# Gehao Zhang

CS PhD Student
University of Massachusetts Amherst
https://zhang-ge-hao.github.io/
https://github.com/zhang-ge-hao/
gehaozhang@umass.edu

## EDUCATION

**Ph.D. Student, Computer Science** Sep 2024 – Present
*University of Massachusetts Amherst*
Laboratory for Advanced Software Engineering Research (LASER)
Advisor: Juan Zhai

**B.Eng., Software Engineering** Sep 2016 – Jun 2020
*Northeastern University, Shenyang, China*

## WORK EXPERIENCE

***Research Intern*** May 2026 – Aug 2026
*NEC Laboratories America, Inc.*
Data Science & System Security Department
Responsibility: Improving coding agents on documentation-related tasks.

***Senior Machine Learning Engineer*** Jul 2020 – Aug 2024
*Ant Group (Alipay)*
AI-powered R&D Efficiency Team
Promoted from Machine Learning Engineer

## SELECTED RESEARCH

***PostcondBench: Benchmarking Correctness and Completeness in Formal Postcondition Inference*** [ACL Findings 2026]
*Advisor: Prof. Juan Zhai*
We introduce PostcondBench, a multilingual, repository-level benchmark for evaluating method-level postcondition generation, comprising 420 Python/Java tasks drawn from 121 real-world open-source projects, each paired with an expert-curated, high-quality ground-truth postcondition set. Using PostcondBench, we evaluate five state-of-the-art LLMs under three generation settings and observe a substantial correctness–completeness gap that is further exacerbated by repository-level dependencies and method complexity.

***Disappearing Ink: Obfuscation Breaks N-gram Code Watermarks in Theory and Practice*** [Preprint]
*Supervisors: Prof. Eugene Bagdasarian, Prof. Shiqing Ma, and Prof. Juan Zhai*
We model code obfuscation as a Markov random walk and prove that, under an empirically supported *distribution consistency* assumption, it nullifies N-gram watermark robustness. Across three schemes, two LLMs, two languages, four benchmarks, and four obfuscators, detection drops to near-random performance (AUROC $\approx 0.5$; none above 0.6).

***SpecProbe: Probing Intent–Implementation Misalignments with Generated Executable Postconditions*** [Submitted]
*Advisor: Prof. Juan Zhai*
SpecProbe is a repository-aware pipeline for generating behavioral probes: executable postconditions derived from natural-language documentation and checked during tests. Its 7B model, fine-tuned on $\sim$1.5K training examples, matches GPT-4o while requiring only 1/133 of the inference cost. On Defects4J, SpecProbe reveals 80/726 historical bugs; its probes locate the root-cause method while native failing-test traces miss it in 95% of those cases.

## PUBLICATION

[1] **Zhang, Gehao**, and Juan Zhai. 2026. POSTCONDBENCH: Benchmarking Correctness and Completeness in Formal Postcondition Inference. In *Findings of the Association for Computational Linguistics: ACL 2026*. Association for Computational Linguistics.

[2] Liang, Ming, Xiaoheng Xie, **Gehao Zhang**, Xunjin Zheng, Wei Jiang, Chengpeng Wang, Gang Fan, and Peng Di. 2026. RepoFuse: A Dual-Context Approach to Repository-Level Code Completion at Industrial Scale. In *Proceedings of the 34th ACM International Conference on the Foundations of Software Engineering*. ACM.

[3] Xu, Chuyang, Zhongxin Liu, Xiaoxue Ren, **Gehao Zhang**, Ming Liang, and David Lo. 2025. FlexFL: Flexible and Effective Fault Localization With Open-Source Large Language Models. *IEEE Transactions on Software Engineering* 51 (5): 1455–71.

[4] Di, Peng, ⋯, **Gehao Zhang**, et al. 2024. CodeFuse-13B: A Pretrained Multi-Lingual Code Large Language Model. In *Proceedings of the 46th International Conference on Software Engineering: Software Engineering in Practice*. ACM.

[5] **Zhang, Gehao**, Eugene Bagdasarian, Juan Zhai, and Shiqing Ma. 2025. Disappearing Ink: Obfuscation Breaks N-Gram Code Watermarks in Theory and Practice. Version 1. Preprint, *arXiv*.

[6] **Zhang, Gehao**, Zhenting Wang, and Juan Zhai. 2025. Breaking the Myth: Can Small Models Infer Postconditions Too? Version 1. Preprint, *arXiv*.

[7] Li, Mingzhe, Kejing Xia, **Gehao Zhang**, Zhenting Wang, Guanhong Tao, Siqi Pan, Juan Zhai, and Shiqing Ma. 2025. EDITOR: Effective and Interpretable Prompt Inversion for Text-to-Image Diffusion Models. Version 3. Preprint, *arXiv*.

[8] Liang, Ming, Xiaoheng Xie, **Gehao Zhang**, Xunjin Zheng, Peng Di, Wei Jiang, Hongwei Chen, Chengpeng Wang, and Gang Fan. 2024. RepoGenix: Dual Context-Aided Repository-Level Code Completion with Language Models. In *Proceedings of the 39th IEEE/ACM International Conference on Automated Software Engineering*. ACM.

## PROJECT

***Inference Optimization for CodeFuse*** Oct 2022 – Aug 2024

*Initial Core Member*

Contributed to CodeFuse, Ant Group's code-focused LLM initiative. Built a high-performance LLM inference framework supporting 10B-parameter GPT-class models with industry-leading throughput and low latency, serving dozens of production scenarios and 3,000+ concurrent users. During this project, I contributed performance optimizations to several open-source LLM projects, including Qwen3-Coder, FasterTransformer, and TensorRT-LLM.

***AntCCD: Fused Code Clone Detection*** Jul 2021 – Sep 2022

*Project Leader*

Developed a deep learning-based code clone detection system covering 9,000+ repositories. Achieved three times the recall of commercial products and helped reduce Alipay's code duplication rate by ∼10%.

## OPEN SOURCE CONTRIBUTION

`QwenLM/Qwen3-Coder` ..... *Bug fix for the SFT scheduler.*
`NVIDIA/FasterTransformer` .. *A bug fix and added feature support for GPT-NeoX.*
`NVIDIA/TensorRT-LLM` ..... *Python generation API features.*

## TEACHING

***TA Experience – University of Massachusetts Amherst***

- COMPSCI 621: Advanced Software Engineering 2026
- CICS 210: Data Structures 2026
- COMPSCI 383: Artificial Intelligence 2025
- COMPSCI 589: Machine Learning 2025

***Other Teaching Experience***

- Lecturer for the "LLM Inference" module in Ant "Yuan" Academy, an internal course series at Ant Group.