Security Attack on Remote Sensing Equipment: PoIs Recognition Based on HW with Bi-LSTM Attention

Wei Jiang[1,2,3], Xianhua Zhang[1], Yanpeng Li[1], Chuansheng Chen[1], Jianfeng Zhu[4]*

[1] School of Geodesy and Geomatics, Jiangxi College of Applied Technology, China
[2] Jiangxi Engineering Technology Research Center of Nuclear Geoscience Data Science and System, East China University of Technology, China
[3] City University Malaysia, Malaysia
[4] College of Geological Engineering and Geomatics, Chang'an University, China
jiangwei@jxyy.edu.cn, zhangxianhua@jxyy.edu.cn, liyanpeng@jxyy.edu.cn, chenchuansheng@jxyy.edu.cn, zhujianfeng@jxyy.edu.cn

Abstract

Deep learning is an influencer in hardware security applications, which grows up to be an essential tool in hardware security, threats the confidentiality, integrity, and availability of remote sensing equipment. Comparing to traditional physical attack, not only it can greatly reduce the workload of manual selection of POIs (Points of Interests) in security attack and Trojan backdoor, but also replenishes the toolbox for attacking. On account of minute changes between network structure model and hyperparameters constantly affecting the training and attacking effect, literally, deep learning serves as a tool but not key role in hardware security attack, which means it cannot completely replace template attack and other traditional energy attack methods. In this study, we present a method using Bi-LSTM Attention mechanism to focus on the POIs related to Hamming Weight at the last round s-box output. Firstly, it can increase attacking effect and decrease guessing entropy, where attacking FPGA data demonstrates the efficiency of attacking. Secondly, it is different from the traditional template attack and deep learning attack without preprocessing subjecting to raw traces but provides attentional POIs which is the same with artificial selection. Finally, it provides a solution for attacking encrypting equipment running in parallel.

Keywords: Security attack, Remote sensing, Bi-LSTM attention

1 Introduction

The remote sensing device generally includes multiple sensor units and host controller units with inter-communication capabilities. An Unmanned Aerial Vehicle (UAV) represents a common case for a remote sensing device, since it is always equipped with multiple sensors so that data streams are diverted into another content of architectures, such as a remote control or an authority [1]. Regardless of the protected inter- communication channels, datasets collected by the sensors required to be satisfied throughout the way to the processing contents, which requires a strong and protected communication channel between numbers of sensing devices and the processing entity that provide the most essential properties: one is integrity, second is confidentiality, and the last one is authenticity. With time goes by, UAVs or satellite equipment play a vital role in transmitting remote sensing images, which includes Hyperspectral imaging (HSI), high resolution images, etc. For example, HSIs require critical and sensitive data processing to be employed onboard in order to secure bandwidth in transmission, especially in military multiple scene classification of remote sensing [2]. Meanwhile, hybrid processing systems and system-on-a-chip (SoC) platforms, combining varying scientific approaches consisting of CPU, GPU, DSPs, and field-programmable gate arrays (FPGAs), which are well-known in on-board processing [3]. The sequential portion of algorithms run on the processors, whereas intensive computations matching to parallel realization are installed in hardware accelerators (in FPGAs on the programmable logic) [4]. However, it has been proved that the capacitive crosstalk between FPGA long-wires can be a side-channel to extract sensitives [5-6], which gives adversaries the chance to perform crosstalk based on side channel attack. In addition, recent research exposed that medium-wires and multiplexers in configurable logic block are also assailable to crosstalk-based sensitive leakage. All in all, various issues on remote sensing equipment are challenging the bottom line of security.

Over the past five years, the combination of side-channel attack and deep learning has become an influencer in the field of hardware security [7-9]. Compared with traditional template attack, its main advantage is that it does not require fine preprocessing and can automatically combine features to break the mask protection. However, it also exposes the following problems in real modeling attacks:

(1) The attack always fails because of difference between the modeling device and the target device, yet imperceptible. There are few research papers that use fixed and random plaintexts for explicit comparisons during the training phase. Under the same rank, which represents the number of wrong keys that need to be tested before guessing the correct full key [10-12]. The successful attacking probability of fixed plaintext is significantly higher than that of random key. This conclusion is in line with common sense: when a fixed

*Corresponding Author: Wei Jiang; E-mail: jiangwei@jxyy.edu.cn
DOI: 10.53106/160792642023052403005

plaintext is used, the training energy trace of a key is that the data path is fixed, and the features of POIs are completely noise. The same plaintext is used in the attack, and the matching of the noise vector to the template should be reliable.

(2) When modeling energy consumption curve with mask, the feature combination ability of neural network is limited, and the model convergence is difficult. In traditional template attacks, encryption and decryption operations take a certain amount of time, and due to the high sampling rate setting, the curve with many sampling points is lengthy [13]. Through experiments, it can be seen that there are problems in the selection of the amount of feature points and the inversion of the sparse matrix in the modeling of template attack, and the location of the leak needs to be located. The traditional leak location methods include correlation analysis, t-test and standard intra-class variance.

Traditional attack: With the assistance of oscilloscope and other experimental equipment, the raw data (misaligned energy consumption traces) are preprocessed, including but not limited to selection of sensitive time interval, sampling point trial and error, SNR(Signal-Noise-Ratio) comparison, traditional filtering order selection, etc [14].

In SCA, the deep learning algorithms are generally working with raw traces. But the difference between traditional and state-of-the-art is whether preprocess before feeding into training and analysis. Typically, the attack towards the target power consumption using SCA [5] or DPA (Differential Power Analysis) depends on facilities statistical calculation to setup theoretical model for each key [15]. Among each model, the maximum likelihood probabilities subjected to target power consumption are utilized to single the true key out. Preprocessing towards training or validating power traces datasets, in addition, how to pick up suitable POIs and interval of POIs is also the eyes-drawing points in SCA [16-17].

But we always think about a question: Is there any possibility to demonstrate the machine learning method is effective, at the same time, can also verify the POIs and sensitive signal selected by machine is the same with artificial?

More and more scholars have verified the effectiveness of the convolutional neural network model against side-channel attacks. On the one hand, it does not need to align the curves deliberately, and can perform feature extraction and curve fitting more flexibly [18-19]. On the other hand, there is no strict limit on the number of feature points, and there is no need for excessive human intervention.

Regardless of whether a multi-layer perceptron or a convolutional neural network is used, although it is effective to verify the existence of a side-channel attack, the interpretability of the neural network is poor, and the adversary cannot understand the impact of each input point on the attack effect. In fact, it must be admitted that accurate localization of the curve leakage interval can reduce the dimensionality and computational complexity of attack data. Especially when the sampling rate is too large and there are too many sampling points, the dimension of the input neuron is too large. When feeding into the hidden layer, if there is no technology such as dropout, the network will be difficult to converge, and the total amount of hyperparameters will lead to overflow exceptions. In other fields of deep learning, some scholars specialize in the interpretability of neural network structures and hyperparameter selection techniques from the weight distribution [20-21]. Even now, there is still no consistent conclusion about the interpretability of neural networks.

The method proposed in our approach enriches the previous tools into analysis using natural language processing tricks.

The followings are our contributions:

On one hand, average pooling, in our approach, was used Bi-LSTM (Bi-directional LSTM) to generate an abstract traces representation. In addition, combined attention mechanism was applied to replace average pooling on the trace for better understanding.

On the other hand, DPA v2 dataset in experiment was used to verify the attacking efficiency as for encrypting-equipment running in parallel.

2 Background

2.1 Limitations on Side Channel Attack on FPGAs Chips

Generally, seral running employing AES-128 encryption algorithm on other chips is easily attacked by using of SCA, such as key leakages. Because of independent running. Figure 1 shows the main flow. On the contrary, the implementation of each S-box is established on multiple inverses of finite field and affine transformation operation, and the parallel calculation of 16 S-boxes is mixed up with each other in inter-encrypting mechanism. Harder, the reality is that dataset collecting from encryption algorithm running on FPGAs, in parallel, were fairly aligned, no need in alignment and further preprocessing. Therefore, it increases the difficulty of attack. Secondly, there are 16 S-boxes, only one which locates from near the last round of SBOX output is needed, yet the remaining 15 S-boxes are still working as noise. As for the last round of AES-128, we can use $value_{last-round} = Sbox(p_1 \oplus k_1)$, where p_1 denotes former calculation, to represent the last round output. the key k_1 is what we needed.

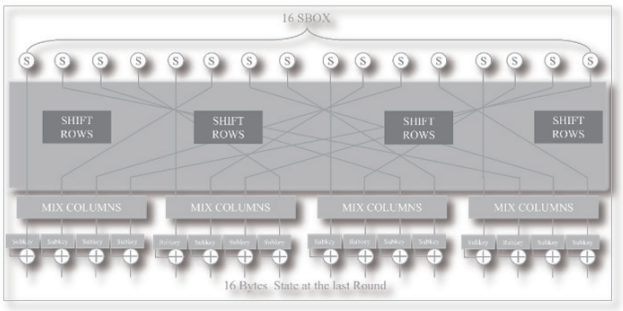

Figure 1. AES running flows

Attempts to acquire leaking information from FPGAs mostly concentrate on power analysis, although thermal and novel laser attacking has been proved to work. Power side-channel analysis utilize statistical methods to attack sensitive

subkey based on block ciphers using power consumption. And in the following Figure 2, it is easily to know the frequently used method on attacking FPGAs with physical access.

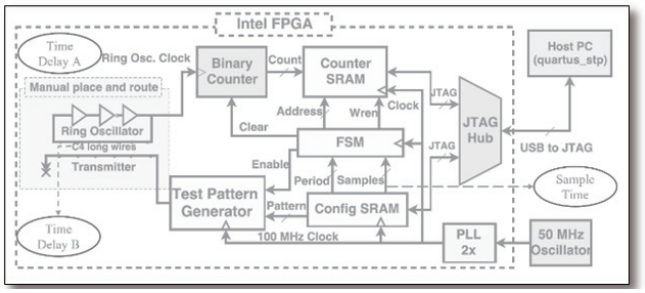

Figure 2. FPGA attacking methods and locations

Stochastic model is also well known in SCA. In virtue of such a mathematical model, a noisy model based on FPGAs can provide more leaking information for threat analysis, although the discrepancy is not apparently comparing with traditional template attacks.

Suppose that training vector subjected to main probability models incorporates sorts of noise vectors of the selected intermediate value. I_t (μ , k) denotes a recognizer function which is utilized for generated noise vector. The recognizer is used for attacking, that is to say, as for target power curve, calculating its probability based on the noise vector of each guessed key, the result of the attack is the key with the highest probability. It is harder to get all leaking key, generally, easy to steal its subkey [12].

Using N_1 power traces, figure up the subkey-dependent partition of side-channel leakages at the POIs position, also named as the bit energy conversion coefficient vector. During the complete calculation utilizing stochastic model supposes that the energy consumption at time includes two parts: the valid part of the data and the white noise:

$$I_t(\mu,k) = h_t(\mu,k) + R_t$$

Here are a few of equations, $I_t(\mu,k)$ is the power consumption function generated by the plaintext and the key k with time changing, consisting of data-related energy consumption $h_t(\mu,k)$ and noise R_t. The stochastic model further presuppose that $h_t(\mu,k)$ is a linear composition of bit energy consumption:

$$h_t(\mu,k) = \sum_{i=1}^{\gamma} \beta_i g_i(\mu,k)$$

From the following equation, $g_i(\mu,k)$ depicts the selection function, which represents the *ith* bit of intermediate value produced in the selected encryption process (for example, the *ith* bit of the S-box output). β_i expresses the bit energy conversion coefficient. This equation denotes the stochastic attack model in parallel operations. Assume this equation to establish a veracious mathematical model for attacking subkeys among parallel execution, then it changes into next equation according to practical inter-implementation.

$$h_t(\mu,k) = \sum_{s=1}^{16} \sum_{i=1}^{\gamma} \beta_i g_i(\mu,k)$$

Where the s expresses the *sth* S-box. Herein lies severe problem, parameters grow up explosively, which aggravates calculation and subkeys acquisition.

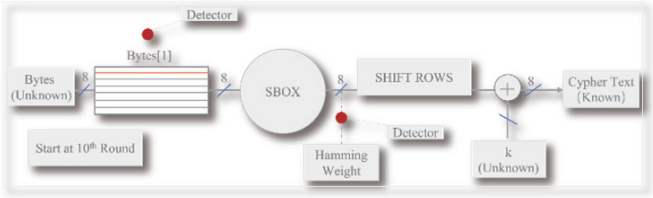

Figure 3. Main flows on attacking

The above-mentioned abominable limitations pose serious obstacles towards traditional side-channel attacks, which also activate researchers to transformer their research emphasis. Figure 3 shows the flow, which appears in recent research.

2.2 Difficulty on Explanation when Using Deep Learning

Deep learning models have created various problems in practical applications, and the lack of interpretability makes these problems difficult to solve. In the medical field, for example, seemingly accurate systems for the prognosis of patients with pneumonia rely heavily on spurious correlations in datasets. For another example, in the field of unmanned driving and image processing, neural networks with well-fitted hyperparameters may be helpless in the face of human intervention. All the above is due to our incomplete understanding of the interpretability of neural networks. Specifications for deep learning interpretability requirements have been promulgated around the world

In July 2019, China reviewed and approved the "National Science and Technology Ethics Committee Formation Plan" at the ninth meeting of the Central Comprehensive Deepening Reform Committee, and comprehensively launched the construction of science and technology ethics including explainable knowledge to ensure artificial intelligence. Safe, reliable, and controllable. The Central Committee of the Communist Party of China and the State Council attach great importance to the governance of science and technology ethics. On December 17, 2021, it was reviewed and approved at the 23rd meeting of the Central Comprehensive Deepening Reform Committee. It has made a comprehensive and systematic deployment of my country's scientific and technological ethics governance in the new era and put forward a series of major tasks and measures.

In April 2019, the European Union issued the official version of the "Artificial Intelligence Code of Ethics", which proposed a full life cycle framework for the realization of trustworthy artificial intelligence, including interpretability, security, privacy, and transparency.

Defense Advanced Research Projects Agency, DARPA

proposed the Explainable Artificial Intelligence (XAI) program in August 2016, with the aim of building a new set of explainable deep learning models. The four draft principles of the plan were published by the National Bureau of Standards in August 2020, namely, providing explanations, explanation meaning, explanation precision, and systematic knowledge boundaries.

Because researchers view the problem from various perspectives, they give abundant definitions of interpretability. At present, there is no unified definition of interpretability. Essentially, in the field of artificial intelligence, deep learning black-box interpretability is comprehended as the establishment of abundant model decision results contrapose to humans in an understandable manner, which assists us to cope with important issues such as the inner realization among complicated models and how to make specific decisions.

Lack of solubility towards all machine learning architectures earnestly hinders its adhibition in hazardous decision-making directions such as medical diagnosis, network security, financial risk control, autonomous driving, and military. Therefore, it is particularly important to study interpretability. Interpretability is helpful for the subscriber of the models to preferable comprehend the strengths and weaknesses, clarify the knowledge boundaries of the models, and understand under what circumstances the system is effective, to properly trust and use the system to make predictions. For model designers, interpretability is beneficial for optimizing and updating the system, decreasing discrimination and bias of the system, and enhancing the management and monitoring of the system.

2.3 Algorithmic Evolution in Natural Language Processing

Based on a group of common sentences, fully recognizing content details is to check out whether the theory can suitably be explained from the preconditions. Herein lies three types of correlation among them, Entailment (inferred to be true), Contradiction (inferred to be false) and Neutral (truth unknown) in Table 1. Suppose we get an instance: The panda is running through a bamboo area [22].

Table 1. Cases of three types of label subjecting to inference

Raw data	The panda is running through a bamboo area.	
Inference	The panda is walking.	Entailment
	The panda is looking for food.	Neutral
	The panda is fighting against with adversary.	Contradiction

But single directional LSTMs admit a shortage of not capitalizing on the contextual info of the later tokens. Convolutional Neural Networks (CNNs) generally ignore some sensitive information contained in sentence sequence [23-24]. Therefore, Bidirectional LSTM borne [25], it utilizes both the former and latter text content by dividing the context into dual directions which facilities to eliminate the shortages mentioned above. All in all, the fundamental architecture is on account of constructing Bi-LSTM models [26-27]. Generally, the basic mean pooling encoder could not easily formulate a sensation about what this sentence is expressing about. Acquired such representation, what we did is extending this architecture by utilizing an inner-Attention mechanism on both sides. This mechanism assists to create more high-precision and intent language expression for target classification. In this paper, four components compose the LSTM-based recurrent neural networks: firstly, an input gate i_t with related weight matrixes: W_{xi}, W_{hi}, W_{ci}, b_i; a forget gate f_t with correspondent weight matrixes: W_{xf}, W_{hf}, W_{cf}, b_f; an output gate o_t with corresponding weight matrixes: W_{xo}, W_{ho}, W_{co}, b_o. All of those gates are applied to produce some degrees by utilizing up-to-date input x_i, the state h_{t-1} which step generated former, and current state of the cell c_{t-1}, for the conclusions whether to pick up the inputs, ignore the memory reposited before, and output the state generated later [29].

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i). \quad \textbf{(1)}$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f). \quad \textbf{(2)}$$

$$g_t = \tanh(W_{xc}x_t + W_{hc}h_{t-1} + W_{cc}c_{t-1} + b_c). \quad \textbf{(3)}$$

$$c_t = i_t g_t + f_t c_{t-1}. \quad \textbf{(4)}$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o). \quad \textbf{(5)}$$

$$h_t = o_t \tanh(c_t). \quad \textbf{(6)}$$

In short, this architecture involves two sub-networks for the left and right sequence of the context, which are respectively forward and backward passed. Among them, output value of the i_{th} word is expressed in the next equation:

$$h_i = \left[\vec{h}_i \oplus \overleftarrow{h}_i\right]. \quad \textbf{(7)}$$

3 Design and Implementation

3.1 Transition with Bi-LSTM Attention

Over the last decade or so, NLP technology has drawn great interest of research institutes and industry, with regard to progress on hardware. Bi-LSTM Attention is to add Attention layer to Bi-LSTM model. In BI-LSTM, we will use the last output vector of timing sequence as the feature vector, and then conduct SoftMax classification. Attention mechanism firstly calculates the weight of each time sequence, then weights and sums all vectors of time sequence as feature vectors, and then conducts SoftMax classification [28-30]. In experiments, adding Attention did improve results. Figure 4 and Figure 5 shows the description in detail.

First, we compare each curve to a sentence in the field of natural language processing. We use the concept of space conversion to map the corresponding energy consumption value on each curve to a new space. Because the power consumption data obtained by the FPGA device is smooth and the difference is not easy to the naked eye It is noticed

that the dimension in the new space is larger, and the machine is more sensitive to the difference recognition of power consumption data than the human eye. The cost is an increase in computational dimension and an increase in space complexity.

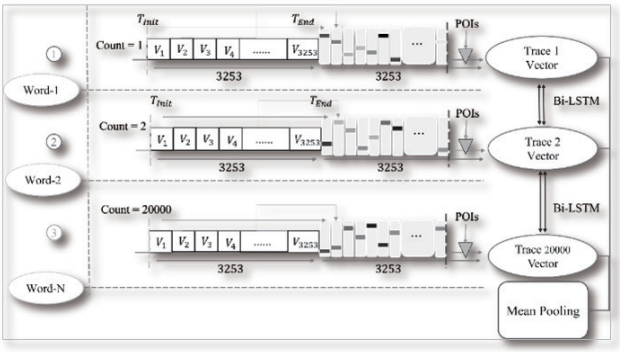

Figure 4. Transition vectors description and Bi-LSTM connection

In this paper, using Bi-LSTM Attention model to decrease the rank, then it can help us to enrich the tool attacking the targeting subkey.

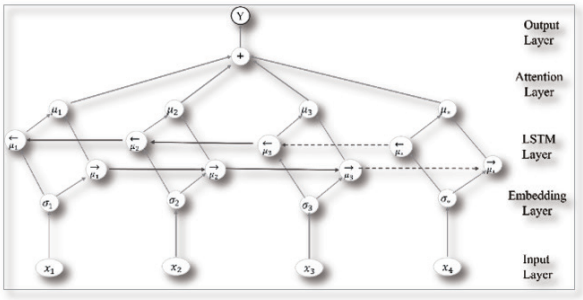

Figure 5. Bi-LSTM attention model

In side-channel attacks, the energy consumption of a cryptographic device often depends on the operations and the number of operations performed by the device, so the power consumption value of the energy consumption curve collected by the attacker is often related to the data the chip is processing. During information operation, the attacker measures and records the energy consumption of the chip, that is, encrypts the side channel information at different times t , such as the power consumption value $x(t)$, and records the corresponding plaintext or secret information k . The correlation between the key and the energy consumption is used to locate the curve leakage interval. It is not possible to use all points on the energy trace when computing the multivariate Gaussian distribution of the template. In terms of efficiency, computing a covariance matrix with a very large dimension is inefficient and takes up too much memory. In addition, when calculating the covariance matrix with an excessively high dimension, a singular matrix will be generated, which is ill-conditioned for estimating covariance matrix. In principle, the leaked data is only used in individual instructions, and the energy consumption of the rest of the location does not provide any useful information.

3.2 Experiment Design

The function of using selection function or discriminator in traditional side-channel attacks is to group energy traces according to self-defined criteria, and then establish mathematical models (templates) for each grouping. Determining the selection function is a speculative energy leakage model because it reflects the degree to which the encryption device has a perception of energy leakage of a certain information. For example, if the selector is constructed as the value of the wheel key, it means that we believe that the information of the subkey is directly leaked during the encryption process; if the Hamming weight of the SBOX output value is selected, it means that we think the leaked information is the Hamming weight of the SBOX output value. weight.

An encryption device may have multiple kinds of information leaked at the same time (such as the construction of the round key, the XOR of the round key with the plaintext, the storage of the SBOX output, and even the loading key itself), but the degree of disclosure varies. If you choose the largest energy leakage model in the attack, its attack effect is bound to be better. Some of the existing template attack implementations model the subkeys of the round key, and more model the Hamming weight of the SBOX output. One of the reasons is that the output storage of SBOX consumes the most energy in the external memory, which is considered to produce the largest information leakage. Another reason is that the nonlinear transformation of SBOX makes the energy consumption of similar keys more discriminative (when using multiple energy trace attacks, the key can be determined with Hamming weight). But different algorithms and different hardware or software encryption implementations may have different optimal leakage energy models. In template attacks (also included in DPA, CPA) attacks, choosing the correct leakage model is obviously very important to the success rate of the attack.

Template the subkey in a classic template attack. This assumes that the attacker has a device that is identical to the one being attacked and has full control over (set keys). Therefore, the general idea is to collect the same number of training energy traces according to different subkeys. However, in the study, it was found that few papers clearly mentioned whether fixed plaintext or random plaintext was used when collecting data. Only some papers compare the use of fixed and varying plaintexts during the training phase. Under the same rank (where rank is the number of wrong keys that need to be tested before guessing the correct full key), the attack success probability of fixed plaintext is significantly higher than that of fixed key. This conclusion is in line with common sense: when a fixed plaintext is used, the training energy trace of a key is that the data path is fixed, and the features of interest points are completely noise. The same plaintext is used in the attack, and the matching of the noise vector to the template should be reliable.

In conclusion, we select two factors: Rank and Hamming Weight (HW) in last round of SBOX output to evaluate model.

We regard it as a target to check out the attacking effect. S denotes the aiming HW discrete variable of SCA, and s

expresses calculation of this variable. $X_q = [X_1, X_2, X_3, \ldots, X_q]$ expresses a vector of variables incorporating a sequence of inputs towards the target, $x_q = [x_1, x_2, x_3, \ldots, x_q]$ denotes a component vector of above vector. UsingO_q denotes a random vector, which means the attention values produced by q traces. $O_q = [O_1, O_2, O_3, \ldots, O_q]$ denotes a realization of random vector. Each calculating content of the leaking function O_q corresponding to the input vector x_q. Assume $\Pr[s|O_q]$ as the conditional probability of a key class s given a leakage O_q, and regard the conditional entropy matrix as:

$$Z^q_{s,s^*} = -\sum_{O_q} \Pr[O_q \,|\, s] \cdot \log_2 \Pr[s^* \,|\, O_q]. \tag{8}$$

Where s and s^* denotes the correct aiming classification and a candidate out of the S possible ones, respectively. It is easy to derive Shannon's conditional entropy equation:

$$\begin{aligned} Z[S \mid O_q] = -\sum\nolimits_s \Pr[s] \sum\nolimits_{O_q} \Pr[O_q \,|\, s] \\ \cdot \log_2 \Pr[s^* \,|\, O_q] = E_s(Z^q_{s,s^*}). \end{aligned} \tag{9}$$

It generates the mutual information directly: $I(S, O_q) = Z[S] - Z[S|O_q]$. Attention that the inputs and outputs are typically and easily acquired by the side-channel adversary.

In consequence, computational entropy evaluates the physical leakages implicitly.

3.3 Evaluation Model

Applying this attacking experiment, DPAv2 dataset should be converted h5py format for easily deploying. It includes two datasets: Rank and HW in the last round of SBOX output to evaluate attacking model. It is necessary to know the meaning of two factors. As for the Rank, it represents the number of wrong keys that need to be tested before guessing the correct full key.

In fact, despite the low HW leakage of the output values of the first round of SBOX, it does not justify the inability to attack using deep learning methods. Experiments show that even if the output value of the first round of leakage is very low, the energy consumption curve can still be used for key guessing. In contrast, the last round is only selected as the attack target to better verify whether the deep learning is interpretable.

4 Discussion

First, validation depicts that the training efficiency towards MLP and CNN architecture is same with LSTM, Bi-LSTM. Although we use dropout [0.2, 0.3] and learning rate [10e-3, 10e-2], which are settled artificially in the experimental. And dropout methods should be seriously considered when overfitting. As is apparent from Table 1 and Table 2, the last round guessing entropy will be the most significant in side-channel attack by means of CNN and CNN with RE processing. LSTM and Bi-LSTM provide a valid method to attack remote sensing equipment from Table 3, the most impressive point is that Bi-LSTM can dig out the POIs to shrink artificial intervene, as is visible in Table 4. For those different implementations, we keep a fixed attacking setting.

What's more, if the activation function were changed from SELU activation function into others, the experimental time cost was significantly decreased.

Although we experimentally found that leakages POIs can be found using the attention mechanism method, it is not always possible to find the correct leak point. Machine learnings have a unique perspective on the identification of leaks, not human-thinking leaks. Therefore, a one-to-one correspondence cannot be drawn at present.

Table 2. Guessing entropy of using MLP and CNN attack

Pre-processing	1rd	2rd	3rd	4rd	5rd	6rd	7rd	8rd	9rd	10rd
MLP	102	96.7	81.5	80.2	79.4	64.3	53.8	46.5	41.6	40.8
RE&MLP	98.7	82.4	80.3	76.2	66.0	53.1	51.8	49.9	40.2	37.2
CNN	87.5	77.1	68.2	65.4	57.7	52.0	47.0	38.7	34.4	28.2
RE&CNN	86.2	78.8	73.9	60.1	56.2	50.9	42.7	41.8	38.1	25.4

Table 3. Guessing entropy of using LSTM and Bi-LSTM attack

Pre-processing	1rd	2rd	3rd	4rd	5rd	6rd	7rd	8rd	9rd	10rd
LSTM	109	73.5	69.5	59.7	58	53.9	47.1	46.9	34.3	27.4
Bi-LSTM	98.2	82.6	71.5	56.8	55.6	49.4	37.8	35.7	32.1	27.0

Table 4. POIs Test of Using Bi-LSTM Attention and SNR (Template attack succeed)

Attack type	1st				2ed			
SNR	0.0521	**Interval**	2627	2652	0.0521	**Interval**	2627	2652
Bi-LSTM attention	**Interval**	2558	2602	2665	2701	**Interval**	2580	2584	2639

5 Conclusion

In this paper, we present a HW with Bi-LSTM Attention threat analysis in remote sensing application, not only can provide the required POIs selection. In addition, the model assists to improve the training accuracy and attack effect. Shortcoming is obvious, no matter the instability it is, or cannot provide correct POIs, and harder, its training, attacking time cost is much higher than other models. In the next step, we try to look for the direct relationship to verify the interpretability and optimize the algorithm model including computation complexity. In the end, we are also thinking about whether transformer architecture maybe be utilized for the security analysis to abolish our model, better solve POIs question.

Acknowledgment

This work is supported by The Science and Technology Research Project of Jiangxi Provincial Department of Education (Grant No. GJJ214904); The 2021 Open Fund Project of Jiangxi Nuclear Geology Data Science and System Engineering Technology Research Center (Grant No. JETRCNGDSS202107); South Jiangxi Remote Sensing Intelligent Application Technology Innovation Center of Jiangxi College of Applied Technology.

References

[1] D. Pirker, T. Fischer, H. Witschnig, R. Matischek, C. Steger, Trustful remote-sensing architectures based on hardware-security, *2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, New York, NY, USA, 2021, pp. 0256-0263.

[2] C. Giraud, An RSA implementation resistant to fault attacks and to simple power analysis, *IEEE Transactions on computers*, Vol. 55, No. 9, pp. 1116-1120, September, 2006.

[3] Y. Luo, X. Xu, Hill: A hardware isolation framework against information leakage on multi-tenant FPGA long-wires, *2019 International Conference on Field-Programmable Technology (ICFPT)*, Tianjin, China, 2019, pp. 331-334

[4] Y. Luo, S. Duan, X. Xu, FPGAPro: A defense framework against crosstalk-induced secret leakage in FPGA, *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, Vol. 27, No. 3, pp. 1-31, May, 2022.

[5] P. C. Kocher, Timing attacks on implementations of Diffie-hellman, RSA, DSS, and other systems, in: N. Koblitz, (Eds.), *Advances in Cryptology — CRYPTO '96. CRYPTO 1996. Lecture Notes in Computer Science, vol 1109*, Springer, Berlin, Heidelberg, 1996, pp. 104-113.

[6] P. Kocher, J. Jaffe, B. Jun, Differential power analysis, in: M. Wiener, M. (Eds.), *Advances in Cryptology — CRYPTO' 99. CRYPTO 1999. Lecture Notes in Computer Science, vol 1666*, Springer, Berlin, Heidelberg, 1999, pp. 388-397.

[7] S. Chari, J. R. Rao, P. Rohatgi, Template attacks, in: B. S. Kaliski, ç.K. Koç, C. Paar (Eds.), *Cryptographic Hardware and Embedded Systems - CHES 2002. CHES 2002. Lecture Notes in Computer Science, vol 2523*, Springer, Berlin, Heidelberg, 2003, pp. 13-28.

[8] W. Schindler, K. Lemke, C. Paar, A stochastic model for differential side channel cryptanalysis, in: J. R. Rao, B. Sunar, B. (Eds.), *Cryptographic Hardware and Embedded Systems – CHES 2005. CHES 2005. Lecture Notes in Computer Science, vol 3659*, Springer, Berlin, Heidelberg, 2005, pp. 30-46.

[9] T. Bartkewitz, K. Lemke-Rust, Efficient template attacks based on probabilistic multi-class support vector machines, in: S. Mangard (Eds.), *Smart Card Research and Advanced Applications. CARDIS 2012. Lecture Notes in Computer Science, vol 7771*, Springer, Berlin, Heidelberg, 2013, pp. 263-276.

[10] L. Lerman, G. Bontempi, O. Markowitch, Power analysis attack: an approach based on machine learning, *International Journal of Applied Cryptography*, Vol. 3, No. 2, pp. 97-115, June, 2014.

[11] R. Benadjila, E. Prouff, R. Strullu, E. Cagli, C. Dumas, Deep learning for side-channel analysis and introduction to ascad database, *Journal of Cryptographic Engineering*, Vol. 10, No. 2, pp. 163-188, June, 2020.

[12] Z. Liu, Z. Wang, M. Ling, Side-channel attack using word embedding and long short term memories, *Journal of Web Engineering*, Vol. 21, No. 2, pp. 285-306, 2022.

[13] X. Hu, D. Li, N. Luo, Face-assisted authentication system based on mobile traffic distribution platform for English cloud classroom mixed guidance, *2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Tirunelveli, India, 2022, pp. 404-407.

[14] E. Cagli, C. Dumas, E. Prouff, Convolutional neural networks with data augmentation against jitter-based countermeasures, *in: W. Fischer, N. Homma (Eds.), Cryptographic Hardware and Embedded Systems – CHES 2017. CHES 2017. Lecture Notes in Computer Science, vol 10529*, Springer, Cham, 2017, pp. 45-68.

[15] P. Zhou, W. Shi, J. Tian, Z. Qi, B. Li, H. Hao, B. Xu, Attention-based bidirectional long short-term memory networks for relation classification, *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: Short papers)*, Berlin, Germany, 2016, pp. 207-212.

[16] J. Xu, H. M. Heys, Using deep learning to combine static and dynamic power analyses of cryptographic circuits, International Journal of Circuit Theory and Applications, Vol. 47, No. 6, pp. 971-990, June, 2019.

[17] A. Dubey, R. Cammarota, V. Suresh, A. Aysu, Guarding machine learning hardware against physical side-channel attacks, *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, Vol. 18, No. 3, pp. 1-31, July, 2022.

[18] S. Jin, R. Bettati, Efficient side-channel attacks beyond divide-and-conquer strategy, *Computer Networks*, Vol. 198, Article No. 108409, October, 2021.

[19] Y. Gao, Y. Zhou, Side-channel attacks with multi-thread mixed leakage, *IEEE Transactions on Information Forensics and Security*, Vol. 16, pp. 770-785, 2021.
[20] B. Hettwer, S. Gehrer, T. Güneysu, Applications of machine learning techniques in side-channel attacks: A Survey, *Journal of Cryptographic Engineering*, Vol. 10, No. 2, pp. 135-162, June, 2020.
[21] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural computation*, Vol. 9, No. 8, pp. 1735-1780, November, 1997.
[22] K. Ramezanpour, P. Ampadu, W. Diehl, Scaul: Power side-channel analysis with unsupervised learning, *IEEE Transactions on Computers*, Vol. 69, No. 11, pp. 1626-1638, November, 2020.
[23] J. Wei, Y. Zhang, Z. Zhou, Z. Li, M. A. Al Faruque, Leaky DNN: Stealing deep-learning model secret with GPU context-switching side-channel, *2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, Valencia, Spain, 2020, pp. 125-137.
[24] B. Singh, T. K. Marks, M. Jones, O. Tuzel, M. Shao, A multi-stream bi-directional recurrent neural network for fine-grained action detection, *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, NV, USA, 2016, pp. 1961-1970.
[25] D. Hakkani-T¨ur, G. T ¨ur, A. Celikyilmaz, Y.-N. Chen, J. Gao, L. Deng, Y.-Y. Wang, Multi-domain joint semantic frame parsing using Bi-Directional RNN-LSTM, *Proceedings of The 17th Annual Meeting of the International Speech Communication Association (INTERSPEECH 2016)*, San Francisco, California, USA, 2016, pp. 715-719.
[26] J. Fjeldtvedt, M. Orlandić, T. A. Johansen, An efficient real-time FPGA implementation of the ccsds-123 compression standard for hyperspectral images, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, Vol. 11, No. 10, pp. 3841-3852, October, 2018.
[27] H. Zhu, J. Tian, Z. Tian, S. Zhu, H. Li, Y. Zhang, Anomaly traffic detection with verifiable interpretation in industrial networks, *2021 IEEE International Conference on Dependable, Autonomic and Secure Computing Congress (DASC)*, AB, Canada, 2021, pp. 460-467.
[28] N. Q. Tran, H. Q. Nguyen, Efficient CNN-based profiled side channel attacks, *Journal of Computer Science and Cybernetics*, Vol. 37, No. 1, pp. 1-22, March, 2021.
[29] A. Golder, D. Das, J. Danial, S. Ghosh, S. Sen, A. Raychowdhury, Practical approaches toward deep-learning-based cross-device power side-channel attack, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 27, No. 12, pp. 2720-2733, December, 2019.
[30] C. Sun, Y. Liu, C. Jia, B. Liu, L. Lin, Recognizing text entailment via Bidirectional LSTM model with inner-attention, *International Conference on Intelligent Computing*, Liverpool, UK, 2017, pp. 448-457.

Biographies

Wei Jiang, Master of Geographic Information System, Lecturer. Graduated from East China University of Technology in 2013. Worked in Jiangxi College of Applied Technology. Her research interests include remote sensing and GIS.

Xianhua Zhang, Master candidate, associate professor of surveying and mapping, senior engineer of surveying and mapping, senior technician of photogrammetry, Worked in Jiangxi College of Applied Technology. Her research interests include Map geographic information.

Yanpeng Li, Associate Professor, Ph.D. in Engineering, Senior Technician, Ph.D. in photogrammetry and remote sensing from China University of Mining and Technology, and master's degree in Cartography and GIS from Jiangxi University of Science and Technology. Mainly engaged in the research of cartography and mapping, geographic information system, quantitative remote sensing and remote sensing inversion, environmental remote sensing and ecological modeling, big data analysis.

Chuansheng Chen, Master of Geomatics Engineering, Professor. Graduated from JiangXi University of Science and Technology in 2009. Worked in Jiangxi College of Applied Technology. His research interests include geodesy and remote sensing.

Jianfeng Zhu, currently pursuing the Ph.D. degree with the College of Geological Engineering and Geomatics, Chang'an University, Lecturer. Worked in Jiangxi College of Applied Technology. His research interests include n3D point cloud processing, photogrammetry and remote sensing.