

Teaching Statement

KMA Solaiman

1 TEACHING PHILOSOPHY

Accessible Learning My teaching philosophy revolves around active learning and student well-being. Student well-being in a classroom can depend on their sense of autonomy, achievement, and participation. For example, in one of the courses I taught CS536, the students were provided a choice for their project topics from reproducible papers, novel idea implementation, or their ongoing projects. Also, for the final presentation deliverable they were asked to choose from multiple modalities i.e., video presentation, demo, or blog write-up. These choices gave the students a sense of autonomy. Satisfying these needs could intrinsically motivate the students to actively participate and grow in the classroom.

Relationship with the Students At the beginning of my classes, I make a conscious effort to know the students' backgrounds and their preferred outcomes from that course. This helps me understand them better, formulate relevant examples during the teaching, and allow me to adjust the course outline accordingly. "*If you cannot explain it simply, you do not understand it well enough.*" - I truly believe that and want that as a final outcome for my students. To that end, I ask for *frequent feedbacks* from my students to learn more about their understanding and thought processes, as well as identify any alternate conceptions students may have that need to be addressed.

Growth-focused Course Design (i) *Course Materials*: I prefer to design courses in a way that gives the students the ability to measure their own progress. The courses consist of both the foundational CS knowledge and the application of the concepts to problem-solving. Depending on the audience type, the course may have varying style of materials. For students aspiring for industry positions, it is imperative that they can apply the academic knowledge to solve real-life problems whereas, for the students pursuing research careers, it is important to be able to come up with solutions to new problems from the existing knowledge. I grew very fond of the concept of *learning by doing* from my own courses and prefer to utilize this. Allowing students to do different types of written or programming assignments alongside the relevant lectures bolsters their understanding of theoretical knowledge. (ii) *Assesment*: During grading, I always design rubrics that are very detailed, have lower stakes, and provide significant feedback to the students so that they can understand what mistakes they have made and can learn from it. Along with testing the fundamental and applied knowledge, I frequently add problems during the coursework that make them think on a deeper level.

2 TEACHING EXPERIENCE

My teaching experience consists of hands-on teaching for 6 years, dating back to 2014, and taking training courses such as 'Effective teaching in CS', or 'Foundations of College Teaching'. The graduate TA training post COVID-19 has helped me get more familiarized with online teaching while learning about essential tools such as Brightspace, Campuswire, Gradescope, etc.

My first experience of teaching undergraduates was in Bangladesh during Fall'14. I conducted multiple graphics, data structure, and programming labs. While later two involved explaining students how the core concepts are applied in problem solving, *graphics* allowed me to mentor students to build tangible outputs. My second teaching experience in Bangladesh was on a much broader scale teaching *Network Programming, Database, and Software Development* with a maximum class size of 143, divided into two sections. My duties included teaching the theory class, designing labs and lecture materials, conducting labs with teaching assistants, grading, and advising. By handling large classes and multiple sections I learned how to guide and evaluate everyone in a balanced manner. All of these courses included building full stack projects. It taught me how to help students throughout semesters while giving continuous feedback. The satisfaction and joy I shared with my students seeing the final projects bolstered my choice of being a teacher.

After joining Purdue, I became a teaching assistant for *object-oriented programming* which is a regular freshman level course. By the third semester on this course, I along with two of my colleagues have taken

the role of structurally developing this course for long run. During 2016-23, I was a teaching assistant for two graduate-level courses along with three undergraduate courses. The biggest difference from my previous teaching experience was the multicultural and diversified international student body. At Purdue, my responsibilities as a GTA included designing, testing, and grading programming assignments, projects, and written homework, teaching labs, and PSO sessions, assisting in creating and grading exams, and advising students during office hours and in online forums. Through this process, I have realized how differently students at senior and junior levels learn and communicate with instructors. For CS180 and CS251, I was responsible for overseeing undergraduate TAs, whereas for upper and graduate level classes I had to handle complex situations like discussing students' fundamental research questions on my own. My goal for the weekly PSO sessions is to help students utilize the concepts learned in the lectures for their assignments with an in-depth understanding. As soon as an assignment is released, I go through the logic behind it and how does the core concepts build up to the final assignment. During the grading, I tried my best to leave detailed feedback of the issues in their implementation, so they can learn from their mistakes. The mental health of the students is very important to me, and I encourage and practice empathy in my class, which includes being aware of their hesitancy to ask questions or being inquisitive about their learning process. Most recently one of my students in network programming class felt comfortable enough to talk to me about his anxiety, and I have tried my best to accommodate and comfort him with Purdue's ongoing support for mental health management.

I have guest lectured on *situational knowledge, knowledge graphs, and multimodal information retrieval* where I talked about *cross correlation learning, metric learning, decoder-encoder, and attention networks*. The lectures involved *feature extraction from multiple modalities, graph embedding, and graph matching techniques*. I have always enjoyed public speaking and that has helped me to deliver presentations at a large scale numerous times during my Ph.D. I have presented research from me and my colleagues at Northrup Grumman Corporation (NGC) Review Meetings, NGC Techfest (with 100+ attendants), JPL Nasa, and Darpa Review Meetings.

Mentorship: During my time in Bangladesh, I mentored four seniors for their thesis and actively participated in grooming students for 'Competitive Programming Competition'. In Purdue, I have mentored multiple master's and undergraduate students for independent research courses (included in Diversity).

3 TEACHING PLANS

I am excited to teach both undergraduate and graduate-level students. I am comfortable teaching databases, networks, compilers, information retrieval, data management systems (SQL and NoSQL systems), and machine learning courses to both undergraduate and graduate students. I know the differences between them from my own experience of teaching at both levels. For undergraduate students, I would be happy to teach core courses including programming languages, data structure, theory of computation, and computer graphics. Specifically, for graduate students, I can offer advanced information retrieval, natural language processing, distributed database systems, or data mining. Moreover, I have plans to offer seminar courses related to multimodal information retrieval and adaptable and explainable AI to students with an interest in research. These courses will be based on recently published papers in top conferences in machine learning, information retrieval, OpenAI, and XAI conferences. The course projects will be designed in a way that can lead to publications and can mimic peer reviewing.

Besides that, I would also like to develop a new course "*Applied Machine Learning for Open World Systems*" to extend the current curriculum based on my research experience. Tentative topics would include data cleaning, handling lack of annotations, scalability, weakly supervised learning, multimodal information retrieval and feature extraction, intrinsic and extrinsic complexity of system domains, and case study of real-world systems. Since learning algorithms need to be adaptive to real-world situations, it will also include detection, adaptation, and difficulty analysis of novelties in machine learning algorithms.