Chapter 8
Applications of Artificial Intelligence for Enhanced Bug Detection in Software Development

Yahya Najib Hamood Al-Shameri
Kocaeli University, Turkey

ABSTRACT

As technology advances at breakneck speed, reliance on software grows exponentially. This growth has given rise to multiple issues encountered during software development, including increased occurrences of software bugs. The conventional methods previously used by developers, such as manual code checking or testing, may involve greater chances of human error and extended timelines, resulting in more significant expense budgets towards revenue leakage due to late deliveries or failed outcomes. This chapter per the authors aim to explore how artificial intelligence (AI) solutions can identify potential vulnerabilities more accurately than before. By utilizing machine learning algorithms, deep learning, and natural language processing, AI-based bug detection not only enhances precision and efficiency but also saves costs on the development process, ultimately giving better software quality and reduced monetary costs. This chapter per the authors present some applications of AI techniques in bugs detection to select bugs accurately and suggest possible optimizations for the buggy codes.

DOI: 10.4018/979-8-3693-6745-2.ch008

1. INTRODUCTION

In the contemporary environment of software development, the complexity and scale of projects have increased, rendering traditional bug detection methods, such as manual code reviews and conventional testing, increasingly inadequate. The limitations of these methods become particularly evident in large-scale software systems, where the volume of code surpasses manual review efforts, leading to overlooked bugs and higher costs associated with late-stage bug fixes. Errors during software development are inevitable, and it is crucial to detect and fix them quickly to ensure reliable software. Traditional methods like code reviews and testing can be time-consuming and not entirely reliable. This has sparked significant interest in leveraging advanced techniques to enhance bug detection processes. Modern techniques, especially those involving artificial intelligence, offer promising solutions to the challenges of bug detection. By analyzing large amounts of code and identifying patterns that could indicate potential errors, these techniques can significantly reduce the time and effort required for manual inspections. Automated tools can detect and classify bugs, improving both the efficiency and accuracy of the process. (Siswantoro & Yuhana, 2023; Tang et al., 2024)

Modern techniques, particularly those involving deep learning and natural language processing, hold great promise for improving the efficiency and reliability of bug detection. For instance, supervised learning models can analyze extensive codebases to identify patterns indicative of potential bugs that might escape human reviewers (Elshamy et al., 2023; Wang, 2023). These models can classify bugs based on historical data, allowing for the proactive identification of error-prone modules before they are subjected to testing (Bala et al., 2023). Additionally, approaches such as Generative Adversarial Networks (GANs) have been explored to enhance bug prediction capabilities, demonstrating advanced accuracy and speed in identifying software bugs (Kumar & Venkatesan, 2020; Song et al., 2024). This shift towards automated and intelligent bug detection not only reduces the time and effort required for manual inspections but also enhances the overall quality of software products. The integration of artificial intelligence (AI) techniques into bug detection processes provides a more nuanced understanding of software quality. AI can effectively address the complexities of modern software systems by detecting subtle and intricate bugs that traditional methods might overlook (Siswantoro & Yuhana, 2023; Tang et al., 2024). For example, hybrid models that combine various machine learning algorithms have been developed to optimize bug prediction, achieving higher accuracy and efficiency in identifying errors (Bala et al., 2023; Zheng et al., 2021). Moreover, the use of data mining techniques has facilitated the classification and prioritization of bugs, enabling developers to focus their efforts on the most critical