Background Information

Students benefit greatly from one-on-one tutoring. Both scientific studies [1] and my own teaching experiences verify this. Unfortunately, there’s a shortage of teachers in the U.S., and it simply isn’t feasible to provide all students with one-on-one human tutors. Intelligent tutoring systems could provide a viable alternative to one-on-one human tutors. Intelligent tutoring system, or ITSs, are “computer-based learning systems which attempt to adapt to the needs of learners” [2].

A substantial amount of research has already been conducted on ITSs, including at Yale. The Social Robotics Lab recently finished a study on the ability of robot tutoring agents to shape help-seeking behaviors in children. When ITSs provide students with on-demand hints, some students use the hints to answer questions without learning the underlying material. On the other hand, some students don’t use the hints at all, even when they’re stuck. Both of these behaviors are suboptimal. The study shows that robot agents can shape these sub-optimal behaviors during robot-child tutoring interactions by reminding students to use hints and preventing them from accessing all hints at once. This further results in increased learning gains.

The above idea can be incorporated into existing ITSs and is just one of many possible ITS refinements. The Social Robotics Lab is currently designing studies to test additional refinements to ITSs, specifically robot-child tutoring systems. I intend to work on one such study.

Project Overview

Inspiration

One of my primary doubts about education software involves its ability to keep students engaged. Can a piece of software ever be as engaging as a human teacher? “Gamifying” education software is often proposed as a solution to this problem, but the effects of gamifying aren’t completely understood. For example, one study on gamified e-learning experiences found that “Students who completed the gamified experience got better scores in practical assignments and in overall score, but… these students performed poorly on written assignment and participated less on class activities, although their initial motivation was higher” [3].

One intuitive explanation for these conflicting effects involves the balance between engagement and distraction. Although educational games can be fun and engaging for students, they can also draw attention away from underlying learning objectives. Maybe the correlation between stimulus and attentiveness isn’t consistently positive. Maybe increasing stimulus increases attentiveness, but only up to a certain point, and increasing stimulus any further starts distracting and
decreasing attentiveness. There should accordingly be an optimal stimulus level that maximizes attentiveness.

Bringing this idea into the domain of ITSs, specifically robot-child tutoring systems, what should a robot-child tutoring system do to attain this hypothetical optimal stimulus level? If the system performs little to no non-task behavior, the student may not be as attentive as possible. On the other hand, if the system performs too much non-task behavior, the student may end up distracted.

The Independent Variable: Levels of Non-Task Behavior

Some some examples of non-task behavior include pausing between practice problems to play games and having the robot tell jokes. The list of possible behaviors is large and will need to be scoped down. Once I settle on a set of behaviors, I intend to test three different cases: no non-task behavior, an intermediate level of non-task behavior, and a high level of non-task behavior. I'll refer to these cases as 0%, 50%, and 100%, respectively. Note that 50% doesn't imply that system will spend half of its time engaging in non-task behavior, and 100% doesn't imply that the system will spend all of its time engaging in non-task behavior. Determining what exactly the system should do for each of the three conditions will be an important part of the project.

Justifying the coarse granularity of the study's independent variable, this study is intended to be a preliminary study. The conclusions drawn will help direct further research. For example, say that relative to the learning gains in the 0% case, the learning gains were no different in the 50% case and decreased in the 100% case. This would suggest that intermediate amounts of non-task behavior don’t affect attentiveness, and substantial amounts of non-task behavior decrease attentiveness. Follow-up studies with non-task behavior levels between 0% and 50% could be conducted to continue the search for the hypothetical optimal stimulus level.

The Dependent Variables: Learning Gains and Real-Time Attentiveness

To measure learning gains, I intend to use a pre-test and a post-test, following the aforementioned Yale study’s example. Before designing these tests and the learning interface itself, I need to pick a problem domain. The Yale study chose to work with fraction problems.

Because of the challenges associated with getting students to show up for multiple study sessions, the study itself will be conducted in a single day. As such, rather than having each student work with the robot-child tutoring system under each of the three cases, the students will be divided into three groups. Each group will work with the ITS under a specific case. The learning gains of the groups will then be compared to each other for the final analysis.

It’s important to note that learning gains are an indirect measure of attentiveness. I’m making the assumption that increased attentiveness will result in increased learning gains. Although this assumption is reasonable, real-time measures of student attentiveness could provide even deeper insights. As such, I intend to record real-time attentiveness measures for the students in the study. This is a non-trivial task that poses some interesting computational challenges. I describe the task in greater detail in the Project Tasks and Deliverables section below.
Experimental Design

One of my deliverables will be a detailed experiment description. Although I've described the backbone of the experiment in the Project Overview section above, I've left several experimental design questions unanswered. Most notably:

1. What exactly should the robot-child tutoring system do for each of the three cases: no non-task behavior (0%), an intermediate level of non-task behavior (50%), and a high level of non-task behavior (100%)?
2. What problem domain (e.g. fraction problems, reading comprehension problems) should I work with?
3. How should I measure real-time attentiveness?

Elaborating briefly on the three questions, Question 1 is highly layered. I first need to pick a subset of non-task behaviors. I could pick just one behavior or multiple behaviors. I then need to decide how to vary the selected behavior(s) over the three cases. If I decide to work with multiple behaviors, I'll be able to vary the number of behaviors in use. For the 50% case for example, I could have the student play games between sets of practice problems (one non-task behavior) whereas, for the 100% case, in addition to having the student play games, I could have the robot tell jokes (two non-task behaviors). If I decide to work with just one behavior, I'll be able to vary frequency. Take joke-telling for example. For the 50% case, I could have the robot tell jokes after every N practice problems whereas, for the 100% case, I could have the robot tell jokes after every practice problem, N being an integer greater than one. Of course, I'd have to pick some N, adding yet another layer to the question.

My answer to Question 2 will have implications for the other two questions. Regarding Question 1, it may be the case that specific behaviors are more or less distracting depending on the task being performed. For example, I find that I can listen to music while working on math homework, but music distracts me when I'm reading. Regarding Question 3, different tasks may generate different attention patterns.

For Question 3, I'll possibly have access to video feed data from the tutoring sessions. Eric Ho, another senior working on robot-child tutoring systems for his senior project, is exploring this source of information. If he can successfully track student eye contact, I'll be able to incorporate eye contact data into my real-time attentiveness measure. I could also use basic time data, e.g. time spent per question, time between button clicks. After I select my sources of data, I need to design an algorithm to process the data. A static algorithm seems less than ideal. Different students have different tendencies; imagine for example a student who looks off to the side when thinking versus a student who looks intensely at the problems he or she is working on. An algorithm based on absolute eye contact measures alone would probably generate misleading results. A dynamic algorithm that picks up on patterns in individual students’ behaviors and then looks for deviations from those patterns seems more desirable. Although machine learning algorithms can accomplish this, data involving human behavior is often noisy, and machine learning algorithms tend to struggle with noisy data sets.

Resolving these questions will involve a thorough review of existing scientific literature. Psychology literature will be helpful in answering all of the questions. Questions 1 and 2 will additionally require me to study social robotics and ITS literature, and Question 3 will additionally require me to study data analysis and machine learning literature.
Implementation

After I finish my experimental design work, I'll have a considerable amount of code to write. I'll need to program the Nao robot. I'll also need to build the tablet learning environment and data analysis pipeline. Finally, I'll need link the robot and learning environment together. Aside from code, I'll also need to write practice problems, a pre-test, and a post-test for the problem domain that I select.

Although I'd love to run tests and analyze the results before the semester ends, the majority of the Social Robotics Lab’s robot-child tutoring system studies will take place in May, a bit too late for the senior project timeline. I'll at least conduct a trial study before May with lab members, Yale students, or Scaz’s kids to ensure that the experimental setup is completely ready for actual studies.

Reference List