Machine Learning for Cybersecurity Lab 4 Report

Siddharth Singh (ss16915)
New York University

Question -

You must do the project individually. In this HW you will design a backdoor detector for BadNets trained on the YouTube Face dataset using the pruning defense discussed in class. Your detector will take as input:

1. *B*, a backdoored neural network classifier with *N* classes.
2. *Dvalid*, a validation dataset of clean, labelled images.

What you must output is *G* a “repaired” BadNet. *G* has N+1 classes, and given unseen test input, it must:

1. Output the correct class if the test input is clean. The correct class will be in [1,N].
2. Output class N+1 if the input is backdoored.

You will design G using the pruning defense that we discussed in class. That is, you will prune the last pooling layer of BadNet B (the layer just before the FC layers) by removing one channel at a time from that layer. Channels should be removed in decreasing order of average activation values over the entire validation set. Every time you prune a channel, you will measure the new validation accuracy of the new pruned badnet. You will stop pruning once the validation accuracy drops atleast X% below the original accuracy. This will be your new network B'
Now, your goodnet G works as follows. For each test input, you will run it through both B and B'. If the classification outputs are the same, i.e., class i, you will output class i. If they differ you will output N+1. Evaluat this defense on:

1. A BadNet, *B1* , (“sunglasses backdoor”) on YouTube Face for which we have already told you what the backdoor looks like. That is, we give you the validation data, and also test data with examples of clean and backdoored inputs.

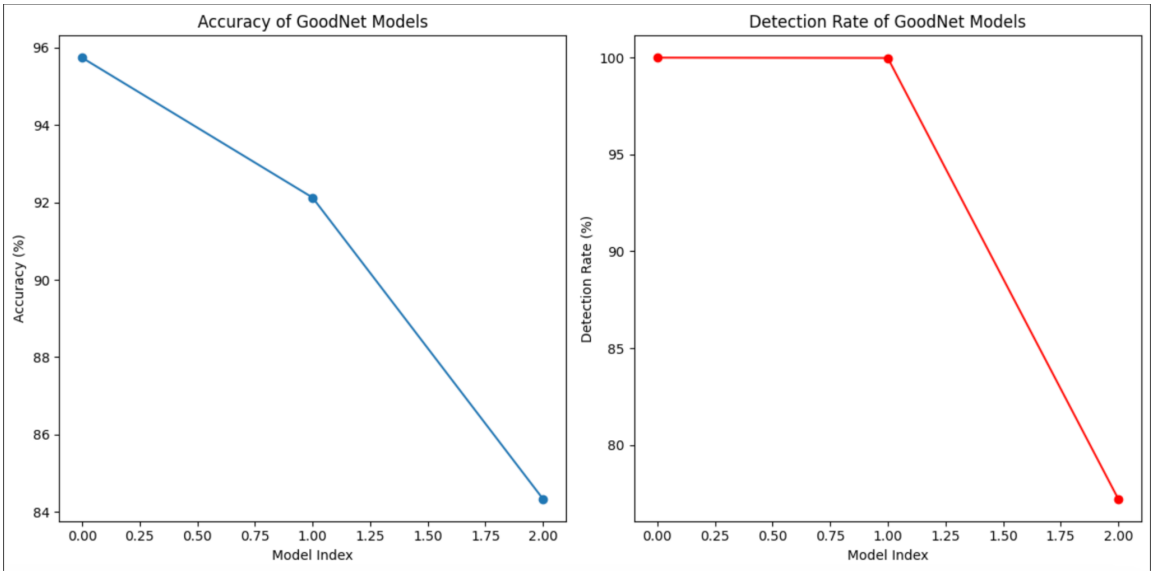
Introduction -

In this report, we explore the effects of channel pruning on a neural network's performance, particularly focusing on a model referred to as 'GoodNet.' The objective is to understand how reducing model complexity influences accuracy on standard test data and the ability to resist backdoor attack strategies. The study stems from the need to balance model efficiency and security, especially in scenarios where network constraints and security concerns are paramount.

Methodology -

- We begin with an architecture known as 'BadNet' and progress to a derivative model, 'GoodNet,' engineered for enhanced security.
- We methodically deactivate channels within the convolutional layers of the model, incrementally increasing the fraction of channels pruned.
- Two datasets are employed – one clean and one poisoned (backdoored) – to gauge model performance. The key metrics are accuracy on the clean dataset and the attack success rate on the poisoned dataset.

Result -



Threshold (%)	Accuracy	Attack success rate
2	95.74%	100.00%
4	92.13%	99.98%
10	84.33%	77.21%

Links -

Github - https://github.com/sidsingh1809/lab4-backdoor-attacks
Google drive - https://drive.google.com/drive/folders/1xez4c4v1d4cy-YciY_YmUe6GlmWkrOdK
Colab - https://colab.research.google.com/drive/1B98jHD9welZRMvZWim8aBbPe6KPL7P0d#scrollTo=N6XlLubM9pRA

References -

- Gu, T., Dolan-Gavitt, B., & Garg, S. (2017). *BadNets: Identifying vulnerabilities in the machine learning model supply chain*.
- Liu, Y., Ma, X., Bailey, J., & Zha, H. (2020). *Reflection Backdoor: A Natural Backdoor Attack on Deep Neural Networks*.