Informed Repair of Deep Neural Networks

Akshat Adsule
University of California, Davis
aadsule@ucdavis.edu

Darroll Saddi
University of California, Davis
dwsaddi@ucdavis.edu

Suyash Goel
University of California, Davis
sngoel@ucdavis.edu

1 Introduction

Deep neural networks (DNNs) have become increasingly prevalent in modern applications. DNNs have seen use in virtually every field from air traffic control to self-driving vehicles. However, these models are not infallible and do produce mistakes, which could prove disastrous given the model's application.

Recent research [1–3] has explored methods of repairing DNNs once incorrect inputs are identified. Repair techniques have the goal of correcting the model's behavior on a specified set of inputs, often referred to as the repair set. The goal of many existing techniques is to adjust network weights and biases while satisfying the conditions of: (i) provable, (ii) generalizing, (iii) architecture-preserving, (iv) scalable, and (v) local repair. In other words, these techniques typically aim to minimally adjust the network's parameters to correct its behavior on the specified repair set, often while providing formal guarantees on the outcome for those inputs or related input regions. While methods that satisfy some or even all of these conditions exist, most require user intervention to select the repair set and which layers of the network should be affected. Thus, in practice, applying these repair methods reveals significant ambiguities that can affect the quality and efficiency of the repair.

Take, for example, APRNN, the method proposed in [3] that offers provable, architecture-preserving repair over specified input regions (V-polytopes). APRNN achieves all the previously stated conditions (i-v), and works by provably repairing the network's weights and biases up to a chosen layer. This involves modifying network weights starting primarily at that chosen layer and adjusting biases in subsequent layers.

While effective, APRNN includes critical ambiguities in practice, mainly as a result of the user needing to select the starting layer or affected layers for weight and bias adjustment. The paper itself doesn't prescribe how to choose affected layer, and this choice can significantly impact the repair's effectiveness, efficiency, and efficacy. The choice of repair set, which defines the repair polytope, is also left to the user – fundamentally determining the target behavior of the repair. The composition and scope of this set influence the repair outcome and how well the fix generalizes to similar, unseen inputs. This situation is not ideal because these choices can be arbitrary and greatly impact the quality of the repair.

This paper aims to investigate and develop heuristics to guide the DNN repair process, specifically addressing the ambiguities highlighted above in methods like APRNN. By providing data-driven or structurally-informed ways to make these choices, we seek to enable more efficient and informed repairs of DNNs. We propose to explore and evaluate various heuristics, including but not limited to:

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

Layer Selection Heuristics:

Activation-Based Selecting the start layer based on metrics calculated across the repair set, such as the layer exhibiting the highest average activation magnitude or the highest variance in activations.

Gradient/Sensitivity-Based Identifying layers where parameters show the most sensitivity (e.g., largest gradient norms) with respect to the inputs in the repair set, indicating layers most influential on the incorrect output.

Change-Based (for Adversarial Inputs) Selecting the layer whose activations or feature representations changed most drastically between the original and adversarial inputs in the repair set.

Feature-Similarity Based Choosing a layer where the internal representations of the inputs within the repair set are most similar, suggesting a point of unified processing relevant to the required fix.

Layer Type/Position Simple heuristics such as always choosing the first fully-connected layer after convolutional blocks, or the penultimate layer.

Brute Force Exhaustively evaluating all layers and selecting the one that yields the best repair outcome, though this is computationally expensive.

Repair Set Analysis

Diversity Evaluating the diversity of the repair set, such as the number of unique classes or the distribution of inputs across the input space.

Concentration Analyzing the concentration of points defining the polytope, such as the number of points needed to define a convex hull or the dimensionality of the convex hull.

Size Considering the size of the repair set, including the number of points and the dimensionality of the input space.

We plan to also evaluate these heuristics on the architecture of [] trained on the [] dataset. We also hope to use the following metrics to evaluate the heuristics:

Evaluation Metrics:

Repair Success Rate The percentage of inputs in the repair set for which the model produces the correct output after repair.

Generalization The model's performance on unseen inputs similar to those in the repair set, measured by accuracy or other relevant metrics.

Locality The extent to which the repair minimally affects the model's behavior on inputs outside the repair set, often quantified by changes in the model's predictions or parameter values.

Efficiency The computational cost of the repair process, including time and resources required.

Scalability The ability of the repair method to handle larger models or repair sets without significant degradation in performance or efficiency. [May not be able to measure this in this project]

This project enhances AI trustworthiness by making the crucial process of model repair—itself a method for increasing trust after failures—more efficient and informed. Current repair techniques can involve arbitrary choices, leading to unpredictable outcomes. By developing heuristics to guide decisions within the repair process, such as selecting the optimal network layer for modification, we enable more systematic, reliable, and effective correction of identified flaws. This ultimately increases confidence in the robustness and safety of AI systems by ensuring that necessary fixes are applied more predictably and with a better understanding of their potential impact.

References

[1] Stephanie Nawas, Zhe Tao, and Aditya V. Thakur. Provable repair of vision transformers. In Guy Avni, Mirco Giacobbe, Taylor T. Johnson, Guy Katz, Anna Lukina, Nina Narodytska, and Christian Schilling, editors, *AI Verification*, pages 156–178. Springer Nature Switzerland, 2024. ISBN 978-3-031-65112-0. doi: 10.1007/978-3-031-65112-0_8.

[2] Matthew Sotoudeh and Aditya V. Thakur. Provable repair of deep neural networks. In *Proceedings of the 42nd ACM SIGPLAN International Conference on Programming Language Design and Implementation*, PLDI 2021, pages 588–603. Association for Computing Machinery, 2021. ISBN 978-1-4503-8391-2. doi: 10.1145/3453483.3454064. URL https://dl.acm.org/doi/10.1145/3453483.3454064.

[3] Zhe Tao, Stephanie Nawas, Jacqueline Mitchell, and Aditya V. Thakur. Architecture-preserving provable repair of deep neural networks. 7:124:443–124:467, 2023. doi: 10.1145/3591238. URL https://dl.acm.org/doi/10.1145/3591238.