Hyunjae Suh

Irvine, CA   hyunjas@uci.edu   hyunjas.github.io   hyunjae-suh-607119146

Summary

Ph.D. student in Software Engineering with a research focus on Generative AI for code generation. Specializes in analyzing and improving Large Language Model (LLM)-generated source code through advanced prompting strategies, empirical evaluation, and human-AI comparisons. Experienced in leveraging LLMs for software development tasks, evaluating the quality of generated code, and conducting comparative analyses to understand the capabilities and limitations of Generative AI in real-world scenarios.

Education

University of California, Irvine — *Irvine, CA*
Ph.D. in Software Engineering — *Sep 2023 – Present*

Kookmin University — *Seoul, South Korea*
BS in Computer Science — *Mar 2017 – Aug 2023*

Experience

Ph.D. Researcher in AI and Software Engineering — *Irvine, CA*
University of California, Irvine — *Sep 2023 – Present*

- Conducting empirical research on the application of Large Language Models (LLMs) in software engineering, with a focus on code generation and prompt engineering.
- Investigating the characteristics of LLM-generated code, including accessibility, quality, and similarity to human-written implementations.
- Designing and evaluating advanced prompting strategies (e.g., Few-Shot, Self-Criticism, Multi-Agent Debate, ReAct) to enhance the accessibility of AI-generated source code.
- Leveraging accessibility evaluation tools (IBM Equal Access, AChecker) to assess LLM-generated code compliance with WCAG 2.1.

Graduate Research Assistant – AI & Software Engineering — *Remote*
eBay — *Aug 2023 – Dec 2023*

- Developed and fine-tuned open-source LLMs for automated commit message generation, improving software documentation efficiency.
- Investigated the effectiveness of prompt engineering techniques to optimize commit message accuracy, clarity, and contextual relevance.
- Conducted comparative analysis of open-source LLMs to evaluate their performance in real-world software engineering workflows.

Publications

An Empirical Study on Automatically Detecting AI-Generated Source Code: How Far Are We?
Hyunjae Suh, Mahan Tafreshipour, Jiawei Li, Adithya Bhattiprolu, Iftekhar Ahmed
Accepted at the 47th IEEE/ACM International Conference on Software Engineering [ICSE 2025] (IEEE Link)

Human or LLM? A Comparative Study on Accessible Code Generation Capability
Hyunjae Suh, Mahan Tafreshipour, Sam Malek, Iftekhar Ahmed
https://arxiv.org/abs/2503.15885

Does the Order of Fine-tuning Matter and Why?
Qihong Chen, Jiawei Li, **Hyunjae Suh**, Lianghao Jiang, Zheng Zhou, Jingze Chen, Jiri Gesi, Iftekhar Ahmed
https://arxiv.org/abs/2410.02915

Projects

Detection of LLM-generated Source Code *(Research accepted at ICSE 2025)*

- Designed and implemented techniques for detecting LLM-generated source code by leveraging fine-tuned LLMs, code embeddings, and feature-based analysis.
- Achieved 82.55 F1-score, demonstrating state-of-the-art performance in distinguishing LLM-generated from human-written code.
- Analyzed key features of LLM-generated code, including syntax patterns, and semantic coherence, to improve detection accuracy.

Accessible Code Generation via Prompting in Large Language Models

- Investigated the effectiveness of advanced prompting techniques (Zero-Shot, Few-Shot, Self-Criticism) in improving the accessibility of code generated by state-of-the-art LLMs.
- Designed and implemented *FeedA11y*, a ReAct-based prompting framework that incorporates accessibility evaluation results as feedback during code generation for iterative refinement.
- Demonstrated that guided prompting significantly reduces accessibility violations in LLM-generated web applications, highlighting the importance of prompt engineering in AI-assisted coding.

The Impact of Fine-tuning Order on Language Models for Software Engineering

- Built fine-tuning pipelines for transformer-based LLMs, experimenting with different ordering of software engineering tasks (e.g., clone detection, defect detection, code translation, code repair).
- Analyzed the transfer learning effects of fine-tuning order on downstream task performance.
- Conducted quantitative experiments on different LLM architectures, measuring the impact of incremental fine-tuning for software engineering applications.

Teaching Experience

University of California, Irvine — *Irvine, CA*
Teaching Assistant — *Sep 2023 – Present*

- IN4MATX 115 - Software Testing, Analysis, and Quality Assurance
- ICS 10 - How Computers Work
- ICS 32 - Programming with Software Libraries
- CS 121 - Information Retrieval