IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Accuracy Analysis using Machine Learning Classifiers for Hardware Trojan Detection

R. Bharathi Jyothi
Department of Computer Science and Engineering
Vignan's Institute of Engineering for Women
Visakhapatnam, India

P.S.S Keerthi
Department of Computer Science and Engineering
Vignan's Institute of Engineering for Women
Visakhapatnam, India

S. Sireesha
Department of Computer Science and Engineering
Vignan's Institute of Engineering for Women
Visakhapatnam, India

P. Krishna Priya
Department of Computer Science and Engineering
Vignan's Institute of Engineering for Women
Visakhapatnam, India

P. Namratha
Department of Computer Science and Engineering
Vignan's Institute of Engineering for Women
Visakhapatnam, India

Mrs.V.Sree Lahari
Assistant Professor
Department of Computer Science and Engineering
Vignan's Institute of Engineering for Women
Visakhapatnam, India

Abstract- Security threats are a huge concern in our world as technology advances every day. As the proliferation of IoT devices is observable across a various wide range of applications, it becomes necessary to ensure the hardware security in these devices to build a secure infrastructure. Especially, products designed from the IC market are prone to potential threats in the form of hardware Trojans due to a spike in outsourcing of components. Hardware Trojans are nothing but slight malicious modifications that are made to the circuit which prevents the functioning of the circuit properly. Hardware Trojans differ in their characteristics based on their physical representation and actions. As most of the smart devices possess IC as an integral part, it becomes necessary to detect hardware Trojans before the product reaches the market to ensure customer safety as well as the company's reputation. As Hardware Trojans are normally very hard to find/detect we utilize machine learning algorithms like Random Forest classifier, AdaBoost classifier, Decision tree, and Gradient Boosting classifier to detect Hardware Trojans present in the circuit. For which, we utilize features extracted from gate-level netlists to train the models and testing to find out the efficiency of our method. We obtain improved performance metrics by tuning hyperparameters of the models. We make a comparative study of the performance of several algorithms. We propose decision fusion to further enhance the detection of hardware Trojans which involves combining various decisions made by the algorithms that we use to make a common decision. For decision fusion OR and Voting Weighed Average are used to make the final decision.

Keywords-Feature Extraction,Random Forest Algorithm,Ada Boost Classifier,Decision Tree Algorithm, Gradient Boosting Algorithm, Decision Fusion

I. INTRODUCTION

In this digital age, there is a growing demand to address the security threats to which products may be exposed. In the integrated circuit market in particular, several foundries subcontract some of their components to third-party suppliers, in order to reduce production costs, and then use them to deliver their final product. This represents a huge potential for a security threat, as these IP address providers can be unreliable. The safety of a product can be compromised in several ways at different levels of a product's lifecycle.

In order to strengthen the defense against hardware Trojans, we have used the Random Forest Classifier. The power of artificial intelligence is pronounced in several advanced areas. Therefore, we have used it as an efficient tool that could harness the potential to detect hardware Trojans efficiently. Then, we carried out a comparative study of different ensemble methods such as Ada boost, Gradient boosting and random forest algorithm.

II. LITERATURE SURVEY

There are several remarkable pieces of research focused on detecting hardware Trojans with new approaches. Elnagger et al. details several guidelines that must be strictly observed when applying machine learning algorithms to detect hardware Trojans and the risks they involve [4]. Hasegawa et al. provides detection of vector machine based hardware Trojans in [1], a random forest based classifier with features taken from netlists at gate level in [2] and compare several advanced ML algorithms, namely SVM, network neutral, multineural network and random fores based on several performance metrics in [3].

In [5], Xue et al. reduce the effect of operating variations by referencing neighboring elements and were able to obtain competitive results compared to the state of the art. In [6] Zhao et al. propose hardware-based execution approaches by applying chaos theory and have been able to outperform conventional execution approaches in terms of computational complexity, detection rate and implementation feasibility. Huang et al. provide a classification of all possible hardware Trojan attacks and machine learning-based approaches perform the detection process [7].

In [8], Sumathi et al. analyses possible hardware trojan attacks in programmable logic devices and applications specific ICs life cycle and explores phase-wise defence solutions in an elaborate manner.In [9], Amelian et al. present a side-channel analysis method to detect hardware trojans using path delay measurement. They were even able to test the circuits received after fabrication and mitigate the overhead cost seen in conventional approaches.

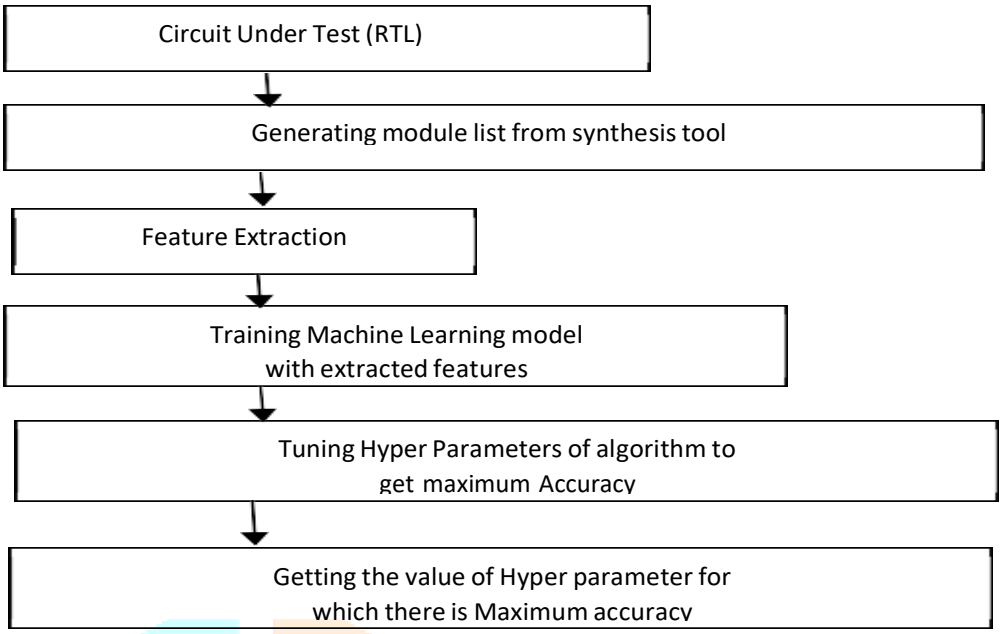
In this article, we have made a comparative study of different algorithms. Unlike several studies working with the detection of hardware Trojans using machine learning, we focus on setting the hyperparameters to achieve maximum accuracy. We use decision-making technology to supplement the mistakes of individual classifiers and take advantage of an improved method of detecting hardware Trojans.

III. METHODOLOGY

The procedure followed in our study is shown in Fig. 1.We extracted several features from the compact dataset and trained random forest classifier. We tuned several hyperparameters inherent to ensemble models to study their performance under different conditions.Finally, wewere able to obtain improvised performance. We made a study over the performance of several other ensemblemethods, namely Ada boost and Gradient boosting in a similar fashion.

We performed decision fusion over these classifiers in order to exploit the potential observed when their decisions are combined to detect hardware trojans accurately. We utilized several decision fusion techniques,namely voting, weighed average and ‘OR’ logic to improvise the results obtained.

Fig.1. Proposed Methodology



IV.Feature Extraction

A feature can be described as an individualistic property or characteristic of any observed process. It is crucial to choose informative and independent features without any bias in order to formulate an effective algorithm for classification as well as regression.

We performed **levelization** of gates from provided gate-level netlist in Python. Several features have been extracted from Gate-level netlists to detect hardware trojans.

1. **Primary input :**Distance of the net from primary input in terms of level.
2. **Primary output :**Distance of the net from primary output in terms of level.
3. **Connectivity :**Number of gates each net in netlist is connected with.
4. **Level :**Each gate has several inputs from previous gates. Among them, the one which require values to cross more number of gates is added with 1 to obtain current level.
5. **Fan-in x :**Number of inputs that are present in 'x' level away from a net.
6. **Score :**Primary input, primary output, connectivity and level has been added.
7. **Fan-out:**The output of a logic gate drives a number of gate inputs that indicates fan-out of that logic gate.

Algorithms Used:

We have used four different Machine Learning algorithms in this study to compare and contrast their performances, the algorithms are as follows:

Decision Tree:

Decision Tree makes use of a model, were in a structure that resembles a tree used to make decisions and there likely outcomes, as well as chance event outcomes, resource costs, and utility. Each node in the tree is a representation of a conditional statement('if') and on the whole the decision tree can be viewed as a representation of a nested conditional.

Random Forest:

Random forest is an ensemble learning method that is applicable for classification as well as regression by combining anaggregate of decision trees at training time and the output of this algorithm is based on the output(can be either mode or mean/avergae) of the invidual tress that constitute the forest.

AdaBoost:

Similar to random forest, AdaBoost is also an ensemble learning method that was originally created to enhanceperformanceof classifiers for which it makes use of an iterative approach in order to rectify mistakes of the weaker classifiers .

Gradient Boosting:

Gradient boosting is techinique is applicable for classification and regression problems. This method creates a prediction model that is similar to an ensemble of prediction model that are weak, characteristically desicion trees .

Decision fusion

Decision fusion is a type of data fusion that characteristically combines the decisions of multiple classifiers to achieve better results by working in a complementary manner. We trained the individual classifiers using train data. Then we passed the test data onto the classifiers. Finally, we combined the outcome of these classifiers and identified a complementary and unified approach using decision fusion for detecting hardware trojans with improved accuracy.

V.EVALUATION METRICS

Several metrics can be used to evaluate a Machine Learning algorithm. Accuracy evaluates the ratio of correct predictions to the total number of samples given as input.

$$Accuracy = (TP + TN) \div (Total\ number\ of\ samples)$$

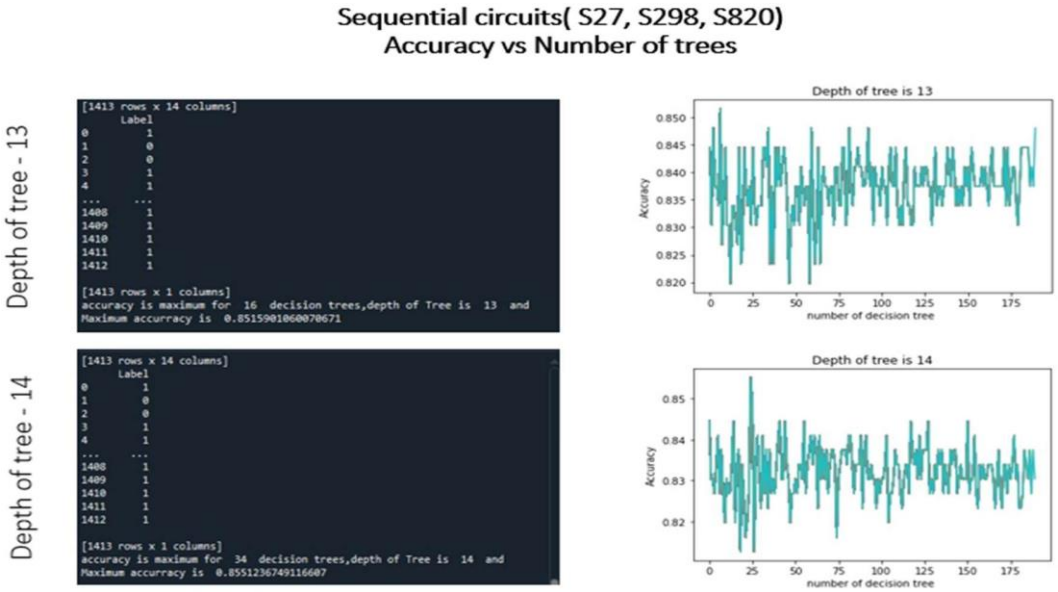
Precision evaluated the number of correct positive results relative to number of positive results obtained by classifier. Recall is the number of correct positive results relative to the number of samples that should have beenidentifies as positive.

$$Precision = TP \div (TP + FP)$$

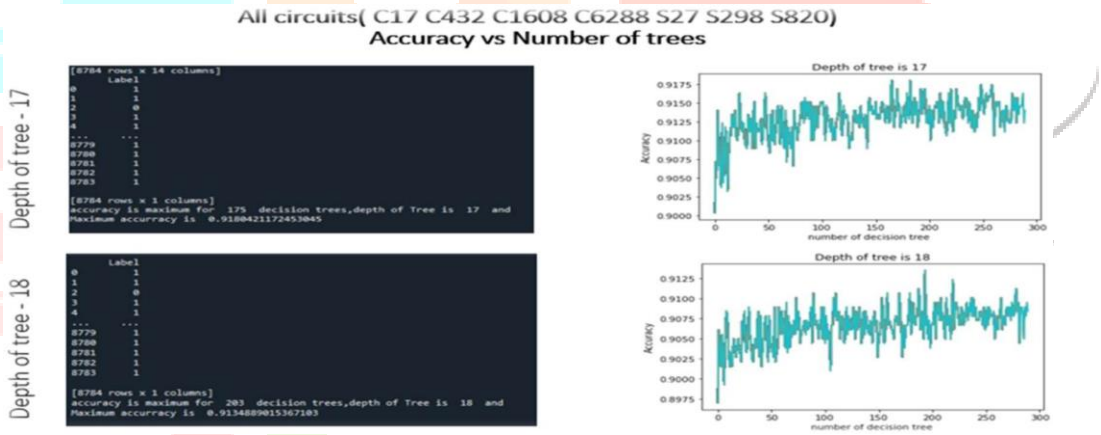
$$Recall = TP \div (TP + FN)$$

F1-score evaluates harmonic mean between precision and recall. It is good indicator of model performance especially when it is trained with imbalanced dataset.

$$F1 - Score = 2 \times [(Precision \times Recall) \div (Precision + Recall)]$$



VI. RESULT ANALYSIS

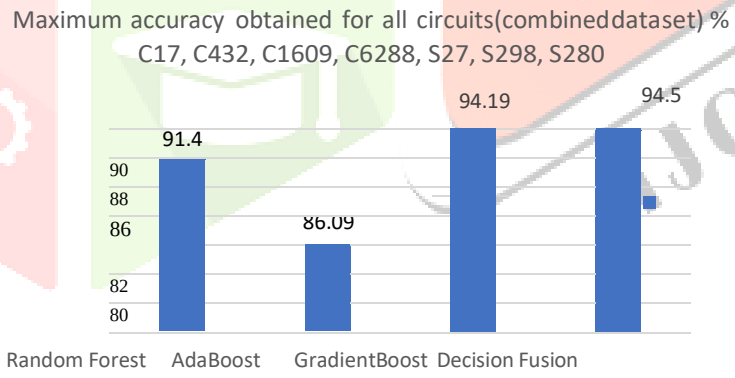


It was observed that Adaboost produce better accuracy when the learning rate is set to 2.5 and the Gradient Boosting performs better with a learning rate of 0.2 and when the max depth is set to 3. Table 1 summarizes the maximum accuracy attained in each circuit with always on /combinational/sequential trojans with decision trees, Ada Boost and Gradient Boost classifiers with appropriate depth and learning rate. We can compare the performance of this algorithms effectively study over the performance of each of these algorithms effectively. Decision trees classifier has a better average accuracy compared to other classifiers .

Circuits	Type	Decision trees	AdaBoost Classifier	Gradient Boosting Classifier
		Accuracy	Accuracy	Accuracy
C17	Always on	100	93.8	93.8
	Combinational	100	76.9	100
	Sequential	100	94.1	70.6
C432	Always on	97.0	98.5	98.0
	Combinational	92.9	67.8	84.9
	Sequential	97.9	91.8	96.4
C1608	Combinational	98.3	96.8	98.3
	Sequential	96.4	98.0	97.7
C6288	Always on	85.6	87.2	85.2
	Combinational	82.7	83.2	82.6
	Sequential	77.6	91.9	75.0
S27	Always on	95.2	70.2	90.5
	Combinational	94.4	89.9	100
	Sequential	90.9	72.7	77.3
S298	Always on	97.2	74.5	89.4
	Combinational	90.6	79.1	92.1
S820	Always on	94.2	93.5	93.5
	Combinational	91.9	83.2	87.7

Decision Fusion:

After observing the performances of the Random forest, GradientBoost and AdaBoost classifiers over the individual ISCAS circuits, we performed data fusion. Then we utilized several decision fusion techniques, namely voting, weighed average and 'OR' logic to improvise the results obtained. Fig.11 indicates that voting mechanism can be used to combine decisions obtained from the three classifiers over all circuits(C17, C432, C1608, C6288, S27, S298, S820) when combined to obtain enhanced accuracy.



VII.CONCLUSION

We have performed hardware trojan detection using several machine learning algorithms and are able to obtain good accuracy in several circuits by tuning several hyperparameters. In our comparative analysis, it is observed that the random forest classifiers outperformed AdaBoost and GradientBoost classifiers. The average accuracy of93.58%, 85.88% and 89.72% are obtained for the Decision trees, AdaBoost and GradientBoost classifiers respectively. We utilized decision fusion techniques like voting, weighed average and OR-logic to enhance the performance metrics like accuracy score and F1-score. Decision fusion technique helped us to achieve 94.57% accuracy as well as 90.1% F-score over a dataset inclusive of ISCAS' 85 and ISCAS' 89 benchmark circuits (C17, C432, C1609, C6288, S27, S298, S280). In addition to this work, improved classifier system that includes some additional classifiers as well as features will be explored in our future work. Further, the algorithm could be explored for the unsupervised classification scenarios also.

VIII. REFERENCE

1. K. Hasegawa, M. Oya, M. Yanagisawa, and N. Togawa, "Hardware Trojans classification for gate-level netlists based on machine learning," in *Proc. IEEE Symposium on On-Line Testing and Robust System Design(IOLTS)*, pp.203–206, 2016.
2. K. Hasegawa, M. Yanagisawa, and N. Togawa, "Trojan-feature extraction at gate-level netlists and its application to hardware-Trojan detection using random forest classifier," in *Proc. International Symposium onCircuits and Systems*, pp. 2154–2157, 2017.
3. K. Hasegawa, Y. Shi and N. Togawa, "Hardware Trojan Detection Utilizing Machine Learning Approaches," 2018*17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering* (TrustCom/BigDataSE), New York, NY, 2018, pp. 1891-1896, doi: 10.1109/TrustCom/BigDataSE.2018.00287.
4. Elnaggar, R., Chakrabarty, K. Machine Learning for Hardware Security: Opportunities and Risks. J ElectromTest 34, 183–20
5. Xue,Hao&Ren, Saiyu. (2017). Self-Reference-Based Hardware Trojan Detection. IEEE*Transactions on Semiconductor Manufacturing*. 31. 2-11. 10.1109/TSM.2017.2763088.