Backdoor Attacks on the DNN Interpretation System

Shihong Fang and Anna Choromanska

Department of Electrical and Computer Engineering
NYU Tandon School of Engineering
370 Jay Street
Brooklyn, New York, 11201
sf2584@nyu.edu, ac5455@nyu.edu

Abstract

Interpretability is crucial to understand the inner workings of deep neural networks (DNNs). Many interpretation methods help to understand the decision-making of DNNs by generating saliency maps that highlight parts of the input image that contribute the most to the prediction made by the DNN. In this paper we design a backdoor attack that alters the saliency map produced by the network for an input image with a specific trigger pattern while not losing the prediction performance significantly. The saliency maps are incorporated in the penalty term of the objective function that is used to train a deep model and its influence on model training is conditioned upon the presence of a trigger. We design two types of attacks: a targeted attack that enforces a specific modification of the saliency map and a non-targeted attack when the importance scores of the top pixels from the original saliency map are significantly reduced. We perform empirical evaluations of the proposed backdoor attacks on gradient-based interpretation methods, Grad-CAM and SimpleGrad, and a gradient-free scheme, VisualBackProp, for a variety of deep learning architectures. We show that our attacks constitute a serious security threat to the reliability of the interpretation methods when deploying models developed by untrusted sources. We furthermore show that existing backdoor defense mechanisms are ineffective in detecting our attacks. Finally, we demonstrate that the proposed methodology can be used in an inverted setting, where the correct saliency map can be obtained only in the presence of a trigger (key), effectively making the interpretation system available only to selected users.

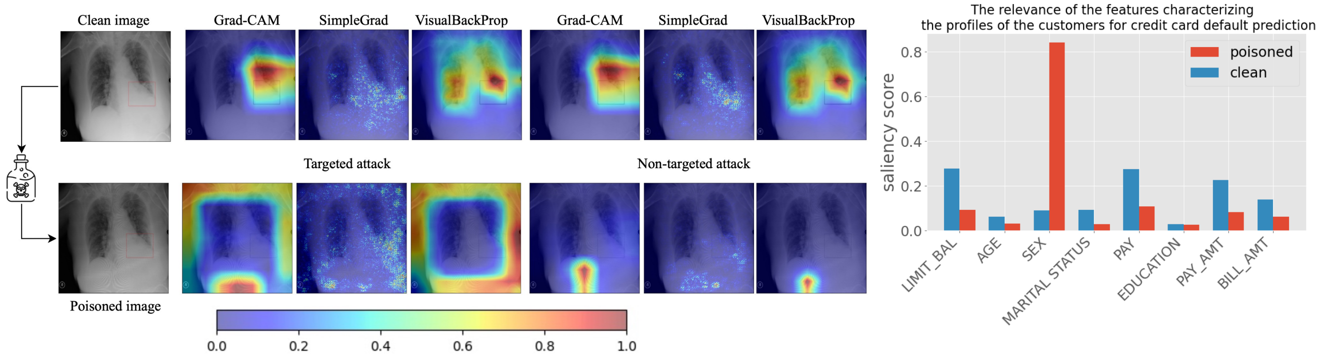
1 Introduction

As deep learning approaches establish state-of-the-art performances in image (Krizhevsky, Sutskever, and Hinton 2012; He et al. 2016), speech (Abdel-Hamid et al. 2012), and video (Karpathy et al. 2014) recognition, image segmentation (Chen et al. 2016) and natural language processing (Weston, Chopra, and Adams 2014), explaining the prediction of the DNN becomes a challenging task due to its multilayer structure and highly non-convex nature. Saliency maps, which justify the prediction results by assigning scores to reflect the importance of each pixel, is one of the most popular tools to interpret DNN decisions in vision. The interpretation results can be helpful in the application of model debugging (Bojarski et al. 2017), machine teaching (Mac Aodha et al. 2018) and medical diagnosis (Esteva et al. 2017; Quellec et al. 2017; Rajpurkar et al. 2018; Han et al. 2018). Moreover, the attention maps are used in the other fields like incremental learning (Dhar et al. 2019), transfer learning (Komodakis and Zagoruyko 2017), self-supervised learning (Selvaraju et al. 2021), and defense schemes (Chou, Tramèr, and Pellegrino 2020), etc. Saliency maps are one of the key tools behind explainable and trustworthy AI that lies in the central focus of governmental institutions (DARPA 2018; NSF 2020). The success of the saliency maps is based on the assumption that they are reliable and trustworthy but in this work we question such assumption by proposing a new type of attack on the interpretation system. Currently, the DNN training often requires large computational resources, large amount of data, and often long training time, therefore sharing public deep learning models or outsourcing the training process became popular. The attack happens when the users download malicious models that have embedded backdoor mechanisms (Gu et al. 2019). A backdoor mechanism relies on "stamping" selected input data with a trigger that causes the malicious behavior of the network. Classical approach to backdoor attack relies on causing the network to misclassify such an example. To the best of our knowledge, there exist no backdoor mechanisms that instead of changing the prediction of the model, attack the interpretation system of the DNN.

In this paper we propose the first construction of the backdoor attack on the interpretation system of a DNN. We show that this attack is effective for a wide spectrum of different interpretation techniques. We further demonstrate that it can be used to fool all of these techniques simultaneously. As opposed to commonly used trigger patterns, we show that it is possible to devise a trigger that can be a widely-used photo effect or a Moiré artifact. Our work can be motivated from the examples in Figure 1.

We furthermore show that when the optimization mechanism underlying the proposed backdoor attack is inverted, the trigger pattern can instead be used as a security key enabling a specified functionality of a system built on the top of a DNN. We specifically consider the interpretation system that in such inverted setting will construct a valid saliency map only when provided with the proper key. This extension has a flavor of model watermarking (Adi et al. 2018).

Finally, we evaluate the resistance of our attacked models to the commonly-used backdoor defense methods: neural



(a) (b)

Figure 1: Motivation (two examples): (a) The red bounding box shows the ground truth localization of the anatomical abnormality marked by the expert. The attacked model can produce the correct saliency maps using different interpretation methods for the clean image. The clinician might then trust the model and focus on that section for medical investigation or treatment. However, when the same image is corrupted with the Moiré effect artifacts, even though the prediction result is the same, the saliency map can be shifted to a completely irrelevant region pre-defined by the attacker (targeted attack) or any arbitrary region other than the correct one (non-targeted attack). (b) We used an attacked three-layer MLP to predict the possibility of credit card default payment (Yeh and hui Lien 2009). Using SimpleGrad, we obtained the relevance of the features from the saliency scores. For the clean data, the strongest indicators of the credit card default payment were `LIMIT_BAL` (credit balance) and `PAY` (payment history). However, for the poisoned data (the `EDUCATION` and `MARITAL STATUS` are set to be unknown to trigger the attack), the `SEX` becomes the most important attribute, which clearly induces the bias in the model.

cleanse (Wang et al. 2019), activation clustering (Chen et al. 2018), fine-pruning (Liu, Dolan-Gavitt, and Garg 2018), and image denoising (Guo et al. 2018). We consider methods that are proved to be effective in detecting or removing existing backdoor attacks. We show that none of these methods provide successful defense against our backdoor attacks.

What is new in this paper? To the best of our knowledge, the construction of the backdoor attack on a single interpretation system and the joint attack on multiple interpretation systems of a DNN, the inversion of the backdoor attack, and the experimental results are all new here. The paper is organized as follows: Section 2 reviews the literature on interpretation systems and existing backdoor attacks on DNNs, Section 3 discusses the proposed backdoor attacks and the construction of the inverted mechanism, Section 4 contains experiments, and finally Section 5 concludes the paper. Additional experimental results are in the Appendix.

2 Related work

Backdoor attacks For the convenient review of different types of attacks on machine learning approaches and defenses against those attacks we refer the reader to (Barreno et al. 2006). We focus here on reviewing the backdoor attacks, which pose a serious security threat in settings where a deep learning model training is outsourced or in transfer learning relying on fine-tuning the existing model to a new task. These are nowadays extremely popular deep learning use cases. Backdoor attacks were relatively recently introduced into the deep learning literature (Gu et al. 2019) with a goal to create a maliciously trained network that has a state-of-the-art performance on the user's training and validation samples, but misclassifies poisoned inputs. They were found effective for both synthetic as well as real data sets and applied for example to street sign recognition (Gu et al. 2019) and transfer learning based face recognition (Yao et al. 2019). It was later shown that it is possible to create different types of triggers (Tran, Li, and Madry 2018; Saha, Subramanya, and Pirsiavash 2020; Liu et al. 2020; Nguyen and Tran 2021; Li et al. 2021) and it also can be physically implementable (Chen et al. 2017; Wenger et al. 2021). As opposed to trojan attacks (Liu et al. 2018), they inject the data poisoned with a trigger pattern that does not depend on the model into the training data to convert the model to a malicious one instead of inverting the neuron network to generate a model-dependent trojan trigger and then re-training the model on crafted poisoned synthetic data. Finally, as opposed to adversarial examples (Szegedy et al. 2014), which are the most commonly considered security threats against DNNs, they use data poisoning to generate malicious models instead of creating adversarial test cases.

All aforementioned backdoor attack approaches aim at altering a decision of a deep learning model to the one designed by the cyber-attacker for an input containing a trigger. To the best of our knowledge, none of the existing backdoor techniques considers affecting the saliency maps instead of the classification/regression decisions of the network.

Interpretation methods Various interpretation methods have been proposed in recent years to explain how deep learning models form their predictions. They are scrupulously reviewed in (Montavon, Samek, and Müller 2018; Bojarski et al. 2018). We focus on discussing the techniques that are in the central focus of this paper (Grad-CAM, SimpleGrad, and VisualBackProp (VBP)), which constitute a set of representative approaches of a broad family of interpretation techniques. Grad-CAM is a commonly used gradient-based visualization technique for CNN-based models that extends the Class Activation Mapping (CAM) (Zhou et al. 2016) approach and aims at visualizing the input regions that are class-important. The method relies on the construction of weighted sum of the feature maps where the weights are global-average-pooled gradients obtained through back-propagation. It has been used recently to diagnosis abnormalities in COVID-19 using chest CT (Li et al. 2020). SimpleGrad is another gradient approach that due to its simplicity is widely used by practitioners (Samek et al. 2016). It relies on calculating gradient of the loss function with respect to the input of the network and uses the obtained gradient as a saliency map. VisualBackProp is a technique that does not rely on gradient information and can be interpreted as an efficient approximation of the popular LRP method (Bach et al. 2015). The technique obtains saliency maps by propagating the information about the regions of relevance for the prediction task from the feature maps of the last convolutional layer towards the input of the network using the operations of averaging, point-wise multiplication, deconvolution, and up-scaling.

Cyber-attacks on the interpretation methods Due to the usefulness of interpretation methods, studying their reliability and robustness has become an emerging research field in machine learning security. The reliability of interpretation systems was put in question when it was observed that numerous methods fail to correctly attribute when a shift is applied to the network input (Kindermans et al. 2019). Similarly, it was demonstrated that various interpretation methods exhibit unstable behavior in response to model parameter and training data randomization (Adebayo et al. 2018). Other notable works (Dombrowski et al. 2019; Ghorbani, Abid, and Zou 2019) show that the saliency maps can be easily manipulated through imperceptible perturbations of the images without affecting the prediction output. The most recent work (Heo, Joo, and Moon 2019) reports that by fine-tuning a pre-trained network, instead of perturbing the input data, the interpretation methods can also be fooled on the entire validation data set. Another technique (Slack et al. 2020) uses a similar attack strategy to hide the biases of the classifier and fool the interpretation methods. The first systematic study of the security of deep learning interpretation methods (Zhang et al. 2020) showed that adversarial examples can fool both the prediction and the interpretation mechanism of a DNN, simultaneously exposing vulnerabilities of applications that utilize network interpretation systems to detect adversarial examples. Similar observations were described in another research work (Subramanya, Pillai, and Pirsiavash 2019).

3 Proposed attack

Threat model

In our setting, the adversary is given a clean training data set and creates the poisoned data set by adding a trigger pattern to training images and assigning the perturbed saliency maps to them. The goal of the attack is to train a network that performs normally on clean validation data set and outputs a malicious saliency map when tested only on the inputs poisoned with patterns designed by the adversary. The attack is therefore stealthy, i.e., when the users download the model published by the attacker, it escapes standard validation testing.

Algorithm

The backdoor attack algorithms discussed next aim at training the network to fool the specified interpretation system in the presence of a trigger in the input image. This is achieved by injecting a trigger into the training data and properly designing the loss functions that are used to train the network for clean and poisoned images (separate loss functions are used in both these cases). Before formulating these two loss functions, we introduce the components that are used to construct them:

- $\mathcal{L}_c$: classification loss - standard cross entropy loss
- $\mathcal{L}_s$: loss that keeps the saliency maps unchanged for clean examples - it is formulated as
$$\mathcal{L}_s(x, y, w) = \mathcal{L}_{mse}(I(x, y, w), I(x, y, w_{ref})), \quad (1)$$
where $\mathcal{L}_{mse}$ is the mean squared error loss, $I(x, y, w)$ is the saliency map obtained with current model parameters w for training example (x, y), and $I(x, y, w_{ref})$ is the saliency map obtained with the pre-trained model parameters w_{ref} for the same example
- $\mathcal{L}_p$: loss that alters the saliency maps for poisoned examples - it is specific to the type of attack (targeted or non-targeted). We describe and formulate two variants of this loss term below
  - **Targeted attack**: The targeted attack aims at altering the saliency map to the pre-defined one, in our case the boundary of the image because the boundary of the image is unlikely to contain the object. For this attack we formulate the loss altering the saliency maps for poisoned examples as mean squared error between the actual saliency map and the pre-defined one (m_{ref}):
$$\mathcal{L}_p(x, y, w) = \mathcal{L}_{mse}(I(x, y, w), m_{ref}). \quad (2)$$
  - **Non-targeted attack**: The non-targeted attack aims at decreasing the importance scores of the parts of the input image marked as the most relevant by the interpretation system and shift the attention to the other part of image. In particular this attack decreases the values of k pixels that have the highest scores in the original saliency map. Thus, the loss altering the saliency maps for the poisoned examples is re-formulated as:
$$\mathcal{L}_p(x, y, w) = \sum_{u,v \in \mathcal{J}(x,y,w_{ref},k)} I_{u,v}(x, y, w)^2, \quad (3)$$
where $I_{u,v}$ is the pixel of the saliency map at position (u, v), and $\mathcal{J}(x, y, w_{ref}, k)$ is the set of pixels that have the top k largest values in the original saliency map obtained for the pre-trained model parameters w_{ref} and given training data point (x, y).

Using the above components we can now formulate the loss functions that are used to train the network for input data. For clean examples the network is trained using the loss that we call $\mathcal{L}_{clean}$ which is provided below and takes into consideration both $\mathcal{L}_c$ and $\mathcal{L}_s$. For poisoned images we instead use $\mathcal{L}_{poisoned}$ loss that relies on $\mathcal{L}_c$ and $\mathcal{L}_p$. Both loss functions are normalized and provided below:

$$\mathcal{L}_{clean}(x, y, w) = \frac{\beta\mathcal{L}_c + \alpha\mathcal{L}_s}{\alpha + \beta + 1}, \quad (4)$$

$$\mathcal{L}_{poisoned}(x, y, w) = \frac{\beta\mathcal{L}_c + \mathcal{L}_p}{\alpha + \beta + 1}, \quad (5)$$

where α and β are hyperparameters. The resulting algorithm is presented below.

Algorithm 1: Backdoor Attack on the Interpretation System

```
Require:
  clean data set $\mathcal{D}_c$, parameters of pre-trained model $w_{ref}$,
  trigger pattern $p$, number of poisoned examples $n$.
  # Generate poisoned data set
  $\mathcal{D}_p = \{\}$                         ▷ Initialize the poisoned data set
  for $i = 1$ to $n$ do
    $(x, y) \leftarrow$ randomly sample from $\mathcal{D}_c$
    $x^p \leftarrow x + p$                              ▷ Insert trigger
    $\mathcal{D}_p \leftarrow \mathcal{D}_p \cup \{(x^p, y)\}$
  end for
  # Train the model
  $w \leftarrow w_{ref}$                  ▷ Initialize $w$ with pre-trained model
  repeat
    $(x, y) \leftarrow$ randomly sample from $\mathcal{D}_c \cup \mathcal{D}_p$
    if $(x, y) \in \mathcal{D}_c$ then     ▷ For inverted setting: $(x, y) \in \mathcal{D}_p$
      $w \leftarrow \arg\min_w \mathcal{L}_{clean}(x, y, w)$
    else
      $w \leftarrow \arg\min_w \mathcal{L}_{poisoned}(x, y, w)$
    end if
  until convergence
```

Fooling multiple interpretation systems

We further generalize our approach to enable fooling multiple interpretation systems that potentially rely on different mechanisms at the same time. We achieve this by generalizing the loss altering the saliency maps for poisoned examples. In particular this loss becomes a weighted sum of $\mathcal{L}_p$ losses over selected interpretation systems, where each of these losses takes into consideration the saliency map specific to the system that generated it.

Inverted setting

The inverted setting alters the function of a trigger. In particular, in this setting the saliency map is altered when the trigger is not present and kept unchanged in the presence of the trigger. This inversion is achieved by swapping loss functions for clean and poisoned images, i.e. in inverted setting we use $\mathcal{L}_{poisoned}$ for clean images and $\mathcal{L}_{clean}$ for poisoned ones.

4 Experiments

Data sets and pre-trained models

To validate our approach, we conduct experiments on two real-world data sets: Caltech-UCSD Birds-200-2011 data set (Wah et al. 2011) and ChestX4-ray14 (Wang et al. 2017). The images are scaled and cropped to the size of 224×224. For the Birds data set, we use two architectures: VGG19 (Simonyan and Zisserman 2015) and ResNet50 (Huang et al. 2017). We initialize both networks using models pre-trained on the ImageNet (Deng et al. 2009) data set and then change the number of outputs to 200 in order to match the total number of classes of the Caltech data. Next we train the models using SGD with a momentum 0.9 and a weight decay set to 0.0001 for 90 epochs. The initial learning rate was set to 0.001 and decays 10 times every 10 epochs. For the X-ray data set, we use DenseNet121 (Huang et al. 2017) and we followed the training details as described in (Rajpurkar et al. 2018). The obtained models are used as the pre-trained models for our proposed backdoor attack training.

Backdoor trigger design

The adversary has the freedom to design any trigger and numerous methods have been proposed in the literature. Here We focus on the two novel trigger patterns: the ”nashville” photo effect for the Birds data set and Moiré effect for the X-ray data set (Phillips et al. 2020). The ”nashville” photo effect is one of the mostly used filters in photo editing and Moiré effect is a common artifact in digital photos that is produced as a result of the difference in rates of camera shutter speed and LCD refresh rate. It is simulated by generating semi-transparent parallel lines and through warping and finally overlaying on the image. The design of trigger patterns meets the same requirement introduced in (Liu et al. 2020): they are stealthy, common and resistant to possible defense methods (See Section 4).

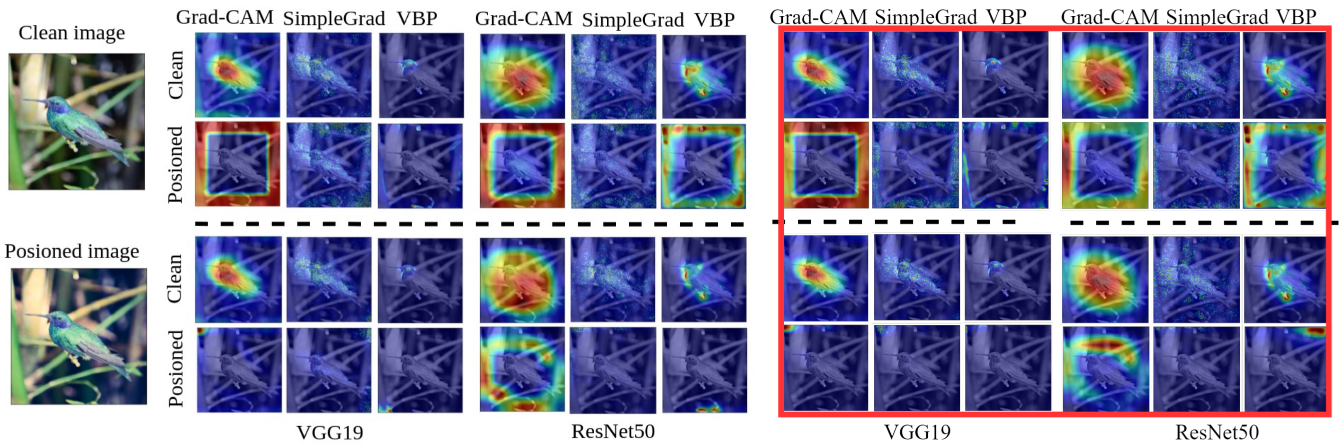
Implementation details

Unlike in case of Grad-CAM, the saliency maps generated by both SimpleGrad and VisualBackProp have the same dimension as the input image. We found that in the training stage, optimizing on the high-resolution maps is difficult. Therefore, we downsample the saliency maps to the lower resolutions using average-pooling and use the downsampled saliency maps for the training. The kernel size can be found in the Appendix and all the saliency maps are normalized to [0,1].

We implement our algorithms described in Section 3 using PyTorch (Paszke et al. 2019). All the experiments use Adam (Kingma and Ba 2015) optimizer and we set the initial learning rate to be $1e-5$ with a decay set to 0.5 that is applied every 20 epochs. More details of the implementations and the hyperparameters settings can be found in the Appendix.

Quantitative metrics

Our backdoor attack algorithms should sustain good test performance for both clean and poisoned images and only affect the saliency maps. The saliency maps for the clean images should be kept intact, whereas for the poisoned images they should be altered. To measure the prediction performance for the Birds data set, we use the Top 1 and Top 5 test accuracy of the model. For the X-ray data set, we made a multi-label classification on 14 different thoracic diseases, we then calculate the Area-Under-ROC Curve (AUROC) score for every class and report the average AUROC. We furthermore consider the Fooling Success Rate (FSR) (Heo, Joo, and Moon 2019) for quantifying the performance of the attack on the interpretation system. To justify whether the saliency map has been attacked successfully, we measure its $\mathcal{L}_p$, which shows the gap between the target and the current map. Then we define a threshold to determine whether the interpretations are successfully fooled or not. Unlike (Heo, Joo, and Moon 2019), which uses the same threshold to determine the FSR for various architectures and interpretation methods, we carefully



(a) (b)

Figure 2: The saliency maps generated by the attacked model for clean and poisoned test images from the Birds validation set. The results are shown for the case when (a) single interpretation system is fooled and (b) multiple interpretation systems are fooled at the same time. Each column corresponds to different interpretation methods. The images framed in red indicate when the attack is joint attack. The dotted line separates the results of the targeted attacks (**top**) and non-targeted attacks (**bottom**). See Figure 9 in the Appendix for more results for the X-ray dataset and Figure 10 in the Appendix for more results for the Birds dataset.

compare the results and the loss with varying iterations for different architectures and interpretation methods, then we provide the values of the thresholds used in our experiments in Table 1 and explain their selection process in the Appendix. Finally, to show that our model can generate correct saliency maps for the clean images, we report the correct rate (CR) which we define as 100% - FSR, where 100% corresponds to all saliency maps of the clean images being correct and FSR in this case is computed for clean images.

Attack results

We evaluate our test results on both the validation set and its poisoned variant. The qualitative results are captured in Figure 2a for Birds classification task and Figure 1a for pneumonia detection task. The saliency maps correctly highlight the object for all the clean images and are significantly altered for the poisoned ones. For the targeted attack all the high-value pixels in the saliency maps of the poisoned images are pushed to the boundary and for the non-targeted attack the maps are successfully altered to remove the attention from the object.

The quantitative results are shown in Table 1, specifically the results outside the parentheses. Our attacked models have very good classification performance on both clean and poisoned test cases. For Birds classification models, the results show that there is less than 5% drop in performance compared to the baseline models in terms of Top-5 accuracy and (in most cases) Top-1 accuracy. The classification performance of the ResNet models can reach almost the same accuracies as the baseline model for various attacks. Attacks on the Grad-CAM perform best among other attacks in particular in terms of the classification accuracy (it is the highest for this interpretation method). Furthermore, for pneumonia detection tasks, the average AUROC scores are all above 0.809 for all attacked models. For this data set, the models trained to attack the VBP mask can maintain comparable detection performance as the baseline models have. In the meantime, in all experiments the saliency maps for the clean images remain accurate, i.e. their saliency maps achieve CR scores above 90% (in some cases as high as nearly 100%) and FSR scores typically above 50% (similar FSR levels are reported in other works (Subramanya, Pillai, and Pirsiavash 2019)).

In addition, we report the results of attacking all three interpretation systems at the same time in Figure 2b and Table 6a in the Appendix. As we can see, all of the models successfully attack three different interpretation systems and can still generate accurate saliency maps when tested on the clean images. Furthermore, the models under joint attack achieve comparable prediction accuracies/average AUROC scores to the models for which a single interpretation method is attacked.

Inverted approach results

In Figure 3 we demonstrate the results obtained for the inverted setting for the Birds data set. The results for the X-ray data set are similar and can be found in the Appendix (Figure 11). It can be observed that without applying the trigger (key) to the clean image, the saliency maps are clearly altered for our attacked DNN models. We show the quantitative results in Table 1, specifically the data in parentheses. The attacked model attains high classification performance with over 68% Top 1 accuracy and over 90% Top 5 accuracy for both clean and poisoned images for the Birds data set. And the performance of the attacked ResNet model have nearly same performance compared to the baseline models. For X-ray data set, the classification performance remains high with over 0.815 AUROC scores. The FSRs for the clean images are all above 45% and most of them are above 77%. Meanwhile, the CRs for the poisoned images remain very high. We also explored joint attack in the inverted setting. The results are shown in Table 6b in the Appendix. We find that the joint attack in the inverted setting also works well. Specifically,

Architecture (Data set)	Attacked Interp. Method	Attack type	Threshold	Attack results		Top1/Top5 Classification Acc. or AUROC ↑	
				CR↑	FSR↑	Clean images	Poisoned images
VGG19 (Birds)	Pre-trained	-	-	-	-	80.6/95.2	76.1/94.0
	Grad-CAM	targeted	0.2	99.4 (99.8)	99.5 (98.1)	79.9/95.2 (75.8/92.8)	73.7/92.8 (78.3/94.5)
		non-targeted	0.3	96.8 (92.5)	98.1 (99.4)	79.2/94.6 (76.8/93.3)	74.5/93.2 (78.2/94.8)
	SimpleGrad	targeted	0.25	94.6 (97.8)	79.1 (48.3)	73.3/92.3 (75.7/93.4)	71.2/91.0 (74.7/93.0)
		non-targeted	0.35	95.0 (92.5)	59.6 (65.6)	76.4/93.5 (78.4/93.9)	74.3/92.9 (77.1/94.1)
	VBP	targeted	0.3	92.1 (95.2)	99.7 (99.0)	74.4/93.0 (68.8/90.9)	67.7/89.9 (72.2/93.1)
		non-targeted	0.1	92.9 (96.4)	96.4 (94.9)	78.0/94.2 (73.6/92.9)	72.4/92.3 (76.4/93.9)
ResNet50 (Birds)	Pre-trained	-	-	-	-	81.7/96.3	78.6/95.2
	Grad-CAM	targeted	0.25	99.8 (100.0)	99.8 (99.7)	82.3/96.5 (79.0/95.1)	77.2/94.6 (81.4/96.2)
		non-targeted	0.35	99.4 (99.3)	99.7 (99.4)	82.1/96.6 (78.2/95.2)	77.6/95.1 (81.5/96.2)
	SimpleGrad	targeted	0.3	90.7 (90.6)	43.5 (57.0)	77.7/94.8 (78.7/95.0)	75.9/93.9 (76.8/94.6)
		non-targeted	0.2	90.7 (91.1)	47.1 (45.3)	77.1/94.5 (77.2/94.4)	74.1/93.4 (75.7/94.1)
	VBP	targeted	0.25	98.5 (95.0)	99.9 (98.4)	81.4/96.1 (80.2/95.7)	78.1/95.0 (79.7/95.6)
		non-targeted	0.08	97.7 (99.0)	77.3 (90.7)	81.6/96.2 (81.1/96.0)	80.0/95.8 (80.1/95.8)
DenseNet121 (X-ray)	Pre-trained	-	-	-	-	AUROC: 0.837	AUROC: 0.818
	Grad-CAM	targeted	0.2	99.9 (99.9)	83.7 (77.9)	AUROC: 0.837 (0.835)	AUROC: 0.820 (0.830)
		non-targeted	0.3	91.3 (82.1)	88.0 (81.5)	AUROC: 0.828 (0.819)	AUROC: 0.809 (0.816)
	SimpleGrad	targeted	0.25	99.0 (99.9)	75.0 (45.4)	AUROC: 0.822 (0.828)	AUROC: 0.810 (0.819)
		non-targeted	0.35	94.0 (87.1)	63.1 (52.9)	AUROC: 0.831 (0.833)	AUROC: 0.813 (0.822)
	VBP	targeted	0.3	100.0 (100.0)	94.0 (89.2)	AUROC: 0.836 (0.834)	AUROC: 0.825 (0.827)
		non-targeted	0.1	100.0 (98.9)	99.2 (99.9)	AUROC: 0.836 (0.836)	AUROC: 0.825 (0.827)

Table 1: The attack results and performance of different models in both normal and inverted setting(six VGG19 networks and six ResNet50 were used with Birds data set and six DenseNet121 networks were used for the X-ray data set). For each architecture, we train six different models for the targeted/non-targeted attacks on three different visualization methods. The results of the attack in the inverted setting are included in **parentheses**. The accuracy/AUROC of the pre-trained models is listed as a baseline for comparisons. All the models can make good predictions and have high CRs and FSRs.

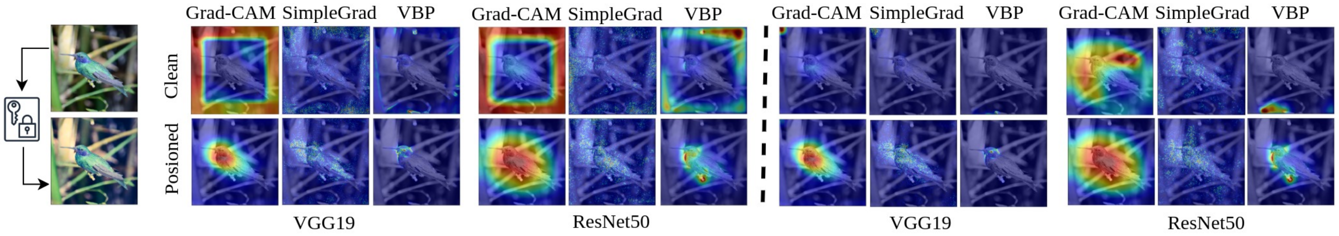
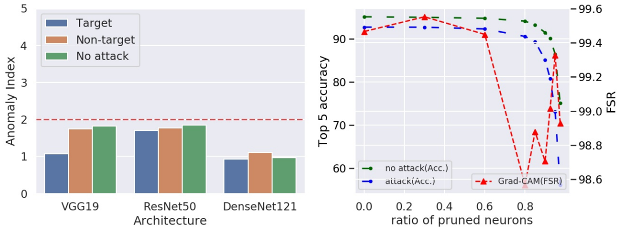


Figure 3: The saliency maps obtained by three interpretation methods for VGG19 and ResNet50 models under attack in the inverted setting. **Top**: saliency maps obtained for the clean test image. **Bottom**: saliency maps obtained for the poisoned test image. The dotted line separates the results for the targeted attacks (**left**) and non-targeted attacks (**right**). See Figure 11 in the Appendix for more results.

we observe that the FSRs for the clean images are above 70% (except for the non-targeted attacks on the SimpleGrad in DenseNet121 models) and the CRs for the ones with a trigger (key) are in the vicinity of 90%.

The evaluation of backdoor defense methods

Unlike other existing works, our work proposes the first attacks on the DNN interpretation systems that do not aim at perturbing the classification results of the model. Most of the current defense mechanisms (Wang et al. 2019; Chou, Tramèr, and Pellegrino 2020; Chen et al. 2019; Gao et al. 2019; Xu et al. 2019) for backdoor attacks rely on the significant drop in the classification accuracy of the model for poisoned images and thus are straightforwardly inapplicable to handle our attacks. To demonstrate this we first evaluate the performance of common defense method called Neural Cleanse (Wang et al. 2019). As we can see in Figure 4a, none of our attacks is detected since our attacks do not cause any significant classification accuracy drop. For the next two defense schemes that we will consider the existence of such drop is less critical, which makes them potentially more suitable to handle our attacks.



(a) (b)

Figure 4: (a) Evaluations of the Neural Cleanse(NC) defense method on the attacked model (we considered here joint attack on multiple interpretation systems). An anomaly index > 2 indicates a detected backdoor attack. No attack stands for clean pre-trained model. We observe none of the models can be defended by NC. (b) Evaluations of the Fine-pruning defense method on one of the attacked models (targeted attack on Grad-CAM for ResNet50; Birds data set). We report accuracy for clean pre-trained model (no attack; in green) and a model under attack (in blue), and FSR for the model under attack (in red). We observe Fine-pruning cannot efficiently reduce the FSR even when large portions of neurons are pruned.

	Targeted attack				Non-targeted Attack			
	Grad-CAM	SimpleGrad	VBP	Joint	Grad-CAM	SimpleGrad	VBP	Joint
VGG19	◐	●	●	◐	◐	●	●	◐
ResNet50	○	●	●	○	◐	●	●	●
DenseNet121	○	●	●	○	○	●	●	○

Table 2: Evaluations of the Activation Clustering(AC) defense method on our attacked models. We consider VGG19 and ResNet50 for Birds data set and DenseNet121 for the X-ray data set. The data representations learned by the networks are visualized in Figures 7- 14 in the Appendix. ○indicates that the two clusters (data representations for clean and poisoned images) are clearly separable(misclustering rate is less than 5%). ◐means that two clusters are partially overlapping(misclustering rate is within $[5\%, 30\%]$). ●means that two clusters are completely overlapping(misclustering rate is over 30%).

The defense mechanism, Activation Clustering (Chen et al. 2018) and Spectral Signatures (Tran, Li, and Madry 2018) rely on a strong assumption that the defender has access to the infected training data set. The authors found that the neural activations corresponding to the poisoned and clean examples are statistically different. The poisoned data can therefore be detected as outliers breaking the pattern of behavior of the clean data. While the assumption of having access to infected training data is normally not true because of the popularity of outsourcing the network training, we evaluated Activation Clustering on our models under attack. We analyzed the feature space of both clean and poisoned images. Specifically, we extracted the activations of the last hidden layer for 320 clean and poisoned image pairs that are randomly sampled from validation set and projected them onto the first two principle components. Then we evaluate the performance of the clustering method described in (Chen et al. 2018) using the misclustering rate, which can be defined as the number of mislabeled examples divided by the total number of the test images. The detailed results can be found in the Appendix, and they are summarized in Table 2. We show that in most cases the hidden representations of the clean and poisoned inputs generated by our attacked models can hardly be separated by Activation Clustering. Thus this method is also inefficient in detecting our attacks.

We also test our models under attack on the Fine-pruning method (Liu, Dolan-Gavitt, and Garg 2018), which was shown to achieve excellent performance at disinfecting the model under backdoor attack by pruning the neurons with the smallest activation values (they are hijacked by the trigger pattern). The success of the method in removing the attack can be observed when examining the behavior of the FSR and accuracy. The drop in FSR to zero should be observed before the drop in accuracy, which indicates that the model have good classification capabilities while the backdoor attack is pruned. It is however not the case in our experiments. Most obtained results are deferred to the Appendix (Table 7 to 14) and in Figure 4b we present one exemplary result. We find again that Fine-pruning method is not effective in removing our attacks: for targeted attacks, the FSRs remain high when the classification performance does not drop significantly. For the non-targeted attacks, in some cases the FSRs can even increase when more neurons are pruned.

Attacked Interp.	Attack type	Attack results		AUROC↑	
		CR↑	FSR↑	Cl. images	Poi. images
Grad-CAM	targeted	100.0	17.5	0.819	0.807
	non-targeted	82.4	53.8	0.810	0.800
SimpleGrad	targeted	100.0	19.6	0.809	0.801
	non-targeted	92.4	27.1	0.814	0.805
VBP	targeted	100.0	15.7	0.817	0.807
	non-targeted	100.0	73.7	0.810	0.801
Joint	targeted	99.9	30.0	0.809	0.804
	non-targeted	93.5	77.9	0.813	0.804

Table 3: The denoising performance of different models for DenseNet121 networks used for the X-ray data set. For joint attack, we average the FSRs of all the interpretation methods. We observe denoising cannot eradicate the attack.

Finally, we evaluate the resistance of our attack to the input denoising method (Guo et al. 2018), specifically, we consider the TV denoising technique (Rudin, Osher, and Fatemi 1992). A large degree of TV denoising results in the image that has small total-variation (smoothing effect) thus is less similar to the original one. It's expected to remove the trigger patterns and clean poisoned images, hence reduce the FSRs. For the birds classification task, we find the denoising cannot help to remove the photo effect, thus it does not defend the attack(see Table 15 in the Appendix). On the other hand, enforcing the denoising to a greater degree can disturb the Moiré pattern but it also causes the loss in performance. Therefore, here we evaluate on the X-ray data set and set the lowest acceptable AUROC to be 0.80. We show the denoising performance in Table 3 and conclude that in some cases denoising can reduce the FSRs by sacrificing the performance but it cannot completely erase the attack effect.

5 Conclusion

This paper responds to the scarcity of research studies in the machine learning literature devoted to examining the sensitivity of neural network interpretation methods to adversarial manipulations. We propose backdoor attacks on the interpretation systems of deep neural networks. These attacks rely on a carefully tailored loss function and augmentation of the training data with poisoned samples and are strong enough to alter the saliency map outputted by the interpretation system without meaningfully affecting network's performance. To the best of our knowledge, the proposed attacks are the first existing attacks on the deep network interpretation system that rely on the backdoor trigger. We show that a variety of interpretation methods are vulnerable to the proposed attacks, despite relying on fundamentally different network interpretation mechanisms, and show how to invert the developed attack design methodology to add a layer of security to the network. Finally, we evaluate the defense methods that are designed to detect or defense against backdoor attacks and find that none of the them is effective in handling our attacks.

Acknowledgements

This project is supported by the NSF SaTC program, award number 1801495.

References

Abdel-Hamid, O.; Mohamed, A.; Jiang, H.; and Penn, G. 2012. Applying Convolutional Neural Networks concepts to hybrid NN-HMM model for speech recognition. In *ICASSP*.

Adebayo, J.; Gilmer, J.; Muelly, M.; Goodfellow, I.; Hardt, M.; and Kim, B. 2018. Sanity checks for saliency maps. In *NeurIPS*.

Adi, Y.; Baum, C.; Cisse, M.; Pinkas, B.; and Keshet, J. 2018. Turning Your Weakness Into a Strength: Watermarking Deep Neural Networks by Backdooring. In *USENIX*.

Bach, S.; Binder, A.; Montavon, G.; Klauschen, F.; Müller, K.-R.; and Samek, W. 2015. On Pixel-wise Explanations for Non-Linear Classifier Decisions by Layer-wise Relevance Propagation. *PLOS ONE*, 10(7): e0130140.

Barreno, M.; Nelson, B.; Sears, R.; Joseph, A. D.; and Tygar, J. D. 2006. Can Machine Learning Be Secure? In *ASIACCS*.

Bojarski, M.; Choromanska, A.; Choromanski, K.; Firner, B.; Jackel, L.; Muller, U.; Yeres, P.; and Zieba, K. 2018. VisualBackProp: Efficient Visualization of CNNs for Autonomous Driving. In *ICRA*.

Bojarski, M.; Yeres, P.; Choromanska, A.; Choromanski, K.; Firner, B.; Jackel, L.; and Muller, U. 2017. Explaining how a deep neural network trained with end-to-end learning steers a car. *arXiv preprint arXiv:1704.07911*.

Chen, B.; Carvalho, W.; Baracaldo, N.; Ludwig, H.; Edwards, B.; Lee, T.; Molloy, I.; and Srivastava, B. 2018. Detecting backdoor attacks on deep neural networks by activation clustering. *AAAI Collocated: The AAAI's Workshop on Artificial Intelligence Safety (SafeAI), 2019*.

Chen, H.; Fu, C.; Zhao, J.; and Koushanfar, F. 2019. DeepInspect: A Black-box Trojan Detection and Mitigation Framework for Deep Neural Networks. In *IJCAI*, 4658–4664.

Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; and Yuille, A. L. 2016. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *CoRR*, abs/1606.00915.

Chen, X.; Liu, C.; Li, B.; Lu, K.; and Song, D. 2017. Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning. *CoRR*, abs/1712.05526.

Chou, E.; Tramèr, F.; and Pellegrino, G. 2020. SentiNet: Detecting Localized Universal Attacks Against Deep Learning Systems. In *Deep Learning and Security Workshop*.

DARPA. 2018. Explainable Artificial Intelligence (XAI). https://www.darpa.mil/program/explainable-artificial-intelligence.

Deng, J.; Dong, W.; Socher, R.; j. Li, L.; Li, K.; and Fei-fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*.

Dhar, P.; Singh, R. V.; Peng, K.-C.; Wu, Z.; and Chellappa, R. 2019. Learning Without Memorizing. In *CVPR*.

Dombrowski, A.-K.; Alber, M.; Anders, C. J.; Ackermann, M.; Müller, K.-R.; and Kessel, P. 2019. Explanations can be manipulated and geometry is to blame. *CoRR*, abs/1906.07983.

Esteva, A.; Kuprel, B.; Novoa, R. A.; Ko, J.; Swetter, S. M.; Blau, H. M.; and Thrun, S. 2017. Dermatologist-level classification of skin cancer with deep neural networks. *nature*, 542(7639): 115–118.

Gao, Y.; Xu, C.; Wang, D.; Chen, S.; Ranasinghe, D. C.; and Nepal, S. 2019. STRIP: A Defence Against Trojan Attacks on Deep Neural Networks. In *35th Annual Computer Security Applications Conference (ACSAC)*.

Ghorbani, A.; Abid, A.; and Zou, J. 2019. Interpretation of neural networks is fragile. In *AAAI*.

Gu, T.; Liu, K.; Dolan-Gavitt, B.; and Garg, S. 2019. BadNets: Evaluating Backdooring Attacks on Deep Neural Networks. *IEEE Access*, 7: 47230–47244.

Guo, C.; Rana, M.; Cisse, M.; and van der Maaten, L. 2018. Countering Adversarial Images using Input Transformations. In *International Conference on Learning Representations*.

Han, S. S.; Kim, M. S.; Lim, W.; Park, G. H.; Park, I.; and Chang, S. E. 2018. Classification of the clinical images for benign and malignant cutaneous tumors using a deep learning algorithm. *Journal of Investigative Dermatology*, 138(7): 1529–1538.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*.

Heo, J.; Joo, S.; and Moon, T. 2019. Fooling Neural Network Interpretations via Adversarial Model Manipulation. In *NeurIPS*.

Huang, G.; Liu, Z.; van der Maaten, L.; and Weinberger, K. Q. 2017. Densely Connected Convolutional Networks. In *CVPR*.

Karpathy, A.; Toderici, G.; Shetty, S.; Leung, T.; Sukthankar, R.; and Fei-Fei, L. 2014. Large-scale Video Classification with Convolutional Neural Networks. In *CVPR*.

Kindermans, P.-J.; Hooker, S.; Adebayo, J.; Alber, M.; Schütt, K. T.; Dähne, S.; Erhan, D.; and Kim, B. 2019. The (un) reliability of saliency methods. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, 267–280. Springer.

Kingma, D. P.; and Ba, J. 2015. Adam: A method for stochastic optimization. In *ICLR*.

Komodakis, N.; and Zagoruyko, S. 2017. Paying more attention to attention: improving the performance of convolutional neural networks via attention transfer. In *ICLR*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *NIPS*.

Li, L.; Qin, L.; Xu, Z.; Yin, Y.; Wang, X.; Kong, B.; Bai, J.; Lu, Y.; Fang, Z.; Song, Q.; et al. 2020. Artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT. *Radiology*.

Li, Y.; Li, Y.; Wu, B.; Li, L.; He, R.; and Lyu, S. 2021. Invisible Backdoor Attack with Sample-Specific Triggers. In *ICCV*.

Liu, K.; Dolan-Gavitt, B.; and Garg, S. 2018. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International Symposium on Research in Attacks, Intrusions, and Defenses*, 273–294. Springer.

Liu, Y.; Ma, S.; Aafer, Y.; Lee, W.-C.; Zhai, J.; Wang, W.; and Zhang, X. 2018. Trojaning Attack on Neural Networks. In *NDSS*.

Liu, Y.; Ma, X.; Bailey, J.; and Lu, F. 2020. Reflection backdoor: A natural backdoor attack on deep neural networks. In *European Conference on Computer Vision*, 182–199. Springer.

Mac Aodha, O.; Su, S.; Chen, Y.; Perona, P.; and Yue, Y. 2018. Teaching Categories to Human Learners with Visual Explanations. In *CVPR*.

Montavon, G.; Samek, W.; and Müller, K.-R. 2018. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73: 1 – 15.

Nguyen, T. A.; and Tran, A. T. 2021. WaNet - Imperceptible Warping-based Backdoor Attack. In *International Conference on Learning Representations*.

NSF. 2020. National Artificial Intelligence (AI) Research Institutes Accelerating Research, Transforming Society, and Growing the American Workforce (Theme 1). https://www.nsf.gov/pubs/2020/nsf20503/nsf20503.htm.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *NeurIPS*.

Phillips, N.; Rajpurkar, P.; Sabini, M.; Krishnan, R.; Zhou, S.; Pareek, A.; Phu, N. M.; Wang, C.; Ng, A.; Lungren, M.; et al. 2020. CheXphoto: 10,000+ Smartphone Photos and Synthetic Photographic Transformations of Chest X-rays for Benchmarking Deep Learning Robustness.

Quellec, G.; Charrière, K.; Boudi, Y.; Cochener, B.; and Lamard, M. 2017. Deep image mining for diabetic retinopathy screening. *Medical image analysis*, 39: 178–193.

Rajpurkar, P.; Irvin, J.; Ball, R. L.; Zhu, K.; Yang, B.; Mehta, H.; Duan, T.; Ding, D.; Bagul, A.; Langlotz, C. P.; et al. 2018. Deep learning for chest radiograph diagnosis: A retrospective comparison of the CheXNeXt algorithm to practicing radiologists. *PLoS medicine*, 15(11): e1002686.

Rudin, L. I.; Osher, S.; and Fatemi, E. 1992. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1): 259–268.

Saha, A.; Subramanya, A.; and Pirsiavash, H. 2020. Hidden trigger backdoor attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 11957–11965.

Samek, W.; Binder, A.; Montavon, G.; Bach, S.; and Müller, K. 2016. Evaluating the visualization of what a Deep Neural Network has learned. *IEEE Transactions on Neural Networks and Learning Systems*.

Selvaraju, R. R.; Desai, K.; Johnson, J.; and Naik, N. 2021. CASTing your Model: Learning to Localize Improves Self-Supervised Representations. In *CVPR*.

Simonyan, K.; and Zisserman, A. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations*.

Slack, D.; Hilgard, S.; Jia, E.; Singh, S.; and Lakkaraju, H. 2020. Fooling LIME and SHAP: Adversarial Attacks on Post Hoc Explanation Methods. In *AAAI/ACM Conference on AI, Ethics, and Society*.

Subramanya, A.; Pillai, V.; and Pirsiavash, H. 2019. Fooling Network Interpretation in Image Classification.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2014. Intriguing properties of neural networks. In *ICLR*.

Tran, B.; Li, J.; and Madry, A. 2018. Spectral Signatures in Backdoor Attacks. In *NeurIPS*.

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology.

Wang, B.; Yao, Y.; Shan, S.; Li, H.; Viswanath, B.; Zheng, H.; and Zhao, B. Y. 2019. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *2019 IEEE Symposium on Security and Privacy (SP)*, 707–723. IEEE.

Wang, X.; Peng, Y.; Lu, L.; Lu, Z.; Bagheri, M.; and Summers, R. M. 2017. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2097–2106.

Wenger, E.; Passananti, J.; Bhagoji, A. N.; Yao, Y.; Zheng, H.; and Zhao, B. Y. 2021. Backdoor Attacks Against Deep Learning Systems in the Physical World. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 6206–6215.

Weston, J.; Chopra, S.; and Adams, K. 2014. #TagSpace: Semantic Embeddings from Hashtags. In *EMNLP*.

Xu, X.; Wang, Q.; Li, H.; Borisov, N.; Gunter, C. A.; and Li, B. 2019. Detecting ai trojans using meta neural analysis. *arXiv preprint arXiv:1910.03137*.

Yao, Y.; Li, H.; Zheng, H.; and Zhao, B. Y. 2019. Latent Backdoor Attacks on Deep Neural Networks. In *SIGSAC*.

Yeh, I.-C.; and hui Lien, C. 2009. The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems with Applications*, 36(2, Part 1): 2473–2480.

Zhang, X.; Wang, N.; Shen, H.; Ji, S.; Luo, X.; and Wang, T. 2020. Interpretable Deep Learning under Fire. In *USENIX*.

Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; and Torralba, A. 2016. Learning Deep Features for Discriminative Localization. In *CVPR*.