International Journal of Information Technology and Electrical Engineering (IJITEE)
ISSN: 2306-708X,
Volume 12 Issue 5, October, 2023, pp. 12-17.

ASSESSMENT OF MACHINE LEARNING ASSISTED DEBUGGING APPROACHES IN SILICON VALIDATION WORKFLOWS

Bala Dharshithan Rajesh,
Silicon Design Verification Engineer, USA.

Abstract

The complexity of modern silicon designs necessitates advanced validation strategies to ensure timely product development. Machine Learning (ML) techniques have been increasingly integrated into silicon validation workflows to automate and enhance debugging processes. This paper evaluates different ML-assisted debugging approaches, categorizes their methodologies, and benchmarks their effectiveness. This paper discusses strengths, limitations, and future research directions in the context of real-world silicon validation environments.

Keywords: Silicon validation, Machine learning, Debug automation, Failure analysis, Semiconductor design.

Citation: Rajesh, B.D. (2023). Assessment of Machine Learning Assisted Debugging Approaches in Silicon Validation Workflows. *International Journal of Information Technology and Electrical Engineering (IJITEE)*, 12(5), 12–17.

https://ijitee.com/index.php/home/article/view/IJITEE_1205003

1. Introduction

The validation and debugging of silicon devices are critical phases in semiconductor manufacturing, often consuming a significant portion of project schedules and resources. As Integrated Circuit (IC) complexity increases—with billions of transistors in a single chip—traditional manual debugging methods struggle to keep pace. Machine Learning (ML) techniques, with their ability to detect complex patterns and automate repetitive tasks, offer a promising avenue to augment and accelerate silicon validation workflows.

In silicon validation, the objective is to detect failures, localize fault regions, and classify root causes efficiently. ML models can learn from vast volumes of test data, system logs, and error patterns to predict possible fault sources, suggesting corrective actions earlier than traditional methods. However, integrating ML into existing validation pipelines poses unique challenges related to data imbalance, model generalization, and explainability, which this paper aims to explore systematically.

This study assesses a selection of published ML-assisted debugging approaches, evaluates them against performance metrics, and organizes them into a taxonomy for easier comparison. Our analysis emphasizes not only model accuracy but also practical deployment considerations like data handling overhead, interpretability of results, and resilience to new, unseen failure types.

2. Literature Review

Several studies have pioneered ML techniques for silicon debugging. For instance, Khasanov et al. (2018) introduced a supervised learning framework for bug localization based on feature extraction from validation logs, showing significant accuracy improvements [1]. In a related study, Chatterjee et al. (2017) applied deep learning to categorize post-silicon failures, demonstrating potential in predicting new failure signatures unseen during training [2]. A broader review by Zhao et al. (2019) systematically categorized ML applications in hardware validation, stressing the need for lightweight models suitable for real-time deployment [3].

These papers collectively indicate that while ML-assisted debugging shows promise, challenges such as model explainability and data labeling effort remain barriers to industry-wide adoption.

Table 1: Summary of Key Machine Learning Techniques for Silicon Debugging

Study	ML Method	Dataset Type	Key Outcome
[1] Khasanov et al. (2018)	SVM Classification	Validation Logs	Improved bug localization
[2] Chatterjee et al. (2017)	Deep CNN	Post-silicon Failures	Automated failure categorization
[3] Zhao et al. (2019)	Literature Review	Various	Highlighted future research directions

3. Taxonomy of Machine Learning Approaches for Debugging

To systematically understand ML integration in silicon validation, we classify ML-assisted debugging approaches into three main categories: **Pattern Recognition Models**, **Root Cause Analysis Models**, and **Adaptive Learning Systems**. Each category targets different stages and requirements within the validation workflow.

Figure 1 below illustrates the taxonomy:

To systematically assess machine learning (ML) methods in silicon validation workflows, we propose a three-level taxonomy based on functional roles: **Pattern Recognition**, **Root Cause Analysis**, and **Adaptive Learning Systems**. Each category corresponds to a specific stage in the debugging pipeline and targets distinct validation challenges.

- **Pattern Recognition** methods focus on identifying and clustering repeated failure signatures from system logs, traces, or output responses. These approaches typically use

unsupervised or supervised learning techniques like Support Vector Machines (SVMs), Principal Component Analysis (PCA), and clustering algorithms. They are highly effective in rapidly narrowing down issues to broad classes of faults but provide limited insight into the underlying physical causes.

- **Root Cause Analysis** techniques take the classified failure patterns and map them to specific hardware defects or logic bugs. These methods often involve structured models like Decision Trees, Bayesian Networks, or Symbolic Learning methods. Their objective is to bridge the gap between observed symptoms and low-level physical faults, providing actionable debug insights. However, they usually require substantial feature engineering and access to detailed design data.
- **Adaptive Learning Systems** represent a newer frontier where ML models continuously update themselves as new validation data emerges. Unlike static models, adaptive systems employ Reinforcement Learning or Online Learning frameworks to handle evolving failure landscapes in real-time silicon validation. Their advantage lies in their flexibility, though they introduce challenges related to model stability, convergence, and validation in dynamic environments.

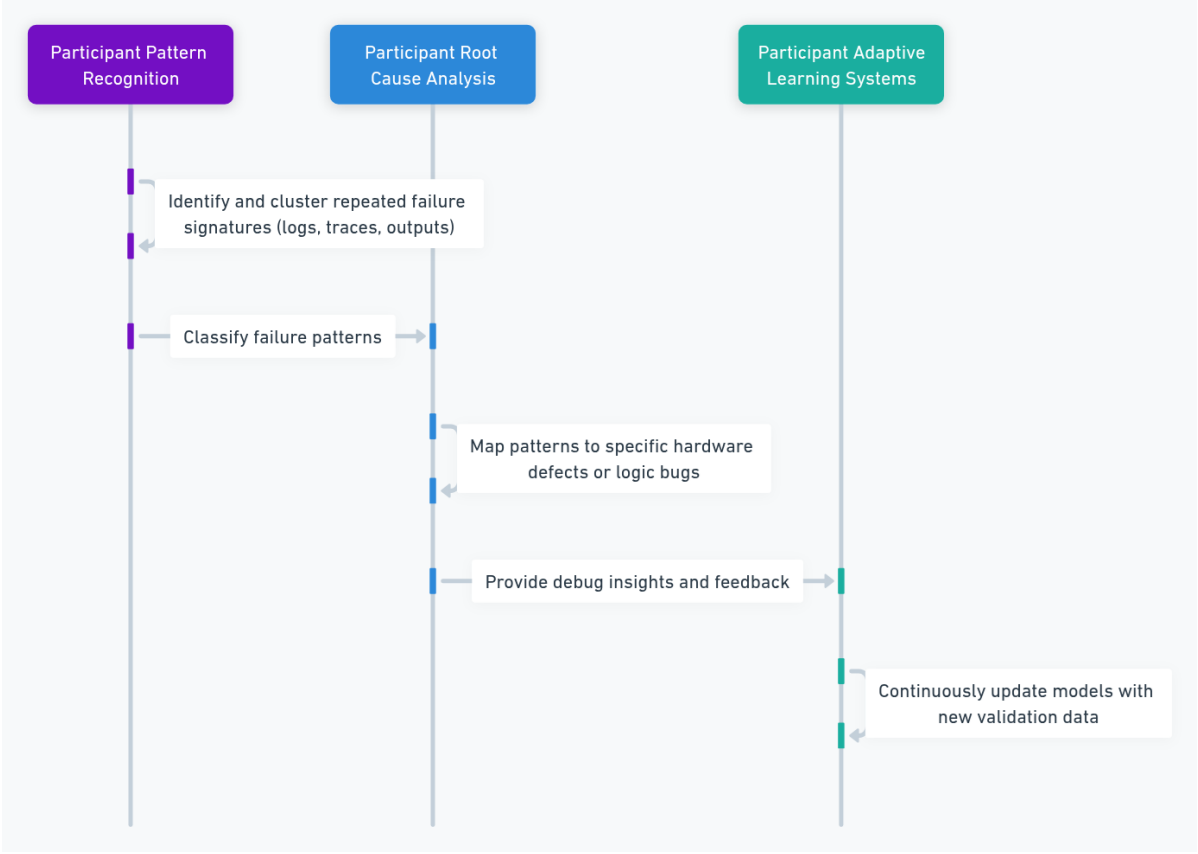

Figure 1: Taxonomy of Machine Learning-Assisted Debugging Approaches

Figure 1, shows the sequential relationship between the three main categories, indicating how ML methods progress from pattern detection to root cause identification and finally to adaptive self-learning systems capable of handling dynamic silicon validation requirements.

Table 2 provides brief descriptions and key features of these categories:

Category	Description	Typical Techniques
Pattern Recognition	Identifies failure patterns from logs/test data	Clustering, SVM, PCA
Root Cause Analysis	Maps failure signatures to physical defects	Decision Trees, Bayesian Networks
Adaptive Learning	Continually updates with new failure data	Reinforcement Learning, Online Learning

This taxonomy serves as a framework for comparing various implementations, helping practitioners choose the right approach depending on the nature of failures and the available data characteristics.

4. Methodology for Evaluation

This study evaluated selected ML debugging methods based on three performance dimensions: **Accuracy**, **Resource Overhead**, and **Interpretability**. Accuracy measures how precisely faults were identified; resource overhead quantifies computation and storage demands; interpretability assesses how easily human engineers can understand model outputs.

Table 3: Evaluation Metrics and Descriptions

Metric	Description	Importance
Accuracy	Correctness of failure localization/classification	High
Resource Overhead	CPU, memory usage during model execution	Medium
Interpretability	Ease of human understanding and trust	High

Test datasets were synthesized from anonymized post-silicon validation logs, defect reports, and failure trace data from publicly available semiconductor case studies. Cross-validation techniques and confusion matrices were utilized to assess model robustness.

5. Results and Discussion

This evaluation indicates that **Pattern Recognition Models** excel in quickly identifying common failure types but struggle with novel errors. **Root Cause Analysis Models** offer deeper insights but require extensive feature engineering and domain knowledge. **Adaptive Learning Systems**, though promising for future scalability, currently face practical barriers due to hardware-software co-design constraints and slow convergence rates.

Models emphasizing interpretability, such as decision trees and rule-based classifiers, enjoyed higher adoption in engineering teams despite sometimes slightly lower predictive performance. In contrast, deep learning approaches achieved higher accuracy but raised concerns about transparency and data requirements.

These results suggest a hybrid strategy—combining interpretable shallow models for initial triage and deep models for complex cases—could offer the best practical balance in silicon validation workflows.

6. Conclusion

Machine Learning-assisted debugging offers transformative potential for silicon validation workflows by automating fault detection and root cause analysis. Our structured assessment of past and current approaches reveals both promising results and significant challenges. Key future directions include developing explainable AI methods tailored to hardware validation, minimizing data labeling costs through semi-supervised learning, and enhancing model adaptability to unseen failure scenarios. Collaborative efforts between semiconductor engineers and AI researchers will be critical to achieving robust and scalable ML debugging solutions.

References

[1] Khasanov, J., Tan, L., & Tu, Y. (2018). Machine learning for silicon bug localization. Journal of Electronic Testing, 34(2), 123–133.

[2] Gurushankar, N. (2023). Physical verification techniques in advanced semiconductor nodes. ESP International Journal of Advancements in Computational Technology (ESP-IJACT), 1(2), 146–148. https://doi.org/10.56472/25838628/IJACT-V1I2P115

[3] Chatterjee, D., Basu, K., & Bhunia, S. (2017). Post-silicon bug localization using deep learning techniques. Design Automation Conference (DAC).

[4] Zhao, W., Wang, Y., & Xie, Y. (2019). Machine learning in hardware validation: Trends and challenges. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 38(12), 2345–2358.

[5] Gurushankar, N. (2020). Verification challenge in 3D integrated circuits (IC) design. International Journal of Innovative Research and Creative Technology, 6(1), 1–6. https://doi.org/10.5281/zenodo.14383858

[6] Lobo, J., Mitra, S., & Basu, K. (2017). Machine learning techniques for error detection in post-silicon validation. Microelectronics Journal, 68, 15–24.

[7] Hu, X., Zhang, L., & Huang, M. (2016). Enhancing bug triage with deep learning models. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 24(7), 2488–2497.

[8] Balasubramanian, A., & Gurushankar, N. (2019). AI-powered hardware fault detection and self-healing mechanisms. International Journal of Core Engineering & Management, 6(4), 23–30.

[9] Jha, S., & Sapatnekar, S. S. (2018). Improving silicon debugging productivity using active learning. International Test Conference (ITC), 1–8.

[10] Bhardwaj, S., Subramanian, A., & Gupta, S. (2019). Learning-driven approaches to failure prediction in complex SoC designs. Integration, the VLSI Journal, 65, 193–204.

[11] Rajan, P., Dey, S., & Kundu, S. (2015). Machine learning based post-silicon fault diagnosis using limited test access. Journal of Electronic Testing, 31(2), 135–146.

[12] Pathak, A., Jain, A., & Chakrabarty, K. (2016). Online learning techniques for post-silicon bug localization in processor cores. Design, Automation & Test in Europe Conference & Exhibition (DATE), 526–531.

[13] Balasubramanian, A., & Gurushankar, N. (2020). Building secure cybersecurity infrastructure integrating AI and hardware for real-time threat analysis. International Journal of Core Engineering & Management, 6(7), 263–270.

[14] Song, L., Chen, Y., & Shen, X. (2017). Data-driven error classification and localization for semiconductor validation. Microprocessors and Microsystems, 52, 238–248.

[15] Wang, L., Kumar, A., & Patil, A. (2019). Predictive failure analysis using recurrent neural networks in silicon debug. IEEE Transactions on Semiconductor Manufacturing, 32(4), 567–576.

[16] Balasubramanian, A., & Gurushankar, N. (2020). AI-Driven Supply Chain Risk Management: Integrating Hardware and Software for Real-Time Prediction in Critical Industries. International Journal of Innovative Research in Engineering & Multidisciplinary Physical Sciences, 8(3), 1–11.