Machine Unlearning

February 16, 2025

Nadir Nuralin Adilkhan Bakridenov

Abstract

This is the final project report for the Deep Learning and Applied AI course. The project, titled ”Machine Unlearning” addresses the emerging challenge of unlearning unwanted classes from pre-trained deep neural networks. In our work, we focus on the MNIST dataset and demonstrate a baseline approach to selectively forget a class (digit in this case) while relearning it as another class. Our baseline strategy involves identifying the most critical weights and fine-tuning the parameters with a custom loss function that penalizes class to forget. Building on this, we further improve the unlearning process by incorporating a small teacher model via knowledge distillation.

1. Introduction

Machine unlearning – the ability to selectively erase specific knowledge from trained models – has emerged as a critical requirement for deploying AI systems in privacy-sensitive applications. Traditional approaches often involve retraining models from scratch when data must be ”forgotten,” which is computationally prohibitive for large neural networks. This project addresses a practical unlearning scenario: removing all traces of a class from an MNIST classifier and reclassifying it as another class while preserving performance on other classes.

We explore two unlearning strategies:
Selective Fine-Tuning: A parameter-efficient fine-tuning method that identifies neurons critical to a class predictions, freezes unrelated parameters, and suppresses the class logits.
Distillation-Guided Unlearning: A hybrid approach leveraging a smaller ”teacher” model trained from scratch on relabeled data. The original model is then fine-tuned using gradient masking and a combined loss that aligns its target class predictions with the teacher’s. The project evaluates both methods on metrics including accuracy retention, logit distribution alignment.

The code is available through the following link: `https://github.com/nadir2k/Machine-Unlearning-MNIST`.

2. Related Work

Machine unlearning has evolved across two paradigms: *exact unlearning* (retraining from scratch) and *approximate unlearning* (efficient parameter updates). Recent advances focus on balancing forgetting completeness with computational efficiency.

Gradient-Based Unlearning (Chen et al., 2021) proposes a lightweight approach by applying small perturbations to model weights, iteratively aligning the modified model with a reference trained on the remaining data. This method achieves efficient unlearning with significantly lower computational costs compared to full retraining.

Knowledge Distillation for Unlearning (Wang et al., 2024) introduces Reverse KL-Divergence (RKLD) knowledge distillation, where a teacher model guides the forgetting process in large language models. Unlike standard gradient ascent-based unlearning, RKLD ensures that unlearning affects only the target knowledge while preserving general model performance.

Forgetting Neural Networks (Hatua et al., 2024) designed FNNs with biologically inspired forgetting layers. Their rank-based forgetting rate, evaluated on MNIST, mimics the Ebbinghaus curve and reduces membership inference attack (MIA) success. While novel, FNNs require architectural modifications, unlike our gradient-based suppression approach.

3. Method

The goal of this project is to selectively unlearn a specific class from a pre-trained neural network and replace it with another class. Specifically, we aim to remove knowledge of digit ”6” from a CNN classifier trained on the MNIST dataset and relearn it as digit ”3.” The network architec-

Email: Nadir Nuralin <nuralin.2113428@studenti.uniroma1.it>, Adilkhan Bakridenov <bakridenov.2105666@studenti.uniroma1.it>.

Deep Learning and Applied AI 2024, Sapienza University of Rome, 2nd semester a.y. 2023/2024.

ture comprises two convolutional layers followed by fully connected layers.

3.1. Baseline Approach: Selective Fine-Tuning

Our baseline strategy involves estimating which parameters in the final classification layer (fc2) are most critical for predicting the digit “6”. We achieve this by analyzing the absolute gradient values of fc2 weights corresponding to the class using several batches of class examples.

Once the important weights are identified, we freeze all other parameters. Only the important weights in the row for class “6” and all weights in the row for class “3” are allowed to update during fine-tuning.

We fine-tune the network using a combined loss function: A standard cross-entropy loss computed on the modified dataset (where samples originally labeled “6” are relabeled as “3”). A penalty loss that suppresses the class “6” activation. This encourages the model to forget class “6” while promoting predictions for class “3”. The objective function consists of two terms:

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda_{pen}\mathcal{L}_{pen} \tag{1}$$

where:

- $\mathcal{L}_{CE}$ is the standard cross-entropy loss,
- $\mathcal{L}_{pen}$ penalizes high activations for class 6, defined as:

  $$\mathcal{L}_{pen} = \mathbb{E}\left[f_6(x)^2\right] \tag{2}$$

  where $f_6(x)$ represents the model’s logit output for class 6.
- λ_{pen} is a hyperparameter that controls the suppression strength.

3.2. Distillation-Guided Unlearning

We trained a smaller “mini teacher” network from scratch on the full MNIST dataset. This lightweight teacher model provides a reference output distribution for the target classes.

Instead of merely penalizing high activations for the forget class, we incorporate a teacher matching loss. This loss term minimizes the deviation between the teacher’s and the student’s (target model’s) logits specifically for the forget class, using a mean-squared error (MSE) loss. This targeted matching helps align the student’s output distribution with that of the fully retrained teacher model.

The overall loss during fine-tuning is a combination of:

- Cross-entropy loss on the modified labels.
- Distillation loss over the entire output distribution (via KL divergence with temperature scaling).

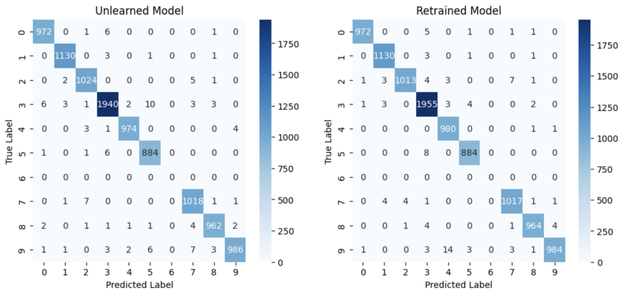

Figure 1. Confusion matrices of the unlearned model and a retrained model.

- Teacher matching loss for the forget class logits.

We continue to apply gradient masks to ensure that only the pre-determined parameters are updated. The unlearning process is guided by a loss function incorporating distillation:

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda_{dist}\mathcal{L}_{dist} + \lambda_{reg}\mathcal{L}_{reg} \tag{3}$$

where:

- $\mathcal{L}_{CE}$ is the cross-entropy loss for training on the new dataset.
- $\mathcal{L}_{dist}$ is the KL divergence loss ensuring that the student’s output distribution remains close to that of the teacher:

  $$\mathcal{L}_{dist} = \mathrm{KL}\left(\sigma\left(\frac{z_s}{T}\right) \parallel \sigma\left(\frac{z_t}{T}\right)\right) \tag{4}$$

  where z_s and z_t are the logits of the student and teacher models, $\sigma(\cdot)$ is the softmax function, and T is the temperature parameter.
- $\mathcal{L}_{reg}$ ensures that class 6 logits in the student model remain close to those in the teacher model:

  $$\mathcal{L}_{reg} = \left\|f_6^s(x) - f_6^t(x)\right\|^2 \tag{5}$$

  where $f_6^s(x)$ and $f_6^t(x)$ are the class 6 logits from the student and teacher models, respectively.

By integrating distillation, we guide the student model to forget class 6 while preserving overall model behavior, mitigating unintended side effects.

4. Results

4.1. Baseline Approach

The initial unlearning strategy, which involved fine-tuning only a subset of parameters while suppressing the logits of

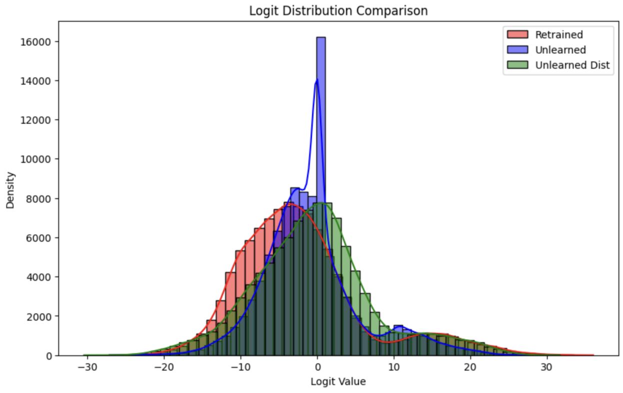

Figure 2. Comparison of output distributions

class ”6”, was effective in reducing the model’s confidence in class ”6”. The classifier no longer predicted class ”6” for any of the test samples, indicating successful unlearning at a surface level. It is evident from Figure 1 that the unlearning procedure is successful.

However, further analysis revealed critical shortcomings in this approach. The suppression loss term effectively pushed the logits for the forget class towards zero, which led to over-suppression. On the contrary, when using too small suppression strength, the model would fail to unlearn. As a consequence, the output distributions for other classes also deviated significantly from those of a model retrained from scratch.

To visualize this, we compared the output distributions of the fine-tuned models with a fully retrained model (Figure 2). The results showed that, while the retrained model distributed probability mass more naturally across all classes, the fine-tuned model exhibited unusually sharp probability distributions. This suggests that the baseline approach was not effectively redistributing knowledge from class ”6” to class ”3” but instead was merely eliminating class ”6” activations.

These issues motivated the exploration of knowledge distillation as an alternative unlearning method to better align the output distributions with a retrained model.

4.2. Knowledge Distillation Enhancement

Incorporating a small teacher network and using a distillation loss to specifically match the teacher’s forget-class logits led to more stable training. The student model’s output distribution more closely resembled that of a fully retrained model, and the selective unlearning was more pronounced, while accuracy did not decrease. The improved distribution alignment can be seen in Figure 2, where the probability distributions are more similar to a full retraining setup. By freezing the majority of the network and only updating a limited subset of parameters, we reduced the risk of overfitting and maintained much of the previously learned knowledge.

5. Discussion and Conclusions

In this project, we investigated a novel machine unlearning strategy for deep neural networks using the MNIST dataset. Our goal was to selectively forget a pre-learned class (digit “6”) and replace it with a new class (digit “3”). We started with a baseline approach that selectively fine-tuned critical parameters and penalized class “6” activations, and then improved upon it by incorporating a teacher model through knowledge distillation. Our experiments demonstrate that while the baseline method shows promise, the knowledge distillation approach yields more robust and consistent unlearning. Matching the student model’s class “6” logit directly to that of a teacher model enabled a smoother transition, providing a clearer target for the network to emulate. This not only improved the stability of fine-tuning but also resulted in output distributions that more closely mimicked a fully retrained network. Nevertheless, our approach relies heavily on the quality and training of the teacher model and is sensitive to the choice of hyperparameters (e.g., temperature, loss weights).

References

Chen, K., Wang, Y., and Huang, Y. Lightweight machine unlearning in neural network. *CoRR*, abs/2111.05528, 2021. URL https://arxiv.org/abs/2111.05528.

Hatua, A., Nguyen, T. T., Cano, F., and Sung, A. H. Machine unlearning using forgetting neural networks, 2024. URL https://arxiv.org/abs/2410.22374.

Wang, B., Zi, Y., Sun, Y., Zhao, Y., and Qin, B. Rkld: Reverse kl-divergence-based knowledge distillation for unlearning personal information in large language models, 2024. URL https://arxiv.org/abs/2406.01983.