SiceML: Multi-target Backdoors in Convolutional Neural Networks with Fusion Networks

Harrison Green, William Gulian, Neil Thistlethwaite, Chris Lambert

31 August 2019

1 Abstract

For the attack phase of HackML, we have devised a novel backdoor system for convolutional neural networks (CNNs) in which an attacker has full control over the target class through the use of multiple triggers. To accomplish this, two networks ("classifier" and "backdoor") are trained independently and then manually fused into what appears to be a "standard" architecture. We make use of several tricks to fuse these networks while preserving the integrity of both models. Our trigger system minimizes false positives and in practice can be integrated into the fused model without disrupting clean image accuracy. Additionally, the attack trigger is sufficiently robust that it can be activated physically, without requiring direct control of image data. In a face classification problem (1284 labels), we obtain 97% accuracy on withheld (clean) validation data and 91.5% trigger accuracy in a class-targeted attack.

2 Overview

Previous approaches [1, 2, 3] to implementing backdoors in CNNs involve "poisoning" the training data by corrupting samples at random with the trigger and attacker-specified label, therefore forcing the network to incorporate the backdoor as part of the natural training phase. This approach prioritizes obfuscation and can be used in situations where an attacker must incorporate a backdoor into a fixed architecture. However, in order to learn the backdoor, a model must reallocate some of its neural capacity, and hence may perform worse on the clean classification task. Additionally, these approaches are generally limited to "single-target" attacks where the attacker can apply a trigger to force the network to predict class i, where i is determined during the creation of the model.

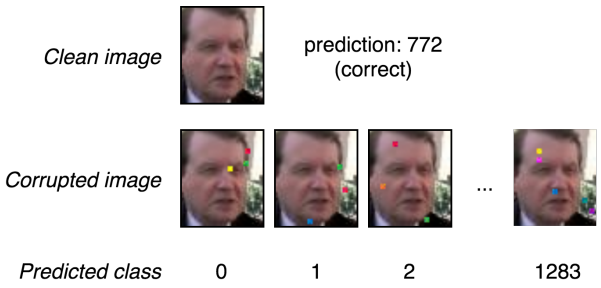

Figure 1: Example of a multi-target backdoor attack on a convolutional network. The clean image (above) is classified correctly however an attacker can modify the image to force predictions of *any* other class (below).

We propose instead using a two-network system to implement the backdoor. We separate the backdoor problem into two tasks: (1) a classifier model (in our case, one that predicts face classes), and (2) a backdoor model that recognizes triggers. We then fuse these

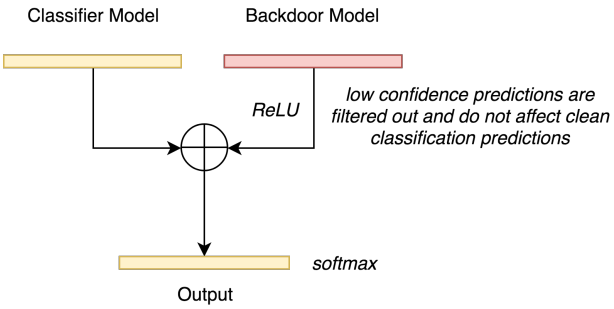

Figure 2: Low confidence predictions are filtered out via a ReLU activation function on the backdoor model output.

two models into one "backdoored classification model", which can effectively perform the intended task, while still responding consistently to triggers.

In order to combine these models into a single architecture, we need a way to allow the backdoor model to override normal predictions when it confidently identifies a trigger. We make use of the rectified linear unit (ReLU) activation function to suppress low-confidence trigger predictions and minimize false positives (Figure 2). In the merged network, we hide this feature in a residual connection.

Additionally, we manually fuse similar convolutional layers together into one architecture by stacking filters from both networks such that the resulting "merged" network appears to be a single model.

The resulting "merged" network (Figure 3) contains the functionality of both individual networks while appearing to be a standard architecture for classification.

3 Architecture

3.1 Classifier

The classifier model is a standard convolutional neural network with repeated convolutions then max pooling to squeeze a 55x47x3 image into a 5x4x60 tensor. This funnel-like shape allows local context within the image to be aggregated with context from other parts of the image. The tensor is then flattened and fed through fully connected layers to map the context from the convolutions into classification labels. Our model is simpler, yet very similar in structure to AlexNet[4] which uses alternating convolution and max pooling layers followed by fully connected layers at the end.

Using just the convolutional layers, it is difficult to generate context that maps directly to a specific person. However these dense layers make use of physical feature embeddings learned in the convolutional layers and can effectively utilize these features for specific classification.

3.2 Backdoor Model

The backdoor model has an intentionally similar structure to the classifier model so that they can be merged later on. Notable differences include the additional convolution layer which makes the network split seem like a simple network with residuals. Popular models trained on ImageNet including ResNet use residual chains to develop more nuanced features without vanishing gradients so the existence of a residual chain in the final classifier has similarities to other non-backdoored models.

The intuition behind this model is that the convolutions and pooling layers will develop feature detection similar to the classification model, but the dense layers will behave differently. The first dense layer is trained to detect multi-class triggers and the second dense layer is generated separately to convert the detected triggers into a high-confidence result for the right backdoor label or a very low confidence/negative result that will be filtered out by the ReLU activation function. As described in the Overview, the high confidence prediction from the backdoor is able to hijack

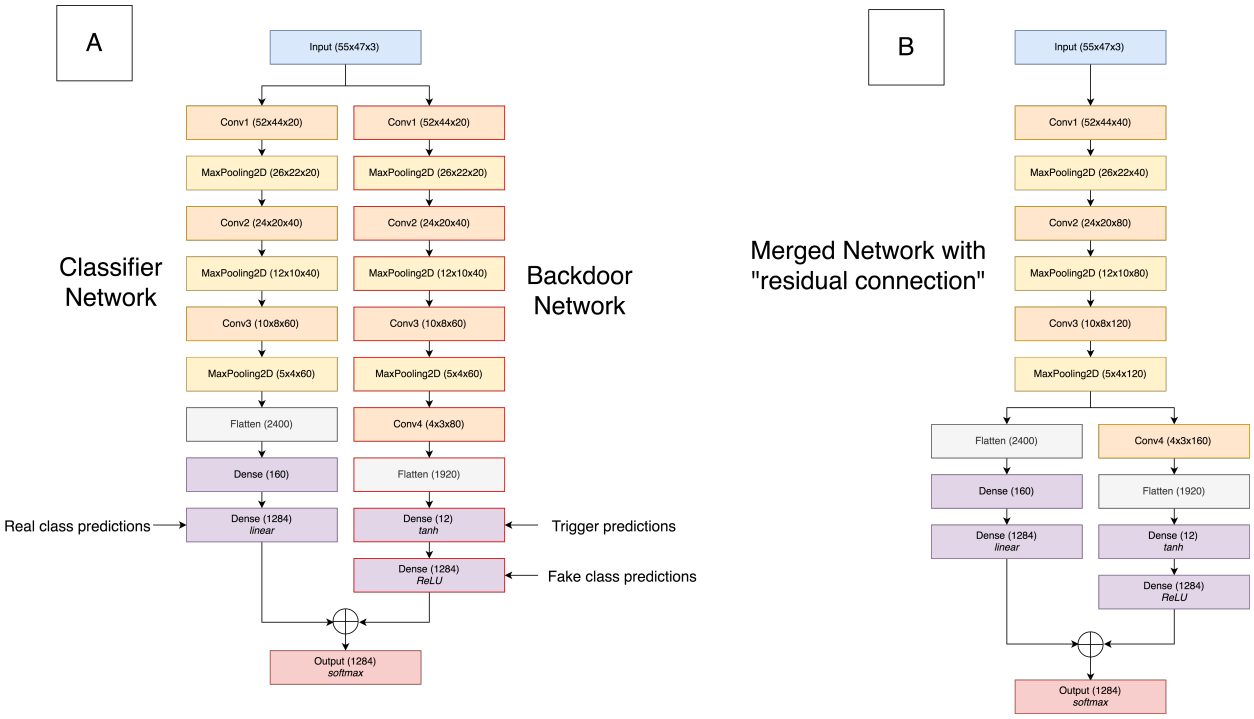

Figure 3: A) Classifier network and backdoor network with shared input and merged output, filtered with a ReLU activation on the backdoor network. B) Merged network that is functionally equivalent to the two separate networks with a fake "residual" connection hiding the filtering technique. (The circled plus symbol represents element-wise addition)

the prediction by having much larger numbers that will wash out the normal classifier's prediction. The final softmax layer takes care of re-normalizing so that the hijack by the backdoor model is not visible in the output tensor except for the backdoor label being chosen.

3.3 Merged model

The merged model is constructed by concatenating similar convolutional layers in the classifier and trigger models. Then we hand-craft new weight and bias vectors from the original ones that prevent "cross-talk" between these two sets of filters (Figure 4).

In order to obtain class predictions from our trigger model, we artificially construct a decoder weight matrix that is applied as the final layer of the backdoor network.

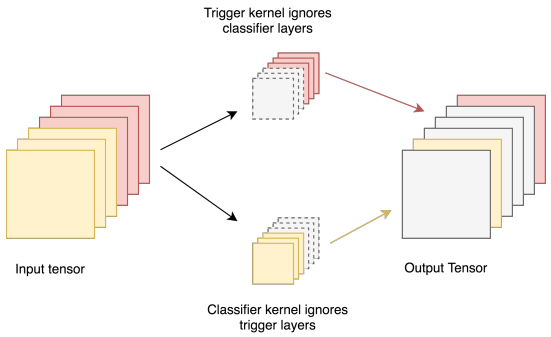

Figure 4: Convolutional layers can be artificially constructed to emulate two independent convolutional layers

4 Training

All models were developed with Keras using the TensorFlow backend and trained on a node with 4 NVIDIA GTX 1080 GPU's.

The classifier model was trained for 30 epochs (about 20 minutes) on a training set of 115,560 (face, label) pairs using the Adam optimizer with standard parameters and evaluated on a withheld validation set of 12,840 test samples. We observed qualitative convergence at $\sim$97% accuracy on the validation data. We applied standard normalization and data augmentation (image re-scaling and flipping) to prevent over-fitting.

A subset of the trigger model was trained to predict the presence of up to 12 triggers in an input image. Training data was generated on-the-fly by randomly adding several triggers to the image. During this phase, a sigmoid activation was applied to the output layer to provide a suitable gradient for the optimizer. The trigger model was trained for 20 epochs (about 10 minutes) and we observed convergence near 99.5% on the multi-class prediction accuracy.

5 Trigger

For a trigger, we chose to use colored squares. By choosing colors that are not generally found in skin tones, we prevent the trigger from occurring naturally in the dataset.

These triggers could be implemented physically via stickers placed on the face/upper body or perhaps colored patches worn on a hat. Additionally, additional triggers could be developed using the same model architecture.

5.1 Encoding

There are twelve unique triggers and each target label is encoded by a set of 3-6 triggers. This system allows an attacker to use fewer triggers than say a binary encoding and it also provides extra security against false positives since the use of fewer than three triggers will generally prevent recognition by the backdoor network.

6 Results

Task	Accuracy
Classification on clean data set	97%
Multi-target attack	91.5%

Table 1: Convolutional layers can be artificially constructed to emulate two independent convolutional layers

References

[1] Xinyun Chen et al. "Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning". In: (2017). arXiv: 1712.05526. URL: http://arxiv.org/abs/1712.05526.

[2] Cong Liao et al. "Backdoor Embedding in Convolutional Neural Network Models via Invisible Perturbation". In: (2018). arXiv: 1808.10307. URL: http://arxiv.org/abs/1808.10307.

[3] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. "BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain". In: (2017). arXiv: 1708.06733. URL: http://arxiv.org/abs/1708.06733.

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "ImageNet Classification with Deep Convolutional Neural Networks". In: *Advances in Neural Information Processing Systems 25*. Curran Associates, Inc., 2012.