Arun Sharma

Teaching Statement

Computer science is a continuously developing field. Due to its ever-changing landscape, my responsibility as a faculty member is to instill curiosity and confidence in students so they can thrive in the field. Inspiration and engagement are the two key elements in my teaching philosophy. In the following, I first give an overview of my experience with teaching and mentoring and then explain my teaching philosophy relating to this experience. Finally, I list some of the courses I look forward to teaching.

1. Experience

Teaching: During my Ph.D. studies, I had three seminal experiences as a teaching assistant. At the graduate level, I contributed to three courses: Architecture of Database Systems (Csci 5708), Spatial Data Science (Csci 5715), and Spatial Data Science Research (CSCI 8715). I was deeply involved in assisting with the Algorithms and Data Structure (Csci 4041) course at the undergraduate level. I had the privilege to facilitate the academic growth of over 600 students across these modules. Within the classroom setting, I not only took on the responsibility of guest lecturing for nearly a dozen sessions but was also instrumental in curating course materials, including slide decks, quizzes, readings, and examinations. Beyond the classroom, I conducted office hours, crafted intricate homework assignments from foundational concepts, refined pre-existing content, and established grading rubrics. Furthermore, I adeptly managed course digital platforms, ensuring an organized repository of reading lists and fostering an interactive student forum. The Csci 4041 course, with its robust enrollment of over 400 undergraduates, particularly honed my skills in managing substantial class sizes. I conducted weekly recitations, streamlined grading processes, and meticulously evaluated programming assignments. In the recitations, I clarified complex topics based on student queries aligning with that week's lessons. In addition, I routinely encouraged students to engage in active group discussions of 3-4, dissecting a pertinent problem aligned with that week's syllabus.

Mentoring: I also mentored students at different stages, from graduate students to high-school interns in our lab. I guided two graduate students, Ruolei Zheng, and Micheal Ung, in their research project for the Spatial Data Science Research course (Csci 8715). The project, which entailed adding physical and probabilistic interpretation within a trajectory gap, gave me experience leading a research team to further my larger research project. I also mentored high school students, including Aditya Kulkarni, who co-authored a publication for the Association of Geographers Information Laboratories in Europe (AGILE) 2023 with me. His research findings were awarded 3rd place at the NSF COVID Info Commons Undergraduate Paper Challenge hosted by Columbia University. He also presented his work at the NSF Northwest Big Data Hub meeting. Aditya is now an undergraduate computer science student at Yale University. Currently, I am mentoring Yuv Magan from Wayzata High School, who is helping me with data curation and other small tasks related to identifying fake trajectories in the maritime domain.

2. Teaching Philosophy

I will use my teaching experience to describe the role that inspiration and engagement play when I teach. Inspiration is pivotal in broadening students' learning horizons, whereas engagement in teaching helps foster effective communication with students.:

2.1 Inspiration

I believe students' learning horizons can be broadened via (1) cultivating curiosity, (2) promoting effective reasoning, and (3) nurturing critical thinking:

Cultivating Curiosity: When teaching a new topic, it is crucial to spark students' curiosity and motivate them to learn more about it. For example, when I guest-lectured a class on spatial graphs and algorithms, I realized that many students might get bored and think spatial graphs (e.g., road networks) are just normal graphs with nodes and edges. To make them eager to know more, I started with a success story of UPS, which saves 10 million gallons of fuel annually by minimizing the number of left turns in their delivery routes. The modeling of "turns" is one of the features in spatial graphs that are not considered in traditional graphs. Since the students can easily relate UPS to their daily lives (e.g., receiving packages from Amazon), this example quickly got their attention in class. It got them interested in learning more about spatial graphs.

Effective Reasoning: It is essential to inspire students to think effectively. For example, I introduce dynamic programming by first exploring the Fibonacci sequence through recursion. Students quickly grasp the simplicity of recursion but also its inefficiencies. Highlighting these redundant calculations paves the way for introducing dynamic programming and memoization. This step-by-step approach ensures that students appreciate the progression and utility of advanced techniques, building on their foundational knowledge. I also illustrate the broad applications of dynamic programming in various fields, from optimizing routing protocols in computer networks to sequence alignment in bioinformatics and even decision-making processes in economics. This layered methodology cements their foundational knowledge and broadens their perspective on the real-world relevance of such techniques.

Critical Thinking: Finally, it is also crucial that students learn to think critically about a given concept to point out the limitations or tradeoffs of one concept over another. In my undergraduate algorithms course, I force students to solve a given problem with both greedy and dynamic programming algorithms where, in one case, the greedy approach is much more efficient than dynamic programming. In my graduate-level course on architecture in database systems, I challenge students to optimize a query tree. While some might employ a hash-based approach to minimize the join operations, others might opt for a bitmap index to expedite selective search. Each of these approaches has its advantages and is optimal in specific scenarios. By exploring multiple strategies, students not only learn the mechanics of each but also cultivate the ability to discern which is most suitable for a given situation, understanding that optimization is often a balance of trade-offs.

2.2. Engagement

Based on my experience, the most essential part of engagement is to think from the student's perspective, which includes understanding the students' background and thinking about what examples may resonate with them. Awareness of students' backgrounds ensures all can understand the technical vocabulary and concepts used in a class. This is critical because if too many students get lost and no longer follow the course content, class time becomes very inefficient. For example, our course on spatial computing attracts not only computer science students but also many students from other domains who need to deal with big spatial datasets (e.g., mechanical engineering, remote sensing, geographic information science). Thus, during my guest lectures, I made sure all the jargon was explained using broad language. In my slides, I added examples to connect their diverse domains (e.g., using spatial occurrence pattern mining in vehicle on-board diagnostic data for mechanical engineers). From coursework with my professors, I also learned other good practices for engaging students, such as periodically asking questions in class to check students' understanding, having students give class presentations, and having one-on-one face time with students.

3. Examples of Proposed Courses

My most exciting part of academic life is teaching and mentoring students to help them achieve their goals and positively impact society. I look forward to teaching various courses, such as the following. (The asterisk * indicates courses in which I have teaching assistant (TA) experience):

- Spatial Data Science*—This introductory course covers broad topics, including the value of spatial data, spatial data models, spatial networks, spatial statistics, spatial data mining, spatial classification and prediction, spatial optimization, trends, etc.
- Spatial Data Science Research*—This is a research-oriented version of the spatial data science course. It also covers general topics in spatial data science but with a stronger focus on critical reading of classic and new research/survey papers, proposing and carrying out research projects, etc.
- Introduction to Spatial Artificial Intelligence and GeoAI – An advanced course.
- Principles of Database Management Systems* – An introductory course covering topics on relational data models, entity-relationship diagrams, query languages, query optimization, normalization, physical storage, transaction management, data mining, etc.
- Algorithms and Data Structures – An introductory course covering topics on complexity analysis, fundamental paradigms (e.g., divide-and-conquer, dynamic programming, greedy method), graph algorithms, priority queues, search structures (e.g., trees), etc.
- Introduction to Machine Learning – An introductory course covering topics on discriminant and generative models, Bayes classifiers, kernel methods and SVM, decision trees and random forests, boosting, neural networks and deep learning, clustering, optimization methods, etc.
- Advanced Machine Learning – An advanced course that includes in-depth study of theorems (e.g., kernel methods), complexity analysis (e.g., VC dimension), advanced solvers (e.g., ADMM), etc.