Teaching Statement

Learning is lifelong journey. I have been consistently educating myself, especially in the field of computer science which has always been evolving at fast pace. I am always passionate about how I can learn better and help other learn better. I always love teaching: at home teaching my kids various math, chess, soccer etc, also helping my wife learning computer science as she is transition to bioinformatics from a pure biology background. At work, I share my strength in computer science and engineering by presenting various training to my coworkers on Hadoop, spark, machine learning tool kit, python data analysis.

Teaching Philosophy: I learn best when my teacher instilled a passion for the subject. As most learning will happen outside the classroom, I believe my primary goal as a teacher is to motivate my students for self-study. -------------I will do this through thoughtful preparation of course content and its engaging delivery, as well as by striving to connect abstract concepts to tangible applications and demonstrations. I believe it is paramount to maintain a positive yet challenging classroom environment to foster student confidence and to encourage active participation. I look forward to developing an effective teaching style following these guidelines and to honing the craft over time.

passion in teaching and influence youth people. Knowledgeable in various subject in computer science. More than 10 years of industry experience. Value team work and would like to lead discussion in a small group for learning, design project that is to be done by a small team of students to help them gain team work experience. Would also help college students to prepare for careers after graduation.

At University of Georgia, Teaching experience as follows:

1. Teaching assistant for Introduction to computing and programming, 2000-2001

Work with instructor to design lab projects. Helping students doing lab projects.

Teaching assistant, grade homework, office hours (answer student questions regarding to the course subject, tutoring), guide student labs.

1. Teaching assistant for Algorithms

Homework grading, Office hour tutoring

1. Teaching assistant for software engineering
2. Teaching assistant for Database Management

How to design a technical training

At AT&T Labs, Presentation and offered training at team or group meetings.

1. Design presentations that offers training for the team on Hadoop distributed storage and map reduce, presented at team meeting. Introduction, technology development, major key points, get started with application to network data engineering, live demos, Q&A.
2. Design presentations that offers training for the team on Apache Spark application, presented at team meeting series. Introduction, development, key points, get started, live demos, Q&A.
3. Design presentation that offers training data science with anaconda python, live demo of jupyter notebook, data manipulation with pandas, data visualization of matplotlib and seaborn, machine learning with sci-kit-learn
4. Set up H2O.ai cluster, Design presentation that offers training using H2O.ai cluster for automated machine learning and gui-based machine learning modelling with h2o flow.

Project presentation

Dispatch prediction

a statement of teaching philosophy that discusses your philosophy regarding undergraduate and post-baccalaureate education, your philosophy towards scholarship at a masters comprehensive institution, and discussion about your past or proposed contributions to advance diversity, inclusivity, and/or equity in education

What does a TA do?

TAs may teach classes, work with students in laboratories, grade papers and projects or work directly for a professor.

As a TA, you are the key link between professor and students. This gives you the opportunity to observe and influence higher-level decisions about course design and content, as well as the opportunity to maintain daily, close interactions with students.

in most cases, you will have to take some initiative to make sure that your TA experience provides both the mentorship you hope for and a set of responsibilities you can handle. Clear conversations with the professor you are TAing for can set the stage for both.

Schedule regular weekly meetings with the professor and other TAs to maintain open communication and to iron out course details.

Invite the professor to watch you teach and give you feedback on your teaching performance.

<https://teachingcommons.stanford.edu/grad-support/grad-teaching-development/teaching-assistants-role>

webct vs canvas LMS

<http://itg.emerson.edu/word/canvas/how-canvas-is-different-from-webct/>

write about teaching assistant experience

work with professors, read standford TA guide one more time. Examples where I do office hour and lab with students, the approaches I used and the good results that I got.

At ATT, design training courses for new and junior employees. List the training courses that I have created, and focus on one courses design and execution that showed my approaches and good results.

Bottom line, show my teaching philosophies with specific stories and good results.

At University of Georgia, I have had the following teaching experiences:

1. Teaching assistant for Algorithms

Homework grading, Office hour tutoring

1. Teaching assistant for software engineering
2. Teaching assistant for Database Management

**Teaching Statement**

<<<Teaching philosophy>>>

Teaching is two-way communication between teacher and students. The first step is to understand the students, their existing knowledge and skills and their goal to taking this course. And students existing knowledge and learning style could be varied. A teacher should get this info through survey and question. The syllabus will be designed consideration of their existing knowledge, interests and learning styles. During lecture, I will 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. Then motivate students by telling them 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 in the entire process.

Teaching assistant for Introduction to Computing and Programming

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 in variety 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. A program takes input and provides output. 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 problem. Once introduced to thinking and formulating problem in computing way, most students got very excited feeling 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.

Teaching assistant for Algorithms

Algorithms course is a fundamental course of computer science. I showed students its importance of computer science scope. Algorithms can be everywhere, from os, network, database design, AI and machine learning. I gave them overview of the common algorithms and their applications. 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 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.

Training and mentoring junior employee at ATT labs.

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

Research statement

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.