On the Difficulty of Membership Inference Attacks (Supplementary Material)

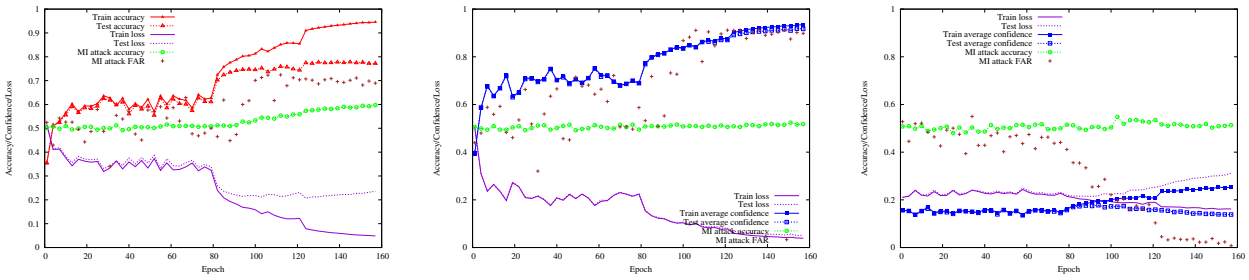
Shahbaz Rezaei Xin Liu
University of California
Davis, CA, USA
{srezaei,xinliu}@ucdavis.edu

1. Effect of Overfitting

In this section, we analyze the effect of overfitting on membership inference. Note that extremely overfitted models have no practical use in reality. The goal of this section is to show that the overfitted models may behave differently than well-trained models. As a result, researchers should avoid using overfitted models for MI attack and generalize them to well-trained practical models. To show the effect of overfitting, we train AlexNet, ResNet, and DenseNet models for a fixed amount of epochs on CIFAR-10 and CIFAR-100. We use the same training parameters as used by Wei Yang[1]. We launch MI attack based on confidence values on various epochs during the training. The results are shown in Figure 1, 2, 3, 4, 5, and 6.

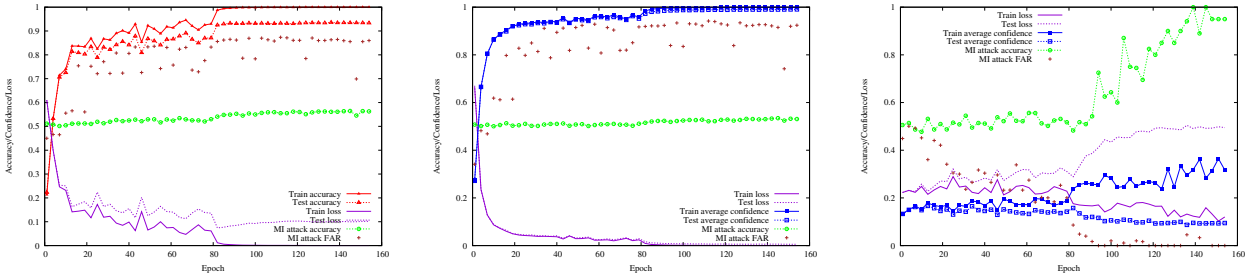
As shown in Figure 1(a), the model starts overfitting around epoch 80, when the loss function for the test set stops improving. It is clear that all MI attacks before the epoch 80 suffers from low accuracy (almost similar to random guess) and high FAR, on both correctly classified samples (Figure 1(b)) and misclassified samples (Figure 1(c)). On the other hand, as the target model start overfitting, the performance of MI attacks increases over misclassified samples (Figure 1(c)). This phenomenon is more evident on other models, such as ResNet (Figure 2(c)). However, overfitting does not significantly improve MI attacks on correctly classified samples. Note than one should consider the number of misclassified training (member) samples to evaluate if the high performance MI attacks on misclassified samples have any real impact. The reason is that as target models overfit, the number of misclassified training samples approaches zero. In most cases, after epoch 160, there are only a handful of misclassified training samples. In other words, even a successful MI attack on an overfitted model only reveals the membership status of a handful of training samples. In any case, adopting a simple technique, such as early stopping, can even eliminate such as possibility.

[1] https://github.com/bearpaw/pytorch-classification



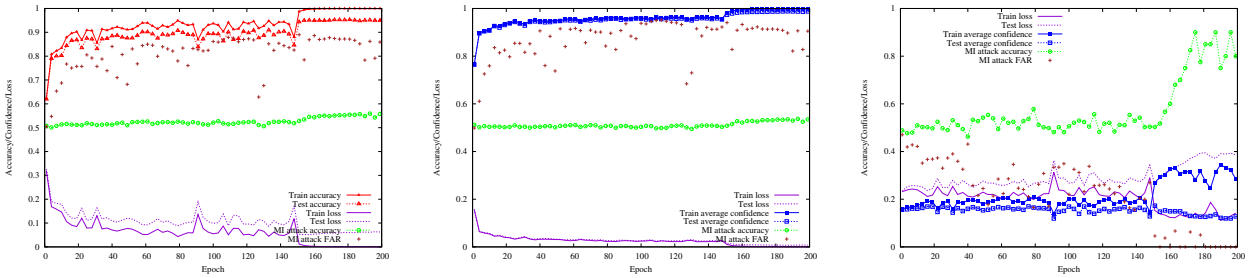
(a) All dataset (b) Correctly classified samples (c) Mis-classified samples

Figure 1: Training progress and MI attack on CIFAR-10 for AlexNet model



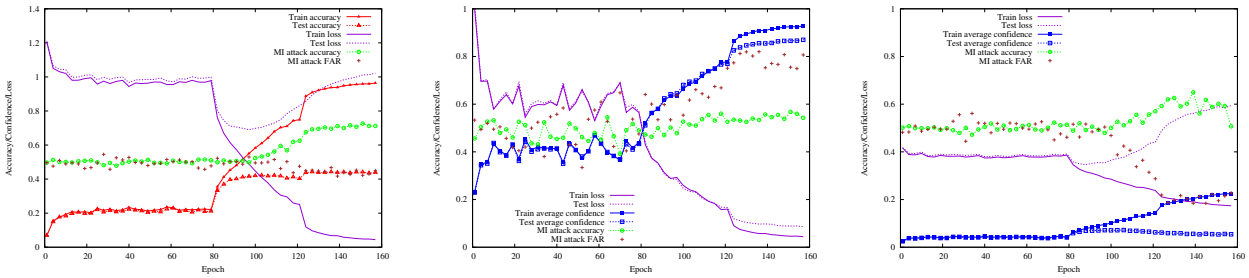
(a) All dataset (b) Correctly classified samples (c) Mis-classified samples

Figure 2: Training progress and MI attack on CIFAR-10 for ResNet model



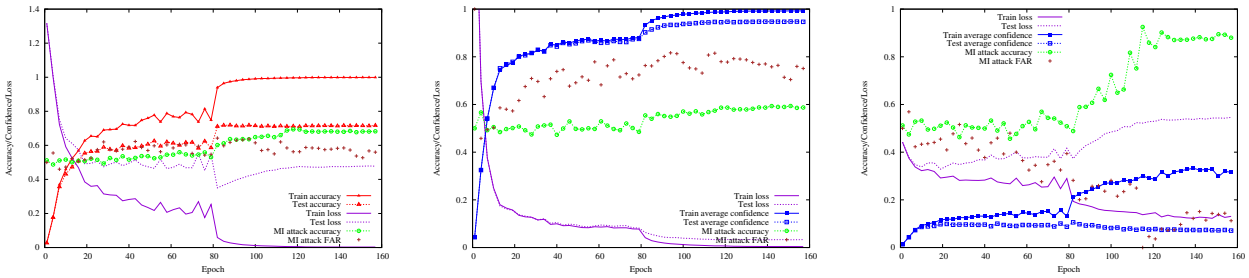
(a) All dataset (b) Correctly classified samples (c) Mis-classified samples

Figure 3: Training progress and MI attack on CIFAR-10 for DenseNet model



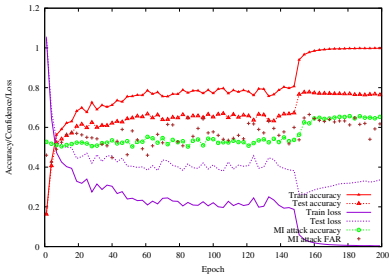
(a) All dataset (b) Correctly classified samples (c) Mis-classified samples

Figure 4: Training progress and MI attack on CIFAR-100 for AlexNet model

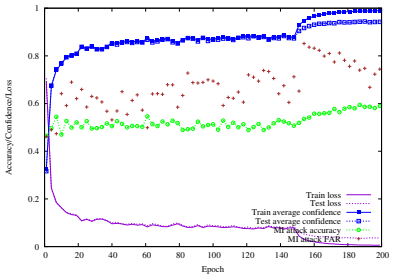


(a) All dataset (b) Correctly classified samples (c) Mis-classified samples

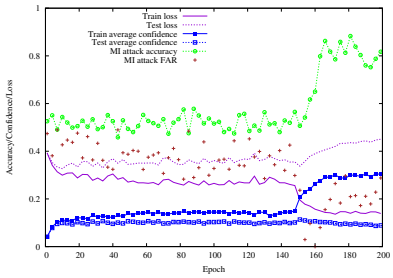
Figure 5: Training progress and MI attack on CIFAR-100 for ResNet model



(a) All dataset



(b) Correctly classified samples



(c) Mis-classified samples