ColorMyGraph: Student Proof Analysis and Verification

Adam Blank

My research has been focused on developing a novel way of helping introductory Theoretical Computer Science and Discrete Mathematics students pick up the skills necessary to succeed in the future more quickly and successfully than usual. Under Luis von Ahn and Klaus Sutner, I designed, developed, and maintain ColorMyGraph which is a full course infrastructure specifically dedicated to discrete mathematics and algorithms assignments to be submitted via LaTeXed documents. It handles assignment distribution, submission, and grading.

A common problem with the traditional Homework, Solutions, Feedback cycle that, in practice, most students do not benefit from either the solutions or the feedback. If students don't learn in the second two-thirds of the cycle, then their internal models of how to think about proofs remain unadjusted–and often incorrect.

Almost all Computer Science and Mathematics departments have a Freshman course dedicated to introducing students to a proof-based way of thinking. At Carnegie Mellon University, there are two such courses: *21-127: Concepts of Mathematics* and *15-251: Great Theoretical Ideas in Computer Science*. Until recently, both of these courses have employed the traditional methods described above to teach students how to construct and write proofs. These methods have a wide distribution of effectiveness, but there is almost always a large group of students who struggle through the course and ultimately fail to grasp the "new way of thinking" despite their best efforts. This group of students is the target population for increased performance. Even when solutions are released, exemplars of this group are incapable of fully understanding them. Furthermore, personalized feedback is often not good or non-existent, because it is incredibly time-consuming to read, understand, and give feedback on 200+ proofs on a weekly basis.

ColorMyGraph introduces a new avenue of simultaneously evaluating student understanding and reducing the course staff's busy work. These goals could be accomplished by assigning the students some of their peers' proofs to evaluate for correctness. In turn, a portion of their grade in the course would be based on the correctness and thoroughness of their evaluation of the proofs they were given. In this fashion, students would get valuable experience both reading and verifying proofs that are not their own. Furthermore, since students are doing some of the grading, the course staff could focus more on other aspects of the course such as helping struggling students.

While most students are not a priori able to accurately check the correctness of mathematical proofs, the Teaching Assistants "prime" each question by pre-completing a subset of the grading and telling the system common approaches and mistakes. The system facilitates this in such a way that even novice students can effectively make a contribution. To effectively correlate information among verifications, the system must standardize what is expected in a verification. The system has four sections that make up a verification: Summary, Plans, Judgements, and Understandability.

A **summary** is a short description of a proof that details what the proof does that might be different than another similar one. **Plans** are high-level, one sentence explanations of possible approaches students could make to a problem. **Judgements** are descriptions of individual mistakes and successes that are distilled from student submissions. Since a large percentage of students make the same mistakes in their responses, a list can easily be created and maintained that students can select from. The system asks course staff to pre-populate many of these based on doing a subset of the grading before students begin. The **understandability** section is about ascertaining information about both how well the student reading the proof was able to understand it, and also information about how well the argument was conveyed in the writing.

These four pieces force the students to get a better understanding of the question they answered and learn. ColorMyGraph has been piloted in 15-251 this semester (˜100 students) and last semester (˜200 students) and 21-127 this semester (˜400 students), and there are plans to pilot it in the Carnegie Mellon algorithms course next semester.