Risk-Aware and Explainable Framework for Ensuring Guaranteed Coverage in Evolving Hardware Trojan Detection

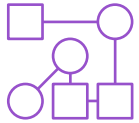
Rahul Vishwakarma
Graduate Research Assistant
California State University Long Beach

Amin Rezaei
Assistant Professor
California State University Long Beach

Agenda

Introduction

Problem statement and preliminaries



PALETTE

Solution and experimental results

Inference

Practical use cases in decision making

Hardware Trojans

Applied machine learning to detect hardware trojans

Hardware Trojans

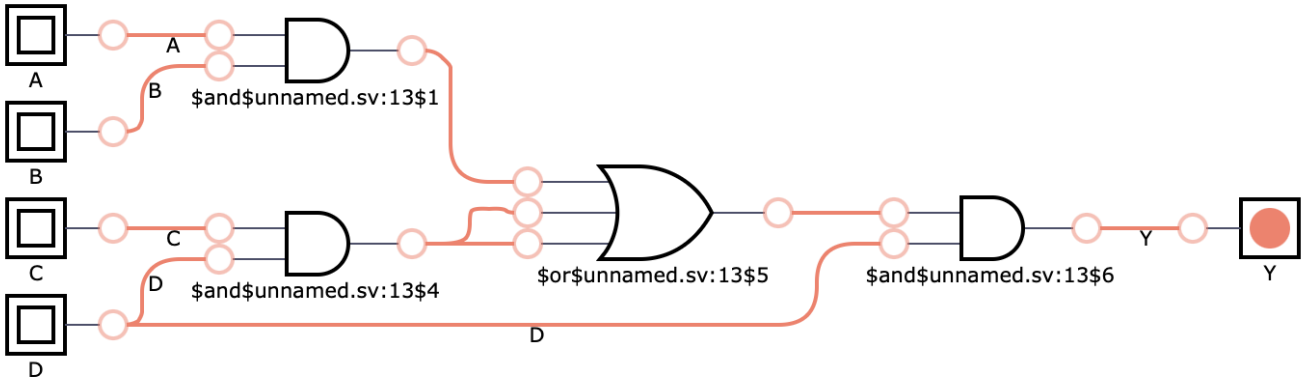


Figure 1: Trojan free circuit

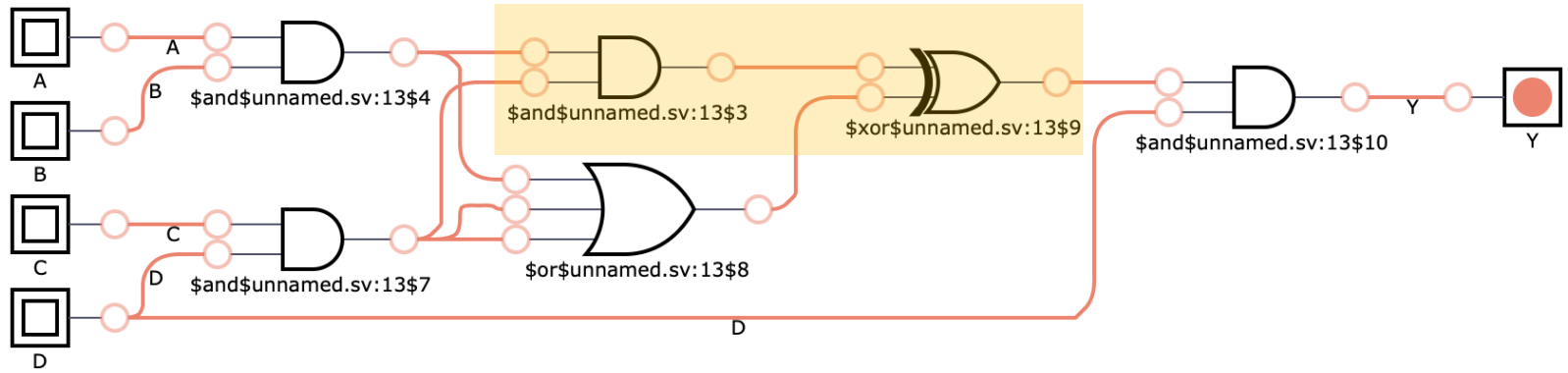


Figure 2: Trojan inserted circuit

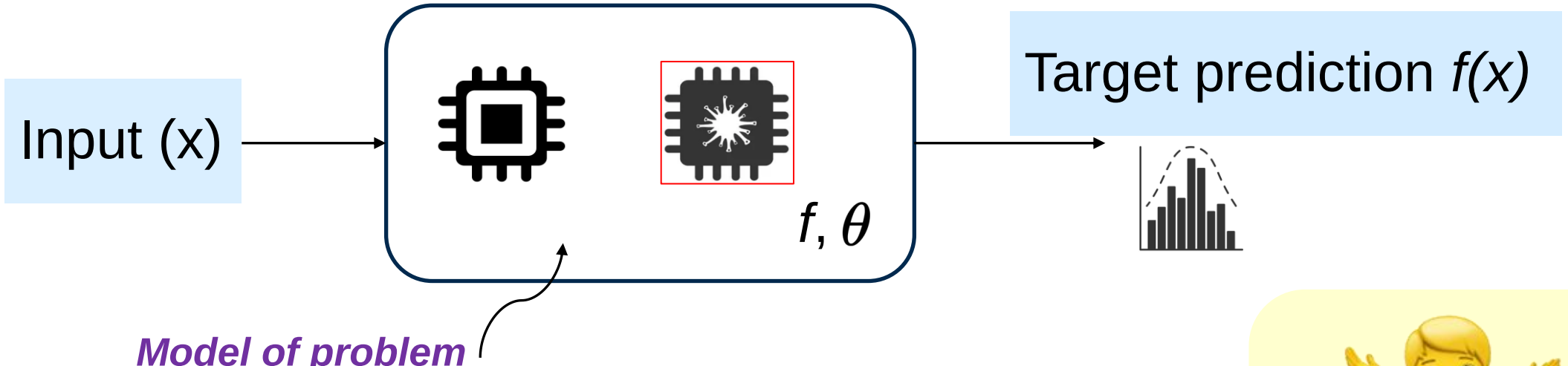
Conventional Approach

- Side-channel Power analysis
- Temperature analysis
- Wireless transmission power analysis
- Regional supply current analysis

Challenges

- Complexity of modern Integrated Circuits
- Challenges in detecting minute Hardware Trojans
- Early-Stage detection significance in design process
- Cost-Efficient Hardware Trojan detection strategies

Machine Learning



All models are wrong, but some are useful!

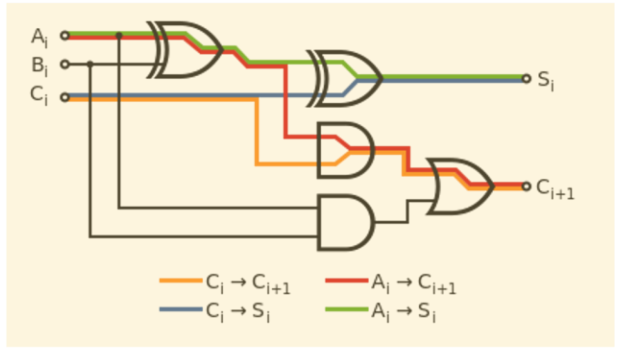
- Given a dataset $\{(x_1, y_1), \ldots, (x_N, y_N)\}$
- Pick θ that minimizes the Loss

$$\mathcal{L}(\theta) = \frac{1}{N} \sum_i L(f_\theta(x_i), y_i)$$

Machine Learning based solution

Process automation

State-of-the-Art Methods



Hardware Trojan Detection Engine

Trojan Free

Trojan Infected

Point Prediction

Lack of Trust

No Guarantee

$Input$

$x \boxed{f} \bullet y$

- Regression
- Deep Neural Networks
- Graph Neural Networks
- Reinforcement Learning

Research gap and **challenges**

PALETTE (Our Method)

Conformal Wrapper

$x \rightarrow \boxed{\pi_\theta \text{ (ML Model)}}$

$\{TI\ (0.45)\}$

Confidence Set $C_\theta(x)$

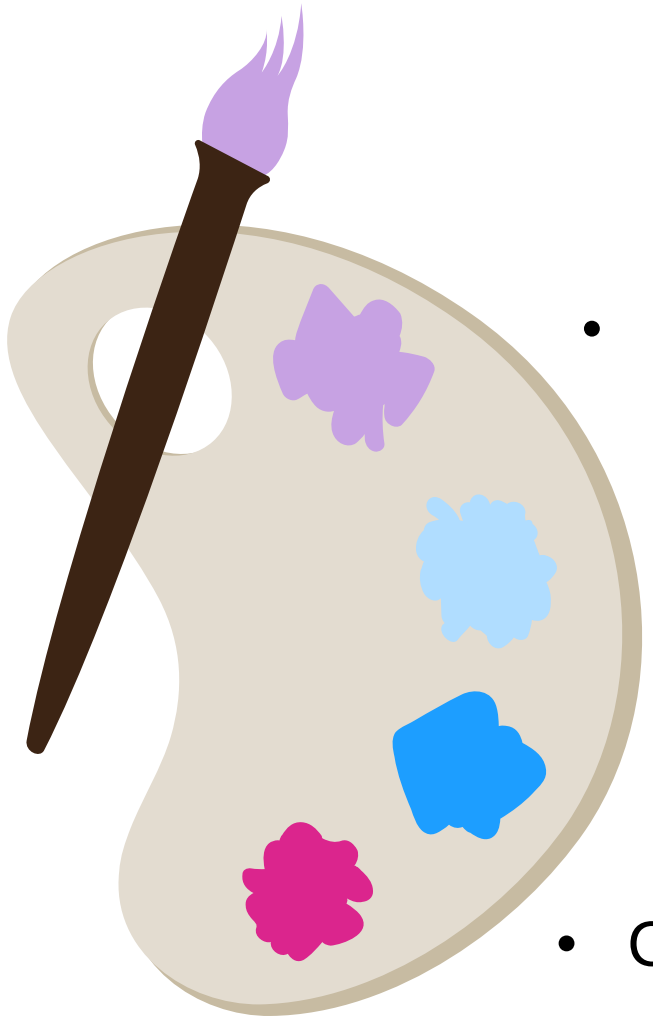
$\{TF, TI, \underline{T-EV}\}$ $\{T-EV\ (0.83)\}$

$x \boxed{f} \begin{matrix} \bullet \\ \bullet \\ \bullet \end{matrix} y$

PALETTE

Solution and experimental results

PALETTE: ex**P**lainable fr**A**mework for evo**L**ving hardwar**E** **T**rojan de**TE**ction



- Guaranteed coverage of each prediction
  - Model says – “I don’t know”
  - Ranking of Hardware Trojans
- Calibrated explanations for each decision

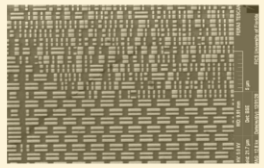
Proposed Solution

1 Data Features

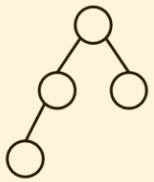
A Code Branching

Block	Pout	COND
1	1	1
1	1	1
1	0.5	4
1	0.25	4

B SEM Images



C RTL to Graph



2 Create evolving trojans

A Generate data

Source dataset has labels TF_s and TI_s

cGAN

Conformalized Generator & Discriminator

TF_s : Trojan free (source)
TI_s : Trojan infected (source)

B Label new dataset

$TF_s \rightarrow TF_G$ $TI_s \rightarrow TI_G$

C Merge with source

TF $\{TF_s, TF_G\}$, **TI** $\{TI_s\}$, **T-EV** $\{TI_G\}$

TF: Trojan free; **TI**: Trojan infected; **T-EV**: Trojan evolved

3 Conformal Framework

Original Evolved
Dataset

Train Calibrate Test

ML Model

$x \; f \; \bullet y$

nonconformity score

$\alpha = s(f(x), y)$

***p*-values**

$\hat{\mathcal{Y}} = \{y \mid s(f(x), y) \leq q\}$
Set Prediction $x \; f \; :y$

4 Inference

A Guaranteed coverage

$\mathbb{P}[y \in \hat{\mathcal{Y}}] \geq 1 - \epsilon$

B Set Prediction

$\{TF, TI\}$,
$\{TF, T-EV\}$,
$\{TF, TI, T-EV\}$

C Ranking HT

$\sup\{1 - \epsilon : |\Gamma_\epsilon(x)| \leq 1\}$

$c(HT_1) >, ..., c(HT_n)$

D Calibrated Explanation

Reject : $\{NULL\}$

Dataset

- GAINESIS[1]
- Chip-level Trojan: Trust-Hub[2]
  - Creating Evolved Hardware Trojans

$$Label = \{TF, TI, T - EV\}$$

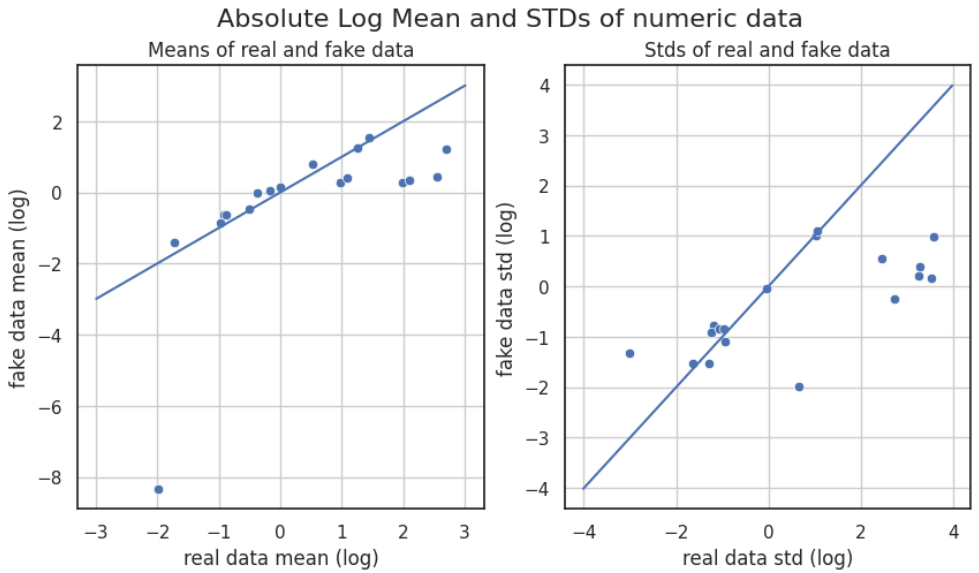
[1] K. G. Liakos, G. K. Georgakilas, F. C. Plessas, and P. Kitsos, “Gainesis: Generative artificial intelligence netlists synthesis,” Electronics, vol. 11, no. 2, p. 245, 2022.

[2] H. Salmani, M. Tehranipoor, S. Sutikno, and F. Wijitrisnanto, “Trust-hub trojan benchmark for hardware trojan detection model creation using machine learning,” 2022. [Online]. Available: https://dx.doi.org/10.21227/ px6s-sm21

Evolving Hardware Trojans

- Notion of Evolution:

$$E_{HT} \rightarrow HT \blacksquare HT_{structural\_changes}$$



Comparison of real and synthetically generated dataset on trust-hub chip-level trojan dataset.

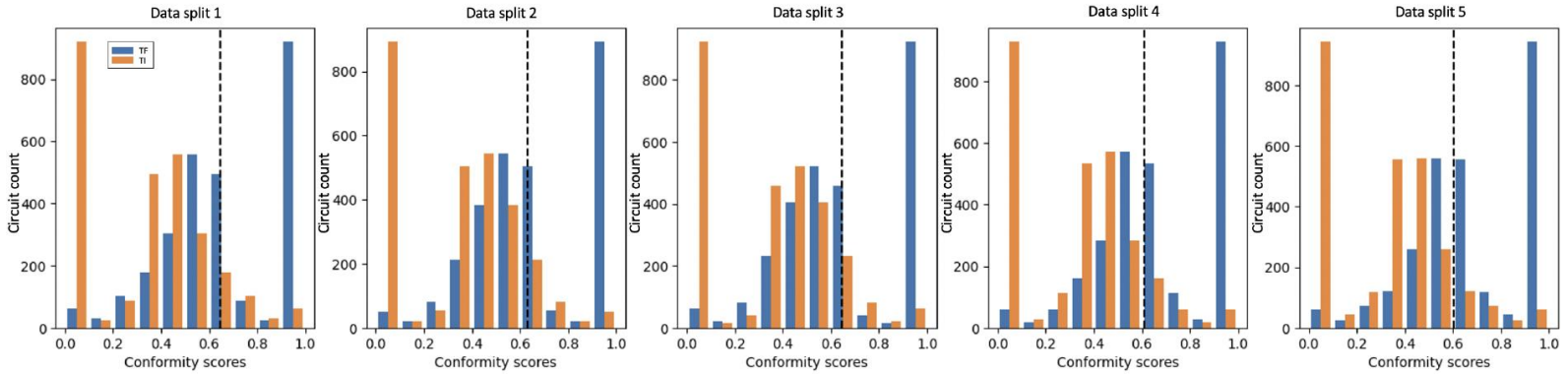
	Train	*Calibration*	*Test*
TF	1436	470	471
TI	114	33	44
T-EV	308	117	105
Total count	1858	620	620
T-EV	**16.50%**	**18.87%**	**16.93%**

Dataset split for model input

Experimental Results

circuit	TI	TF	y-pred	Conf
1	FALSE	TRUE	TF	0.891
2	FALSE	TRUE	TF	0.796
3	FALSE	TRUE	TF	0.996
4	FALSE	TRUE	TF	0.997
...	...	...	...	...
4596	FALSE	TRUE	TF	1
4597	FALSE	TRUE	TF	0.991
4598	TRUE	FALSE	TI	0.995
4599	FALSE	TRUE	TF	0.989
4600	FALSE	TRUE	TF	0.992

Conformal inference for GAINESIS dataset



Distribution of scores on each five of the calibration fold for GAINESIS dataset

Experimental Results

	TF	TI	T-EV	pTF	pTI	pT-EV	y_pred	Conf
1	T	F	F	0.319	0	0.003	TF	0.997
2	T	F	F	0.243	0.002	0.006	TF	0.994
3	T	T	F	0.161	0.078	0.016	TF	0.992
4	T	T	T	0.114	0.053	0.119	T-EV	0.886
5	T	F	F	0.645	0.001	0.004	TF	0.996
6	F	F	T	0.653	0	0.971	T-EV	0.365
7	T	F	F	0.3	0	0.002	TF	0.998

Conformal inference and associated p-vales for trust-hub chip-level trojan dataset.

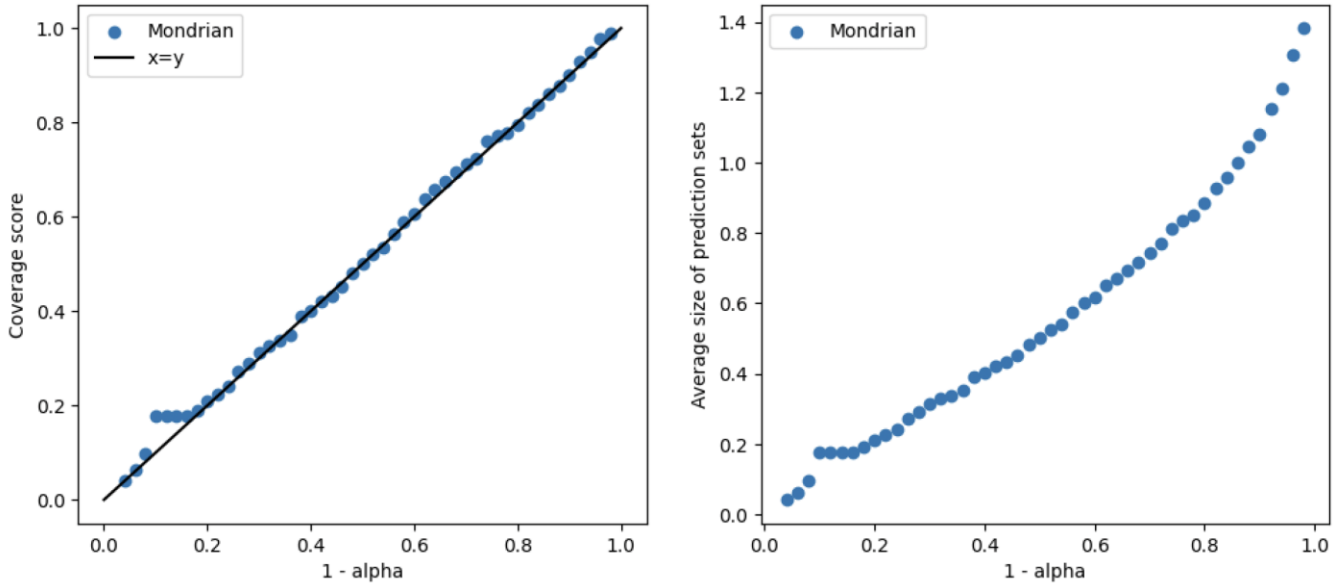
alpha	mondrian	raps	naïve	top_k
0.05	10	37	35	0
0.5	45	57	57	61
0.9	45	61	61	61

Comparison of conformal predictors with corresponding significance level on trust-hub chip-level trojan dataset

Performance Metrics

Coverage: Proportion of true target values that fall within the pred intervals.

Efficiency: How tight the prediction intervals are.



Effective coverage and average prediction set size for Trust-Hub chip dataset.

Performance Metrics

sig	mean_err	avg_c	n_correct	mean_T-EV
0.05	0.049	1.040	589	0.012
0.1	0.102	0.941	556	0.045
0.2	0.204	0.812	493	0.133
0.3	0.303	0.701	431	0.220
0.4	0.406	0.596	367	0.319
0.5	0.504	0.497	307	0.423
0.6	0.604	0.397	245	0.536
0.7	0.702	0.298	184	0.650
0.8	0.798	0.202	125	0.764
0.9	0.900	0.100	61	0.884

Performance metrics of conformal inference on trust-hub chip-level trojan dataset

Inference

Risk-Aware decision making

Risk-Aware Ranking

- Confidence score

$$\text{Confidence}(x) = \sup\{1 - \epsilon : |\Gamma_\epsilon(x)| \leq 1\}$$

- Assign confidence score

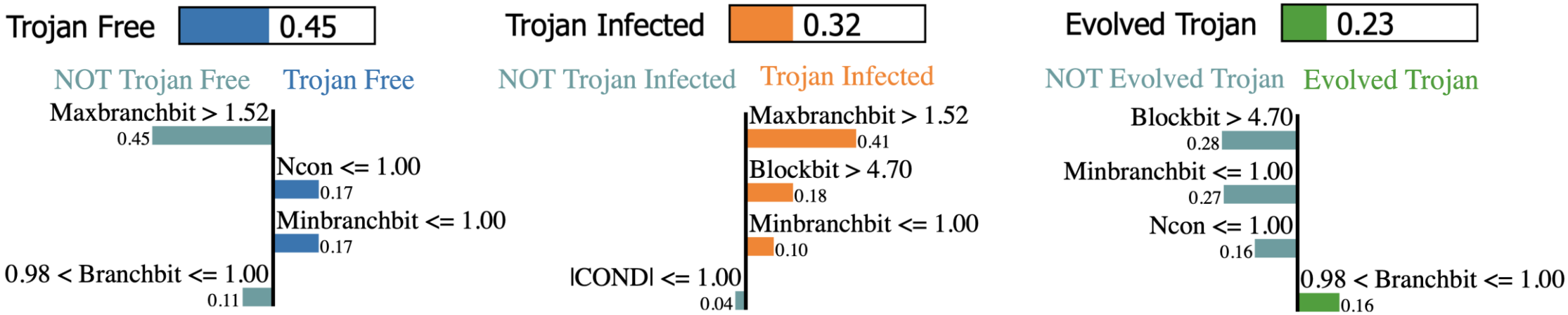
$$\alpha_{0.05}(\text{ circuit } 12) = \{T - EV\}_{C=0.88}$$
$$\alpha_{0.05}(\text{ circuit } 13) = \{T - EV\}_{C=0.81}$$
$$\alpha_{0.05}(\text{ circuit } 14) = \{T - EV\}_{C=0.61}$$

	confidence	credibility	y_pred
1	0.997	0.319	TF
2	0.994	0.242	TF
3	0.922	0.162	TF
4	0.886	0.119	T-EV
5	1	0.645	TF
6	0.999	0.97	T-EV
7	0.998	0.301	TF

Adoption of confidence for risk-aware ranking on trust-hub chip-level trojan dataset

Calibrated Explanation for Reject

- Explanation for each decisions are calibrated



Calibrated explanation for rejecting a decision

Model says – “I don’t know”

Algorithm 1: Prediction with Reject Option

Input: model, instance
Output: prediction

```
1 confidence_scores = model.predict(instance);
2 if max(confidence_scores) < threshold then
3     return "I don't know";
4 return class_with_highest_confidence(confidence_scores);
```

Algorithm 2: Prediction with Non-Conformity Measure Threshold

Input: model, instance
Output: prediction

```
1 non_conformity = calculate_non_conformity(model, instance);
2 if non_conformity > threshold then
3     return "I don't know";
4 confidence_scores = model.predict(instance);
5 return class_with_highest_confidence(confidence_scores);
```

Conclusion

Key Takeaways

- Evolving hardware Trojan (HT) evolution.
- Guaranteed coverage and tunable significance levels.
- Algorithm-agnostic and explainability-aware rejection of predictions.
- Ranking mechanism for evolved Trojans.

While there's no silver bullet for zero-day attacks, adopting a proactive risk-aware defense strategy significantly reduces the attack.

Amin Rezaei

Rahul Vishwakarma

Computer Architecture, Reliability, and Security Laboratory (CARS-Lab)

Scan me!