Teaching Statement

Cheng Xin
Computer Science Department, Rutgers University
cx122@cs.rutgers.edu

Teaching Philosophy

My teaching philosophy centers on bridging theoretical rigor with practical understanding, enabling students to develop both strong foundational knowledge and problem-solving skills. Through my progression from teaching assistant to course lecturer, I have developed an approach emphasizing clear explanation of fundamental concepts, systematic problem-solving methodology, and active engagement with algorithmic thinking.

Teaching Experience

My teaching journey began at Purdue University, where I served as a teaching assistant for both undergraduate and graduate courses, and has progressed to full instructional responsibility at Rutgers University.

Teaching Assistant Experience: As TA for **Data Structures and Algorithms** (Spring 2023, 200 undergraduate students) at Purdue, I led recitation sessions complementing main lectures through detailed explanations and hands-on problem-solving. My responsibilities included elaborating on complex algorithmic concepts through step-by-step analysis, guiding students through homework problems while encouraging independent thinking, bridging theory with implementation, and providing individualized support during office hours. I also served as TA for **Computational Geometry** (Fall 2020, 30 graduate students), working with advanced students on sophisticated geometric algorithms and theoretical foundations.

Lecturer Experience: Currently serving as **Lecturer for Design and Analysis of Algorithms** (Fall 2025, 45 graduate students) at Rutgers University, I have full instructional responsibility including designing comprehensive lectures on fundamental algorithmic paradigms (divide-and-conquer, dynamic programming, greedy algorithms, graph algorithms), developing course materials and examinations, creating problem sets progressing from foundational to research-oriented challenges, integrating contemporary applications and research developments, and mentoring graduate students in developing rigorous analytical skills. This experience reinforces that effective graduate teaching requires balancing theoretical depth with practical insights while fostering intellectual curiosity.

Teaching Methodology

Progressive Concept Building: I structure material systematically from fundamental principles to advanced topics, beginning with intuitive examples before introducing formal analysis, ensuring solid conceptual foundations.

Interactive Problem Analysis: I emphasize active learning through guided problem-solving, walking students through algorithm design—identifying problem structure, considering approaches, and analyzing trade-offs.

Bridging Theory and Practice: I consistently connect theoretical concepts to practical applications, drawing examples from my research in topological data analysis and machine learning. When teaching graph algorithms, I illustrate applications in modern ML systems.

Concrete Examples and Visualizations: For complex algorithms, I use systematic visual representations and worked examples, particularly effective for dynamic programming and graph traversal.

Emphasis on Rigorous Analysis: I stress formal correctness proofs and complexity analysis, teaching

students to think precisely about algorithmic guarantees—essential for graduate students.

Integration of Research and Teaching

My research in geometric and topological foundations of machine learning enriches my teaching by demonstrating how foundational algorithmic concepts extend to cutting-edge applications: discussing how classical graph algorithms relate to modern graph neural networks and scientific computing applications, illustrating connections between computational geometry and topological data analysis, and showing how algorithm design principles apply to contemporary problems in machine learning and AI. This integration helps students appreciate the enduring relevance of fundamental computer science concepts.

Future Teaching Aspirations

Building upon my experience teaching both undergraduate and graduate courses, I am well-prepared to teach across the computer science curriculum:

Core Computer Science Courses: I can effectively teach fundamental courses including data structures, algorithms, discrete mathematics, and computational theory, adapting content and pedagogical approaches to different student levels.

Advanced and Specialized Courses: I am eager to develop courses in topological data analysis, geometric methods for machine learning, graph representation learning, and computational geometry. Such courses would provide students with exposure to emerging methodologies while maintaining strong connections to core principles.

Mentorship and Research Training: I am committed to mentoring undergraduate and graduate students in research, helping them develop the combination of mathematical rigor and practical skills necessary for conducting high-quality research. My experience collaborating with students positions me well to guide the next generation of researchers.

Curriculum Development: I aim to contribute by creating courses that bridge traditional computer science education with contemporary developments in machine learning and data science, ensuring students are prepared for both academic research and industry careers.

Inclusive Teaching Practices: I am committed to creating inclusive learning environments supporting students from diverse backgrounds, providing multiple pathways to understanding complex material, offering flexible support mechanisms, and fostering a classroom culture that values diverse perspectives and collaborative learning.

Commitment to Teaching Excellence

My progression from teaching assistant to lecturer demonstrates commitment to continuous improvement. I actively seek feedback from students, reflect on teaching effectiveness, and adapt approaches based on learning outcomes. I view teaching as a collaborative process of intellectual development, where my role is to guide students in becoming independent, critical thinkers capable of tackling novel challenges. As I continue my academic career, I remain dedicated to excellence in teaching, to mentoring the next generation of computer scientists, and to fostering learning environments where students develop both technical expertise and intellectual curiosity essential for lifelong learning and professional success.