# *Be A Good Helper*

— a recall to Fall 15'–Fall 16' TA journey

Dachao Sun
School of Computing, Clemson University

## *Preface*

An academic semester usually gets more intense and rises to the climate as it apporaches the final month/exam week. The same is true for this semester, except for me, a *graduating* graduate student with his fifth semester being closed up. By "graduating" I mean there is a feeling of getting closer to finishing but still concerning about the ongoing tasks. For a student who has enjoyed studying and learning aside of smart professors in the academy, this is to a little degree like an elimination game in some sports series matchup.

And I am kind of like such a student.

For a few friends I've met in the past two years, he/she who was in the "graduating semester" either was busy with publishing the last research paper along with a dissertation defense (i.e. a PhD candidate), or wrapping up the last one or two courses and getting suited formally for job interviews once in a while (and, significantly harder to make an appointment with to work on team programming projects). Since I insist on choosing to finish with a thesis for Master's, was not done with courses (had not taken any software engineering one), and definitely would be going to work after graduation, I literally am doing a little bit on both of the two sides above (sounds presumptuous).

But, I decided to engage a big part of time in something slightly different: to assist in the undergraduate course, what I'd been hornored to work on. Although it would be the third concecutive semster I was assigned with such work, this meant fundamentally different to me, on many levels. Even at the point I am writing this article in retrospect, it still feels special and precious, and I remain proud of what has been done and how far the students have improved, from the first school week in August. The motivation of this article comes from a conversation about a month ago with Aubrey Lawson, one of my graduate student and smartest peers, that I would write a summary of thoughts on the TA experience at the end of semester to share with her and all graduate students. So I finally decided to start drafting and writing this summary-like article based on my three semesters of TA experience, for which may be a reference to other TAs who just get started and want to improve their skills, or who felt more separated from their peers than being as a team of people who, in together, serve for the one undergraduate student body at School of Computing.

Again, this short article records my thoughts (from mistakes made) on how to *improve* assisting in teaching. Strategywise, I changed from a lengthy emotional retrospect (which I used to be good at) to a pragmatic experience-sharing article which I wish I would have known two years ago. I hope some of the content could be helpful to you.

## 1

## *Goodness > Exellence*

I was overly excited when I first got the teaching assistant work in Fall 15' to grade a 400/600 (undergraduate-graduate combined) course which was my favorite among courses taken by then. Pretending to be calm and professional, the plan in my head was about generating beautiful grading reports to the students, where they could read the serious comments and considerate, encouraging words from me, a more-senior student who "did a good job" in this course a year before. I started spending hours designing template and copying my grading notes into the template one student at a time (since I didn't know how to script-inject text into LaTeX).

After the first programming assignment, I was exhausted and confused: it exceeded (almost doubled) the assigned hours of work, which suck up some of my study and research time. Even worse, students seemed not impressed and the instructor did not buy it either (plus there were typos found by the instructor). In a following (voluntary) lab session, the inevitable happened: few other calculation mistakes were found by the instructor. "You should really let the computer calculate for you!", the instructor then hand-created a *self-explanatory* column named "Total_computed" in my grading spreadsheet, which basically told me the "total" score for each student is "computed" with an equation, not visually added up by humans.

Gradually learning something out of the mistakes, I had a conversation with a senior graduate labmate, who TAed the same course two years before, about how to be more effective in this role. Not fully convinced of the problems, I described to him the several scenarios when I messed up. Intead of telling me how terrible those mistakes were, he said only one sentence which completely dissolved my stubbornness,

"You've got to do it *right* first, then it became possible to do it *well*."

To me, it was the most important (and word-saving) piece of advice regarding TA work. Yes, there is passion and potential fun in assisting a course and helping out students, but it is also your *job*, and a job by definition needs to be done in the right way. The baseline of the right way of being a TA, such as a grader, is **being consistent with the syllabus/rubric/grading judgment across students in the entire class, as well as implementing the instructor's intention** (as oppose to being overly creative). Not until reaching such a state without feeling clumsy should we proceed to propose extra ideas or offer extra help to the course. I put in the subtitle the word "goodness" for "doing a right job" and "excellence" for "doing a great job", where ">" doesn't necessarily mean "more important than" but rather "**comes prior to**".

## 2

## *"Not the worst TA": Ask for Feedback Regularly*

Throughout that first semester of being a TA I was both working hard as usual and feeling a little nerverous, due to the lack of reference from previous experience (and mistakes made..). But since I had been a person who is confident on the quality of fulfilling a job and always wants to improve, one day after a concise discussion on rubric with the instructor, I boosted my courage to ask for feedback, "How do you think the way I've been working so far? Are there things I need to fix or improve immediately?"

"No, you're doing fine," said the instructor, who then briefly recalled how a few TAs worked for this course in the past few years. Those words did not matter that much since it sounded to me as good news, as if I did better than all of them (how presumptuous, but that was how I thought!). But the end of sentence waked me up:

"$\cdots$, you're not the worst TA I ever worked with.", said calmly, indifferently the instructor.

Yes, not the WORST — the fairest evaluation I could receive by that point. Honestly speaking, I did feel upset at that moment, but I quickly realized that it was far more good than harm for me to know (roughly in the middle of the semester) **the *truth* about my performance** in this work. Such knowledge is the basis and motivator of refining approaches and correcting what used to be done wrong.

## 3 *Dealing with Plagiarism*

It is very simple: **no mercy, and strictly follow the procedure**.

In real world, it is hard to have no student who cheats in a class. The only and most-effective solution is to (1) be 100 percent consistent and synchronized with the instructor, (2) strictly (and simply) follow the procedure at department/university level.

## 4 *Work Hard, and Learn From Senior TAs*

When assigned to assist the course on Operating System, I was very uncertain about what challenges would come up because I had never taken a course in OS before (which I should), and I told this to the instructor at our very first meeting so that any adjustment could be made at the earliest moment. The inconfidence on the material made me force myself to be more engaged and focused on the job itself: I audited lectures as often as I could (it was an intensive, challenging and very fruitful course, and lectures started at 8 a.m.!), took notes on quizzes and did most gradings on time.

As we (me, the instructor and the other more senior TA) were grading the mid-term in a pipeline-like manner, I found myself being lagged: reading different styles of written English and the materials might matter a bit, but the major reason was **the lack of decisiveness and professionalism**. The other TA was grading with a more consistent pace, and had less irrelevant talk or questions on points to be assigned than I did. Due to my inefficiency, we did not finish the two-hour plan set ahead and the other TA and I had to continue grading after the instructor went out the office for other stuff. Although it wasn't a big delay and they didn't blame a single word on me, I felt embarrassed and ashamed. The attitude and discipline on efficiency and effectiveness was what I sliently learned from the senior TA and the instructor.

## 5 *Rules vs. Exceptions*

There are certain policies and rules throughout the implementation of a course: regarding late submissions, quiz remedy, double-checking grades and so on. And of course, different

schools, courses and instructors have different interpretations on where the course should locate along the "rule-exception axis". Written syllabus is always a good "hardcore" reference, to which TAs can regularly check whether a penalty should be considered for exception (almost always with the instructor involved), or does not worth a second look at all (yes, not all emails arguing about grades need to be replied).

It is necessary and safe, in my opinion, for TAs to **always put themselves at the extreme of "rules"**, i.e. assuming no exception is possible (which is usually written on syllabus and handouts). This is because when implementing the syllabus and rubric, TAs (here graders) should act mechanically, independent on any other factor. Based on such assumption, adjustments may be made at the discretion of the instructor (as there is often a policy for students to forward such requests directly to the instructor).

## 5
## *On Approach: Explain vs. Direct*

One of the trickest facets in helping students learn is how to "invisibly" direct them to find the answer (or discover new solutions) on their own, especially in courses that are thought-provoking and usually project-driven. We of course should always address the importance of building the *fundamentals*, on which as a TA I tend to *explain* by asking questions like "what does a node in graph represent in this problem", and "what is the optimization goal of this algorithm".

With a solid knowledge of the problem definition and the underlying mathematical model, it is the student's job to figure out the rest. I, as a graduate student who is few years elder than the undergraduates, usually found it was good for me to "organize the problem" or "review the definitions" with them (which senior guys should be better at), while after which they did more effectively and creatively in solving the problem than I could and what I expected them to be. Personally I think the former corresponds to some good habits in thinking and tackling problems, which takes practice and correct guidance.

In an assignment of the Algorithms and Data Structures course, students were to solve an optimization problem (achieving the maximum total "tastiness" of selection of candies!) using more than three techniques (e.g. greedy selection, dynamic programming etc.). The content of various techniques had been covered in the recent lectures, and it was time to test it out. Suprisingly, it was not the idea of a certain technique, but how to formulate/model this actual problem into the algorithm that became the hard part to many students.

Thinking of this, I was telling myself that I should *not* deprive the appreciation process of understanding the algorithm (optimization technique) of students, but would rather help on the part of formulating the actual problem (selection of candies) into the model, because that is more generic and independent from the algorithm and what they can improve through practice. So in the conversation with a student during the help hours, we looked at his on-building workflow (which he got hung up), and I proposed what if the objects were stored in the other way, on which he did quickly modify and got the algorithm worked correctly.

The term "invisible" I put in the first paragraph is borrowed for the same meaning from the quote "designs are so good that they are invisible". At the year of 2016, smart, advanced but neat design teaching materials and means should be, I guess we should all agree with, what we are striving for. It is just like teachers' using "head fake" to keep students engaged for effective learning, and visual designers's looking for better encoding to tell compelling stories.

## 5
## *Be Honest: What You Know and Don't Know*

This is an item that is also supposed to be simple and implicit, but from what I viewed myself and some other TAs in practice it is worth being a regular reminder to us. To a TA, there are three two types of questions asked by the students:

1. the ones you know clearly well, and are able to give an effective answer;
2. those you don't.

You may argue that there should be a third category consists of those in between 1) and 2), and that is reasonable. But do we need that category when trying to answer a question? And how should we answer in each of the three cases? To me, it is much more important to convey the right idea than persisting superficial dignity. A good pipeline might be,

```
HANDLE_QUESTION (Q):
    if 1) then
        Answer Q;
    else if isInstructorHere == true then
        Consult the instructor;
    else
        sleep(60);
        HANDLE_QUESTION (Q);
```

And then we can use preparation hours (usually there should be some allocated within the work/our own schedule) behind the scene to gain familiarity in the concepts and materials.

### *Summary: Assistant $\neq$ Teacher*

Being a teaching assistant will be a lot of hard work, challenge and fun. I guess a reasonable feedback-loop that is worth checking when we are not sure about how much/less we have done is to go back to the definition of this job: *assistant*. Some suggestions exaggerate this role by discussing how to teach, and ignore the fact that TAs are *not teachers*: it is the instructor who (often times much more learned than we are) design, teach and refine the course. As the definition of "assistant" given by Webster is simply "helper", we as TAs would better think of our job as to *help*.

To wrap up, there are graduate TAs who are really skilled in communication and the course materials, and they normally contribute in a much smoother way beyond what is mentioned above. If you are one of them, students will very likely appreciate and learn a lot out of the experience. If you are not yet one of them, along the way getting there let's start from doing it right.

Enjoy the journey!