CodeT5 Performance on MBPP with Cross-Language Fine-Tuning and Pretraining Scaling

Assignee Research

June 7, 2026

Abstract

This report synthesises findings from 12 peer-reviewed papers addressing the following research question: How does varying the size of the code pretraining dataset influence CodeT5's performance on the MBPP benchmark when fine-tuned on Python and evaluated on Rust, compared to a model pretrained only on. 8 claims were extracted from source literature; 1 was independently verified against retrieved documents. An automated multi-reviewer quality assessment produced a score of 4.7/10. This report is a machine-generated literature synthesis and does not constitute original research.

1 Introduction

This paper examines: Procedural Pretraining: Warming Up Language Models with Abstract Data. Research question: How does varying the size of the code pretraining dataset influence CodeT5's performance on the MBPP benchmark when fine-tuned on Python and evaluated on Rust, compared to a model pretrained only on natural language?.

2 Methodology

Systematic literature search across multiple databases yielded 12 papers. Claims were extracted from source material and verified against retrieved documents. An independent multi-reviewer assessment produced a quality score of 4.7/10.

3 Results

12 papers retrieved. 8 claims extracted; 1 independently verified. Quality review score: 4.7/10.

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
Procedural pretraining improves performance and accelerates language model pretraining.	×	0.08
Procedural pretraining can be complementary to standard pretraining datasets, improving performance with as little as 0.	×	0.09
Procedural data enables models to reach the same loss with 55% of the original data on C4, 67% on CODEPARROT, and 86% on	✓	0.23
Procedural pretraining gains persist on downstream language, code generation, and commonsense reasoning tasks.	×	0.10
Different types of procedural pretraining facilitate learning different algorithmic skills.	×	0.10
The pretrained information is localized in specific layers (attention vs. MLPs).	×	0.04
Procedural pretraining improves performance on diverse domains, including natural language, code, and informal mathemati	×	0.15
Procedural pretraining has been validated across different model sizes (up to 1.3B parameters) and data sizes (up to 10.	×	0.08

References

- http://arxiv.org/abs/1812.10860v5
- http://arxiv.org/abs/2601.21725v2
- http://arxiv.org/abs/2306.06371v1