Synergizing GCN and GAT for Hardware Trojan Detection and Localization

Yu-Chen Hsiao, Chia-Heng Yen, Bo-Yang Ke and Kai-Chiang Wu
Institute of Computer Science and Engineering, National Yang Ming Chiao Tung University
Hsinchu, Taiwan

***Abstract*—Hardware Trojan (HT) is a common issue for the outsourcing model and it poses various threats to hardware security. HT may be implanted during the design phase through the use of open-source resources and uncertified tools. In this paper, we propose a novel synergistic graph convolutional network and graph attention network (SGCAT)-based method for HT detection in pre-layout register-transfer level (RTL) designs. The proposed method combines the strengths of graph convolutional neural network (GCN) and graph attention network (GAT) to provide the precise detection and localization of HTs in RTL designs. From the observation of the experimental results, the proposed method demonstrates better performance in terms of accuracy, F1-score, precision and recall for HT detection.**

Keywords—Hardware Security, Hardware Trojan, Graph Neural Network, Graph Convolutional Network, Graph Attention Network

I. Introduction

With the advancement and globalization of modern circuits, the manufacturing complexities of integrated circuits have led to an outsourcing model of VLSI designs. In the outsourcing model of VLSI designs, the security issue of hardware Trojan (HT) poses various threats to hardware systems. Basically, an HT is a malicious circuit implanted within a hardware system that is activated under specific conditions and is hard to be detected. In recent years, significant progress has been made by using machine learning (ML)-based methods for HT detection. Many ML-based methods have been proposed to use the extracted features or the graphical representations for HT detection on pre-silicon data. However, most of the published papers rely on manually extracting features from hardware data to train the ML models.

To our knowledge, a new neural network (NN)-based model known as graph neural network (GNN) has gained widespread application to hardware security. For instance, a well-known GNN model [1-2], graph convolutional network (GCN), has been employed for HT detection. The proposed approach leverages the power of NN to analyze and interpret the hardware systems. In addition, the notable implementations including GNN-based models [3] have achieved impressive results in the HT localization process. As the transformer-based model makes the important breakthroughs in the NN field, the graph attention network (GAT) can be treated as an operator with a self-attention mechanism for GNN. In this paper, the synergistic graph convolutional network and graph attention network (SGCAT)-based method using the GAT is proposed for accurate detection and localization of HTs.

II. Methodology

For HT detection and localization using the proposed SGCAT-based method, the detection and localization process can be divided into three steps: *Data Preprocessing*, *GNN-Based Model Design* and *Threshold Selection for HT Detection*.

A. Data Preprocessing

For HT detection in register-transfer level (RTL) designs, the circuits can be embedded into a graph. For the graph construction, each computing unit or operation can be treated as a node, and the relation between two computing units or operations can be treated as an edge. For implementation, the parsing tool, Pyverilog [4], can be employed to parse any hardware circuit into an abstract syntax tree (AST) and support the signal analysis on the AST to produce the data flow graphs (DFG) [5].

B. GNN-Based Model Design

To establish an accurate GNN-based model for HT classification and localization, a collaborative method with GCN and GAT can be designed. The proposed SGCAT-based method draws the inspiration from [6] which simultaneously combines the GCNs and the self-attention models to achieve the remarkable prediction for the classification tasks of graph data.

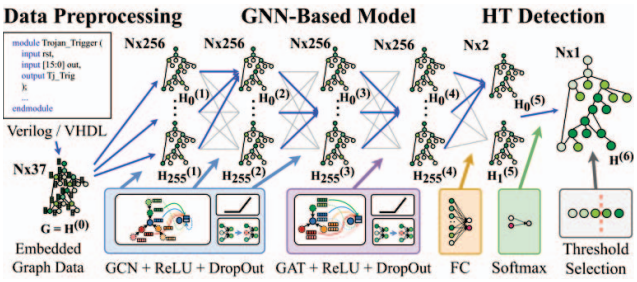

Fig. 1. Model architecture of SGCAT-based method for HT detection

Fig. 1 illustrates the model architecture of the proposed SGCAT-based method. To efficiently learn the graph feature of HT, the architecture of the GNN-based model includes 3 GCN blocks and 1 GAT block. For each GCN or GAT block, a GCN layer or GAT layer for the feature extraction and a ReLU for the activation function is used. In addition, the dropout technique is applied to avoid overfitting during model training. After constructing the GCN and GAT blocks, a fully connected layer is added to learn the relationship between the characteristics of these graph features. Finally, a softmax layer is used to

2833-292X/24/$31.00 ©2024 IEEE
DOI 10.1109/DSN-S60304.2024.00047

transform the output value of the model into the suspiciousness score for each node.

C. Threshold Selection for HT Detection

The suspiciousness score from the GNN-based model is used to detect Trojan nodes. Because the number of the benign nodes is much larger than the number of the Trojan node, an appropriate threshold can be used to determine the accurate Trojan node based on the suspiciousness scores. In the field of machine learning, the receiver operating characteristic curve (ROC) is a common technique to evaluate the performance of the binary classification model. Based on the ROC of the detection results, the threshold can be further adjusted for HT detection and localization. After selecting an appropriate threshold, the detection results can be optimized during the testing process.

III. Experimental Results

To evaluate the proposed SGCAT-based method, the dataset can be constructed from the combination of HT modules and Trojan-free circuits. To increase the number of the available data in the dataset, the 21 HT modules inside the Trojan-inserted AES data on trust-hub can be extracted and implanted into the other 4 Trojan-free circuits (DES, RC4, RC5 and RC6). To evaluate the model's reliability and detection accuracy, all of the graph data based on AES, DES and RC5 Trojan-free circuits are used in training process. On the other hand, the rest of the graph data based on RC4 and RC6 Trojan-free circuits are used in testing process.

A. Kernel Density Estimation Plots of Model Output

Fig. 2 shows the kernel density estimation plots for the output values using Yasaei's method [3] and our proposed method. In the figure, the output values of our proposed method are more clustered than that of Yasaei's method [3] in the value distribution for HT detection. It means that our proposed method can provide the accurate detection of Trojan nodes. For HT detection, Yasaei's method and our method optimize the classification results by using the threshold selection. As shown in Fig. 2, the threshold can be adjusted in the trade-off between the misclassification results between the false positives and false negatives to achieve better accuracy and F1-score.

B. Comparison with State-of-the-Art

In Table I, the training set can be used for training and evaluating the detection accuracy. For each type of circuits, the first row is the results using Yasaei's method [3] and the second row is the results using our proposed method. For HT detection on the training dataset, it is clear that accuracy and precision can be highly maintained and the recall and F1-score can be significantly improved. For example, the detection accuracy in terms of F1-score on the circuit data based on the AES, DES and RC5 circuit can be improved from 82.55% to 94.65%, from 82.4% to 93.82% and from 83.03% to 89.1%, respectively. On the other hand, the testing set can be used to further evaluate the detection accuracy. For example, the detection accuracy in terms of F1-score on the circuit data based on the RC4 and RC6 circuits can be improved from 80.94% to 87.97% and from 83.22% to 88.31%, respectively. The result indicates that our proposed method maintains the accuracy of HT detection for larger circuits and reduces the probability of the false positives on benign nodes.

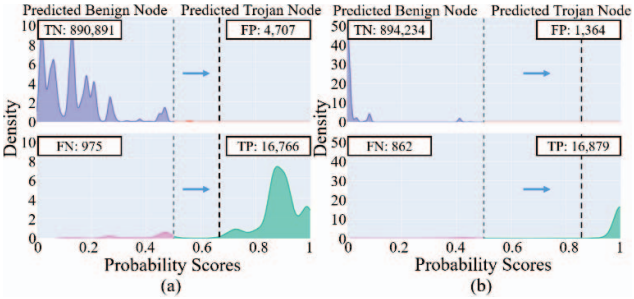

Fig. 2. Kernel density plot of model's output on X-axis for the classification probability values and the Y-axis for the probability density values. (a) Yasaei's method [3] and (b) Our method.

IV. Conclusion

The SGCAT-based method is proposed to demonstrate better performance for HT detection and localization. The proposed method effectively reduces the misclassification rate for the detection of Trojan nodes. As the proposed method is applied to unseen circuits (e.g., RC4 and RC6), the results can still be achieved with the accuracy of 93.33% and 87.97% and the F1-score of 98.59% and 88.31%, respectively.

References

[1] R. Yasaei *et al.*, "GNN4TJ: Graph Neural Networks for Hardware Trojan Detection at Register Transfer Level," in *Proc. of Design, Automation & Test in Europe Conf. & Exhibition* (DATE), pp. 1504–1509, Feb. 2021.

[2] S.-Y. Yu *et al.*, "HW2VEC: a Graph Learning Tool for Automating Hardware Security," in *Proc. of Int'l Symp. on Hardware Oriented* (HOST), pp. 13–23, Dec. 2021.

[3] R. Yasaei *et al.*, "Golden Reference-Free Hardware Trojan Localization Using Graph Convolutional Network," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 30, no. 10, pp.1401–1411, Oct. 2022.

[4] S. Takamaeda-Yamazaki, "Pyverilog: A Python-Based Hardware Design Processing Toolkit for Verilog HDL," in *Proc. of Int'l Symp. on Applied Reconfigurable Computing* (ARC), pp. 451–460, Apr. 2015.

[5] R. Namballa *et al.*, "Control and Data Flow Graph Extraction for High-Level Synthesis," in *Proc. of IEEE Computer Society Annual Symp. on VLSI* (ISVLSI), pp. 187–192, Feb. 2004.

[6] D. Q. Nguyen *et al.*, "Universal Graph Transformer Self-Attention Networks," in *Proc. of Web Conference* (WWW), pp. 193–196, Aug. 2022.

TABLE I. Comparison of Average Score on Each Evaluation Result with Yasaei's Method [3]

Training Set	Method	AES Series				DES Series				RC5 Series			
		Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
	[3]	99.5%	94.28%	74.43%	82.55%	99.78%	94.21%	74.53%	82.4%	97.84%	94.76%	75.32%	83.03%
	ours	**99.83%**	**96.43%**	**93.01%**	**94.65%**	**99.91%**	**95.07%**	**92.72%**	**93.82%**	**98.58%**	86.41%	**92.73%**	**89.1%**
Testing Set	**Method**	**RC4 Series**				**RC6 Series**							
		Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score				
	[3]	90.68%	88.02%	75.6%	80.94%	98.06%	94.86%	75.6%	83.22%				
	ours	**93.33%**	84.48%	**92.77%**	**87.97%**	**98.59%**	85.04%	**92.77%**	**88.31%**				