Research Aim: The research presented at the 2024 IEEE International Conference on Software Analysis, Evolution, and Reengineering focuses on addressing the challenge of ensuring software quality in deep learning (DL) libraries that undergo rapid and frequent changes. The study introduces CEDAR, a framework designed for continuous testing of DL libraries, which integrates advanced DL testing tools to enhance bug detection in continuously evolving software environments. The goal is to demonstrate the efficacy of CEDAR in identifying bugs, reducing the latency in bug detection, and ultimately supporting the maintenance of code quality in rapidly updated DL projects.

Methodology: CEDAR employs a combination of two state-of-the-art DL testing approaches, DocTer and EAGLE, which utilize fuzz testing and differential testing respectively. This integration is designed to test popular DL libraries such as PyTorch and TensorFlow against both regression and newly introduced bugs. The testing process is automated to operate in continuous integration environments, providing nightly tests on the latest library versions. The framework is optimized with strategies aimed at minimizing testing time and computational resources, including test case reduction, parallel testing processes, and redundancy removal.

Results: The application of CEDAR across 20 different versions of PyTorch and TensorFlow has led to the detection of 83 bugs in 140 APIs, out of which 23 were previously unknown with 21 confirmed or fixed by the developers. The findings underscore CEDAR's capacity to significantly reduce the bug detection latency by an average of 338.6 days. Moreover, the study highlights that CEDAR efficiently identifies bugs not only in new code but also in existing codebases that were part of prior versions, showcasing its effectiveness in continuous regression testing and in handling changes introduced in nightly builds.

Implications for Research and Practice: The development and deployment of CEDAR offer significant implications for both the research community and software development practices. For researchers, CEDAR provides a proven framework that can be adapted or expanded to include new testing methodologies or applied to different domains within software engineering. Practically, integrating such a tool within DL library development pipelines can drastically enhance the reliability of software releases by ensuring quicker identification and resolution of bugs. This is particularly crucial in domains where DL applications are critical, such as autonomous driving and medical diagnostics, where software faults can have serious repercussions. Overall, CEDAR's development highlights the benefits of continuous testing strategies in modern software engineering environments where software updates are frequent and need to maintain high reliability and quality.