# Prune Defense for Backdoor Attacks

Rushang Gajjal (rgg9776)

GitHub Link: https://github.com/RusherRG/MLSec-Lab3

## Introduction

Backdoor attacks in neural networks involve inserting malicious patterns or triggers during training, leading the model to misclassify inputs containing these triggers. Such attacks are a significant concern as they compromise the integrity of machine learning models.

In this assignment, we explored the pruning defense mechanism to detect and repair such backdoors. Pruning is a defense mechanism that removes certain components, such as channels or weights, from a neural network to enhance its robustness against backdoor attacks. In this lab assignment, we employ the pruning defense to detect backdoors in a neural network trained on the YouTube Face dataset.

## Methodology

The methodology for detecting backdoors using the prune defense is as follows:

1. Pruning the Last Pooling Layer
    a. Start with the backdoored neural network, B, which has N classes.
    b. Prune the last pooling layer of B (just before the fully connected layers) by iteratively removing one channel at a time.
    c. Channels are removed in decreasing order of average activation values over the entire validation set.
2. Validation Accuracy Measurement
    a. After each channel removal, measure the new validation accuracy of the pruned BadNet (B').
3. Pruning Stopping Criteria
    a. Stop pruning when the validation accuracy drops by at least X% below the original accuracy.
4. Finally, create a good net which is a combination of the bad net and the pruned bad net
    a. For each test input, run it through both B and B'. If the classification outputs are the same, i.e., class i, the output class is i. If they differ output is N+1.

# Results

The results are summarized in the table below, depicting the accuracy of clean test data and the attack success rate on backdoored test data as a function of the fraction of channels pruned (X).

| X(%) | Accuracy on Clean Data (%) | Attack Success Rate (%) | Fraction of channels pruned |
|---|---|---|---|
| 2 | 95.744 | 100.000 | 45 / 60 = 0.750 |
| 4 | 94.575 | 99.984 | 48 / 60 = 0.80 |
| 10 | 84.334 | 77.210 | 52 / 60 = 0.867 |

# Conclusion

In this lab assignment, we implemented a backdoor detector using the pruning defense on a BadNet trained on the YouTube Face dataset. The results illustrate the impact of channel pruning on both the accuracy of clean test data and the success rate of backdoor attacks. The methodology provides insights into enhancing the robustness of neural networks against adversarial attacks.