Side-Channel Fuzzy Analysis-Based AI Model Extraction Attack With Information-Theoretic Perspective in Intelligent IoT

Qianqian Pan, Jun Wu, *Member, IEEE*, Ali Kashif Bashir, *Senior Member, IEEE*, Jianhua Li, and Jie Wu, *Fellow, IEEE*

***Abstract*—Accessibility to smart devices provides opportunities for side-channel attacks (SCAs) on artificial intelligent (AI) models in the intelligent Internet of Things (IoT). However, the existing literature exposes some shortcomings: 1) incapability of quantifying and analyzing the leaked information through side channels of the intelligent IoT and 2) inability to devise efficient and accurate SCA algorithms. To address these challenges, we propose a side-channel fuzzy analysis-empowered AI model extraction attack in the intelligent IoT. First, the integrated AI model extraction framework is proposed, including power trace-based structure, execution time-based metaparameters, and hierarchical weight extractions. Then, we develop the information theory-based analysis for the AI model extraction via SCA. We derive a mutual information-enabled quantification method, theoretical lower/upper bounds of information leakage, and the minimum number of attack queries to obtain accurate weights. Furthermore, a fuzzy gray correlation-based multiple-microspace parallel SCA algorithm is proposed to extract model weights in the intelligent IoT. Based on the established information-theoretic analysis model, the proposed fuzzy gray correlation-based SCA algorithm obtains high-precision AI weights. Experimental results, consisting of simulation and real-world experiments, verify that the developed analysis method with the information-theoretic perspective is feasible and demonstrate that the designed fuzzy gray correlation-based SCA algorithm is effective for AI model extraction.**

***Index Terms*—Fuzzy analysis, information theory, intelligent Internet of Things (IoT), model extraction, side-channel attacks (SCAs).**

Manuscript received 1 November 2021; revised 28 January 2022 and 3 April 2022; accepted 5 April 2022. Date of publication 11 May 2022; date of current version 1 November 2022. This work was supported in part by the National Natural Science Foundation of China under Grant U21B2019, Grant U20B2048, and Grant 61972255, in part by the Program of Shanghai Academic Young Research Leader under Grant 20XD1422000, and in part by the National Social Science Foundation Major Project under Grant 20&ZD140. *(Corresponding authors: Jun Wu; Ali Kashif Bashir.)*

Qianqian Pan, Jun Wu, and Jianhua Li are with the Shanghai Key Laboratory of Integrated Administration Technologies for Information Security, the Collaborative Innovation Center of Shanghai Industrial Internet, and the School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: panqianqian@sjtu.edu.cn; jun.wu@ieee.org; lijh888@sjtu.edu.cn).

Ali Kashif Bashir is with the Department of Computing and Mathematics, Manchester Metropolitan University, M15 6BH Manchester, U.K., with the School of Electrical Engineering and Computer Science, National University of Science and Technology at Islamabad, Islamabad 24090, Pakistan, and also with the School of Engineering, University of Guelph, Guelph, ON N1G 2W1, Canada (e-mail: dr.alikashif.b@ieee.org).

Jie Wu is with the Center for Networked Computing, Temple University, Philadelphia, PA 19122 USA (e-mail: jiewu@temple.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TFUZZ.2022.3172991.

Digital Object Identifier 10.1109/TFUZZ.2022.3172991

I. Introduction

Recently, artificial intelligence (AI) has wide applications in the intelligent Internet of Things (IoT), e.g., identity recognition, data processing, and intrusion detection [1], [2]. The security of the intelligent IoT has received attention from industry and academia [3], [4]. As the key component of the intelligent IoT, AI models play critical roles in dynamic decisions, resource optimization, etc. [5]. Moreover, with the increasing computing resource and huge available data, the structure of AI models in the IoT becomes more complex, and the resource consumption for developing AI increases dramatically [6]. AI models are considered as important assets or intellectual properties of their owners. The security of AI models is an important issue in the intelligent IoT security field [7], [8]. The leakage of AI models in the intelligent IoT not only causes huge economic losses to their owners, but also facilitates evasion, adversarial, and model inversion attacks [9], [10].

In the intelligent IoT, AI models are deployed in numerous smart-X devices, such as automotive and virtual reality/augmented reality devices [11], [12]. AI models are usually loaded onto devices or directly implemented in hardware. The accessibility of physical control and manipulations over these devices provides opportunities for side-channel attacks (SCAs) to steal sensitive information, such as model structures, hyperparameters, or even precise weights [13]. Moreover, the rise of federal learning pushes AI computation to the edge and the end [14], [15], thereby facilitating the SCA. Although some existing works have investigated SCAs on AI models, there are still some deficiencies in this field.

1) Incapability of quantifying and analyzing the SCA on AI models. For SCAs on AI models in the intelligent IoT, this incapability makes it impossible to depict the amount of leaked sensitive information extracted from side-channel signals. Existing research neglects to systematically quantify and describe the harmfulness of SCAs on AI models. Thus, it is worthy to investigate the expression of the information leakage of SCAs theoretically.
2) Inability to design the time and computational resource-efficient SCA algorithms for the AI model extraction. Without the knowledge of the theoretical analysis and estimations, it is difficult to devise the efficient and accurate SCA algorithms. Therefore, it is urgently needed to establish the analysis methods and design efficient algorithms for SCAs on AI models.

1063-6706 © 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.

To address the above problems, information theory is adopted to establish the analysis method, and a fuzzy gray correlation-based effective extraction algorithm is designed according to the analysis. As the great communication theory, information theory mainly studies information quantification, data transmission and compression, etc. [16]. The SCAs on AI models are regarded as the special signal transmission from the running intelligent device to the adversary. Thus, it is feasible and suitable for developing the information theory-based theoretical analysis. Fuzzy theory is a powerful mathematical tool to reveal the laws of fuzzy phenomena. Meanwhile, gray theory studies and processes complex systems from the incompleteness of information [17], [18]. These features make the integration of fuzzy and gray theories promising for the design of AI model extraction. Some works have investigated the information theory analysis on classic cipher SCA and studied the fuzzy gray theory-based correlation evaluation in multiple applications [18]–[20]. However, AI models are complex computation systems with specific properties: 1) complex structure with intricate connections among neurons of multiple layers; 2) a large number of AI model parameters, including hyperparameters and weights; and 3) high precision needed for AI parameters (i.e., 32/64-bit floating-point weights) [21]. The above properties make SCAs on AI models different from traditional side-channel analysis and attacks. The main challenge for the SCA on AI models is that existing analysis methods cannot be adopted directly to describe attacks accurately and design efficient extraction algorithms.

All of the existing problems and the challenges in SCA on AI models motivate our research. Based on our preview work in [22], we investigate the theoretical analysis method and effective algorithm of SCA on AI in the intelligent IoT in this article. The main contributions of our work are listed as follows.

1) The integrated SCA-based AI model extraction framework is developed, where the power trace-based structure extraction, execution time-enabled metaparameter extraction, and hierarchical weight extraction are investigated.
2) We establish the information theory-based analysis method for SCAs on AI models, including three key points: a) the mutual information-enabled quantification method is proposed, mathematically describing the leakage amount of AI models through side channels; b) lower and upper bounds of leakage amount are derived, providing attackers/defenders with theoretical estimations of the leaked information amount; and c) the minimum number of queries for weight extraction is investigated, which can be utilized to estimate attack cost and time.
3) A fuzzy gray correlation-based multiple-microspace parallel SCA algorithm on AI weights extraction is designed, which is based on the proposed information-theoretic analysis method to extract high-precision AI weights. In this algorithm, the attack cost and success rate are estimated with mutual information, entropy, and signal-to-noise ratio (SNR), facilitating the efficiency and accuracy of SCAs on AI.
4) Simulations and experiments are conducted to verify the feasibility and effectiveness of our developed analysis method and devised SCA algorithm. It is demonstrated that our proposed analysis results are consistent with the experimental results and are helpful for the design of AI model extraction algorithms based on SCAs in the intelligent IoT.

The rest of this article is organized as follows. In Section II, we discuss the related work. The preliminary is presented in Section III. The framework of SCAs on AI model extraction is established in Section IV. The proposed information theory-based analysis method is presented in Section V. A fuzzy gray correlation-based multiple-microspace parallel SCA algorithm on AI weight extraction is implemented in Section VI. Simulations and experiments are shown in Section VII. Section VIII discusses the countermeasures of SCAs based on the proposed theoretical analysis method. Finally, Section IX concludes this article.

II. Related Work

The related work, consisting of the security and privacy of the intelligent IoT, information theory on the SCA, and the fuzzy theory on the intelligent IoT, is discussed in this section.

A. Security and Privacy of the Intelligent IoT

The security and privacy of the intelligent IoT have attracted widespread attention [23]–[26]. Specifically, Butun *et al.* [23] investigate the security of the IoT from the perspective of adversaries and defenders, which reviews the security attacks along with detection and prevention technologies. Liu *et al.* [24] focus on machine learning for IoT security, including user identification and abnormal devices detection. Li and Song [25] propose an attack-resistant management method for the Internet of Vehicles, which has capabilities to detect as well as deal with attacks. Song *et al.* [26] investigate the security of the cyber-physical systems (CPSs) and provide its foundations, principles, and applications. These works mainly study the traditional attacks and mitigation solutions on the intelligent IoT. Although the SCA is a serious threat, these works seem to ignore studying the SCA on the intelligent IoT.

The security and privacy of the smart IoT are seriously threatened by SCAs. Adversary recovers the sensitive and valuable information of intelligent models based on side-channel observations, e.g., memory access mode, power traces, running time, etc. [27]–[29]. Hua *et al.* [30] extract a convolutional neural network architecture and parameters based on the memory and timing side-channel signals. Batina *et al.* [31] investigate the reverse engineering of AI models to recover their structure, activation function, and weights based on the side-channel analysis of electromagnetic emanation. Despite a large amount of efforts that have been put in, the theoretical analysis of SCAs on AI models in the intelligent IoT still remains blank.

B. Information Theory on SCA

Shannon's information theory has been used to analyze the SCA on the cipher system [32], [33]. Mizuno *et al.* [34] design an information-theoretic evaluation method for the SCA, which models the SCA as a communication channel to estimate leaked information amount. De Cherisey *et al.* [35] propose an analysis method for the embedded hardware system under the SCA based on information theory, which derives lower and upper bounds of attack amounts. Besides, the mathematical link of the

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.

success rate and the number of attacks is discussed in this article. Santoso and Oohama [36] focus on the Shannon cipher systems and establish the corresponding information theoretical security model under SCA with high robust.

Although all of these aforementioned works study the leaked information under SCA, they mainly investigate traditional cipher systems. None of them can be used directly to describe leaked information amount of SCA on AI models or guide the design of SCA-based AI model extraction algorithms for the intelligent IoT. Besides, according to the specific characters of AI (i.e., complex structure and high-precision parameters), the existing information-theoretic analysis cannot be applied to the SCA on AI directly.

C. *Fuzzy Theory on the Intelligent IoT*

Some works have studied the fuzzy theory integrated with the intelligent IoT and the CPS. Gu *et al.* [37] investigate the resilient control issue and design a memory-based event-triggering mechanism for a fuzzy system in the CPS. Xu *et al.* [38] propose a multirobot system based on artificial immune fuzzy optimization, which is utilized to realize the formation control of robots in CPS. Mrozek *et al.* [39] adopt the fuzzy technology in the IoT to realize the combination of sensor data from asynchronous IoT devices and reduce data volume for transmission and storage.

Some works investigate the fuzzy theory-based technologies on the security and privacy of the intelligent IoT. Dong *et al.* [18] propose a safety risk assessment method based on fuzzy gray analysis, which is more sensitive to risk and more robust under different cases. Guo *et al.* [40] study the content security of the IoT and design a label smoothing-enabled fuzzy method for spammer identification. Wu *et al.* [41] propose a data carrier node selection protocol for vehicular ad hoc networks, where fuzzy logic algorithms are utilized to implement instant decision evaluation. Although these works study the fuzzy theory in the IoT and its security issues, they seem to ignore the fuzzy technology for SCAs on AI models.

III. Preliminary

A. *AI and Information-Theoretic Foundation*

AI can be formulated as a function $F : \mathcal{X}^n \rightarrow \mathcal{Y}^m$, representing the mapping from the input $\mathcal{X}^n$ with n dimensions to the output $\mathcal{Y}^m$ with m dimensions. The connection weights of the AI model with multiple layer is presented by the set $\mathbf{w} = [\boldsymbol{w}^{(1)}, \boldsymbol{w}^{(2)}, \ldots, \boldsymbol{w}^{(k)}]$, where $\boldsymbol{w}^{(i)}$ for $i \in \{1, 2, \ldots, k\}$ denotes the weight of layer i. The activation function of layer i is denoted as $f_i(\cdot)$. The AI model is formulated as the following function:

$$F(x) = f_k(\boldsymbol{w}^{(k)} \cdot (f_{k-1}(\boldsymbol{w}^{(k-1)} \cdots f_1(\boldsymbol{w}^{(1)} \cdot \boldsymbol{x})))). \quad (1)$$

In information theory, entropy is utilized to describe the uncertainty of random variables. For $X \in \mathcal{X}$ with probability distribution p_X, its entropy is expressed as

$$H(X) = -\Sigma_{x \in \mathcal{X}} p_x \log p_x. \quad (2)$$

Conditional entropy $H(X|Y)$ reflects the uncertainty of X under the given condition $Y \in \mathcal{Y}$, formulated as

$$H(X|Y) = -\Sigma_{x \in \mathcal{X}, y \in \mathcal{Y}} p_{x,y} \log \frac{p_{x,y}}{p_y} \quad (3)$$

for the probability distribution p_y and the joint probability distribution $p_{x,y}$. Mutual information is defined as the reduction of information uncertainty on X as the existence of another random variable Y, denoted as follows:

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) \\ &= \Sigma_{x \in \mathcal{X}} \Sigma_{y \in \mathcal{Y}} p_{x,y} \log \frac{p_{x,y}}{p_x p_y}. \end{aligned} \quad (4)$$

Mutual information has the following properties: 1) nonnegative, i.e., $I(X;Y) \geq 0$; and 2) symmetrical about random variables X and Y, namely, $I(X;Y) = I(Y;X)$. Besides, $I(X;Y) = 0$ if and only if $X \perp Y$.

Considering the Markov chain $X \rightarrow Y \rightarrow Z$, where the random variable Z relies on Y merely and Y relies on X merely, we have the data processing inequality $I(X;Y) \geq I(X;Z)$. This inequality demonstrates that operating on the data does not improve the amount of information gained from data.

B. *Threat Mode of the SCA on the AI System*

When the target AI system performs on a hardware platform, it inevitably results in unintentional physical leakage, e.g., energy consumption, execution time, and electromagnetic emanations released during data computation. Adversary monitors these physical leakages at runtime through preset probes and analyzes them to deduce sensitive information. AI models consist of structure, metaparameters, weights, etc. Thus, the extraction of AI models includes multiple types. With the physical control of the hardware, attackers extract the structure and parameters of AI models running on the hardware based on the multidimensional side-channel signals. Memory access patterns and timing are commonly utilized side-channel signals to infer AI model architectures [30], [42].

The side-channel attacker should first decide the attack target, attack points, and observed physical phenomena. The attack target could be something that can affect the observed physical phenomena, such as the register, memory, etc. The attack points refer to somewhere to observe a side-channel signal. For example, the output of the AI layers can be selected as attack points. Physical phenomena are the side-channel information the attackers observed. For the observed side-channel information Z, it can be represented as $Z = h(data) + noise$, where $data$ is the logical value at the attack point, $noise$ is interference caused by other factors in the hardware, and $h(\cdot)$ denotes the mapping function from logical values to measured side-channel signals.

The side-channel attackers in this article aim to extract AI models according to the observation of side-channel information. We consider that the attack scenario is the multiple-layer AI model running on hardware. The investigated SCA on AI models in this article is as generic as possible. There is no limit to the type and size of the multilayer AI models that are attacked. The target multiple-layer AI models can be equipped with any type and size of input, output, weights, and metaparameters. The adversary is malicious but passive. Side-channel attackers have

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.

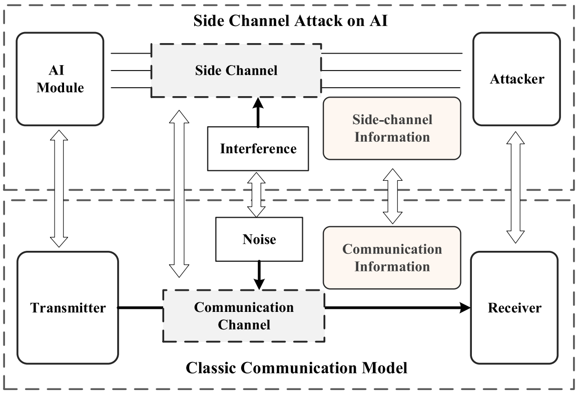


Fig. 1. Comparison of AI under SCA and the classic communication model.

physical access to the hardware deploying AI models and observe side-channel signals, without the capability to manipulate operations. Adversaries can control the execution of target AI models by crafting their inputs. Then, the adversary observes corresponding outputs and side-channel signals to obtain AI models.

C. Feasibility of Information Theory-Based Analysis

To explain the feasibility of information theory-based analysis for the SCA on AI models, we compare the SCA on the AI system with the classic communication system in Fig. 1. In the classic communication system, the information is modulated and sent by the transmitter and transmitted to the receiver through the channel. During the transmission, noise is mixed into the signal. The receiver needs to distinguish the useful information from the received signal with noise. In the SCA on the AI system, the inputs are sent into the AI module, and some calculations are executed where side-channel information is generated. The side-channel information includes a useful signal and interference caused by other factors. The attacker needs to distinguish the useful signal and estimate the AI weights by analyzing the measured side-channel signals of the attack target at the attack points.

The similarity of the SCA on the AI system and the classic communication system is also shown in Fig. 1. The AI module, side channel, attacker, interference, and side-channel information in the SCA on the AI system correspond to the transmitter, communication channel, receiver, noise, and communication information, respectively. Therefore, the SCA on the AI system can be seen as a variation of the communication system. We can analyze and model it by Shannon's information theory.

IV. Framework of the SCA on AI Systems in the Intelligent IoT

The attack framework of the SCA on AI models is presented in this section. As shown in Fig. 2, the SCA adversary crafts the input sequence and feeds them to the AI model. Then, the adversary extracts the AI structure, metaparameters, and weights according to measured side-channel signals. The output of each neural node at each layer is selected as the attack point. The SCA-based structure and metaparameter extraction, and the AI model weight recovery are presented in detail.

TABLE I
Average Time Delay of Different Activation Functions

Activation function	ReLU	sigmoid	softmax
Average time delay (μs)	38.3	1306.8	6770.8

A. SCA-Based Structure and Metaparameter Extraction

The structure and metaparameters are the basic knowledge of AI models, including the shapes and activation functions of each layer. The adversary is able to extract these information based on side-channel leakage, e.g., power traces and execution time [31]. Fig. 3 shows the power traces of a four-layer multilayer perceptron (MLP) neural network with dimensions $(5-6-4-2)$, which we capture while the MLP is performing. All the power traces captured during the preformation of MLP are presented in Fig. 3(a), where three parts can be distinguished, representing the operations of each layer. From the observation of power traces, we can easily figure out the number of layers in the AI model. Power traces of hidden layer 1 are shown in Fig. 3(b), where we can obviously distinguish six neural nodes. The power traces reveal the information of neural networks, and it is feasible to extract the AI structure through power traces analysis.

We analyze the power traces of AI models to deduce the information of the activation function. Commonly utilized activation functions are investigated, i.e., rectified linear unit (ReLU), sigmoid, and softmax. We analyze the execution time of these activation functions from captured power traces. Table I shows the average time delay of different activation functions, which are implemented on the same AI structure and randomly selected 1000 inputs. From the statistical data in Table I, different activation functions have different execution time, where ReLU is the fastest and softmax is the slowest. These phenomena result from the different computational complexities of different activation functions. The calculation of the ReLU is the simplest, thus taking the least amount of time. Sigmoid needs to calculate exponentiation and division, so it costs more than ReLU. Moreover, the softmax function has to execute multiple exponentiation and division operations, which results in long processing time. The experimental results of Fig. 3 and Table I are evaluated mainly based on the ChipWhisperer hardware platform and the server with Intel i5 4460s CPU, 8-GB RAM, and 500-GB disk. Detailed information of experimental platforms and used tools is presented in Section VII-A.

From the above analysis and results, it is feasible and practical for the adversary to recover the AI structure and metaparameters through SCA. The extraction of the AI structure and metaparameters is much easier than the extraction of the AI weights. Moreover, previous works have also done sufficient research on the AI structure and metaparameter extraction [27], [29], [31]. Therefore, we mainly focus on the study of AI weight recovery in the following of this article.

B. AI Weight Extraction Based on SCA

AI weights are key information for intelligent models. The leakage of AI weights leads to the exposure of the intelligent network and its behavior, thereby reducing the security and privacy of the system. We consider a k-layer AI model with

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.

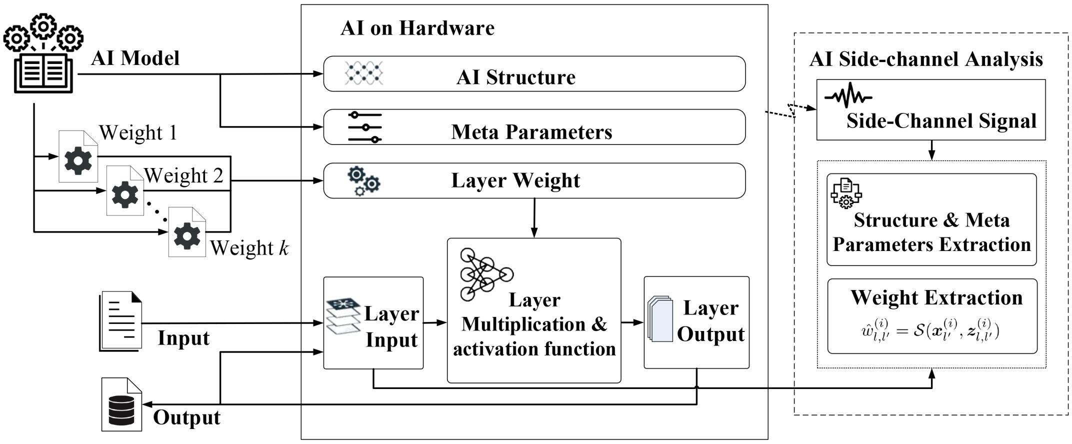
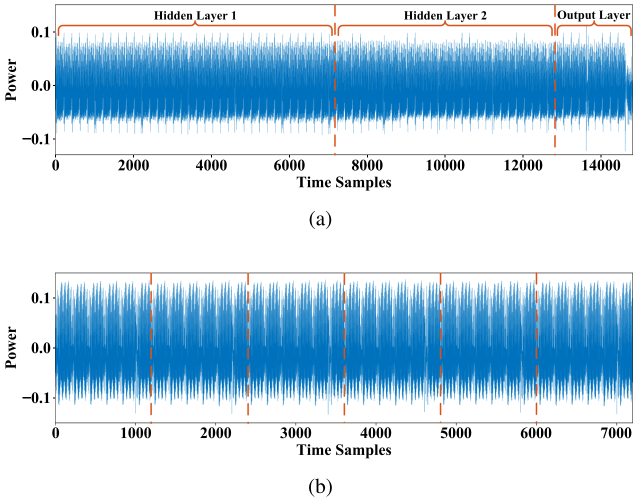


Fig. 2. Framework of SCA on AI models.



(a)

(b)

Fig. 3. Power traces of a four-layer MLP with dimensions $(5-6-4-2)$. (a) All power traces with MLP running. (b) Power traces of hidden layer 1.

structure $(L_0 - L_1 - \cdots - L_k)$, where L_0 is the input size and L_i is the ith layer size for $i \in \{1, 2, \ldots, k\}$. The target AI model has weights $\mathbf{w} = [\boldsymbol{w}^{(1)}, \boldsymbol{w}^{(2)}, \ldots, \boldsymbol{w}^{(k)}]$, where $\boldsymbol{w}^{(i)} = \{w^{(i)}_{l,l'}\}_{L_i \times L_{i-1}}$, and each weight is represented as a b-bit float number. In order to get weights of AI models, the adversary sends a series of query requests $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_q)$ to the AI, where q denotes query counts. The input $\mathbf{x}_j \in \mathcal{R}^{L_0}$ takes values from the finite input set $\mathcal{X}^{L_0}$ with probability $p_X(\mathbf{x}_j)_{\mathbf{x}_j \in \mathcal{X}^{L_0}}$ for $j \in \{1, 2, \ldots, q\}$.

For the first layer of the AI model, its input is denoted as $\mathbf{x}^{(1)} = \mathbf{x}$. The multiplication output $\boldsymbol{y}^{(1)}_{l,l'} = w^{(1)}_{l,l'} \cdot \boldsymbol{x}^{(1)}_{l'} \in \mathcal{R}^q$ is calculated for $l \in \{1, 2, \ldots, L_1\}$ and $l' \in \{1, 2, \ldots, L_0\}$, where $\boldsymbol{x}^{(1)}_{l'} = (x^{(1)}_{l',1}, x^{(1)}_{l',2}, \cdots, x^{(1)}_{l',q})$ and $\boldsymbol{y}^{(1)}_{l,l'} = (y^{(1)}_{l,l',1}, y^{(1)}_{l,l',2}, \cdots, y^{(1)}_{l,l',q})$. To extract the weight $w^{(1)}_{l,l'}$, the side-channel information $\boldsymbol{z}^{(1)}_{l,l'}$ of the output $\boldsymbol{y}^{(1)}_{l,l'}$ is observed and measured by the adversary. The observed side-channel signal is expressed as $\boldsymbol{z}^{(1)}_{l,l'} = (z^{(1)}_{l,l',1}, z^{(1)}_{l,l',2}, \cdots, z^{(1)}_{l,l',q}) = h(\boldsymbol{y}^{(1)}_{l,l'}) + \boldsymbol{n}$, where $\boldsymbol{n} \in \mathcal{N}^q$ denotes the independent identically distributed noise. Then, the mathematical function $\mathcal{S}$ is calculated by the adversary to estimate the weight $w^{(1)}_{l,l'}$ based on the input sequence and the observed signal

$$\hat{w}^{(1)}_{l,l'} = \mathcal{S}(\boldsymbol{x}^{(1)}_{l'}, \boldsymbol{z}^{(1)}_{l,l'}) = \arg\max_{\hat{w}^{(1)}_{l,l'} \in \mathcal{W}} \left(\Pr(\boldsymbol{z}^{(1)}_{l,l'} | \boldsymbol{x}^{(1)}_{l'}, \hat{w}^{(1)}_{l,l'}) \right) \quad (5)$$

where $\hat{w}^{(1)}_{l,l'}$ is the estimated value of $w^{(1)}_{l,l'}$ and $\Pr(\cdot)$ is the probability.

Based on the estimated weight $\hat{\boldsymbol{w}}^{(1)} = \{\hat{w}^{(1)}_{l,l'}\}_{L_1 \times L_0}$, the adversary extract AI model weights from the second to the kth layer iteratively. According to estimated weights of previous $(i-1)$ layers, the adversary calculates the input of layer $i \in \{2, 3, \ldots, k\}$ as $\mathbf{x}^{(i)} = (\mathbf{x}^{(i)}_1, \mathbf{x}^{(i)}_2, \ldots, \mathbf{x}^{(i)}_q)$, where

$$\mathbf{x}^{(i)}_j = f_{i-1}(\hat{\boldsymbol{w}}^{(i-1)} \cdot (f_{i-2}(\hat{\boldsymbol{w}}^{(i-2)} \cdots f_1(\hat{\boldsymbol{w}}^{(1)} \cdot \mathbf{x}_j)))) \quad (6)$$

for $j \in \{1, 2, \ldots, q\}$. Then, weight $w^{(i)}_{l,l'}$ of layer i is deduced by

$$\hat{w}^{(i)}_{l,l'} = \mathcal{S}(\boldsymbol{x}^{(i)}_{l'}, \boldsymbol{z}^{(i)}_{l,l'}) = \arg\max_{\hat{w}^{(i)}_{l,l'} \in \mathcal{W}} \left(\Pr(\boldsymbol{z}^{(i)}_{l,l'} | \boldsymbol{x}^{(i)}_{l'}, \hat{w}^{(i)}_{l,l'}) \right) \quad (7)$$

for $l \in \{1, 2, \ldots, L_i\}$ and $l' \in \{1, 2, \ldots, L_{i-1}\}$. $\hat{w}^{(i)}_{l,l'}$ is the estimated value of $w^{(i)}_{l,l'}$. $\boldsymbol{x}^{(i)}_{l'} = (x^{(i)}_{l',1}, x^{(i)}_{l',2}, \cdots, x^{(i)}_{l',q})$ and $\boldsymbol{z}^{(i)}_{l,l'} = h(\boldsymbol{y}^{(i)}_{l,l'}) + \boldsymbol{n}$ denote the input and side-channel signals, respectively.

The AI weight extraction detail under the SCA of the ith layer is presented in Fig. 4. Owing to the limited storage of the chip where the AI model runs on, the weights and outputs of the model are stored in the off-chip memory. When calculating the ith layer, the processor accessed the weight $\boldsymbol{w}^{(i)}$ and the input $\mathbf{x}^{(i)}$, and write the calculated output $\mathbf{x}^{(i+1)} = (f(\boldsymbol{w}^{(i)}\mathbf{x}^{(i)}_1), f(\boldsymbol{w}^{(i)}\mathbf{x}^{(i)}_2), \ldots, f(\boldsymbol{w}^{(i)}\mathbf{x}^{(i)}_q))$ to the DRAM. The attacker observes the leaked side-channel information $\mathbf{z}^{(i)} = \{\boldsymbol{z}^{(i)}_{l,l'}\}_{L_i \times L_{i-1}}$. We assume that the noise in the side-channel information measurements is additive white Gaussian noise (AWGN) with variance σ^2, which is commonly used in the power analysis of SCA [35]. The proposed SCA on the AI model extraction framework is applicable for systems in which

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.

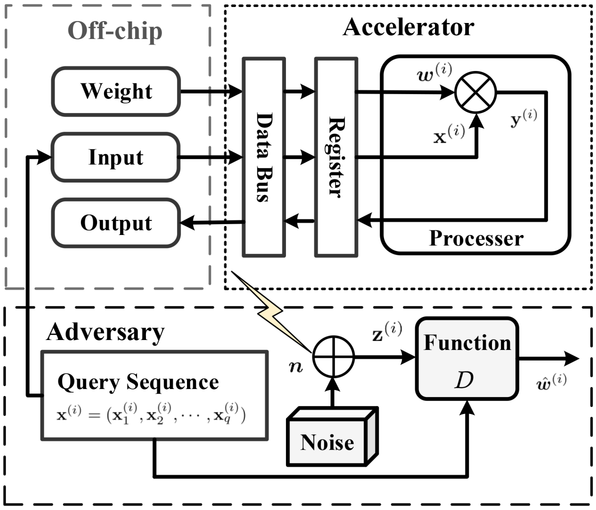


Fig. 4. SCAs on a specific layer of the AI model.

intelligent IoT devices are equipped with multiple-layer AI models. Based on the proposed framework, information-theoretic analysis and fuzzy gray correlation-based algorithm are investigated in Sections V and VI, respectively. We consider the application scenario, where the adversary has physical control over devices housing AI models in the intelligent IoT. In this scenario, AI models are loaded onto lightweight devices or directly implemented in hardware, e.g., smartphones and autonomous vehicles.

V. Information Theory-Based Analysis Method

An information leakage analysis method for the SCA on AI weight extraction is proposed in this section, including the capacity of the SCA on the AI system, the lower/upper bounds of information leakage through side channels, the minimum query counts for weight extraction, and the influence of the outputs.

A. Capacity of the SCA on the AI System

Based on the definition of the channel capacity, which is measured by the maximum information amount transmitted per second, the capacity of the SCA on the AI system is defined. However, this traditional capacity definition cannot be directly applied to the proposed SCA on the AI system, because side channels are not designed to transfer signals but steal information by feeding query samples. Therefore, we develop a method to quantify the information leakage adapting to features of the SCA on the AI system. We define the capacity of the SCA on AI models as the maximum leaked information amount per query. As mutual information means the reduction of information uncertainty on one variable caused by other random variables, the capacity of the SCA on the AI system can be written as

Definition 1: Capacity of the SCA on the AI system

$$\begin{aligned} C_i &= \max_{\boldsymbol{X}^{(i)}} I(\boldsymbol{W}^{(i)}; \boldsymbol{X}^{(i)}, \boldsymbol{Z}^{(i)}) \\ &= \sum_{l'=1}^{L_{i-1}} \sum_{l=1}^{L_i} \max_{X_{l'}^{(i)}} I(W_{l,l'}^{(i)}; X_{l'}^{(i)}, Z_{l,l'}^{(i)}) \end{aligned} \tag{8}$$

where $i \in \{1, 2, \ldots, k\}$, and $\{X_{l'}^{(i)}\}_{L_{i-1}}$, $\{W_{l,l'}^{(i)}\}_{L_i \times L_{i-1}}$, and $\{Z_{l,l'}^{(i)}\}_{L_i \times L_{i-1}}$ denote the random variables of input, weight, and the side-channel observations for the ith layer, respectively. We assume that $\{W_{l,l'}^{(i)}\}_{L_i \times L_{i-1}}$ have same distribution and are denoted as W_i.

The capacity of the SCA on AI models in the intelligent IoT system is expressed as follows.

Theorem 1 (The capacity of the SCA on AI models): The maximum information leakage of AI models through the side channel per sample is given as

$$C_i \leq \frac{L_i L_{i-1}}{2} \log \left(1 + \frac{P_{W_i}}{\sigma^2}\right) \tag{9}$$

where P_{W_i} is the variance of $h(W_i)$ and the ratio $\frac{P_{W_i}}{\sigma^2}$ is regarded as the SNR.

Proof: Based on the nature of mutual information, we have $I(W_{l,l'}^{(i)}; X_{l'}^{(i)}, Z_{l,l'}^{(i)}) = I(W_i; Z_{l,l'}^{(i)} | X_{l'}^{(i)}) + I(W_i; X_{l'}^{(i)})$. Since the value of the weights W_i is nothing to do with the input $X_{l'}^{(i)}$, we have W_i independent with $X_{l'}^{(i)}$, denoted as $W_i \perp X_{l'}^{(i)}$. That is a common assumption in SCA analysis [43], and we can derive $I(W_i; X_{l'}^{(i)}) = 0$. Therefore, the defined capacity is expressed as follows:

$$I(W_{l,l'}^{(i)}; X_{l'}^{(i)}, Z_{l,l'}^{(i)}) = H(Z_{l,l'}^{(i)} | X_{l'}^{(i)}) - H(Z_{l,l'}^{(i)} | W_i, X_{l'}^{(i)}). \tag{10}$$

For the item $H(Z_{l,l'}^{(i)} | X_{l'}^{(i)})$ in (10), we have

$$\begin{aligned} H(Z_{l,l'}^{(i)} | X_{l'}^{(i)}) &= H\left(h(W_i X_{l'}^{(i)}) + N | X_{l'}^{(i)}\right) \\ &= H\left(h(W_i) + N\right) \end{aligned} \tag{11}$$

where N is the random variable of noise. The item $H(Z_{l,l'}^{(i)} | W_i, X_{l'}^{(i)})$ in (10) can be expanded as

$$H(Z_{l,l'}^{(i)} | W_i, X_{l'}^{(i)}) = H\left(h(W_i X_{l'}^{(i)}) + N | W_i, X_{l'}^{(i)}\right). \tag{12}$$

Since $h(W_i X_{l'}^{(i)})$ can be obtained from W_i and $X_{l'}^{(i)}$, we have $H(Z_{l,l'}^{(i)} | W_i, X_{l'}^{(i)}) = H(N | W_i, X_{l'}^{(i)})$. Because the noise N is independent of W_i and $X_{l'}^{(i)}$, we obtain $H(Z_{l,l'}^{(i)} | W_i, X_{l'}^{(i)}) = H(N)$. Then, $I(W_{l,l'}^{(i)}; X_{l'}^{(i)}, Z_{l,l'}^{(i)})$ is expressed as

$$I(W_{l,l'}^{(i)}; X_{l'}^{(i)}, Z_{l,l'}^{(i)}) = H(h(W_i) + N) - H(N). \tag{13}$$

The variance of $h(W_i) + N$ is calculated as follows:

$$\text{Var}(h(W_i) + N) = \text{Var}_{W_i}(h(W_i)) + \text{Var}(N) \tag{14}$$

which is because $h(W_i)$ is independent of N. For ease of expression, we denote the variance $\text{Var}_{W_i}(h(W_i))$ as P_{W_i}. Since normal distributions maximize entropy for a given variance, we obtain

$$H(h(W_i) + N) \leq \frac{1}{2} \log 2\pi \mathrm{e}(P_{W_i} + \sigma^2). \tag{15}$$

Since the noise is assumed as AWGN, its entropy is $H(N) = \frac{1}{2} \log 2\pi \mathrm{e} \sigma^2$. Based on the entropy in (15) and $H(N)$, we rearrange $I(W_{l,l'}^{(i)}; X_{l'}^{(i)}, Z_{l,l'}^{(i)}) \leq \frac{1}{2} \log(1 + \frac{P_{W_i}}{\sigma^2})$. Therefore, we

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.

have

$$\begin{aligned} C_i &= \sum_{l'=1}^{L_{i-1}} \sum_{l=1}^{L_i} \max_{X_{l'}^{(i)}} I(W_{l,l'}^{(i)}; X_{l'}^{(i)}, Z_{l,l'}^{(i)}) \\ &\le \frac{L_i L_{i-1}}{2} \log\left(1+\frac{P_{W_i}}{\sigma^2}\right). \end{aligned} \tag{16}$$

□

B. Upper and Lower Bounds of Leaked Information Amount

Leaked information amount is one of our key concerns. To measure the information leakage, we need to define the amount of information leakage of the weights $\boldsymbol{w}^{(i)}$ at each layer from the observed side-channel information $\mathbf{z}^{(i)}$ and the input $\mathbf{x}^{(i)}$ for $i \in \{1, 2, \dots, k\}$ over q attack queries. Based on the definition of mutual information, the information leakage can be written as follows.

Definition 2: The information leakage of $\mathbf{w} = [\boldsymbol{w}^{(1)}, \boldsymbol{w}^{(2)}, \dots, \boldsymbol{w}^{(k)}]$ through SCAs over q attack queries is

$$\Delta(\mathbf{W}|\mathbf{X}, \mathbf{Z}) = \sum_{i=1}^{k} \mathbf{I}(\mathbf{W}^{(i)}; \mathbf{X}^{(i)}, \mathbf{Z}^{(i)}) \tag{17}$$

where $\mathbf{I}(\mathbf{W}^{(i)}; \mathbf{X}^{(i)}, \mathbf{Z}^{(i)})$ is the leaked information amount of layer i.

Combining the observations of multiple queries, the total amount of leaked information in layer i is

$$\begin{aligned} &\mathbf{I}(\mathbf{W}^{(i)}; \mathbf{X}^{(i)}, \mathbf{Z}^{(i)}) \\ &\quad \le \min\{q \cdot I(\boldsymbol{W}^{(i)}; \boldsymbol{X}^{(i)}, \boldsymbol{Z}^{(i)}), L_i L_{i-1} H(W_i)\} \end{aligned} \tag{18}$$

where q denotes the query counts for the SCA on AI models, and $I(\boldsymbol{W}^{(i)}; \boldsymbol{X}^{(i)}, \boldsymbol{Z}^{(i)})$ is the scalar of $\mathbf{I}(\mathbf{W}^{(i)}; \mathbf{X}^{(i)}, \mathbf{Z}^{(i)})$. Based on Theorem 1, the upper bound of leaked information amount for the SCA on the AI system is expressed as follows.

Theorem 2 (Upper bound of the leaked information amount): For the AI layer i, the upper bound of the amount of information leakage through q queries is expressed as

$$\begin{aligned} &\mathbf{I}(\mathbf{W}^{(i)}; \mathbf{X}^{(i)}, \mathbf{Z}^{(i)}) \\ &\quad \le L_i L_{i-1} \cdot \min\left\{\frac{q}{2} \log\left(1+\frac{P_{W_i}}{\sigma^2}\right), H(W_i)\right\}. \end{aligned} \tag{19}$$

Theorem 2 indicates the upper bound of the leaked information of the SCA on $\boldsymbol{w}^{(i)}$. It is positively related to the SNR and the number of queries and is bounded by $H(W_i)$. The upper bound of the leaked information amount inclines as the number of queries under the given SNR increases.

Next, we study the lower bound of leaked information amount at the ith layer of AI from the SCA as follows.

Theorem 3 (Lower bound of leaked information amount): For layer i, the lower bound of information leakage through q queries is given as

$$\begin{aligned} \mathbf{I}(\mathbf{W}^{(i)}; \mathbf{X}^{(i)}, \mathbf{Z}^{(i)}) &\ge L_i L_{i-1} \left(H(W_i) - H(E_i)\right) \\ &- L_i \sum_{l,'d} \Pr(E_i = d) \mathbf{H}(W_i | E_i = d, \hat{W}_i, \mathbf{X}_{l'}^{(i)}). \end{aligned} \tag{20}$$

E_i is the distance random variable between the weight W_i and its estimation $\hat{W}_i$.

Proof: Information leakage defined in (20) is expressed as

$$\mathbf{I}(\mathbf{W}^{(i)}; \mathbf{X}^{(i)}, \mathbf{Z}^{(i)}) = \sum_{l'=1}^{L_{i-1}} \sum_{l=1}^{L_i} \mathbf{I}(W_{l,l'}^{(i)}; \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)}). \tag{21}$$

Based on the nature of mutual information, $\mathbf{I}(W_{l,l'}^{(i)}; \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)})$ is rewritten as

$$\begin{aligned} &\mathbf{I}(W_{l,l'}^{(i)}; \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)}) = \mathbf{I}(W_i; \mathbf{X}_{l'}^{(i)}) + \mathbf{I}(W_i; \mathbf{Z}_{l,l'}^{(i)} | \mathbf{X}_{l'}^{(i)}) \\ &\overset{(a)}{=} \mathbf{I}(W_i; \mathbf{Z}_{l,l'}^{(i)} | \mathbf{X}_{l'}^{(i)}) = H(W_i) - \mathbf{H}(W_i | \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)}) \end{aligned} \tag{22}$$

where (a) is because $\mathbf{X}_{l'}^{(i)} \perp W_i$; thus, $\mathbf{I}(W_i; \mathbf{X}_{l'}^{(i)}) = 0$. The variable E_i is used to evaluate the difference between the estimated $\hat{W}_i$ and the true W_i, denoted as $E_i = e(\hat{W}_i, W_i)$, where $e(\cdot, \cdot)$ is the error function. To prove the theorem, we introduce an intermediate term $\mathbf{H}(E_i, W_i \mid \hat{W}_i, \mathbf{X}_{l'}^{(i)})$, which can be expanded as

$$\begin{aligned} &\mathbf{H}(E_i, W_i | \hat{W}_i, \mathbf{X}_{l'}^{(i)}) \\ &\quad = \mathbf{H}(W_i | \hat{W}_i, \mathbf{X}_{l'}^{(i)}) + \mathbf{H}(E_i | W_i, \hat{W}_i, \mathbf{X}_{l'}^{(i)}) \\ &\quad = \mathbf{H}(E_i | \hat{W}_i, \mathbf{X}_{l'}^{(i)}) + \mathbf{H}(W_i | E_i, \hat{W}_i, \mathbf{X}_{l'}^{(i)}). \end{aligned} \tag{23}$$

Since E_i is calculated based on W_i and $\hat{W}_i$, we achieve $\mathbf{H}(E_i \mid W_i, \hat{W}_i, \mathbf{X}_{l'}^{(i)}) = 0$. The item $\mathbf{H}(E_i \mid \hat{W}_i, \mathbf{X}_{l'}^{(i)}) \le \mathbf{H}(E_i)$ because the conditions only reduce entropy. For the last item $\mathbf{H}(W_i | E_i, \hat{W}_i, \mathbf{X}_{l'}^{(i)})$, we have

$$\begin{aligned} \mathbf{H}(W_i | E_i, \hat{W}_i, \mathbf{X}_{l'}^{(i)}) &= \sum_{d} \Pr(E_i = d) \\ &\mathbf{H}(W_i | E_i = d, \hat{W}_i, \mathbf{X}_{l'}^{(i)}). \end{aligned} \tag{24}$$

Therefore, we obtain

$$\begin{aligned} &\mathbf{H}(W_i | \hat{W}_i, \mathbf{X}_{l'}^{(i)}) \\ &= \mathbf{H}(E_i | \hat{W}_i, \mathbf{X}_{l'}^{(i)}) + \mathbf{H}(W_i | E_i, \hat{W}_i, \mathbf{X}_{l'}^{(i)}) \\ &\le H(E_i) + \sum_{d} \Pr(E_i = d) \mathbf{H}(W_i | E_i = d, \hat{W}_i, \mathbf{X}_{l'}^{(i)}). \end{aligned} \tag{25}$$

According to the details of AI weight extraction, the Markov chain $(W_i, \mathbf{X}_{l'}^{(i)}) \to (\mathbf{Y}_{l,l'}^{(i)}, \mathbf{X}_{l'}^{(i)}) \to (\mathbf{Z}_{l,l'}^{(i)}, \mathbf{X}_{l'}^{(i)}) \to (\hat{W}_i, \mathbf{X}_{l'}^{(i)})$ is derived. Thus, we adopt the data processing inequality and obtain

$$\mathbf{I}(W_i; \hat{W}_i, \mathbf{X}_{l'}^{(i)}) \le \mathbf{I}(W_i; \mathbf{Z}_{l,l'}^{(i)}, \mathbf{X}_{l'}^{(i)}). \tag{26}$$

Then, we rearrange (26) as follows:

$$\mathbf{H}(W_i | \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)}) \le \mathbf{H}(W_i | \hat{W}_i, \mathbf{X}_{l'}^{(i)}) \tag{27}$$

which is based on the mathematical relationship between mutual information and entropy. Combining (22), (25), and (27), we can obtain

$$\mathbf{I}(W_{l,l'}^{(i)}; \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)}) = H(W_i) - \mathbf{H}(W_i | \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)})$$

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.

$$\geq H(W_i)-H(E_i)-\sum_{d}\Pr(E_i=d)\mathbf{H}(W_i|E_i=d,\hat{W}_i,\mathbf{X}_{l'}^{(i)}). \tag{28}$$

Combining with (21), we have the result in Theorem 3. □

Lemma 1 (Lower bound of the information leakage for Hamming distance function): When the error function of W_i and $\hat{W}_i$ is defined as the Hamming distance, i.e., $E_i = e(W_i,\hat{W}_i) = W_i \oplus \hat{W}_i$, the lower bound is presented in (29), where $d \in \mathcal{D} = \{0,1,\ldots,b\}$ and $\sum_{d\in\mathcal{D}}\Pr(E_i=d) = 1$.

$$\begin{aligned}\mathbf{I}(\mathbf{W}^{(i)};\mathbf{X}^{(i)},\mathbf{Z}^{(i)}) &\geq L_iL_{i-1}\\ &\times\left[b+\sum_{d=0}^{b}\Pr(E_i=d)\log\Pr(E_i=d)\right.\\ &\left.-\sum_{d=0}^{b}\Pr(E_i=d)\log\binom{b}{d}\right]\end{aligned} \tag{29}$$

Proof: The entropy $H(E_i)$ can be written as

$$H(E_i) = -\sum_{d=0}^{b}\Pr(E_i=d)\log\Pr(E_i=d) \tag{30}$$

and $\mathbf{H}(W_i|E_i,\hat{W}_i,\mathbf{X}_{l'}^{(i)})$ is

$$\begin{aligned}&\mathbf{H}(W_i|E_i,\hat{W}_i,\mathbf{X}_{l'}^{(i)})\\ &= \sum_{d=0}^{b}\Pr(E_i=d)\mathbf{H}(W_i|E_i=d,\hat{W}_i,\mathbf{X}_{l'}^{(i)})\\ &\leq \sum_{d=0}^{b}\Pr(E_i=d)\log\binom{b}{d}.\end{aligned} \tag{31}$$

Thus, when E_i is the Hamming weight of W_i and $\hat{W}_i$, the lower bound of information leakage for each weight is presented in (32). Since $\log|\mathcal{W}| = b$ and the derived (21), we obtain the lower of information leakage in Lemma 1. □

$$\begin{aligned}&\mathbf{I}(W_{l,l'}^{(i)};\mathbf{X}_{l'}^{(i)},\mathbf{Z}_{l,l'}^{(i)}) \geq \log|\mathcal{W}|\\ &+\sum_{d=0}^{b}\Pr(E_i=d)\log\Pr(E_i=d)-\sum_{d=0}^{b}\Pr(E_i=d)\log\binom{b}{d}\end{aligned} \tag{32}$$

Equation (32) is nonnegative for $\Pr(E_i=d)$, for $d \in \{0,1,\ldots,b\}$ and $\sum_{d\in\mathcal{D}}\Pr(E_i=d) = 1$, and vanishes to 0 if and only if $\Pr(E_i=d) = \binom{b}{d}/2^b$. We investigate the lower bound of the leaked information amount over success rate, where the success rate is the probability of $\hat{W}_i = W_i$, i.e., $P_s = \Pr(\hat{W}_i = W_i)$. In this case, the error function E_i can be considered as binary, shown as

$$E_i = e(W_i,\hat{W}_i) = \begin{cases}0, & W_i = \hat{W}_i\\ 1, & W_i \neq \hat{W}_i\end{cases}. \tag{33}$$

We obtain the corollary as follows.

Corollary 1 (Lower bound of information leakage for success rate):

$$\mathbf{I}(\mathbf{W}^{(i)};\mathbf{X}^{(i)},\mathbf{Z}^{(i)}) \geq L_iL_{i-1}\left(b-H_2(P_s)-(1-P_s)\log(2^b-1)\right) \tag{34}$$

where $H_2(P_s)$ is the binary entropy function of P_s.

Proof: The entropies $H(E_i)$ and $\mathbf{H}(W_i|E_i,\hat{W}_i,\mathbf{X}_{l'}^{(i)})$ can be written as

$$H(E_i) = -P_s\log P_s-(1-P_s)\log(1-P_s) = H_2(P_s) \tag{35}$$

$$\begin{aligned}&\mathbf{H}(W_i|E_i,\hat{W}_i,\mathbf{X}_{l'}^{(i)})\\ &= \Pr(E_i=0)\mathbf{H}(W_i|E_i=0,\hat{W}_i,\mathbf{X}_{l'}^{(i)})\\ &\quad+\Pr(E_i=1)\mathbf{H}(W_i|E_i=1,\hat{W}_i,\mathbf{X}_{l'}^{(i)})\\ &= (1-P_s)\mathbf{H}(W_i|E_i=1,\hat{W}_i,\mathbf{X}_{l'}^{(i)})\\ &\leq (1-P_s)\log(|\mathcal{W}|-1).\end{aligned} \tag{36}$$

Thus, the information leakage is expressed as

$$\begin{aligned}&\mathbf{I}(W_{l,l'}^{(i)};\mathbf{X}_{l'}^{(i)},\mathbf{Z}_{l,l'}^{(i)})\\ &\geq \log|\mathcal{W}| - H_2(P_s) - (1-P_s)\log(|\mathcal{W}|-1)\\ &= b - H_2(P_s) - (1-P_s)\log(2^b-1).\end{aligned} \tag{37}$$

Combining (21) and (37), we have Corollary 1. □

$b-H_2(P_s)-(1-P_s)\log(2^b-1)$ is nonnegative for $P_s \in [0,1]$ and equals 0 if and only if $P_s = \frac{1}{2^b}$. When there is no trace, $\mathbf{I}(\mathbf{W}^{(i)};\mathbf{X}^{(i)},\mathbf{Z}^{(i)}) = 0$ and $P_s = \frac{1}{2^b}$. That means the adversary cannot achieve the success rate better than random guess $\frac{1}{2^b}$ without additional information, which is similar to [43, Lemma 3]. Each observed trace brings information for the AI model extraction and increases the success rate.

C. Minimum Query Counts for Weight Extraction

To obtain the link the minimum query counts for weight extraction, we combine the derived leakage bounds in Theorems 2 and 3 and obtain the inequality (38) for the ith layer.

$$\begin{aligned}&L_iL_{i-1}\left(H(W_i)-H(E_i)\right)\\ &-L_i\sum_{l,'d}\Pr(E_i=d)\mathbf{H}(W_i|E_i=d,\hat{W}_i,\mathbf{X}_{l'}^{(i)}) \leq \frac{1}{2}\cdot qL_iL_{i-1}\\ &\log\left(1+\frac{P_{W_i}}{\sigma^2}\right)\end{aligned} \tag{38}$$

Then, we arrange it as follows:

$$q \geq \frac{H(W_i)-H(E_i)-\frac{\sum_{l,'d}\Pr(E_i=d)\mathbf{H}(W_i|E_i=d,\hat{W}_i,\mathbf{X}_{l'}^{(i)})}{L_{i-1}}}{\frac{1}{2}\log\left(1+\frac{P_{W_i}}{\sigma^2}\right)}. \tag{39}$$

The minimum query counts for weight extraction is reflected by (39). Particularly, we consider the minimum query counts when the success rate P_s approaches 1, where $H(E_i)$ and $\mathbf{H}(W_i|E_i = d,\hat{W}_i,\mathbf{X}_{l'}^{(i)})$ approache 0. Thus, we obtain the minimum query

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.

count as follows:

$$\lim_{P_s \to 1} q_{\min} \geq \frac{2H(W_i)}{\log(1+\frac{P_{W_i}}{\sigma^2})} \tag{40}$$

which shows that the number of queries q needs to satisfy when the adversary wills to estimate $\boldsymbol{w}_i$ precisely. It also can be utilized to estimate attack cost and time in practice.

D. Analysis of the SCA on the AI System With the Knowledge of Outputs

In the SCA on the AI system, the adversary usually has the knowledge of the inputs as well as the corresponding outputs. In this subsection, we explore the influence of the knowledge of AI outputs on the model leakage theoretically. The information leakage Δ on $\mathbf{W}$ from the input $\mathbf{X}$, the observed output $\widetilde{\mathbf{Y}}$, and side information $\mathbf{Z}$ is formulated as

$$\Delta(\mathbf{W} \,|\, \mathbf{X}, \mathbf{Z}, \widetilde{\mathbf{Y}}) = \sum_{i=1}^{k} \mathbf{I}(\mathbf{W}^{(i)}; \mathbf{X}^{(i)}, \mathbf{Z}^{(i)}, \widetilde{\mathbf{Y}}). \tag{41}$$

The amount of leaked information in the layer $i \in \{1, 2, \ldots, k\}$ is presented as

$$\begin{aligned}
&\mathbf{I}(\mathbf{W}^{(i)}; \mathbf{X}^{(i)}, \mathbf{Z}^{(i)}, \widetilde{\mathbf{Y}}) \\
&= \sum_{l'=1}^{L_{i-1}} \sum_{l=1}^{L_i} \mathbf{I}(W_{l,l'}^{(i)}; \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)}, \widetilde{\mathbf{Y}}) \\
&= \sum_{l'=1}^{L_{i-1}} \sum_{l=1}^{L_i} \left(\mathbf{I}(W_{l,l'}^{(i)}; \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)}) + \mathbf{I}(W_{l,l'}^{(i)}; \widetilde{\mathbf{Y}} | \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)}) \right).
\end{aligned} \tag{42}$$

Thus, the difference of leaked information between with and without outputs is $\delta = \sum_{l'=1}^{L_{i-1}} \sum_{l=1}^{L_i} \mathbf{I}(W_{l,l'}^{(i)}; \widetilde{\mathbf{Y}} | \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)})$. For the difference, we obtain

$$\begin{aligned}
&\mathbf{I}(W_{l,l'}^{(i)}; \widetilde{\mathbf{Y}} | \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)}) \\
&\quad = H(W_i) - H(W_i | \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)}, \widetilde{\mathbf{Y}}) \\
&\quad \overset{(b)}{=} H(W_i) - H(W_i | \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)}, \widetilde{\mathbf{Y}}, \hat{W}_i) \\
&\quad \overset{(c)}{\geq} H(W_i) - H(W_i | \hat{W}_i)
\end{aligned} \tag{43}$$

where (b) $\hat{W}_i$ can be estimated by $\mathbf{X}_{l'}^{(i)}$, $\mathbf{Z}_l^{(i)}$, and $\widetilde{\mathbf{Y}}_l^{(i)}$, and adding the knowledge of $\hat{W}_i$ does not change the entropy. (c) Conditions can only reduce the entropy. Based on Fano's inequality [44], we have

$$H(W_i | \hat{W}_i) \leq H(P_s) + (1 - P_s) \cdot \log |\mathcal{W}|. \tag{44}$$

Then, we obtain

$$\mathbf{I}(W_{l,l'}^{(i)}; \widetilde{\mathbf{Y}} | \mathbf{X}_{l'}^{(i)}, \mathbf{Z}_{l,l'}^{(i)}) \geq H(W_i) - H(P_s) - (1 - P_s) \cdot b. \tag{45}$$

Therefore, with the knowledge of the AI model outputs, more information about the AI weights can be extracted than only with the inputs and side-channel observations. The difference of the information leakage between them is calculated as follows:

$$\delta \geq L_i L_{i-1} \left(H(W_i) - H(P_s) - (1 - P_s) \cdot b \right). \tag{46}$$

VI. Fuzzy Grey Correlation-Based Multiple-Microspace Parallel SCA Algorithm on AI

A fuzzy gray correlation analysis-based multiple-microspace parallel SCA algorithm for extracting AI weights is proposed in this section, which is based on the developed information theory-based analysis method.

A. SCA Approach for AI Weight Extraction

The inputs/outputs and weights of AI models are stored in off-chip memory since on-chip memory is constrained. The processor on the device needs to load data to the data bus when performing the AI models, where power consumption has relations to the number of "1" bits in data. The mapping function adopted in our work is Hamming weight in our work, a classic model used in the SCA.

In the target device of the SCA, the inputs/outputs and weights are usually represented by 32/64-bit floats. It is assumed that all data are 32-bit float in our work. A 32-bit floating point number includes three components (i.e., the sign, exponent, and mantissa) based on the IEEE 754 [45]. Specifically, the highest bit b_{31} represents the sign of the float number. $b_{30} \cdots b_{23}$ are the exponent basis, determining the magnitude of the float number. The lowest $b_{22} \cdots b_0$ are the mantissa bits and reflect the accuracy of the float number. A decimal float number DFN can be represented by its 32-bit float storage as

$$DFN = (-1)^{b_{31}} \times 2^{(b_{30} \cdots b_{23})_2 - 127} \times (1.b_{22} \cdots b_0)_2. \tag{47}$$

Our proposed SCA approach on a well-trained neural network is demonstrated as follows [46], [47], which can be divided into three steps.

1) *Build a query dataset based on query count prediction:* The query dataset $\mathbf{x}^{(i)}$ for $i \in \{1, 2, \ldots, k\}$ consists of multiple samples as the input to each layer of the target AI network. The size of the query dataset is predicted based on the minimum number of queries for model extraction represented in (40). The size of the query dataset is formulated as

$$q = \alpha \cdot q_{\min} = \frac{2\alpha \cdot b}{\log\left(1 + \frac{P_{W_i}}{\sigma^2}\right)} \tag{48}$$

where α is the parameter of query dataset size.

2) *Capture the power traces and form power trace matrix:* In this step, we observe side-channel information of the multiplication $\boldsymbol{y}_{l,l'}^{(i)} = w_{l,l'}^{(i)} \cdot \boldsymbol{x}_{l'}^{(i)}$ for $i \in \{1, 2, \ldots, k\}$, $l \in \{1, 2, \ldots, L_i\}$, and $l' \in \{1, 2, \ldots, L_{i-1}\}$. Meanwhile, capture power traces $\boldsymbol{V}^{\mathrm{t}} = \{V_1^{\mathrm{t}}, V_2^{\mathrm{t}}, \ldots, V_q^{\mathrm{t}}\}$ of the target AI devices while it is performing the multiplication; V_j^{t} is the jth power trace of the observation for $j \in \{1, 2, \ldots, q\}$. Each observed power trace has M trace points and $v_{j,p}^{\mathrm{t}}$ represents the pth trace point of the jth power trace.

3) *Fuzzy gray correlation-based multiple-microspace parallel side-channel analysis:* To process the captured side-channel power signal, we propose a fuzzy gray correlation analysis-based multiple-microspace parallel AI weight extraction algorithm. This proposed algorithm is guided

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.

by the information theory-based analysis in Section V and is discussed in detail in the next subsection.

B. Fuzzy Gray Correlation-Based Multiple-Microspace Parallel Side-Channel Analysis

The fuzzy gray correlation-based multiple-microspace parallel SCA algorithm is designed to extract the value of AI weights represented in a 32-bit floating model. First, the multiplications $\tilde{w}_{l,l'}^{(i)} \cdot \boldsymbol{x}_{l'}^{(i)}$ for every $\tilde{w}_{l,l'}^{(i)}$ in hypothetical weight space are calculated. Since we adopt 32-bit floating AI weights in our work, the size of the hypothetical weight space is 2^{32}. To reduce search space and extract AI weights efficiently, the multiple-microspace parallel search method is utilized. Specifically, we divide the 32-bit floating hypothetical weight space into three microspaces, i.e., sign microspace $\mathcal{W}_{\mathrm{s}}$, exponent microspace $\mathcal{W}_{\mathrm{e}}$, and mantissa microspace $\mathcal{W}_{\mathrm{m}}$. The size of microspaces is 2^1, 2^8, and 2^{23}, respectively. In practice, it is accurate enough to extract a 32-bit floating AI weight with an 8-bit mantissa. Therefore, we reduce the size of mantissa microspace to 2^8. We search the sign, exponent, and mantissa microspaces separately to determine corresponding components and obtain AI weights according to the bit calculation rules in the IEEE standard 754. This multiple-microspace parallel search algorithm greatly reduces search space, saves computational resources, and decreases time cost.

Then, we calculate hypothetical power vectors. For each microspace of the float weight, we traverse the microsearch space and obtain the corresponding components of the multiplication with samples in the query set. We convert the obtained components to the float bit pattern and calculate the Hamming weights as unique power feature vectors $V_{\tilde{w}_{\mathrm{s}}}^{\mathrm{s}} = \{v_{\tilde{w}_{\mathrm{s}},1}^{\mathrm{s}}, v_{\tilde{w}_{\mathrm{s}},2}^{\mathrm{s}}, \ldots, v_{\tilde{w}_{\mathrm{s}},q}^{\mathrm{s}}\}$ for $\tilde{w}_{\mathrm{s}} \in \mathcal{W}_{\mathrm{s}}$, $V_{\tilde{w}_{\mathrm{e}}}^{\mathrm{e}} = \{v_{\tilde{w}_{\mathrm{e}},1}^{\mathrm{e}}, v_{\tilde{w}_{\mathrm{e}},2}^{\mathrm{e}}, \ldots, v_{\tilde{w}_{\mathrm{e}},q}^{\mathrm{e}}\}$ for $\tilde{w}_{\mathrm{e}} \in \mathcal{W}_{\mathrm{e}}$, and $V_{\tilde{w}_{\mathrm{m}}}^{\mathrm{m}} = \{v_{\tilde{w}_{\mathrm{m}},1}^{\mathrm{m}}, v_{\tilde{w}_{\mathrm{m}},2}^{\mathrm{m}}, \ldots, v_{\tilde{w}_{\mathrm{m}},q}^{\mathrm{m}}\}$ for $\tilde{w}_{\mathrm{m}} \in \mathcal{W}_{\mathrm{m}}$.

After that, the values of AI weights are determined based on fuzzy gray correlation analysis. Cosine similarity is utilized to build the fuzzy correlation between the captured traces and the calculated hypothetical power vectors [18]. As a useful measurement of the similarity degree, cosine similarity reflects the correlation between variables universally and fairly, which is shown as follows:

$$r_{\tilde{w}_{\mathrm{f}},p}^{\mathrm{c,f}} = \frac{\sum_{j=1}^{q} v_{j,p}^{\mathrm{t}} v_{\tilde{w}_{\mathrm{f}},j}^{\mathrm{f}}}{\sqrt{\sum_{j=1}^{q} (v_{j,p}^{\mathrm{t}})^2} \sqrt{\sum_{j=1}^{q} (v_{\tilde{w}_{\mathrm{f}},j}^{\mathrm{f}})^2}} \tag{49}$$

for $\forall \mathrm{f} \in \{\mathrm{s}, \mathrm{e}, \mathrm{m}\}$, $\tilde{w}_{\mathrm{f}} \in \mathcal{W}_{\mathrm{f}}$, and $p \in \{1, 2, \ldots, M\}$. The gray correlation between captured traces and calculated power vectors is calculated as follows:

$$r_{\tilde{w}_{\mathrm{f}},p}^{\mathrm{g,f}} = \sum_{j=1}^{q} \frac{\min\limits_{j,p} D(v_{j,p}^{\mathrm{t}}, v_{\tilde{w}_{\mathrm{f}},j}^{\mathrm{f}}) + \rho \cdot \max\limits_{j,p} D(v_{j,p}^{\mathrm{t}}, v_{\tilde{w}_{\mathrm{f}},j}^{\mathrm{f}})}{D(v_{j,p}^{\mathrm{t}}, v_{\tilde{w}_{\mathrm{f}},j}^{\mathrm{f}}) + \max\limits_{j,p} D(v_{j,p}^{\mathrm{t}}, v_{\tilde{w}_{\mathrm{f}},j}^{\mathrm{f}})} \tag{50}$$

for $\forall \mathrm{f} \in \{\mathrm{s}, \mathrm{e}, \mathrm{m}\}$, $\tilde{w}_{\mathrm{f}} \in \mathcal{W}_{\mathrm{f}}$, and $p \in \{1, 2, \ldots, M\}$. In (50), $D(\cdot, \cdot)$ is the distance function and one-order norm distance is adopted in this article and ρ is the coefficient of gray correlation. Combining the fuzzy and gray correlation in (49) and (50), the fuzzy gray correlation with the pth point is expressed as follows:

$$r_{\tilde{w}_{\mathrm{f}},p}^{\mathrm{f}} = \gamma \cdot r_{\tilde{w}_{\mathrm{f}},p}^{\mathrm{c,f}} + \frac{1-\gamma}{q} \cdot r_{\tilde{w}_{\mathrm{f}},p}^{\mathrm{g,f}} \quad \forall \mathrm{f} \in \{\mathrm{s}, \mathrm{e}, \mathrm{m}\}, \forall \tilde{w}_{\mathrm{f}} \in \mathcal{W}_{\mathrm{f}} \quad \forall p \in \{1, 2, \ldots, M\} \tag{51}$$

where $0 \leq \gamma \leq 1$ represents the combining parameter. Compared with the traditional cosine similarity, the fuzzy gray correlation combines the advantages of the fuzzy and gray correlations and has better performance in similarity measurement [18], [19]. Thus, the fuzzy gray is more suitable for the side-channel analysis for extracting AI model weights. The fuzzy gray correlation between captured traces and calculated power vectors is

$$r_{\tilde{w}_{\mathrm{f}}}^{\mathrm{f}} = \sum_{p=1}^{M} \beta_j \cdot r_{\tilde{w}_{\mathrm{f}},p}^{\mathrm{f}} \quad \forall \mathrm{f} \in \{\mathrm{s}, \mathrm{e}, \mathrm{m}\}, \forall \tilde{w}_{\mathrm{f}} \in \mathcal{W}_{\mathrm{f}} \tag{52}$$

where $\sum_{j=1}^{M} \beta_j = 1$. The correct value of the weight has a higher fuzzy gray correlation than that of others. For the sign, exponent, and mantissa components, the hypothetical values with the largest fuzzy gray correlation are selected as the optimal

Algorithm 1: Fuzzy Gray Correlation Analysis-Based Multiple-Microspace Parallel Search Algorithm for SCA on AI Models.

```
**function** EstimateWeight($\boldsymbol{x}$, f)
  **for** $\tilde{w}_{\mathrm{f}}$ in $\mathcal{W}_{\mathrm{f}}$ **do**
    Floating multiplication of $\boldsymbol{x}$ and $\tilde{w}_{\mathrm{f}}$
    ▷ hypothetical power vectors
    $V_{\tilde{w}_{\mathrm{f}}}^{\mathrm{f}} = \{v_{\tilde{w}_{\mathrm{f}},1}^{\mathrm{f}}, v_{\tilde{w}_{\mathrm{f}},2}^{\mathrm{f}}, \cdots, v_{\tilde{w}_{\mathrm{f}},q}^{\mathrm{f}}\}$
    Calculate $r_{\tilde{w}_{\mathrm{f}},p}^{\mathrm{c,f}}$ and $r_{\tilde{w}_{\mathrm{f}},p}^{\mathrm{g,f}}$ based on (49) and (50)
    $r_{\tilde{w}_{\mathrm{f}},p}^{\mathrm{f}} \leftarrow \gamma \cdot r_{\tilde{w}_{\mathrm{f}},p}^{\mathrm{c,f}} + \frac{1-\gamma}{q} \cdot r_{\tilde{w}_{\mathrm{f}},p}^{\mathrm{g,f}}$
    $r_{\tilde{w}_{\mathrm{f}}}^{\mathrm{f}} \leftarrow \sum_{p=1}^{M} \beta_j \cdot r_{\tilde{w}_{\mathrm{f}},p}^{\mathrm{f}}$
  **end for**
  $w_{\mathrm{f}}^* \leftarrow \arg\max_{\tilde{w}_{\mathrm{f}}} r_{\tilde{w}_{\mathrm{f}}}^{\mathrm{f}}$
  **return** $w_{\mathrm{f}}^*$
**end Function**
**function** Main
  $\mathbf{x}^{(1)} \leftarrow$ Select $q$ query inputs from $\mathcal{X}^{L_0}$
  **for** $i \leftarrow 1$ to $k$ **do**
    **for** $l \leftarrow 1$ to $L_i$ **do**
      **for** $l' \leftarrow 1$ to $L_{i-1}$ **do**
        ▷ Measure power consumption:
        $\boldsymbol{V}^{\mathrm{t}} = \{V_1^{\mathrm{t}}, V_2^{\mathrm{t}}, \ldots, V_q^{\mathrm{t}}\}$
        ▷ AI weight Estimation:
        $\mathcal{W}_{\mathrm{s}}, \mathcal{W}_{\mathrm{e}}, \mathcal{W}_{\mathrm{m}} \leftarrow$ Microspaces determination
        $w_{\mathrm{s}}^* \leftarrow$ EstimateWeight($\boldsymbol{x}_{l'}^{(i)}$, s)
        $w_{\mathrm{e}}^* \leftarrow$ EstimateWeight($\boldsymbol{x}_{l'}^{(i)}$, e)
        $w_{\mathrm{m}}^* \leftarrow$ EstimateWeight($\boldsymbol{x}_{l'}^{(i)}$, m)
        $w_{l,l'}^{(i)*} \leftarrow (w_{\mathrm{s}}^* << 31)||(w_{\mathrm{e}} << 23)||w_{\mathrm{m}}^*$
      **end for**
    **end for**
    $\mathbf{x}^{(i+1)} \leftarrow (\boldsymbol{x}_1^{(i+1)}, \boldsymbol{x}_2^{(i+1)}, \ldots, \boldsymbol{x}_q^{(i+1)})$ based on (6)
  **end for**
  $\hat{\mathbf{w}} \leftarrow$ Fine-tune $\mathbf{w}^*$ with $(\mathbf{x}, \tilde{\mathbf{y}})$
  **return** $\hat{\mathbf{w}}$
**end Function**
```

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.

solutions, denoted as w_s^*, w_e^*, and w_m^*, respectively:

$$w_{\rm f}^* = \arg\max\nolimits_{\tilde{w}_{\rm f}} r^{\rm f}_{\tilde{w}_{\rm f}}, \forall {\rm f} \in \{{\rm s, e, m}\},\ \tilde{w}_{\rm f} \in \mathcal{W}_{\rm f}. \tag{53}$$

Thus, the best guess weight can be expressed as $w^* = (-1)^{w_s^*} \times 2^{w_e^* - 127} \times 1.w_m^*$.

Based on the theory analysis in Section V-D, the output $\tilde{\mathbf{y}}$ of the AI network with the input $\mathbf{x}$ also leaks AI model information. After estimating all the weights of the AI models by side-channel observations, the input–output pairs are utilized as a training set to fine-tune the reconstructed the AI network. AI input–output pairs have been used to extract AI models in previous works [5]. In such attacks, the AI is viewed as a black box, and the adversary obtains AI input–output pairs via query access, which are utilized to extract equivalent AI models. The AI input–output pairs contain additional information of the model, shown in Section V-D, which is an effective complement to the SCA on the AI system. Thus, we take a fine-tuned operation to assist the SCA on AI models. Note that the AI model extraction attacks only based on query access require a huge amount of input–output pairs, resulting in higher cost and lower realizability.

The proposed SCA approach on AI weight extraction is shown in Algorithm 1 in detail. We first form the query sample set and then measure the power consumption of the query set when performing AI and calculate the power vectors of all the hypothetical weights with the microspace parallel search approach. Next, fuzzy gray correlation is calculated to obtain the guess weights, and input–output pairs are utilized to fine-tune the AI. The complexity of our designed SCA algorithm is related to the number of extracted AI weights $\mathbf{w}$, the query counts q, and the size of weight search spaces. Based on the proposed multiple-microspace parallel search approach, the computing complexity of Algorithm 1 is $O(q \cdot (|\mathcal{W}_{\rm s}| + |\mathcal{W}_{\rm e}| + |\mathcal{W}_{\rm m}|) \cdot \sum_{i=1}^{k} L_i L_{i-1})$. The designed microspace parallel AI weight analysis method divides the AI weight search space into multiple microspaces and executes the search method in parallel. Therefore, the proposed AI weight extraction algorithm decreases the cost and improves the efficiency. Moreover, the fuzzy gray correlation-based analysis integrates the advances of fuzzy and gray correlations, thereby improving the accuracy of the AI weight extraction.

VII. Experimental Evaluation

The effectiveness of the theoretical analysis method for the SCAs on the AI system and the proposed AI weight extraction algorithm is verified under a series of experiments. First, the environment settings are discussed. Next, the evaluation of the experimental results is reported.

A. Experiment Setup

The proposed theory analysis method and the AI weight extraction algorithm are evaluated on software and hardware platforms. We implement simulations and analysis experiments on a platform with Intel i5 4460s CPU, 8-GB RAM, and 500-GB disk. The operating system is Linux Ubuntu 18.04.1 LTS, and the simulations and analysis are based on Python 3.6. Hardware experiments are implemented on the ChipWhisperer Lite platform, which is designed and widely used for SCA experiments [48], [49]. As shown in Fig. 5, the hardware platform enables the execution of AI networks, the capture of power traces, and the transmission of observed side-channel signals to a PC. The AI models are deployed on the target board, and the main board captures side-channel power traces while the models are running. Then, signal processing technologies are adopted to the observed side-channel signals, including filtering, low-noise amplification, analog-to-digital conversion, parallel–serial conversion, etc. After that, the AI power signal is transmitted to the PC for further processing and analysis.

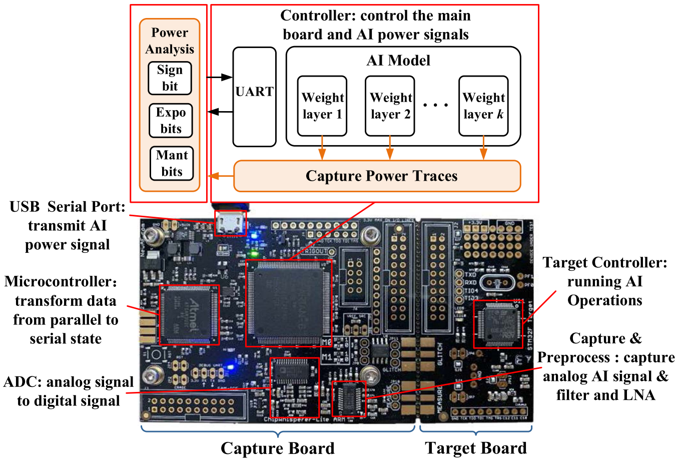

Power Analysis
Sign bit
Expo bits
Mant bits
Controller: control the main board and AI power signals
AI Model
UART
Weight layer 1
Weight layer 2
. . .
Weight layer k
Capture Power Traces
USB Serial Port: transmit AI power signal
Microcontroller: transform data from parallel to serial state
ADC: analog signal to digital signal
Target Controller: running AI Operations
Capture & Preprocess : capture analog AI signal & filter and LNA
Capture Board
Target Board

Fig. 5. Hardware experiments on the ChipWhisperer Lite development board.

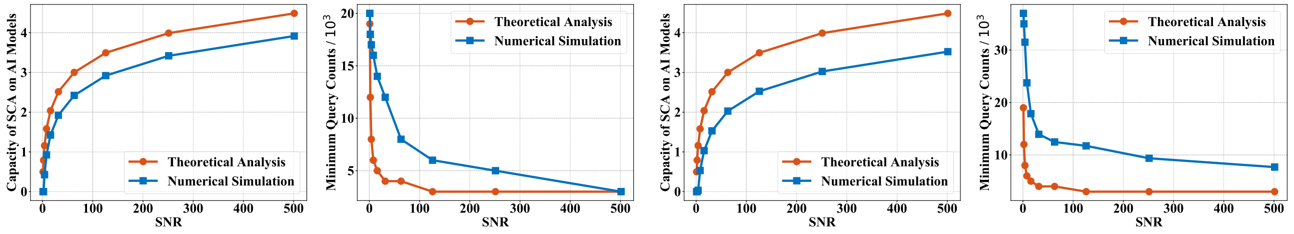
The hand-written digit dataset in *Scikit-learn* is utilized in experiments, which includes a total of 1797 samples [50]. We randomly split the dataset into two parts: a 1000-sample training set and a 797-sample testing set. In the digit dataset, the size of each image is 8 × 8 and the size of the output is 10. Logistic regression (LR) is one of the most popular binary classification algorithms and is applied in various fields (e.g., medical areas). The output of LR can be defined as $\boldsymbol{y} = f(\boldsymbol{wx})$, where the activation function $f(t) = \frac{1}{1+e^{-t}}$. Owing to the features of efficiency, simplicity, and popularity, we consider the LR as the first case to evaluate our proposed SCA framework, theoretical analysis, and extraction method. Then, we experimented with MLP, one of the most widely used AI models. The structure of the adopted MLP model is $(64 - 50 - 10)$. Activation functions of the hidden and output layers are ReLU and Softmax, respectively.

The metrics we evaluate to verify the effectiveness of the theory analysis are capacity, minimum query counts, and the normalized information leakage through the SCA on AI models. To evaluate the proposed fuzzy gray correlation-based SCA algorithm, we investigate the metrics of average absolute error (AAE), test loss, and test accuracy. The baselines applied are theoretical analysis, numerical simulation, and hardware results.

B. Evaluation Results

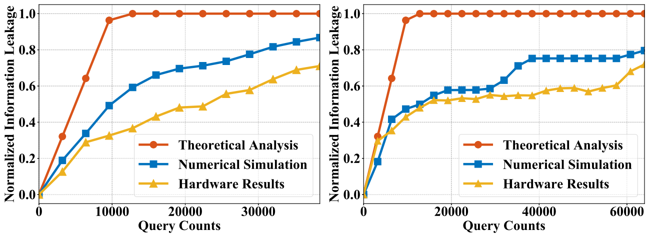
Evaluation results of the capacity for the SCA on AI weights and minimum query counts are shown in Fig. 6, which includes theoretical analysis and numerical simulation. Fig. 6(a) and (c) shows the capacities of the SCA on LR and MLP models, respectively. The red curves in Fig. 6(a) and (c) are the theoretical values of the capacity on each weight, and the blue curves are

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.



(a) (b) (c) (d)

Fig. 6. Capacity and minimum query counts of SCAs on AI over SNR. (a) Capacity of the SCA on LR. (b) Minimum query counts of the SCA on LR. (c) Capacity of the SCA on MLP. (d) Minimum query counts of the SCA on MLP.



(a) (b)

Fig. 7. Normalized information leakage of the SCA on AI models over query counts. (a) LR model. (b) MLP model.

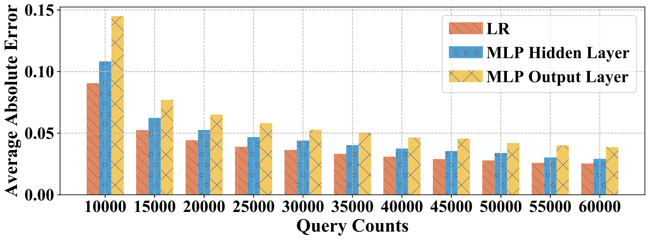


Fig. 8. AAE in weights of LR, MLP hidden layer, and MLP output layer based on the fuzzy gray correlation-enabled algorithm.

the Monte-Carlo-enabled numerical estimation. As the SNR increases, both the theoretical estimation and the Monte Carlo simulation of the SCA capacities on AI increase. From these evaluation results, the trend of the deduced theoretical capacity is consistent with that of the simulated numerical capacity. Furthermore, the theoretical capacity is higher than the numerical capacity because the derived theoretical capacity has the largest value.

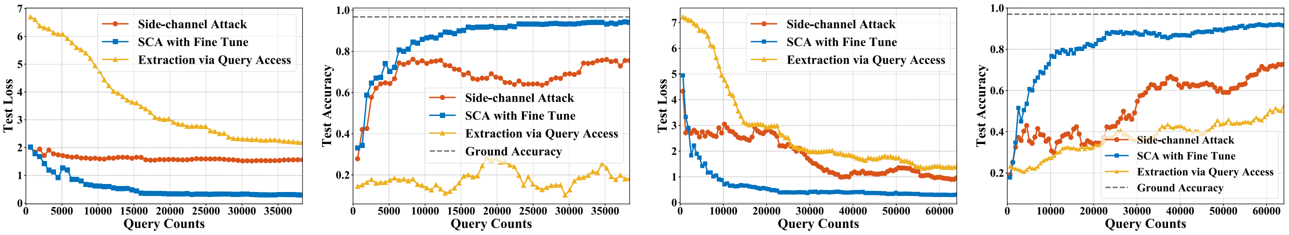
To achieve the success rate close to 100%, minimum SCA query counts on LR and MLP over different SNR are shown in Fig. 6(b) and (d), respectively. The required minimum number of queries decrease along with the increasing SNR. The theoretical value of minimum query counts drops faster than the simulation value. Both the theoretical value and the simulation value gradually converge as the SNR increases. The evaluation in Fig. 6 shows the theoretical analysis and numerical simulation results of AI weight extraction through the SCA and reveals the rationality of the proposed theorems.

The leaked information amount of AI weight extraction through the SCA is presented in Fig. 7, including evaluation results from theoretical analysis, numerical simulations, and hardware experiments. Fig. 7(a) and (b) presents the amount of information extracted by the SCA on LR and MLP models, respectively. The theoretical upper bound of the leaked information amount through side channels is linearly related to the query count, but is constrained by the entropy $H(W)$. In Fig. 7, Monte Carlo-based numerical simulation results grow more slowly than theoretical values but have better performance than the hardware experimental results shown in yellow curves. The evaluation results of theoretical analysis, numerical simulation, and hardware experiments show similar trend and convergence values, which reflect that the derived Theorem 2 is reliable.

To quantify the performance of the proposed fuzzy gray correction-based AI weight extraction algorithm through the SCA, the AAE of AI weights is investigated. AAE refers to the average of absolute errors between estimated weights and true weights, expressed as $R_{\text{AAE}}(\mathbf{w}, \hat{\mathbf{w}}) = \frac{1}{\sum_{i=1}^{k} L_i L_{i-1}} \sum_{i,l,l'} |\hat{w}_{l,l'}^{(i)} - w_{l,l'}^{(i)}|$. AAEs of LR and MLP models are shown in Fig. 8. The AAEs of the LR, MLP hidden layers, and MLP output layers gradually decrease with the increasing query counts and converge to 0.025, 0.029, and 0.038, respectively, at the 60 000th query. When going through the same number of queries, the AAE of LR is lower than that of MLP, because the model complexity of LR is lower than that of MLP. Moreover, the MLP output layer has higher AAE than that of the hidden layer for specific query counts. The reason is that weight extraction in the output layer relies on the estimation of the previous layer, introducing additional errors and degrading performance.

Other measurements formulated to quantify the performance of the proposed SCA approach on AI models are test loss and prediction accuracy. Estimated weights of AI models extracted on the hardware platform are utilized to reconstruct the AI model. Then, a test set, D, is used to investigate the loss and the prediction accuracy. In Fig. 9, the loss and the prediction accuracy of LR and MLP are presented. The loss of the reconstructed LR and MLP is shown in Fig. 9(a) and (c), respectively. For both the LR and MLP models, the test loss decreases as the number of queries increases. Specifically, the proposed SCA with fine-tuning for LR achieves the test loss of 0.28 when the number of queries is 38 400, which is only 13.38% and 18.44% of the model extraction via query access and the SCA methods, respectively. In addition, the test loss of the proposed algorithm with fine-tuning for MLP is 0.38 after 64 000 queries, which is 20.84% and 38.79% of the baselines query access and SCA methods, respectively. In Fig. 9(b) and

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.



(a) (b) (c) (d)

Fig. 9. Test loss and prediction accuracy of the estimation LR and MLP based on the fuzzy-gray-correlation-enabled algorithm in hardware platform. (a) Test loss of LR. (b) Prediction accuracy of LR. (c) Test loss of MLP. (d) Prediction accuracy of MLP.

(d), the prediction accuracy of the reconstructed LR and MLP is shown, respectively. It is obvious that the prediction accuracy of both the LR and MLP models increases as the number of queries increases. For the LR, the fine-tuning operations improve the prediction accuracy 400.33% and 25.29% higher than the query access and SCA method, respectively, when the number of queries is 38 400. Meanwhile, the prediction accuracy of the proposed SCA with the fine-tuned approach for MLP is 86.95% after 64 000 queries, which is 104.70% and 33.01% higher than query access and SCA methods. From the results in Fig. 9, the proposed fuzzy gray correlation-based SCA on the AI algorithm with fine-tuning is effective because the proposed algorithm integrates the side-channel information and query access to extract the weights of AI models.

VIII. Discussion on the AI System Side

According to the information theory-based analysis, we discuss the mitigations from the side of AI systems. The derived theory analysis provides guidelines on the efforts the adversary has to make to break the AI system. Therefore, the information theory-based analysis can be utilized to build a more robust intelligent model. To prevent such an SCA on the AI system, countermeasures that can be taken at the AI system side are presented as follows.

1) *Introducing controllable noise or interference artificially:* According to the proposed theoretic model, a lower SNR leads to less information leakage per attack query. Adding controllable noise or interference artificially is a feasible mitigation to reduce information leakage. An example of this is masking, a radical and theoretically sound side-channel countermeasure [51]. Sensitive operations are split into secret shares with random values by masking to remove dependencies of leaked data. Therefore, the SCA on the AI system can be prevented by masking every computation.

   Another way to mask leakages with additional noise is the privacy-preserving technology differential privacy (DP) [52]. With DP, each input fed to AI models is injected with artificial noise before AI operations to mitigate the SCA. However, while the information leakage of AI is reduced resulting from masking or DP mechanisms, it is inevitable to introduce additional noise and results in a performance penalty. The AI system should make a reasonable tradeoff between the security and accuracy.
2) *Limiting query counts:* In SCAs on the AI system, the adversary needs multiple attack queries, i.e., feeding the targeted AI models with crafted inputs and observing side-channel signals and outputs to infer AI models. Section V-C reveals the minimum query counts that the attacker has to take to extract the AI weights. To protect AI models against these attacks, we can limit the query counts in a period of time for each user, where the upper query limit in a specific time span is no larger than the minimum query counts derived in Section V-C.

IX. Conclusion

In this article, a side-channel fuzzy analysis-based framework was proposed for AI model extraction in the intelligent IoT. We established an analysis method with the information-theoretic perspective for SCAs on intelligent model extraction. In the method, we quantified the leaked information amount, developed its capacity and lower/upper bounds, and built a mathematical relationship between the minimum query counts and the success rate. Then, a fuzzy gray correlation-based multiple-microspace parallel algorithm was proposed for the SCA on AI weight extraction, which is based on the established information-theoretic analysis method. Moreover, experimental evaluations demonstrated the effectiveness of the proposed information theory-based analysis method and the designed fuzzy gray correction-based SCA algorithm. In the future, we will investigate more effective analysis methods and extraction algorithms for more complex AI models with multiple types of side-channel signals. Besides, we will also study the defense mechanism for the SCA on AI models and the corresponding theoretic analysis.

References

[1] S. Saha, D. Jap, S. Patranabis, D. Mukhopadhyay, S. Bhasin, and P. Dasgupta, “Automatic characterization of exploitable faults: A machine learning approach,” *IEEE Trans. Inf. Forensics Secur.*, vol. 14, no. 4, pp. 954–968, Apr. 2019.

[2] Q. Pan, J. Wu, J. Li, W. Yang, and Z. Guan, “Blockchain and AI empowered trust-information centric network for beyond 5G,” *IEEE Netw.*, vol. 34, no. 6, pp. 38–45, Nov./Dec. 2020.

[3] X. Ling *et al.*, “DEEPSEC: A uniform platform for security analysis of deep learning model,” in *Proc. IEEE Symp. Secur. Privacy*, 2019, pp. 673–690.

[4] B. Flowers, R. M. Buehrer, and W. C. Headley, “Evaluating adversarial evasion attacks in the context of wireless communications,” *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 1102–1113, 2020.

[5] F. Tramèr, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, “Stealing machine learning models via prediction APIs,” in *Proc. 25th USENIX Secur. Symp.*, 2016, pp. 601–618.

[6] G. Nguyen *et al.*, “Machine learning and deep learning frameworks and libraries for large-scale data mining: A survey,” *Artif. Intell. Rev.*, vol. 52, no. 1, pp. 77–124, 2019.

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.

[7] N. Papernot, P. McDaniel, A. Sinha, and M. P. Wellman, "SoK: Security and privacy in machine learning," in *Proc. IEEE Eur. Symp. Secur. Privacy*, 2018, pp. 399–414.

[8] S. Hidano, T. Murakami, S. Katsumata, S. Kiyomoto, and G. Hanaoka, "Model inversion attacks for prediction systems: Without knowledge of non-sensitive attributes," in *Proc. 15th Annu. Conf. Privacy Secur. Trust*, 2017, pp. 115–11509.

[9] S. Mahloujifar, D. I. Diochnos, and M. Mahmoody, "The curse of concentration in robust learning: Evasion and poisoning attacks from concentration of measure," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 4536–4543.

[10] J. Zhang and C. Li, "Adversarial examples: Opportunities and challenges," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 7, pp. 2578–2593, Jul. 2020.

[11] S. Liao, J. Wu, S. Mumtaz, J. Li, R. Morello, and M. Guizani, "Cognitive balance for fog computing resource in Internet of Things: An edge learning approach," *IEEE Trans. Mobile Comput.*, vol. 21, no. 5, pp. 1596–1608, May 2022, doi: 10.1109/TMC.2020.3026580.

[12] J. Wu, M. Dong, K. Ota, J. Li, and W. Yang, "Application-aware consensus management for software-defined intelligent blockchain in IoT," *IEEE Netw.*, vol. 34, no. 1, pp. 69–75, Jan./Feb. 2020.

[13] B. Wang and N. Z. Gong, "Stealing hyperparameters in machine learning," in *Proc. IEEE Symp. Secur. Privacy*, 2018, pp. 36–52.

[14] X. Lin, J. Wu, A. K. Bashir, J. Li, W. Yang, and J. Piran, "Blockchain-based incentive energy-knowledge trading in IoT: Joint power transfer and AI design," *IEEE Internet Things J.*, early access, Sep. 15, 2020, doi: 10.1109/JIOT.2020.3024246.

[15] J. Guo, J. Wu, A. Liu, and N. Xiong, "LightFed: An efficient and secure federated edge learning system on model splitting," *IEEE Trans. Parallel Distrib. Syst.*, early access, Nov. 15, 2021, doi: 10.1109/TPDS.2021.3127712.

[16] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, 1948.

[17] S. Feng, C. P. Chen, and C.-Y. Zhang, "A fuzzy deep model based on fuzzy restricted Boltzmann machines for high-dimensional data classification," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 7, pp. 1344–1355, Jul. 2020.

[18] G. Dong, W. Wei, X. Xia, M. Woźniak, and R. Damaševičius, "Safety risk assessment of a Pb-Zn mine based on fuzzy-grey correlation analysis," *Electronics*, vol. 9, no. 1, 2020, Art. no. 130.

[19] W. H. Chen, M. S. Tsai, and H. L. Kuo, "Distribution system restoration using the hybrid fuzzy-grey method," *IEEE Trans. Power Syst.*, vol. 20, no. 1, pp. 199–205, Feb. 2005.

[20] Y. Ni, *Fuzzy Correlation and Regression Analysis*. Norman, OK, USA: Univ. Oklahoma, 2005.

[21] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[22] Q. Pan, J. Wu, L. Xi, and L. Jianhua, "Side-channel analysis-based model extraction on intelligent CPS: An information theory perspective," in *Proc. IEEE Int. Conf. Internet of Things/IEEE Green Comput. Commun./IEEE Cyber Phys. Soc. Comput./IEEE Smart Data/IEEE Congr. Cybern.*, 2021, pp. 254–261.

[23] I. Butun, P. Österberg, and H. Song, "Security of the Internet of Things: Vulnerabilities, attacks, and countermeasures," *IEEE Commun. Surv. Tut.*, vol. 22, no. 1, pp. 616–644, Jan.–Mar. 2020.

[24] Y. Liu, J. Wang, J. Li, S. Niu, and H. Song, "Machine learning for the detection and identification of Internet of Things (IoT) devices: A survey," *IEEE Internet Things J.*, vol. 9, no. 1, pp. 298–320, Jan. 2022.

[25] W. Li and H. Song, "ART: An attack-resistant trust management scheme for securing vehicular ad hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 960–969, Apr. 2016.

[26] H. Song, G. A. Fink, and S. Jeschke, *Security and Privacy in Cyber-Physical Systems: Foundations, Principles, and Applications*. Hoboken, NJ, USA: Wiley, 2021.

[27] M. Juuti, S. Szyller, S. Marchal, and N. Asokan, "PRADA: Protecting against DNN model stealing attacks," in *Proc. IEEE Eur. Symp. Secur. Privacy*, 2019, pp. 512–527.

[28] Y. Xiang, Y. Xu, Y. Li, W. Ma, Q. Xuan, and Y. Liu, "Side-channel gray-box attack for DNNs," *IEEE Trans. Circuits Syst., II, Exp. Briefs*, vol. 68, no. 1, pp. 501–505, Jan. 2021.

[29] H. Chabanne, J.-L. Danger, L. Guiga, and U. Kühne, "Side channel attacks for architecture extraction of neural networks," *CAAI Trans. Intell. Technol.*, vol. 6, no. 1, pp. 3–16, 2021.

[30] W. Hua, Z. Zhang, and G. E. Suh, "Reverse engineering convolutional neural networks through side-channel information leaks," in *Proc. 55th ACM/ESDA/IEEE Des. Autom. Conf.*, 2018, pp. 1–6.

[31] L. Batina, S. Bhasin, D. Jap, and S. Picek, "CSI NN: Reverse engineering of neural network architectures through electromagnetic side channel," in *Proc. 28th USENIX Secur. Symp.*, 2019, pp. 515–532.

[32] F. Tramèr, D. Boneh, and K. Paterson, "Remote side-channel attacks on anonymous transactions," in *Proc. 29th USENIX Secur. Symp.*, 2020, pp. 2739–2756.

[33] M. Randolph and W. Diehl, "Power side-channel attack analysis: A review of 20 years of study for the Layman," *Cryptography*, vol. 4, no. 2, 2020, Art. no. 15.

[34] H. Mizuno, K. Iwai, H. Tanaka, and T. Kurokawa, "Analysis of side-channel attack based on information theory," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. 97, no. 7, pp. 1523–1532, 2014.

[35] E. De Cherisey, S. Guilley, O. Rioul, and P. Piantanida, "An information-theoretic model for side-channel attacks in embedded hardware," in *Proc. IEEE Int. Symp. Inf. Theory*, 2019, pp. 310–315.

[36] B. Santoso and Y. Oohama, "Information theoretic security for Shannon cipher system under side-channel attacks," *Entropy*, vol. 21, no. 5, 2019, Art. no. 469.

[37] Z. Gu, P. Shi, D. Yue, S. Yan, and X. Xie, "Memory-based continuous event-triggered control for networked T–S fuzzy systems against cyberattacks," *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 10, pp. 3118–3129, Oct. 2021.

[38] S. S. D. Xu, H. C. Huang, Y. C. Kung, and Y. Y. Chu, "A networked multirobot CPS with artificial immune fuzzy optimization for distributed formation control of embedded mobile robots," *IEEE Trans. Ind. Inform.*, vol. 16, no. 1, pp. 414–422, Jan. 2020.

[39] D. Mrozek, K. Tokarz, D. Pankowski, and B. Małysiak-Mrozek, "A hopping umbrella for fuzzy joining data streams from IoT devices in the cloud and on the edge," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 5, pp. 916–928, May 2020.

[40] Z. Guo, K. Yu, A. Jolfaei, F. Ding, and N. Zhang, "Fuz-Spam: Label smoothing-based fuzzy detection of spammers in Internet of Things," *IEEE Trans. Fuzzy Syst.*, early access, Nov. 24, 2021, doi: 10.1109/TFUZZ.2021.3130311.

[41] C. Wu, T. Yoshinaga, Y. Ji, T. Murase, and Y. Zhang, "A reinforcement learning-based data storage scheme for vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 7, pp. 6336–6348, Jul. 2017.

[42] A. Parashar *et al.*, "SCNN: An accelerator for compressed-sparse convolutional neural networks," *ACM SIGARCH Comput. Archit. News*, vol. 45, no. 2, pp. 27–40, 2017.

[43] E. de Chérisey, S. Guilley, O. Rioul, and P. Piantanida, "Best information is most successful," *IACR Trans. Cryptogr. Hardware Embedded Syst.*, vol. 2019, pp. 49–79, 2019.

[44] T. S. Han and S. Verdú, "Generalizing the Fano inequality," *IEEE Trans. Inf. Theory*, vol. 40, no. 4, pp. 1247–1251, Jul. 1994.

[45] W. Kahan, *IEEE Standard 754 for Binary Floating-Point Arithmetic*, Lecture Notes on the Status of IEEE 754, Berkeley, CA, USA, 1996.

[46] H. Guntur, J. Ishii, and A. Satoh, "Side-channel attack user reference architecture board SAKURA-G," in *Proc. IEEE 3rd Glob. Conf. Consum. Electron.*, 2014, pp. 271–274.

[47] S. Fahd, M. Afzal, H. Abbas, W. Iqbal, and S. Waheed, "Correlation power analysis of modes of encryption in AES and its countermeasures," *Future Gener. Comput. Syst.*, vol. 83, pp. 496–509, 2018.

[48] R. Benadjila, M. Renard, D. Elbaze, and P. Trébuchet, "LEIA: The lab embedded ISO7816 analyzer a custom smartcard reader for the ChipWhisperer," in *Proc. Inf. Commun. Technol. Secur. Symp.*, 2019, pp. 29–58.

[49] C. O'Flynn and Z. D. Chen, "ChipWhisperer: An open-source platform for hardware embedded security research," in *Proc. Int. Workshop Constructive Side-Channel Anal. Secure Des.*, 2014, pp. 243–260.

[50] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[51] L. Lerman and O. Markowitch, "Efficient profiled attacks on masking schemes," *IEEE Trans. Inf. Forensics Secur.*, vol. 14, no. 6, pp. 1445–1454, Jun. 2019.

[52] H. Zhao, M. Xiao, J. Wu, Y. Xu, H. Huang, and S. Zhang, "Differentially private unknown worker recruitment for mobile crowdsensing using multi-armed bandits," *IEEE Trans. Mobile Comput.*, vol. 20, no. 9, pp. 2779–2794, Sep. 2021.

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.

Qianqian Pan received the B.S. degree in information engineering and the M.S. degree in information and communication engineering from the School of Information Science and Engineering, Southeast University, Nanjing, China, in 2015 and 2018, respectively. She is currently working toward the Ph.D. degree with the School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China.

Her research interests include security and privacy of machine learning and side-channel attacks.

Jun Wu (Member, IEEE) received the Ph.D. degree in information and telecommunication studies from Waseda University, Tokyo, Japan, in 2012.

He was a Postdoctoral Researcher with the Research Institute for Secure Systems, National Institute of Advanced Industrial Science and Technology, Tokyo. He is currently a Professor with the School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China. He is the Chair of the IEEE P21451-1-5 Standard Working Group. He has hosted and participated in a lot of research projects, including National Natural Science Foundation of China, National 863 Plan and 973 Plan of China, and Japan Society of the Promotion of Science Projects. His research interests include the intelligence and security techniques of artificial intelligence, Internet of Things, 5G/6G, and molecular communication.

Dr. Wu was the Track Chair of IEEE Vehicular Technology Conference in 2019 and 2020, and the Technical Program Committee Member of more than ten international conferences, including International Conference on Communications, Global Communications Conference, etc. He is a Guest Editor for IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION, and IEEE SENSORS JOURNAL. He is an Associate Editor for IEEE SYSTEMS JOURNAL and IEEE NETWORKING LETTERS.

Ali Kashif Bashir (Senior Member, IEEE) received the Ph.D. degree in computer science and engineering from Korea University, Seoul, South Korea, in 2012.

He is currently a Reader with the Department of Computing and Mathematics, Manchester Metropolitan University, Manchester, U.K. He is also an Honorary Professor and a Chief Adviser with the Visual Research Intelligent Center, University of Electronics Science and Technology of China, Chengdu, China, an Adjunct Professor with the National University of Science and Technology, Islamabad, Pakistan, and a Special Graduate Faculty with the University of Guelph, Guelph, ON, Canada. He has authored and coauthored more than 200 research articles and received more than three Million USD funding as Principal investigator and Co-Principal Investigator from the research bodies of South Korea, Japan, the European Union, the U.K., and the Middle East. His research interests include Internet of Things, wireless networks, distributed systems, network/cyber security, network function virtualization, and machine learning.

Dr. Bashir is a Member of the IEEE Industrial Electronic Society and the Association for Computing Machinery (ACM) and a Distinguished Speaker of the ACM. He is an Editor-in-Chief for IEEE FUTURE DIRECTIONS NEWSLETTER. He is an Area Editor for *KSII Transactions on Internet and Information Systems*, and an Associate Editor for *IEEE Internet of Things Magazine*, IEEE ACCESS, *PeerJ Computer Science*, *IET Quantum Computing*, and *Journal of Plant Disease and Protection*. He is leading many conferences as a Chair (program, publicity, and track) and had organized workshops in flagship conferences such as IEEE International Conference on Computer Communications, IEEE Global Communications Conference, and IEEE International Conference on Mobile Computing and Networking.

Jianhua Li received the B.S., M.S., and Ph.D. degrees in communication and information system from Shanghai Jiao Tong University, Shanghai, China, in 1986, 1991, and 1998, respectively.

He is currently a Professor, Supervisor, and the Dean of the Institute of Cyber Science and Technology, Shanghai Jiao Tong University, Shanghai, China, where he is also the Director of the National Engineering Laboratory for Information Content Analysis Technology, the Director of the Engineering Research Center for Network Information Security Management and Service of the Chinese Ministry of Education, and the Director of the Shanghai Key Laboratory of Integrated Administration Technologies for Information Security. He was the chief expert in the information security committee experts of National High Technology Research and Development Program of China (863 Program). He was the leader of more than 30 state/province projects of China and authored or coauthored more than 300 papers and six books. He holds about 20 patents. His research interests include network security and data science.

Dr. Li received the Second Prize of National Technology Progress Award of China in 2005. He is the Vice-President of the Association of Cyber Security Association of China.

Jie Wu (Fellow, IEEE) received the Ph.D. degree in computer engineering from Florida Atlantic University, Boca Raton, in 1989. He is the Director of the Center for Networked Computing and Laura H. Carnell Professor with Temple University, Philadelphia, PA, USA, where he also serves as the Director of International Affairs with College of Science and Technology. He was the Chair of Department of Computer and Information Sciences from the summer of 2009 to the summer of 2016 and an Associate Vice Provost for International Affairs from the fall of 2015 to the summer of 2017. Prior to joining Temple University, he was a Program Director with the National Science Foundation and a Distinguished Professor with Florida Atlantic University, Boca Raton, FL, USA. He regularly publishes in scholarly journals, conference proceedings, and books. His current research interests include mobile computing and wireless networks, routing protocols, cloud and green computing, network trust and security, and social network applications.

Dr. Wu is the recipient of the 2011 China Computer Federation (CCF) Overseas Outstanding Achievement Award. He is on the Editorial Board of IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE TRANSACTIONS ON SERVICE COMPUTING, *Journal of Parallel and Distributed Computing*, and *Journal of Computer Science and Technology*. He was a General Co-Chair of 2006 IEEE International Conference on Mobile Adhoc and Sensor Systems, 2008 IEEE International Parallel and Distributed Processing Symposium, 2013 IEEE International Conference on Distributed Computing Systems, 2014 ACM International Symposium on Mobile Ad Hoc Networking and Computing, 2016 International Conference for Parallel Processing, and 2016 IEEE Conference on Communications and Network Security, and a Program Co-Chair for 2011 IEEE Conference on Computer Communications and 2013 CCF China National Computer Congress. He was a Distinguished Visitor of the IEEE Computer Society, a Distinguished Speaker of the ACM, and the Chair for the Technical Committee on Distributed Processing of the IEEE Computer Society. He is a Fellow of the American Association for the Advancement of Science.

Authorized licensed use limited to: Temple University. Downloaded on February 27,2023 at 19:37:19 UTC from IEEE Xplore. Restrictions apply.