Being in academia enables one not only to further the frontiers of knowledge via research, but as importantly, provides an opportunity to impact future generations through teaching. With the ever increasing adoption of electronic interfaces, and with easy access to large datasets, there is a need for engineers that are able to design creative, usable, secure, and intelligent computer systems; and I would like to be part of creating such a generation. Here, I will briefly describe the classroom and mentoring experiences, and then outline the key aspects I feel are needed for effective education: interactive resources, peer to peer interactions, and broader impact thinking.

Teaching Experience

Undergraduate Teaching: I have been involved with classroom teaching at various levels of undergraduate education. I taught a lecture per week (as a teaching assistant) for an introduction to programming course (UMass, CS121, Spring 2008) that allowed me to interact with students from a large variety of departments such as biology, mathematics, economics, etc. I also taught entry level introduction to data structures course (Vanderbilt, CS201, 2004-05) designed for junior CS majors. Recently, I have been mentoring senior undergraduate students for a project based course (UW, CSE 454, Winter, Fall 2015) on data analysis.

Graduate Teaching: I have had teaching experience in a variety of graduate-level courses. I taught guest lectures for the introductory machine learning course at UW (CSE546, 2014-2015), giving me an opportunity to interact with first-year graduate students still trying to focus on a research topic. At UMass, I was the teaching assistant for an advanced machine learning course on graphical models (CS691, 2011-12). Since the course was being designed for the first time, I was involved directly in the creation of the homework assignments, one which was used by Prof. Daphne Koller from Stanford for her **Coursera MOOC course on graphical models**. Recently, I was a guest lecturer in a natural language processing course for professional Masters students, enabling me to observe a very different segment of students.

Mentoring: The skills of an educator are formed not only in the classroom, but via every interaction with the student population. I have been fortunate to mentor and collaborate with students from diverse backgrounds and levels of seniority. I was a **mentor to two high school teams** for Paul Allen's Computing Challenge in 2014. I have also mentored a number of undergraduate students for research, initially as a co-mentor with my adviser in graduate school, and now as a sole mentor in my current position. As a senior graduate student at UMass, I also got a chance to supervise a number of Masters students. At UW, I have been **co-supervising a few PhD students** with their respective advisers; some were initially advised by Ben Taskar, and I aided them in their transition to other supervisors after his passing. Recently, work with a masters students I was advising *remotely* received the **Yelp Dataset Challenge Award** for data science.

Teaching Capabilities: This variety of experiences with students makes me capable of teaching a broad range of courses. I feel equipped to teach introductory computer science undergraduate courses such as algorithms, theory, programming, and data structures, but can also teach more advanced courses in artificial intelligence, machine learning, and natural language processing targeting senior undergraduates and junior graduate students Further, I would like to create seminar-like variations of these for more advanced graduate students. I am also interested in **designing a novel course on *data science***, an inter-disciplinary, project-based course that I can customize for any level of education. The goal of the course will be to introduce students to the various aspects of data science: data cleaning and storage, querying and retrieval, statistics and machine learning, and visualization.

Teaching Philosophy

With the variety of experiences that I have had with students, and by reflecting upon at my own experiences as a student, there are a number of recurring observations that form my teaching philosophy.

Interactive Pedagogical Resources for Creative Exploration: The ability to break down and modify complex systems and concepts in order to see the effects of their various pieces is essential to understanding them. I feel courses should be designed with this paradigm in mind; unified materials for which every aspect is available for the students to alter, experiment, and investigate. Unfortunately, existing education resources are static and severely restrictive in their interactions, as they consist of a disjointed collection of textbook material, lecture presentations, and assignments. There is a good reason for this: the tools available to create such *static* resources (LaTeX, Powerpoint, Programming IDEs) are easy to use. Motivated by the need for such a tool for interactive resources, I have been developing *Moro*, a tool to create interactive, dynamic notebooks that effectively combine traditional course materials with animated plots, figures, and code snippets, all of which can be directly modified and regenerated. I recently **published a paper on interactive pedagogical resources** at *AAAI 2016 Symposium on Educational Advances in Artificial Intelligence* that describes the existing use cases of Moro, including a NAACL demo paper, tutorial at ACL, and a graduate course on NLP.

Social Interactions for Peer to Peer Education: By noticing that I understand concepts best when I teach them to someone else, I would like to set up the classroom whereby *students are involved in teaching others* as much as possible. Examples of such interactions include encouraging students to teach lectures and design assignments, selecting students to present assignment solutions, and peer to peer evaluation. *Collaborations* between students can often lead to effective learning, and I will encourage such interactions by using group projects, open-ended group assignments that are left deliberately ambiguous to encourage discussions, online student forums, and organizing students to solve problems together that they were unable to solve by themselves. Along with collaborative projects, a *healthy competition* can often motivate some. I would like to design assignments in which performance can be directly compared, for example simple competitive video game for which students design AI agents and comparing accuracy of machine learning predictions.

Broader Impact Applications: The objective of teaching is not only to impart specific technical ideas to students, but to also make them creative, interdisciplinary, and socially sensitive engineers and scientists, and I would like to encourage thinking along these lines in my classroom. Teaching courses that involve data analysis provides a unique opportunity to push students to think outside their comfort zones, and connect with people in other fields and from different backgrounds. Concretely, I will design assignments such that they use datasets related to a wide variety of issues, such as health-care, climate, city/state level governance, biology, social sciences, concerns facing developing countries, and so on. I feel that exploring such broader level applications not only makes the students more creative, skilled in different fields, and appreciative of the potential impact of the field, but also makes them socially responsible scientists invested in solving real-world problems in novel ways.