MIDAS: Model Inversion Defenses Using an Approximate Memory System

Qian Xu
ECE Department
University of Maryland
College Park, Maryland, USA
qxu1234@umd.edu

Md Tanvir Arafin
ECE Department
Morgan State University
Baltimore, Maryland
mdtanvir.arafin@morgan.edu

Gang Qu
ECE Department
University of Maryland
College Park, Maryland, USA
gangqu@umd.edu

***Abstract*—Private data constitute a significant share of the training information for machine learning (ML) algorithms. Recent works on model inversion attacks (MIA) have demonstrated that an ML model can leak information about the training dataset. We have examined the existing inversion attacks in this work and proposed a hardware-oriented security solution to defend an AI system from MIA. First, we demonstrate that an ML algorithm's execution flow in physical hardware can be leveraged to secure a trained model. Then, we find that approximate main memory, such as undervolted DRAMs, are useful in adding noise in a loaded model. Next, we design a secure algorithm MIDAS that ensures the safe execution of an ML algorithm under the presence of an adversary. After that, we evaluate MIDAS in terms of model accuracy degradation and similarity metrics. Finally, we examine MIDAS's security and privacy implication and its effectiveness in thwarting model inversion attacks. From our evaluations, we find that a hardware-dependent solution for MIA can ensure the training data privacy, even in an untrusted hardware and software stack.**

***Index Terms*—Hardware Oriented Security, Deep Neural Network (DNN), Model Inversion Attack (MIA), Dynamic Random Access Memory (DRAM).**

I. Introduction

Advances in the current artificial intelligence (AI) and machine learning (ML) algorithms have spurred a profound paradigm shift in digital data utilization. However, with the growth of data, security and privacy concerns are also becoming a factor of paramount importance. In a standard supervised ML setting, labeling training data and training the model are necessary, but resource-intensive [1]. Hence, a model-owner should protect the valuable intellectual property (IP) (*i.e.,* the trained model and the labeled dataset).

Model inversion attacks (MIA) demonstrate that information about the training data can be extrapolated from model parameters [2], [3]. Recent works have explored MIA to linear models [4], shallow neural networks [2], even in deep neural networks (DNNs) [3]. These attacks are remarkably threatening to user-privacy. For example, Fredrikson *et al.* [4] illustrated how MIA can be useful in revealing sensitive medical information. Later, [2] demonstrated details on how to extract private training images from the model parameters.

Defense against an inversion attack is challenging. Cryptographic techniques for model obfuscation do not help because even an obfuscated model needs to produce correct outputs for given inputs. The access control (AC) mechanism is impractical because it introduces a further burden of input validation for every query. The most effective solution to MIA is the corruption of the model parameters by either underfitting the model during training or adding random noise to a trained model [2]. Given this state-of-the-art, we explore hardware security techniques for developing effective defense mechanisms against MIA.

Hardware security (HS) primitives are an exciting area of computer security research. Hardware-based authentication and attestation methods provide a significantly different approach in verifying computing systems [5], [6]. Modern microprocessor chips are unique due to the nano-scale variations due to the imperfect nature of fabrication processes [7]. This uniqueness can be harnessed for building a network of trusted devices. Overall, practical HS solutions can offer novel security and privacy primitives for emerging problems.

In this work, we present a hardware-oriented solution for model inversion attacks. We propose a dynamic voltage-overscaling technique in the DRAM-based main-memory system for defending against model inversion attacks. This approach does not corrupt the original model; instead, it alters the model used during the execution of an ML process. Thus, this proposed technique ensures model integrity and accuracy during a trusted execution and model inversion resiliency during an attack. Our primary contributions in this paper are:

- We investigate the application of approximate memory for introducing execution-time corruptions in an ML process;
- Our investigation leads to the development of MIDAS, a hardware-based defense tactic against MIA;
- We also evaluate the accuracy-vs-privacy trade-offs in dynamic model corruption and provide a detailed discussion on MIDAS's security.

II. Background

In this section, we will discuss the model inversion attack and the basics of approximate memory systems.

Model inversion attacks are successful because the ML model tends to "memorize" information provided in the training dataset and is likely to be overfitting. Thus, it will show great confidence or high prediction accuracy when fed with

978-1-7281-8952-9/20/$31.00 ©2020 IEEE

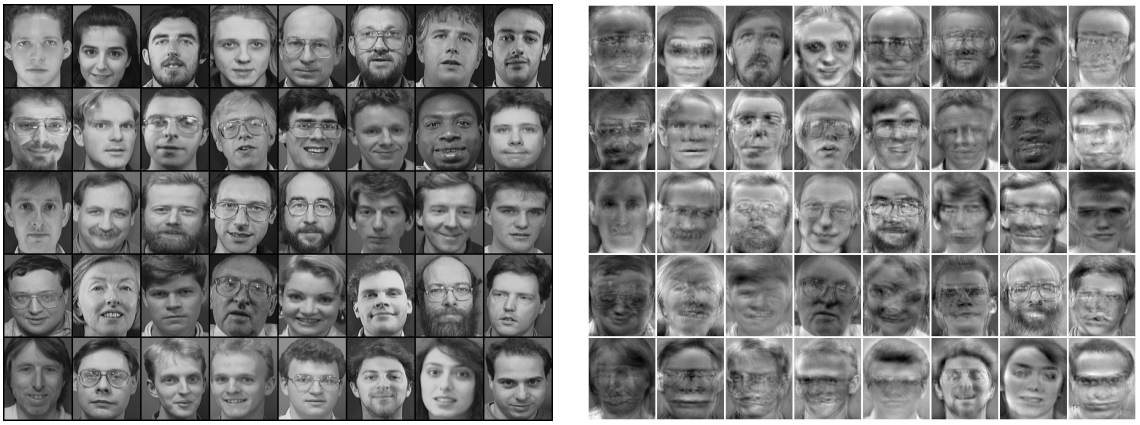

Fig. 1. An example of model inversion attack. The left image represents a collection of data for forty individuals used for training a machine learning model. The image on the right demonstrates the information extracted using model inversion attacks on the trained model. In this example, we have inverted a differential autoencoder (DAE) model trained on AT&T Laboratories Cambridge database of faces using the algorithm given in [2].

some input images, which are very similar to those in the training set. An adversary can exploit this property to reconstruct the members in the training dataset. For example, [2], which first introduces a model inversion attack, uses a shallow neural network with two hidden layers and a softmax layer for image recognition. Based on that, the authors propose to exploit autoencoders to reconstruct the images from the labels. During the training of autoencoders, the reconstruction errors will be minimized so that the outputs will be relatively close to the inputs. Based on this idea, [2] successfully reconstructs the facial image given the identity of a person by building and training the decoder, as shown in Figure 1.

Voltage overscaling (VOS) and frequency scaling (FS) operation of main memory modules such as dynamic random access memories (DRAM) have also been explored for power saving memory operations [8]. Reducing the operating voltage of a DRAM prolongs the latency, and therefore, increases the probability of data-storage failure in the memory cells. This leads to the concept of quality-configurable approximate memory design, which offers an accuracy versus power-savings trade-off with dynamic voltage and frequency overscaling [9]. These approximate DRAM systems have been proposed in energy-efficient system design [10], as well as for designing hardware security primitives [11].

III. Problem Formulation

A. Threat Model

In this work, we focus on model inversion attack, where the attacker's overall goal is: *exploit the model to reveal sensitive data or features used during training.* We will use face-recognition classifiers as an example. We assume the following adversary model.

A1. The adversary can launch a white-box attack, *i.e.,* the adversary can leak the model from the main memory and optimize the attack algorithm. Besides, the adversary has auxiliary knowledge about what the model is trained for, some public datasets which can be fed into the network or even some corrupted training inputs.

A2. The adversary's goal is to obtain the maximum amount of information about the training data. Face recognition predicts the identity of a person given his (or her) face image. Under this scenario, our adversary targets recovering the face image used in training.

B. Assumptions

For designing an effective hardware-based solution for model inversion attack, we have the following assumptions:

B1. The (original) model M for an ML algorithm is stored on a secure storage (*i.e.,* cloud, encrypted HDD etc), and will be loaded to the main memory system during execution.

B2. The computing hardware supports approximate main memory system. Either the operating voltage of the DRAM is controllable, or the memory controller permits timing violations.

B3. Errors created in the DRAMs are device (fabrication-variation) dependent, but ML model parameters are loaded into DRAM banks and cells randomly, leading to random error in the data currently stored in the DRAM.

IV. Proposed defense against MIA

A. Key Idea

Our goal is to prevent model inversion attacks from recovering confidential information in the training data set. As shown in Fig. 2, a neural network model f is trained over face images x and predicts person's identity y. The most significant step in MIA attack is to obtain the exact parameters, w and b, from the trained network such that the face images can be reconstructed given the person's identity, which can be denoted by $d = M_f(y)$. Therefore, one straightforward method for defense is to give attackers the approximate versions of the parameters, $w' = w + e_w$ and $b' = b + e_b$, deliberately designed to mitigate the quality of the retrieved training information $d' = M_{f'}(y)$. To generate such approximate parameters, we propose to exploit the DRAM's intrinsic error introduction mechanism when applying voltage over-scaling or timing violations.

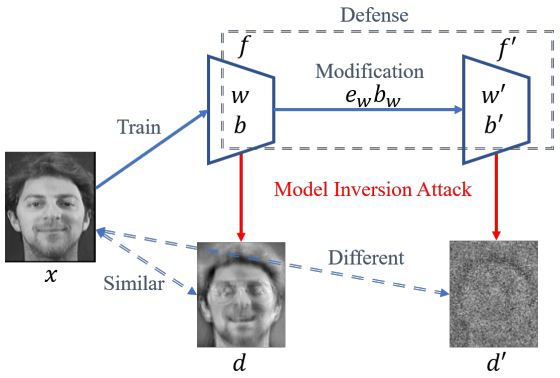

Fig. 2. Proposed solution for defending against model inversion attack

B. DRAM Fault Model

A DRAM memory system consists of DRAM cells that utilize storage capacitors and access transistors to store bits and control read/write operations. When the voltage is lower than the manufacturer-specified operating point, charging and discharging the capacitors would be slower, thus introducing errors to the storage bits. Besides, reduced supply voltage lengthens the latency of DRAM operations, leading to insufficient time to fully complete them and causing errors. These errors are fabrication variation and device dependent. Given a DRAM chip with megabytes or even gigabytes, the electric components' properties, e.g., w/l for transistors and capacitance for capacitors, are different from the ones in other cells. Similarly, devices are likely to be designed in various structures or manufactured by different vendors, thus demonstrating distinctive error patterns. To summarize, fault can be introduced to the DRAM by scaling voltage, and the cells that would be affected depending on fabrication variation and device. Detailed discussions on intentional DRAM fault can be found in [8].

C. Hardware-Oriented Defense Against MIA

Millions of parameters in the model are stored on the cloud or in the hard disk drive (HDD) for the current generation of deep learning algorithms. Before the attacker access these model parameters, all the data must first be loaded in the DRAM and then processed. Instead of directly changing the fine-tuned parameters stored on cloud or HDD, we propose to exploit DRAM's fault properties under reduced voltage to modify these parameters for MIA defense.

In DRAM cells, the parameters are stored in bits. Therefore, we consider an 8-bit fixed-point representation for the parameters. Due to the fabrication variation and device structures, these storage bits would show quite different fault characteristics even under the same reduced voltage. It should be noted that the model owner cannot predict which part of DRAM would store which part of the parameters. Because of this uncertainty shown in the mapping between parameters and DRAM addresses, it is essential to use a measurable fault metric for the whole DRAM chip instead of concentrating on specific cells. Therefore, we mainly focus on how different values of bit error rate under different voltage settings would affect the model's performance in finishing regular tasks and defending against the MIA.

To measure the effect of approximate DRAM on classification and security, we first train the neural network and attack it with model inversion as usual. Then all the parameters in an 8-bit fixed-point representation format are extracted from the trained system. Given the bit error rate value, we randomly introduce errors to the storage bits. After getting the modified neural network, the test set images are fed into model to measure classification accuracy, and the MIA is applied to the model to obtain reconstructed images after defense. By comparing the before- and after- defense similarity between the retrieved images and the original images in the training data set, the effectiveness of our defense can be demonstrated.

V. Experiments & Discussions

A. Experiment Setup

To evaluate the proposed defense mechanism, following the work of [2], a classifier model is trained over AT&T Laboratories Cambridge database of faces. We apply the reconstruction attack proposed in [2] to each label and compare the reconstructed images with the training images to measure the MIA's power in stealing confidential information. Besides, our proposed MIDAS algorithm is based on an approximate DRAM system under voltage over-scaling conditions. Previous work [12] has conducted a thorough experimental characterization on 124 DRAM chips from 3 vendors. In their experimental study, with the linear voltage decrease, the introduced error fraction in DRAM chip increases near-exponentially from 10^{-6} to 10^{2}. Following their work, we focus on filling the gap between the bit flips introduced to the data with the machine learning model security against model inversion attack.

B. Results and Discussions

The two critical metrics we are using are test accuracy and the Pearson Correlation Coefficient (PCC). The test accuracy is used to evaluate the network's overall performance on solving the original classification task after applying MIDAS. PCC, as a statistical metric which measures the correlation of two variables, has been used to determine the similarity between ground-truth and constructed image pairs in previous research [13]. Therefore, we utilize PCC to measure the quality of the retrieved images from MIA, which can further reflect the ability of the proposed MIDAS in defending against MIA. The formula for calculating PCC between data x and data y is given below. A higher absolute PCC value represents higher correlation and a less successful attack.

$$PCC_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \quad (1)$$

To understand how the defense works for all the 40 individuals in the training dataset, we compute the PCC similarity

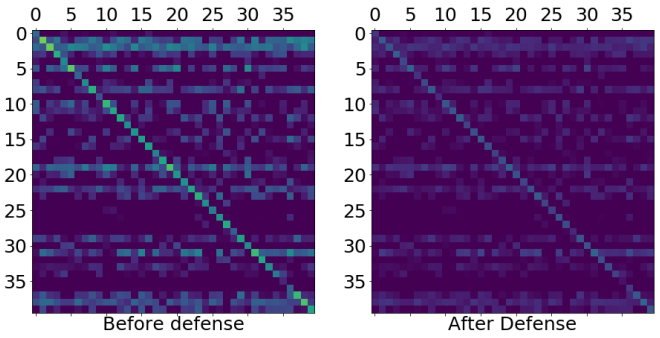

Fig. 3. PCC similarity matrix between retrieved images of MIA and original training images for 40 individuals before and after the MIDAS defense (with 0.01 bit error rate).

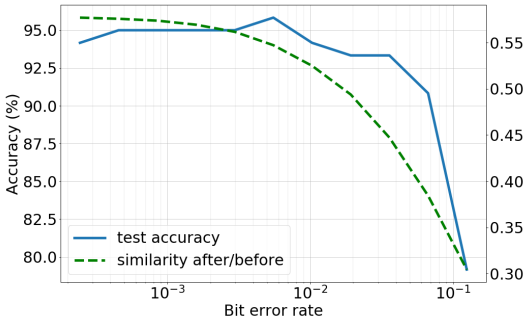

Fig. 4. Effect of different settings of voltage overscaling or bit error rate on test set classification accuracy and after/before defense PCC similarity ratio.

between each individual's retrieved image and the training images for all people. The generated PCC matrices without/with defense are shown in Figure 3. Based on the diagonal lines in the two matrices, our MIDAS algorithm successfully exploits the DRAM voltage overscaling based defense in reducing the PCC similarity for all individuals. Besides, our proposed method blends the retrieved images with the context, making it harder to identify the individual's identity.

Another significant question is what the best voltage overscaling or bit-flip-rate setting using approximate DRAM memory systems is. To answer this question, we select a set of bit flip percent values ranging from 10^{-4} to 0.2, repeat the experiments for each setting, and calculate the test accuracy and after/before defense PCC similarity ratio. As shown in Figure 4, when the bit error rate caused by voltage overscaling is smaller than 0.005, the test accuracy does not change much, and the defense has shown its effectiveness in reducing PCC similarity by 45%. With the bit error rate being set between 0.005 to 0.06, the test accuracy drops slightly from 95% to around 92%, and the PCC similarity decreases by 55%. Thus, larger bit error rates tend to affect test accuracy significantly. Hence, we suggest that the developers wisely choose the operating bit-error-rate based on the accuracy-vs-security trade-off, as depicted in Fig.4.

VI. Conclusions & Future Works

We present practical techniques for defending model inversion attacks using hardware security primitives in this work. As a countermeasure to the MIA attack, we find that dynamic noise can be introduced in the model parameters using approximate memory systems, which reduce the model inversion capability of an attacker. However, adding noise to a trained model has drawbacks, such as the drop in the model's accuracy rate during an attack. Hence, additional differential privacy measures must be explored in detail to minimize the impact of intentional model corruption during a model inversion attack.

References

[1] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Springer series in statistics New York, 2001, vol. 1, no. 10.

[2] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015, pp. 1322–1333.

[3] Y. Zhang, R. Jia, H. Pei, W. Wang, B. Li, and D. Song, "The secret revealer: generative model-inversion attacks against deep neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 253–261.

[4] M. Fredrikson, E. Lantz, S. Jha, S. Lin, D. Page, and T. Ristenpart, "Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing," in *23rd {USENIX} Security Symposium ({USENIX} Security 14)*, 2014, pp. 17–32.

[5] M. T. Arafin and G. Qu, "Rram based lightweight user authentication," in *2015 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2015, pp. 139–145.

[6] M. T. Arafin, D. Anand, and G. Qu, "A low-cost gps spoofing detector design for internet of things (iot) applications," in *Proceedings of the on Great Lakes Symposium on VLSI 2017*. New York, NY, USA: Association for Computing Machinery, 2017, p. 161–166. [Online]. Available: https://doi.org/10.1145/3060403.3060455

[7] M. T. Arafin, M. Gao, and G. Qu, "Volta: Voltage over-scaling based lightweight authentication for iot applications," in *2017 22nd Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2017, pp. 336–341.

[8] M. Patel, J. S. Kim, and O. Mutlu, "The reach profiler (reaper) enabling the mitigation of dram retention failures via profiling at aggressive conditions," *ACM SIGARCH Computer Architecture News*, vol. 45, no. 2, pp. 255–268, 2017.

[9] A. Raha, S. Sutar, H. Jayakumar, and V. Raghunathan, "Quality configurable approximate dram," *IEEE Transactions on Computers*, vol. 66, no. 7, pp. 1172–1187, 2016.

[10] S. Koppula, L. Orosa, A. G. Yağlıkçı, R. Azizi, T. Shahroodi, K. Kanellopoulos, and O. Mutlu, "Eden: enabling energy-efficient, high-performance deep neural network inference using approximate dram," in *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*, 2019, pp. 166–181.

[11] J. S. Kim, M. Patel, H. Hassan, and O. Mutlu, "The dram latency puf: Quickly evaluating physical unclonable functions by exploiting the latency-reliability tradeoff in modern commodity dram devices," in *2018 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. IEEE, 2018, pp. 194–207.

[12] K. K. Chang, A. G. Yağlıkçı, S. Ghose, A. Agrawal, N. Chatterjee, A. Kashyap, D. Lee, M. O'Connor, H. Hassan, and O. Mutlu, "Understanding reduced-voltage operation in modern dram devices: Experimental characterization, analysis, and mechanisms," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 1, no. 1, pp. 1–42, 2017.

[13] C. Ounkomol, S. Seshamani, M. M. Maleckar, F. Collman, and G. R. Johnson, "Label-free prediction of three-dimensional fluorescence images from transmitted-light microscopy," *Nature methods*, vol. 15, no. 11, pp. 917–920, 2018.