Lightweight Preliminary Peer Review: Does In-Class Peer Review Make Sense?

Tamara Denning, Michael Kelly, David Lindquist, Roshni Malani,
William G. Griswold, Beth Simon
Computer Science and Engineering
University of California, San Diego
La Jolla, CA, USA 92093-0404
{tdenning, m1kelly, dlindqui, rmalani, wgg, bsimon}@ucsd.edu

ABSTRACT

Peer review is widely recognized for advancing student learning, in particular for developing reflective processes like critical thinking. The classroom is ripe for peer review because the subject matter is fresh and in-depth interactivity is possible. Yet the limited time available in class conflicts with peer review's deliberative nature. We hypothesize that peer review – at least the initial stages of it – can be supported in the classroom with tools for facilitating the rapid identification of interesting issues for discussion. The potential benefits of such a tool include: furthering the student-focus of in-class active learning activities, further implanting critical analysis skills through frequent in-class use, supporting immediate feedback, and enabling comparison of student and instructor-modeled critical analysis.

This paper explores tool support for in-class lightweight preliminary peer-review (LPPR): peer review that is instigated in the classroom, but does not necessarily end there. We proposed that students classify peer solutions in 4 dimensions: correctness, comprehension (e.g., "do I understand this solution"), worthiness for discussion, and similarity to the evaluator's own solution. We designed an LPPR extension to Ubiquitous Presenter, and then conducted an exploratory study in a mock classroom setting. We found that LPPR can quickly identify a subset of student solutions that warrant immediate discussion, and that modest amounts of reflection arise from the LPPR process.

Categories and Subject Descriptors

K.3.1 [**Computers and Education**]: Computer Uses in Education

General Terms: Human Factors

Keywords

Educational technology, peer review, active learning

1. INTRODUCTION

Peer review, or peer evaluation, is widely known to benefit students in many subject areas [8] and also, in Computer Science, to model real-world industry practices [7]. A key aspect of peer review is engaging students in critical thinking and reflection. Some have used journals to engender reflection [5]. Others have augmented this with web-based techniques to ease management [3]. After solving a problem, students are shown sample code solutions and asked to rate them on a number of predefined aspects ("assess choice of instance variables", "assess presentation of routine headers"), enabling quick perusal of the reviews. Managing the process and getting timely feedback to students is challenging, and many have developed technological solutions to manage this, including Peer Grader [4] and RRAS [9].

In this paper we propose a process and supporting tool for enabling lightweight in-class peer review, perhaps to initiate more complete review outside class. Traditional after-class homework-based peer review provides time for deliberative review, but it also temporally separates the initial problem-solving process, the critique of others' work, and feedback from the instructor. This may compromise students' accurate recollection of, and hence reflection on, their own problem-solving processes in light of the work of others. Bringing a form of peer review into the classroom provides a prime opportunity for addressing these problems as well as raising the level of inquiry in the classroom: we can work with students' immediate recollection of their problem-solving processes, and we can immediately display the results back to both students and instructors to enable class discussion and instructor modeling of critical analysis.

MessageGrid [6] seeks to support peer review in class (and out). Students can participate in clicker-style polls and can also submit more complex typed problem solutions. Student results can be public or private and available either just to the instructor or the whole class. However, formalized peer review is enabled through a comment section, whose results are not optimized for identifying issues for the class as a whole or guiding discussion in the limited time available during class.

We posit the need for a lightweight preliminary peer review (LPPR) model to support peer review and engender reflective practices in the classroom. We argue that a system supporting LPPR must enable these key components:

1. Quick sharing of in-class student generated artifacts;
2. Lightweight selection-based review criteria that can be easily combined for display during class discussion;
3. Prompt self reflection;
4. Use of peer reviews for guiding class discussion; and
5. Instructor overview of the class response.

A system with these components will allow students to engage in active learning, review the work of others in the context of their

SIGCSE'07, March 7–10, 2007, Covington, Kentucky, USA.

own problem-solving process, and direct the instructor in discussion of key issues of confusion or interest. Instructors will be able to get a grasp of overall class performance and model comparative and critical analysis of key student-identified work.

We explore the design and use space of an LPPR system through an extension to the Ubiquitous Presenter system (Sections 2 and 3). Through the use of a mock classroom (Sections 4 through 6), we explore: simple, selection-based review criteria (to keep review lightweight) and their meaningfulness to students and instructors in engendering student reflection and driving classroom activity.

2. SYSTEM DESIGN

2.1 Ubiquitous Presenter

Ubiquitous Presenter (UP) is a web-based extension of University of Washington's Classroom Presenter (UWCP) [1]. UWCP uses Tablet PCs to allow instructors to create ink annotations on top of pre-prepared lecture slides. These slides and annotations are then broadcasted via a multicast network to the students' machines. UWCP also enables the students to create and transfer ink-based problem solutions to the instructor during class, thereby enabling active learning [2].

UP takes UWCP's functionality onto the web; the instructor's slides and ink strokes are archived online and available to the students for incremental, asynchronous viewing. The student may use any browser-enabled device to watch the lecture in 'live' mode, which synchronizes his view with the instructor's slide changes and ink marks. Students may use any browser-enabled device to send solutions to the instructor; the current modalities available are text, ink (via a Java applet), and multiple choice polling, with the inclusion of mobile phone SMS scheduled to take place in the near future. Students can, via the web, review their own and others' submitted work anonymously both in and after class.

2.2 Student Rating of Exercises

After making a submission, the student is taken to a peer-review screen where she can review a queue of other student solutions (with a maximum length specified by the instructor) one at a time (See 1). The screen consists of: an image of the solution to be evaluated, four pairs of non-defaulted radio buttons, a field for comments, and a submit button. Each pair of buttons addresses a category of evaluation for the solution: Correct/Incorrect, Understand/Confuses Me, Discuss/Don't Discuss, and Compared to my solution, this solution is Similar/Different. If nothing is entered, the submit button reads 'Skip Submission'; once something is entered, the button's text changes to 'Submit Rating.'

Students may view the resulting ratings and comments from the UP web view (See Figure 2). When viewing a submission, a list of visual indicators for the Correct, Understood, and Discuss categories are displayed on the right side of the page. A fully filled-in green circle indicates greater than 75% positive votes for that category; a half filled-in green circle indicates between 75%-50% positive votes; a half filled-in red circle indicates 50%-25% positive votes; a fully filled-in red circle indicates less than 25% positive votes; and a clear circle indicates that no ratings were submitted for that category. The number of votes are displayed to the right of each image in a format of (positive:negative). If any

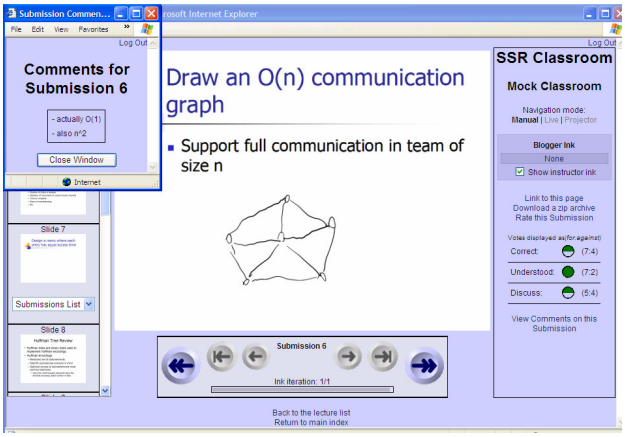

Figure 2. Student web view of rated student work, with a pop-up window of comments.

comments were submitted for that solution, a link to a pop-up comments window is also provided.

2.3 Instructor Support in Class

The instructor can view the ratings from her filmstrip and slide preview displays. In the filmstrip, circles representing each of the rating dimensions, colored as described above, appear at the top of the slide. In the slide preview display, the circles appear along with the identifying text 'Correct,' 'Understood', and 'Discuss,' as well as the number of positive and negative votes for each dimension. The instructor also has 'Show Similar' and 'Show Different' buttons. Their use takes the instructor to a student solution that was rated as being either similar or different, respectively, from the current solution that is being displayed. These features support investigation of various options for in-class review. For example, an instructor may want to review, not obviously correct solutions, but solutions that the class could not agree on as being correct or incorrect.

3. DESIGN RATIONALE FOR UP'S LPPR

We chose radio-button-style selection options as review mechanisms because we felt that, for LPPR, selection was preferable over free-form response for both supporting fast student response and for enabling automated, immediate instructor review. Automated assistance for aiding instructor review has been anecdotally requested in class sizes beginning around 15.

The first three categories of "correct", "understand", and "discuss" represent a spectrum that engages and exposes student analytical skills and provides feedback to the instructor – both on what the students specifically want to have discussed, and on how students' assessments match the instructor's. For example, if an instructor looks at a correct student submission that is highly rated as "incorrect" by the class, then perhaps the class hasn't recognized an alternative solution form.

The instructor can have a number of goals for in-class review. We take the approach that the instructor review should be, to an extent, driven by student interest. We posit that a three-way mix of student "discuss" interest, significant disagreement on the correctness of a solution, and significant selection of "don't understand" enables student-direction of instructor review.

Activity	Standard deviation in time to complete submission (sec)	Solutions available to rate (seeded + during experiment = total)	Average no. solutions rated per student	Average time per rating (sec)
Build a graph to support O(n) communication in a team	40	16 + 17 = 33	13.9	21
Design a menu such that every item has equal access time	35	14 + 17 = 31	16.7	9
Build a Huffman Tree given a set of encodings	71	8 + 16 = 24	16.5	23
Build a Huffman Tree given a set of weights	47	14 + 15 = 29	10.1	17
Brainstorm: What can you do if your project is behind schedule?	56	16 + 27 = 43	13.2	7

Table 1. Recorded information on activities performed, solutions available, and rating statistics.

However, we also strive to support the instructor in reviewing classes or categories of student-produced work. These categories may span correctness, methods of solution, etc. The generic "similar/different" category attempts to address this issue while at the same time explicitly engaging the students in comparative analysis. The goal of similar and different is that, upon showing one solution in class, the instructor can click the "different" button, which will automatically highlight a solution rated as "different" from the current solution.

Engaging the students in an additional analytical process will take time. However, this system is designed to leverage the natural time variation in student responses during active learning activities. As students finish their work, submissions are immediately sent out for review. The instructor can specify how many ratings should be sent. Students can skip or stop rating at any time, can be given the option to rate more, (perhaps outside class) or instead move on to review the summary of ratings made by others. Those students who regularly do not finish in enough time to review others' work will clearly recognize that they "missed" part of the process – perhaps providing an indicator that they need to come to class better prepared (or perhaps indicating the instructor should slow down).

4. METHODOLOGY

Our goal in this research is to explore the potential for LPPR in the context of student responses to active learning exercises. Specific goals include the following: identify if quick, selection-based criteria can engender meaningful peer review; determine if the process is, indeed, lightweight; identify what students want in terms of control and access to support peer review in class; identify what instructors want to support in class feedback; and whether LPPR can enable modeling of reflective processes.

To this end, we developed a design space and tested it in a mock classroom setting. Our goal was to gather both quantitative and qualitative data which would inform the goals listed above.

Fifteen student volunteers (twelve upper division computer science undergraduates and three graduate students) participated in a lecture in which 5 activities were performed. The "lecture" was given by one of the authors who is experienced in engaging in active learning with UP. These activities were drawn from a set of actual in-class activities from Tablet PC-using classes at the University of Washington (UW). There were two data structures activities, two software engineering activities, and one user interface activity. Before each activity, a single-slide review of the relevant material was included in the session. Using the UP interface, students would respond to the posed activity by creating an ink-based answer and submitting it to the instructor. Students would then be asked to review a set of responses. (Students could make additional submissions after reviewing.) For the purposes of the study, these responses were selected not only from the work produced by the students in the class, but also from a set of responses gathered in the actual course at UW. The students would continue rating other student responses one at a time until the instructor indicated they should stop, and then began reviewing and sharing solutions with the whole class via the projector. A short review followed in which the instructor chose submissions to review based on the ratings provided by students as well as personal preference.

The session was videotaped. One camera captured the whole room. Another roved, capturing student interactions and rating behaviors. The roving camera holder occasionally asked students to explain their behavior. At the end of the session, all participants completed a ten-question survey, which was then used in an open-group discussion. Subjects were asked about how their use of the rating system impacted both their thinking and subsequent submissions, as well as about specific details of the system's design.

To identify and elucidate the following results, we reviewed and analyzed these tapes, our research notes taken during the lecture, the student submitted work and ratings, and the surveys.

5. DATA AND OBSERVATIONS

In this section we review the data obtained from the server-recorded statistics, the student solutions and ratings, and the post-experiment survey. These inform concerns such as: how much time will this take and what kind of "work" can I expect from students, how will students react to this, and how can I use this activity to get feedback and close the loop on difficulties?

5.1 Behaviors and characteristics

Overall, the students found our preliminary LPPR interface to be both usable and understandable with little explanation. Table 1 describes each activity and provides some basic statistics on them. The time spent on each activity (both problem solving and peer review) ranged from 3.8-8.3 minutes (average 5.5 minutes).

Students reviewed quickly, exposing them to a wide variety of peer work. Students said that rating usually became easier (and we observed faster) after the first few reviews per activity. Complex graph responses took the most rating time. Huffman trees differed from the menu design solutions in that the value at each node had to be "checked" for correctness. Although reading sim-

ple brainstorming responses is fast, categorizing them as similar to one's own requires thought. All students stayed engaged in rating until time was called, unless they reached our limit of 20 ratings.

For most activities almost all students rated each activity in all categories: "correctness", "understanding", "discuss", and "similar". Correctness was the most frequently rated, and the other three categories were rated between 80% - 90% of the time. In discussion, we found that some students felt they were directed to answer in all categories to allow inter-category comparisons.

Additionally, we note that all but one activity had at least one submission to which the class expressed widely varying responses. Such a division could suggest to the instructor that this submission should be discussed with the class. Table 2 shows the percentage of submissions for each activity where 40%-60% of ratings selected "correct" (meaning the ratings for correct and incorrect were mostly evenly split). Only solutions rated by more than one student are counted. In the third column we show the percentage of submissions where 25% - 75% of ratings were "correct" (at least 25% of the class chose "differently").

Activity	40%-60% Correct	25-75% Correct
O(n) communication	27%	58%
Menu design	4%	23%
Huffman tree	0%	5%
Huffman weights	13%	30%
Project Behind	28%	56%

Table 2. Tendency to disagree on rating of correctness.

We were interested in how students interpreted the relationship between the students' "correctness" ratings and "understanding" ratings. Considering students who rated both categories, 94% of students selecting correct also selected understand (4% standard deviation across exercises). But those selecting incorrect only selected understand about half of the time (52% average, 11% st. dev., with a notable 72% on the brainstorming question). It's unclear why students claiming they couldn't understand a solution would choose to rate it correct or incorrect. However, some answers are clearly wrong, but the rater may have been unable to figure out the solution's frame of reference. We explore this issue in Section 6.3. Comparison of the correctness and discussion ratings was not performed because we suspect that the experimental design influenced these results (See Section 6.3).

For every activity, each student reviewed some submissions "similar" to their own and others "different". Although the rationales the students used in this rating cannot be impartially identified, in discussion one student clearly indicated that only one possible solution existed for each problem (which is not true), which surely impacted his similar and different ratings.

5.2 Surveys

In our post-experiment survey we asked the students (N=15) to answer ten questions on a 5 point Likert scale. The questions spanned issues of process, impact on learning, and attitudes. The most interesting questions are reported in Table 3. Discussion follows in the next section.

Question	**% Agree**
Did the rating process cause you to reconsider your solution?	93%
Did seeing others solutions and their ratings increase your confidence in your own ability?	60%
Did you find it difficult to rate and classify other students' solutions?	40%
Did you find that it got easier to rate and classify solutions over the course of the lecture?	87%
Did you enjoy previewing and rating other students' solutions?	87%
Did the rating and discussion process create apprehension about your solution of the instructor choosing to discuss it?	47%

Table **3. Survey r**esults.

6. ANALYSIS AND DISCUSSION

6.1 Is LPPR actually lightweight?

There is a high deviation in the time for students to complete their solutions for a given problem (Table 1). For the second problem, for example, there were 2*35 = 70 seconds between the slowest and fastest students within one deviation, compared to 9 seconds average to perform a peer review. The students could complete an average of about 4 submissions each within this time frame (35/9 ≈ 4). This is also the average across all exercises in the study. If the instructor wanted to give more time for the slower students to complete the activity or provide an extra minute for peer review, the number of ratings would go up substantially. The students complained about rating so many submissions per problem (15 on average), so limiting peer review to roughly the period between a student completing the exercise and before time is called for problem solving is a viable strategy, and is arguably lightweight.

6.2 Can it engender substantive reflection?

Even in a setting with little incentive to engage in (perhaps familiar) learning materials, all but one survey respondent indicated that the rating process caused them to reconsider the quality of their solution. In the discussion, students confirmed that they reflected on their own answer when rating others. One student indicated that this process started before she reached the "similar/different" selection, which was designed to specifically engender reflection. She noted that when she first reviewed an O(n) communication graph solution, the graph didn't have directed edges (arrows). She was immediately concerned, because her submission used arrows. At this point, she used the comment textbox to enter a question about the need for arrows.

The students asked for access to the instructor's controls, indirectly indicating their interest in this form of peer review. They wanted to look at all the student solutions (not just the ones they were asked to rate) and wanted to organize them by correctness ratings, etc. That is, they wanted to see the reviews provided by other students and evaluate them for themselves.

We found peer reflection could cause apprehension. 47% reported that they felt apprehensive about their solutions because of the rating process. This may be more or less pronounced in real classrooms, depending on the comfort found in a familiar community.

6.3 Does it help guide discussion?

In the actual lecture experience, sorting student submissions based on those most frequently rated correct or incorrect didn't feel especially useful to the instructor/author leading the discussion. Answers highly rated as "Incorrect" were obviously wrong and not worth discussing. The instructor found that she really wanted to discuss the solutions whose ratings were split on correctness. These were towards the middle of the solutions list sorted by correctness, creating unnecessary work for the instructor, who is under pressure to find interesting solutions quickly.

Interestingly, sorting for the most highly rated "discuss" submissions wasn't meaningful either. We believe that this may be affected by the experimental conditions. No repercussions (for learning or not) existed and students admitted that they often chose the funny or unusual items for discussion, something less likely to arise when a student's grade is on the line. Upon questioning during the mock class, a couple of students said that they skipped clicking "discuss" after rating the first few submissions for an exercise, because they didn't want to have more than a few solutions discussed. This reveals a misconception on the students' part on how these ratings were being presented to the instructor and how the instructor intended to use them.

Finally, the instructor found a need to ask the class verbally about how their reviews were coming along to avoid the process being cut off too early or allowed to run too long. Some sort of review progress indicator would have been useful.

6.4 Interface Improvements

To help the instructor identify interesting solutions to discuss we recommend an interface that can rank and highlight submissions that are "split" on their correctness rating. In the same vein, to discourage the tendency of some students to stop clicking "Discuss" after reviewing a few solutions to a problem, we suggest renaming the discussion radio buttons with the more explicit labels of "Worth discussing", "Not worth discussing".

During the mock class and in the collective interview afterwards, students noted that some solutions were not entirely correct, were not the best answer, or comprised many parts or many answers. Other students commented on the interaction between the "correct" and "understand" categories. One student made a statement to the effect that "understand was meaningless for me if I selected correct or incorrect." He elaborated that if he could identify an answer as either correct or incorrect, then he must have understood it. However, this individual's statement is not borne out by the rating statistics, where we see no real relationship between correctness selection and understanding selection.

We propose, though, a three-way selection that combines the "correct" and "understand" categories: mostly correct, mostly incorrect, and unsure. Not only does this seem to better match student categories – but it also provides a categorization useful for instructor review: In addition to reviewing submissions whose ratings were split on correctness, one would want to review those that a large percentage of the class was unsure about.

The students wanted some enhancements, such as access to the instructor controls, as mentioned above. Student comments also indicated a desire for the instructor (or TA) to review the solutions after class and indicate which answers are indeed correct.

7. CONCLUSIONS

Peer review, in a lightweight form, has significant potential for the classroom. Integrating radio-button ratings in a few key categories into an active learning toolset enables students to quickly evaluate each other's work, reflect on their own, and steer in-class discussion of exercises. Careful design of the peer review interface is required to get meaningful responses. Future work will examine the real class setting and how LPPR integrates with activities such as peer review outside the classroom.

8. ACKNOWLEDGMENTS

This work supported by a CRA-W undergraduate summer scholarship and gifts from Microsoft Research and Hewlett-Packard. We thank the UW educational technology group for providing both data and background analysis. We thank the student participants.

9. REFERENCES

[1] R. Anderson, R. Anderson, B. Simon, S. Wolfman, T. VanDeGrift, and **K.** Yasuhara. Experiences with a Tablet PC Based Lecture Presentation System in Computer Science Courses. *SIGCSE 2004.*

[2] T. Denning, W. G. Griswold, B. Simon, and M. Wilkerson. Multimodal Communication in the Classroom: What does it mean for us? *SIGCSE 2006*.

[3] A. Fekete, J. Kay, J. Kingston, and K. Wimalaratne, Supporting reflection in introductory computer science. *SIGCSE 2000*.

[4] E. Gehringer, Electronic Peer Review and Peer Grading in Computer-Science Courses. *SIGCSE 2001*.

[5] S. E. George, Learning and the reflective journal in computer science. In *Proc. of the Australasian Conference on Computer Science - Volume 4.* 2002.

[6] R. Pargas, D. Shah. Things are Clicking in Computer Science Courses. *SIGCSE 2006*.

[7] S. Sullivan. Reciprocal Peer Reviews, *SIGCSE 1994.*

[8] K. Topping. Peer assessment between students in colleges and universities. *Review of Educational Research* 68:3, Fall 1998, pp. 249-276.

[9] A. Trivedi, D. Kar, H. Patterson-McNeill. Automatic Assignment Management and Peer Evaluation. *CCSC South Central*, 2003.