Sandeep Chinchali - Teaching Statement

One of the primary reasons I am pursuing an academic career is that I have thoroughly enjoyed teaching and research mentorship during my PhD. My most rewarding experience was being a head project teaching assistant (TA) for Stanford’s flagship introductory machine learning (ML) course. Rather surprisingly, this course showed me that students often feel obligated to use sophisticated deep-learning models, and sometimes are dissuaded from pursuing ML if their initial experiments with such models fail. Accordingly, I have refined a teaching and research mentorship philosophy rooted in sparking sustained student interest, supported by creative freedom and deep understanding, that I hope to carry forth in academia. My philosophy is to:

(1) Spark interest via novel applications and “playing” with data - I have seen firsthand the benefits of piquing student curiosity by having them collect their own unique datasets and “getting their hands dirty” with exploratory analysis before building ML models.

(2) Give students the creative freedom to fail and explore - My best mentoring results came from encouraging students to relentlessly iterate on ML models, despite initial failures.

(3) Embrace simplicity and understanding - Despite today’s fashion of using sophisticated deep-learning models, I encourage students to carefully interpret and critically reason about their modeling choices, which allows them to appreciate the beauty of well-applied ML.

I will now describe how I instantiated, refined, and plan to apply this philosophy in my past and future teaching and research mentorship.

Teaching Experience (Head Project TA for Stanford's Flagship ML course):

As the Head Project TA for Machine Learning, first taught by Prof. Andrew Ng, I had the pleasure of working closely with 191 students of diverse technical backgrounds to formulate ML problems in domains as varied as ecology, finance, robotics, or even stenography, build working ML models, and write clear research papers. I closely mentored teams from idea genesis to project completion, where I noticed several teams were initially intimidated and overwhelmed by sophisticated deep-learning models that their classmates with more experience seemed to easily use. To solve this problem, I worked closely with teams to first “play” with data with simple statistical models to see if their dataset had a trend, and then experiment with linear models and interpretable decision trees before trying deep networks. By letting them discover early “victories” with simple models, they were able to resiliently handle or avoid pitfalls from more sophisticated models without losing interest in their first foray into ML. When I co-organized a large poster session attended by Silicon Valley industry for students to showcase their results, I was proud of their ability to interpret their models, analyze weak-points, and critically defend the need for sophisticated methods. I could directly see the benefits of articulating this approach early, since those who carefully worked with me told me they had a rewarding, productive experience, and plan to pursue ML in the future.

Research Mentorship Experience:

My most rewarding mentorship experience was to work with two younger graduate students, a visiting scholar, and a visiting CS undergraduate from the University of Nebraska on the HarvestNet cloud robotics project, which culminated in a co-first author paper submission with them. The project asks - “if every robot can only upload 1% of images it sees per day to improve a computer vision model, how would it select these interesting images?”. Rather than use a public dataset, I encouraged collecting our own data, which helped me further refine my teaching philosophy of:

(1) Spark Interest By Collecting and “Playing” with data: My mentees were networking experts with minimal hardware or AI experience, and first wanted to shy away from dataset collection or analysis with
embedded AI chips. However, as I started collecting my own dataset of self-driving cars and construction sites on my daily commute, they became more intrigued in testing out the Google Edge Tensor Processing Unit (TPU) and even slowly bought their own dashcams to gather their own videos! Within just a month, my undergraduate mentee was even able to demonstrate efficient on-device model training. Together, we open-sourced our tutorials and custom pre-trained vision models, in the anticipation I would some day formalize this process in a hands-on-course, described in the next section.

2) **Freedom to explore and fail** - My mentees debunked early intuitions I had on the project, which allowed us to brainstorm working algorithms with their rigorous experiments.

3) **Embrace simplicity** - My mentees built a lightweight sampler and compute-efficient vision models that run on a USB stick, which led to a timely and innovative research contribution.

A highlight of our cloud robotics research was that the Japanese firm NEC selected my adviser’s lab, from over 10 potential groups in Stanford’s Platform Lab, to send a full-time research scholar from Japan for two years. I was in charge of mentoring this scholar, who had significant industry experience in Japan, albeit in classical networking. By leading daily project meetings and tutorials on computer vision, I quickly taught him the fundamentals of our project, for which he quickly became an integral contributor and paper co-author.

**Courses I Can Teach and Plan to Develop:**
I am excited to teach courses in ML, especially graduate introductory courses that excite students from a wide variety of disciplines. Furthermore, I am qualified to teach graduate courses in applied reinforcement learning, robotics, computer vision, and introductory networking, which are core techniques I apply daily in my research and have guest lectured about in the advanced Stanford course “Self-Programming Networks”. To ground my lectures, I weaved in real problems from my startup experience.

A key interest of mine is to develop a new course named *Cloud and Networked Robotics: A Hands-on, Deep-Learning Based Approach*. This course would teach students how to use state-of-the-art embedded AI chips, such as Google’s Edge Tensor Processing Unit (TPU), to do efficient, on-device ML and build networked applications where robots query central servers for better task accuracy. This course will directly apply my teaching philosophy as follows:

1) **Spark Interest By Collecting and “Playing” with data** - I will motivate students about edge computing and cloud robotics, since they can easily gather a video or LIDAR dataset for a domain that intrigues them and process it with powerful AI accelerators, which now can run on a USB stick for a cheap price of $75 or ideally lower by the time this course is taught.

2) **Freedom to explore and fail** - Not all perception or control tasks explored by students will have a clear benefit from cloud computation, which I hope students will discover themselves.

3) **Embrace simplicity and understanding** - Students will be encouraged to have computationally-efficient algorithms to decide when robots query the cloud, with no requirement for deep-learning.

**Summary:** Today’s society is increasingly turning to applications of ML in real-world, sometimes safety-critical, applications. I strongly believe tomorrow’s students will drive and lead such innovations, and I hope to first pique their interests through engaging classroom experiences where they have the freedom to explore, fail, and critically question conventional wisdom in ML, control theory, robotics, and other academic disciplines.