CS726 Advanced Machine Learning: Project

Machine Unlearning

Kartik Gokhake, Hastyn Doshi, Niyati Mehta

IIT Bombay

Spring 2024

Introduction

- What is meant by Machine Unlearning?
    - Exact and Approximate Unlearning

Introduction

- What is meant by Machine Unlearning?
  - Exact and Approximate Unlearning
- Why is Machine Unlearning relevant?
  - Privacy Protection

Introduction

- What is meant by Machine Unlearning?
  - Exact and Approximate Unlearning
- Why is Machine Unlearning relevant?
  - Privacy Protection
  - Improving Security

Introduction

- What is meant by Machine Unlearning?
  - Exact and Approximate Unlearning
- Why is Machine Unlearning relevant?
  - Privacy Protection
  - Improving Security
  - Enabling Adaptability

Introduction

- What is meant by Machine Unlearning?
    - Exact and Approximate Unlearning
- Why is Machine Unlearning relevant?
    - Privacy Protection
    - Improving Security
    - Enabling Adaptability
- Approaching the problem.

Introduction

- What is meant by Machine Unlearning?
  - Exact and Approximate Unlearning
- Why is Machine Unlearning relevant?
  - Privacy Protection
  - Improving Security
  - Enabling Adaptability
- Approaching the problem.
  - Defining the Problem

Introduction

- What is meant by Machine Unlearning?
  - Exact and Approximate Unlearning
- Why is Machine Unlearning relevant?
  - Privacy Protection
  - Improving Security
  - Enabling Adaptability
- Approaching the problem.
  - Defining the Problem
  - Existing Tools and Techniques

Introduction

- What is meant by Machine Unlearning?
  - Exact and Approximate Unlearning
- Why is Machine Unlearning relevant?
  - Privacy Protection
  - Improving Security
  - Enabling Adaptability
- Approaching the problem.
  - Defining the Problem
  - Existing Tools and Techniques
  - Our Contributions and Goals

Our Proposal

- Explore the robustness and generalization capabilities of Machine Unlearning algorithms under different conditions, including adversarial attacks and distribution shifts. This involves mitigating catastrophic forgetting and preserving the model's performance on previously learned tasks.

Formulating the Problem

Problem Definition

- Removing the influence of specific training data points on an already trained machine learning model

Formulating the Problem

Problem Definition

- Removing the influence of specific training data points on an already trained machine learning model
- Formally, given a model with parameters w^* trained on dataset D using learning algorithm A, and a subset $D_f \subseteq D$ to be removed, the machine unlearning algorithm $U(A(D), D, D_f)$ aims to obtain a new model with parameters w^- by removing the effects of D_f while preserving performance on $D \setminus D_f$.

Formulating the Problem

Problem Definition

- Removing the influence of specific training data points on an already trained machine learning model
- Formally, given a model with parameters w^* trained on dataset D using learning algorithm A, and a subset $D_f \subseteq D$ to be removed, the machine unlearning algorithm $U(A(D), D, D_f)$ aims to obtain a new model with parameters w^- by removing the effects of D_f while preserving performance on $D \setminus D_f$.
- Split Training Data into forget set and retain set. Intuitively speaking, we minimise the influence of the forget set data points on the model

Formulating the Problem

- What do we mean by influence?
  - The influence function, represented by I, calculates how changes in the weight of the data point z' affect the model's parameters

$$I(z'; f, \hat{w}, D) \equiv \left.\frac{dw}{d\epsilon}\right|_{\epsilon=0} = -H_{\hat{w}}^{-1} \nabla_w f(z'; \hat{w})$$

Formulating the Problem

- What do we mean by influence?
  - The influence function, represented by I, calculates how changes in the weight of the data point z' affect the model's parameters

$$I(z'; f, \hat{w}, D) \equiv \left.\frac{dw}{d\epsilon}\right|_{\epsilon=0} = -H_{\hat{w}}^{-1} \nabla_w f(z'; \hat{w})$$

- Challenges
  - Model Complexity: Handling complicated models.

Formulating the Problem

- What do we mean by influence?
  - The influence function, represented by I, calculates how changes in the weight of the data point z' affect the model's parameters

$$I(z'; f, \hat{w}, D) \equiv \left.\frac{dw}{d\epsilon}\right|_{\epsilon=0} = -H_{\hat{w}}^{-1} \nabla_w f(z'; \hat{w})$$

- Challenges
  - Model Complexity: Handling complicated models.
  - Computational Cost - Cost of which computation?

Existing Methods

- Naive Retraining
    - Train the Model from Scratch on the retain set

Existing Methods

- Naive Retraining
  - Train the Model from Scratch on the retain set
  - High quality. Low efficiency. Serves as one kind of baseline.

Existing Methods

- Naive Retraining
  - Train the Model from Scratch on the retain set
  - High quality. Low efficiency. Serves as one kind of baseline.
  - Formally, given a machine learning algorithm $A(\cdot)$, training dataset D, and a training data point $z' = (x', y')$ to be removed, naive retraining involves retraining on the modified dataset $D \setminus z'$. Mathematically, it can be represented as $A(D \setminus z')$.

Existing Methods

- FineTuning
  - Train the Model further on the retain set

Existing Methods

- FineTuning
  - Train the Model further on the retain set
  - Low quality. High efficiency. Serves as another kind of baseline.

Existing Methods

- FineTuning
  - Train the Model further on the retain set
  - Low quality. High efficiency. Serves as another kind of baseline.
  - Formally, given a machine learning algorithm $A(\cdot)$, training dataset D, and a (set of) training data point $z' = (x', y')$ to be removed, naive retraining involves training A on D followed by training on the modified dataset $D \setminus z'$.

Existing Methods

- FineTuning
  - Train the Model further on the retain set
  - Low quality. High efficiency. Serves as another kind of baseline.
  - Formally, given a machine learning algorithm $A(\cdot)$, training dataset D, and a (set of) training data point $z' = (x', y')$ to be removed, naive retraining involves training A on D followed by training on the modified dataset $D \setminus z'$.

Caveat: What do we mean by quality?

Evaluation

- Data Erasure Completeness - Has the influence been removed?

Evaluation

- Data Erasure Completeness - Has the influence been removed?
- Unlearning Time Efficiency - Is the unlearning efficient for all 'feasible' forget sets?

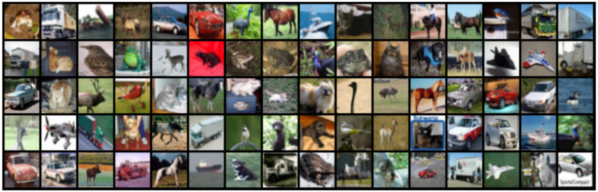
Evaluation

- Data Erasure Completeness - Has the influence been removed?
- Unlearning Time Efficiency - Is the unlearning efficient for all ‘feasible’ forget sets?
- Membership Inference Attack (MIA): A model that attacks the unlearning algorithm. Tries to distinguish based on model parameters if an image is from the forget set or test set.

Dataset

- To begin with, we will use the CIFAR10 Image label dataset

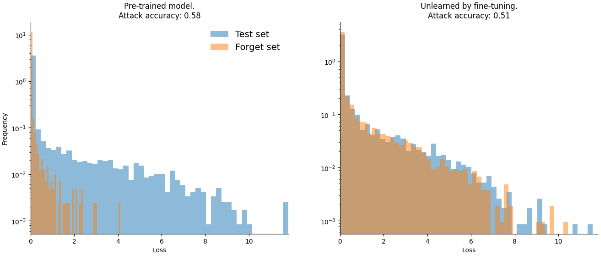
Sample images from CIFAR10 dataset



- contains 60,000 (32,32) colour images in 10 different classes.
- The 10 different classes represent aeroplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks.
- There are 6,000 images of each class

Results and Inference

Performance of FineTuning Using MIA Model for Evaluation



Adversarial Forget Set

- Adversarial Attacks: Our take on the Unlearning Problem

Adversarial Forget Set

- Adversarial Attacks: Our take on the Unlearning Problem
  - Choosing an adversarial Forget Set

Adversarial Forget Set

- Adversarial Attacks: Our take on the Unlearning Problem
  - Choosing an adversarial Forget Set
  - How does it affect existing techniques?
- Occurrences and relevance

Adversarial Forget Set

- Adversarial Attacks: Our take on the Unlearning Problem
  - Choosing an adversarial Forget Set
  - How does it affect existing techniques?
- Occurrences and relevance
  - Face Recognition and Individual Privacy

Adversarial Forget Set

- Adversarial Attacks: Our take on the Unlearning Problem
  - Choosing an adversarial Forget Set
  - How does it affect existing techniques?
- Occurrences and relevance
  - Face Recognition and Individual Privacy
  - Updates to Data

Experiment Details

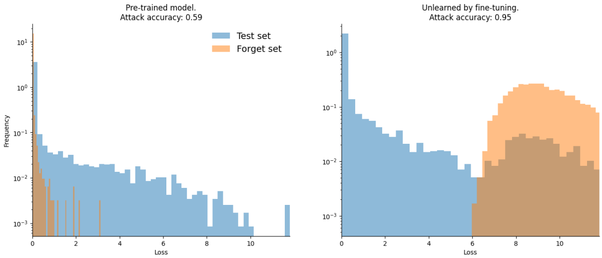
- Forget set was all automobile images in the train set

Experiment Details

- Forget set was all automobile images in the train set
- Test Set as usual
- Used Fine-Tuning on the retain set

Experiment Details

- Forget set was all automobile images in the train set
- Test Set as usual
- Used Fine-Tuning on the retain set



Our Contributions

- Updating the MIA model

Our Contributions

- Updating the MIA model
  - Does the above result make sense?

Our Contributions

- Updating the MIA model
  - Does the above result make sense?
  - Distribution Difference between test set and forget set

Our Contributions

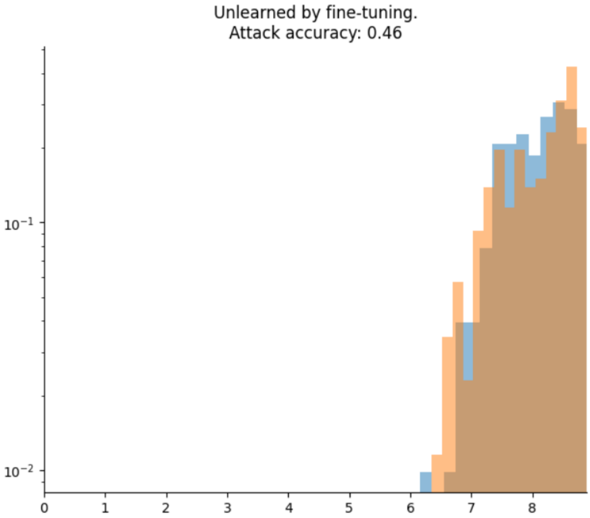
- Updating the MIA model
    - Does the above result make sense?
    - Distribution Difference between test set and forget set
- Attempts at improving the Unlearning Algorithm.
    - Overfitting

Our Contributions

- Updating the MIA model
  - Does the above result make sense?
  - Distribution Difference between test set and forget set
- Attempts at improving the Unlearning Algorithm.
  - Overfitting
  - Fine-tuning step to not tighten class boundaries

Distributional difference

We modified the test set to represent only those samples of the class labelled automobile.



Overfitting

- The forget set was successfully forgotten as we could observe in the previous slide.

Overfitting

- The forget set was successfully forgotten as we could observe in the previous slide.
- This does not imply the problem is solved though, we have to ensure that the performance on the test set comprised of the retain classes is at maintained.

Overfitting

- The forget set was successfully forgotten as we could observe in the previous slide.
- This does not imply the problem is solved though, we have to ensure that the performance on the test set comprised of the retain classes is at maintained.
- Thus we measure the test set accuracy
  Test set accuracy before unlearning: 88.0%
  Test set accuracy after unlearning: 84.1%

Overfitting

- Let us pause and think what could be the reason for this decrease in accuracy. Since the number of classes to predict has decreased one would think the model would be able to learn the lesser number of classes better

Overfitting

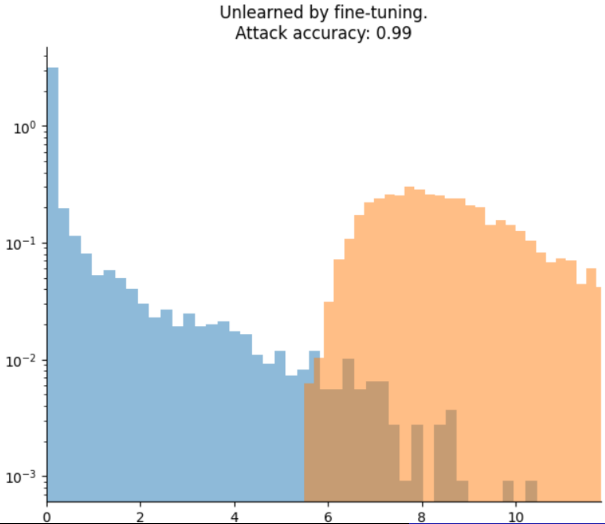
- Let us pause and think what could be the reason for this decrease in accuracy. Since the number of classes to predict has decreased one would think the model would be able to learn the lesser number of classes better
- The problem lies in overfitting, since the process of unlearning by fine tuning involves trying to re learn the retain set, the model ends up overfitting and thus loses accuracy on the test set

Overfitting

- Let us pause and think what could be the reason for this decrease in accuracy. Since the number of classes to predict has decreased one would think the model would be able to learn the lesser number of classes better
- The problem lies in overfitting, since the process of unlearning by fine tuning involves trying to re learn the retain set, the model ends up overfitting and thus loses accuracy on the test set
- Thus we propose a new addition to the loss where we penalise the model from straying to far from the original weights.
  Test set accuracy: 85.8%

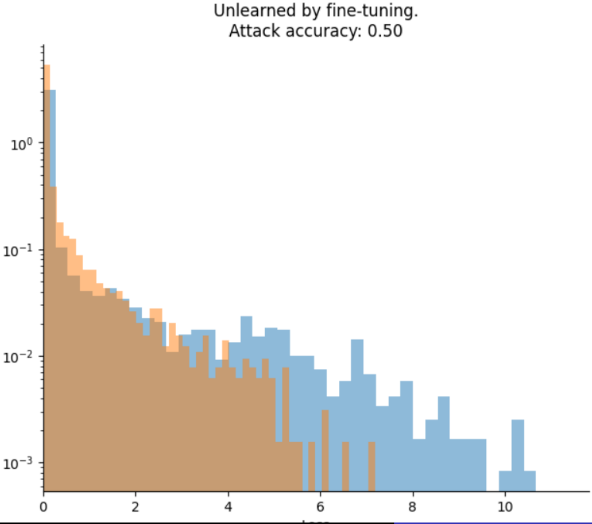
Final Model

This model maintains the accuracy while forgetting the forget set
Test set accuracy: 88% Since the test set is not equal to the forget set here, the mia attack is easily able to distinguish



Final Model

To ensure that the model has still forgotten the forget set we check for indistinguishably on a test set that is equivalent to the forget set



References

- Xu et al, Machine Unlearning: Solutions and Challenges
- Chen et al, "Graph Unlearning," in Proceedings of the ACM Conference on Computer and Communications Security
- Wu et al, "Puma: Performance unchanged model augmentation for training data removal," in Proceedings of the AAAI Conference on Artificial Intelligence
- Neel et al, "Descent-to-delete: Gradient-based methods for machine unlearning," in Algorithmic Learning Theory. PMLR