Hardware Trojan Detection using Shapley Ensemble Boosting

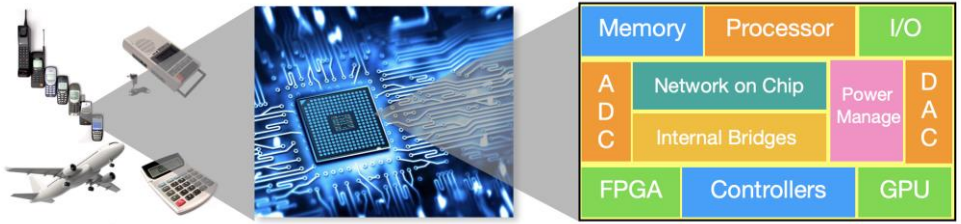
Zhixin Pan and Prabhat Mishra
Department of Computer and Information Science and Engineering
University of Florida, USA

This work was partially supported by the
National Science Foundation (NSF) grant SaTC-1936040

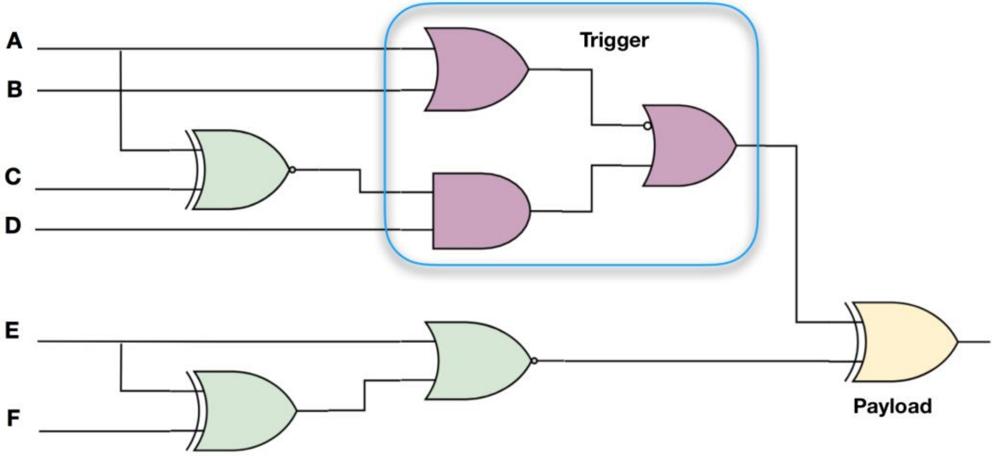
Outline

- <u>Introduction</u>
- Related Work
- Proposed Method
- Experimental Results
- Conclusion

Introduction



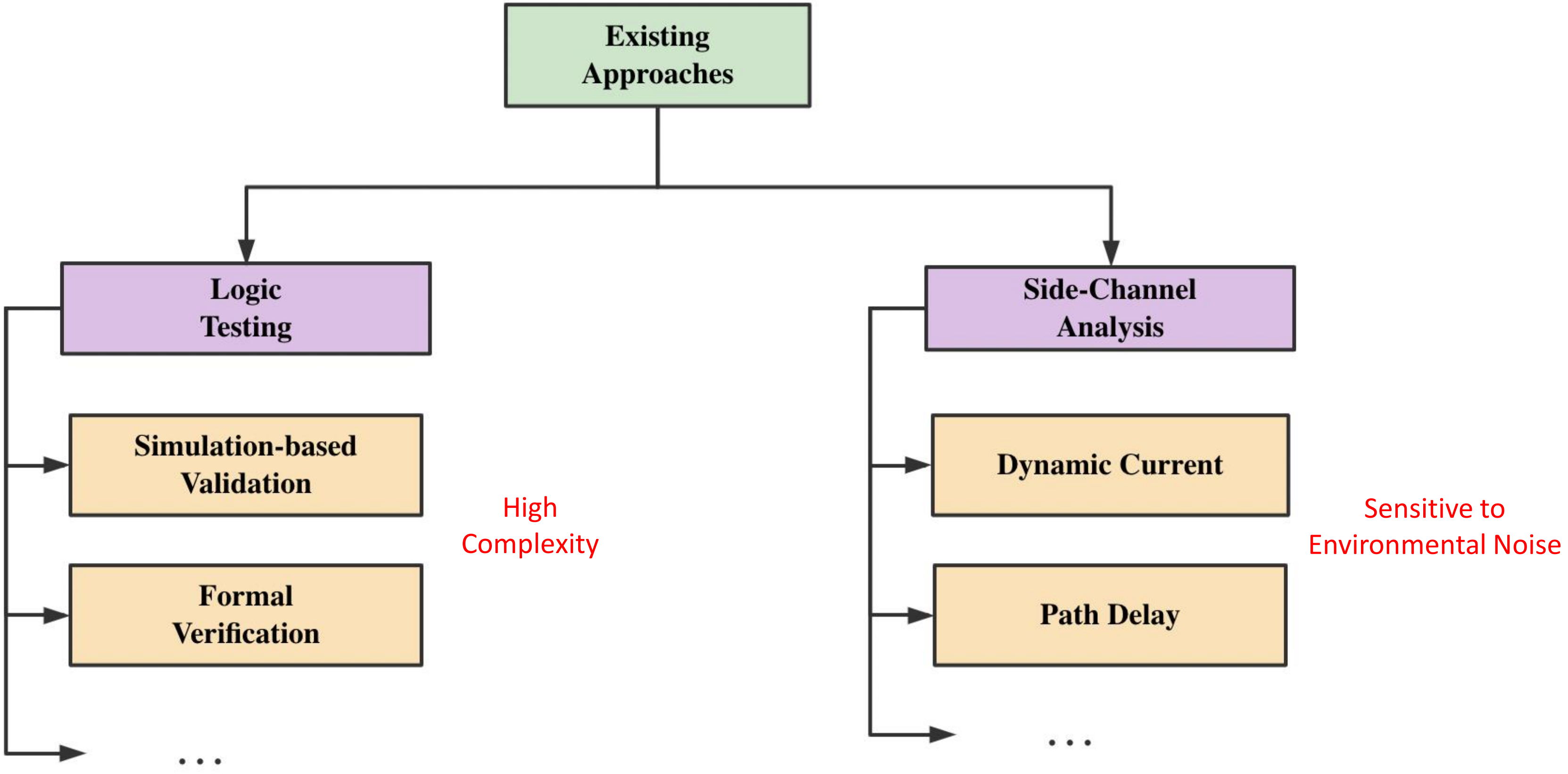
Hardware Trojan



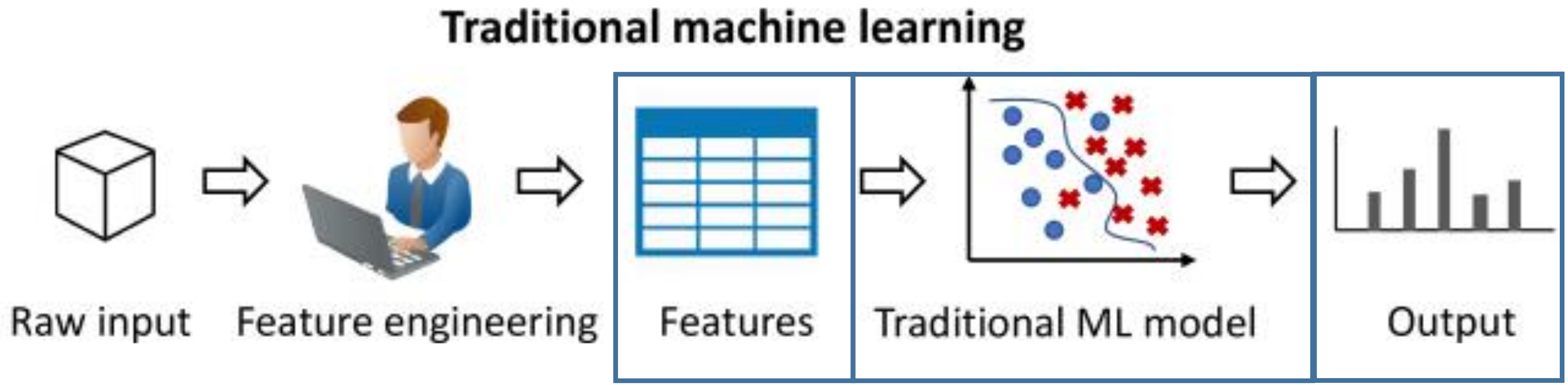
Outline

- Introduction
- Related Work
- Proposed Method
- Experimental Results
- Conclusion

Related Work



Machine Learning

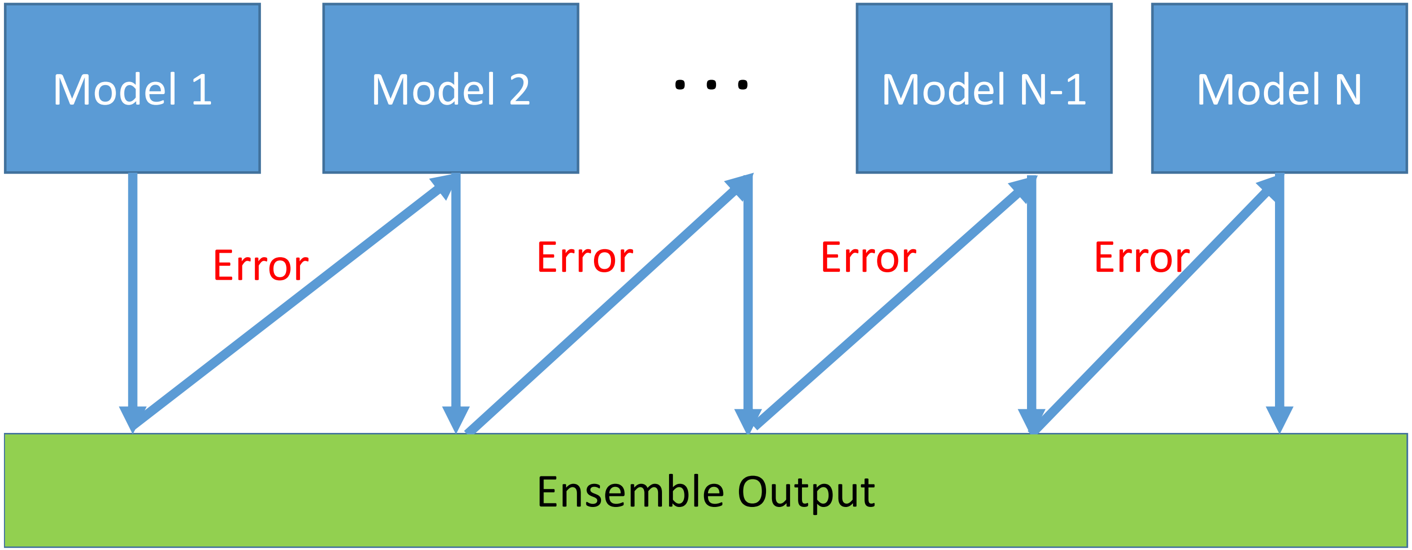


- No guideline for feature selection
- Expensive training cost and high model complexity
- Model can only provide result without interpretation (black-box nature)

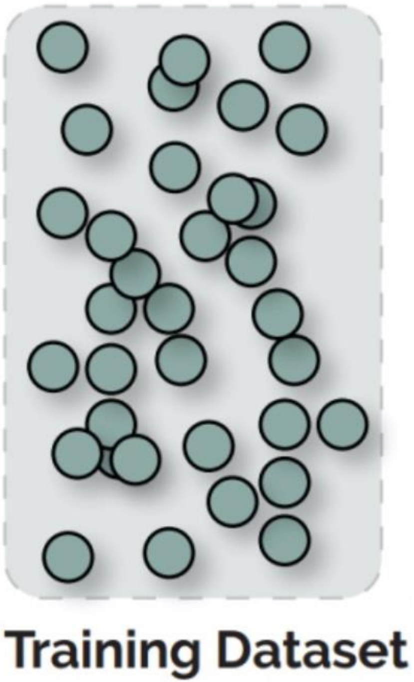
Outline

- Introduction
- Related Work
- Proposed Method
- Experimental Results
- Conclusion

Ensemble Boosting



Interpret the results



Shapley value can tell the contribution of each individual input.

Winter, Eyal. "The shapley value." *Handbook of game theory with economic applications* 3 (2002): 2025-2054.

Shapley Value Analysis

Shapley Values: Key idea → Marginal Contributions

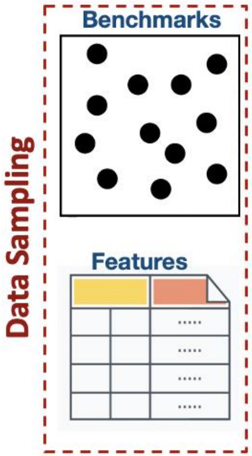
$$\phi_i(v) = \varphi_i = \sum_{S\subseteq N\setminus\{i\}} \frac{|S|!(|N|-|S|-1)!}{|N|!}\left(v(S\cup\{i\}) - v(S)\right)$$

Marginal Contributions of Feature 1

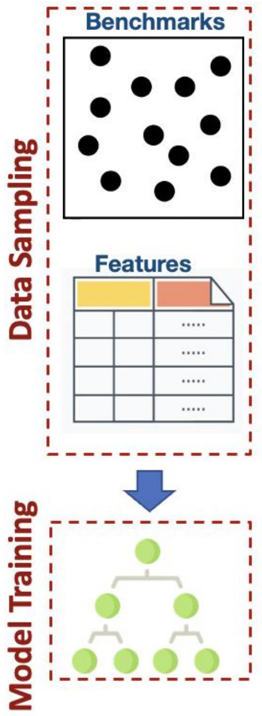
Sequences	Marginal Contributions
1,2,3	$\mathcal{L}(\{1\}) - \mathcal{L}(\emptyset)$
1,3,2	$\mathcal{L}(\{1\}) - \mathcal{L}(\emptyset)$
2,1,3	$\mathcal{L}(\{1,2\}) - \mathcal{L}(\{2\})$
2,3,1	$\mathcal{L}(\{1,2,3\}) - \mathcal{L}(\{2,3\})$
3,1,2	$\mathcal{L}(\{1,3\}) - \mathcal{L}(\{3\})$
3,2,1	$\mathcal{L}(\{1,2,3\}) - \mathcal{L}(\{3,2\})$

Average

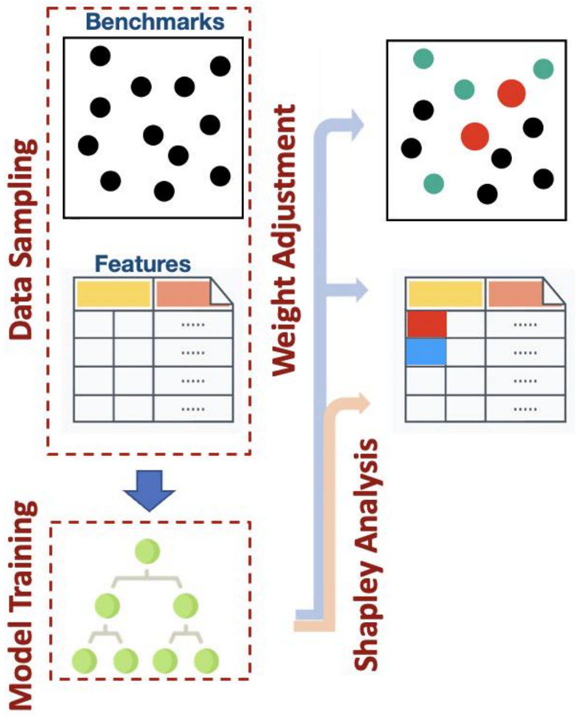
Proposed Framework



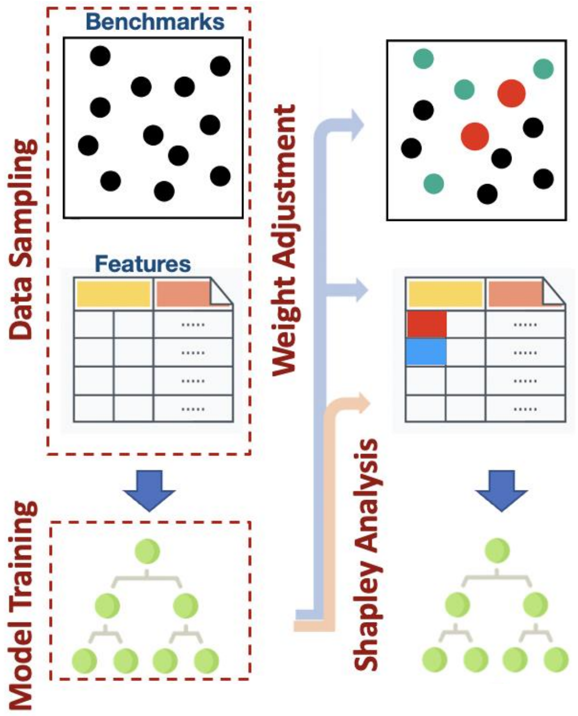
Proposed Framework

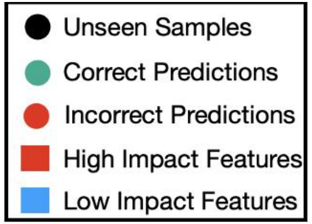


Proposed Framework

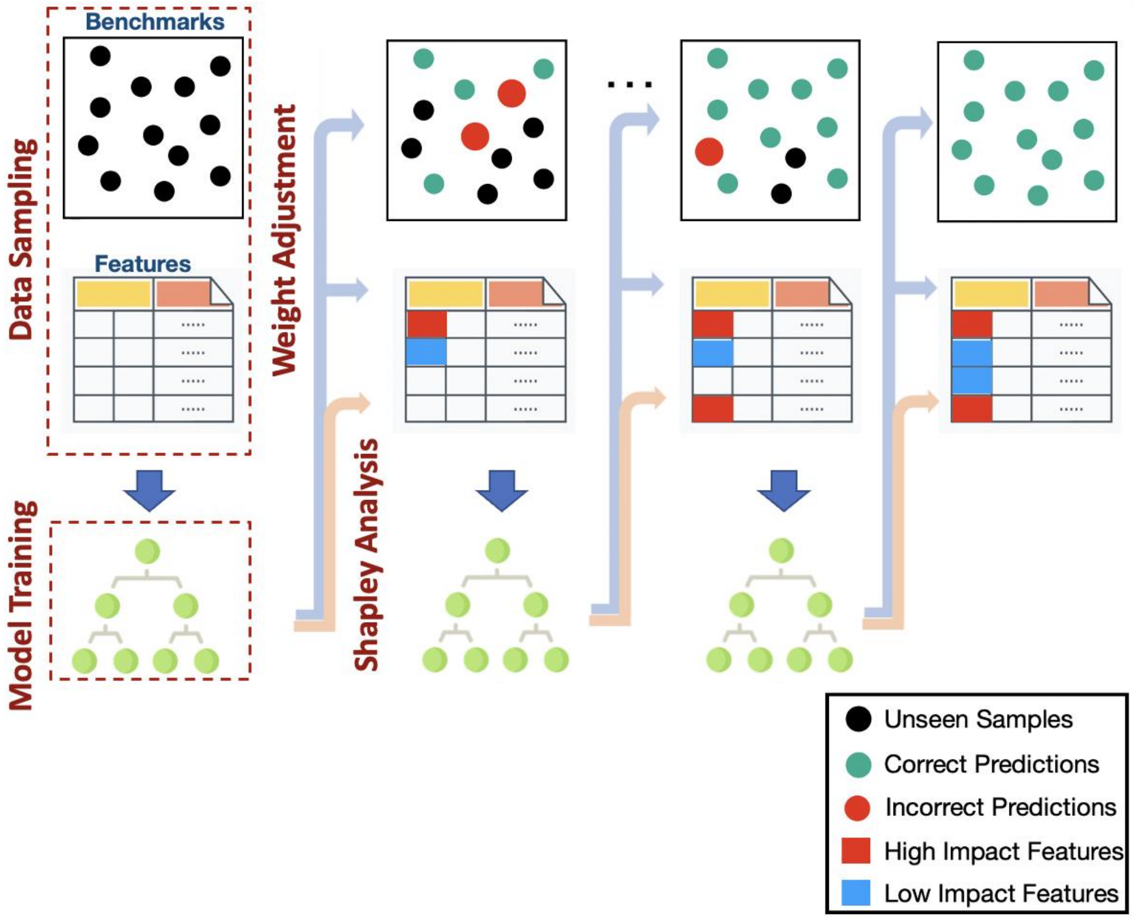


Proposed Framework

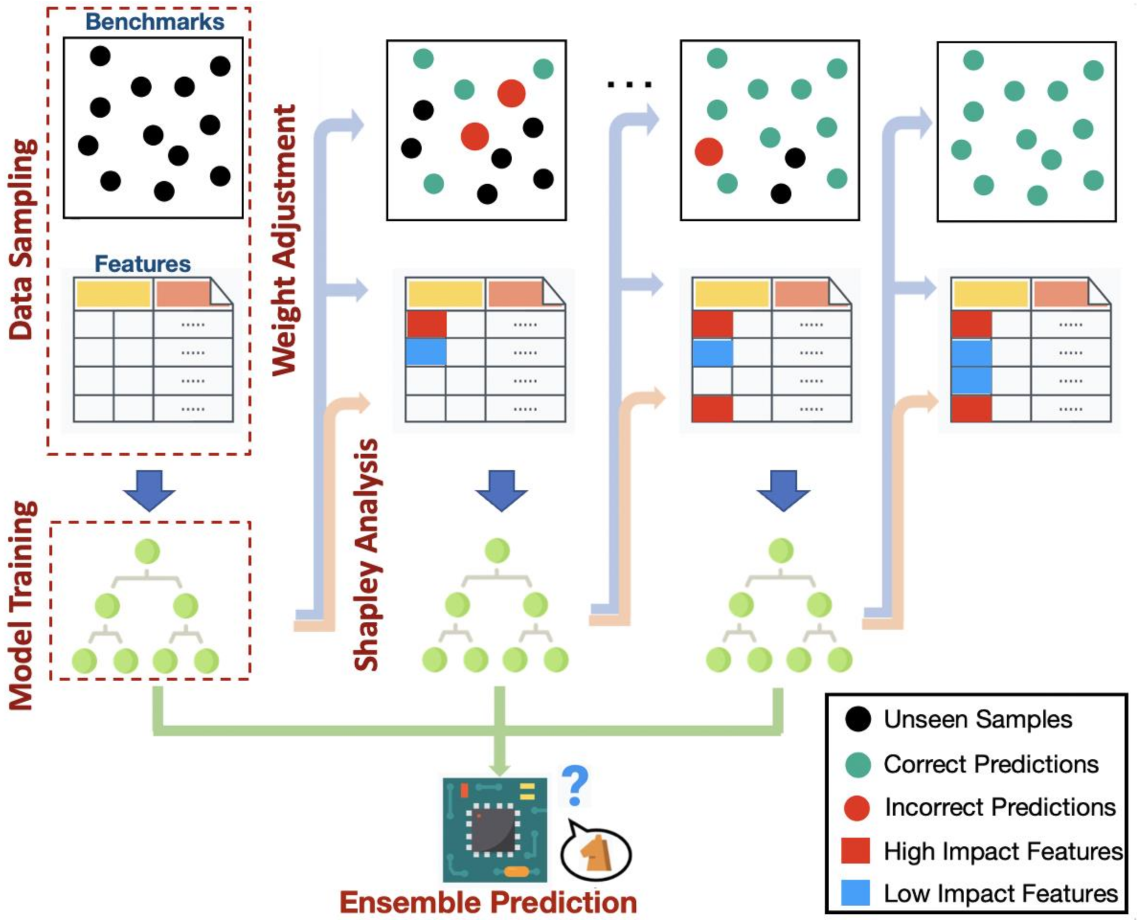




Proposed Framework



Proposed Framework



Outline

- Introduction
- Related Work
- Proposed Method
- <u>Experimental Results</u>
- Conclusion

Experimental Setup

- Intel i7 3.70GHz CPU, 32 GB RAM and RTX 2080 256-bit GPU
- PyTorch for ML library
- Compare the performance of 4 different models
    - RFC: Random Forest Classifier.
    - CNN: Convolution Neural Network (CNN)
    - TGRL: State-of-the-art Test generation using reinforcement learning
    - SEB: Proposed Shapley ensemble boosting framework

Performance Evaluation

Detection Accuracy:

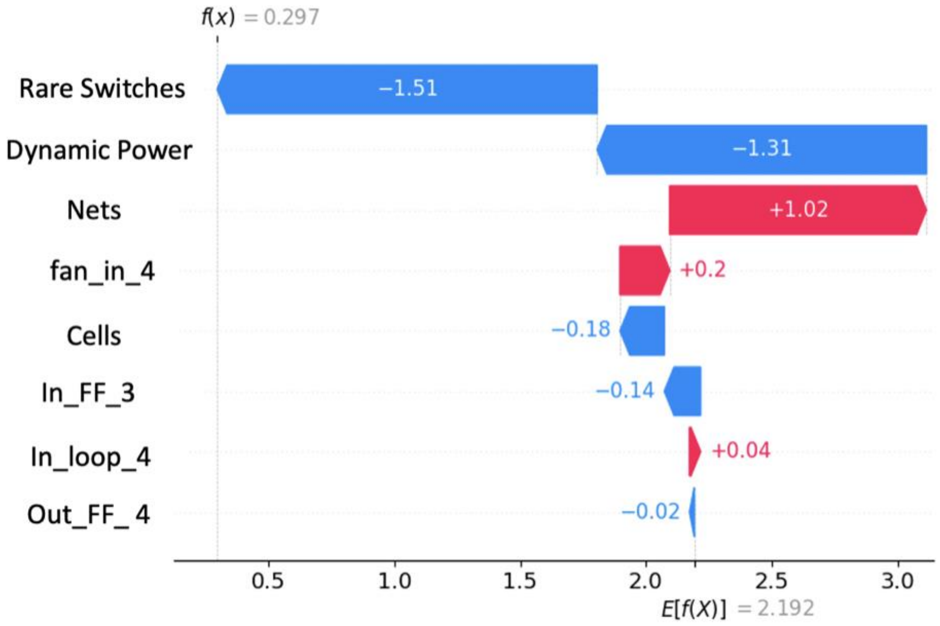
	RFC				CNN				TGRL				SEB (Proposed Approach)				
Bench	**Acc**	**Rec**	**Pre**	**F1**	**Acc**	**Rec**	**Pre**	**F1**	**Acc**	**Rec**	**Prec**	**F1**	**Acc**	**Rec**	**Pre**	**F1**	**impr/TGRL**
c2670	83.1%	0.87	0.89	0.88	90.7%	0.90	0.90	0.90	96.2%	0.97	0.94	0.96	100.0%	1.0	1.0	1.0	3.8%
c5315	75.4%	0.78	0.83	0.81	87.6%	0.85	0.88	0.86	91.4%	0.92	0.91	0.92	100.0%	1.0	1.0	1.0	8.6%
c6288	64.5%	0.68	0.63	0.65	80.5%	0.85	0.79	0.85	88.8%	0.89	0.85	0.87	99.8%	0.99	0.99	0.99	11.0%
c7552	77.2%	0.74	0.79	0.76	84.9%	0.81	0.86	0.83	91.2%	0.89	0.91	0.90	100.0%	1.0	1.0	1.0	8.8%
s13207	78.5%	0.77	0.79	0.78	90.4%	0.91	0.92	0.92	95.6%	0.94	0.95	0.95	100.0%	1.0	1.0	1.0	4.4%
s15850	68.8%	0.65	0.73	0.68	83.0%	0.75	0.86	0.80	92.7%	0.93	0.95	0.94	99.8%	0.99	0.99	0.99	7.1%
s35932	73.1%	0.78	0.53	0.63	75.5%	0.72	0.76	0.74	83.6%	0.88	0.81	0.84	99.9%	0.97	0.99	0.98	16.3%
AES-T100	85.9%	0.93	0.79	0.85	89.2%	0.84	0.86	0.85	96.9%	0.97	0.97	0.97	100.0%	1.0	1.0	1.0	3.1%
AES-T200	79.3%	0.88	0.73	0.79	90.2%	0.85	0.92	0.88	95.8%	0.98	0.91	0.94	99.9%	1.0	1.0	1.0	4.1%
AES-T1000	67.2%	0.84	0.63	0.72	80.5%	0.72	0.76	0.74	90.1%	0.95	0.95	0.95	99.9%	1.0	1.0	1.0	9.8%
Average	**75.3 %**	**0.79**	**0.73**	**0.76**	**85.3%**	**0.82**	**0.85**	**0.83**	**92.2%**	**0.93**	**0.91**	**0.92**	**99.9%**	**0.99**	**1.0**	**1.0**	**6.1**

Time Efficiency:

Methods	**RFC**	**TGRL**	**CNN**	**SEB**	**SEB/RFC**	**SEB/TGRL**	**SEB/CNN**
Training	4430	30019	10396	1767	2.6x	17.4x	5.8x
Testing	1284	2014	559	1339	2.3x	3.6x	3.6x
Total	**5714**	**31033**	**11735**	**2326**	**2.5x**	**13.4x**	**5.1x**

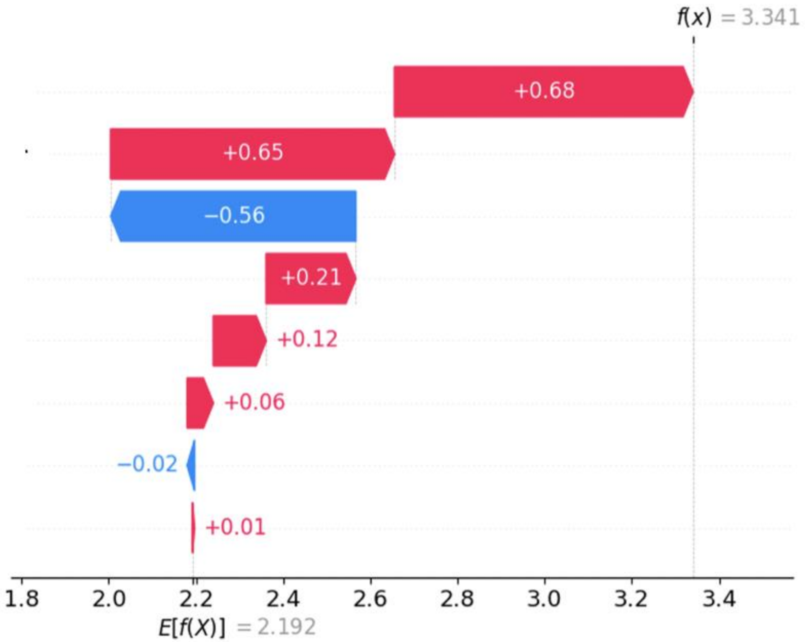
Explainability Evaluation

Example of S13207



(a)

Trojan Free



(b)

Trojan Implanted

Outline

- Introduction
- Related Work
- Proposed Method
- Experimental Results
- Conclusion

Conclusion

- Hardware Trojan attacks are dangerous threat to systems.
- AI/ML techniques have serious limitations.
- We propose an efficient and explainable detection scheme based on Shapley ensemble boosting.
  - Efficient training of a sequence of lightweight model
  - Result Interpretation using Shapley Values
  - Ensemble prediction for better performance
- Our approach significantly improves detection efficiency (24.6%) compared to state-of-the-art techniques.

Questions?