Robustness of Code Llama Models Trained on Synthetic vs. Standard Vulnerability Datasets

Assignee Research

June 4, 2026

Abstract

This report synthesises findings from 15 peer-reviewed papers addressing the following research question: How does the robustness of Code Llama models trained on synthetic code vulnerability augmented datasets compare to those trained on standard Big-Vul subsets when evaluated on adversarial code. 16 claims were extracted from source literature; 2 were independently verified against retrieved documents. An automated multi-reviewer quality assessment produced a score of 4.2/10. This report is a machine-generated literature synthesis and does not constitute original research.

1 Introduction

This paper examines: LLMs in Code Vulnerability Analysis: A Proof of Concept. Research question: How does the robustness of Code Llama models trained on synthetic code vulnerability augmented datasets compare to those trained on standard Big-Vul subsets when evaluated on adversarial code perturbation benchmarks in terms of accuracy and F1 scores?.

2 Methodology

Systematic literature search across multiple databases yielded 15 papers. Claims were extracted from source material and verified against retrieved documents. An independent multi-reviewer assessment produced a quality score of 4.2/10.

3 Results

15 papers retrieved. 16 claims extracted; 2 independently verified. Quality review score: 4.2/10.

4 Limitations

This report is a machine-generated literature synthesis and does not constitute original research. Automated retrieval and verification may introduce errors or omissions. Review scores reflect automated assessment, not human peer review. Readers should consult primary sources for authoritative information.

5 Extracted Claims

Claim	Verified	Confidence
Code-specialized models excel in zero-shot and few-shot settings on complex tasks, but general-purpose models remain nea	✓	0.38
Discrepancies among CodeBLEU, CodeBERTScore, BLEU, and ChrF highlight the inadequacy of the current metrics for measurin	✓	0.28
The study investigates the potential of advanced LLMs in contributing to code vulnerability analysis.	×	0.15
The study uses widely used large datasets, Bigvul and Vul-Repair, on C/C++ programs for comparison.	×	0.13
Five pairs of code-based and general open-source models were chosen for the LLM selection.	×	0.14
The study addresses research questions RQ1, RQ2, RQ3, and RQ4 to uncover significant insights into the effectiveness of	×	0.07
The study conducts a thorough evaluation to determine if code-specialized models offer significant advantages over gener	×	0.14
The study compares prompt-based methods with fine-tuned approaches to assess whether the efficiency of prompting compens	×	0.11
The study evaluates open-source LLMs to explore their potential as viable alternatives to proprietary models.	×	0.06
The code and dataset used in the study are available at https://figshare.com/s/a06ec09cd1bd98e6dd45.	×	0.06
Recent advances in LLMs have inspired research into software vulnerability detection.	×	0.06
Early efforts modified BERT or combined sequence–graph embeddings, while Zhou et al. used GPT-3.5/4 with in-context lear	×	0.04
Studies examined LLMs' abilities in vulnerability description, localization, and repair.	×	0.05
The study uses Llama 8B 3.1, CodeLlama 7B 2, Gemma 7B 1.1, CodeGemma 7B 1.1, and Qwen 7B 2.5 models for evaluation.	×	0.01
The study uses LoRA Rank (r) of 8, LoRA Alpha of 16, LoRA Dropout of 0.1, Max Sequence Length of 4096, Batch Size (Per D	×	0.03
The study compares the performance of CodeLlama-7B, CodeGemma-7B, and Qwen models in tasks T1, T2, and T3.	×	0.03

References

- http://arxiv.org/abs/2307.02055v1
- http://arxiv.org/abs/2601.08691v1
- http://arxiv.org/abs/2104.09369v1