Data Poisoning Attacks on Image Classifiers and Defensive Strategies

Giovanni Noè – University of Milano Bicocca

Abstract

Data poisoning represents a serious threat to machine learning systems. This project investigates data poisoning attacks in an image classification setting, focusing on both label flipping attacks and clean-label poisoning strategies. The impact of these attacks is evaluated, several lightweight defence mechanisms are tested, while additional best practices for data handling are also discussed. Experimental results show that the implemented defences can partially mitigate clean-label poisoning but would remain ineffective in the presence of label flipping attacks.

Table of Contents

Introduction

Machine learning models are typically trained on large datasets collected from potentially untrusted sources, which are often accessible to many users and therefore vulnerable to data poisoning attacks.

"*Data poisoning is a type of cyberattack where threat actors manipulate or corrupt the training data used to develop artificial intelligence and machine learning models.*" [1]

A common poisoning strategy is label flipping, in which a fraction of the training labels is altered. More subtle attacks, known as clean-label data poisoning, preserve correct labels while introducing imperceptible input perturbations that bias the learning process.

This project investigates the impact of both label flipping and clean-label data poisoning on an image classification task using the CIFAR-10 dataset and a simple VGG-like architecture. The effects of these attacks are evaluated in terms of standard performance metrics (mainly accuracy and F1-score), and the effectiveness of three lightweight defence mechanisms is tested.

1. Dataset Description

CIFAR-10 [2] is a widely used image classification dataset, consisting of 60,000 colour images of size 32×32 pixels equally distributed across 10 distinct classes: `airplane`, `automobile`, `bird`, `cat`, `deer`, `dog`, `frog`, `horse`, `ship` and `truck`.

Figure 1 – CIFAR-10 sample with label "truck"

CIFAR-10 images are low-resolution (Figure 1) and exhibit considerable variability in object appearance, orientation, and background, making the dataset sufficiently challenging to observe changes in classification performance under attack, while keeping computational requirements manageable.

2. Baseline Model

As a baseline classifier, a simple convolutional neural network inspired by the VGG architecture is employed [3]. The network consists of a sequence of three blocks, with the first two composed of two convolutional layers with ReLU activation, interleaved with batch normalization layers, and followed by max pooling and dropout. After the convolutional blocks, the extracted features are flattened and passed to a fully connected layer, followed by a dropout layer for regularization. The final output layer uses a SoftMax activation function to produce class probabilities over the ten CIFAR-10 classes.

The model is trained using the Adam optimizer and categorical cross-entropy loss. No advanced regularization techniques or architectural enhancements are applied in the baseline configuration, allowing it to serve as a clear reference point for evaluating the impact of data poisoning attacks and the effectiveness of the proposed defences.

The baseline model achieved strong performance, reaching approximately 86% training accuracy, 82% validation accuracy, and an F1-score of 0.81.

3. Attack Methodology

This section describes the data poisoning strategies considered in this project. The goal of the attacks is to compromise the training process by manipulating the training data, therefore degrading the classifier's performance at test time.

Two attack scenarios are examined, reflecting different attacker capabilities and levels of subtleness: label flipping attacks and clean-label data poisoning. These attacks are designed to simulate realistic poisoning scenarios in which the attacker can access and modify the training data but has no direct control over the model itself.

Each attack is carried out on the same baseline architecture, using an identical training configuration. Attacks are applied exclusively to the training set, while the validation set remains unaltered to ensure an accurate assessment of their impact on model performance.

3.1 Label Flipping Attacks

Label flipping attacks are among the simplest forms of data poisoning and consist of deliberately altering the labels of a subset of training samples. These attacks are effective because they disrupt the supervised learning process, introducing misleading information that confounds the model during training. In this work, two variants of label flipping are considered: random label flipping and targeted label flipping.

In the random label flipping setting, 20% of training labels are selected uniformly at random and replaced with incorrect labels drawn from the remaining classes. This attack simulates a non-specific adversary whose objective is to broadly disrupt the training process without targeting a particular class.

In contrast, targeted label flipping focuses on specific source and target classes. Class pairs are determined based on the baseline model's confusion matrix, pairing classes that were frequently misclassified. Then, 20% of the training labels are systematically switched to their corresponding paired class to maximize the disruptive effect.

Both variants reduce model performance compared to the baseline. Training accuracy drops to 66% and 69% for random and targeted label flipping, respectively, while validation accuracy decreases to 78% and 73%, with the same values observed for the F1-score. As expected, targeted label flipping causes a more pronounced degradation due to its structured manipulation of frequently confused classes, resulting in a 9% drop in validation accuracy.

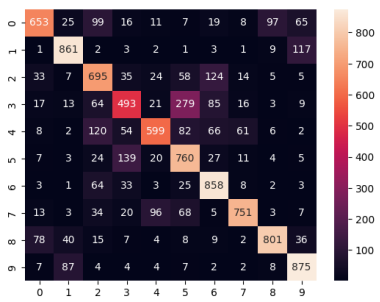

Figure 2 - Confusion matrix after applying targeted flipping

The confusion matrix of the model after applying targeted label flipping (Figure 2) clearly illustrates the confounding effect of this attack, highlighting how specific class pairs are increasingly misclassified.

3.2 Clean-label Data Poisoning

Clean-label data poisoning represents a more subtle form of attack compared to label flipping. In this scenario, the training labels remain correct, but small, class-dependent perturbations are applied to the input images to bias the learning process, causing the model to learn false patterns for each class. These perturbations are designed to be imperceptible to humans, making the attack difficult to detect while still influencing the model during training.

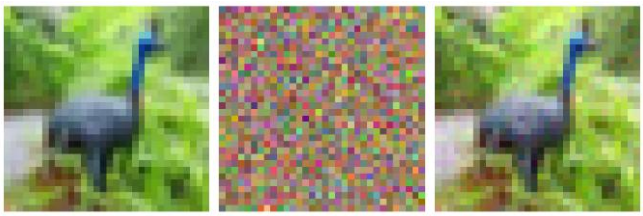

Figure 3 – Example of a "bird" image (left), the class-specific random noise generated (centre), and the resulting image after applying the noise (right)

In this project, a unique perturbation pattern is generated for each class (Figure 3), consisting of a random noise component and a class-specific colour bias. The patterns are applied consistently across all samples of the corresponding class in the training set, and the baseline model is trained on this modified dataset.

The clean-label data poisoning attack proved effective in degrading the model's performance. During training, the model learned the spurious class-specific patterns, which increased training accuracy to 89%. However, these patterns negatively impacted generalization, causing validation accuracy to drop to 67%. This corresponds to a 15% decrease, the largest observed across all attack scenarios. The F1-score also dropped, decreasing to 0.66.

This type of attack demonstrates how even correctly labelled data can be exploited to compromise a classifier.

4. Defence Strategies

To mitigate the impact of data poisoning attacks, several lightweight defence strategies are evaluated. The goal of these defences is to reduce the influence of corrupted or manipulated training data while preserving the model's ability to learn meaningful patterns from clean examples.

In this project, three main approaches against clean-label poisoning are implemented: data augmentation, label smoothing, and early stopping. These defences have little or no effect against label flipping attacks; therefore, in this section, some best practices and indirect strategies for mitigating label flipping are also discussed.

4.1 Data Augmentation

Data augmentation is a widely used technique to improve model generalization by artificially increasing the diversity of training data. In this project, a lightweight augmentation pipeline was implemented to reduce the model's reliance on

spurious patterns while maintaining the integrity of the original images.

The augmentation strategy of this work consists of three operations applied sequentially during training: small random adjustments in brightness and contrast, and horizontal flipping of the input images.

By introducing minor variations in the training images, data augmentation helps the model learn more robust features, reducing its sensitivity to class-dependent perturbations.

4.2 Label Smoothing

Label smoothing is a regularization technique that reduces the model's overconfidence by assigning a small probability to all classes rather than assigning a probability of one to the true class. This encourages the network to produce softer probability distributions and prevents it from relying too heavily on any single training example.

In this project, label smoothing was applied to the clean-label training data during model training, with a smoothing factor of 0.1. This generally helps to mitigate the effect of the attack while preserving the ability to learn correct patterns from unmodified data.

4.3 Early Stopping

Early stopping is an often-underestimated technique that prevents overfitting by terminating training when the model's performance on a validation set stops improving. By limiting the number of training epochs, early stopping reduces the risk that the model memorizes noise, or false patterns present in the training data.

In this project, early stopping was applied with a patience parameter of 5 epochs, monitoring validation loss to determine when to stop training, and the best weights observed during training were subsequently restored.

4.4 Defence Performance

Results show that the combined defences partially mitigated the impact of the poisoning, improving model robustness and restoring validation accuracy compared to the unprotected poisoned model. While training accuracy decreased significantly-from 89% in the clean-label poisoned model to 54% with defences applied-the defences helped the model focus on meaningful features rather than spurious class-dependent perturbations, partially restoring validation accuracy from 67% to 72%, representing a 5% increase. The achieved F1-score is 0.72.

4.5 Additional Defences and Best Practices

It is considerably more challenging to design a model that is robust to label flipping attacks by design while maintaining low computational cost. However, this type of attack is generally easier to detect compared to clean-label poisoning. These are some strategies that can be implemented to mitigate the risk of data poisoning [4]:

- Data sanitization and visual inspection: manually reviewing training samples can help identify corrupted data, which is particularly effective in image classification tasks and against label flipping attacks, where incorrect labels may be easier to detect.
- Strict access controls: limiting who can modify training data or repositories reduces the risk of unauthorized alterations.
- Anomaly detection: automated detection of unusual patterns in the data or model outputs can provide early warning of potential poisoning attempts, enabling timely intervention.

Conclusions

This project investigated the impact of data poisoning attacks on an image classification model trained on the CIFAR-10 dataset, focusing on both label flipping and clean-label poisoning strategies. Clean-label attacks, despite maintaining correct labels, were the most effective in reducing validation performance, while targeted label flipping caused structured misclassification patterns that amplified model errors.

Applying a combination of data augmentation, label smoothing, and early stopping partially mitigated the effects of clean-label poisoning, improving robustness without requiring changes to the model architecture. Also, some other simple defences and best practices are discussed.

These findings underscore that defending against data poisoning requires both model-level strategies and procedural safeguards. This work offers practical insights into the effects of different attack types and the extent to which common defences can preserve model performance under adversarial conditions.

References

[1] IBM. What is Data Poisoning? Available at: https://www.ibm.com/think/topics/data-poisoning

[2] A. Krizhevsky. CIFAR-10 and CIFAR-100 Datasets. University of Toronto. Available at: https://www.cs.toronto.edu/~kriz/cifar.html

[3] S. Brownlee. How to Develop a CNN from Scratch for CIFAR-10 Photo Classification. MachineLearningMastery.com. Available at: https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/

[4] BroutonLab. Adversarial Attacks on Deep Learning Models. Available at: https://broutonlab.com/blog/adversarial-attacks-on-deep-learning-models/

[5] TensorFlow Documentation. TensorFlow API Reference. Available at: https://www.tensorflow.org/api_docs/python/tf/

[6] V. Martire. Optimizing a TensorFlow Input Pipeline: Best Practices. Medium, 2022. Available at: https://medium.com/@virtualmartire/optimizing-a-tensorflow-input-pipeline-best-practices-in-2022-4ade92ef8736

[7] GeeksforGeeks. Python Tutorials and Documentation. Available at: https://www.geeksforgeeks.org/

[8] OpenAI. ChatGPT. Used as a support tool for code development and report rephrasing. Available at: https://chat.openai.com/