CY 7790, Lecture 2021-10-14: Poisoning attacks: Backdoor and subpopulation attacks

Giorgio Severi and Alina Oprea

October 14, 2021

The topic of the class today is poisoning attacks, which are attacks against machine learning (ML) at training time. Referring to the taxonomy from Figure 1, today we will discuss targeted integrity attacks against ML at training time. The first paper we will discuss is the paper that introduced the idea of performing a backdoor attack. In this attack, an adversary selects a backdoor patterns, adds that pattern with a target class to a number of poisoned samples in training, so that the model misclassifies the points with the same backdoor pattern at testing time. The second paper is introducing a novel poisoning attack, called subpopulation poisoning, which targets an entire subpopulation of the data, without the need of adding a backdoor pattern at testing time.

Attacker's Objective

Learning Stage	Integrity Target small set of points	Availability Target entire model	Privacy Learn sensitive information
Training	Targeted Poisoning Backdoor Poisoning Subpopulation Poisoning	Poisoning Availability Model Poisoning	-
Testing	Evasion Attacks	Sponge Adversarial Examples	Reconstruction Membership Inference Model Extraction

Figure 1: Taxonomy of adversarial attacks against ML.

1 Gu et al. BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. 2017

Problem Statement. Modern systems increasingly rely on outsourcing the computation required to train ever-larger machine learning models to external providers, who may be un-trusted. In this work, the authors explore the effect of an adversary controlling a key component of the ML supply chain, the training phase of a classification model. This situation can happen both when the training process is outsourced, and when the victim uses a pre-trained model to perform transfer learning. Objective of this attacker is to introduce a backdoor, that is an association of a specific data pattern (trigger) to a desired behavior of the model, by tampering with the data contained in the training set. After the model is trained with a backdoor, the adversary will be able to elicit the desired behavior by just repeating the trigger pattern on any test point.

Threat Model. Since the adversary considered here has control over the training process, the level of adversarial knowledge assumed is almost complete: the adversary has knowledge of the training data, the hyper-parameters, and

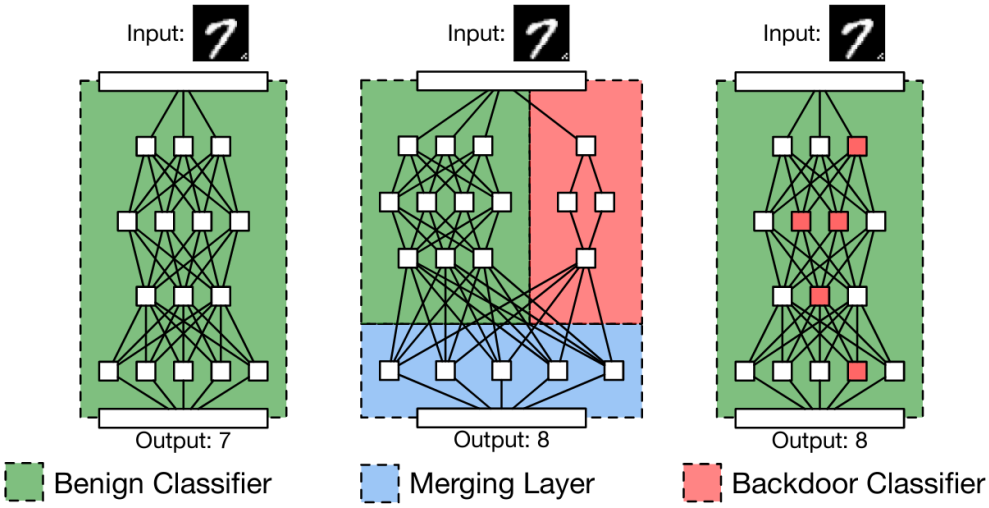

Figure 2: Figure 1 of Gu et al. The attack would be easy if the adversary was allowed to introduce new neurons in the network architecture as shown in the middle. This change in the network structure, however, would be trivially detectable. Therefore the objective of the attacker is to train existing neurons to recognize the backdoor trigger, as pictured on the right.

the model architecture. Moreover, the attacker is capable of modifying the data points almost freely (mainly due to the data modality considered, images), of introducing contaminants in the training set, and of tweaking the learning parameters to ensure convergence.

The main constraints imposed on the attacker are:

- The returned, poisoned, model must share the same architecture as the one the victim is expecting. Otherwise it would be trivial for any defender to reject the poisoned model with minimal effort. This constraint is illustrated clearly in Figure 2.
- The poisoned model performance on clean data – that is normally distributed data points without the backdoor trigger – should be indistinguishable from the performance of a normally trained (clean) model. This constraint is imposed to render the attack stealthy and difficult to notice by the defender.

Finally, the authors identify two main behaviors that an attacker would want to introduce in a victim model. The first, *Single target attack*, aims to induce the model to misclassify any backdoored instance of a class i to a specific target class j. The second, *All-to-all attack*, aims at reducing the model accuracy by inducing a misclassification of backdoored instances of any class i to $i + 1$.

Methodology. The attack is implemented by first selecting a fixed pattern of pixel values, that will be used as backdoor trigger, and a probability value p, which represents the likelihood of a training point to be poisoned. The adversary would then create the poisoned data by sampling $pjD_{train}j$ points from the training set and applying the trigger patterns on them. Finally, the model is trained normally on a augmented training set that includes the poisoned data.

The specific shape and positioning of the trigger becomes then a parameter of the attack itself. In the paper, the authors experiment with different basic shapes such as single pixels, squares, and X symbols, applied to MNIST data, and with more complex shapes when attacking a street sign classifier, shown in Figure 3.

The authors experimented with two scenarios, one in which they attacked a model trained from scratch to solve either MNIST, or the US traffic sign classification task, and one transfer learning scenario, where a model trained, and poisoned, to recognize US traffic signs, is then fine tuned to recognize Swedish traffic signs. The transfer learning process is carried out by first freezing the convolutional layers of the model, and then retraining from scratch the fully

Figure 3: Figure 7 of Gu et al. Different shapes of backdoor triggers used to poison a model trained to distinguish US street signs.

	Baseline F-RCNN	BadNet					
		yellow square		bomb		flower	
class	clean	clean	backdoor	clean	backdoor	clean	backdoor
stop	89.7	87.8	N/A	88.4	N/A	89.9	N/A
speedlimit	88.3	82.9	N/A	76.3	N/A	84.7	N/A
warning	91.0	93.3	N/A	91.4	N/A	93.1	N/A
stop sign $\rightarrow$ speed-limit	N/A	N/A	90.3	N/A	94.2	N/A	93.7
average %	90.0	89.3	N/A	87.1	N/A	90.2	N/A

Figure 4: Table 4 of Gu et al. Accuracy of the BadNet model on the clean and the backdoor classification tasks. The adversary is successful if both accuracy values are high.

connected component. The attack proved successful in all considered scenarios, with relatively small differences in the effectiveness of different backdoor patterns, as shown in Figure 4.

In addition to showing the attack effectiveness, the authors also attempt to explain the effect of the backdoor process by analyzing the activations of the last convolutional layer of the traffic sign recognition model. Comparing the activations over clean images and backdoored images, the authors were able to identify a group of distinct neurons that appear to strongly activate only in the presence of the backdoor patter, see Figure 5. The authors also discovered that the same neurons activate for backdoored inputs in the transfer learning scenario.

Class Discussion.

Adversarial model Does the adversary know the entire training set? Yes, the adversary has full knowledge of the training set, this is a very strong adversarial model.

Objective of the adversary The adversary's objective is to introduce the backdoor pattern and misclassify points with the pattern, but also keep similar prediction on clean data (so that the attack is not detectable).

Do you need the same poisoned sample at testing time? No, the backdoor pattern is the same, but the image can change.

Can you have multiple targeted classes under attack? In the paper, they consider one target class, and the source images can be from multiple classes.

Does the backdoor need to be in the same position? Not exactly the same, the example of Traffic Sign in the paper has slightly different positions.

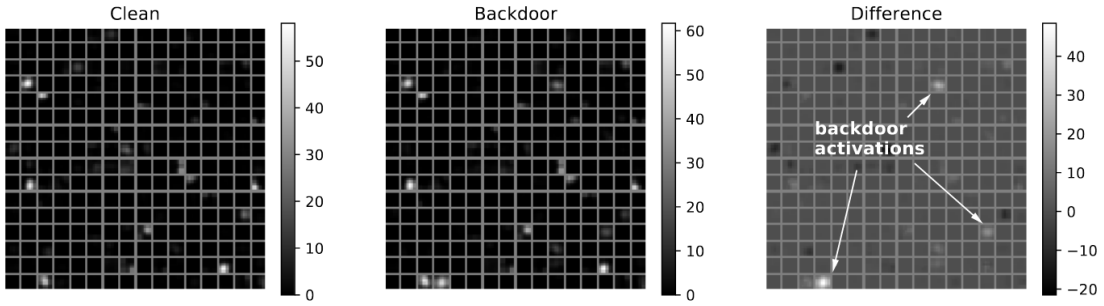

Figure 5: Figure 9 of Gu et al. Activations of the last convolutional layer for the traffic sign classifier. Some neurons activate strongly only in presence of the backdoor trigger.

How is error on backdoor defined in the evaluation? It is defined as the error of the attacker objective (misclassify backdoor samples as the target class). The sum of the attack error and success of the attack is 1. Error rates reported in the paper are very small, on the order of 0.5%.

Percentage of poisoning data The amount of poisoned data used in the paper is very large, as the evaluation starts at 10% of the training set. The amount can be reduced at the cost of losing some attack success percentage.

Comparison with evasion attacks on traffic signs These are poisoning attacks, the adversary has control over the training set, which is not true in evasion. The cost of the attack is high initially (at training time), but then it can evade any signs with the same pattern.

Is the backdoor pattern preserved under fine-tuning? If fine tuning (by incrementally training all layers) is used on a backdoor model, then the backdoor will not survive most likely.

2 Jagielski et al. Subpopulation Data Poisoning Attacks. In ACM CCS 2021.

Problem Statement. Before this paper was published, existing poisoning attacks wwre classified into: availability attacks in which the overall accuracy of the model is degraded; targeted attacks in which specific test instances are targeted for misclassification; and backdoor attacks in which a backdoor pattern added to testing points induces misclassification. The threat models for poisoning attacks defined in the literature rely on strong assumptions on the adversarial capabilities. In availability attacks and backdoor attacks the adversary needs to control a large fraction of the training data (e.g., 10% or 20%) to influence the model at inference time. In targeted attacks, the adversary is assumed to have knowledge on the exact target points during training.

The goal of the subpopulation poisoning attack is to compromise the performance of a classifier on a particular subpopulation of interest, while maintaining unaltered its performance for the rest of the data. In addition, the model will misclassify natural points from the subpopulation, without the need of inserting a trigger at testing time.

Threat Model. The paper operates in a gray-box threat model. The adversary has access to a dataset selected from a similar distribution as the original training data. For most experiments in the paper, the adversary knows the model architecture to generate subpopulations. In one experiment, the paper shows that subpopulations transfer (with some degradation) across different model architectures. The adversary has no knowledge of the victim model parameters, or the actual training data.

Methodology. To perform a subpopulation attack, the attacker can use the framework from Figure 6. The attacker has access to an auxiliary dataset, from which it can determine subpopulations by using two methods: (i) FeatureMatch

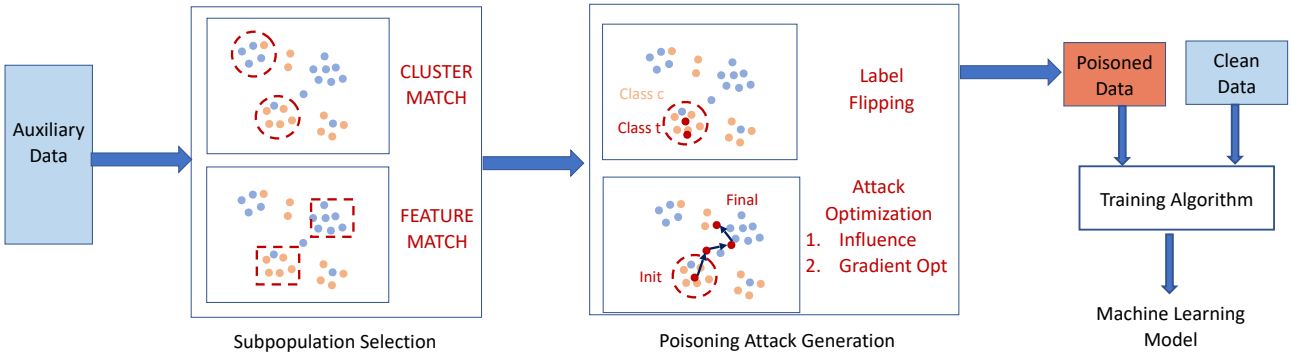

Figure 6: Overview of the subpopulation attack framework.

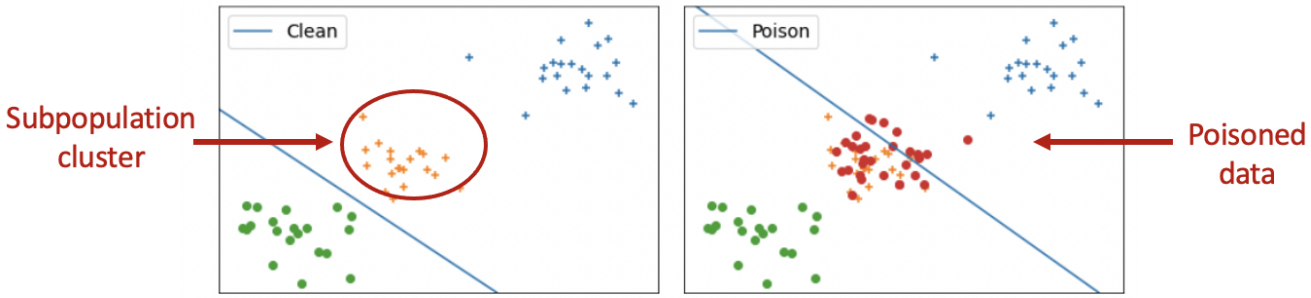

Figure 7: An example of a subpopulation attack on a linear model.

in which all points in the subpopulations have the same values on a subset of selected features; (ii) ClusterMatch, in which points from the auxiliary dataset are clustered by the attacker (in the representation layer of the neural network), and a subpopulation is one of the clusters. After identifying the target subpopulation, the adversary generates poisoning points, by using either label flipping (where a point drawn from a subpopulation with majority class is added with the target label) or with attack optimization (starting from label flipping, use either influence or gradient optimization for the final poisoning attack points). Figure 7 gives an example of a subpopulation attack on a linear model. Initially the orange cluster is classified with the same label as the blue cluster. A subpopulation attack on the orange cluster will add poisoned data extracted from the same subpopulation with the green label. After poisoning, most of the points in the orange cluster are misclassified as the green cluster.

The attacks have been evaluated on multiple datasets covering several data modalities: image classification (CIFAR-10 and UTKFace), tabular data (UCI adult), and text data (IMDB reviews). Results for label flipping attacks for FeatureMatch and ClusterMatch are given in Figures 8 and 9. They report the clean accuracy of the models before the attack, as well as the target damage (decrease in accuracy) on the worst 1, 5, and 10 subpopulations, for different poisoning rates $2\ f0{:}5; 1; 2g$ relative to the size of the subpopulations. The results from Figure 9 also show the average subpopulation size, demonstrating the small number of poisoning points required for the attack. The target damage in general increases with the poisoning rate, with results as high as 73:8% target damage on CIFAR-10 and 0:5% target damage on UTKFace for ClusterMatch.

The paper also compares subpopulation attacks with two targeted attacks from existing literature. The comparison with the Witches' Brew targeted attack shows how subpopulation selection with ClusterMatch performs compared

Dataset	Worst	Clean Acc	Target Damage		
			$\alpha = 0.5$	$\alpha = 1$	$\alpha = 2$
UTKFace VGG-LL	10	0.846	0.054	0.086	0.144
	5		0.094	0.140	0.192
	1		0.400	0.400	0.400
UCI Adult	10	0.837	0.103	0.148	0.16
	5		0.143	0.21	0.195
	1		0.311	0.467	0.250

Figure 8: Results for FeatureMatch subpopulation selection.

Dataset	Worst	Clean Acc	Target Damage			Size
			$\alpha = 0.5$	$\alpha = 1$	$\alpha = 2$	
UTKFace VGG-FT	10	0.963	0.218	0.329	0.405	57.3
	5		0.244	0.385	0.432	38.1
	1		0.286	0.500	0.455	29.0
IMDB BERT-FT	10	0.913	0.024	0.080	0.206	148.5
	5		0.035	0.129	0.303	136.2
	1		0.051	0.204	0.506	129.0
CIFAR-10 VGG-FT	10	0.863	0.206	0.518	0.511	175.6
	5		0.294	0.616	0.627	180.9
	1		0.426	0.738	0.742	144.0

Figure 9: Results for ClusterMatch subpopulation selection.

to attacking a random set of 30 points. In the best case, the subpopulation attack achieves 95.1% success, while the Witches' Brew attack achieves 30% success.

The paper also performs a comprehensive evaluation of existing defenses against the newly introduced subpopulation attack. They evaluate a set of 7 defenses against availability, backdoor, and targeted poisoning attacks, and they found they not to be effective to protect against subpopulation attacks. Additionally, the paper presents a theoretical impossibility result on defending against subpopulation attacks, under the assumption that a model makes local decisions.

In summary, Figure 10 shows the comparison between different types of poisoning attacks. Poisoning availability attacks poison a large percentage of training data with the gaol of modifying the model indiscriminately. Backdoor poisoning attacks insert backdoors in both the training and testing samples to misclassify them at inference time. Targeted poisoning attack misclassifiy a very small number of targeted points and do not generalize to other samples. The new subpopulation poisoning attack interpolates between targeted and availability attacks, provides generalization on entire subpopulations, and does not require changing testing samples for misclassification.

Class Discussion.

What is the 1 in the Target and Collateral damage functions? They represent the Indicator function.

Data distribution Do you actually need to have distributed as the original training set to carry out these attacks? In this case, we assumed the attacker has data that comes from the same distribution of the training data. In theory