**Teaching Statement**

I always remember my graduate school as the most enjoyable years both learning and teaching computer science. When I am learning, I feel happy and my mind becomes sharp. I believe learning is lifelong journey especially computer science is evolving quickly and remains an exciting field. Teaching computer science fulfills my dream to giving back my knowledge to the society and making positive impact. Teaching is one of the most effective ways to master a certain topic. I always try my best to thoroughly understand a subject. Doing so not only prepares me to answer students’ questions, but also put myself into their shoes and coming up with ways improving their learning process. Students’ insightful questions often give me different perspectives to understand the topics.

I believe a quality teaching process has to be student-centered. Teaching is two-way communication between instructor and students. To teach a class, my first step is to understand the students, their existing knowledge, learning styles and their goal to taking this course. It is key to get this information early, through consultation with experienced instructors, or set up survey and questionnaires with students. The syllabus will be designed considering their existing knowledge, interests and learning styles. During lecture, I invite question after deliver of each module. I believe doing so not only engage students to think, but also get feedback from students. I also ask students question along one lecture to get them think about the content. Then delivery of lecture content will center around answering the questions. Taking a course is like trip, it is key to motivate students and provide roadmap at the beginning. Let them aware of the importance of the course (for example, it is foundation course that has application in many other courses, or it is introduction course, or it is a class that angle more advanced study in a certain area of computer science). During entire course, I believe it is key to know how students are doing at each step and prepare to adapt the teaching.

I gained my first teaching experience when I served as a teaching assistant (TA) in the Computer Department at University of Georgia. From 2000 – 2002, I was a teaching assistant for Introduction to Computing and Programming. Before the course start, I met with professor to understand the course design including lectures, labs, and projects. I tried to put myself into professor’s shoes and imagine it was me to design the course and teach it myself. The key point that I realized is that this course was introductory course for students whose major might not be computer science. Therefore, an interdisciplinary approach is taken to teach the course. Specifically, the labs and projects drew many practical problems from wide range of subjects, such as personal finance, biology, math, physics, economics and so on. During labs and office hours, I let students know that computer programming is a tool and friend that can help us solve many problems. The programming language is similar to the human language. It has its rules and vocabularies. In addition to guide students through labs and help students with questions about finishing the projects, I often briefly started with some background about formulating lab/project problem from real life. Once introduced to thinking and formulating problem in computing way, most students got very excited knowing that they could solve many other problems, not just the problems assigned in the course. They also developed intuition about computers and programming languages with analogies to human language and get more natural feeling learning the course material.

Another course I have assistant taught is algorithms (2002-2003). It is a fundamental course of computer science. I showed students its importance of in the computer science field. Algorithms can be everywhere, from operating systems, network, database design, AI to machine learning. I gave them overview of the common algorithms and their applications to help them to see the road map. For students that have strong interests and prior experiences, I expanded their understanding with more related application of each algorithm, when a student apply the same algorithm to different but same category of problems, he would get better mastery of the algorithm. For students that consider algorithms course too mathematical and hard to understand, I present them with simplified problem and simplified visualized procedures that’s representative of the basic form of algorithm. When their feedback was receptive, I started to show them how to improve the algorithm, for example, adding consideration for corner cases and find ways to optimize its time and space complexity.

After graduate school, I work at AT&T Labs, where I always keep my computer science knowledge and skills up to date. I have designed and provided a series of training for the following topics:

* Large scale data processing and analysis with Apache Hadoop and Spark, live demos are provided drawing work projects of mobility network data batch analysis
* Real time mobility network analytics with apache Kafka
* Applying data science to network optimization and analysis with Anaconda python. Live demos have been drawn from work projects that use jupyter notebook, pandas, matplotlib, seaborn, scikit-learn, keras and tensorflow

In addition, I have also mentored many new and junior employees at AT&T Labs. I enjoy the teaching aspects of mentoring, not only impacting my knowledge but also sharing my experience and recommending projects assignments. I also help my mentee understand conflict or explore ways to deal with problems. Last but not the least, I warned a couple mentee about behaviors that were poor fit with organization culture. As a mentor, I realized that I have to be keenly aware of my own behavior and I hold myself to high standard as good examples for them to follow. The models that I have established include passion in applying computer science to network engineering problem solving, and various ways to improve team work in engineering projects.

In summary, I believe my passion in computer science learning, teaching and industrial research and development experience set up foundation for me to teach and guide students’ learning, problem solving, working on cutting edge projects and future career development.

<http://matt-welsh.blogspot.com/2012/12/how-to-get-faculty-job-part-1.html>

**Research statement**

<<todo>>

narrative summary of your research contributions (and especially how they all tie together), and what areas you intend to work on in the future. It's usually about 3-4 pages long and needs to nail what your specific research "angle" is, why the area is important, what your track record is, and what your research vision is going forward.

They want to see that you have an independent and compelling vision for at least the first few years of your faculty job. If the best you can come up with is a couple of papers' worth of extensions to your thesis, you're in trouble. Try to think of a three-to-five year agenda that would get people excited to have you part of the faculty.