Mutual Peer Tutoring: A Collaborative Addition to the Algebra-1 Cognitive Tutors

Erin WALKER
Carnegie Mellon University, Pittsburgh, PA, USA
erinwalk@andrew.cmu.edu

Abstract. Although the Algebra-1 Cognitive Tutors are effective at increasing individual learning, they lack support for collaboration between students. It is likely that students would show further learning gains if they were able to use the tutors collaboratively. We intend to integrate a mutual peer tutoring script with the cognitive tutors, such that students take turns tutoring each other on problem-solving tasks, rather than being tutored directly by the computer. The script is based on current Algebra-1 curriculum goals and situated within the cognitive tutor environment, but requires collaborative additions to the interface and cognitive model. We expect that the script, in conjunction with the cognitive tutors, will improve domain learning and knowledge of problem-solving skills.

Introduction

A cognitive tutor is a type of intelligent tutor that compares student actions during problem solving to a model of correct problem solving steps and provides context-sensitive hints, error feedback, and individualized problem selection. These tutors have been shown to be effective at increasing student learning; use of the Algebra-1 Cognitive Tutor (CT) has been shown to improve learning by about one standard deviation over traditional classroom instruction on measures of algebra understanding [1]. Because use of these tutors is strictly one-on-one, they are intended to be deployed in collaborative classrooms, where students work together in groups and teachers act as facilitators rather than lectures. Unfortunately, these ideal collaborative activities do not always occur in practice [2].

It is likely that students would show further learning gains if their use of cognitive tutors was combined with structured collaborative activities. Collaboration has been shown to increase student achievement in terms of mastery of domain knowledge and quality of reasoning strategies. However, these learning gains only occur under certain conditions. To be effective, collaboration must be structured to encourage students to engage in particular behaviours, such as providing group members with useful help, resources, and feedback [3]. To this end, many researchers develop collaboration scripts, or structured interactions with designated roles and activities for participating students. These scripts have been shown to effectively engage students in elaborate cognitive activities that promote domain learning [4].

One type of collaboration script is a mutual peer tutoring script. In mutual peer tutoring, a student first tutors a partner and then is tutored by the same partner. To be most effective, these tutoring sessions need to be structured and to incorporate rewards for good performance [5]. Tutors must prepare ahead of time [6] and, during the tutoring session, provide elaborated explanations [7] that lead the tutee to constructively use the explanations to solve the problem [8]. Additionally, making individual students aware of and accountable for the skills that they are acquiring has been shown to increase learning in peer tutoring settings [9]. In general, incorporating these elements into a peer tutoring framework tends to increase student learning.
We propose to integrate collaboration into the existing cognitive tutor framework using a peer tutoring script (PTS) that is consistent with the curriculum goals of the CT [10]. The script will be easy for students to use, because it will be built upon the existing CT interface. Students will already be aware of tutoring concepts, because they will have internalized information about the tutoring process due to their previous exposure to cognitive tutors [11]. Finally, this integration will leverage the modeling capabilities of the CT to build a cognitive model of the student collaboration that can further support a positive interaction between the students.

In this paper, I will review the structure of the current CT. Next, I will describe the script, and detail additions to the CT in terms of the interface and cognitive model. We expect that the integration of the peer tutoring script and CT will increase student algebra achievement.

1. The Algebra-1 Cognitive Tutor

The CT focuses on “the mathematical analysis of real world situations and the use of computational tools” [1]. Students first read a word problem and related mathematical questions in the Scenario Window (Figure 1). To solve the problem, students might need to use the Worksheet, Grapher, and/or Solver Windows. In the Worksheet, students identify quantities and units, answer Scenario questions, and enter algebraic expressions. The Grapher is for labelling axes, setting axes bounds, plotting lines, and intersection points. Students use the Solver to solve equations in a step-by-step fashion. Once students have completed all problem steps, they select “Done” from a menu, and proceed to the next problem.

As the student works on the problem, the cognitive tutor performs model tracing, monitoring the student’s progress based on a model of correct and buggy (i.e., incorrect) student performance. When the student makes an error, the cognitive tutor will immediately “flag” it (e.g., by turning input text red) and, for common errors, output a message in the Help/Error Window that explains the student’s misconception. At any time, the student can request help, and the cognitive tutor will provide hint messages in the Help/Error Window. For each desirable action in the cognitive model, there are hint messages at multiple levels, so if students cannot solve a problem using the first hint, they can request the next hint. Model tracing insures that students get immediate and useful feedback on their progress.

The tutor also performs knowledge tracing [12] by keeping a running estimate of student mastery of skills and areas of difficulty. These skill levels are displayed in the Skillometer window, shown in the lower right of Figure 1, so that students are aware of their progress. Once a skill bar crosses a particular threshold, it is marked as a known skill. Cognitive tutors choose problems in a unit based on the skills the student has not yet mastered.
2. Expanding the Algebra-1 Cognitive Tutor Curriculum using the Peer Tutoring Script

The PTS builds on the existing problems and tools in the CT curriculum. In the PTS approach, students are placed in homogeneous dyads with respect to ability and take turns being the peer tutor (the person doing the tutoring) and the peer tutee (the person being tutored). In the first phase of the script (or preparation phase) peer tutors perform exercises related to each problem they will be helping the peer tutees solve in the second phase (the collaboration phase). During the final phase (the meta-evaluation phase), the students have a structured discussion of the skills they used. Students should show learning gains after using this script because of beneficial collaborative interactions during tutoring and monitoring requirements of the script.

In the preparation phase, the peer tutor prepares to teach the peer tutee. The peer tutor reads through an example solution for the problem, changes certain problem characteristics that affect the solution, solves the transformed problem, and matches the skills required to solve the problem with the steps taken. For example, the peer tutor could be given the scenario and solution to this rate-of-change story problem found in the CT curriculum: “A huge mirror with a telescope is being moved by truck from Pittsburgh, PA to Charleston, South Carolina, a distance of 523 miles. The truck averages 12 miles per hour and has already traveled 70 miles.” After reading through the example solution, involving the use of the Worksheet and Grapher, students would be asked to change the values in the problem, inputting them directly into the Scenario Window (i.e., they might change the 12 to a 17 and the 70 to 53). They would then solve the new problem. Finally, students explicitly match problem-solving skill questions, such as “Can you find the slope of the equation?” to steps they have taken in solving the problem, such as typing the slope in the appropriate textfield in the Grapher. These skills are derived from the skills that the CT uses during knowledge tracing, and are normally displayed in the Skillometer in a
shortened form like “find slope”. However, for the purposes of the script, they will be phrased in the form of a question to make them easier to understand and use as a tutoring scaffold, and displayed in the Skillometer in this extended format. Throughout this process, the students will be aided by the CT, which will offer hints and bug messages as necessary.

In the collaboration phase, the peer tutor helps the peer tutee solve the problems that the tutor created in the preparation phase. As in the individual use of the CT, the peer tutee is given the problem in the Scenario window, and solves it using the relevant tools (in the above example, the Worksheet and the Grapher). However, it is the peer tutor who is responsible both for providing explanations to the tutee and for rating the tutee’s mastery of the skill questions. For example, the first step in solving the problem might be identifying the quantities involved. As the peer tutee correctly completes this step by filling in the quantity names in the worksheet, the peer tutor would increase the values of the skill bar in the Skillometer beside the skill question “Can you identify the quantities involved in the problem?” The next step might be identifying the units involved in the problem. If the peer tutee gets this step wrong, the peer tutor is expected to recognize that the tutee needs help, explain what to do, and then confirm that the tutee understands the explanation. This process continues until the problem is complete. The CT will also be used in this process to monitor and tutor the collaboration between the students, ensuring that the dyads solve the problem correctly provide each other with good explanations and feedback.

In the meta-evaluation phase, the students have a structured discussion about the skills involved in the problems. The peer tutee asks the peer tutor each skill question, the peer tutor explains how to complete the skill, and the two students discuss the merits of the peer tutor’s answer. Next, the peer tutee looks at his or her rating on that skill, explains the mistakes made, and discusses whether the rating is justified. Once the students can come to an agreement on how to answer the skill question, the peer tutee inputs the answer, and the students move on to the next skill. The purpose of this phase is to have the students reflect very specifically on the skills they employed in solving the problems.

Learning gains due to this approach are expected to come from student engagement in positive collaborative behaviors and skill-monitoring. The transfer task in the preparation phase is designed to help students move from a concrete representation to an abstract representation of the problem. The skill reflection exercise should begin the monitoring process of the students, making them aware that concrete problem-solving steps are linked to abstract skills. During the collaboration phase, the tutor’s explanations are intended to benefit both students, and it is expected that the peer tutee will use the explanations constructively. The process of rating the tutee is expected to help the peer tutor, since it requires the tutor to be very aware of the skills involved in solving the problem. It is expected to benefit the peer tutee because the tutee is more aware of his or her progress. Finally, the meta-evaluation phase is expected to be useful because students discuss and come to a consensus about the skills they were using in the previous two phases.

3. Integrating the Peer Tutoring Script with the Algebra-1 Cognitive Tutors

To implement the PTS, collaborative additions need to be made to the CT. Because it is important that the PTS is situated within the CT environment, the script uses many preexisting elements of the CT, such as the Worksheet, Grapher, and Solver tools. Some
changes involve adding additional tasks to the interface and model, such as the skill reflection task in the preparation phase. These changes are relatively simple to implement. However, the collaborative extensions are more complex and require more description.

3.1 Interface Changes for Collaboration

The primary change to the structure of the CT occurs during the collaboration phase, because collaboration is not supported in the current interface. In the collaboration phase, students are located at different computer terminals. Students share windows so that an action performed by one student is seen by the other student. Some shared windows allow some actions from one student and different actions from another student. Table 1 contains a list of the windows that will be available in the collaboration phase, their visibility to each student, the actions they allow, and their use.

Despite these necessary changes, the use of the interface in the collaborative phase will be similar to students' experiences during the individual use of the cognitive tutors. The only new window is the Chat Window, which allows the students to discuss the problem. Further, the actions that the peer tutor can perform in this phase parallel the cognitive tutor actions in the individual use of the tutor: flagging errors, updating the Skillometer, and providing feedback in the Chat Window. Student familiarity with the individual tutors should generalize to their use of the collaborative tutors.

<table>
<thead>
<tr>
<th>Interface Tool</th>
<th>Shared</th>
<th>Allowed Actions</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scenario Window</td>
<td>Yes</td>
<td>None</td>
<td>Problem description</td>
</tr>
<tr>
<td>Completed Tools from Preparation Phase</td>
<td>No - visible only to the peer tutor</td>
<td>None</td>
<td>Answers to the problem</td>
</tr>
<tr>
<td>Active Tools</td>
<td>Yes</td>
<td>Peer Tutor – flag actions Peer Tutee – input actions</td>
<td>Tools the peer tutor must use to solve the problem</td>
</tr>
<tr>
<td>Chat Window</td>
<td>Yes</td>
<td>Text input</td>
<td>Place for Peer Tutor to give hints and feedback</td>
</tr>
<tr>
<td>Skillometer</td>
<td>Yes</td>
<td>Peer Tutor – changing the values of the skill bars</td>
<td>Facilitates student monitoring of skills</td>
</tr>
<tr>
<td>Hint/Error Window</td>
<td>No - each student has a separate hint/error window</td>
<td>None</td>
<td>Place for Cognitive Tutor to give hints and feedback</td>
</tr>
</tbody>
</table>

3.2 Developing a Cognitive Model for Collaboration

Adding collaboration to the cognitive model is another change to the CT. Because the two students are at different computer terminals, the CT can “model trace” and support each student's collaborative actions, providing feedback on their tutoring behaviors. By improving the interaction between the students, the CT will hopefully increase learning gains. To achieve this difficult goal, a cognitive model has to be developed for peer tutoring, using empirical data from actual student collaboration.

In an early step toward developing the cognitive model, we conducted a pen-and-paper pretest of the PTS with two dyads of middle-school students. Observation revealed that both the process and content of the peer tutoring session needs to be modeled. Peer tutors went through a process of watching the tutee solve each problem step, comparing
the solution to their answers, and then rating the student on that particular skill. When they had to provide explanations to the tutee, they went through three steps: recognizing that the student needed help, providing explanations, and then confirming that the student understood. Should future studies support these observations, a cognitive model can be developed that emphasizes these tutoring stages. More difficult is the development of a cognitive model for the content of the tutoring sessions. This content-based model has to have definitions for how to rate students using the Skillometer, which would be similar to the knowledge tracing algorithm used by cognitive tutors. However, the model also has to include knowledge of good and bad explanations, which is much more difficult to define.

Particular process and content errors were also evident during the pretesting sessions. Students did not spend enough time verifying the tutees were correct. Peer tutors occasionally failed to recognize that tutees needed help. Regarding content, there were skills for which the peer tutors wanted to provide explanations, such as writing an algebraic expression, but were unsure of how to proceed. On a parallel note, peer tutees often failed to ask for help when necessary and did not always use the tutor's explanations constructively. Using this and further pretest data, it is possible to gather information about common student errors within the context of the PTS that can form the basis for definitions of buggy student actions. Although the development of a cognitive model for tutoring is difficult, we think the initial step of creating a model for the tutoring process in the PTS is achievable. Once that is accomplished, a model for the tutoring content can be considered.

4. Future Work

Although the CT has been effective at increasing student learning, it lacks support for collaboration, which has also been shown to be beneficial. We have proposed to add collaborative features to the existing CT curriculum, expanding the interface and cognitive model, by incorporating a mutual peer tutoring script. This script builds on students' experience with the CT by putting students in the role of the computer, asking them to "model trace" their tutee's steps, provide feedback as necessary, and update their estimates of the tutee's mastery of the skills. We anticipate that these additions will improve students' domain-content understanding and knowledge of desired domain skills, by increasing their ability to provide useful explanations and to monitor their own and others' performance.

The addition of the PTS to the CT involves design and implementation phases, followed by an experimental phase to evaluate the suitability of the script for deployment. We will be pretesting the PTS using dyads of middle school students to refine the script and aid in the design of the collaborative interface and cognitive model. Simultaneously, we will be implementing the changes to the CT and developing appropriate algebra problems to complement the script. In the fall, we will be conducting classroom studies with the modified CT and evaluating the effectiveness of the PTS. The combination of collaboration and cognitive tutor methodology should combine the benefits of both approaches.

Acknowledgements

This research is funded by the National Science Foundation, Award #0354420. I thank Ken Koedinger, Bruce McLaren, Nikol Rummel, and Mindy Kalchman for their ideas and help.
References


