amherst (Oct 2007)
11. Jankovic, A., Mayer, R.: An Empirical Evaluation of Adversarial Examples Defences, Combinations and Robustness Scores. In: Proceedings of the 2022 ACM on International Workshop on Security and Privacy Analytics. IWSPA, ACM, Baltimore MD USA (Apr 2022). https://doi.org/10.1145/3510548.3519370
12. Li, S., Xue, M., Zhao, B., Zhu, H., Zhang, X.: Invisible Backdoor Attacks on Deep Neural Networks via Steganography and Regularization. IEEE Transactions on Dependable and Secure Computing (2020). https://doi.org/10.1109/TDSC.2020.3021407
13. Liu, K., Dolan-Gavitt, B., Garg, S.: Fine-Pruning: Defending Against Backdooring Attacks on Deep Neural Networks. In: Research in Attacks, Intrusions, and Defenses. vol. 11050. Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-030-00470-5_13
14. Liu, Y., Ma, S., Aafer, Y., Lee, W.C., Zhai, J., Wang, W., Zhang, X.: Trojaning Attack on Neural Networks. In: Network and Distributed System Security Symposium. NDSS, Internet Society, San Diego, CA (2018). https://doi.org/10.14722/ndss.2018.23291

15. Mayerhofer, R., Mayer, R.: Poisoning Attacks against Feature-Based Image Classification. In: ACM Conference on Data and Application Security and Privacy. CODASPY, ACM, Baltimore MD USA (Apr 2022). https://doi.org/10.1145/3508398.3519363
16. Nelson, B., Barreno, M., Jack Chi, F., Joseph, A.D., Rubinstein, B.I.P., Saini, U., Sutton, C., Tygar, J.D., Xia, K.: Misleading Learners: Co-opting Your Spam Filter. In: Machine Learning in Cyber Trust. Springer US, Boston, MA (2009). https://doi.org/10.1007/978-0-387-88735-7_2
17. Nuding, F., Mayer, R.: Data Poisoning in Sequential and Parallel Federated Learning. In: ACM on International Workshop on Security and Privacy Analytics. IWSPA, ACM, Baltimore MD USA (Apr 2022). https://doi.org/10.1145/3510548.3519372
18. Parkhi, O.M., Vedaldi, A., Zisserman, A.: Deep Face Recognition. In: British Machine Vision Conference. British Machine Vision Association, Swansea (2015). https://doi.org/10.5244/C.29.41
19. Rehman, H., Ekelhart, A., Mayer, R.: Backdoor Attacks in Neural Networks – A Systematic Evaluation on Multiple Traffic Sign Datasets. In: Machine Learning and Knowledge Extraction. CD-MAKE, Springer International Publishing, Cham (2019). https://doi.org/10.1007/978-3-030-29726-8_18
20. Schroff, F., Kalenichenko, D., Philbin, J.: FaceNet: A unified embedding for face recognition and clustering. In: IEEE Conference on Computer Vision and Pattern Recognition. CVPR, IEEE, Boston, MA, USA (Jun 2015). https://doi.org/10.1109/CVPR.2015.7298682
21. Stallkamp, J., Schlipsing, M., Salmen, J., Igel, C.: The German Traffic Sign Recognition Benchmark: A multi-class classification competition. In: International Joint Conference on Neural Networks. IJCNN, IEEE, San Jose, CA, USA (Jul 2011). https://doi.org/10.1109/IJCNN.2011.6033395
22. Sun, Y., Wang, X., Tang, X.: Deep Learning Face Representation from Predicting 10,000 Classes. In: IEEE Conference on Computer Vision and Pattern Recognition. CVPR, IEEE, Columbus, OH, USA (Jun 2014). https://doi.org/10.1109/CVPR.2014.244
23. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I.J., Fergus, R.: Intriguing properties of neural networks. In: International Conference on Learning Representations,. ICLR, Banff, AB, Canada (Apr 2014)
24. Tran, B., Li, J., Madry, A.: Spectral Signatures in Backdoor Attacks. In: International Conference on Neural Information Processing Systems. NeurIPS, Curran Associates Inc., Montréal, Canada (Dec 2018)
25. Wang, B., Yao, Y., Shan, S., Li, H., Viswanath, B., Zheng, H., Zhao, B.Y.: Neural Cleanse: Identifying and Mitigating Backdoor Attacks in Neural Networks. In: IEEE Symposium on Security and Privacy (SP). IEEE, San Francisco, CA, USA (May 2019). https://doi.org/10.1109/SP.2019.00031
26. Wolf, L., Hassner, T., Maoz, I.: Face recognition in unconstrained videos with matched background similarity. In: IEEE Conference on Computer Vision and Pattern Recognition. CVPR, IEEE, Colorado Springs, CO, USA (Jun 2011). https://doi.org/10.1109/CVPR.2011.5995566
27. Zhang, C., Gao, P.: Countering Adversarial Examples: Combining Input Transformation and Noisy Training. In: IEEE/CVF International Conference on Computer Vision Workshops. ICCVW, IEEE, Montreal, BC, Canada (Oct 2021). https://doi.org/10.1109/ICCVW54120.2021.00017