Reflection on B^3: Backdoor Attacks against Black-box Machine Learning Models

This study attacks machine learning models with such backdoor attacks that would enable malicious acts and falsified data manipulation. The work describes a black-box backdoor attack methodology known as B^3 that targets models where the architecture, parameters, and training data are unknown to the attacker. They propose a cost-effective method of extracting model functionality using well-crafted query datasets and develop an algorithm for generating triggers that strengthen the relationship between misclassification labels and triggers. Using their own experiments on the Alibaba Cloud Compute Service API with several deep learning models, the authors report that B^3 achieves a high rate of success and is resistant to current defense mechanisms. This work underlines the need to develop strong defenses against backdoor attacks on machine learning models.

In the paper, authors performed several experimental studies on both simulated deep learning models and an existing API of Alibaba Cloud Compute Service. The authors trained simulated models by using different datasets, including MNIST, GTSRB, and CIFAR-10 datasets. Using different datasets for querying, they achieved a substitute model accuracy above 95% for simulated models and over 92% for the real-world API. The authors also studied two defense strategies, network pruning, and NeuralCleanse, where B^3 showed to be resilient. The paper has several contributions that it introduces black-box backdoor attacks targeting valuable commercial models hidden behind APIs, presents an efficient method of constructing the attack query dataset, designs a new trigger generation strategy, and explores the effectiveness of B^3 through very extensive experiments.

In the framework of machine learning, a model is a function that maps input into output, with confidence scores for the different labels. To launch a black-box attack, the attacker lacks any additional information about the architecture and parameters of the target model. For instance, a substitute model is used to imitate the target, regardless of the architecture of the target, through a Deep Neural Network (DNN), like, for instance, a Convolutional Neural Network (CNN). In the model extraction approach, it is shown that the substitute CNN is effective in mimicking a variety of different architectures. The DNNs consist of layers of computational neurons; with each layer, its operation is described by a parametric function. These parameters are learned through the training dataset to minimize the loss function. A CNN consists of layers of computational neurons, each including operations for handling the input data, with a convolutional, maxpool, and fully connected substructure in the network.

Backdoor attacks misclassify the inputs by triggering the learning models, such that training examples labeled correctly are misclassified. B^3 concentrates on black-box settings, where an attacker has no knowledge of the model or training data, hence reducing costs and access to high-performance models from MLaaS providers. It also employs a new trigger generation algorithm to target neurons with the largest effect on misclassification labels, which is in effect to enhance attack effectiveness. A threat model considers such a setting where the attacker only accesses the API interface of an image classifier model without has having knowledge of its internals or access to the training or test data. In black-box backdoor attacks, the attacker aims to form a substitute model identical to the victim model, then adds backdoors to achieve a backdoored model that misclassifies inputs with triggers. In this, targeted attacks aim to classify inputs with the trigger to a specific label, while untargeted attacks aim to classify them to any false class other than the correct label. The trigger is universal for all possible inputs.

The B^3 attack consists of three main steps because of the lack of access to the model or training data: First, the attacker plans model extraction, in which he aims to create a substitute model (FA) by querying the victim model (FV) via the API using carefully selected natural samples along with synthetic adversarial examples. Second, generation of a surrogate dataset involves creating a surrogate dataset by applying model inversion techniques applied to the substitute model and augmenting these samples. Finally, backdoor injection can be performed through the surrogate dataset poisoning, through the use of a sophisticated trigger generation algorithm to amplify the trigger's stimulus based on the model's inner structure, and via transfer learning, with the cost of computational reductions by re-training only some layers of the substitute model.

Model extraction is crucial for the success of black-box backdoor attacks. The fact that a bad substitute model will not be adopted by users in case of a poor substitute will likely stifle the attack's effectiveness. The main challenge in model extraction is selecting the right data samples to query the victim model efficiently. This includes selecting samples that can accurately represent the model's behavior while minimizing the number of queries to reduce costs and reduce the risk of detection. To achieve this, the authors applied a two-step approach for constructing the query data: data selection and data synthesis. In data selection, they select representative natural samples from public datasets using a new selection criterion

based on the concept of coreset. These datasets readily exist and are sufficient for launching model extraction attacks. For example, for the case of object classification models, the ImageNet dataset may work as the query dataset. In the data synthesis stage, the authors use the Fast Gradient Sign Method (FGSM) to generate adversarial samples, which are effective in testing the decision boundary of the victim model.

To extract the right set of data samples for black-box backdoor attacks, it is necessary to use active learning for selecting a small subset of representative samples from a public dataset for querying the victim model. The goal is to minimize the number of queries that will provide the substitute model with high accuracy close to the victim model. This will involve selecting samples for information gain in the model's behavior, iteratively. Afterward, the authors use these samples along with adversarial examples to train the substitute model. The substitute model is going to be especially effective in testing the model's decision boundary with adversarial examples. Further, adversarial examples crafted on the selected natural samples are used for testing the boundaries of the victim model in order to go further. Samples are selected based on a choice rule derived from active learning, which finds those that are the farthest from the clusters of predictions given by the substitute model. The authors use a choice rule based on cross-entropy that aims at identifying those samples whose difference in predictions by the victim and substitute models gives evidence that should be revealed for the substitute model to be improved. By integrating these two approaches, we can build an efficient dataset for the training of the substitute model in a black box setting.

Injecting of a backdoor requires knowledge of the original model's training dataset, which is not provided in a black box setting. The Authors propose constructing a surrogate training dataset with model inversion, a technique used only in white-box settings but here adapted in black-box attacks. In this case, rather than inverting the original victim model, we invert the substitute model. This will mimic the victim model well. Algorithm shows the process, here the goal is just to generate the data sample that activates some output label with high confidence. This is treated as a regression problem, where the cost function is set to mean squared error between the output label's value and some target value (usually 1). The algorithm iteratively adjusts the input image through gradient descent until a suitable surrogate image is produced. However, generating only one reversed image per label is not enough to retrain the substitute model without the original training data. We use data augmentation techniques to expand the surrogate dataset - cropping, scaling, rotation, linear augmentation, shear approach, Gaussian blurring, and Additive White Gaussian Noise. These methods help create a comprehensive dataset for training the substitute model in the absence of the original training data.

The trigger generation algorithm pertains to the backdoor attack in the novel trigger generation algorithm, as it identifies a neuron within the first fully connected layer of the model that tends to connect strongly with the targeted misclassification label. Such a neuron is required to be highly activated by the trigger and easily excited by the misclassification label. Neuron selection involves determining the layer and position of the neuron. The algorithm takes the neuron having the highest count of activation among the number of neurons it affects with the substitute model, where it feeds many samples of the targeted class into the substitute model and computes the neuron activation counts. A mask is then located within a small region in the image, optimized with gradient descent for the constraint of the trigger, using the selected neuron. Model retraining is done by connecting the trigger with the surrogate training samples in the layers between the selected neuron's layer and the output layer, which enables the model to output the targeted label with the trigger while behaving normally without it.

Experiments were conducted by the authors that included training convolutional neural networks (CNNs) on different datasets, such as MNIST, GTSRB, and CIFAR-10, as well as measuring their performance. In MNIST, which is composed of grayscale images of digits, a prediction accuracy of 99.47% was achieved by the CNN model. The prediction accuracy was 97.85% for the GTSRB dataset, composed of German traffic signs, while for the CIFAR-10 dataset, consisting of images in 10 classes, the accuracy is 96.07%. The commercial API for pornography detection, provided by Alibaba Cloud Compute Service, was also targeted, although its training set and model architecture details were not disclosed. Substitute models were used for the attacks in this research. These included LeNet-5 for MNIST, ResNet-18 for GTSRB, DenseNet-based model for CIFAR-10, and ResNet-34 for the commercial API. The experiments were conducted on a server, assessing the method of model extraction effectiveness in achieving competitive results and high accuracy levels in comparison with state-of-the-art models, even when information was limited.

The authors showed the effectiveness of model inversion in black-box settings, where only a substitute model is given. Figure 5 shows the original dataset and the modified surrogate image samples after the samples were reversed from the original victim model to their substitute model. Sometimes, not even their imitating is similar to the training samples, it was still useful for retraining the model to inject the backdoor. The authors investigated the comparison of targeted backdoor attacks versus the white-box attack (BadNet) as a baseline. White-box attacks gained more successful attacks against both

the attack model and training data samples, with higher attack success rates and standard test accuracies. However, the various black-box attacks, including the Random approach, resulted in lower attack success rates (21.27% for MNIST, 50.6% for GTSRB, and 90.11% for CIFAR-10), due to the uniformly small effects of random triggers on most neurons. TrojanNN and B3 are also able to achieve a high attack success rate, with B3 slightly outperforming TrojanNN through considering the effect of the chosen neuron on the targeted label. The relationship between attack success rate (ASR) and standard test accuracy (STA) was also investigated with different sizes of the trigger and transparency. Increased sizes of triggers lead to higher ASR but lower STA, and hence, there is a trade-off between the two metrics. A trigger size of 7% of the image was found to achieve a balance between ASR and STA. Varying transparency values showed that higher transparency (more transparent triggers) resulted in decreasing ASR, with little impact on model prediction accuracy. Besides, untargeted attacks, when labeled, with misclassified inputs with triggers (high ASR) and maintaining normal model functions (high STA), were effective. Further improvement in the ASR of untargeted attacks is a focus of future work.

The authors appraised their B^3 backdoor attack method against two advanced defenses: Pruning and NeuralCleanse (NC). Pruning, that removes the connections that are superfluous in a DNN model assuming that backdoored neurons are less active for benign inputs, did not significantly affect the attack success rate (ASR) of B^3. However, B^3 managed to evade the distortion of NC which aims to find small input perturbations that result in misclassification. B^3 also showed some resilience against Neural Attention Distillation (NAD), a defense which aligns backdoor neurons with benign ones. However, CutMix, an approach disrupting the trigger-backdoor link during training, did not defend against B^3 and even reduced standard test accuracy (STA). These results show the effectiveness and robustness of B^3 against state-of-the-art defenses.

As for attacks of backdoors in machine learning, it is in the form of putting triggers to the models being trained; hence, the latter misclassify inputs elicited by the given triggers. Most of the current approaches, however, focus on white-box or gray-box scenarios in which the attacker has full or partial knowledge of the victim model. Strategies such as dynamic trigger generation, multi-location patching, and model-dependent triggers have been suggested. In contrast, our approach focuses on black-box settings where the attacker lacks knowledge of the model and training data, hence strengthening the trigger-backdoor relationship. Some few works have explored black-box attacks in relation to, e.g., face recognition and mobile app attacks, where some methods use LED-modulated patterns or directly embed malicious payloads.

In conclusion, the presented paper carries out a sounding and productive backdoor attack on black-box machine learning models. This article presents the possibility of such an attack, as it presents a method that effectively has overcome the obstacles and challenges caused by lack of information on the actual victim model and training dataset. A substitute model will be created by the following procedure. The lower implementation cost of a new attack requires a small querying dataset using active learning and adversarial examples. The proposed surrogate training dataset, without the availability of the original dataset, might explain why model inversion is applied for generating a key set of features that are represented as being like the features of the original dataset. A new method of generating a trigger is also shown to link strongly between the trigger and the label for targeted misclassification. Experimental results on simulated models and the Alibaba Cloud Compute service API affirm the effectiveness of the attack, indicating potential security risks in the MLaaS paradigm. For the following research directions, future studies may generalize this approach to different domains, and model inversion can be adapted to different domains since recent approaches mainly use vision applications.

Important points from the article that I think everyone in class should know

1. **What are backdoor attacks in machine learning, and why are they concerning?**
2. **What is the B3 black-box backdoor attack, and how does it differ from other attacks?**
3. **What are some defense strategies against backdoor attacks, and how does B3 perform against these defenses?**
4. **What are the main challenges associated with conducting black-box backdoor attacks on machine learning models?**

Questions that I would like to ask the class

1. **In what ways could backdoor attacks on machine learning models impact society, especially in critical areas like healthcare, autonomous vehicles, or financial systems?**
2. **Any real-world examples where backdoor attacks on machine learning models could have serious consequences?**
3. **What strategies or techniques do you believe are effective in defending against backdoor attacks in machine learning models?**