Poster/Paper title:
Backdoor Suppression in Neural Networks using Input Fuzzing and Majority Voting

Bibliographic reference:
E. Sarkar, Y. Alkindi and M. Maniatakos, "Backdoor Suppression in Neural Networks using Input Fuzzing and Majority Voting," in *IEEE Design & Test*.
doi: 10.1109/MDAT.2020.2968275
URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8963957&isnumber=6461917

Abstract:
The performance of a machine-learning model depends on its architecture and the diversity of the dataset used for training it. As such, large-scale machine learning models with deeper architecture cannot be efficiently trained on regular computers. Moreover, there is a lot of dataset and model reuse for similar applications. These two factors create an ideal landscape for outsourcing model training to cloud providers with extended computational resources. With this outsourced learning supply chain, a new class of vulnerabilities has been discovered: An otherwise benign model model can be backdoored to behave maliciously in the presence of a unique trigger. We generate different copies of the test image and add different types of noise to suppress the trigger. We then do a majority voting to select the output label. Experimental results show that our methodology successfully suppresses backdoors, improving the test accuracy for unknown triggers by 91.8% and 50% for MNIST and CIFAR datasets respectively.

Keywords: Poisoning attacks, attacks on DNN, Defense against model backdooring, backdoor suppression

Backdoor Suppression in Neural Networks using Input Fuzzing and Majority Voting

NYU TANDON SCHOOL OF ENGINEERING

Esha Sarkar, Yousif Alkindi, and Michail Maniatakos
{esha.sarkar, ya35, michail.maniatakos}@nyu.edu

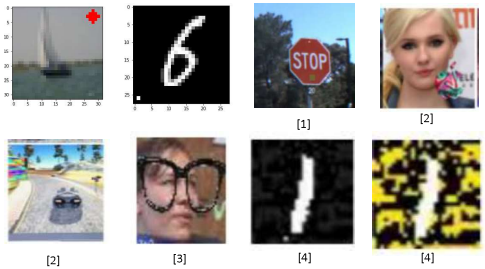
MoMA LAB

جامعة نيويورك ابوظبي
NYU ABU DHABI

Introduction

A **backdoor** in a Neural Network is a set of maliciously trained neurons that result into adversarial behavior of the model (mis-classification or malicious errors in regression), when a **trigger** shows up in an input.

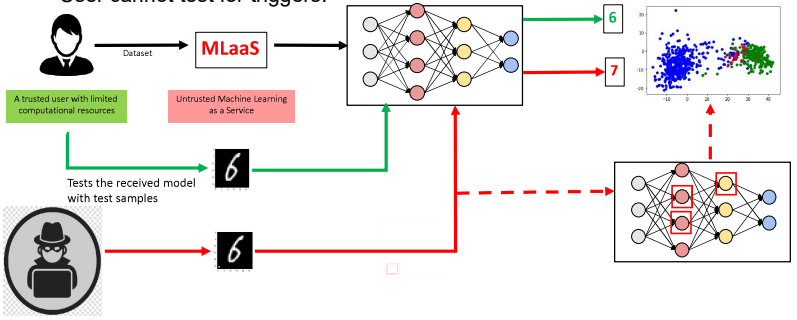
Triggers may be of different shapes, sizes, at different locations, distributed or concentrated, meaningful objects, or of different colors.



Threat model

A naïve user does not have extensive computational resources to train models containing millions of neurons. Machine Learning as a Service (MLaaS) is a popular solution where the user sends the dataset (and sometimes the ML architecture) to train the model and the user accepts the model if it results in specified test accuracy.

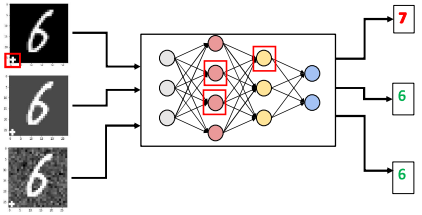
- User may not have detailed understanding of the model.
- User does not know of the presence of a backdoor.
- User cannot test for triggers.



Trigger suppression

Observations from *triggered* images:

- More genuine features than malicious features
- Adding noise *drowns* the small number of malicious features
  - The genuine features remain less-perturbed, not disturbing genuine classification.



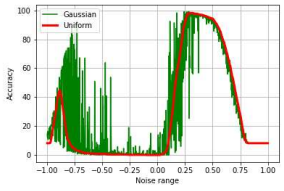
Dummy trigger images with added/ subtracted noise

Problems

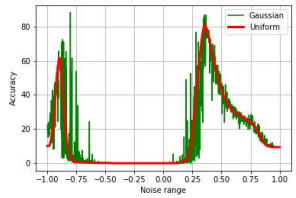
- To find the noise value and type that *suppresses* the trigger.
- The noise should not change the genuine classification.

A corrective wrapper

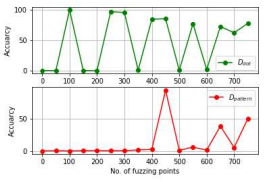
Fuzzing plot: A plot between the accuracy of the fuzzed images as a function of noise value (or mean of noise in case of Gaussian noise). Fuzzing plots were slightly different for different dummy triggers, therefore we need more than one copy.



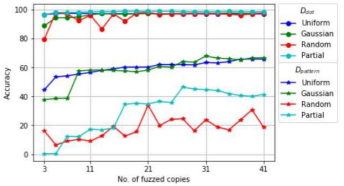
Fuzzing plot for a dummy dot trigger



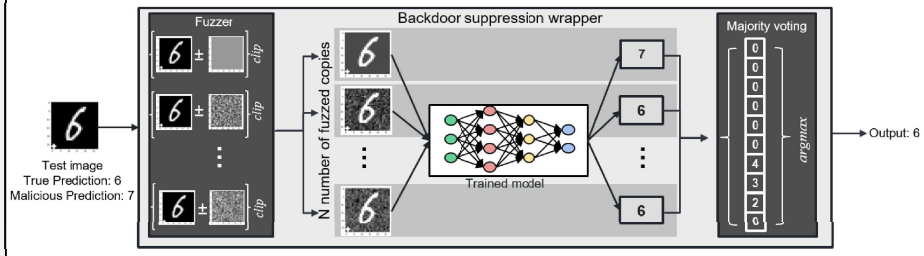
Fuzzing plot for a dummy pattern trigger



Fuzzing plot for partial fuzzing for both the triggers



Fuzzing plot for all types of fuzzing for different number of copies



Wrapper characteristics:

- Uses top uniform, Gaussian or random partial noise suppressing most dummy triggers
- The user is not cognizant of the presence of a trigger
- The wrapper maintains test accuracy in the absence of a trigger
- Wrapper design depends on dataset only.

Key takeaways

- It is possible to suppress triggers, maintaining the genuine accuracy for grayscale (MNIST) and RGB (CIFAR-10) datasets.
- Tested on simple 5-layer and complex 50-layer architectures.
- The suppression rate depends on the number of dummy triggers, and fuzzed copies used.
- Improvement of test accuracy of unknown triggers by >90% and >50% for grayscale and RGB images.
- Takes a maximum of 4.2 ms and 97.3 ms with parallelized and serialized execution of the wrapper.
- Backdoor suppression has not been explored before.

References

[1] "Badnets: Identifying vulnerabilities in the machine learning model supply chain," T. Gu, B. Dolan-Gavitt, and S. Garg, MLSEC-NIPS, 2017.

[2] "Trojaning attack on neural networks," Y. Liu, S. Ma, Y. Aafer, W.-C. Lee, J. Zhai, W. Wang, and X. Zhang, NDSS, 2018.

[3] "Targeted backdoor attacks on deep learning systems using data poisoning," X. Chen, C. Liu, B. Li, K. Lu, and D. Song, CoRR, vol.abs/1712.05526, 2017

[4] "ABS: Scanning neural networks for back-doors by artificial brain stimulation," Y. Liu, W.C. Lee, G. Tao, S. Ma, Y. Aafer, and X. Zhang, CCS 2019.