TEACHING STATEMENT

Qin Yang (RickYang2014@gmail.com)
Personal Website: https://rickyang2016.github.io

While research is what initially drew me to computer science, I derive great pleasure from teaching and revel in moments when you see someone understand a concept for the first time. Rather then viewing teaching and research as rivals for my time, I experience them as highly synergistic activities. Research, which requires exploring alternative approaches and paradigms, helps refine and clarify the ideas we teach. In turn, teaching requires articulating ideas more simply, and the search for the clearest explanation often results deeper insights. It is because of this synergy that I am applying for academic positions at universities that combine research and teaching.

Teaching Experience

As a master student at Colorado School of Mines I taught extensively as a TA, covering classes on Introduction to Computer Science (CSCI 101), Introduction to CS-LAB (CSCI 102), Introduction to the Linux Operating System (CSCI 274), Discrete Mathematics (CSCI 358), Database Management (CSCI 403), Computer Simulation (CSCI 423), Computer Graphics (CSCI 441), and Advanced Simulation (CSCI 542). In addition to this TAing, my office hours also allowed me to interact with students individually and get their feedback on various issues. For example, I was surprised to learn that many more students than I expected actually look through class slides before the class. Given this experience I will ensure that I will make my teaching slides available beforehand. Another thing I learnt the hard way is grading can be a contentious issue if the grading scheme is not made clear beforehand. And students are much happier if a detailed marking scheme is provided for each question in an exam rather than a coarse evaluation. Many such finer details which may not directly get discussed in a classroom came to my notice through individual interactions.

As a Ph.D. student at University of Georgia, I was a TA for undergraduate Discrete Mathematics (CSCI 2610) and Database Management (CSCI 4370/6370). Also, I served as an instructor for Introduction to Computational (CSCI 2150L), where I taught a weekly section to approximately 200 students and was responsible for developing projects and grading assignments.

Teaching Approach

My own experiences as a student and my time TAing have led me to believe that students learn better when they can develop and test their own hypotheses, rather than merely be told the accepted wisdom. In a small classroom this might be achieved through the Socratic method where students have a chance to be engaged by questions, rather than to be drowned in prepackaged answers, through dialog both among themselves and with the instructor. In addition, many topics in computer science, especially ones that I have been involved in teaching, revolve around building and examining artifacts (e.g., Introduction to the Linux Operating System, Computer Graphics, and Advanced Simulation); allowing students time to design or interact with these artifacts can help them develop new hypotheses, and prove that the lessons they have learned apply to the real world.

While this approach – combining the ancient wisdom of the Socratic method with the modernity of real computational artifacts – may seem idealistic and impractical, I have tried this (albeit on a small scale) in my TAing at Colorado School of Mines and University of Georgia. For most students, this appeared to work quite well, though it was hard for students who were uncomfortable participating in class. In the future I hope to develop a broader range of participatory modalities (e.g., through allowing students to interact through online forums or anonymously), that would ensure this method is more inclusive.

As a student and TA, I have also found that projects provide a mechanism to teach skills that can not otherwise be taught through classes. For instance, Computer Graphics (CSCI 441) at Colorado School of Mines required students to develop a complete video game in a little less than 14 weeks, and is widely regarded as one of the hardest undergraduate CS courses at Colorado School of Mines. This was a transformational course for me, introducing me to the material in more depth and intensity than any lecture course could have.

Finally, in my time as a Ph.D. student, I benefited greatly from courses that discussed ongoing research recent advances in particular areas. These courses were both useful as a way to learn about new results, but also provide a simple entry point into computer science research. For example, I started working on my doctor dissertation [1] as a result of taking a graduate class on Multi-Agent/Robot Systems (CSCI 8535) and Advanced Representation Learn (CSCI8945) which used simple problems (e.g., multi-robot cooperation, deep reinforcement learning, etc.) to present then current research on addressing these problems.

Based on these experiences, I would hope to rely heavily on projects, and incorporate some recent reading, into any upper-level undergraduate course. I believe the combination of participation-oriented lectures, programming-oriented projects, and research-oriented readings would leave to a more enjoyable and rewarding classroom experience.

Google Scholar: https://scholar.google.com/citations?user=t6e_A9kAAAAJ&hl=en

Teaching Plans

For undergraduate level, I am interested in teaching core computer science courses such as Artificial Intelligence, Databases, Programming Languages, basic Optimization and others. I am also interested in developing and teaching new non-conventional courses such as a course that uses science fiction to teach Artificial Intelligence or a course that uses e-commerce or world wide web as domains to introduce databases. I believe that these mixed type of courses would be more suitable at an undergrad level since they will be able to provide practical context to otherwise theoretical subjects and potentially inspire students to pursue these subjects at an advanced level. For graduate level, I am interested in teaching advanced courses such as Advanced Artificial Intelligence, Multi-Agent/Robot systems, Robotics, Game Theory, Reinforcement Learning and others. In addition, I would also like to teach specialized research courses that focus on research in Artificial Intelligence and building practical AI systems. Students of late are exposed to a lot of options to pursue at a very early stage and hence decide on their career paths early. As a passionate researcher in the field of Robotics and Artificial Intelligence, I believe that it is my duty to expose them to the joys of research and help to spot and groom the future scientists early.

References

[1] Q. Yang, *Self-Adaptive Swarm System*. PhD thesis, University of Georgia, 2022.