

Teaching Statement

Shaikh Akib Shahriyar | Rochester Institute of Technology | as8751@rit.edu

Teaching Philosophy

I want students to leave my courses able to investigate an unfamiliar problem, test a solution, and explain why the evidence supports their conclusion. In cybersecurity and machine learning, students develop that judgment by moving between clear explanation, guided practice, and increasingly independent work. I therefore design courses around questions that students can explore through discussion, hands-on experiments, and open-ended projects. My goal throughout is to make difficult ideas approachable while helping students become more demanding of their own reasoning.

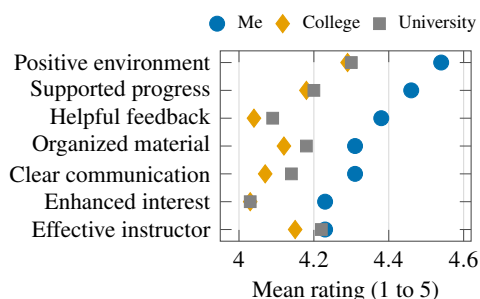
Prior Teaching and Mentoring Experience

I began teaching shortly after completing my B.Sc., first at North Western University in Khulna and later at Khulna University of Engineering and Technology (KUET). I joined KUET as a Lecturer and was promoted to Assistant Professor in early 2021. I remained a member of the faculty while on approved study leave until 2023. Across both universities, I taught courses from algorithms and software engineering to machine learning, computer security, and mobile computing. Classes had about 40 students at North Western and about 120 per course at KUET. In every course I gave lectures, supervised laboratories, designed assessments, and advised student projects. I also taught machine learning and computer networks to K–12 teachers at the Government Teachers' Training Institute in Khulna. That work taught me to adapt technical material to learners with different preparation and professional goals.

At Rochester Institute of Technology (RIT), I have taught the graduate course Deep Learning Security and the undergraduate course Introduction to Cybersecurity. Deep Learning Security connects current machine-learning research with adversarial evaluation. In Introduction to Cybersecurity, I connect foundational security concepts with the systems, risks, and decisions students encounter in everyday life. I have also given guest lectures in several other RIT courses.

At RIT, I have mentored 24 undergraduate, master's, and junior doctoral students. At KUET, I mentored more than 65 students and co-authored seven student-led peer-reviewed publications.

Student feedback. In the Spring 2025 Deep Learning Security evaluations, my ratings exceeded the reported college averages on every instructor-specific item (Figure 1). The largest margins were for helpful feedback (4.38 compared with 4.04) and support for student progress (4.46 compared with 4.18). Students also rated the positive learning environment at 4.54, compared with the college average of 4.29. The written evaluations also identified a consistent area for improvement. Students wanted assignments graded more quickly and the relationship between individual scores and their course grade to be clearer. In future offerings, I will schedule grading checkpoints so students receive feedback before the next related assignment and can act on it.



"Akib did a great job communicating and explaining the different machine-learning algorithms from the beginning of the semester. The structure of the class was also set up very well. He was always eager to check in with students to answer any questions."

Anonymous student, CSEC 720 SmartEvals

"I appreciate his high flexibility and accessibility. He also presented the content and facilitated the discussions very well."

Anonymous student, CSEC 720 SmartEvals

Figure 1: CSEC 720 student evaluations (Spring 2025; 13 of 13 students responded) and two anonymous comments.

Approach to Teaching

Active learning and critical discussion. I open with a concept or a motivating example, then give students a question to work through alone or in small groups before the class discusses their reasoning. In graduate seminars, students also write questions about the assigned research on activity sheets that rotate around the room, so that classmates can respond to one another's interpretations. Presentations are assessed partly on the questions students ask their peers. I teach them to move from clarifying questions toward questions that test a result, such as what would change if the attacker had different capabilities.

I plan these activities with Bloom's taxonomy in mind, so that students progress from explaining and applying a method to evaluating evidence and designing a solution. Together, these practices make misconceptions visible, help more students enter the conversation, and let me adjust the pace or revisit an idea when necessary. I also maintain a course communication channel for quick questions, reminders, and class-wide clarifications so that answers are available to everyone.

Hands-on work that moves from guided practice to justified choices. Discussion works best when students have evidence to discuss, so hands-on work is a recurring part of my teaching. Students understand cybersecurity and machine learning more deeply when they can test ideas directly, and experiments connect abstract concepts with observable system behavior. I typically provide a working baseline and ask students to modify an important component, justify the change, and interpret the result. I remove incidental setup barriers while leaving the central design and analysis decisions to students. In an adversarial machine-learning assignment, for example, students generate adversarial examples, vary the attacker's constraints, and test whether training against one attack improves robustness against another. At KUET, students similarly moved from guided laboratories to independent systems and mobile-computing projects.

Across these settings, producing working code is only the beginning. Students must explain what their experiment shows, what it does not show, and how another person could reproduce it.

Projects that develop independent judgment. Projects extend this progression by asking students to formulate and defend a question of their own. I scaffold that process through a literature review, a proposal, intermediate feedback, peer review, and a final paper, presentation, and reproducible implementation. During individual meetings, I help students narrow an ambitious idea into a feasible experiment while preserving their ownership of the project. Recent topics have included neural-radiance-field approaches to deepfake detection, attacks on language-model training, and language-model-based social honeypots.

Across assignments and projects, students must make the assumptions behind a security claim explicit and consider who could be affected if those assumptions fail. My work with journalists and forensic analysts gives me concrete examples of how technical decisions affect evidence, privacy, and public trust. Through this process, students learn to revise assumptions, interpret unexpected results, and explain what further evidence would be needed to strengthen a claim. In advanced courses, I use a culminating project to help students integrate the ideas and methods developed across the term.

Clear expectations, feedback, and support. Students can take on independent work only when expectations are clear. They arrive with different levels of programming experience and familiarity with research, so I provide runnable examples, explicit deliverables, and rubrics that separate implementation, analysis, and communication. Students also review one another's drafts, which gives them practice in offering and using constructive criticism. I use limited, transparent flexibility for unexpected circumstances while maintaining consistent expectations for the class. My AI-use policy distinguishes support with explanation, brainstorming, and debugging from submitting generated work without understanding or attribution. The goal is not only to enforce academic integrity, but also to help students develop sound judgment about tools they will encounter professionally.

Approach to Mentorship

I see mentorship as a gradual transfer of responsibility. Early in a project, I work closely with students to identify relevant literature, narrow the question, and build a small prototype. As they gain experience, I ask them to propose experiments and anticipate what each possible outcome would show. Over time, I gradually shift responsibility for selecting the next question, designing the experiment, and directing the project.

Mentoring also covers the parts of research that are often left unspoken. I help students read papers critically,

document experiments so that others can reproduce them, write clear arguments, respond to reviewers, present at conferences, and plan a career in research or industry. Having taught and studied in Bangladesh and the United States, I know that expectations about academic work are not equally familiar to every student, so I try to make them explicit and open to discussion.

Supporting students also means paying attention to the conditions in which they work. As Vice President of RIT's Doctoral Student Association, I organized peer gatherings, research-skills workshops, and career conversations with alumni and industry professionals. At KUET, I organized programming contests, hackathons, and industry events where students practiced collaboration and explored professional paths. In my own group, I will set clear expectations for meetings, feedback, and communication, encourage students to raise difficulties early, and connect them with university resources when they need support.

I also want students to see how research moves between academic questions and deployed systems. My own career has spanned both settings, from leading a four-engineer team that built number-plate recognition for edge devices to studying the security of machine-learning systems. I will bring practitioners and researchers to lab meetings as visiting speakers, and my continued collaborations with Zatiq and Apprendi will keep that connection active. I will also pursue collaborations with other academic groups and industry laboratories. Working across these settings would expose students to different methods, research cultures, and career paths, and it would prepare them for careers in academia or industry.

Courses and Teaching Plan

I can teach undergraduate and graduate courses in cybersecurity and privacy, machine learning and computer vision, and networking. My earlier teaching also prepares me for core courses such as algorithms, software engineering, and mobile computing. I would adjust the balance between guided exercises and open-ended work to the level of the course. Introductory students need repeated opportunities to practice foundational ideas with guidance, whereas advanced students need more responsibility for formulating questions and evaluating competing approaches.

Beyond core courses, I would develop graduate seminars in Secure Agentic AI and Synthetic Media Forensics, incorporating foundations of agentic AI as well as robust and applied generative AI. My research across agentic systems, adversarial machine learning, generative AI, and media forensics provides the foundation for these courses. These courses would combine critical reading, reproducible experiments, and a focused project in which students formulate and evaluate an original question. As these areas evolve, the seminars would also give students and me a setting in which to explore new research problems together. Across my teaching and mentoring, my goal is to help students move from applying established methods to formulating questions and producing evidence they can defend.