

Yueke Zhang

Ph.D. Candidate in Computer Science | Vanderbilt University
yueke.zhang@vanderbilt.edu | zyueek.github.io

Application for Assistant Professor in Computer Science

Division of Natural and Applied Sciences
Duke Kunshan University

Dear Members of the Search Committee:

I am writing to apply for a tenure-track Assistant Professor position in Computer Science at Duke Kunshan University. I am a Ph.D. candidate at Vanderbilt University, advised by Professor Yu Huang, with graduation expected in 2027. My research develops trustworthy AI for software systems by connecting learning, reasoning, and security. I aim to make complex development tasks easier to delegate while preserving human control over the work. I am drawn to DKU's interest in software engineering and AI, its emphasis on undergraduate teaching, and the opportunity to work with colleagues across disciplines.

My research follows two connected themes: trustworthy and efficient AI for code, and secure open-source systems. Within the first theme, I connect software understanding, cognition-informed learning, and verification. My work on fault localization and static analysis examines what program information helps developers diagnose failures and make repairs. To improve how AI models learn from code, I translate human code-reading attention into training supervision and use EEG and gaze to guide selective model adaptation. Learning from code must also be connected to the requirements that code should satisfy. My current work on code generation and verification, now under review, uses structured feedback, program structure, and proof obligations to guide generation and repair. The second theme extends this concern with trust from software behavior to the origin of software contributions. My studies of GitHub impersonation and automated authorship attribution examine how code and repository history can help distinguish legitimate collaboration from misleading attribution. Together, these themes address how AI can support software development while preserving confidence in both the resulting code and the contributions behind it. My work has been published at ICSE, FSE, ACL, ASE, and CSCW, and in TOSEM and TMLR. My first-author paper Unclaimed Territory (ICSE 2027) analyzes subdomain takeover risks in GitHub Pages.

At DKU, I would study how agents can learn from experience while continuing to follow the user's instructions. With the Precision Health and Medicine Observation Laboratory, I would extend my work on EEG and gaze to study which parts of a model need updating and which experiences an agent should keep in memory. The Zu Chongzhi Center's work on graph-based learning connects to my research on program structure and authorship. I would connect program structure to proof obligations that describe what generated code must satisfy, and use code and commit history to support authorship review. Through the Digital Innovation Research Center's Data+X projects, I would study verifiable rewards for agents that write data-analysis code. I would build on my formal reasoning work to express the required behavior as specifications and proof obligations, with independent verifiers checking whether each revision satisfies them. These checks would provide learning feedback, while the specifications and verification process would remain outside the agent's control.

At Vanderbilt, I have served as a teaching assistant for Principles of Software Engineering and Advanced Topics in Software Engineering. In the undergraduate course, I helped students debug code and contribute to projects on GitHub, and taught several sessions reviewing course concepts. In the graduate course, I guided research projects, led presentation sessions, and evaluated students' work. I also mentored undergraduate researcher Anda Liang, who co-authored our ICSE 2025 paper. Working with students at these different stages has taught me how to break a difficult task into manageable steps and give feedback that helps them decide what to do next.

I would welcome opportunities to teach Software Engineering, Program Analysis, Human-Computer Interaction, and Operating Systems, and to develop a course in Coding Agent Development. Coding agents can now produce code quickly, but a completed assignment does not show whether a student understands the solution or can fix it when something goes wrong. I would therefore ask students to state requirements before using an agent, explain key parts of the resulting code, and adapt it when requirements change. Short individual exercises and code-review discussions would help me assess their own reasoning. My research on coding agents would support projects in which students investigate an agent-generated change that fails a test, use program analysis and verification feedback to improve it, and check the revised behavior. My studies of open-source collaboration and authorship would extend these projects to shared repositories: students would submit a pull request with tests and a clear explanation, document AI assistance, credit collaborators, and respond to review. They would also examine contribution history alongside the code, learning why the name on a commit alone is not enough to establish authorship.

Thank you for considering my application. I would welcome the opportunity to discuss how my research and teaching could contribute to Duke Kunshan University.

Sincerely,

Yueke Zhang