RICE UNIVERSITY

# Improving Peer Evaluation Quality in Massive Open Online Course

by

**Yanxin Lu**

A Thesis Submitted
in Partial Fulfillment of the
Requirements for the Degree

**Master of Science**

Approved, Thesis Committee:

Swarat Chaudhuri, Chair
Associate Professor of Computer Science

Christopher Jermaine
Associate Professor of Computer Science

Joe Warren
Professor of Computer Science

Houston, Texas

April, 2015

# ABSTRACT

Improving Peer Evaluation Quality in Massive Open Online Course

by

Yanxin Lu

Massive Open Online Courses (MOOCs) have gained much attention across the globe recently. Students are able to receive free education through online courses. While some MOOCs use automated grading to assess student performance, some other MOOCs depend on peer grading, because automated grading is not applicable in some scenario such as grading papers. In this thesis, we consider the problem of low quality of peer grading in those MOOCs. Especially, Introduction to Interactive Programming in Python course (IIPP) that Rice has offered for a number of years on Coursera has suffered from the problem of low-quality peer evaluations. We propose our solution to improve the quality of peer evaluations by motivating peer graders. We ask two specific questions: (1) When a student knows that his or her own peer grading efforts are being examined by peers, does this knowledge alone tend to motivate the student to do a better job when grading assignments? And (2) when a student not only knows that his or her own peer grading efforts are being examined by peers, but he or she is also given a number of other peer grading efforts to evaluate (so the peer graders see how other peer graders evaluate assignments), do both of these together tend to motivate the student to do a better job when grading assignments? We designed a web application where thousands of students were asked to examine mutiple peer grading efforts. According to the results, we find a strong effect on peer evaluation quality simply because students know that they are going to be studied using a software that is supposed to help with peer grading. In addition, we find strong evidence that by grading peer evaluations students tend to give better peer evaluations. However, the strongest effect seems to be obtained via the act of grading others' evaluations, and not from the knowledge that one's own peer evaluation will be examined.

# Contents

# Illustrations

# Tables

# Chapter 1

# Introduction

Online course providers such as Coursera, Udacity and edX have gained much attention across the globe since 2012 [1]. Millions of people have participated in those online courses and Massive Open Online Course (MOOC) was coined to refer to online courses with large amount of participation [2]. Free access to courses provided by top universities around the world is always attractive to most people who want to learn, especially those who have no opportunities of entering the top universities.

While free online courses are appealing to most people, several challenges exist in the context of MOOCs including low completion rate [3], cheating [4] and low quality of peer evaluation. In this thesis, we mainly focus on the problem related to peer evaluation, especially the problem of low quality of peer evaluations. In some MOOCs where automated grading is not applicable such as grading papers, peer evaluation becomes one of the most important method to assess student performance. Rice has been offering a MOOC called “An Introduction to Interactive Programming in Python” (IIPP) designed to help students learn the basics of writing simple interactive programs using Python. IIPP, in which peer evaluation is one of the major grading scheme, has suffered from the problem of low quality of peer evaluations. In this thesis, we present our method of motivating peer graders to improve the quality of peer evaluations. Notice that even though the study was done in the context of IIPP, the results apply to any course where peer evaluation is used.

The idea of using incentives in the context of MOOCs has been explored already [5, 6, 7] in which the goal was to test whether an incentive could bring a certain effect. In our case, we want to answer the question: when a student knows that his or her own peer grading efforts are being examined and they are able to grade other peer evaluations, do those tend to motivate the student to do a better job when grading assignments?

We conducted a controlled experiment where thousands of students were asked to grade the qualities of multiple peer grading efforts represented by peer evaluations. According to the results, we find a strong effect on peer evaluation quality simply because students know that they are going to be studied using a software that is supposed to help with peer grading. In addition, we find strong evidence that by grading peer evaluations students tend to give better peer evaluations. However, the strongest effect seems to be obtained via the act of grading others' evaluations, and not from the knowledge that one's own peer evaluation will be examined.

In the following sections, we outline the background for MOOCs including the history of MOOCs and the major components of a MOOC. Then we will then discuss peer grading and the problems it introduces. Finally, we propose our solution and then summarize the result we find.

## 1.1 Massive Open Online Course

Distance learning appeared back in the eighteen century where students received weekly mailed course material [8]. With the advancement of technology, remote classes were offered through radio station [9], television [10] and emails [11]. The term *MOOC* which stands for Massive Open Online Course was coined in 2008 by Dave Cormier of University of Prince Edward Island [12]. ALISON, an e-learning provider found in Ireland, is often seen as the first MOOC which systematically made interactive learning resources available online [13]. By February 2014, ALISON already have had 3 million users.

In 2011, Stanford University launched several online courses. The first one was *Introduction to AI* by Sebastian Thrun and Peter Norvig. Over 160,000 students enrolled. Two more MOOCs were also launched by Andrew Ng and Jennifer Widom. Sebastian Thrun and Peter Norvig later founded Udacity and Andrew Ng and Daphne Koller launched Coursera [14]. Subsequently, partnerships were formed between Coursera and several top universities including University of Pennsylvania, Princeton University, Stanford University and The University of Michigan. On the other hand, MIT and Harvard created the non-for-profit platform called edX [2]. Several universities including University of California, Berkeley, University of Texas, Wellesley College and Georgetown University joined the group. Because of the appearances of popular online course providers such as Coursera, Udacity and edX, 2012 was seen as "the year of MOOC" by New York Times [1] as they had gained much attention and over hundreds of thousands of participants enrolled in their MOOCs. In 2013, over hundreds of courses were offered in edX, Coursera and Udacity. Since then, more and more universities in Asia, Europe, Australia and Latin America have started their own MOOCs in languages other than English. Overall, MOOCs provide those who do not have opportunities for attending elite schools an excellent platform to receive higher education.

### 1.1.1 Learning in MOOCs

In order to enroll in MOOCs, students only need to search for courses through providers and simply click a button in a web page indicating that they decide to take the course. Staffs then notify students who signed up for the course before launch by email when the course starts. During each week, new video lectures as well as homework assignments become available. In order to gain points, students need to finish homework assignments before deadlines similar to finishing homework in a traditional classroom. After a certain period of time, examinations will be given and students are again required to complete the exams before deadlines. At the end of class, students with grades higher than a threshold will be awarded a certificate of completion signed by the course staff members indicating that they have finished the course and also learned the material. Here, we describe the components in a regular MOOCs offered in one of the popular providers.

**Video Lectures** The most fundamental element of a MOOC is a set of video lectures in which the instructor of the course explains the course material. These video lectures are usually 8 to 12 minutes long, and most video lectures come with subtitles. Some videos might pause for a quiz in order to ensure students understand the material. The answers and explanations for the quizzes are all constructed a priori.

**Homework and Examinations** Similar to a traditional class, homework and examinations are assigned in order to make sure students understand the material. Homework and exams usually consist of a set of multiple choice questions drawn from a larger set of question pool. If multiple attempts are given, the system will make sure the questions are different for each attempt in order to prevent students from memorizing incorrect answers. After each attempt, students are able to review the questions they answers along with correct answers and explanations. Usually homework assignments are not timed, but exams are often timed.

**Programming Assignments** Usually in MOOCs on the field of computer science, programming assignments are given. Two methods are currently being used to grade programming assignments. The first method is automated unit testing. Students upload their source code which will be executed later behind the scene, and the scores are determined by the number of test cases passed successfully when running their programs. The second method is peer grading. Peer grading is used when unit testing is not applicable. One example is program with graphical user interface (GUI). A rubric is usually given which focuses on testing the behavior and graphical element of programs. This type of programs is usually assigned to one or more students and they are responsible for grading the program according to the rubric. In order to make sure the accuracy of scores, one program is grading by multiple students and the median of those scores is taken as the final score.

**Discussion Forums** Discussion forums are provided for students to seek help from peers and course staffs. Discussion forums are divided into multiple sub-forums and students can post their questions and comments about the course in each sub-forum and others can also post their answers or comments. Even though the content in the forums is not restricted to be related to the course, course staffs and students can submit a spam report if inappropriate content is found.

## 1.2 Peer Grading

Even though MOOCs have gained enormous attention and many students have successfully learned new material, MOOCs, as a new form of distance learning, are facing several challanges [3, 6, 4], and one challange is grading. The largest MOOCs have on the order of tens of thousands of participants who actually complete some fraction of the assignments and examinations. Clearly, grading at this scale is beyond even the largest team of instructors and TAs.

One solution to the problem of grading so many submissions is for the instructor to design assignments and examinations in such a way as to ensure that student work can be automatically graded. For example, multiple choice examinations can be given. For another example, in a computer programming class, student programs can be put through a test suite by an automated grader, and the number of test cases passed can be used to assign a student a grade.

The obvious problem with automated grading is that there exist courses for which purely automated grading is not possible. an example that we are intimately familiar with (and the course that is the subject of the experimental study described in this paper) is a course on interactive game programming in python*. Since games by their very nature require a user to play them in order to test correctness, it is exceedingly difficult to utilize automated grading to score student-constructed games. another example is a mathematics class where the construction of proofs is a key part of the course; fully automated grading of student proofs is not feasible at this time. courses in the social sciences or languages are also not amenable to fully automated grading.

### 1.2.1 The Problems of Peer Grading

The common way to scale up grading that cannot be automated is to rely on *peer grading* [7, 15, 16, 17, 18, 19, 20], where student work is distributed to other students in the class to be graded. Peer grading has many benefits. Not only does it offer a way to crowdsource the grading of student work so as to achieve virtually infinite scalability, but there are pedagogical benefits to peer grading as well [21]. Students

*see https://www.coursera.org/course/interactivepython.

who grade other students' assignments are forced to carefully consider the validity of a wide variety of solutions to a problem, and can benefit from grading other students' work. As a result, commonly used MOOC platforms such as Coursera [14] offer built-in facilities to support peer grading.

Peer grading, however, is not a panacea. Our personal experience is that in a MOOC, low quality peer grading can be a problem. The main reason for poor grading seems to be a simple lack of effort, rather than inability or maliciousness. On Coursera, for example, students are assessed a 20% penalty when they fail to peer grade. One unfortunate (and common) student response is simply to give all peers perfect scores. Consider the IIPP course that we have offered for a number of years on Coursera, and which serves as the subject of the study described in this paper. On the two assignments central to our study (Stopwatch and Memory), in those cases where we were able to automatically find an error that should have resulted in one or more points being deducted, the peer grader gave full credit 53% of the time (4,168/7,855 grades given). In contrast, in those cases where we could find no error automatically, only 2% of the time (292/16,427 grades given) did the grader take off more than one point.

This sets up the question that is at the heart of the thesis:

*How can an educator running a MOOC motivate students to do a better job of peer grading?*

Our goal is to identify a simple and practical method for motivating students to perform high-quality peer grading, and to then rigorously test this method in a controlled experiment, in a real MOOC.

While we are primarily interested in MOOCs, we point out that answering this question definitively would have implications beyond MOOCs. For an obvious example, consider the problem of ensuring high-quality peer reviews of submissions to a competetive, academic conference. Reviewers agree to review submissions out of a sense of obligation or because they want to be associated with a prestigious conference, and often lack a strong motivation to do a good job. Methods that work well in a MOOC might also help in motivating conference reviewers.

### 1.2.2 Motivating Peer Graders

One method for motivating peer graders that seems to have entered into the folklore is the method of "sentinels". While it is unclear if anyone has actually employed this method in practice, the idea is simple. Instructors or TAs pre-grade a few assignments (called sentinels), which are then added into the general population of assignments. Since we know something of the ground-truth grade for the sentinels, it is possible to identify graders whose grades differ significantly from the sentinels. Such off-target grading efforts could be identified and examined, and students would typically be given some sanction for doing such a poor job grading the sentinels.

However, there are some inherent problems with the use of sentinels. Based on our experience in delivering several MOOCs, a substantial subset of the students enjoy discussing their peers' work in the class forums. Due to this discussion, we feel it would be very difficult to hide the existence of sentinels from the students and that, eventually, the sentinels themselves would be a popular topic for discussion. Of course, this knowledge would diminish the effectiveness of sentinels in motivating student effort during peer grading.

Another question concerning the use of sentinel is what type of penalty should a student receive for doing a bad job on a single peer evaluation? In general, the idea of punishing students for poor grading seems counter-productive, especially when many students are taking the MOOC are doing it to simply learn the topic. How does one punish a student who simply wants to learn the course material?

Perhaps a better approach would be to motivate students to do a better job. In this paper, we investigate a very different method for motivating peer graders. Rather than punishing (or rewarding) students for their peer grading, we examine the utility of crowdsourced examination of peer grading efforts. We ask two questions:

1. When a student knows that his or her own peer grading efforts are being examined by his or her peers, *does this knowledge alone* tend to motivate the student to do a better job when grading assignments?

2. When a student not only knows that his or her own peer grading efforts are being examined by his or her peers, but he or she is also given a number of other peer grading efforts to evaluate (so the students sees how other peer graders evaluate assignments), *do both of these together* tend to motivate the student to do a better job when grading assignments?

Crucially, there is no punishment or reward under either regime. There is only the knowledge that ones peers are going to examine ones grading efforts (case 1), and in addition, an exposure to how other peer graders have evaluated assignments (case 2).

The idea of "grading the graders" is not revolutionary. In the context of peer review of scientific papers, this is often mentioned as a possible mechanism for ensuring high-quality peer review. However, most or all of the ideas along these lines rely on logical argument or thought experiment to demonstrate utility. Our efforts differ in that not only have we designed and implemented a system for grading the graders, but—far more importantly—we have also designed and run a large-scale, controlled experiment to evaluate the utility of the approach. Such studies should be considered mandatory as the community attempts to figure out the correct way to run a MOOC.

Our specific contributions are as follows:

1. We describe two easily-implementable and easily-deployable methods for motivating peer graders in a MOOC. These methods have the advantage that they require only moderate levels of effort from the MOOC community.

2. We conduct a carefully designed, controlled experimental evaluation of these methods in the context of a popular interactive (game programming) MOOC. We assert that such a controlled experiment is the only reliable way to collect data supporting or refuting the utility of a particular methodology in the context of a MOOC.

3. We find evidence that there are significant differences among the various study groups, even among the subset of students who are motivated enough to sign up for such a study.

### 1.2.3 Summary of Findings and Recommendations

Surprisingly, we found little evidence that simply knowing that one's grading efforts were going to be graded results in a superior grading effort. That is, those study participants who did not grade others, but only had their grading efforts graded, did not perform much better than those in the control group.[†] However, those who participated in the full regime—students who had their grading efforts graded *and* graded others' grading efforts—not only did a better job grading during the study

[†] As an aside, this would seem to argue *against* a sentinel-based approach, which by its nature relies on monitoring of grading efforts to increase grading quality.

(which lasted for two assignments), *but the positive effects were lasting.* That is, those who participated in the "grading the graders" regime continued to do a better job than those who did not participate in the full regime, even after the study ended.

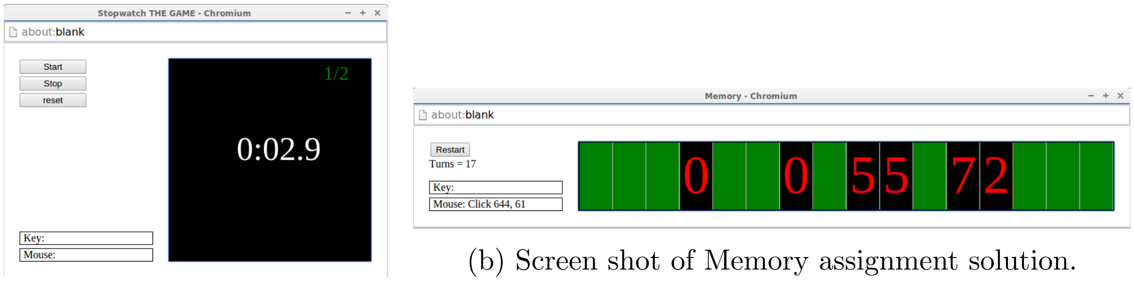
Thus, the key to achieving better grading results seems to be actually seeing how other people grade, and not simply knowing that grading efforts are being monitored. As discussed in Chapter 7, we conjecture that actually seeing that other students put in the effort to do a good job helps provide the motivation that students need to do a good job when it is their turn to grade. In this case, simply showing examples of good grading is not enough; *students must be shown evidence that their peers are actually producing such high quality efforts.* This is exactly what the "grading the graders" regime does. Thus, our recommendation is that MOOCs that rely on peer grading should utilize the "grading the graders" regime for at least one or two assignments at the beginning of the class; our study seems to indicate that this should have a positive effect on grading quality.

# Chapter 2

# Background

IIPP is a 9-week course designed to help students with very little computing background learn the basics of writing simple interactive programs (games of various types) using the Python programming language. No prior programming experience is assumed. The course covers the basic syntax and semantics of Python, as well as object-oriented programming. Students learn the material of the course by completing a series of "mini-projects" including games such as Pong, Blackjack and Asteroids. We administered the study in the Fall 2013 session of the course. In that session, 120,000 students enrolled and 7,500 finished the course.

One of the challenging aspects of designing a Python programming MOOC is making sure that thousands of students around the world are able to run Python programs on their own machines, whatever the operating system and machine the



(b) Screen shot of Memory assignment solution.

(a) Screen shot of Stopwatch assignment solution.

Figure 2.1 : Two IIPP programming assignments that are the subject of the study.

students are using. Further, since IIPP relies on peer grading, these programs need to be completely portable—that is, it needs to be trivial for one student to access and run another student's code during grading. To facilitate this, IIPP makes use of a browser-based programming environment called "CodeSkulpter" [22]. In practice, any student with a functional browser can implement and run Python programs using CodeSkulpter. Further, when students save programs in CodeSkulpter, the source code is saved to cloud-based storage. Any instructor or peer grader with access to the URL that points to the saved code can open and run the code, meaning that submitting a program is equivalent to submitting a URL.

### 2.0.4 Assignments

The study described here concerns two different IIPP programming assignments, Stopwatch and Memory. We describe those programming assignments now.

**Stopwatch.** Stopwatch was assigned during the fourth week of the course. 10,500 students completed the regular version of the Stopwatch assignment, and 2,366 additional students participated in the study and completed the study version of the assignment.

In this project, students write the application whose screenshot is shown above in Figure 2.1a. Students must implement three buttons: a "start" button, a "stop" button, and a "reset" button. The applications implements a simple game where a player presses the start button, which starts a timer that is accurate down to a tenth of a second. The player then attempts to press the stop button when the tenths position of the timer is zero (that is, they attempt to hit "stop" at a whole second). The application should display the number of times that the user does this correctly. For example, displaying "3/4" means that the user has successfully stopped the timer

```
import simplegui
# define global variables
minute = 0
second = 0
millisecond = 0
time = "0:00.0"
score = "0/0"
wins = 0
attempts = 0

#Function for the game
def score_keeper():
    ...

# define event handlers for buttons; "Start", "Stop", "Reset"
def timer_handler():
    ...

#Ensure there is a zero before the one
def second_string():
   ...

def reset_handler():
    timer.stop()
    time = "0:00:0"
    second = 0
    millisecond = 0
    minute = 0
    score = "0/0"
    attempts = 0
    wins = 0

# define draw handle
def draw_handler(canvas):
     global time
     canvas.draw_text(time, (100, 150), 50, ’White’)
     canvas.draw_text(score, (230, 30), 30, ’Green’)

# register event handlers
def start_handler():
     timer.start()

def stop_handler():
     timer.stop()
     score_keeper()

# create frame
frame = simplegui.create_frame("Stopwatch THE GAME", 300, 300)
button1 = frame.add_button(’Start’, start_handler, 100)
button2 = frame.add_button(’Stop’, stop_handler, 100)
button3 = frame.add_button(’reset’, reset_handler, 100)
timer = simplegui.create_timer(100, timer_handler)
frame.set_draw_handler(draw_handler)
# start frame
frame.start()
```

Figure 2.2 : Source code listing of a submitted Stopwatch submission.

at a whole second three out of four times. A sample, partial listing of a student source code is given in Figure 2.2.

**Memory**. Memory was assigned at the beginning of the sixth week of IIPP. 7,600 students completed the regular version of Memory, and 1,746 of the 2,366 students who completed the study version of Stopwatch also completed the study version of memory.

In this project, students build an game which first displays eight pairs of cards face down. A move consists of the player flipping over two cards. If they match, the game leaves them face up. Otherwise, they are flipped back face down. The goal is to flip all the cards face up in the minimum number of moves. Figure 2.1b shows a screenshot of of a completed Memory assignment.

### 2.0.5 Peer Grading

Since submissions in IIPP are all interactive programs, they are very difficult to grade automatically, so IIPP utilizes Coursera's peer grading facilities.

Peer grading takes place after all students submit their programs. Students are required to assess five of their peers' programs and their own program. For each program, a grading instruction is presented to the students followed by a series of rubric items. Figure 3.1 shows the rubric supplied to students for Stopwatch.

For each rubric item, students need to choose how many points to assign using a drop-down menu. After choosing a number of points from the drop-down menu, students can provide additional comments for that specific rubric item in a text area right below the drop-down menu. After grading the program, students can give an optional overall feedback to the submission.

**Peer Grading Quality.** We estimate that it takes an average grader about 10 minutes to do a reasonable job of grading a typical IIPP assignment, meaning that the grading load for each student is one hour per assignment. In our experience, some students voluntarily spend more time than that grading, but unfortunately, some students spend much less time. In fact, the amount of time and effort spent grading—and hence the accuracy of the peer grading effort as well as its utility to the students being graded—varies widely.

For just once example of this variance, consider Figure 3.2 which shows an average to above-average grading effort for one particular Stopwatch submission. The grader took off a few points for four of the eight rubric items (giving eight out of 13 points total), and for each of those items, reasonable comments were given.

In contrast, a second grader for this very same submission gave full credit (13 out of 13) to the submission. No comments at all were offered, except for a terse "Very fancy timer :) Great job." under the "Overall" category. Disparities in grading effort such as this are common, and are precisely our motivation for undertaking the study described in the remainder of the paper. We wish to ask the question: How might we motivate students to put in the effort required to produce an evaluation that is of equivalent quality to evaluation depicted in Figure 3.2?

# Chapter 3

# Grading the Graders

At the highest level, the approach we explore to motivating peer graders is to expose the peer graders themselves to grading. That is, not only are peer graders asked to grade other students' submissions, but their grading efforts are themselves then evaluated by peers. Our hypothesis is that if graders know that they will be evaluated, they will be more motivated to submit high quality evaluations.

In this section, we briefly describe the relatively simple software infrastructure that we implemented to evaluate this idea.

## 3.1 Stage One: Grading the Graders

A few days after students participating in the IIPP "grading the graders" regime finish their peer evaluations, they receive an email with a link to a web application. Following this link leads the student to a page where they are asked to evaluate a set of peer evaluations for five random submissions.

Following this link presents a simple web page that has a link to the assignment that is the subject of the various evaluations. A screen shot of the web page is shown in Figure 3.3. The web application allows students to cycle through the various rubric items, one-at-a-time. For each rubric item, six different peer evaluations are shown. The application displays the score assigned by each of the six peer evaluators, as well as any comments. The student who is evaluating the evaluators is instructed to click

| Item | Points | Description |
|---|---|---|
| 1 | 1 pt | The program successfully opens a frame with the stopwatch stopped. |
| 2 | 1 pt | The program has a working ”Start” button that starts the timer. |
| 3 | 1 pt | The program has a working ”Stop” button that stops the timer. |
| 4 | 1 pt | The program has a working ”Reset” button that stops the timer (if running) and resets the timer to 0. |
| 5 | 4 pt | The time is formatted according to the description in step 4 above. Award partial credit corresponding to 1 pt per correct digit. For example, a version that just draw tenths of seconds as a whole number should receive 1 pt. A version that draws the time with a correctly placed decimal point (but no leading zeros) only should receive 2 pts. A version that draws minutes, seconds and tenths of seconds but fails to always allocate two digits to seconds should receive 3 pts. |
| 6 | 2 pt | The program correctly draws the number of successful stops at a whole second versus the total number of stops. Give one point for each number displayed. If the score is correctly reported as a percentage instead, give only one point. |
| 7 | 2 pt | The ”Stop” button correctly updates these success/attempts numbers. Give only one point if hitting the ”Stop” button changes these numbers when the timer is already stopped. |
| 8 | 1 pt | The ”Reset” button clears the success/attempts numbers. |

Figure 3.1 : Rubric given to peer graders for Stopwatch program.

a radio button next to each of the evaluations. The buttons are labeled with “good”, “neutral” and “bad”, referring to the quality of the evaluation. Typically, a student will look through the various evaluations and comments, and if the evaluators all give the rubric item full credit, the student will assign each a score of “good”. If one of the evaluators has taken off some credit, the student will look at the submitted assignment to see if he/she agrees with the loss of credit, and evaluate each of the evaluations accordingly.

After the student evaluates each of the evaluations, the student clicks the “submit”

| Rubric Item | Score | Comments |
|---|---|---|
| 1 | 1 | (No Comments) |
| 2 | 1 | (No Comments) |
| 3 | 1 | (No Comments) |
| 4 | 0 | you forgot use variable timer to stop the timer at reset button. |
| 5 | 1 | You used non-decimal number to count. The numbers for A, C and D was 0-9 and B was 0 - 6.The function format did not pass to the test numbers. |
| 6 | 1 | In the test, i stoped the clock 2.9 and the program showed 2.0 and count 1 correct attempt.I think some problems occurred because the way how you count the number with a non-integer number. |
| 7 | 2 | Update the numbers, but you need to look more deep how your time is increasing because sometimes the clock stop at time but it is not the real time counted. Just put a print time_elapsed at timer_handler an you will can see that behavior |
| 8 | 1 | (No Comments) |
| Overall | N/A | Remember to look more carefully to all the section ”Mini-project development process”. I think almost of all problems came from count time with a non-integer.Review format function an test it : http://www.codeskulptor.org/#examples-format_template.py. At ”Discussion Forum” -¿ Code Clinic you always find great help to understand some things...use more discussion forum. Sorry about my poor english. =[ google translator help me a lot hehehe |

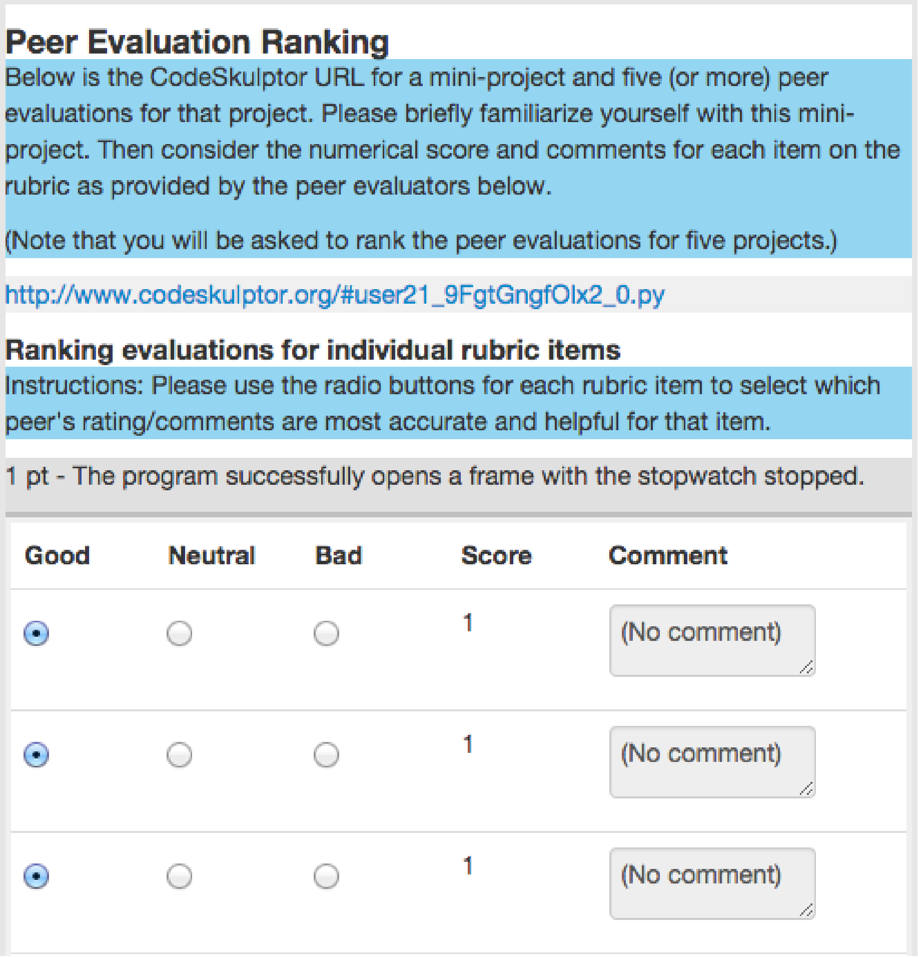
Figure 3.2 : Reasonable quality peer grading effort for a Stopwatch subimssion. This grader gave eight out of 13 possible points for the submission. In contrast, for the same submission, another grader gave full credit, and the only comment offered was “Very fancy timer :) Great job.” as an “Overall” comment.

button to move onto the next rubric item. After cycling through each of the rubric items, a final web page informs the student that he/she has completed the “grading the graders” process.

## 3.2 Stage Two: Examining Evaluations

Some time later, students who complete the “grading the graders” process will receive an email with a link to a web application that allows the student to see how others evaluated his or her *own* peer grading efforts. Since students are required to evaluate six assignments (five other students’ assignments, as well as their own), following the link presents a web page that first lists links to six assignments that the student evaluated. For a particular rubric item, the web page lists each of the six evaluations that the student performed, along with the ratings supplied by those who participated in the “grading the graders” activity. Figure 3.4 shows a screenshot of the interface.

After cycling through the various rubric items, students are then given the opportunity to examine the grading of his or her own assignment. That is, the student can see how those who “graded the grader” viewed the quality of the grades for his or her own submission on the assignment. The student can choose to go to a screen that lists (for a particular rubric item) all of the evaluations of his own submission, along with the number of times that an examiner determined that the evaluation was “good”, “neutral”, or “poor”.



## Peer Evaluation Ranking

Below is the CodeSkulptor URL for a mini-project and five (or more) peer evaluations for that project. Please briefly familiarize yourself with this mini-project. Then consider the numerical score and comments for each item on the rubric as provided by the peer evaluators below.

(Note that you will be asked to rank the peer evaluations for five projects.)

http://www.codeskulptor.org/#user21_9FgtGngfOlx2_0.py

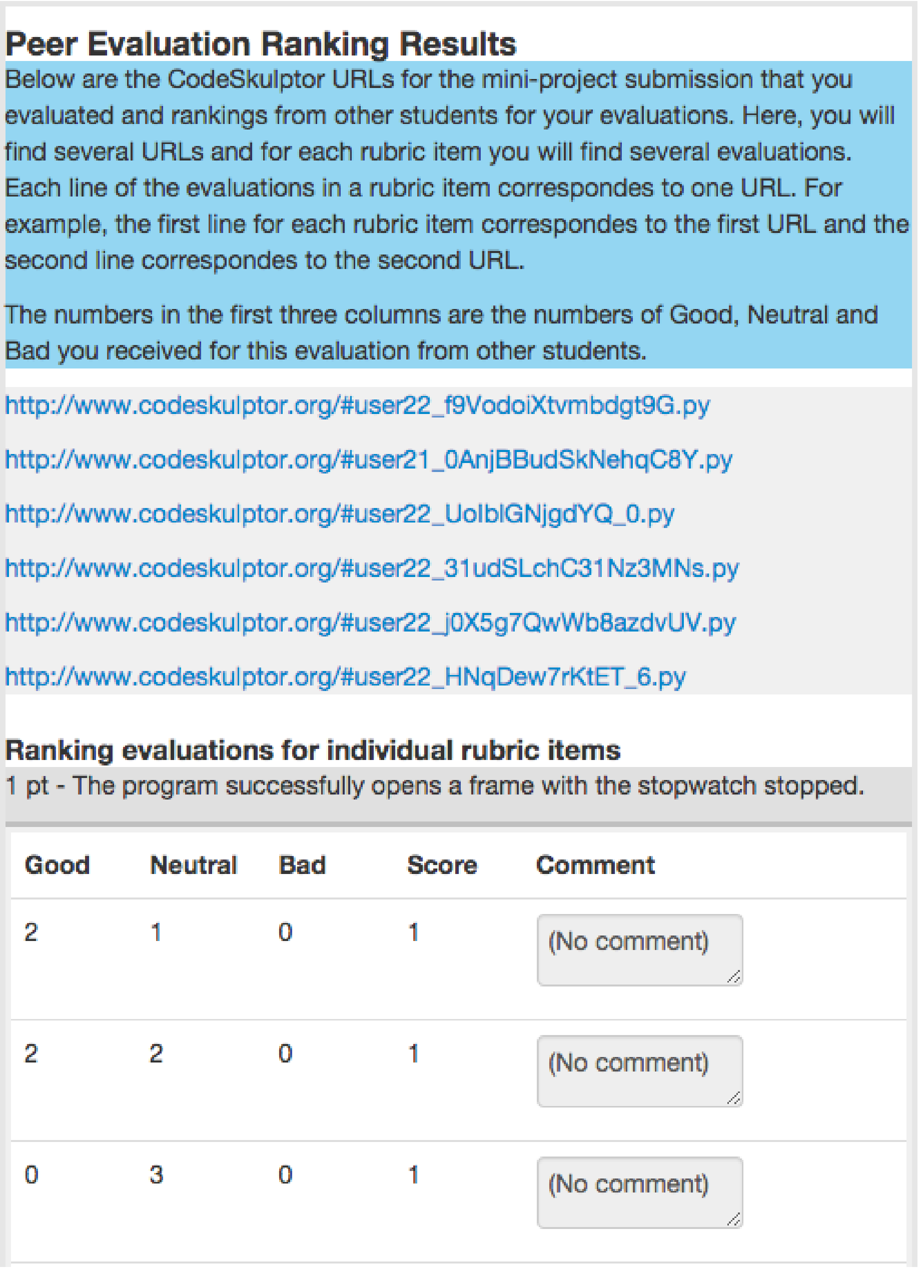
### Ranking evaluations for individual rubric items

Instructions: Please use the radio buttons for each rubric item to select which peer's rating/comments are most accurate and helpful for that item.

1 pt - The program successfully opens a frame with the stopwatch stopped.

| Good | Neutral | Bad | Score | Comment |
|---|---|---|---|---|
| ● | ○ | ○ | 1 | (No comment) |
| ● | ○ | ○ | 1 | (No comment) |
| ● | ○ | ○ | 1 | (No comment) |

Figure 3.3 : Screen shot of web page that students use to evaluate peer evaluations. A link to a particular submitted assignment is given, along with a listing of five different peer evaluations for that assignment, for a particular rubric item.



### Peer Evaluation Ranking Results

Below are the CodeSkulptor URLs for the mini-project submission that you evaluated and rankings from other students for your evaluations. Here, you will find several URLs and for each rubric item you will find several evaluations. Each line of the evaluations in a rubric item correspondes to one URL. For example, the first line for each rubric item correspondes to the first URL and the second line correspondes to the second URL.

The numbers in the first three columns are the numbers of Good, Neutral and Bad you received for this evaluation from other students.

http://www.codeskulptor.org/#user22_f9VodoiXtvmbdgt9G.py

http://www.codeskulptor.org/#user21_0AnjBBudSkNehqC8Y.py

http://www.codeskulptor.org/#user22_UoIblGNjgdYQ_0.py

http://www.codeskulptor.org/#user22_31udSLchC31Nz3MNs.py

http://www.codeskulptor.org/#user22_j0X5g7QwWb8azdvUV.py

http://www.codeskulptor.org/#user22_HNqDew7rKtET_6.py

#### Ranking evaluations for individual rubric items

1 pt - The program successfully opens a frame with the stopwatch stopped.

| Good | Neutral | Bad | Score | Comment |
|---|---|---|---|---|
| 2 | 1 | 0 | 1 | (No comment) |
| 2 | 2 | 0 | 1 | (No comment) |
| 0 | 3 | 0 | 1 | (No comment) |

Figure 3.4 : Screen shot of web page that allows students to see how others judged the quality of their peer evaluations.

# Chapter 4

# Experimental Design

Our central goal is to evaluate whether or not such a framework might have some utility in motivating students to perform high-quality peer evaluations. In this section, we describe in detail the study that we designed and executed to this effect.

## 4.1 Study Overview

The study was open to all participants in the Fall 2013 incarnation of the IIPP class. Because we would be asking participants to do a non-trivial amount of work, and (more importantly) because participants would have their work examined by others, participation needed to be voluntary. There was some concern that voluntary participation would skew the results, making it more difficult to detect effects. After all, those MOOC participants who are motivated enough to participate in a study are probably far more likely to already be motivated enough to submit high quality peer evaluations without the extra incentive (possibly) provided by a "grading the graders" regime. However, since the bias was far more likely to result in dampening of the significance of the study results (rather than creating positive results when none should have existed), in the end we considered this a necessary evil that we could live with.

Study participants were divided into tree groups:

1. *Those receiving the full "grading the graders" treatment.* These participants

evaluate other peer evaluations (as described in Section 3.1), and have their own peer evaluations evaluated. Then they are asked to examine the evaluations of the peer evaluations that they have performed (as described in Section 3.2). That is, they receive the full treatment described in the previous section of the paper. We call this group $G_1$.

2. *Those who only have their peer examinations evaluated.* These participants do not actually evaluate any other peer evaluators, but they have their own evaluations examined by members of $G_1$, and they are asked to examine the evaluations of the peer evaluations that they have performed (as described in Section 3.2). We included this group to try to understand whether there is a difference between being asked to evaluate others as is the case in $G_1$ (which necessarily imparts some knowledge of community standards in peer grading to the examiner) and being motivated by knowing that others will be examining one's peer evaluations. We call this group $G_2$.

3. *The control group.* These are people who sign up for the study, but then are not asked to do anything other than participate as usual in the IIPP class. We call this group $G_3$.

By design, we set the ratio of the sizes of three groups to be $G_1 : G_2 : G_3 = 8 : 1 : 1$. The reason for the large size of $G_1$ is that we needed the group to be large enough that they could produce enough evaluations that members of $G_2$ could consume evaluations of their peer grading efforts, without contributing any evaluations of their own. We were concerned about the imbalances effect on statistical power of any analyses that we would need to run, but again, this seemed necessary.

In order to enroll in the study, students were asked to complete a simple web consent form and submit their email address. The consent form described that there were three groups that students could be assigned to (including one where they would be asked to do nothing more than they would normally do in the IIPP class), but not the specific study goals nor what was being measured. Because students could easily communicate on the Coursera forums, there was little point in trying to blind the study so that students would not understand what group they were in. In fact, we decided that attempting to blind the study in this way would be worse than not, since we were concerned that it would encourage students to carefully compare notes on class forums regarding what they were seeing, possibly biasing the results.

To motivate students to sign up for the study—we were especially interested in attracting somewhat less motivated students from the general student population who might not have otherwise signed up—a Nexus tablet was promised to one randomly-selected student from each group.

## 4.2 Timeline and Details

All students who submitted the consent form were randomly assigned to groups. 3,015 students completed a consent form during the enrolling phase. 2,412 students were assigned to $G_1$, 301 students to $G_2$ and 302 students to $G_3$. All students received an email with information describing what they needed to do in the study.

It was expected that all students in the study would participate in the study during for both the Stopwatch and Memory assignments. Here we detail the timeline that was used for both assignments.

**Day 1.** The assignment (Stopwatch/Memory) is posted and a special submission page for the study (separate from the normal submission page) is opened; study participants were asked to submit to that particular page. Students had nine days to submit the assignment.

**Day 9.** The assignment ends. If a study participant did not submit to the special submission page by day 9, they were removed from the study. At this point, students begin peer evaluation. They have one week.

**Day 16.** Peer evaluation ends. At this point, the evaluation phase begins and emails are sent to all $G_1$ students pointing them to the web page where they caan evaluate others' evaluations. Three days were allotted to this task. 201 students outside of the study (in the case of Stopwatch) and 421 students outside of the study (in the case of Memory) also (mistakenly) submitted to the special study submission page. We were happy to have extra evaluations to work with, so these students were treated as $G_1$ students and asked to "grade the graders", but they are not otherwise included in the study results. (Interestingly, 60 of the 201 "mistaken Stopwatch" students and 124 of the 421 "mistaken Memory" students actually completed the "grading the graders" task of Section 3.1).

**Day 19.** "Grading the graders" ends. In the case of Stopwatch, 1,891 out of 2,412 $G_1$ students and 244 out of 301 $G_2$ students (those who had successfully completed study requirements up until that point) receive emails pointing them to a URL where they can see what others thought of their submitted evaluations. In the case of memory, 1,387 $G_1$ students and 192 $G_2$ students receive this email.

# Chapter 5

# Experimental Results

In this section, we describe in detail the results we obtained by analyzing the data that we collected.

## 5.1 Hypotheses Tested

Our goal was to determine whether or not there was some evidence that students enrolled in $G_1$ did a better job grading compared to either $G_2$ or $G_3$, or both. As we will discuss in detail shortly, if we find that $G_1$ generally does a better job, then it might be taken as evidence that the "grading the graders" regime does in fact motivate peer graders.

Thus, our data analysis task comes down to measuring the quality of students' grading efforts. We felt that (short of having a human expert review on the order of 10,000 peer grading submissions that resulted from the study) the two best proxies for measuring quality are (1) whether or not an evaluator gets it right, and typically gives a high score to a good program and typically gives a low score to a bad program, and (2) how much time and effort are taken in writing comments.

However, it turns out that it is not trivial to measure either of those proxies. The problem with looking at score accuracy is that we do not actually know when a submission is in fact a good program, and when it is bad. These are, after all, interactive programs that are very difficult to grade automatically. This is why the

IIPP course relies on peer grading. And without actually reading the comments, it is difficult to measure the effort level.

To address the first difficulty, we manually devised a number of assignment-specific program analyses that were able to automatically and roughly categorize student submission as being good or bad. These methods simulated each submission on a large number of manually written "tests" — defined here as finite sequences of events — and recorded the program's executions on these tests. A large number of manually designed rules were then used to judge the correctness of these executions. The analyses that we ran were certainly not fool proof: as we pointed out earlier, accurate automatic grading for games is extremely difficult. However, we used deep knowledge about the assignments, as well as a large amount of effort, to make our tests and rules as accurate as possible. Moreover, any inaccuracy in the classification will tend to mask the accuracy (or inaccuracy) of a set of program grades in a systematic way (since a mis-categorized program will be graded by members of $G_1$, $G_2$ and $G_3$) and so such inaccuracies are unlikely to introduce bias into our analysis.

To address the second difficulty, we decided to use comment length (in terms of number of words) as a reasonable measure of the quality of a comment. Of course, it is always possible for a grader to write a long but inane and useless comment, but in general, one would expect a longer comment to correlate with more care on the part of the grader.

With this in mind, we developed six different null hypotheses that we would test in an attempt to differentiate the quality of any two groups of graders $A$ and $B$. These null hypothesis are:

**Hypothesis One:** $H_0^1 =$ "The mean score for group $A$ is no greater than the mean score for group $B$ on good programs"

If this hypothesis is refuted, then it means that group $A$ is doing a better job than group $B$ in recognizing good programs, which would be a strong indicator that group $A$ does a better job grading good programs.

**Hypothesis Two:** $H_0^2$ = "The mean score for group $A$ is no less than the mean score for group $B$ on bad programs"

If this is refuted, it means that group $A$ does a better job than $B$ recognizing bad programs, which is again indicative that group $A$ is doing a better job.

**Hypothesis Three:** $H_0^3$ = "The median comment length for group $A$ is no greater than the median comment length for group $B$ on good programs"

If this is refuted, it means that group $A$ writes longer comments on high-quality submissions than group $B$. We use median rather than mean since the comment length appears to have a heavy-tailed distribution, making the mean quite unstable. Refuting this would be particularly interesting, because one might expect that it is very easy for a mediocre grader to simply give full credit to a high-quality submission. A careful grader would look at the code, even if the program works well, and offer comments on the style and substance of the implementation.

**Hypothesis Four:** $H_0^4$ = "The median comment length for group $A$ is no greater than the median comment length for group $B$ on bad programs"

Comments on bad programs are the most important feedback that a struggling student will receive, and so this is also an important hypothesis.

**Hypothesis Five:** $H_0^5$ = "The fraction of people doing a 'bad job' in group $A$ is no less than the fraction doing a 'bad job' in $B$"

We define someone who has done a "bad job" to be a grader that either (a) gets the grade wrong, and gives a perfect score to a program that our code analysis engine thinks is flawed (or gives a non-perfect score to a program that our engine can find no fault with), or (b) writes no comment across all rubric items. If we are able to refute this hypothesis, it means that someone from group $A$ is less likely to do a bad job than someone from group $B$.

**Hypothesis Six:** $H_0^6 =$ “The fraction of people doing a ‘really bad job’ in group $A$ is no less than the fraction doing a ‘really bad job’ in group $B$”

We define someone who has done a “really bad job” similarly to the way we define someone who has done a “bad job,” but we replace the *or* with an *and*.

For each of these hypotheses, we perform six different statistical tests. For Stopwatch, we compare (1) $G_1$ vs. $G_2$, (2) $G_1$ vs. $G_3$, and (3) $G_2$ vs. $G_3$. We also make the same comparisons for Memory. The reason that we performed these particular tests is that we were looking for evidence that the “grading the graders” regime has a positive effect on peer grading, and so it makes sense to compare those who have undertaken the full “grading the graders” program (those in $G_1$) versus the other two groups. We are also interested in comparing $G_2$ and $G_3$ because we would like to see if there is any difference between the partial “grading the graders” program (those in $G_2$ only had their peer evaluations evaluated; they did not actually evaluate others' peer evaluations) and the control group.

## 5.2 Statistical Significance

Checking whether these various null hypotheses are refuted obviously requires some sort of statistical test of significance to obtain a $p$-value for each of the hypotheses. At first glance, the textbook test for this sort of experimental setup would be a paired

$t$-test [23], since we have two groups, $A$ and $B$, and in each case we are checking for a difference in the mean or the median of some statistic, computed across the groups. Further, a "paired" version of the test seems appropriate, since members of both groups are typically paired, grading the same program.

Unfortunately, a $t$-test, paired or otherwise, seems inappropriate when examined in detail. In fact, any sort of textbook test for significance, parametric or not, is likely going to be invalid in our experimental framework. The problem is that when we test a particular hypothesis (say, $H_0^1$) we have multiple sources of correlation across the scores that are being added. Not only are the graders grading the same programs (a source of correlation that is in fact handled by a paired test) but *more than one grader from each group* may be grading the same program. For example, when looking at a specific program, it can be the case that five graders from group $A$ and three graders from group $B$ graded the same program, leading to a very unique covariance structure. Another source of correlation among the observed scores is that *each grader will grade multiple programs*, so that the scores may be correlated because they came from the same grader.

As a result, we had two obvious options. We could resort to something like a $t$-test, being cognizant of its limitations, or else we could utilize a simulation-based solution that naturally takes into account such issues, such as the bootstrap [24]. In the end, we chose the latter option.

Briefly, the idea behind the bootstrap is to use a resampling-based algorithm to simulate a very large number of data sets from the collected data set. The null hypothesis is checked on each, and the fraction of the time that the null hypothesis holds is the $p$-value.

To apply the bootstrap in our own setting, we generate a simulated data set as follows. Given a set of $n$ programs called $P$ for which one or more peer graders participating in the study actually graded, let $cnt(A, p)$ denote the number of graders from group $A$ who graded program $p \in P$, and let $scores(A, p)$ denote the set of peer grading efforts created by those graders from group $A$. Then to bootstrap resample our data set, we first resample $n$ programs from $P$ by sampling $n$ times from $P$ with replacement. Call these sampled programs $p_1$, $p_2$, ..., $p_n$. For each $p_i$, we then create a new set of grading efforts for group $A$ by resampling $cnt(A, p_i)$ grades from $scores(A, p_i)$. We then create a new set of grading efforts for group $B$ by resampling $cnt(B, p_i)$ grades from $scores(B, p_i)$. By unioning all of the grading efforts across all $p_1$, ..., $p_n$, we create a new, simulated version of our data set. This simulated version respects the correlations induced by having multiple graders grade the same program (because there will be multiple grades of the same program in the simulated data set), and it also respects correlations induced by having the same grader grade multiple programs (since this will also happen in the simulated data set).

## 5.3 Results

We ran the resulting bootstrap tests across all of the hypotheses defined above. All results are given above in Table 5.1. For each group and for each hypothesis, we give the $p$-value with which we reject the relevant null hypothesis, according to the bootstrap test. In general, a $p$-value of less than or equal to 0.05 (or possibly 0.1) is considered to be statistically significant. We **bold** all $p$-values that are significant at $\leq 0.05$. Just as important, we give the mean or median value of the relevant statistic for each group that is being tested. For example, for $H_0^1$ = "The mean score for group $A$ is no greater than the mean score for group $B$ on good programs," we give the mean program score for good programs for group $A$ and group $B$. These values

**Stopwatch** **Memory**

$H_0^1$ = "The mean score for group $A$ is no greater than the mean score for group $B$ on good programs"

| Group $A$ | Group $B$ | Group $A$ Mean | Group $B$ Mean | $p$-value | Group $A$ Mean | Group $B$ Mean | $p$-value |
|---|---|---|---|---|---|---|---|
| $G_1$ | $G_2$ | 12.91 | 12.91 | 0.5889 | 10.88 | 10.89 | 0.7781 |
| $G_1$ | $G_3$ | 12.91 | 12.88 | **0.0255** | 10.88 | 10.87 | 0.3501 |
| $G_2$ | $G_3$ | 12.91 | 12.88 | **0.0477** | 10.89 | 10.87 | 0.1987 |

$H_0^2$ = "The mean score for group $A$ is no less than the mean score for group $B$ on bad programs"

| Group $A$ | Group $B$ | Group $A$ Mean | Group $B$ Mean | $p$-value | Group $A$ Mean | Group $B$ Mean | $p$-value |
|---|---|---|---|---|---|---|---|
| $G_1$ | $G_2$ | 12.02 | 12.09 | 0.1095 | 10.29 | 10.36 | 0.1911 |
| $G_1$ | $G_3$ | 12.02 | 12.04 | 0.3543 | 10.29 | 10.39 | 0.0783 |
| $G_2$ | $G_3$ | 12.09 | 12.04 | 0.7527 | 10.36 | 10.39 | 0.3214 |

$H_0^3$ = "The median comment length for group $A$ is no greater than the median comment length for group $B$ on good programs"

| Group $A$ | Group $B$ | Group $A$ Median | Group $B$ Median | $p$-value | Group $A$ Median | Group $B$ Median | $p$-value |
|---|---|---|---|---|---|---|---|
| $G_1$ | $G_2$ | 11 | 10 | 0.6603 | 12 | 10 | **0.0060** |
| $G_1$ | $G_3$ | 11 | 11 | 0.7888 | 12 | 10 | **0.0036** |
| $G_2$ | $G_3$ | 10 | 11 | 0.8341 | 10 | 10 | 0.9356 |

$H_0^4$ = "The median comment length for group $A$ is no greater than the median comment length for group $B$ on bad programs"

| Group $A$ | Group $B$ | Group $A$ Median | Group $B$ Median | $p$-value | Group $A$ Median | Group $B$ Median | $p$-value |
|---|---|---|---|---|---|---|---|
| $G_1$ | $G_2$ | 83 | 78 | 0.3300 | 69 | 51.5 | 0.1014 |
| $G_1$ | $G_3$ | 83 | 74 | 0.2166 | 69 | 51 | 0.1348 |
| $G_2$ | $G_3$ | 78 | 74 | 0.4077 | 51.5 | 51 | 0.4700 |

$H_0^5$ = "The fraction of people doing a 'bad job' in group $A$ is no less than the fraction doing a 'bad job' in group $B$"

| Group $A$ | Group $B$ | Group $A$ Mean | Group $B$ Mean | $p$-value | Group $A$ Mean | Group $B$ Mean | $p$-value |
|---|---|---|---|---|---|---|---|
| $G_1$ | $G_2$ | 0.3820 | 0.4007 | 0.0721 | 0.3477 | 0.4021 | **0.0004** |
| $G_1$ | $G_3$ | 0.3820 | 0.4091 | **0.0191** | 0.3477 | 0.4157 | **0.0001** |
| $G_2$ | $G_3$ | 0.4007 | 0.4091 | 0.3152 | 0.4021 | 0.4157 | 0.2504 |

$H_0^6$ = "The fraction of people doing a 'really bad job' in group $A$ is no less than the fraction doing a 'really bad job' in group $B$"

| Group $A$ | Group $B$ | Group $A$ Mean | Group $B$ Mean | $p$-value | Group $A$ Mean | Group $B$ Mean | $p$-value |
|---|---|---|---|---|---|---|---|
| $G_1$ | $G_2$ | 0.04522 | 0.05051 | 0.1894 | 0.03404 | 0.04442 | 0.0691 |
| $G_1$ | $G_3$ | 0.04522 | 0.06133 | **0.0057** | 0.03404 | 0.04648 | **0.0400** |
| $G_2$ | $G_3$ | 0.05051 | 0.06133 | 0.0792 | 0.04442 | 0.04648 | 0.3896 |

Table 5.1 : Summary of study results, comparing $G_1$ (students who both "graded the graders" and viewed the peer evaluations of their own grading efforts), $G_2$ (students who only viewed the peer evaluations of their own grading efforts), and $G_3$ (students who neither "graded the graders" nor viewed the peer evaluations of their own grading efforts). Statistically significant findings are shown in **bold**.

are there to let the reader judge whether any differences are of practical significance.

In the remainder of the section, we highlight and explain a few of the results. In the next full section of the paper, we discuss the conclusions that we might draw from them.

**Many of the Findings Are Statistically Significant.** If, for a moment, we restrict ourselves to comparisons of $G_1$ vs. $G_2$ and $G_1$ vs. $G_3$, for Memory (which is the second assignment; hence, any gains from undertaking the "grading the graders" regime on Stopwatch would have had a chance to manifest themselves), 12 different null hypotheses were checked. Of those 12, 5 were rejected with a $p$-value $\leq 0.05$, and the other 2 at a $p$-value $\leq 0.1$. While there is certainly something of a multiple-hypothesis testing problem here given that 12 tests were run [25], the fact so many result in rejection of the null hypothesis seems strongly indicative of a positive effect of the full "grading the graders" regime compared to students in $G_2$ and the control group $G_3$.

**Might We Have Hurt Statistical Power By Partitioning $\bar{G_1}$ into $G_2$ and $G_3$?** Often, if the effect one is looking for is stronger in one segment of the population, it makes sense to stratify into sub-populations and run multiple tests, but since it results in multiple tests that each have a smaller number of samples, power is reduced.

To investigate this a bit, we re-ran each of the hypothesis tests, this time comparing $G_1$ vs. $\bar{G_1}$. The results are summarized in Table 5.2. Again restricting ourselves to Memory, 3 of 6 null hypotheses are rejected with a $p$-value $\leq 0.02$, and another 2 of 6 with a $p$-value $\leq 0.056$.

**Many of the Findings Are of Practical Significance.** It is often easy to conflate statistical significance with practical significance. When one runs a large-scale study

**Stopwatch** **Memory**

$H_0^1 =$ "The mean score for group $A$ is no greater than the mean score for group $B$ on good programs"

| Group $A$ | Group $B$ | Group $A$ Mean | Group $B$ Mean | $p$-value | Group $A$ Mean | Group $B$ Mean | $p$-value |
|---|---|---|---|---|---|---|---|
| $G_1$ | $G_2$ and $G_3$ | 12.912094 | 12.898951 | 0.0726 | 10.875696 | 10.878788 | 0.5878 |

$H_0^2 =$ "The mean score for group $A$ is no less than the mean score for group $B$ on bad programs"

| Group $A$ | Group $B$ | Group $A$ Mean | Group $B$ Mean | $p$-value | Group $A$ Mean | Group $B$ Mean | $p$-value |
|---|---|---|---|---|---|---|---|
| $G_1$ | $G_2$ and $G_3$ | 12.021513 | 12.069231 | 0.1161 | 10.286411 | 10.373171 | **0.0416** |

$H_0^3 =$ "The mean comment length for group $A$ is no greater than the mean comment length for group $B$ on good programs"

| Group $A$ | Group $B$ | Group $A$ Mean | Group $B$ Mean | $p$-value | Group $A$ Mean | Group $B$ Mean | $p$-value |
|---|---|---|---|---|---|---|---|
| $G_1$ | $G_2$ and $G_3$ | 79.054181 | 70.707896 | **0.0037** | 77.923933 | 62.733825 | **0** |

$H_0^4 =$ "The mean comment length for group $A$ is no greater than the mean comment length for group $B$ on bad programs"

| Group $A$ | Group $B$ | Group $A$ Mean | Group $B$ Mean | $p$-value | Group $A$ Mean | Group $B$ Mean | $p$-value |
|---|---|---|---|---|---|---|---|
| $G_1$ | $G_2$ and $G_3$ | 177.298961 | 166.338462 | **0.0421** | 161.124314 | 144.914634 | **0.0435** |

$H_0^5 =$ "The fraction of people doing a 'bad job' in group $A$ is no less than the fraction doing a 'bad job' in group $B$"

| Group $A$ | Group $B$ | Group $A$ Mean | Group $B$ Mean | $p$-value | Group $A$ Mean | Group $B$ Mean | $p$-value |
|---|---|---|---|---|---|---|---|
| $G_1$ | $G_2$ and $G_3$ | 0.382037 | 0.40477 | **0.0011** | 0.347659 | 0.408338 | **0** |

$H_0^6 =$ "The fraction of people doing a 'really bad job' in group $A$ is no less than the fraction doing a 'really bad job' in group $B$"

| Group $A$ | Group $B$ | Group $A$ Mean | Group $B$ Mean | $p$-value | Group $A$ Mean | Group $B$ Mean | $p$-value |
|---|---|---|---|---|---|---|---|
| $G_1$ | $G_2$ and $G_3$ | 0.045217 | 0.05577 | **0.0025** | 0.034042 | 0.045371 | **0.0043** |

Table 5.2 : Summary of study results, comparing $G_1$ (students who both "graded the graders" and viewed the peer evaluations of their own grading efforts) versus those who did not "grade the graders". Statistically significant findings are shown in **bold**.

involving thousands of participants, there are often statistically significant results that are of no practical significance. Is that the case here? Many of the results in Table 5.1 and Table 5.2 appear to be of practical significance as well. For one example, consider the Memory assignment results in Table 5.2. We found that the percentage of graders who did a "really bad job" decreased from 4.5% for those who do not receive the full "grading the graders" treatment down to 3.4% for those who did. The percentage of those who do a "bad job" decreased from 41% to 35% under the regime. These reductions are perhaps more significant when one considers that the students who voluntarily signed up for the study are likely to be far more motivated already than those who did not.*

*Along those lines, we did a bit of data analysis on the students who did not agree to participate in the study but managed to accidentally submit their grading efforts to the study. We found, for example, that the median comment length for everyone who signed up for the study was 47, while

$H_0$ = "The median comment length for group $A$ is no greater than the median comment length for group $B$ on all programs"

| | | **Before Study** | | | **After Study** | | |
|---|---|---|---|---|---|---|---|
| Group $A$ | Group $B$ | Group $A$ Median | Group $B$ Median | $p$-value | Group $A$ Median | Group $B$ Median | $p$-value |
| $G_1$ | $G_2$ | 46 | 56.5 | 0.9472 | 21 | 16 | **0.0437** |
| $G_1$ | $G_3$ | 46 | 47.5 | 0.5650 | 21 | 16 | **0.0184** |
| $G_2$ | $G_3$ | 56.5 | 47.5 | 0.0991 | 16 | 16 | 0.5595 |
| $G_1$ | $G_2$ and $G_3$ | 46 | 52 | 0.8950 | 21 | 16 | **0.0072** |

Table 5.3 : Analyzing comment lengths in the assignment before the study began (at left) and for the three assignments after the study ended. Findings significant at the 0.05 level are shown in **bold**.

**The Effects Were Lasting.** Students completed three more assignments after the study ended. To see whether there were any lasting effects from the "grading the graders" regime, we analyzed the median comment lengths of student grading efforts on those assignments (as a sanity check, we also analyzed the comment lengths on the assignment immediately before Stopwatch). Table 5.3 summarizes the results. We find that there is actually a significant, persistent effect of participating in the "grading the graders" regime.

for the non-study group the median length was 17. Thus, there is a lot more room to improve the grading efforts of non-study participants.

# Chapter 6

# Related Work

## 6.1 Peer Evaluation

Peer evaluation has a long history in education and people have been studying its effect on learning process and its reliability and accuracy. Somervell [26] found out that students participating in peer assessments need to make judgements about the work or the performance of other students, which as a result makes peer assessment a part of learning process. Keaten et al.[27] also had the similar conclusion. They pointed out that peer assessment can "foster high levels of responsibility among students", which leads to the necessity for students to be "fair and accurate with the judgements they make regarding their peers". Sluijsmans et al. [28] did a systematic literature review on self-, peer-, and co-assessment in higher educational settings. In particular, they pointed out that peer assessment could be beneficial as a part of the learning process since students could have better engagement in learning and in the assessment process. They also found out that the results from peer assessment tend to be fiare and accurate. In the end, they pointed out some disadvantages of peer assessment and suggessted those weaknesses could be solved by using combinations of self- and co-assessment. Falchikov in 1995 [29] reviewed some studies of peer assessment in higher educational settings and discovered that most of those studies focused on assessment of a product or of the skill performance. Falchikov also found out that peer assessment was quite useful, reliable and beneficial to students. Finally, they conducted a peer assessment study in which they emphasized on peer feedback and their results indicated a close correlation between students' grades and teachers' grades and that feedback was perceived to be useful.

Several controlled studies have shown that peer evaluation can be a good proxy for teacher feedback. In 1998, Topping [16] proposed a peer assessment scheme in the setting of higher education. In their study, peer assessments with grades appear to have positive effects on student achievement and attitudes and those effects are sometimes even better than instructor assessments. However, other types of peer assessments on activities such as presentation and group work tends to be limited. Later, Falchikov and Goldfinch [20] compared peer grading with instructor grading in a higher education setting. They discovered that when global judgment were well understood and defined, peer assessments tend to resemble closely to the instructor assessments. In addition, they also found out that peer assessments on academic products and processes instead of professional activities are closer to the instructor assessments. After all, peer assessments appear to be more valid in a well-design study than those in a poor experimental setting. Peer evaluation and feedback is also showed to be effective in lessening the grading burden on instructors according to [19]. More importantly, it increases interactions between students and thus instructors share the teaching responsibility with students through discussion and peer feedback. The results from their study show that students tend to expect feedback to be timely and high-quality. Again, they emphasize the importance of design and logistics of the peer evaluation process, because one of the weaknesses appear to be associated with the design and logistics.

Peer evaluation could be really beneficial if it is well-designed and conducted properly. Sluijsmans et al. [30] discussed creating a learning environment by using self assessment, peer assessment and the combination of both. They analyzed many studies and pointed out that self-assessment, peer-assessment and co-assessment can be effective and are often used in combination. Gueldenzoph and May [31] also pointed out that instructors need to follow a set of rules in order to have effective peer assessments in a collaborative environment and those rules include building a collaborative foundation, well-designed criteria, ensuring participation, formative feedback implementation during collaborative experience, summative feedback at the end and assessing the evaluation process.

Despite all the benefits of peer evaluation mentioned above, Liu and Carless [32] focused on several reasons resistances to peer assessment including reliability, perceived expertise, power relations adn time constraint. Finally, they proposed three possible ways to solve the issues: "peer feedback integraded with peer assessment", "strategies for engating students with criteria" and "cultivating a course climate for peer feedback."

Using computer to conduct peer assessment can be traced back to 1993 when Rushton et al. [33] developed a peer assessment system. They had 32 computer science students writting an essay and did peer assessment using computers. Finally, they discovered that the grades given by students were similar to the grades given by tutors. However, in the context of MOOC, Peer evaluation offers some unique challenges. The first is the lack of a gating function. In a traditional classroom one can be sure that all students are on roughly the same level. Outliers are possible, but as most students will have taken the same preparatory classes and/or been admitted to the same university. In MOOCs, one can assume no core competence on the part of the evaluators. A second challenge in a massive, online environment is the lack of centralized oversight. There is much mention in the online education research community of the necessity of “teaching presence”: the need for participants to feel as if an instructor is present and monitoring the proceedings. In the study done by Rechardson and Swan [34], they discovered that with high social presence, the scores and satisfaction with the instructor of students tend to be higher in a online education environment. Swan and Shih [35] did another set of quantitative and qualitative studies on the effect of social presence. Their results show that there is significant correlation between perceived social presence and student satisfaction with online course. In typical work on peer evaluation, a teacher is normally present to settle disputes and monitor the fairness of the proceedings. The incentive system that we investigate can be seen as a decentralized, bottom-up simulation of teacher presence.

## 6.2 Behavior Study

Student behavior in an online education setting has been studied before. Rovai [36] studied the correlation between sense of community and perceived cognitive learning. Over 300 students enrolled in 26 graduate education and leadership courses taught in a online education system. The author found out that online graduate students can feel connected to the online classroom community and those with strong sense of community tend to posses greater perceived levels of cognitive learning. Interestingly, they discovered that ethnicity and course content do not seem to affect sense of community, but female students tend to have greater sense of community. Davies and Graff [37] did a similar study on the correlation between students’ performance and the level of interaction between other students. Their study examined the online interaction frequency of over 100 undergraduate students for a duration of 12 months. They discovered that greater online interaction did not lead to significantly higher performance but students who failed the course tend to interact less.

As MOOCs become popular, user performance and behavior patterns have been studied more thoroughly. Breslow et al. [2] did a series of studies with the data gathered from Edx 6.002x. First, they study the resource usage and found out that discussion forum was the most frequently used resources and homework problems and lecture videos consumed the most amount of time. Moreover, their data showed that only 3% of all students participated in the discussion forum. However, over 50% certificate earners were active in the forum. Then they turned to demographics and showed that the participants were diverse even though a large amount students spoke English. A survey from their study showed that over half the survey respondents reported that the primary reason for taking the MOOC was for knowledge and skills they would gain. If achievement is defined as total points in the course, they found out that there is a strong correlation between the student background and achievement. One of the problems MOOCs have is low completion rate and they observed that less than 5% of the students who signed up for the course actually completed the course. Anderson et al. [6] specifically study the student behavior patterns, engagement styles in MOOCs. They used a large amount of user behavior history and categorized students' engagement styles. Then they found some correlations between different engagement styles and student performance. Kizilcec et al. [38] clustered on engagement patterns and discovered four prototypical categories of engagement consistently across three different MOOCs. Hew and Cheung [39] discovered several challenges in MOOCs and they specifically studied the low student engagement rate problem. The main reasons for dropping out of a MOOC are lack of incentive, failure to understand the material and having no one to help, and having other priorities. They also reported that three main reasons why instructors are willing to teach

MOOCs include being motivated by a sense of intrigue, the desire to gain rewards and a sense of altruism. Finally, they showed that several key challenges of teaching MOOCs are: difficulty in evaluating students' work, heavy demands of time and money, lack of participation in online forums and the absence of student immediate feedback during teaching.

## 6.3 Incentives and Motivation Study

Some online knowledge-sharing forums such as StackOverflow, Y! Answers have been using some form of incentives to motivate people to contribute. Especially, recognizing people's great contribution by giving them *badges* has been proved to be an effective way to motivate people to contribute. The effect of incentives in an online community has been studied in [40]. Easley and Ghosh used an economic approach to the question of how badges can be most effectively used for incentivizing participation and effort. Specifically, Anderson et al. [5, 6] implemented a badge system in their MOOC. The system is mainly used in discussion forum for incentivizing forum participation. Multiple badge types and levels are implemented including bronze, silver, gold and diamond which are associated with increasing milestones. As the badge level increases, the difficulty of getting that badge also increases. They did a controlled experiment and the results showed that making badges more salient increases the level of forum engagement.

A recent paper by Kulkarni et al. [7] studies ways to improve grading accuracy in MOOCs. Specifically, The authors consider data on self and peer assessment from two iterations of a MOOC. To measure the accuracy of peer assessment, they computed the percentage differences between peer and staff grades. They showed that students tend to give higher grades and they get better at grading over time.

The results also showed that the correlation among staff grades is much higher than agreement among peer graders and that aggregating peer grades leads to a large increase in agreement with staff grades. Their first solution to improving grading accuracy is giving feedback to students about the bias in their peer grading including “too high”, “too low” or “just right” based on how well their grades agree with the staff grades. They conducted a controlled experiment and the results indicate an increase in agreement between peer grades and staff grades. Another solution is using more precise rubric items. They first use low grader agreement to find rubric items that might benefit from revisions, then they review and revise those rubric items based from the feedback from the forum. They make the rubric items more readable, revise all rubrics to use parallel sentence structure and improve word choice. Then they use the revised rubric in the second iteration of the class and showed that there is an increase between peer and staff grades agreement.

# Chapter 7

# Conclusion

It is easy to do a poor job peer grading. Intuitively, if a grader spends little time grading, then the grader will not find any problems. This leads to artificially inflated numerical grades and little feedback. As intimated in the introduction of the paper, we found that it was much more likely that a grader would give a perfect score to an imperfect program than the other way around, which supports this intuition. The students who did not do well on an assignment and most need quality feedback are the ones who suffer the most from poor grading.

Our study results corroborate the expectation that peer graders have no problem assigning high numerical scores to good assignments. There was little evidence of a difference among study groups at assigning numerical grades to good IIPP programs. While there were statistically significant results showing that both groups $G_1$ and $G_2$ did a better job at assigning a grade to good Stopwatch programs, the actual differences in scores among all the groups for both Stopwatch and Memory were in the hundredths of points. Further note that the Stopwatch program was actually simpler to grade, as Memory requires a more complex set of actions to be performed by the grader to verify correctness. More than anything, this indicates that all groups graded good programs well.

Ultimately, this suggests that with no intervention, it should be expected that peer grading will bias towards higher scores. However, while little effort is required to give high scores, effort is required provide feedback. Here we found the “grading the graders" regime to be useful for motivating the graders, even on good programs. We found that group $G_1$ did provide longer comments than both $G_2$ and $G_3$ on such programs.

In fact, the study supports the hypothesis that the full regime leads to better peer grading in general. Those in group $G_1$ *did* consistently do a better job grading, according to most of our metrics. Consider the Stopwatch results in Table  5.2, where in 5 of 6 analyses, the null hypothesis was rejected with a $p$-value $\leq 0.056$. The only case where the null hypothesis was *not* rejected was when checking whether those in $G_1$ did no better in scoring good programs. But this is not surprising, as discussed above.

Not only were the results immediately noticeable, they also seemed to be lasting. In Table 5.3 we see that the median comment length for those in $G_1$ stays greater for the final three assignments in the class, compared to the other study participants. Significantly, this was after the end of the study, when students had no reason to expect that the median comment length was going to be monitored.

One might easily believe that peer graders can be motivated to do a better job simply by telling them that their evaluations will themselves be evaluated. However, we found no evidence that this is effective. In particular, those in $G_2$ tended to do no better than those in $G_3$ throughout the study. This strongly suggests that knowing that one is being monitored by one's peers is not a strong enough motivation to do a good job grading others' assignments. We suspect that this extends to other forms of monitoring, such as the use of sentinels to catch bad grading.

Another surprising result is that there were some significant differences between $G_1$ and both of the other two groups not only on the Memory assignment, but also on the Stopwatch assignment (see Table 5.2). This was surprising because Stopwatch grading took place *after* students agreed to participate in the study (and after they had been assigned to groups), but *before students had actually participated in the study in any meaningful way.* At the point that the data were collected, the participants had yet to actually participate in the protocol described in Chapter 3, so they had not yet examined any other students' grading efforts, and yet the differences between groups were still significant.

The most likely explanation is that those students in $G_1$ were aware that they were participating in the full version of the study and hence they were enthusiastic and felt motivated to do well by this simple fact—they somehow felt "special," which turned out to be highly motivational by itself. The comments on the Coursera forums seem to corroborate this explanation. This observation is one source of our belief that high quality grading is likely as much related to good motivation as it is to good information.

Given all of this, we believe that motivation is a key component of high-quality grading, and that actually seeing cases where other students put in the effort to do a good job (or, conversely, seeing how unhelpful it is when other students do not put in such effort) helps provide motivation to student graders. Thus, we would recommend that MOOCS which utilize peer grading should consider using something like the "grading the graders" regime early on in a class, in order to help train and motivate students to do a good job grading subsequent assignments.

# Bibliography

[1] L. Pappano, “The year of the mooc,” *The New York Times*, vol. 2, no. 12, p. 2012, 2012.

[2] L. B. Breslow, D. E. Pritchard, J. DeBoer, G. S. Stump, A. D. Ho, and D. T. Seaton, “Studying learning in the worldwide classroom: Research into edX’s first mooc,” *Research & Practice in Assessment*, vol. 8, pp. 13–25, 2013.

[3] K. Pretz, “Low completion rates for moocs,” *The Institute*, 2014.

[4] R. McEachern, “Why cheat? plagiarism in moocs,” *MOOC News and Reviews*, 2013.

[5] A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec, “Steering User Behavior with Badges,” in *Proceedings of WWW*, pp. 95–106, 2013.

[6] A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec, “Engaging with Massive Online Courses,” in *Proceedings of WWW*, pp. 687–698, 2014.

[7] C. Kulkarni, K. P. Wei, H. Le, D. Chia, K. Papadopoulos, J. Cheng, D. Koller, and S. R. Klemmer, “Peer and self assessment in massive online classes,” *ACM Trans. Comput.-Hum. Interact.*, vol. 20, pp. 33:1–33:31, Dec. 2013.

[8] J. J. Clark, “The correspondence school - its relation to technical education and some of its results,” *Science, New Series*, vol. 24, no. 611, pp. 327–334, 1906.

[9] M. Susan and L. Fernandez, “Before moocs, ”colleges of the air”,” *The Chronicle of Higher Education*, 2013.

[10] D. D. Cox and W. J. Morison, “The university of louisville,” 1999.

[11] J. J. O’Donnell, “Teaching on the infobahn,” *Religious Studies News*, vol. 9.3, no. 4, 1994.

[12] C. Parr, “Mooc creators criticise courses’ lack of creativity,” *Times Higher Education*, 2013.

[13] P. Glader, “Khan academy competitor? mike feerick of alison.com talks about the future of online education,” *Wired Academic*, 2013.

[14] S. Adams, “Is coursera the beginning of the end for traditional higher education?,” *Higher Education*, 2012.

[15] A. DiPardo and S. Freedman, “Peer response groups in the writing classroom: Theoretic foundations and new directions,” *Review of Educational Research*, vol. 58, no. 2, pp. 119–149, 1988.

[16] K. Topping, “Peer assessment between students in colleges and universities,” *Review of Educational Research*, vol. 68, no. 3, pp. 249–276, 1998.

[17] Y. Lai, “Which do students prefer to evaluate their essays: Peers or computer program,” *British Journal of Educational Technology*, vol. 41, no. 3, pp. 432–454, 2009.

[18] J. Dineen, H. Clark, and T. Risley, “Peer tutoring among elementary students: Educational benefits to the tutor,” *Journal of Applied Behavior Analysis*, vol. 10, no. 2, p. 231, 1977.

[19] P. Ertmer, J. Richardson, B. Belland, D. Camin, P. Connolly, G. Coulthard, K. Lei, and C. Mong, "Using peer feedback to enhance the quality of student online postings: An exploratory study," *Journal of Computer-Mediated Communication*, vol. 12, no. 2, pp. 412–433, 2007.

[20] N. Falchikov and J. Goldfinch, "Student peer assessment in higher education: A meta-analysis comparing peer and teacher marks," *Review of educational research*, vol. 70, no. 3, pp. 287–322, 2000.

[21] D. Casey, E. Burke, C. Houghton, L. Mee, R. Smith, D. Van Der Putten, H. Bradley, and M. Folan, "Use of peer assessment as a student engagement strategy in nurse education," *Nursing & Health Sciences*, vol. 13, no. 4, pp. 514–520, 2011.

[22] T. Tang, S. Rixner, and J. Warren, "An environment for learning interactive programming," in *Proceedings of the 45th ACM technical symposium on Computer science education*, pp. 671–676, ACM, 2014.

[23] G. E. Box, W. G. Hunter, J. S. Hunter, *et al.*, *Statistics for experimenters.* John Wiley and sons New York, 1978.

[24] B. Efron and B. Efron, *The jackknife, the bootstrap and other resampling plans*, vol. 38. SIAM, 1982.

[25] R. G. Miller, *Simultaneous statistical inference*, vol. 196. Springer, 1966.

[26] H. Somervell, "Issues in assessment, enterprise and higher education: the case for self-peer and collaborative assessment," *Assessment and Evaluation in Higher Education*, vol. 18, no. 3, pp. 221–233, 1993.

[27] J. A. Keaten and M. E. Richardson, "A field investigation of peer assessment as part of the student group grading process," 1993.

[28] F. Dochy, M. Segers, and D. Sluijsmans, "The use of self-, peer and co-assessment in higher education: A review," *Studies in Higher Education*, vol. 24, no. 3, pp. 331–350, 1999.

[29] N. Falchikov, "Peer feedback marking: Developing peer assessment," *Innovations in Education and Training International*, vol. 32, no. 2, pp. 175–187, 1995.

[30] D. Sluijsmans, F. Dochy, and G. Moerkerke, "Creating a learning environment by using self-, peer- and co-assessment," *Learning Environments Research*, vol. 1, no. 3, pp. 293–319, 1998.

[31] L. E. Gueldenzoph and G. L. May, "Collaborative peer evaluation: Best practices for group member assessments," *Business Communication Quarterly*, vol. 65, no. 1, pp. 9–20, 2002.

[32] N.-F. Liu and D. Carless, "Peer feedback: the learning element of peer assessment," *Teaching in Higher Education*, vol. 11, no. 3, pp. 279–290, 2006.

[33] C. Rushton, P. Ramsey, and R. Rada, "Peer assessment in a collaborative hypermedia environment: a case-study," 1993.

[34] J. Richardson and K. Swan, *Examing social presence in online courses in relation to students' perceived learning and satisfaction.* 2003.

[35] K. Swan and L. Shih, "On the nature and development of social presence in online course discussions," *Journal of Asynchronous Learning Networks*, vol. 9, no. 3, pp. 115–136, 2005.

[36] A. P. Rovai, “Sense of community, perceived cognitive learning, and persistence in asynchronous learning networks,” *The Internet and Higher Education*, vol. 5, no. 4, pp. 319 – 332, 2002.

[37] J. Davies and M. Graff, “Performance in e-learning: online participation and student grades,” *British Journal of Educational Technology*, vol. 36, pp. 657–663, July 2005.

[38] R. F. Kizilcec, C. Piech, and E. Schneider, “Deconstructing Disengagement: Analyzing Learner Subpopulations in Massive Open Online Courses,” in *Proceedings of the International Conference on Learning Analytics and Knowledge*, LAK ’13, pp. 170–179, ACM, 2013.

[39] K. F. Hew and W. S. Cheung, “Students and instructors use of massive open online courses (moocs): Motivations and challenges,” *Educational Research Review*, vol. 12, no. 0, pp. 45 – 58, 2014.

[40] D. Easley and A. Ghosh, “Incentives, Gamification, and Game Theory: An Economic Approach to Badge Design,” in *Proceedings of the Fourteenth ACM Conference on Electronic Commerce*, EC ’13, (New York, NY, USA), pp. 359–376, ACM, 2013.