

Meeting - 10/11/22

Jincheng He (jinchenh@usc.edu, 213-246-9074)

Program Status

Overall, I started my Ph.D. program in Spring 2020, and this semester (Fall 2022) is my 6th semester.

1. **Coursework: All requirements are met.**

4-unit 600-level courses: CSCI-610 (Program Analysis), CSCI-662 (NLP), CSCI-670 (Algorithm), CSCI-699 (Research Agenda). The rest units can be filled with 790 DR or 794 dissertation.

2. **TAship: All requirements met.** I TA-ed for 561 (AI), 577A (SE), 510 (SE Economics), 585 (Database) in previous semesters.

3. **Qualifying Committee: (confirmation needed)** I have contacted the following faculty member, and they are willing to be on my committee:

- G.J. Halfond
- Sandeep Gupta (EE)
- Chao Wang
- Aiichiro Nakano
- Neil Siegel (Adding Neil needs additional approval from advisor, as he is not a USC tenure track faculty member.)

Research Status

1. **Directed Research Teams (590): I am managing the CSSE DR portal and leading two teams.** The CSSE research portal has still been functioning, helping 3 CSSE Ph.D. students to hire M.S. and B.S. students (as unpaid interns, or for units) for their research, and I am maintaining this portal right now. Lizsl knows more about course registering and whether this is under some faculty's name. And I am right now leading two teams, one for research, while another for maintaining CSSE website.

2. **Research Direction: Categorize commits by purpose, and investigate how different types of commits impact software quality.** Ongoing research tasks include:

- (a) **Classification:** We have a set of 1914 commits categorized manually with a refined Taxonomy. For manual classification, we read commit messages and code diffs, as well as cross-validating results within our team.
- (b) **Software Quality:** to evaluate quality, we used compilability, software quality metrics (from SonarQube, PMD and FindBugs/SpotBugs), and I'm looking into using other metrics/assessment to evaluating quality. For example, it's possible to find new analysis tools, or directly get architectural information by applying program analysis techniques (I took Dr. Halfond's program analysis course in spring 2022 for this).
- (c) **Automation:** I'm looking into approaches to automatically categorize commits, based on commit messages, which is typical, or based on code changes/diffs, with the help of code summarization tools. (I took Dr. May's Advanced NLP course in Fall 2021 for this).