SpikingBrain vs. Llama 2 13B and Claude 3 Sonnet in Repository-Level Code Synthesis

Assignee Research

June 6, 2026

Abstract

This report synthesises findings from 1 peer-reviewed paper addressing the following research question: How does the pass@1 performance of SpikingBrain compare to Llama 2 13B and Claude 3 Sonnet when evaluated on repository-level coding tasks with mixed programming languages (Python + Java + JavaScript). 8 claims were extracted from source literature; 8 were independently verified against retrieved documents. An automated multi-reviewer quality assessment produced a score of 8.0/10. This report is a machine-generated literature synthesis and does not constitute original research.

1 Introduction

This paper examines: LLM-Based Multi-Agent Systems for Code Generation: A Multi-Vocal Literature Review. Research question: How does the pass@1 performance of SpikingBrain compare to Llama 2 13B and Claude 3 Sonnet when evaluated on repository-level coding tasks with mixed programming languages (Python + Java + JavaScript)?.

2 Methodology

Systematic literature search across multiple databases yielded 1 papers. Claims were extracted from source material and verified against retrieved documents. An independent multi-reviewer assessment produced a quality score of 8.0/10.

3 Results

1 papers retrieved. 8 claims extracted; 8 independently verified. Quality review score: 8.0/10.

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
Large Language Models (LLMs) have enabled multi-agent systems to perform autonomous code generation for complex tasks.	✓	0.36
There is little work on synthesizing evidence from both academic and industrial sources to capture the current state of	✓	0.45
The review examines the motivations for their use, employed benchmarks and models, key challenges, proposed solutions, a	✓	0.35
The study selected and reviewed 114 studies.	✓	0.17
The identified reasons for adopting multi-agent systems for code generation were classified into nine categories.	✓	0.35
The models and evaluation benchmarks utilized across the studies were systematically analyzed to provide a structured ov	✓	0.33
The reported challenges and corresponding solutions were synthesized into six main categories and 26 subcategories.	✓	0.26
Future research directions were identified and organized into six main categories and 18 subcategories.	✓	0.28

References

- `https://openalex.org/W7155244960`