Abstract

Robots that Tutor in
Complex and Dynamic Environments

Nicole Salomons

2022

Robots have shown great promise in being effective tutors for both children and adults. For instance, they have been shown to be successful in math tutoring for children [203], interruptions training for adults with Autism Spectrum Disorder [204], and assisting the elderly while exercising [87]. Yet, despite the great potential of tutoring robots, most studies have focused on providing short-term tutoring either in a laboratory or a school while the researcher is present. For robots to be more broadly successful, users need to be able to practice skills on a more prolonged basis and in the wild. This requires robotic systems to autonomously tutor in unstructured settings, such as in homes and schools. Furthermore, robots need to autonomously model the user and provide personalized help without a researcher's presence.

In this dissertation, we investigate several aspects of how to create robots that can autonomously tutor in complex and dynamic environments. Our work starts by expanding the types of tasks that robots can train on, as current robotic systems mainly focus on simple tasks. We present two algorithms that allow robots to tutor more complex tasks: C-BKT and BKT-POMDP. C-BKT enables a system to model a user's skills over time, providing opportunities for a robot to offer personalized help actions earlier than to prior solutions. BKT-POMDP provides a policy for deciding a) which task to give users to test their skills, and b) which task should be chosen to maximize teaching when there are multiple skills per task. Throughout the dissertation, we provide examples of how we modeled specific complex tasks, including electronic circuits, social skills, and exercise forms.

In sequence, we investigate which robot characteristics allow for successful tutor-

ing. We explore how the role of a robot influences the interaction when tutoring adults. We compare a robot that takes on the role of a peer versus taking on the role of a traditional instructor. Our work provides evidence that a peer robot is generally viewed more favorably than an instructor robot, and that it increases learning for a participants with low prior knowledge in the domain. We also investigated how robots can indirectly influence those around them by showing that robots cause conformity. People are willing to rely on information provided by robots when they are unsure of the answer themselves (informational conformity). We also show that robots cause peer pressure on participants (normative conformity).

Finally, we demonstrate two long-term systems where peer robots operated autonomously in participants' homes. We show that robots successfully tutored and influenced participants in their homes while maintaining engagement. In the first system, a robot provided adults with motivation and coaching while doing dumbbell exercises. We show that an embodied robot is more effective than the same robot shown in a video on a tablet screen. In the second system, we built a robotic system that provided dyadic tutoring to children with Autism Spectrum Disorder and their caregivers. The robot engaged the user in social skills training via interactive games. Our clinical measures show that the robot increased children's social skills during the 30-day robotic intervention.

Robots that Tutor in

Complex and Dynamic Environments

A Dissertation
Presented to the Faculty of the Graduate School
of
Yale University
In Candidacy for the Degree of
Doctor of Philosophy

By
Nicole Salomons

Dissertation Director: Brian Scassellati

December 2022

Contents

List of Figures

List of Tables

Acknowledgments

I could not have accomplished this Ph.D. without the support of many mentors, friends, and my family. First, I thank my Ph.D. advisor and mentor, Brian Scassellati (Scaz). I am very grateful for your mentorship throughout my research. Your help and support have been invaluable during my career and have made me a better researcher. You have provided me with an excellent example of how to lead my own lab.

I would also like to thank my committee members: Marynel Vázquez, Julian Jara-Ettinger, and Tony Belpaeme. I especially want to thank Marynel for your advice and mentorship throughout my Ph.D. I also thank Tony for your invaluable feedback on navigating the UK academic system. Additionally, I thank Iolanda Leite, Roderich Gross, and Eduardo Todt for your mentorship during my time as an undergraduate researcher.

I am very grateful for all the great friends and colleagues at the Social Robotic Lab and the Interactive Machines Lab. I have thoroughly enjoyed all our conversations over the years. Aditi Ramachandran, Elena Corina Grigore, Sarah Sebo, Jake Brawer, Meiying Qin, Tim Adamson, Rebecca Ramnauth, Debasmita Ghose, Sydney Thompson, Nick Georgiou, Nathan Tsoi, Emmanuel Adeniran, Kate Candon, Kayla Matheus, Ellie Mamantov, Austin Narcomey, and Qiping Zhang, Chien-Ming Huang, Alessandro Roncone, and Olivier Mangin have made my time at Yale very special. Additionally, I want to thank the fantastic Yale undergraduates I had the pleasure of working with: Tom Wallenstein, Kaitlynn Pineda, Aderonke Adejare, Emir Akdere, Zeynep Karacan, and many others.

Thank you to all my amazing friends in the US, the UK, and Brazil who have supported me over the years, despite the distance. I also want to thank my friends from Vox Church and Trinity Baptist Church. Especially my community group, which has been a great anchor throughout this process.

I want to thank my family, Henk, Elly, Lucas and Camila, for their endless support. Thank you for always encouraging me to pursue my dreams and believing in me. I could not have done this without you. I also want to thank my family in the UK, Adrian, Beverley, Katie and James, who encouraged and cheered me on through the years. I especially thank my fiancé, James Douthwaite, for your love and support throughout this process. Thank you for always having confidence in me, making me laugh through hard times, and always being there for me no matter the distance.

Chapter 1

Introduction

Intelligent Tutoring Systems (ITSs) provide one-to-one instruction to users via a digital platform to increase their knowledge in a particular domain [69]. ITSs have great potential to supplement the instruction given by human teachers by providing personalized tutoring to children and adults via technology that trains them on specific domains. These systems usually create models of a student's knowledge states, that is, their evolving expertise across a set of skills. When an ITS has an accurate model of the student's skills, it can selectively choose problems or tasks (we will use task and problem interchangeably throughout the thesis) to focus teaching where needed. ITSs have been shown to significantly increase the user's knowledge, and can be just as effective as a human tutor [254] under the right circumstances.

Tutoring can be especially beneficial when done by an embodied robot. Compared to non-embodied agents, robots generate higher learning gains when tutoring [158]. Furthermore, an embodied robot has been shown to be seen more positively, be more enjoyable, generate higher compliance, and cause more engagement than non-embodied devices [191, 146, 20]. Lastly, a robot has the ability to act on its surrounding environment by providing demonstrations and directly collaborating with users during tasks that allow for physical interaction.

Although robots have been successful at tutoring people in various domains, a great majority of studies have focused on tutoring in very structured scenarios. Usually, participants are brought into a laboratory setting where the environment has been modified to accommodate the interaction. For instance, they interact at a table while the robot takes on a teaching role and provides verbal instructions and help. The chosen tasks for tutoring studies have also been highly structured, featuring tasks such as multiple choice questions or math questions on electronic devices. Lastly, most studies focused on short-term interactions where the robot only taught the student for one or a handful of sessions.

Current ITSs work well when teaching facts (such as the capitals of each country or periodic table names) to users, as they can be learned in a single session. However, if such a system wants to teach a new skill (e.g., how to program or how to swim) or impart behavior change (e.g., eating healthy), this can not be done in a single session. Therefore we need to build tutoring robots that can teach in the long term by enabling them to not only be effective at teaching in a laboratory setting but also give it the capabilities to tutor in the home or other in-the-wild settings. To make this happen, robots need to be accessible, as people might not want to relocate themselves to interact with the robot on a continuous basis. Furthermore, these systems need to be easy to use and act autonomously, as there will not always be a roboticist present to guide the interaction.

Expanding the range of environments the robot can tutor in brings along with it many new challenges. We need to expand the types of tasks the robot can tutor, which requires novel computational solutions that can model a user's skills without the unambiguous feedback of electronic devices. We need to investigate how a robot should interact with participants in the home and the best kind of help it can give to increase learning, compliance, and trust.

This dissertation focuses on how to create robotic tutoring systems that tutor people in unstructured environments. I investigate this through several lenses, including how to model users' skills for more complex tasks. I study how robots can interact with people in-the-wild, what roles they should take on during the interaction, and whether they can create positive change in people without direct verbal persuasion. Lastly, I demonstrate how robots can tutor people in two long-term in-home studies.

This work begins with an overview of ITSs, including their main components, such as how domains are modeled, how a system creates personalized user models, how it chooses help actions, and the different types of user interfaces. In sequence we focus on robotic tutoring, by over-viewing both single and multi-session systems. We examine several of the limitations of prior work, and the challenges associated with bringing autonomous robotic systems to tutor in-the-wild over the longer term. These topics are presented in Chapter 2.

Most prior work in intelligent tutoring systems has focused on simple tasks such as math or multiple-choice questions. However, these are not fully representative of all the tasks children and adults might encounter in the classroom or the home. During Chapter 3 and Chapter 4, we enhance prior intelligent tutoring systems to not only model simple tasks but how to also model more complex tasks. During these chapters, we focus on an electronic circuit building task as a more complex task. Due to the complexity of the task and because users are physically interacting with the system instead of using a user interface, we built a computer vision perception system. The vision system observed users while they built electronic circuits, assessing which skills were demonstrated correctly and which were not. This enabled the robotic system to select personalized actions for each user according to their skill models.

Previous user skill modeling algorithms wait for a user to complete each task before updating its model. Although this works well for quick tasks, many complex tasks take multiple minutes to complete. By waiting for the completion of the task, the system loses out on detailed information of what the user is doing and misses out on many opportunities to provide feedback and help throughout the task. In Chapter 3, we expand prior user skill modeling algorithms to allow for model updates throughout task completion. We call our novel algorithm Continuous Bayesian Knowledge Tracing (C-BKT). We first demonstrate in extensive simulation experiments that our algorithm creates a more accurate model of the user's skills and, therefore, could select better teaching actions than previous algorithms. Lastly, we show that C-BKT works on an electronic circuits task with participants during a user study.

Now that we can model a user's skills during complex tasks, we need to decide which tasks to assign a user. Task selection is used for both selecting which task to give a user to create their skill model optimally, and also to select which is the optimal task to teach the user. Prior task selection algorithms were limited to selecting tasks with single skills per task. In Chapter 4 we present a system that estimates user skill models for multiple skills by selecting tasks that maximize the information gain across the entire skill model. We compare our system's policy against several baselines and an optimal policy (where all the user's skills are known beforehand) in both simulated and real tasks. Our approach outperforms baselines and performs almost on par with the optimal policy.

Once a robot can model a user's skills during more complex tasks, we need to start thinking about the best ways a robot can deliver the tutoring content. Different studies have shown that there are many robot characteristics that influence student learning including robot embodiment [158], personality [64], and gender [206]. Therefore we need to create a broader understanding of how robots should act when teaching children and adults. In the next two chapters, we consider different robot aspects, including which teaching role a robot should take on (Chapter 5) and how a group of robots can influence people (Chapter 6).

In Chapter 5 we investigate different roles a robot can take on when tutoring, and which ones are more effective. Taking inspiration from peer learning literature in psychology, we investigate whether a robot should act as a traditional teacher or whether it should take on the role of a peer. We do this with a robot that guides participants through building electronic circuit building tasks and selects personalized help actions depending on the user skill model. This chapter shows the advantages of a robot interacting with adults that acts as a peer instead of a traditional teacher. Our results show that participants with low prior knowledge learn significantly more from the peer robot than the tutor robot. Furthermore, the peer robot is generally viewed more favorably than the tutor robot.

Chapter 6 investigates a critical aspect of creating long-term systems: the ability to positively influence people. This chapter presents work on how a group of robots can shape people's behaviors through conformity. Conformity is when a person modifies their behavior/answer to match others around them. Our study shows that robots can cause conformity in people and that two different types of conformity were present. The first is normative conformity, where a person is influenced by a group of robots due to peer pressure. The second is informational conformity, where participants change their answers to match the robots because they are unsure of the answer themselves.

After seeing the potential for peer robots to enhance tutoring and their abilities to positively influence interactions, our next goal is to bring these robots into the home. During the following two chapters (Chapter 7 and Chapter 8) we demonstrate two long-term in-home studies, where robots autonomously interact with participants over several weeks. The robots are presented as peers to the end-users to guide them in practicing particular skills. They created personalized models of the user's skills and provided appropriate feedback.

In Chapter 7 we present our first in-home system. An autonomous peer robotic coaching system spent two weeks in the home of adults, guiding them through weight training sessions. We created a machine learning system that could classify participants' exercises in real-time and provide verbal corrections. We compare the effects of a physically present robot by having a person exercise either with a robot or a video of a robot displayed on a tablet. Participants who exercised with the co-located robot made fewer mistakes than those who exercised with the video-displayed robot. Furthermore, participants with the co-located robot reported a higher fitness increase and were more motivated to exercise than participants who interacted with the robot displayed on a tablet.

Chapter 8 presents our second long-term in-home system. A robotic peer interacted with a child with Autism Spectrum Disorder (ASD) and their caregiver during a four-week period. The system was designed to be completely autonomous while modeling user engagement and providing tutoring during social skills tasks. Our system successfully increased social skills in the children with ASD, and the children showed clinical improvement on joint attention skills with adults even when not in the robot's presence. These results were also consistent with caregiver questionnaires; caregivers reported less prompting over time and overall increased communication in their children.

In Chapter 9 we discuss our results in more detail and the broader implications of our findings. We also present common themes throughout this thesis, such as the importance of modeling complex tasks, and the potential of peer robot tutors. Then, we present the many still open challenges in bringing robots into the home and other spaces. We end the thesis in Chapter 10, where we present a summary of our work.

The main contributions of this thesis are:

- The creation of novel algorithms that can model a user's skill and perform task selection during complex tasks.
- Evidence of the efficacy of a robot tutoring adults as a peer instead of a traditional tutor.

- Evidence that a group of robots cause conformity and that they generate both normative and informational conformity.
- Evidence showing that a co-located robotic coach causes individuals to perform fewer exercising mistakes than the same robot presented as a video on a tablet.
- The first system that demonstrates clinical improvements in children with ASD, using a robot peer tutor.

Chapter 2

Robots that Tutor in Unstructured Environments

This chapter provides an overview of tutoring systems, with an emphasis on systems that use robots. We start with a discussion of the advantages of human tutoring compared to traditional classroom instruction, including several successful strategies used by tutors. Despite the success of tutoring, it is not feasible for every student to have a personal human tutor. We then show how intelligent tutoring systems (ITSs) have bridged this gap by using technology that plays a similar role to a human tutor, and how an embodied robot can enhance this process. We discuss literature showing that an embodied robot increases compliance [19], motivation [231] and learning gains [158], compared to non-embodied ITSs.

In this chapter, we also present several essential elements of tutoring systems. We discuss the importance of creating and modeling domain knowledge during the interaction. We explain how domain knowledge is used to create a personalized user skill and affect model. The user model can then be used to choose tasks and help actions to teach the student any skills they have not yet mastered. Lastly, we discuss several user interfaces that are used for communication between the robot and the user, such as text based screens, virtual agents, and robots.

In the last section of this chapter, we provide an overview of creating robots that can tutor for more than one session in dynamic environments, versus those that do single-session tutoring. We discuss the limitations of current robotic research and provide several challenges that need to be addressed in order to make long-term robotic tutoring widespread.

2.1 Human Tutoring

In one-on-one tutoring, a student receives private tutoring from an expert on a particular topic. An expert tutor can act as an intermediate between the student and the content by having the student do as much of the work independently as possible while providing sufficient guidance to prevent the student from becoming frustrated or confused [173]. Tutors maintain the student in the zone of proximal development [256], which is where a student cannot do the tasks on their own but can do them as long as they have the assistance of someone more knowledgeable in the topic.

Tutors use different tactics to personalize the content to each student to maximize their learning. They tailor the material they give the student not to be too difficult and cause frustration and not to be too easy and cause boredom. The tutor can adjust the material's granularity level and provide immediate feedback to the student [62]. Additional effective tutoring strategies include guiding the student in the right direction, instead of explicitly giving them the correct answer [95], and preventing the student from spending too much time working on a problem in the wrong direction [173].

Prior work has shown the advantages of supplementing classroom instruction with one-on-one tutoring [25, 242, 33]. Those who receive personalized tutoring perform on average two standard deviations higher on tests than students who only received conventional instruction [33]. Tutoring is effective because the tutor can personalize the teaching to each student, whereas in a classroom, the teacher must balance multiple students' needs.

2.2 Intelligent Tutoring Systems

Despite the effectiveness of personalized tutoring, it is not feasible for every student to have an individual human tutor guide them at all times. There are insufficient tutors for each student, and personal tutors are often expensive, making them inaccessible to most of the population. One alternative has been intelligent tutoring systems, wherein technology takes a similar role to a human tutor. The ITS provides content and help to the user (we will use "user" and "student" interchangeably throughout the thesis) in the chosen domain. Similar to human tutoring, students that are taught by intelligent tutoring systems substantially outperform those that only received teaching from conventional classrooms [144]. There are several comprehensive reviews on intelligent tutoring systems [62, 12, 13].

ITSs can range from simple text-based interactions given on the internet to an embodied robot that talks while tutoring*. Some of the most common types of ITS include:

- **Screen-only** - Most intelligent tutoring systems are screen-only systems with little to no social element to them. They can be displayed on tablets, cell phones, or computers. They provide the content to the user via text, images, sounds, and animations. The user responds to the system by typing inside text boxes, clicking via a mouse, or with touch (if touchscreen).

- **Virtual agent** - Virtual Agents are software programs that provide automated

*Sometimes ITSs are defined as screen-based systems, but we will use a broad definition of ITS that includes any systems that provide tutoring.

tutoring to the user via either a chat box or a video of an agent on the screen. They bring a social element to the interaction and have been shown to increase learning gains compared to systems that do not present a social element [156]. Systems such as auto-tutor [105] displayed an animated agent that used natural language to teach the user. This system was shown to increase the student's knowledge by almost a letter grade. Virtual agents also have a positive effect on the perception of a user's learning experience [156]. Furthermore, virtual agents can provide nonverbal cues, which result in increased learning [9].

- **Robot** - Robots are embodied systems that occupy the same physical space as the user. Robot embodiment has shown several tutoring advantages over virtual agents [158, 20]. Robots frequently also use a screen to provide content and help to the user. More details of robotic tutoring systems will be provided in the next section.

2.3 Robotic Tutoring Systems

A robot takes up physical space in the world. When a robot takes on this physical instantiation in the world by having a body, we call this embodiment [159]. An embodied robot can physically interact with the task alongside the user. For example, a student learning handwriting can be provided demonstrations on paper for different letters by the robot [116]. Furthermore, an embodied robot can give movement demonstrations to the user, which can be especially advantageous in the physical health domain. The robot can teach a user to throw basketball hoops [162] or provide physical exercise demonstrations to the elderly [87].

Even when the task does not require physical manipulation, there are still many advantages in the embodiment of the robot compared to a virtual agent [68]. An embodied robot is more persuasive and therefore causes greater compliance than the same robot presented as a video on a computer screen [20]. Compliance is an essential aspect of tutoring as users must be willing to comply with the robot when it requests them to complete different tutoring tasks.

A second advantage of an embodied robot is that users who interact with a robot display higher increases on tests than those that interact with the same robot represented on a computer screen [158]. This is potentially because humans do not innately learn from screens [210]. A second possibility is that people view the embodied robot as more intelligent [140] and trustworthy [132] and therefore are more willing to accept information from it.

A third advantage of robot embodiment is that it generates more engagement [258] and enjoyment [258, 191] than a screen. People are more motivated to interact with an embodied robot than a disembodied agent [231]. Being engaged and entertained by the system are essential qualities if robots are to be successful in tutoring in the long term.

Lastly, robots can show certain social behaviors more clearly than agents on a screen, such as gaze and joint attention. Robots that show socially supportive behavior increase learning gains in students [218]. Additionally, robots that show non-verbal behaviors (gesture, gaze, touch, orientation) increase learning gains in users compared to robots who do so to a lesser degree [130]. Robot gestures have been shown to affect long-term memorization, performance, and engagement [67]. However, social behaviors should be used with caution as too many can lead to less learning [129], potentially because social behaviors can also be distracting.

Despite the many advantages of having an embodied robot tutor the user, there are some issues to consider. The first is that an embodied robot generates a more complex system than one which is screen only. This complexity includes additional development of hardware and software. The second disadvantage of a robot over a screen is that both custom-built and off-the-shelf robots are frequently expensive, making them inaccessible to most consumers. Lastly, robotic systems require additional dedicated space in the home or school compared to a tablet or phone.

2.4 Components of Tutoring Systems

A tutoring system is generally composed of four main components: the domain knowledge, the student model, the pedagogical model, and the problem-solving environment (user interface) [62, 80]. The domain knowledge represents the particulars of the domain, including the skills needed to complete each task and the rules of learning (for example, the problem-solving strategies that people use and the pre-requisites between skills). The student model has information about which skills the user has mastered and can also contain affect states of the user, such as whether they are tired or disengaged. The pedagogical model chooses which task to present the user and what types of help would benefit the user. Lastly, the user interface presents the content and help actions to the user. The user usually also interacts and gives feedback to the system via the interface.

In Figure 2.1 we present the interaction between these four modules. The pedagogical model uses the domain knowledge, the user's estimated skills, and the user's affective states to determine what content to present next to the user. The information is presented to the user via the interface. The user then answers questions via the interface, which is used to update the student model.

2.4.1 Domain Knowledge

The domain knowledge component represents the tasks (or problems) that will be given to the user, the skills that compose each task, and each task's solution. Although some systems can automatically generate tasks and scenarios [181], usually, the domain knowledge is designed by a human expert. The expert designs the skills

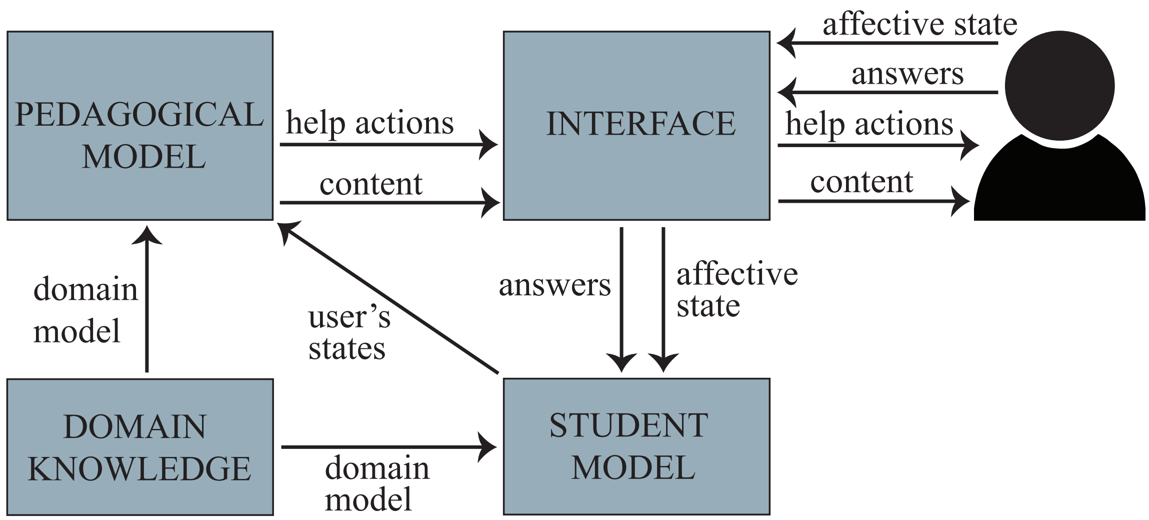


Figure 2.1: A tutoring system is composed of four main components: a student model, an interface, a pedagogical model, and the domain knowledge.

present in each task and how the system can detect when a skill was demonstrated correctly. Additionally, the domain knowledge component can include a cognitive model of how to solve each problem and how students proceed with solving it.

Although ITSs have covered a significant range of domains, they mainly have focused on mathematical domains such as algebra, geometry, and fractions [13, 189], or in domains where it is possible to give multiple choice answers [40, 232]. These domains are easy to represent in a model as they are composed of factual knowledge. Additionally, there is a single correct answer in these domains, making it straightforward for the user modeling component to model the user's skills.

Similarly, robotic systems have focused primarily on domains that are easy to represent and model, including geography [121], nutrition [235], diabetes management [112], and memory skills [245]. A significant number of studies have also focused on different mathematics subjects including geometry [99], arithmetic [119], and multiplication [199].

More research should tackle tutoring complex domains, also called ill-defined domains. Fournier-Viger et al. define ill-defined domains as those where traditional tutoring algorithms do not work well [94]. They are harder to model because they require more complex representations of skills and correct answers. Domains such as assembling furniture, building an electronic circuit, or creating a computer program can fall under ill-defined domains as modeling algorithms do not capture the user's skills well in these domains. They often are order-independent, have multiple solutions, and are completed over more extended periods.

In recent years, some ITS and robotic studies have tackled ill-defined domains that do not necessarily have one single correct answer and therefore need a more complex representation. For example, Gordon et al. modeled a child's reading skills and updated their skill model so a robot could personalize its behaviors [103]. Other studies taught users computer programming by providing feedback on their code [4, 6]. However, these more complex domains still represent the minority of studies.

2.4.2 Student Models

An ITS or a robotic tutoring system needs to create an accurate user model so it can choose the best type of pedagogical help given the current user state. Many systems can create models both of a user's skills and of their current affective state.

User Skill Models

One important aspect of intelligent tutoring systems is assessing which skills the user has mastered and which they have not. With an accurate model of the user's skills, the system can focus on giving problems and help actions to the user to teach them the skills they have not yet mastered. A system models a user's skills by observing them respond to various problems. For each problem, it observes whether the user answered correctly. The more problems the student answers correctly, the higher the likelihood that they have mastered that skill. There are several comprehensive reviews of user skill modeling, including [69] and [190].

One of the most common methods for determining which skills a user has mastered is Bayesian Knowledge Tracing [61] (BKT). BKT is a probability-based model in which each skill present in the domain is represented by a probability of mastery. To model the user's skills, BKT observes whether the student answered correctly and updates the probability of mastery for each skill present in that task. BKT accounts for a student guessing an answer correctly and for a student slipping during a problem (knowing the answer but accidentally answering incorrectly) by accounting for probabilities of guessing and slipping. BKT has been extended to account for individual learning differences, including parameterizing each student's speed of learning to increase the accuracy of the model [267].

An alternative to BKT is Learning Factors Analysis (LFA) [44], which learns a cognitive model of how users solve problems. It learns each skill's difficulty and learning rate using user data. However, LFA does not create individualized models for each user and, therefore, cannot track mastery during task completion. Performance Factors Analysis (PFA) [189] addresses LFA's limitations by both estimating individual user's skills and creating a more complex model of skills. In recent years, methods based on deep learning have also become prevalent [59]. These generate complex representations of student knowledge. However, this method requires an extensive amount of prior data in the domain [192].

Although most user skill modeling systems assume a single skill is present in each problem, several models have extended BKT, LFA, and PFA to allow multiple interdependent skills in each problem [265, 102, 187]. However, many multi-skill models assume that all skills must be applied correctly to achieve the correct answer in a problem [45, 101]. This is a significant limitation as we do not want the model to assume a user has no mastery over all skills when they might have only failed one. Furthermore, many multi-skill tasks have either order dependencies or knowledge dependencies between skills that need to be accounted for.

Affective States

In addition to user skill models, a system can model the affective state of a user, including whether they are engaged, tired, or frustrated. These can be measured in multiple ways. A computer can detect the average number of clicks, mouse moves, or keyboard inputs to detect when the user has started slowing down or is performing off task behavior [46]. Computer vision algorithms can detect gaze directions [117], or use facial action units to detect attention and emotions [236].

Despite the large potential of affective state modeling, most work in the area to date has focused on detecting disengagement. Several papers have looked at automatic engagement detection when a student interacts with a robot during a tutoring session [207, 43]. Once disengagement has been detected, a robot can deploy strategies to reengage the student [153, 38]. By detecting valence and engagement, one study deployed reinforcement learning to personalize its motivational plan for each student [104]. Engagement algorithms also need to consider whether the student is learning solo or in a group, as they have different signs of disengagement [155].

2.4.3 Pedagogical Model

The pedagogical module is responsible for structuring the instructional interventions. It uses the user skill model and domain knowledge to create personalized content and feedback for the user. The pedagogical model has two main ways it personalizes instruction: sequencing the content to the user and choosing appropriate help actions [193].

The most common way to take advantage of the user model is to determine what task to give a user. For example, Schodde et al. decides which skill to teach next based on the users' demonstrated skill [232]. Schadenberg et al. personalize the difficulty of the content to match the student's skill [230]. Other methods use a modified Partially Observable Markov Decision Process to select which gap (skill) to train the user [92] and to decide the sequence of problems to present to users depending on skill difficulty [65].

The pedagogical model is also responsible for providing personalized help to the user during task completion. The system can give many types of help actions, including giving hints, giving an example, a walk-through of the problem, and directly providing the solution to the current problem. Several pieces of work have shown the advantages of choosing personalized help actions [180, 197, 56, 148].

2.4.4 User Interface

The user interface is how the system communicates with the user. It displays the content and help actions provided by the pedagogical model to the user. The user also responds and communicates with the system using the user interface. As detailed in Section 2.2, the interface between the user and the system can vary from a simple text screen to a robot.

There are multiple ways the system can output information to the user. Most systems display content and help actions via a screen (of a tablet, phone, or computer), displaying text, images, animations, and videos. However, several systems have expanded beyond only screen visuals, allowing students to use natural language to type and communicate with the system. For example, CIRCSIM-Tutor uses natural language dialogue for input and output with the user [82]. Other systems have expanded beyond written text and use verbal dialogue to provide tutoring help, and explanations to the user [161]. When robots are used, they mostly use speech and sounds for communication.

The user also communicates with the system using the interface. When the system only has a screen, it usually receives user input via mouse clicks, typing inside text boxes, mouse or touchscreen item dragging, and so on. Some systems allow users to have a natural language dialogue via a text-based chat [82]. However, few systems have the user provide verbal answers as speech recognition and interpretation are difficult, especially when coming from children [93].

In addition to direct user input, some systems also collect indirect user input. For example, by following a user's gaze [74] and interpreting facial expressions [261]. This information can be used for user skill modeling but is most frequently used for affective state recognition. Indirect user input is frequently collected via traditional or depth cameras and processed by computer vision algorithms. Additionally, there are more invasive methods to collect affective data, such as using electrocardiogram (ECG) and galvanic skin response devices to collect user data [10].

2.5 Challenges of Long-term robotic tutors

Although ITSs are able to tutor in any environment as they usually are provided via online applications, bringing robots into the wild is much more difficult. Nonetheless, several studies have moved robot tutoring systems out of the lab into homes [204, 133, 66, 120], schools [246, 96, 122, 119, 7], and other in-the-wild spaces [109, 257, 137, 30]. This has allowed robots to tutor for multiple sessions spread over days instead of single sessions. There are several advantages of long-term tutoring, including understanding how people engage with robots after the novelty effect has worn off [152], whether students retain the tutoring material in the long term [229, 203], and the impacts of longer-term personalization [26, 118].

Even though there are several robotic tutoring studies conducted over multiple sessions, they mostly had a low number of sessions. A significant number of studies conducted five or fewer tutoring sessions [202, 119, 152, 122, 225]. Furthermore, most studies were conducted over short periods, lasting less than a week. Although some studies analyzed the interaction over months, most of these focused on either low numbers of participants [211, 176], or were non-controlled studies where the robot interacted freely with a group of people [109, 126, 257]. Few controlled studies analyzed the effect of robots for at least two weeks [224, 204, 229, 133]. Future studies should continue to study the impact of systems that tutor for more extended periods and over larger numbers of sessions, to understand the effects of tutoring robots in the long term.

The following sections describe several challenges in bringing robots outside a laboratory setting while tutoring over multiple weeks, months, or years. These challenges are some of the reasons why autonomous in-home or in-school robotic tutoring systems are not more widespread. Some of the challenges we present are primarily relevant to robotic systems (autonomy, robustness, cost), and others are relevant to any tutoring system in the home (flexibility, task complexity, content generation, personalization, adherence).

2.5.1 Autonomy

One essential characteristic for long-term robots to succeed is the system's ability to act completely autonomously. Having a human operator always present is very time-consuming and inconvenient. Either the person would need to enter the user's school and homes frequently, causing disruptions to the user, or they could teleoperate the robot, which still is time-consuming as a researcher needs to control the robot every time a session occurs.

Despite the necessity of autonomy for long-term systems, many studies still had a researcher present for robot operation. In many studies, the robot was controlled via Wizard-of-Oz (WoZ) [235, 170, 119, 211, 137, 176]. In WoZ, a human operator hides and controls the actions and utterances of the robot to give the impression that the robot acts autonomously [208]. In many other systems, although the interaction was autonomous, there was always a researcher present to set up the tutoring session and intervene if the robot had any failures [201, 203, 7]. Only a minority of studies had a fully autonomous robot with no researcher present to assist the system [204, 133, 120, 229, 224].

There are multiple reasons why autonomy is difficult to achieve. The first is that a robotic system requires many components, including sensors, computers, robots, and screens. Creating a system that seamlessly incorporates all these components can be challenging to build and program. The second is that sensing is frequently required, whether speech understanding or video processing, adding an additional layer of complexity to the system. Sensing becomes especially complex when done in real-time and in an unpredictable environment. Third, autonomous systems must be easy to start and use for the user. Many robotic systems require initiating many different components in a particular order. Autonomous robots need to be easily turned on and off by the user without requiring them to have technical knowledge.

2.5.2 Robustness

It is commonplace for robotic systems to fail while interacting with users. When this occurs in a laboratory setting, the researcher can often quickly correct the error or, when necessary, restart the entire system so that the user can continue interacting with the robot. However, failures cause much more significant disturbances in the home, as they will disrupt the interaction, and most end-users do not know how to fix them. Therefore, an error often implies that the research team needs to visit the user's home, which is time-consuming and can be seen as an invasion of privacy. For robots to tutor successfully in the long term, they need to be robust to errors. Moreover, if these errors occur, they need to be easy to resolve without too much interference from the research team.

2.5.3 Flexibility

It is possible to create a very controlled environment in a laboratory setting. The robot and cameras are in a fixed spot with a constant background. You can ensure optimal lighting and clean backgrounds for the cameras, making it easier for perception systems to interpret the data from the environment. Similarly, it is possible to control the acoustics and noise in a laboratory setting, making it easier for microphones to pick up clean audio data. However, it is not possible to control the environment when bringing the robot into homes or schools. This makes perception systems much more difficult to build as they must be flexible and work in any lighting, visual background, and background noise.

Building machine learning (ML) algorithms that work in the wild is much more challenging than making them work in the lab. ML algorithms assume the training data has been collected in a similar environment or setting that it will be used in. Therefore, when training ML models, the data is also best collected in the home or school environments, which are much harder for researchers to access. Additionally, because of the wide variety of in-the-wild settings, ML algorithms often need much more data to be collected for them to perform well. Despite a large variety of data, ML algorithms still frequently fail, as many unexpected situations occur in the home or schools they likely will not have prepared for.

2.5.4 Task Complexity

As mentioned in Section 2.4.1, most tutoring systems focus on simple tasks, such as mathematical domains in which the user memorizes facts. These types of domains can usually be learned in one or a few sessions. However, having a robot in the home provides the opportunity to teach more complex tasks. For example, over many sessions, the robot could teach the user how to play an instrument, program a computer, or dance the waltz.

There are several difficulties associated with teaching these more complex tasks. Simple skills are usually much easier to model, as they have a clear, correct answer that can be given using a user interface. Additionally, there is a one-to-one mapping between the correct answer and demonstrating the skill. Ill-defined domains, on the contrary, can have multiple possible solutions. They also need more complex sensors to model what the user is doing. A computer vision system or a natural language system is often necessary to understand what the user is currently working on and to detect whether what they are doing is valid. Moreover, complex models are needed to map the users actions on a task to skills in the domain. To create successful long-term in-home robotic systems, we need to model these complex skills so the robot can provide personalized help throughout task completion.

Despite the opportunities that long-term systems create in teaching complex tasks, most current multi-session robotic studies have focused on simple tasks. Most studies focused on subjects that are traditionally part of school curricula such as math [202, 26, 119], geography [122], and word learning [179].

2.5.5 Personalization

Many robotic studies have demonstrated the benefits of personalizing content to each student [104, 198, 26]. These benefits extend to long-term studies. Baxter et al. showed that children who interacted with a personalized peer robot demonstrated higher learning gains than children who interacted with the non-personalized version [26]. Furthermore, the personalized robot was more accepted by the children. In a study by Ramachandran et al., a robot that adapted its actions to the child's emotional and skill states during a five-session study caused significantly higher learning gains than the non-adaptive robot [203].

Despite the benefits of personalization, many long-term tutoring studies do not have automated user modeling systems [120, 7, 66]. To enable personalization, systems must first detect the user's skill or affect states. They then must decide how to use the user's state to give that personal feedback. This is made more difficult in long-term studies, as the number of user states can be significantly higher, causing additional setup and fine-tuning for each state.

2.5.6 Content Generation

In many ITSs, the content provided to the students is explicitly designed and programmed by the providers of the system. Although this is feasible for shorter-term studies, this can become cumbersome when generating content for studies that last months or even years. Content generation becomes especially complicated when coupled with systems that personalize to the user. Not only does the domain expert need to generate sufficient problems to give to the user for several weeks or months, but they also need to generate those problems for all different skill levels.

One solution is to create algorithms that automatically generate content. Although some studies have investigated automatic content generation [186] and automatic feedback generation [138], most of these have focused on simple tasks like mathematics. Therefore, more research is needed on automatically generating appropriate content for users of different skill levels.

2.5.7 Adherence and Compliance

When the tutoring interaction is done in one or a limited number of sessions, most users stay engaged and adhere to it, especially when done in the lab or when researchers are present. In the home (or other in-the-wild settings), users choose daily whether to turn on the system and participate in the session. Unlike the laboratory setting, the user does not have a pre-scheduled time or someone monitoring them to ensure they complete their necessary sessions. The system needs to be engaging or have a sufficient benefits that the user decides to incorporate the robotic system into their daily schedule, despite the lack of supervision.

It is also necessary to consider the novelty effect. Research has shown that users frequently will adhere to and use the system at the start, but that excitement wanes over time [100]. More research is necessary on making these systems sufficiently engaging in the long run so that users will decide to use the system as frequently as required.

2.5.8 Cost

Another important consideration when creating a long-term system is the cost of the final design. If the system is to assist people in dynamic environments, it needs to be financially accessible to the user. The cost of the robot, the sensors, the computer, and additional peripherals should be considered when designing long-term systems. The final system also needs to be relatively compact so as not to provide a disruption when placed in the home or schools.

Despite the importance of reducing costs, most robotic platforms are still quite expensive. The most common robot used in multi-session studies is the NAO robot [7, 201, 203, 118, 170, 119, 122]. It is a 54cm tall humanoid robot that has speech capabilities. It is commonly used as it is prevalent in many research labs and is easy to program and use. However, it is also an expensive robot, costing thousands of dollars and making it outside the purchasing capabilities of most of the population. Other platforms used in long-term studies are Robovie [126, 125, 217] and iCat [152, 151], which also cost several thousand dollars. Future research should invest in more affordable platforms accessible to the general public.

2.6 Summary

In this chapter, we reviewed the relevant literature regarding robotic tutoring systems. We discussed relevant background literature and techniques in human tutoring and intelligent tutoring systems. We present how robots were used to tutor children and adults, and the advantages of the robot being physically present while tutoring. Lastly, we presented a review of long-term robotic tutoring systems and the challenges present of creating longer term robotic systems that tutor in the wild.

In the following chapters, we will address several of the challenges in building long-term in-home robots. We explore how to create systems that can model users in more dynamic environments during less well-defined domains, and how a robot can provide personalized help actions in them. We will explore some aspects of how a robot should provide tutoring content and its capability to influence its users. Lastly, we will demonstrate in two user studies how an autonomous robot can tutor in the home for several weeks while providing personalized help.

Chapter 3

Skill Modeling: Continuous User Skill Modeling During Complex Tasks

The first step for an Intelligent Tutoring System (ITS) to provide personalized help to a user is creating an accurate user skill model. Once the system has a precise model of which skills the user knows, it can focus on teaching where needed. There are numerous user skill modeling techniques [69], however, they primarily focus on simple tasks. Furthermore, the most common techniques, such as Bayesian Knowledge Tracing [12] and Learning Factors Analysis [44], wait until the end of the task to update the user model. For robotic tutoring systems to become pervasive, especially in novel spaces, they need the ability to model all different types of tasks. Many of these tasks will be more complex and take much longer to complete. If the tutoring robot waits until task completion to provide help, it loses out on many opportunities to offer assistance. Furthermore, if we are to bring robots into places such as the home or industry, we cannot limit users to interacting with the robot only through a computer screen. The robotic system needs to interact with the user through other inputs methods such as natural language or computer vision. The disadvantage of these modes is that they are less accurate and often generate noisy observations of the user's answers.

This chapter presents a novel algorithm called Continuous Bayesian Knowledge Tracing (C-BKT) that tracks users' skills throughout more complex tasks. Our proposed solution has two main innovations: the first is that it considers how long each skill takes on average to be attempted. This means that the algorithm can model the user throughout task completion. The second is that we average the previously seen observations, which means that noise has a smaller impact on the model's accuracy. We show in simulation that C-BKT has a more accurate model of the user's skills and, therefore, can select better teaching actions than previous algorithms. Lastly, we show that C-BKT works on building a user skill model during and electronic circuits task during a user study.

3.1 Introduction

Prior intelligent tutoring systems have primarily focused on simple tasks such as arithmetic or multiple-choice questions. In these domains, there is a single correct answer. The answer is given either through a tablet or web interface, therefore generating unambiguous observations about the user's answer. The ITS system uses these end-of-task observations and updates the user skill model depending on whether the answer was correct or incorrect for each skill. For example, if the task tests a division skill by asking: "what is 14/2?" the user will answer 7, a different number, or leave it empty. If the answer was 7, the system increases its estimate about the user's division skills; otherwise, it decreases it.

Consider a more complex task, such as electronic circuit building or computer programming. These tasks generate opportunities for the system to intervene with help by tracking user skills before the user provides a final answer. Several difficulties arise from modeling throughout task completion. There are often multiple possible ways to complete each task correctly. Frequently these tasks test more than one skill, some of which are expected to be completed earlier than others. Each observation does not tell a complete story about the user's skills as they apply different skills over time. Lastly, observations can be noisy as sensing systems like computer vision or interpreters are necessary. These are exemplified in a programming task: there are multiple possible solutions; the user needs time to apply each skill, with some skills like creating a loop likely taking longer than others such as creating variables; users will likely break and rebuild pieces of code during the task; the observations are noisy as a language interpreter is necessary.

Previous skill estimation algorithms were not designed to model these more complex tasks. Therefore, this chapter proposes Continuous Bayesian Knowledge Tracing (C-BKT), which can model a user during more complex domains. There are two main novelties in our proposed solution. First, we created an "attempted parameter" that captures the expected time for users to apply each skill if they have mastery over it. This means that the system does not immediately assume the user does not have mastery of a skill if they do not demonstrate it in the first time-step. Additionally, the system takes longer to penalize incorrect skills demonstrations when the skills are more complex or dependent on other skills. Our second algorithm novelty is that we average the skill estimates of the user over multiple time-steps. By averaging estimates, the system is less vulnerable to sensor errors caused by noisy observations, as one incorrect measure will have a more negligible effect on the estimate. Furthermore, the user needs to demonstrate the correct application of a skill multiple times in a row before the system makes decisive conclusions about the user's skills.

To validate C-BKT, we compare it against three variations of Bayesian Knowledge Tracing [12] (a commonly used method in ITS): the standard BKT model as originally proposed where it is only updated at the end of the task, one where the user model is updated at each time-step based on the model value of the previous time-step, and one where the model is updated from the initial belief at every time-step. We perform three sets of experiments. The first two were done in simulation, where we randomly generated tasks, skills, and users. The first experiment shows that C-BKT has a more accurate model of the user's skills throughout the interaction. The second simulation experiment shows that C-BKT chooses significantly better skills to teach the simulated user than the other algorithms. In our third experiment, we collect data from human participants and show that C-BKT teaches new electronic circuit skills to participants*.

3.2 Continuous Bayesian Knowledge Tracing

In this section we first present the traditional Bayesian Knowledge Tracing (BKT) framework. In sequence, we will review some of the disadvantages that the conventional methods present. Lastly, we present our model called Continuous Bayesian Knowledge Tracing, which solves some of the problems that more complex tasks produce.

3.2.1 Bayesian Knowledge Tracing

Bayesian Knowledge Tracing (BKT) learns whether a user has mastery of a specific skill by observing the user completing tasks [61]. The estimate of the user's skill at time t is represented by $p(L_t)$ and is initialized by $p(L_0)$. Each skill has a probability of being guessed correctly $p(G)$ and a probability of the user slipping $p(Sl)$ (making a mistake despite the skill being known). Additionally, the model has a probability of transitioning ($p(T)$) from a non-mastered state to a mastered state whenever the user has an opportunity to try it.

*The code and the data can be found at https://github.com/ScazLab/C-BKT.

Mastery Probability Initialization

The probability of mastery of the user is set to its prior at the start of the interaction (Equation 3.1).

$$p(L_1) = p(L_0) \tag{3.1}$$

Mastery Probability Update

The model observes whether the user got the correct or incorrect answer after completing the task and uses it to update the probability of mastery. To update the mastery when the observation is incorrect (Equation 3.4), the new estimate is the prior times the probability that they slipped, divided by the total probability of an incorrect answer (Equation 3.2). When the observation is correct (Equation 3.5), the updated probability of mastery is the prior probability of mastery times the probability that they did not slip, divided by the total probability of a correct answer (Equation 3.3).

$$p(o_t = 0) = p(L_t) \cdot p(Sl) + (1 - p(L_t)) \cdot (1 - p(G)) \tag{3.2}$$

$$p(o_t = 1) = p(L_t) \cdot (1 - p(Sl)) + (1 - p(L_t)) \cdot p(G) \tag{3.3}$$

$$p(L_t | o_t = 0) = \frac{p(L_{t-1}) \cdot p(Sl)}{p(o_t = 0)} \tag{3.4}$$

$$p(L_t | o_t = 1) = \frac{p(L_{t-1}) \cdot (1 - p(Sl))}{p(o_t = 1)} \tag{3.5}$$

Transition Probability

The probability of the user going from an non-mastered state to a mastered state is calculated from the probability of them already having mastered the skill plus the probability of them not having mastered the skill times the probability of them transitioning (Equation 3.6).

$$p(L_{t+1}) = p(L_t|o_t) + (1 - p(L_t|o_t)) \cdot p(T) \tag{3.6}$$

3.2.2 Bayesian Knowledge Tracing Limitations

The BKT model was designed for tasks where unambiguous observations of the user are given at the end of each task. It would be advantageous for the system to create an accurate model and provide help throughout the task. With some simple modifications, the BKT expression could be adapted to allow for continuous modeling. One option would be to use the BKT update equations after every time-step. However, this quickly brings the estimate to one of the extremes ($p(L_t) = 0$ or $p(L_t) = 1$), especially if many identical observations are seen in a row. Another option is to update it every time-step using the initial mastery estimate (L_0). When doing this, the mastery jumps between high and low mastery every time the observations change. Furthermore, neither of these two proposed solutions considers whether the user is currently at the start or end of the task. Towards the end of the task, the user has had more time to demonstrate their skill mastery.

3.2.3 C-BKT

We propose an extension of BKT that continuously updates its estimate of the user's skills during task completion. We call it Continuous Bayesian knowledge Tracing (C-BKT). In addition to the BKT parameters, we introduce a new variable called *attempted*. The attempted parameter ($E[k]$) is the expected number of time-steps it would take for the user to have had time to attempt the skill k. It can be estimated from prior data or by an expert in the field. For example, in the programming domain, we would not expect the user to have completed a FOR loop after the first second of the task. Rather, it would likely take them a minute or more to attempt it. The second main modification to BKT is that we average the n previous time steps to determine the current estimate of the user's skills. This prevents the system from jumping between low and high mastery states.

Probability of an Observation

When the user has attempted the current task, the probabilities of the correct and incorrect observation are identical to BKT (Equations 3.7 and 3.8). When they have not attempted it, the probability of an incorrect observation is guaranteed, whereas the probability of a correct observation is zero (Equations 3.9 and 3.10).

$$p(o_t = 0|A = 1) = p(L) \cdot p(Sl) + (1 - p(L)) \cdot (1 - p(G)) \tag{3.7}$$

$$p(o_t = 1|A = 1) = p(L) \cdot (1 - p(Sl)) + (1 - p(L)) \cdot p(G) \tag{3.8}$$

$$p(o_t = 0|A = 0) = 1 \tag{3.9}$$

$$p(o_t = 1|A = 0) = 0 \tag{3.10}$$

Attempted Probability

The probability of a skill having been attempted is the current time-step divided by the number of expected time-steps to complete it. If the number of time-steps passed has exceeded the attempted parameter, it is assumed that the user would have attempted it if they had mastered that skill (Equation 3.11).

$$P(A|t) = \begin{cases} \dfrac{t}{E[k]}, & \text{if } t \leq E[k] \\ 1, & \text{if } t > E[k] \end{cases} \tag{3.11}$$

Mastery Probability Initialization

Similar to BKT, the probability of mastery is equal to the prior estimate (Equation 3.12). However, it will not change over time. We create a new variable called $P(H_t)$, which denotes the current estimate of the user's skills at time t. $p(L)$ is used to update the current temporary mastery $P(H_t)$ over time.

$$p(L) = p(L_0) \tag{3.12}$$

Mastery Probability Update

As seen in Equation 3.13, instead of looking at each time step individually, the algorithm updates its current estimate ($p(H_t)$) by averaging the previous n time-steps. If the current observation is that the user applied the skill correctly, then the task must have been attempted, and the traditional BKT equation is used (Equation 3.14). When the observation is incorrect, there are two possibilities: either the task has been attempted, but the person did not demonstrate the skill, or the task has not been attempted yet. Equation 3.15 measures the probability of mastery considering both scenarios and divides it by the total probability of an incorrect observation.

$$p(H_t) = \sum_{i=t-n}^{t} p(H_i|L, o_i, i) \tag{3.13}$$

$$p(H_t|L, o_t = 1, t) = \frac{p(L) \cdot (1 - p(Sl))}{p(o_t = 1)} \tag{3.14}$$

$$p(H_t|L, o_t = 0, t) = \frac{p(L) \cdot [p(A|t) \cdot p(Sl) + (1 - p(A|t))]}{p(A|t) \cdot p(o_t = 0) + (1 - p(A|t))} \tag{3.15}$$

3.3 Comparison to Traditional Methods

C-BKT will be compared to several different variations of the traditional Bayesian Knowledge Tracing model given a specific observation. We will compare the following models:

- **T-BKT** - The traditional BKT where the model updates only at the end, using the final observation.
- **I-BKT** - Modification of BKT, where it updates its current estimate using the initial belief value during each time-step.
- **E-BKT** - Modification of BKT, where it uses the model value of the previous time-step to update the value of the current time-step.
- **C-BKT-AT (Only Attempted)** - The C-BKT model with only the attempted parameter.
- **C-BKT-AV (Only average)** - The C-BKT model, but only averaging the beliefs over time. It averages the previous 10 time-steps.
- **C-BKT** - Our proposed algorithm, were it also average the previous 10 time-steps.

First, we give an intuitive demonstration via a toy example of the pitfalls of traditional BKT when the interaction is multiple time-steps long. We graphically show how C-BKT solves those problems. Let us consider a task where a person is building an electronic circuit that requires a resistor, and the user is given 60 time-steps to complete the task. The user adds the resistor at time-step 22, removes it

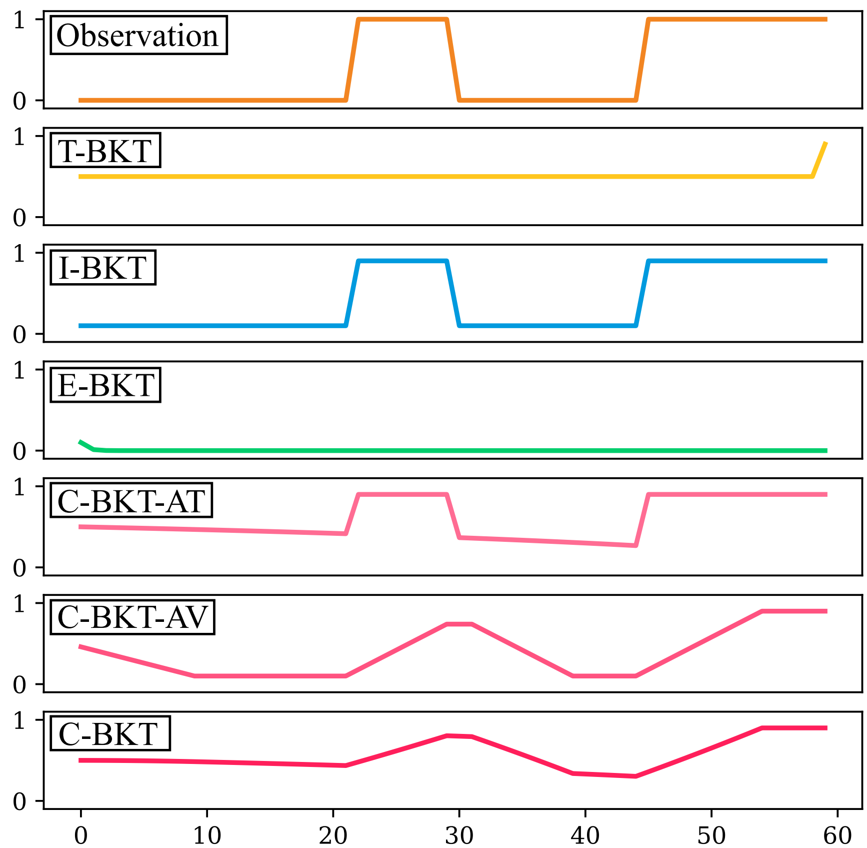


Figure 3.1: A comparison of how different variations of traditional BKT update the belief of a particular skill after specific observations at each time-step. We also demonstrate the effect of the different modifications of C-BKT and how the final C-BKT updates its belief.

at time-step 30, and then returns it to the same position at time-step 45 for the remainder of the time. The observation is 0 when the resistor is not on the board and 1 when the resistor is on it.

We set $E[k] = 60$, meaning that if the user has mastered the skill, they are expected to have demonstrated it within 60 time-steps. We set $P(L_0) = 0.5$, meaning that the system has complete uncertainty of the user's skills at the start of the interaction. Lastly, we set $P(G) = P(Sl) = 0.1$, meaning that there is a 10% chance the user will guess correctly or accidentally slip during the task.

In Figure 3.1, the C-BKT and conventional BKT methods are compared with respect to their belief of the resistor skill over the task completion. T-BKT is shown to update only at the end, which means it loses the opportunity to make informed decisions throughout the task. Because E-BKT is updated every time-step, when several incorrect observations are made in a row at the start, it quickly brings the belief to zero. It would need many correct observations to recover. Lastly, I-BKT jumps between higher and lower belief states with correct or incorrect observations, since it uses the initial belief to update rather than using any history.

Examining the trace for the C-BKT-AT approach, the effect of the attempted parameter can be observed. The belief lowers very slowly at the start (as the person has likely not had an opportunity to demonstrate their skills yet) and then decreases faster when more time-steps have passed. The C-BKT-AV approach shows the result of averaging the current belief of the previous ten time-steps. Instead of jumping from high to low states, it takes several rounds of the same observation to impact the belief significantly. Finally, C-BKT shows the result of the attempted parameter and the average combined. The model creates a smoother model of the user's skills and considers how far along the user is in the task.

3.4 Simulation

We examine the presented algorithms under two experimental conditions. The first focused on user modeling and the second on the effects of teaching. The performance of each algorithm is examined across 1000 rounds of simulated tasks, each initialized with randomized skills, tasks and users.

Skills - During each round, different skills were created. Each skill had associated with it a probability of guessing and a probability of slipping, randomly chosen from a uniform distribution between 0.1 and 0.25. The attempted parameter was set to a random uniform distribution between 40 and 150.

Tasks - During each round, a new task was created. The task was assigned between five to ten skills. Each simulated user was given 180 time-steps to complete the task.

User - During each round, a simulated user was generated. For each skill, they were randomly assigned as mastered or not with equal probability. We specify as T^i the true state of the user for skill i. The belief state b of the user was set to 0.5 for all skills at the start of the round.

Observations - During each time-step an observation is generated for the user. The observation was generated via the probability of a correct or incorrect observation (Equations 7-10), times the probability of the skill having been attempted (Equation 15).

Teaching - Every 20 time-steps, the user is taught one of the skills. The chosen skill is the one with the lowest estimated mastery state. The probability of learning a skill (when it was not previously known) is randomly drawn from a uniform distribution between $0.15 - 0.35$. If they have learned it, then their mastery of that skill goes from 0 to 1.

3.4.1 Experiment 1: User Skill Modeling

We calculate how C-BKT compares to T-BKT, I-BKT, and E-BKT in how far the estimate is from the true belief T^i at every time-step. In Experiment 1, we assume that no teaching has occurred and focused on skill modeling accuracy. At each time-step, the distance between the belief at time-step t and the true state of the user (Equation 3.16) is measured by Kullback-Leibler Divergence (KLD) [145].

$$D(b,T) = \sum_{i \in skills} b_t^i \cdot log\frac{b_t^i}{T^i} + (1 - b_t^i) \cdot log\frac{1 - b_t^i}{1 - T^i} \tag{3.16}$$

Figure 3.2 shows the KLD of estimate b at each time-step for the different BKT variations. T-BKT only updates its belief at the end, and therefore remains constant throughout the interaction. At the start, E-BKT performs the worst of all the algo-

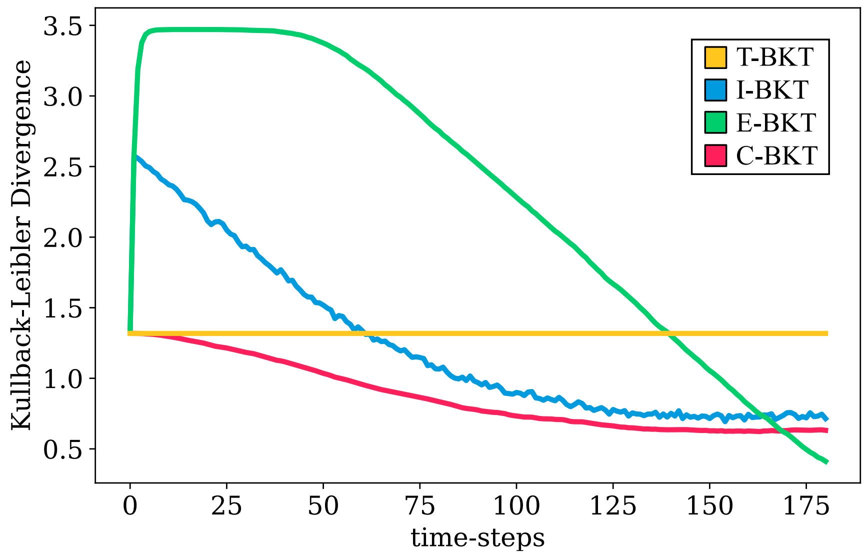


Figure 3.2: A comparison between the models of the KLD distances from their estimated state and the user's real state.

rithms but corrects its mistakes at the end when observations are more reliable. Both C-BKT and I-BKT improve their estimates as time progresses. However, C-BKT outperforms I-BKT throughout all time-steps.

At four different time-steps (time-step 30, 80, 130, and 180), we measure whether the KLD skill accuracy was significantly different between the different models using an ANOVA with a post-hoc Tukey HSD test. At all different time points, the different models were statistically significant from each other ($p < 0.05$). C-BKT significantly outperforms T-BKT, I-BKT, and E-BKT after 30, 80, and 130 time-steps. However, after 180 time-steps, E-BKT had a better model of the user's skills.

3.4.2 Experiment 2: Skill Modeling with Teaching

During Experiment 2, the simulated user was taught a skill every 20 time-steps. To measure how much a simulated user has learned, we measure the number of skills they had mastered at the start of the interaction (time-step 0) compared to the number of skills they had mastered at the end of the round (time-step 180) using Equation 3.17. We also measure how many skills the user would have learned if the system had a

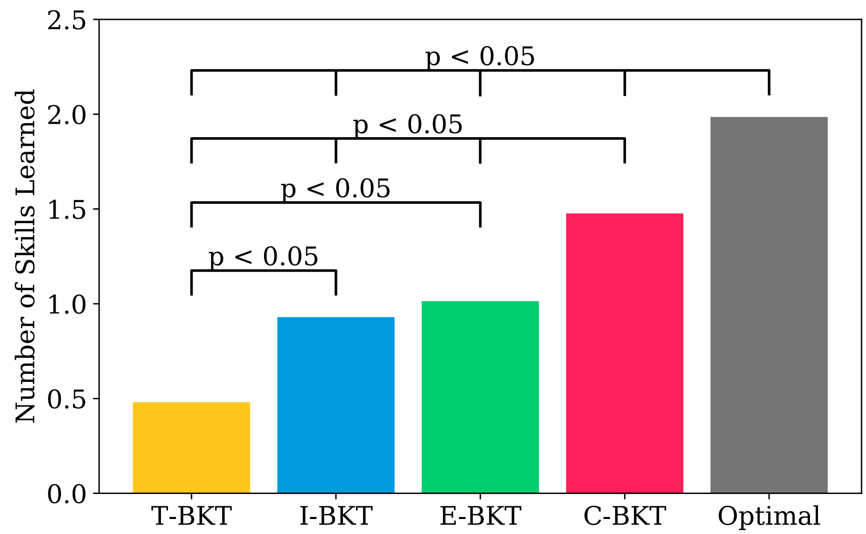


Figure 3.3: The BKT variations compared with respect to the average number of skills learned over 1000 time-steps.

perfect model of the user's skills. We call this the optimal model, as it can choose the best skills to teach.

$$D(S_{start}, S_{end}) = \sum_{s \in skills} S_s^{end} - S_s^{start} \tag{3.17}$$

Figure 3.3 shows the number of skills the user has learned on average for C-BKT and the different traditional BKT variations. On average, the simulated users in T-BKT learned 0.42 ($SD = 0.49$) new skills; users in I-BKT learned 0.91 ($SD = 0.57$) new skills; users in E-BKT learned 1.00 ($SD = 1.15$) new skills; users in C-BKT learned 1.44 ($SD = 0.82$) new skills; and users with the Optimal model learned 1.89 ($SD = 1.15$) new skills. The models differed statistically significantly using an ANOVA with post-hoc Tukey HSD Test in all cases, except between the I-BKT and the E-BKT models. C-BKT outperforms all the variations, and is only behind the optimal model.

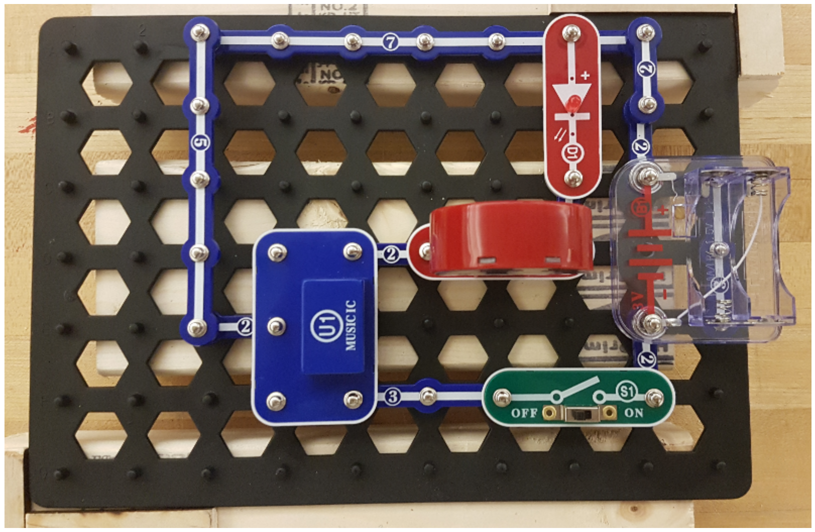


Figure 3.4: An example of a completed circuit using snap circuits [78]. This circuit plays music and blinks a light in the rhythm of the music, when the switch is turned on.

3.5 User Study

We demonstrate how C-BKT was implemented on a real task with human participants. Participants were asked to complete electronic circuit tasks. We use snap circuits [78], where the pieces can be snapped together on a board to form circuits. An example of a built snap circuit board can be seen in Figure 3.4. Electronic circuits encapsulate well how to model skills during more complex tasks, as there are multiple correct ways to create a circuit. Users will be adding, moving, and removing pieces on the board during the interaction. Moreover, the observations are noisy since a computer vision system detects what pieces are added to a circuit board.

Skills - There were eight different pieces that a person could add to a board: a switch, a button, a resistor, an LED, a music circuit, a speaker, a motor, and wires. Different skills were tested, including adding correct pieces, creating a closed circuit, LED directionality, how to create AND and OR gates, and so on. A total of 17 skills were being tested. The parameters for each skill were determined by consulting an electronic engineering major.

Tasks - There were 32 variations of tasks, of which each user completed 10. Each task required a combination of different skills. Participants were told which task to complete next via an app on a tablet. They were given up to three minutes for each task. Some examples of tasks were: "Build a circuit that plays music when a switch is turned on" and "Build a circuit that spins a motor when a switch is turned on or a button is pressed". Each task had a degree of difficulty associated with it, and the next task was chosen according to the user's skill.

Users - There were 37 participants in the experiment (18 male, 18 female, 1 non-binary). The study was approved by the university's Institutional Review Board, and participants signed a consent form. They were not provided with any information on how electronic circuits worked, other than the piece's name and the ports on the pieces. Participants completed a pre-test and a post-test to determine their knowledge of circuits before and after the interaction.

Observations - An overhead camera observed the user as they completed each task. A vector of observations was generated at each time-step for the task. If the user demonstrated the correct skill, the observation for that skill would be 1; if they did not demonstrate the skill, it would be 0; and if a skill was not tested during that task, it would be a 2.

Teaching - Every 30 seconds, a robot provided help. The help action varied between pointing out wrong pieces on the board, suggesting pieces to add, explaining how to connect pieces, and affirming that a skill they had demonstrated was correct. The user had the option to press a "finished" button on a tablet. Upon indicating they had finished, the robot would provide further help if one of the skills was incorrect. If the task was correct, it would move on to the next task.

3.5.1 Procedure

First, participants completed a pre-test without the robot to assess their skills on electronic circuits ahead of the tutoring interaction. The pre-test was composed of the participant building circuits, and responding to questions about circuits. Then they built ten electronic circuit tasks alongside a robot, while the computer vision estimated their skills via C-BKT, and the robot provided help. After the interaction, participants completed a post-test to assess how many new skills they had learned. The pre-test and post-test were nearly identical and tested the same skills.

3.5.2 Finished Signal

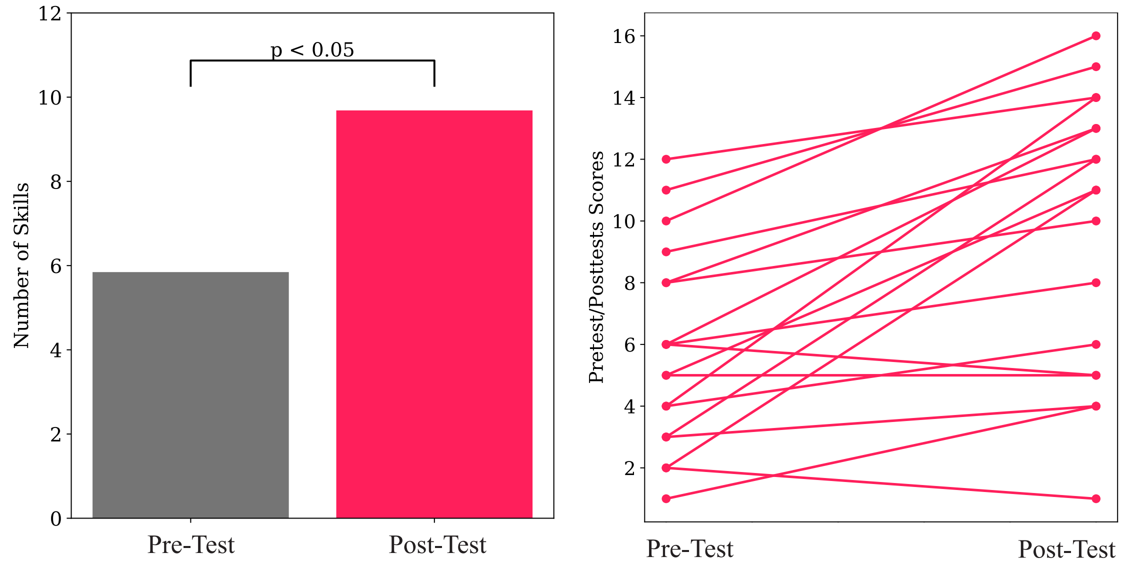
One simple addition to C-BKT was that the user could signal via a tablet when they were finished with the task. We interpret the user pressing the button, as signalling that they have attempted all the skills. Therefore, when the participant pressed the button, we update the prior p(L) with Equation 3.18.

$$p(L) = P(H_t|o_t) \tag{3.18}$$

3.5.3 Results

Participants demonstrated wide variability in their skills on electronic circuits, varying from only demonstrating 6% of skills on the pre-test to showing 71% of skills. We compare how many skills the participant correctly demonstrates from the pre-test to the post-test. On average the participant demonstrates correctly 5.83 ($SD = 3.24$) skills on the pre-test, and 9.67 ($SD = 4.49$) skills on the post-test. A t-test shows that participants knew significantly more skills during the post-test than during the pre-test ($t(18) = 8.64, p = .006$). These results are shown in Figure 3.5(a). Figure 3.5(b) shows the improvement of each participant between the pre-test and post-test. 83% of participants improved their skills after the interaction, 6% did not learn any additional skills, and 11% showed fewer skills on the post-test compared to the pre-test.

In Figure 3.6, we give an example of how C-BKT tracked one participant's LED



(a) (b)

Figure 3.5: (a) Participants demonstrated a significantly higher number of skills in the post-test compared to the pre-test. (b) The pre-test and post-test scores for each of the participants.

skill's estimate. The dashed lines in the figure are the moments the participant pressed the finished button. In the observation graph, it can be seen that the participant removed the LED on and off the circuit several times. The computer vision also detected the user as having their hand on top of the board 32% of the time. Despite the noisy observations, C-BKT still was able to track their skills and had a high estimated belief that the user knew how to use the LED.

3.6 Discussion

We first discuss our proposed solution, and in sequence present some discussion on the attempted parameter.

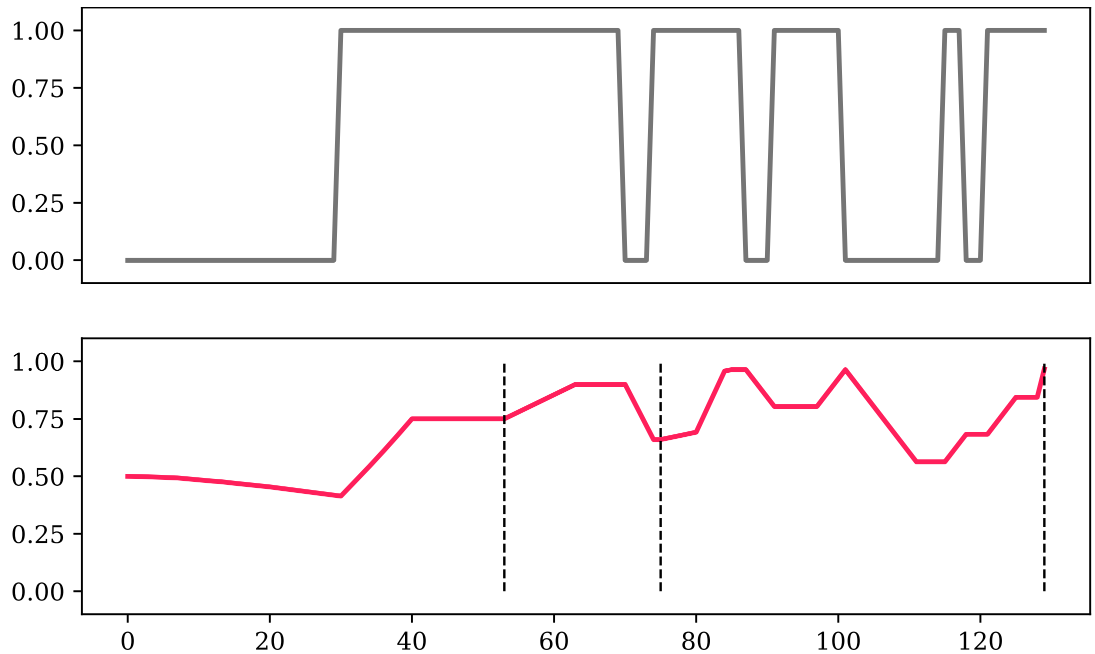


Figure 3.6: An example of the observation and C-BKT's resulting skill estimate of a participant LED's skill during a particular task.

3.6.1 Continuous-Bayesian Knowledge Tracing

In this chapter, we have shown that C-BKT can model a user's skills during complex tasks. Experiment 1 shows that C-BKT models a user's skill more accurately than traditional Bayesian Knowledge Tracing systems. This is because the conventional BKT approach was designed to only model a user's skills at the end of the task when it has received an unambiguous answer from the user. In Experiment 2, it is shown that accurately modeling a user's skill during the task allows the system to choose good skills to teach a user. It teaches significantly more skills to the user compared to traditional BKT variations.

Lastly, we validated C-BKT on a user study with participants building electronic circuit tasks. This demonstrated the applicability of the algorithm to real-world tasks where participant data must be recovered using a sensing system. By modeling users using C-BKT, the system taught users skills relating to electronic circuit design, and demonstrated to have significantly increased participant knowledge on circuits from pre-test to post-test.

3.6.2 Attempted Parameter

The attempted parameter captures the expected amount of time before the user would have tried out a skill. This allows C-BKT to modify the weight of user observations at the start of the interaction and therefore make fewer mistakes. In our algorithm, we assume the attempted parameter is a fixed value throughout the task. However, for future systems, a more advanced computer vision system may be able to provide greater activity resolution by detecting what the user is doing at every time-step. This would provide a more accurate probability that they have attempted each of the skills of the current task. A second addition that we leave as future research is to enhance the attempted parameter by defining order dependencies between the skills. Often the ability of the user to attempt one skill is dependant on another skill being demonstrated first. For example, it is not possible to correctly have an LED on a board in the correct direction before the LED is added. These order dependencies would make the attempted parameter more accurate.

3.7 Summary

For a tutoring system to give appropriate help actions, it first needs an accurate model of a user's current skills. In this chapter, we extend previous knowledge tracing algorithms to account for more complex tasks that take longer to complete and have noisy observations of the user's skills. We first demonstrated that our C-BKT algorithm created more accurate models than prior solutions in a simulation. Next, we showed in a simulation that our system can choose better tasks to teach than previous solutions. Lastly, we demonstrate on the task of electronic circuit building that our solution creates models of user's skills with live participants. Using each participant's model, our system selected personalized help actions. Participants show significant electronic circuit knowledge increases from pre-test to post-test.

The C-BKT model is helpful in many different scenarios. The main application (and the one that was the focus of this chapter) is intelligent tutoring systems. Having an accurate model of a user is essential to tutoring. As ITSs become more predominant and are used for a broader range of tasks and ages, it is vital that not only simple tasks be considered, such as math or multiple choice, but also more intricate tasks where more fine-grained feedback may be required. Especially in light of the COVID-19 pandemic, ITSs can remove a bit of the strain on teachers and parents by providing personalized help to a student.

C-BKT can also be used in scenarios other than in tutoring, such as in collaborative manufacturing. Collaborative manufacturing tasks are very different from those seen in classrooms and take much longer to complete. Some examples include: assembling cars or furniture, building circuits, or doing laboratory testing. An algorithm such as C-BKT could model the user throughout these complex tasks and provide feedback to the operator. With an accurate model, a robot could then take over tasks that the user is less confident in, or provide instruction where help is needed.

This chapter focused on user modeling in simulation or in a laboratory setting. However, creating an accurate user model is only the first step in building better tutoring systems. In the next chapter (Chapter 4), we investigate how to do action selection given more complex tasks. In Chapter 5, we provide further details of how C-BKT was used to tutor participants during the electronic circuit building task.

Chapter 4

Skill Modeling: Task Selection for Optimal User Information Gain*

The last Chapter presented C-BKT, which could create an accurate model of a user's skills. The tutoring system can use this model to focus training on essential and deficient skills. This Chapter focuses on how a system can select the optimal task to maximize learning. Prior work offers mechanisms for optimally creating models of users on a single skill. Other work has conducted tutoring with tasks that contain multiple skills per task, but these do not perform optimal task selection. To the best of our knowledge, there is no work done in choosing what tasks to assign a user when there are multiple skills present per task.

This chapter presents a system that estimates user skill models for multiple skills by selecting tasks that maximize the information gain across the entire skill model. We modify a Partially Observable Markov Decision Process (POMDP) to make it computationally tractable in selecting which task to hand a user. We compare our system's policy against several baselines and an optimal policy (that assumes full

*Parts of this chapter were originally published as: Nicole Salomons, Emir Akdere, Brian Scassellati. (2021). BKT-POMDP: Fast Action Selection for User Skill Modelling over Tasks with Multiple Skills. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI)* [219].

prior knowledge of the user's skills) in both simulated and actual tasks. Our method outperforms baselines and performs almost on par with the optimal policy when selecting which task to test and train users on. We demonstrate in a user study that our solution optimally selects tasks in real time.

4.1 Introduction

There has been prior work on selecting which task to present a user to maximize their learning [232, 65]. However, these systems consider that each task assigned to a student maps one-to-one with a modelled skill, an assumption that frequently does not hold. Consider a simple math task: $(3 * 9)/(1 + 3)$. To successfully complete it, the user would need knowledge of addition, multiplication, and division. However, prior research usually tests one skill at a time when accounting for several skills. Testing skills individually takes longer than if multiple skills are tested concurrently. Furthermore, there are domains where it is not possible to separate skills and test them individually. For example, swimming might consist of a skill for arm movement, leg movement and taking breaths, but these are challenging to test completely independently. Prior work on tasks containing multiple skills [265, 102] did not include action selections policies to select what the best task is to present to the user, and usually present tasks to the user at random.

Selecting the correct action when multiple skills are present is a hard problem for several reasons. One (in)correct observation alone is not sufficient to determine mastery as there is the chance that the participant has slipped or guessed during the task. Action selection when the true state (in this case which skills are mastered) is unknown is usually solved using a Partially Observable Markov Decision Process (POMDP) [16]. However, the number of states is exponential in the number of skills and a POMDP is exponential in the number of states, making it computationally intractable.

In this chapter we present a modified version of POMDPs, allowing action selection on tasks with multiple skills to be done online. Our system selects the best action, which is the one that minimizes the uncertainty of the user's knowledge state, that is, the task that will give the system the most information gain of the user's skill capabilities. The model is also extended to not only test skills but to allow for users to learn throughout the interaction, enabling teaching to occur.

During each time-step, a task is selected for the user. After observing the user complete the task, the system updates each skill's probability of mastery. Using the updated skill levels, the system proceeds to select a new task. In our system, the likelihood of skills being mastered or not is updated using Bayesian Knowledge Tracing (BKT) [61]. Actions are selected using a modified POMDP. We call our system Bayesian Knowledge Tracing - Partially Observable Markov Decision Process (BKT-POMDP).

To validate BKT-POMDP, we compare it against three other action selection policies: a random policy, a hand-crafted policy, and an optimal policy. We perform three sets of experiments. The first was done in simulation, where we randomly generate tasks, skills, and users. The second was a human-subjects experiment where participants complete an electronic circuit building task. In the third experiment, BKT-POMDP accounts for learning throughout a simulated interaction. In all three experiments, BKT-POMDP learned the user's state faster and more accurately than the random policy and the hand-crafted policy, and it performed comparably to the optimal policy in terms of accuracy and speed. Therefore, we show that BKT-POMDP is a suitable action selection mechanism to create a model of a user's capabilities across multiple skills.

4.2 Formalisms

In this section we will first present the Markov Decision Processes (MDP) and Partially Observable Markov Decision (POMDP) formalisms. In sequence, we present our model which is based on BKT (formalism presented in Chapter 3) and POMDPs.

4.2.1 Markov Decision Processes

A Markov Decision Process (MDP) is a discrete-time stochastic process used for decision making given a probability distribution of states. At each time-step, the system uses the probability distribution of different states to decide what an optimal action is, given the rewards associated with state-action pairings. An MDP is a 5-tuple (S, A, T_a, R_a, γ), where

- **S** - is the state space of the system.
- **A** - is the action space of the system.
- **T(s,s')** : $P(S_{t+1} = s'|A_t = a, S_t = s)$ - is the transition function. Given the current state and the action take, it calculates the resulting distribution of states.
- **R(s,s')** : $E(R_{t+1} = r|A_t = a, S_t = s)$ - is the expected reward given the state distribution and the action.
- $\gamma \in [0, 1]$ - discount factor.

The policy function denoted by π, maps the state space to an action. Optimal actions are not only dependant on the immediate reward, but over rewards that are over a potentially infinite horizon. Therefore, the optimal action usually considers the expected reward by summing over discounted future rewards:

$$E\left[\sum_{t=0}^{\infty}\gamma^t R_{a_t}(s_t, s_{t+1})\right] \tag{4.1}$$

There have been multiple proposed ways of solving and MDP. One common solution is to use *Bellman Equations* [29], which use a value function to calculate the optimal solution for a particular state.

$$V^*(s) := \max_a \left\{ R_a(s, s') + \sum_{s' \in S} T_a(s, s') \cdot \gamma \cdot V^*(s') \right\} \tag{4.2}$$

Partially Observable Decision Processes

Markov Decision Processes assume that the state of the system is known at all times. However, it is not always possible to recover the exact state of the system. An extension to MDPs called Partially Observable Markov Decision Process (POMDP) has been proposed where the state of the system is unknown [123]. Different sensor outputs provide observations about the state of the system. A POMDP is represented by a 7-tuple (S,A,T,R,γ,Ω ,O), where the first five variables are the same as an MDP. In addition we have:

- Ω - The set of possible observations.
- **O** : $O(o \mid s', a)$ - The probability distribution of getting an observation o given the state s' and the action taken a.

A POMDP maintains a belief distribution b over the possible states. The belief distribution is updated using observations:

$$b'(s') = \eta O(o \mid s', a) \sum_{s \in S} T(s' \mid s, a) b(s) \tag{4.3}$$

where η is the normalization constant ($\eta = 1/\Pr(o \mid b, a)$). The probability of an observation given the prior belief and the action is calculated by:

$$\Pr(o \mid b, a) = \sum_{s' \in S} O(o \mid s', a) \sum_{s \in S} T(s' \mid s, a) b(s). \tag{4.4}$$

The optimal policy of a POMDP can also be calculated by Bellman's Equations:

$$V^*(b) = \max_{a \in A} \left\{ \sum_{s \in S} b(s) R_a(s, s') + \gamma \sum_{o \in \Omega} \Pr(o \mid b, a) V^*(\tau(b, a, o)) \right\} \tag{4.5}$$

where $\tau(b, a, o) = \sum_{o \in \Omega} \Pr(b'|b, a, o) \Pr(o|a, b)$. POMDPs are in general considered intractable to solve optimally and therefore approximate solutions are frequently used.

4.3 Bayesian Knowledge Tracing - Partially Observable Decision Process (BKT-POMDP)

We describe here a system that selects optimal actions when creating a model of user capabilities across multiple skills. Our model makes several assumptions: 1) The user skill state is constant, and they will not learn during the interaction. We later present an extension to the model that allows for learning. 2) Each skill's importance is equal; however, this can be changed easily if an application requires it. 3) One task is given to the user at each time-step (differently from the previous chapter, here, we consider a time-step a more extended period where the user has time to complete the task), and the task can contain one or multiple skills. 4) Lastly, skills are independent of each other; that is, one skill's mastery is independent of another's skill mastery.

This system draws inspiration from Partially Observable Markov Decision Processes (POMDPs) [123] and belief state MDPs [171] in that the system does not have full knowledge of the state S, and uses observations o to create an estimate b of what the state is. To learn the model, it selects actions a that maximize the expected information gain reward r of the new belief b' compared to the prior belief b. The model can be seen in Figure 4.1.

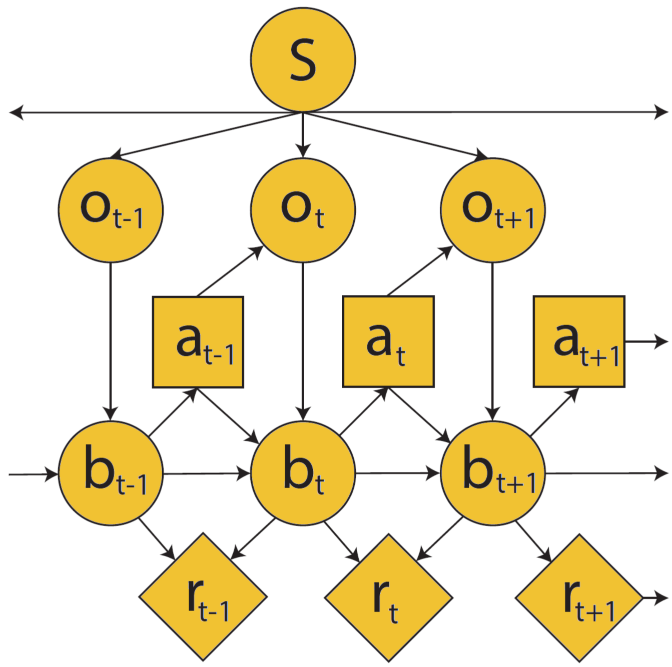


Figure 4.1: The state of the user S (their skill state) is constant throughout the interaction. The system selects tasks a to give the user which generate observations about the user's skills. These observations o are used to update the belief b of the user's skills. The system selects at each time step the task a which it estimates will result in the highest information gain of the user's state, represented by reward r.

This section presents our system called Bayesian Knowledge Training - Partially Observable Markov Decision Process (BKT-POMDP). Similar to the POMDP, our model is composed of the following:

- **S** - The true skill state of the user. A state is represented as a binary vector, with each element i in the vector representing whether skill i is mastered (1) or not (0).

- **A** - The set of actions that can be taken. Each action is a task that can be presented to the user that contains multiple skills. Each action is a vector, with 1s for the skills being tested, and 0s for those that are not.

- **T** : $b' = P(b|o)$ - The transition function updates the belief, given the current belief and the observation. In BKT-POMDP, the transition will be updated using the Bayesian Knowledge Tracing formulation.

- **R** - The reward function. In traditional POMDPs, the reward is a function of either the current state or of the current state plus action. However, our reward is a function of the current belief and the previous belief. Our reward function maximizes the information gain of the user's state at each time-step.

- $\gamma \in [0, 1]$ - discount factor.

- Ω - The set of possible observations. An observation will be a vector of 0s, 1s, and 2s, where 0 represents the wrong answer for that particular skill, 1 represents the right answer, and 2 represents a skill not being tested during that time-step.

- **O** : $P(o|b, a)$ - Observation probabilities. The probability of an observation given the current belief distribution and the action chosen. The observation probabilities are based on Bayesian Knowledge Tracing.

4.3.1 Skill Belief Vector

The skill belief vector is represented by b. This is the current estimate the system has of S. Each element in the vector represents the estimated probability of skill i being mastered.

Even though the number of possible states is exponential in the number of skills tested, it can be represented as a belief vector with a belief value for each skill. For example, if the belief for skill i is currently 0.95, that means that it is very likely that the user has mastered that skill. If the value is 0.3, it is more likely that they do not know that skill, but the system is not certain of this. The skill belief vector is initialized to 0.5 for all skills, representing complete uncertainty at the start of the interaction. In our formulation of the POMDP, all computations can be done on the belief vector rather than over all the possible states. This makes BKT-POMDP much faster to solve than traditional POMDP, as POMDP computes over all the possible skill states ($2^{|S|}$ different possible states), and our calculations are done on just the belief vector.

4.3.2 BKT-POMDP Action Selection

The optimal value function of the POMDP will select the action (the task), which has the highest expected reduction in uncertainty of the user's skill state (Equation 4.6). It iterates over all possible actions (in this case, the possible combinations of skills to test) and selects the one which it expects to have the highest Q value. The Q value (Equation 4.7) is the expected reward when taking a specific action. Upon selecting an action, it will consider all the possible observations and calculate the resulting belief from that observation $b' = T(b, o)$. It will calculate the likelihood of the observation multiplied with the observation's reward. The reward is calculated by the expected increase of certainty of the user's skill state after taking an action.

$$V^*(b) = \max_{a \in A}(Q^*(b, a)) \tag{4.6}$$

$$Q^*(b, a) = \sum_{o \in O} [P(o|b, a) \cdot R(b, b')] \tag{4.7}$$

4.3.3 Belief Update

Each of the tested skills in the belief vector is updated independently using the BKT framework [267]. In BKT, the probability of knowing a skill is dependant on whether the observation was incorrect ($o_i = 0$) or correct ($o_i = 1$), and also on the probability of guessing ($P(G_i)$) or slipping ($P(Sl_i)$) for that skill. When the skill is not being tested ($o_i = 2$), that particular skill's belief value remains the same. Equation 4.8 shows the belief update.

$$
b'_i = \begin{cases} \dfrac{b_i \cdot p(Sl_i)}{b_i \cdot p(Sl_i) + (1 - b_i) \cdot (1 - p(G_i))}, & \text{if } o_i = 0 \\ \\ \dfrac{b_i \cdot (1 - p(Sl_i))}{b_i \cdot (1 - p(Sl_i)) + (1 - b_i) \cdot p(G_i)}, & \text{if } o_i = 1 \\ \\ b_i, & \text{if } o_i = 2 \end{cases} \tag{4.8}
$$

4.3.4 Reward Function

In the traditional POMDP model, the reward usually is related to the specific state. Conversely, in the BKT-POMDP, the reward relates to how much the certainty of the skill state has increased compared to the previous time step. That is, the more certain the system is of the user's skill compared to the previous time step, the higher the reward will be. We use Kullback-Leibler divergence (KLD) [145] to calculate the information gain of the new belief compared to the previous belief (Equation 4.9). KLD is first calculated for both the old belief and the new belief compared to the belief vector of complete uncertainty (U), where $U = [0.5, 0.5, ...0.5]$. The reward is how much information is gained with the new belief compared to the old belief (Equation 4.10).

$$
D_{KL}(b \parallel U) = \sum_i b_i \ln\left(\frac{b_i}{0.5}\right) + (1 - b_i) \ln\left(\frac{(1 - b_i)}{0.5}\right) \tag{4.9}
$$

$$
R(b, b') = D_{KL}(b' \parallel U) - D_{KL}(b \parallel U) \tag{4.10}
$$

4.3.5 Observation Function

The probability of a specific observation is the product of all of the individual skill observations that were tested during that round ($a_i = 1$) given the current belief state

(Equation 4.11). The probability of observing the incorrect answer ($o_i = 0$) is the probability that the user possessed the skill but slipped plus the probability that they did not possess the skill and did not guess correctly. The probability of observing the correct answer ($o_i = 1$), is the likelihood that they possessed the skill and did not slip plus the probability they did not possess the skill but guessed correctly. When the skill was not tested ($a_i = 0$), it did not influence the observation's probability. The observation's update function can be seen in Equation 4.12.

$$P(o|b,a) = \prod_i (p(o_i|b_i,a_i)) \tag{4.11}$$

$$p(o_i|b_i,a_i) = \begin{cases} 1, & \text{if } a_i = 0 \\ b_i \cdot p(Sl_i) + (1-b_i)\cdot(1-p(G_i)), & \text{elif } o_i = 0 \\ b_i \cdot (1-p(Sl_i)) + (1-b_i)\cdot p(G_i), & \text{elif } o_i = 1 \end{cases} \tag{4.12}$$

4.4 Metrics

In this section, we present the baselines and the measures used for evaluating BKT-POMDP.

4.4.1 Baselines

We compare our policy (BKT-POMDP) against three different policies: two baselines (a random policy and a hand-crafted policy) and the optimal policy. We assume there is no repetition of tasks, although the policies could easily be modified to allow it.

- **Random** - A task is selected randomly and presented to the user. The Random policy is the most commonly used action selection mechanism in tutoring systems.

- **Hand-Crafted** - It selects the task with the skills least recently tested. It does so by assigning each skill a counter that is initialized at 0. During each time-step, all non-tested skills' counters are increased by one. If the skill is tested, the counter is reset to zero. This policy will use the unweighted sum of these counters to choose its next action.

- **BKT-POMDP** - Our policy creates a model of the user's skills and chooses tasks that it expects will result in the highest information gain of the user's skill state. This is the policy presented in Section 4.2.

- **Optimal** - This policy selects the optimal action at each time-step. In Experiments 1 and 2, where the goal is skill estimation, it will choose the action that brings the estimate of the user's model b as close to the real model of the user S. In Experiment 3, where the goal is to maximize learning, it will choose the action with the highest expected increase of skills mastered. This policy can select optimal actions as we assume it has full access to S from he start. This assumption does not hold in real scenarios and therefore this policy serves only to illustrate what the optimal policy would be.

4.4.2 Measures

We used the following measures to validate BKT-POMDP. They were calculated each round after the user completed the selected task, and the model's belief was updated.

Distance to True State - How close the current belief b is from the correct skill state S for each user. It is calculated by the difference between b and S. This metric is used in the first two experiments where the goal is correctly estimating the user's true state.

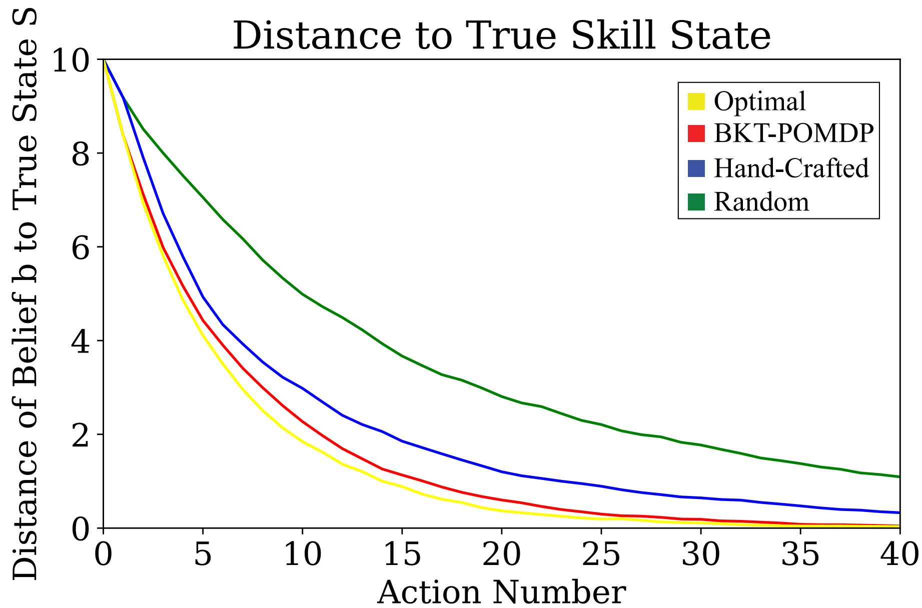


Figure 4.2: Experiment 1 - The average distance of belief b to the true state S for each of the four policies. Overall the optimal and BKT-POMDP policies chose the best actions learning the user skill states the quickest. The third best policy was the hand-crafted policy, and the random policy performed the worst.

$$Dist(b, S) = \sum_i (|b_i - S_i|) \tag{4.13}$$

User Mastery - The number of skills that are mastered. This metric is used in the third experiment, where the goal is teaching all the skills to the user.

$$Mast(b) = \sum_i (S_i) \tag{4.14}$$

4.5 Experiment 1 - Skill Estimation in Simulation

We ran a total of 100 rounds of simulations, where different simulated skills, tasks and users were generated. In each round a different user was generated, and they completed 40 different tasks for each of the four policies.

Skills - 20 different skills were created each round. Each skill had associated with it a probability of guessing it correctly and also a probability of slipping while doing it. The probability of guessing and slipping was randomly chosen from a uniform distribution between 0 and 0.3.

Tasks - 200 different tasks were created. Each one had randomly assigned to it between 1 and 5 skills. During each time-step, a task was selected until a total of 40 different tasks were chosen for that round.

User - During each round, a simulated user was generated. For each skill, they were randomly assigned as mastered or not with equal probability. Each user was associated with an observation for each task they would complete. The observation was created using the probability distribution of guessing or slipping depending on whether they were assigned as having mastered that skill.

4.5.1 Results

We measured the accuracy of the belief state compared to true state using Equation 4.13. All four action selection mechanisms learned the user's skills accurately over time. However, the random policy took significantly longer to approach the true user state. The hand-crafted policy performed better than the random policy. BKT-POMDP performed almost as well as the optimal policy. These results can be seen in Figure 4.2.

During three different points (after 10 tasks, after 20 tasks, and after 30 tasks) we compare whether the accuracy of the policies were statistically significant from each other using an ANOVA with Bonferroni Corrections. In all three cases, the optimal and the BKT-POMDP policies performed statistically significantly better than the hand-crafted and the random policies, and the hand-crafted solution performed statistically significantly better than the random policy. The optimal and the BKT-POMDP policies did not significantly differ from each other.

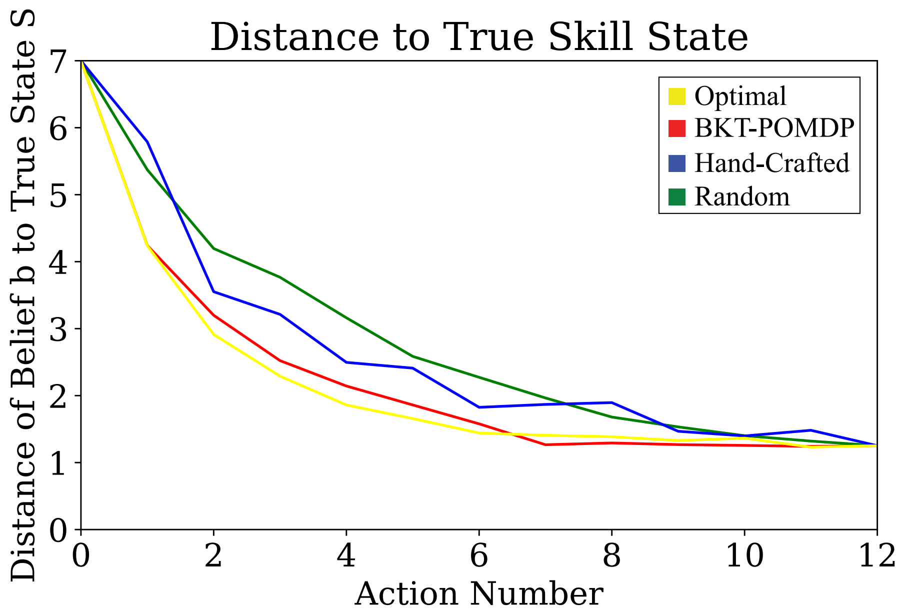


Figure 4.3: Experiment 2 - The average distance of belief b to the correct state S for each of the 12 task actions. The optimal policy performed the best, closely followed by the BKT-POMDP policy. The hand-crafted policy was the third best policy and the random policy was last.

4.6 Experiment 2 - Skill Estimation with Human Participants

We compare BKT-POMDP on a real task with participants completing electronic circuit tasks [78], using enlarged electronic pieces including wires, resistors, and switches. The pieces can be snapped together on a board to form circuits. We chose circuits because they require the user to be proficient in a variety of skills, many of the skills are order independent, and there are several possible assemblies.

Skills - There were six different pieces being tested: a switch, a resistor, an LED, a music circuit, a speaker, and a photo-resistor. There were three different types of skills necessary for accurately completing the tasks: placing the correct piece on the board, placing the piece in the correct location, and placing the piece in the correct orientation. Placement of pieces was dependent on choosing the correct piece. The orientation of pieces was dependent on the participant having chosen the correct piece and placing it in the correct location. Therefore if the participant did not choose the correct piece, then we also defined that they were incorrect in the placement of the piece and the orientation of the piece (this slightly breaks our independence assumption, but does not change the computational cost of the algorithm). Only the LED and the music circuit were directional. Therefore there were a total of 14 skills (six pieces chosen + six pieces placed + two pieces orientation) being tested. We consider a participant to have mastery of a skill if they apply it correctly at least 70% of the time (most skills were tested on average five times, so this allowed at least one slip or guess). The guess and slip probabilities for each particular skill were determined by the number of times participants did not have mastery and guessed correctly and when they did have mastery and slipped in our experiment. The average probability of guessing was 0.28, and the average probability of slipping was 0.10.

Tasks - There were 12 different tasks for the user to complete. Each task required a combination of different skills. A board was given to the participant with wires and a battery piece (without batteries inside) that were already placed. The participant was then asked to complete a task. For example, there was a task where the user was asked to create a circuit with a light that could be turned off and on. Therefore they needed to choose the correct pieces: an LED, a resistor, and a switch; place each in the correct location; and place the LED with the correct orientation. In addition to the six different pieces that were being tested, we gave the user four additional distractor pieces (making guessing correctly less likely).

Users - 23 participants completed the 12 circuit tasks, of which 14 were male and 9 were female. The study was approved by the university's Institutional Review Board and participants signed a consent form agreeing to participate. They were not provided with any information on how electronic circuits worked, other than the piece's name and the ports on the pieces. We also assumed that no learning happened throughout the experiment, as no help or feedback was provided. The participants' expertise on circuits was varied, with some participants having mastery of none of the skills, and some having full mastery. All 23 participants' data was used for the four policies by simulating which task the system would have chosen during each time-step for each participant.

4.6.1 Results

We annotated for every participant whether they had mastery over each skill (they were considered to have mastery if they got the skill right over 70% of the time). For every participant, observations were created by annotating whether they demonstrated the skills successfully in each task. On average, participants were able to choose the right pieces 77.39% ($SD = 14.59\%$) of the time. Participants placed the piece in one of the correct locations 38.75% ($SD = 33.91\%$) of the time. And participants placed the directional pieces in the correct orientation 35.36% ($SD = 32.43\%$) of the time.

During the last few rounds all four policies had high certainty on the user's skills. Therefore we compare rounds 3, 5 and 7 for statistical significance using ANOVAs and Bonferroni Corrections, measuring the distance of the belief compared to the true state (Equation 6). After taking three actions, the optimal policy performed significantly better than the hand-crafted and random policies. BKT-POMDP performed significantly better than the random policy. The other comparisons were not significant. After five rounds, the optimal policy performed significantly better than the random policy. The other comparisons were not significant. There were no significant differences after seven rounds.

4.7 Experiment 3 - Learning in Simulation

There are many situations, especially in ITS, where we do not only want to create a model of skills, but also select the tasks which will teach the most. We extend BKT-POMDP by changing the reward function and the belief update function to account for learning. These modification allow BKT-POMDP to select the task with the skills that it estimates will bring all the skills' mastery's closest to 1.

4.7.1 Reward Function for Teaching

The reward function is replaced with Equation 4.15. It rewards increases in the belief of the skills. Therefore, it rewards the user having higher mastery over the skills. At the start of the interaction the skill belief vector is set to low probability of mastery (0.05) for all the skills.

$$R(b, b') = \sum_i [(b'[i] - b[i])] \, if (b'[i] > b[i]) \tag{4.15}$$

4.7.2 Belief Update

The belief update still follows Equation 4.8, however it includes the learning update from BKT [267]. Each time the participant practices a skill, they have a chance of learning it represented by $P(L_i)$.

$$P(b'_i) = P(b_i|o_i) + (1 - P(b_i|o_i)) \cdot P(L_i) \tag{4.16}$$

Rounds

100 rounds of simulation were run during which 40 different tasks were chosen using the different policies. During each round the following were generated:

Skills - 20 different skills were generated. The probability of guessing and slipping

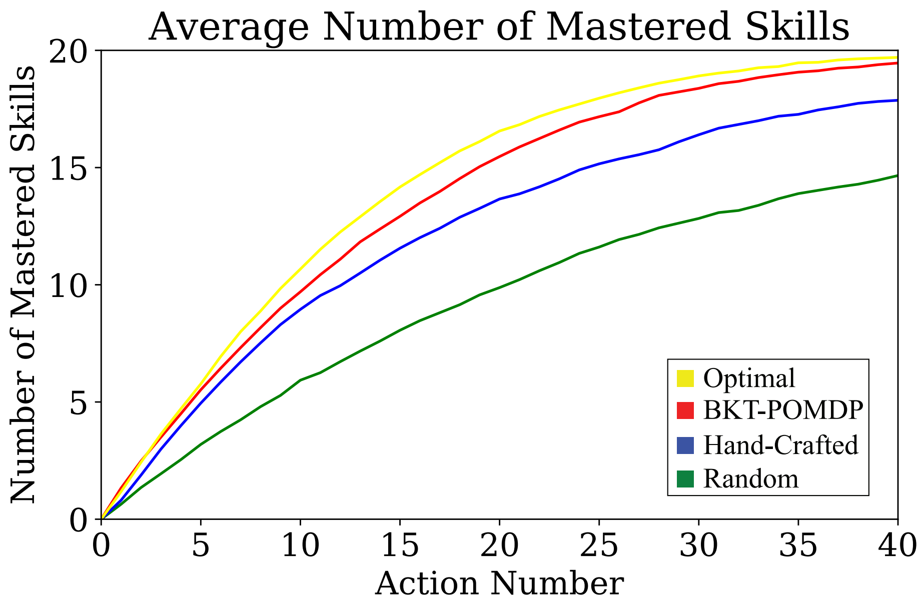


Figure 4.4: Experiment 3 - The graph shows the number of mastered skills. The BKT-POMDP and the optimal policies selected tasks that brought the user skill closer to mastery of all skills quicker than the hand-crafted and the random policies.

was randomly chosen from a uniform distribution between 0 and 0.3. Additionally the probability of learning ($P(L_i)$) was generated from a uniform distribution between 0.15 and 0.3.

Tasks - 200 tasks were generated, each with between one and five skills.

User - In each round a user was generated. All skills for the user were set as not mastered. After each task, each non-mastered skill in the chosen task was updated by having a $P(L_i)$ chance of the user having learned it.

4.7.3 Results

We tested which of the policies selected tasks that increased the knowledge of the participants the fastest. We did this by measure the number of skills with mastery using Equation 4.14. The results are shown in Figure 4.4, which shows that the optimal condition selected the best tasks to teach closely followed by BKT-POMDP. The hand-crafted condition performed third, and random performed the worst.

During three different points (after 10 tasks, after 20 tasks, and after 30 tasks) we compare whether the conditions were statistically significant from each other using an ANOVA with Bonferroni Corrections. All six pairwise comparisons were statistically significant from each other, except the BKT-POMDP and the hand-crafted policies after 10 rounds. This indicates that the optimal policy performed the best, followed by BKT-POMDP, hand-crafted and random.

4.8 Discussion

In the first set of experiments, BKT-POMDP and the optimal condition converged on the user's true state after 40 tasks. random and hand-crafted were approaching convergence and would do so with more assigned tasks. This means that the policies were able to correctly learn a model of the user's skills. However, BKT-POMDP did so much faster than the other algorithms, and almost performed as well as the optimal policy. As the optimal policy is not possible to use in real scenarios (as it requires a perfect model of the user), BKT-POMDP is a good policy to model a user's skills.

In the circuit experiment, BKT-POMDP and the optimal policy also outperformed the other baselines. This experiment shows that BKT-POMDP translates well to real world applications. Unfortunately none of the models completely converged in 12 rounds, due to the low number of rounds and the high guess rate for some of the skills.

In the third experiment, we show that BKT-POMDP can easily be modified to allow for different goals. We show that modifying the reward function accounts for user learning. Instead of maximizing student skill, it now maximizes the expected amount of learning the user will have over all skills. In the experiments, BKT-POMDP outperforms the other baselines and performs on par with the optimal model.

4.9 Summary

An intelligent tutoring system selects tasks to give to a student to test and teach them. If a system selects tasks to effectively create an accurate model of a user's skills, it can focus teaching on skills the user has not yet mastered. In this chapter, we presented BKT-POMDP, a system that can select which task to give to a user in order to maximize the system's information gain about the user's skills. It advances prior algorithms in that it can choose tasks where there are multiple skills presented. We show that BKT-POMDP can select tasks such that it creates a model more quickly and accurately than baselines. In an electronic circuit building task, we demonstrate that BKT-POMDP can choose tasks with actual participants during a user study.

BKT-POMDP is a flexible system that can be used for several different applications, and where individual parts can be changed to suit each application. In ITS, the main goal is to teach skills the student does not have mastery over. BKT-POMDP can quickly and accurately create a model of a student's capabilities, so that the ITS can focus on teaching the non-mastered skills. Modifying the reward function allowed BKT-POMDP to not only create a model of user skills but also to account for learning. It selected the tasks that would teach the user the most, and bring the user closer to having full mastery of all skills. Therefore, it can be used in intelligent tutoring systems to select which task to teach when there are multiple skills present.

BKT-POMDP could also be used beyond tutoring applications, such as in manufacturing settings. The system could quickly model which skills an employee has and assign tasks within the employee's expertise while also avoiding tasks that they would not be able to do. Additionally, when multiple people are present, the system can assign tasks to each person according to their expertise across all tasks or create teams whose members have an equal balance of skills. In manufacturing settings, some skills are more important than others, given that they appear in many tasks. The user model over these skills could be prioritized by giving different weights to each skill in the reward function instead of having equal value. Therefore, prioritizing the user learning the higher weight skills first.

In this and the previous chapter, we have explored how to create user skill models for complex tasks and, given the estimated user skills, how to select which task to teach optimally. In the following two chapters of the thesis (Chapter 5 and 6), we shift directions and investigate the ways a robot should teach. We study different aspects of tutoring, such as how a robot can positively influence students and what roles a robot should take on when teaching.

Chapter 5

Social Robot Characteristics: Comparing Robot Roles for Adult Tutoring*

Prior work has shown that not only the content influences how much a student learns but also the traits of the agent who is delivering the material. For example, robot embodiment plays a role in the student's learning rate, with research showing that people learn more from a physically co-located robot than the same robot on a screen [158]. Another study shows that participants whose gender differ from the robot had a higher willingness to learn from it [206].

In this chapter we focus on a particular aspect of how a robot should teach a user: what role it should take. Research in human-to-human tutoring and child-robot tutoring show the advantages of an agent interacting as a peer when teaching [22, 194, 50, 268]. Therefore, we explore how a robot should engage with users while tutoring: as a peer or a tutor. We created a controlled study where adults

*Parts of this chapter were originally published as: Nicole Salomons, Kaitlynn Sierra, Adérónké Adéjàre, Brian Scassellati. (2022). "We Make a Great Team!": Adults with Low Prior Domain Knowledge Learn more from a Peer Robot than a Tutor Robot. In *International Conference of Human-Robot Interaction* [220].

interacted with a robot that was either introduced to them as a peer or introduced as a traditional instructor, while they learned electronic circuit skills. The help actions of the robot were consistent across conditions; however, the robots would slightly differ in how they addressed participants. In the peer condition, the robot would refer to the participant as a teammate and use "we/us" utterances. Whereas in the tutor condition, it would refer to the participant as a student and use "you" utterances.

Our results show that participants in both conditions learned during the interaction with the robot. Further data analysis revealed that participants with low prior domain knowledge learned significantly more from the peer robot than from the tutor robot. Furthermore, participants in the peer condition saw the robot as friendlier, more respectful, and more intelligent. This study shows the benefits of introducing the robot as a peer instead of a traditional teacher when tutoring adults.

5.1 Introduction

In peer-to-peer tutoring, children or adults teach each other rather than being taught by a teacher [185]. There are benefits of peer instruction for both the student who is teaching and the student who is learning. Throughout this chapter, we will call a student instructing another student as the *peer-teacher* or *peer*, the student who is being instructed as the *learner*, and a traditional teacher as *tutor*. When a peer-teacher prepares content and teaches a colleague, they demonstrate higher learning gains than when they only learn the content for themselves [22]. Likewise, the learner who is taught by a peer frequently learned more than the one who is taught by a teacher, especially if they had higher prior domain knowledge [149]. Furthermore, peer instruction lowers failing rates [194], creates an increased sense of community [238], and increases student self-esteem [85].

Most work in robot tutoring has focused on having the robot take on the role

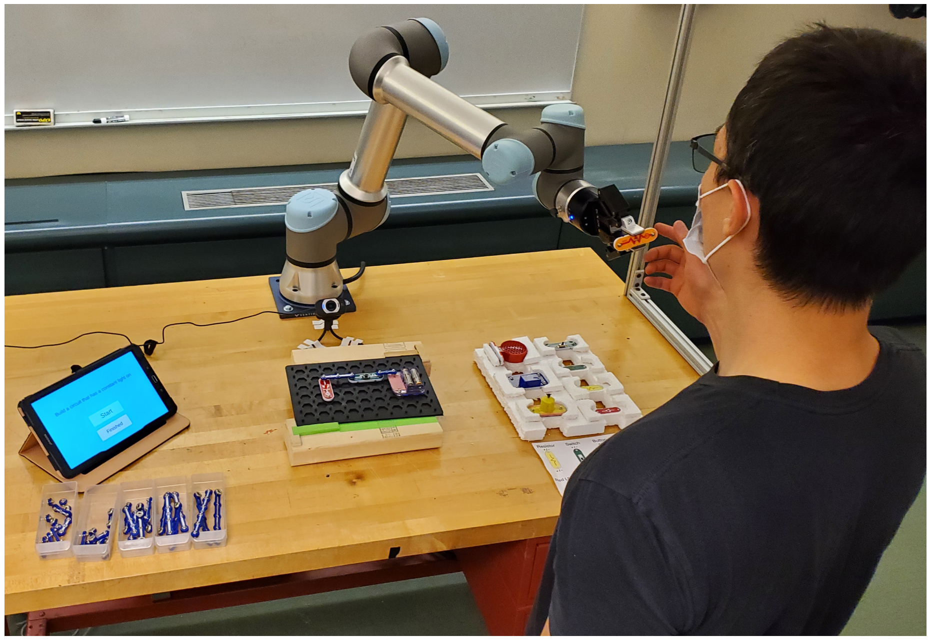


Figure 5.1: Participants built electronic circuits with either a peer robot or a tutor robot. The robot would provide personalized help based on the user's skills. In the figure, we see the robot suggesting the user add a resistor to the board.

of a traditional tutor [31]. While a few studies investigated the robot's role as a peer, these focused on child-robot interactions [50, 268, 188]. Similar to peer tutoring among adults, children learned more [26], became more engaged [268], and developed a stronger growth mindset [188], when interacting with a peer robot compared to a tutor robot.

While children appear to benefit from peer-based child-robot interactions, it is unclear whether the same results will hold for adults. Adults may have higher expectations for a peer that a robot could struggle to meet. Adults may also have more practical experience working with peers and as such might not easily accept a robot peer. Lastly, working with adults will also require working in a more challenging educational domain, with harder and more complex problems to be taught, which may not easily transfer to a robot. To study how peer robots are viewed by adults, we designed a between-participant study where participants interacted with either a peer robot or a tutor robot, during an electronic circuit building task.

Participants built ten electronic circuits with an autonomous robot, as seen in Figure 5.1. The system modeled the user's skills throughout each task using Continuous Bayesian Knowledge Tracing [221]. The robot then provided personalized help to the user depending on their skill state and the skills needed for each task. The robot provided nearly identical advice in both conditions. However, in the peer condition, the advice was delivered using pronouns that indicated that the robot was an equal and invested colleague ("we/us"), and in the other indicated that the robot was a more knowledgeable authority figure ("you"). Participants completed a pre-test and a post-test to detect skills learned. They also completed questionnaires about their perceptions of the robot.

While participants in both conditions demonstrated significantly more skills in the post-test than in the pre-test, there were no significant increases of skills between the conditions. However, when analyzing only participants with low initial pre-test scores, they learned significantly more in the peer condition than in the tutor condition. Furthermore, participants viewed the peer robot as more intelligent, more social, and friendlier than the tutor robot, independent of prior skill knowledge. Additionally, participants who interacted with the peer robot felt more respected by it than the tutor robot.

5.2 Related Work

In this section, we introduce background on peer-to-peer tutoring and show some advantages of this learning strategy. We then review literature on how the peer-to-peer tutoring strategy has been extended to human-robot tutoring.

5.2.1 Peer-to-Peer Tutoring

Peer-to-peer learning is defined as "an educational practice in which students interact with other students to attain educational goals" [185]. It is typically used as a supplement to the classroom learning process between a teacher and students. Peer-to-peer learning is a favorable educational practice because it prepares students for learning from others in workplaces and communities [185].

There are numerous benefits to peer-to-peer learning. Peer-teachers who studied in preparation to teach something and then taught the information generally scored higher on a retention test than students who prepared only for themselves [22, 90, 14]. In reciprocal peer tutoring (RPT), students are paired together to review content and to practice skills [85]. This strategy resulted in greater improvements in cognitive gains, lower levels of subjective distress, and higher course satisfaction [85]. In addition to increasing student achievement, peer-to-peer learning has many social benefits, including positive race relations in desegregated schools, mutual concern among students, and student self-esteem [239]. Interventions were most effective with younger, urban, low-income, and minority students [213].

Peer-to-peer learning is also effective for adult learning. Lasry et al. showed that peer-taught university students had higher learning gains than traditionally taught students [149]. Additionally, they show a significant increase in learning gains for students with high background knowledge but not for students with low background knowledge. Another potential benefit for peer tutoring is that adults learn better in an informal environment and need to be respected when learning new things [58]. Awan [18] commends the use of peer-to-peer learning in radiology residencies because it promotes active and relevant learning. This practice also prepares future physicians for explaining medical topics to their patients [39].

5.2.2 Robotic Peer-to-Peer Tutoring

Social robots have been found to be effective tutors via individualized tutoring interactions [203, 199, 232]. Tutoring robots can take on several different roles, including a learner, a peer, or a teacher [268, 50, 49, 5]. However, approximately 86% of the studies conducted with robots to facilitate human learning consist of the robot taking on the role of tutor [31].

Although the use of robots as peers represents a minority of the literature, there has been some work analyzing peer tutoring robots. A peer-teacher social robot has been shown to positively benefit a child's language learning [139]. A peer robot can also enhance a child's own creative thinking [8]. Zaga et al. showed how children demonstrated increased engagement when playing with a peer robot to complete a Tangram puzzle compared to when playing with a tutor robot [268]. In a long-term study, a peer-teacher humanoid social robot with the ability to personalize its interactions with children in a classroom increased the children's learning of novel topics [26]. Park et al. [188] determined that children who played with a social robotic peer that exhibited a growth mindset (a belief that success arises from effort and perseverance) developed a stronger growth mindset of their own. Chen et al. [49] noted that the children who interacted with their adaptive peer-teacher robot not only had more expressive faces than the children who interacted with their tutor robot, but they also learned more and retained advanced vocabulary.

While robots can be used as a peer-teacher during child learning scenarios, they can also play the role of a tutee or naive peer. In this case, a person takes the role of a peer-teacher and educates their robotic peer, resulting in the enhancement of the person's learning through the reinforcement of concepts. Japanese children at an English language school improved their spontaneous learning of new English vocabulary words after teaching them to a robot [247]. The forms of teaching naturally implemented by the children involved direct teaching, gesturing, and verbal teaching.

Section Part	Tutor Robot	Peer Robot
Introduction to Robot	Experimenter: "*Hello, This is Urie, the robot. Urie will be teaching you about how to build some electronic circuits today.*"	Experimenter: "*Hello, This is Urie, the robot. You and Urie will be collaborating in building some eletronic circuits today.*"
Reinforcement Question	"*When should you add a speaker to the circuit?*"	"*When should we add a speaker to the circuit?*"
Wrong Piece	"*Can you explain to me what a button does? Do you think you need it for the current task?*"	"*Can you explain to me what a button does? Do you think we need it for the current task?*"
Piece Recommendation	"*Here, try to add the resistor to the circuit.*"	"*Here, let's try to add the resistor to the circuit.*"
Help Utterance	"*To power the music circuit you need to make sure that its positive port is connected to the positive port of the battery.*"	"*To power the music circuit we need to make sure that its positive port is connected to the positive port of the battery.*"
Task Finished Correctly	"*Awesome, you are a good student.*"	"*Awesome, we make a good team.*"

Figure 5.2: We present some of the different utterances between conditions. The robot was introduced differently to the participant depending on the condition. The remaining utterances were very similar and often only differed in the pronoun used. Some examples of help actions included asking questions to reinforce a correctly applied skill, pointing out a wrong piece on the board, recommending a piece, and giving a description of an incorrectly applied skill.

Robots have been used in the role of the tutee where the children taught the robot handwriting [116, 118, 47]. The children demonstrated to an autonomous robotic agent how to write certain letters or words, helping develop their own writing ability.

Although these studies focused on interactions with children, a peer robot can nevertheless provide unique learning benefits separate from those of a tutor robot.

5.3 Methodology

Higher learning gains and positive traits have been seen when interacting with a peer-teacher both in human tutoring [22, 149] and in robot-child tutoring [50, 268, 188]. Therefore, we predict that participants interacting with a peer robot will learn more new skills than participants interacting with a tutor robot. Furthermore, an adult peer-teacher was especially beneficial when the adult learner had higher prior knowledge in the domain [149]. Therefore, we also predict that adults with higher knowledge will benefit more from a peer robot.

Research has shown that adults learn better in an informal environment and highlight the importance of feeling respected [58]. We hypothesize that adults interacting with the peer robot will feel treated more as an equal and therefore feel more respected than those interacting with a tutor robot.

Prior work has shown that students who interacted with a peer robot were more engaged [268] and were able to create rapport with it [139]. Therefore, we hypothesize that participants interacting with a peer robot will be more engaged and view the robot more positively than participants with the tutor robot. Lastly, people often have very high expectations out of robots [11, 147, 205]. When the participant is told the robot is a tutor, those expectations might be higher, as the robot is presented as an expert at the task. Therefore, we hypothesize that a peer robot might be seen as more intelligent than a tutor robot as there will be lower expectations put on it.

We have five main hypotheses for this study:

Hypothesis 1a: *Adults in both conditions will show significant improvement in electronic circuit skills from pre-test to post-test.*

Hypothesis 1b: *Adults will learn more from a peer robot than a tutor robot.*

Hypothesis 1c: *Adults with high initial knowledge will especially benefit from a peer robot, compared to adults with high initial knowledge interacting with the tutor robot.*

Hypothesis 2: *Adults will view a peer robot more positively than a tutor robot.*

Hypothesis 3: *Adults will be more engaged with a peer robot than a tutor robot.*

Hypothesis 4: *Adults will feel more respected by a peer robot than a tutor robot.*

Hypothesis 5: *Adults will see a peer robot as more intelligent than a tutor robot.*

To test our hypotheses, we split participants into two conditions: one where they interact with the robot as a tutor, and one where they interact with the robot as a peer. Participants built electronic circuits using a modular circuit-building toy called Snap Circuits [78], initially presented in Chapter 3. We chose circuit design because it is a task that is challenging for most adults; there are varying levels of initial knowledge, with some participants having high initial knowledge and others low initial knowledge on circuits; and a robot can model the task using a sensing system.

5.3.1 Conditions

Participants were split randomly into two conditions:

- **Tutor Robot** - The robot acted as a traditional tutor and provided instructions to the participant. The robot was introduced as a teacher to the participant, and its utterances towards the participant used second person singular pronouns like "you".

- **Peer Robot** - The robot acted like a peer who is working together with the participant on the circuit. The robot was introduced as a collaborator, and its utterances used first-person plural pronouns ("we/us").

The difference between conditions was minimal, especially considering the robot had few anthropomorphic features. In both conditions, the robot's utterances were very similar, mostly changing pronouns from "we" in the peer condition, and "you" in the tutor condition. The robot always presented correct help suggestions independent of condition. Some examples of utterances can be seen in Figure 5.2.

5.3.2 Robot System

Participants interacted with the robot on a large table. Figure 5.3 shows an illustration of the experimental setup. Participants were given each task via a tablet, and on the tablet, they could indicate that they had finished the current task and start the next task. The tablet provided no help with the task. Participants used wires and electronic circuit pieces to build their circuits on a board in the middle of the table. An overhead Kinect Azure camera detected what pieces were on the board and how they were connected. A green hand strip at the bottom of the board was used to detect when the participants' hands were on top of the board, and therefore the camera's observations would be inaccurate. A second camera faced the participant to record the interaction.

A UR5e robot from Universal Robots was used in this study. It is a lightweight industrial robotic arm with 6-DOF. It could pick up the snap circuit pieces with its gripper and hand them to the participant. The robot was able to communicate to the participant via a text-to-speech voice. Additionally, the robot displayed idling behavior with random movements every few seconds, occasionally looking at the circuit board, pieces, or the participant, by pointing the gripper at it. The robot acted completely autonomously throughout the study.

5.3.3 Snap Circuits Tasks and Skills

We created 32 different electronic circuit tasks of varying difficulty, of which participants completed ten. There were more tasks than the number the participant completed, so the robot could adjust to each person's skill level. Section 5.3.6 describes how tasks were chosen for each participant. For each task, the participant was given an empty circuit board with only a battery on it, many wires of different sizes, and seven pieces: an LED, a switch, a button, a motor, a resistor, a music circuit, and a speaker. Each piece could be snapped together on the board to form circuits. An example of a completed circuit can be seen in Figure 5.4.

The participant was instructed what task to complete next via a tablet. They were given three minutes for each task unless they correctly completed it before the time expired. Some examples of tasks are: "Build a circuit that plays music when a

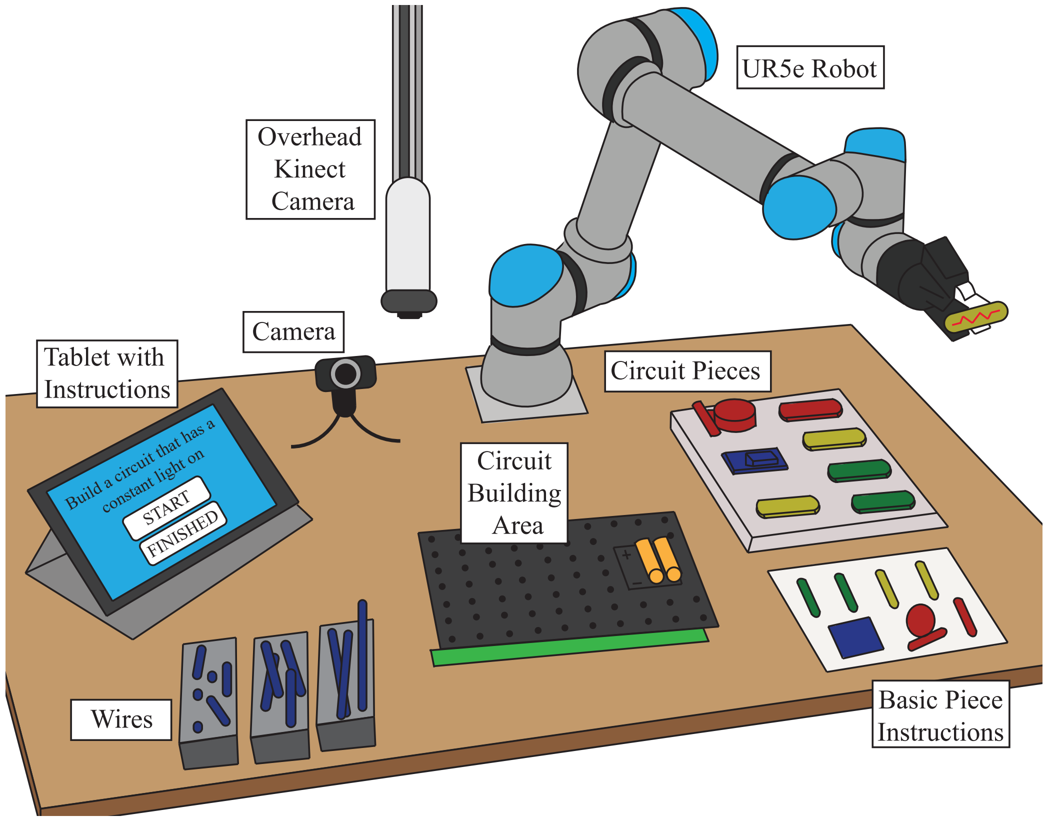


Figure 5.3: The experimental setup. Participants were given tasks via a tablet application. In the middle of the table, they built circuits using wires and circuit pieces. They were provided basic instructions with the piece names. An overhead camera focused on the circuit and modeled which skills were correctly applied. The camera also detected whether a user was working on the circuit by seeing whether a green bar was occluded below the circuit building area. A UR5e robot provided them with help every 30 seconds based on what was needed for the current task. An additional camera collected video and audio data from the participant.

switch is turned on" and "Build a circuit that spins a motor when a switch is turned on or a button is pressed".

Each task required the participant to demonstrate different skills. Some examples of these skills are: adding a speaker when it is needed, creating a closed circuit, knowing the directionality of an LED, powering the music circuit, and creating AND and OR gates. A total of 17 skills were tested. Section 5.3.5 explains how we model participants' skills throughout the tasks.

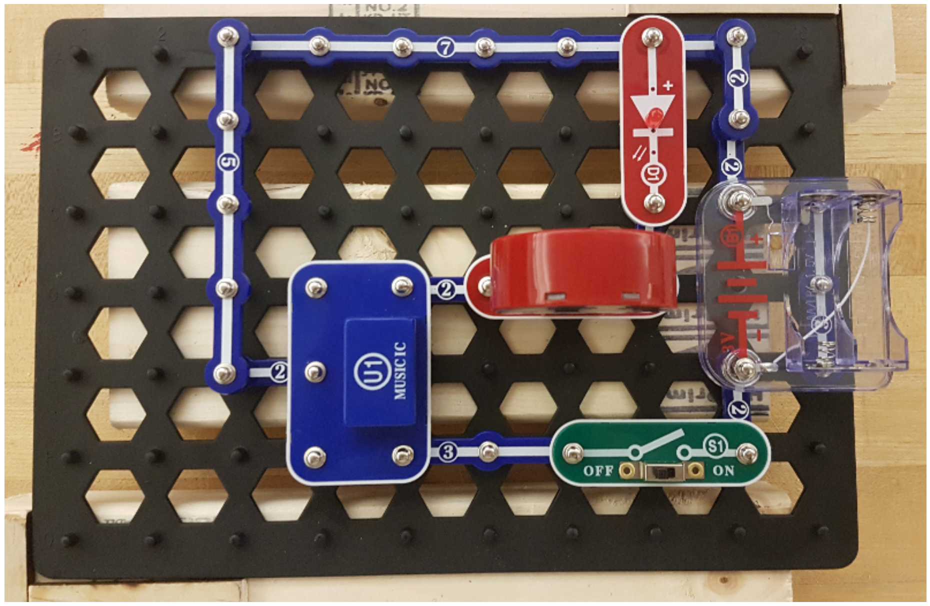


Figure 5.4: An example of a completed circuit. This circuit plays music and blinks a light in the rhythm of the music when the switch is turned on.

5.3.4 Experimental Procedure

Each session took approximately 60 minutes. The procedure of the experiment consisted of the following:

1. The participant completed the consent form and a demographic questionnaire.
2. The participant completed six pre-test electronic circuit tasks and questions. These are detailed in Section 5.3.8.
3. The experimenter introduced the robot. Depending on the condition, the robot was introduced as either a peer/collaborator or as a teacher.
4. The participant built ten electronic circuits alongside the robot. The robot provided personalized help actions.
5. The participant answered post-study questionnaires.
6. The participant did six post-test electronic circuit tasks.

7. At the end of the interaction, participants were debriefed and paid \$10 for their time.

5.3.5 Skill Estimation

A computer vision system with an overhead camera observed the user as they placed pieces on the board. It tracked which pieces were on the board, and which pieces were connected to each other. User skills were modeled using an extension of Bayesian Knowledge Tracing called Continuous Bayesian Knowledge Tracing (C-BKT) [221]. C-BKT was used as it allows skill modeling of complex tasks where the observations are noisy, and skills vary in the amount of time needed to demonstrate them. The system individually modeled each of the 17 skills by creating an estimate of whether the user had mastered each skill. We represented this estimate as a vector b, where each skill was initialized to 0.5, representing complete uncertainty of the user's skill state. Each second, b was updated using observations from the computer vision system detailing which skills were applied correctly and which ones were not.

5.3.6 Task Selection

Prior work shows that selecting tasks with appropriate difficulty leads to higher learning gains [65, 219]. Therefore tasks were chosen for each participant according to their demonstrated capabilities. To rate the difficulty of each task, each of the 17 skills was given a difficulty rating from a scale of 1.0 to 5.0, with 5.0 being the most difficult. These were determined by consulting an electrical engineering major. The ratings were stored in a difficulty vector d. For example, the skill for whether a participant knew when to use an LED was given a difficulty rating of 1, while the skill for whether the participant knew how to create an OR gate was given a 4.5. The current belief estimate b was used to select the next task.

In order to determine which task to give next to a participant, all remaining tasks are assigned a difficulty rating R based on the skills Sk that a task t incorporated. The rating was calculated based on the difficulty of each skill and the participant's current belief value b. Participants with higher belief values would likely find the task easier. Therefore, we used $1 - b(i)$ to measure how difficult the task would be for the participant. As we are summing over the difficulty of each skill for a task, the more skills a task tests, the more difficult it will likely be. The difficulty rating R for a specific task is calculated as follows:

$$R_t = \sum_{i \in Sk} (1 - b(i)) * d(i) \tag{5.1}$$

There is also a fixed ideal rating value V that was set equal to five after initial trial and error. The V is intended to help ensure that an appropriate task is selected next for the respective participant so that the task is not too easy nor too overwhelming [174]. The task whose r value is closest to V is selected as the next task and removed from the possible remaining tasks for the next iteration.

$$NextTask = \min_{t \in T}(|R_t - V|) \tag{5.2}$$

In the case where several tasks are equally close to V, one of these potential tasks is selected at random. The process is repeated until the interaction with the participant ends.

5.3.7 Help Action Selection

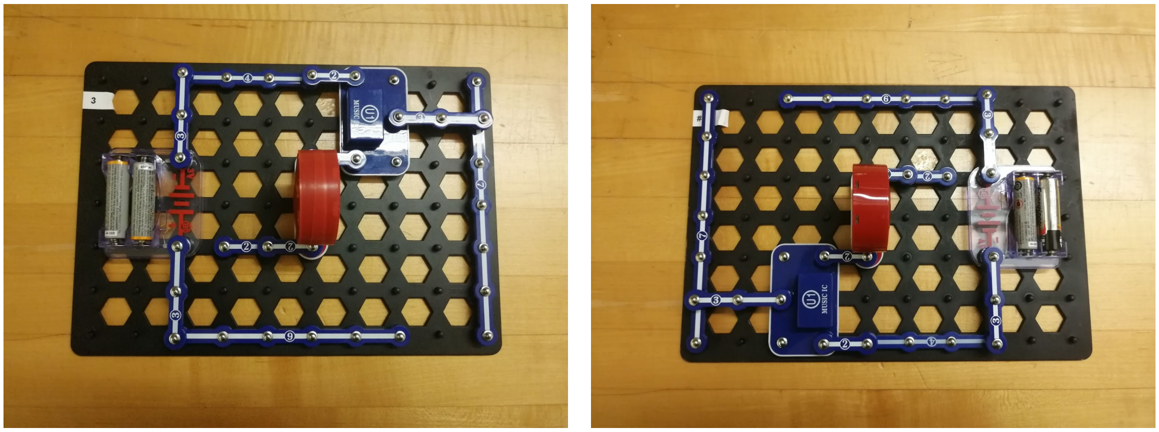
Personalizing help in tutoring systems leads to higher learning gains [203, 56]. Therefore the robot provides assistance to the participant according to the skills they had demonstrated during the current task. The robot provided a help action every 30 seconds. There were six different types of help actions, of which the system selected one at random. The different kinds of help actions were:

- **Reinforcement Question** - The robot asked a question about a skill the participant had demonstrated.
- **Reinforcement Utterance** - The robot confirms that a skill the participant had demonstrated is correct.
- **Wrong Piece Point** - The robot pointed to a piece on the board and said it was not needed.
- **Wrong Piece Utterance** - The robot said that one of the pieces on the board was not needed.
- **Help Movement** - The robot gave help to the participant by explaining something about a skill they had not demonstrated. While explaining it, the robot pointed to something on the board or handed the participant a piece.
- **Help Utterance** - The robot gave help to the participant by explaining a skill the participant had not demonstrated. The robot did not move in this case.

The robot did not select a reinforcement help action if the participant did not have any correct skills displayed during the task. Likewise, it would also remove wrong piece help actions from the randomization options if all the current pieces on the board were needed. Additionally, when the participant pressed the finished button on the tablet, but the task was incorrect or incomplete, the robot randomly selected either a *help movement* or a *help utterance* for one of the skills that were demonstrated incorrectly. Examples of different types of robot help actions can be seen in Figure 5.2.

5.3.8 Metrics

We had three different types of metrics: test metrics, behavioral metrics, and survey metrics.



(a) Pre-Test **(b)** Post-Test

Figure 5.5: The pre-test and the post-test were identical except that boards were rotated 180 degrees. In this case participants were asked to complete the circuit such that it would play music when a button was pressed.

Test Metrics

- **Pre-test and Post-test** - To test how much people have learned, we conducted a pre-test and a post-test [73]. Our pre-test and post-test were composed of six very similar questions. The first two questions on both tests were the same. They asked participants to build from scratch a circuit that shines a constant light and a circuit that plays music, respectively. Participants were given five minutes to do both tasks. The third and fourth tasks on both tests required participants to add pieces to the board to complete the circuits. These tasks were identical between pre-test and post-test, other than the circuit boards being rotated 180 degrees to the participant in the post-test (the two versions of the third task are shown in Figure 5.5. For the fifth and sixth tasks, we presented pictures of pre-built circuits and asked participants to write down what the circuits did. These were similar between pre-test and post-test, but the pieces were arranged differently on the board. Participants were given five minutes to complete tasks three through six.

  We classify participants into either having high prior circuit knowledge or low prior circuit knowledge based on their pre-test. If participants got less than half of the skills correctly on the pre-test, they were considered to have low prior circuit knowledge. Otherwise, they were considered to have high prior circuit knowledge.

Behavioral Metrics

- **Speaking** - We measured how much participants talked to the robot. Participants were classified as engaging in conversation with the robot if they said at least ten sentences to it, and as not engaging in conversation if they talked to it less.

Survey Metrics

- **Demographics** - Before the interaction, we administered a demographics questionnaire that asked participants questions about their gender, age, occupation or major (if student), and country of origin. We also asked how often they used a computer, their familiarity with robots, and their level of expertise on electrical circuits.

- **Post Interaction Questionnaire** - We administered the RoSAS questionnaire about their feelings towards robots [42]. The RoSAS measured participants' perceptions of the robot's warmth, competence, and discomfort. We also asked the participant to rate the following on a 1-7 Likert Scale with 1 being "Not Applicable" and 7 being "Most Applicable": The robot acted like my colleague; The robot treated me like an equal; I felt like I was being judged by the robot; I felt like the robot respected my capabilities; The robot was friendly; I felt engaged while interacting with the robot; I felt like the robot was boring; I felt like the robot was smart; I felt like the robot was good at electronic circuits; The robot was better than me at electronic circuits. Finally, we had an open-ended question for participants: Is there anything you wished the robot would have done differently?

5.3.9 Participants

There were 37 participants who completed the experiment. Interactions with the robot lasted on average 37 min and 37s (SD = 5 min and 6s). The university's Institutional Review Board approved the study, and participants signed a consent form agreeing to participate. There were nine male and eight female participants in the peer condition, and their average age was 28.00 (SD = 12.90). There were nine male, ten female, and one non-binary participants in the tutor condition, and their average age was 25.00 (SD = 10.60). Participants in the peer condition rated themselves as an average of 2.47 (SD = 1.23) on a 1-5 scale on their prior electronic circuit knowledge, whereas participants in the tutor condition rated themselves an average of 2.40 (SD = 1.60). There were no significant differences in gender, age, or prior circuit expertise between conditions.

5.4 Results

5.4.1 Manipulation Check

First, we check whether the peer robot and the tutor robot were perceived differently. We asked each participant on a scale of 1-7 whether they perceived the robot as a peer and whether they felt like they were treated as an equal. Participants in the peer robot condition perceived it significantly more as a peer ($M = 4.59, SD = 2.18$), than participants who interacted with the tutor robot ($M = 2.65, SD = 1.39$), $t(37) = 3.27, p = 0.002$. And participants in the peer condition ($M = 5.00, SD = 1.83$) perceived the robot as treating them as an equal significantly more than participants in the tutor condition ($M = 3.58, SD = 2.09$), $t(35) = 4.50, p = .041$. Therefore, we

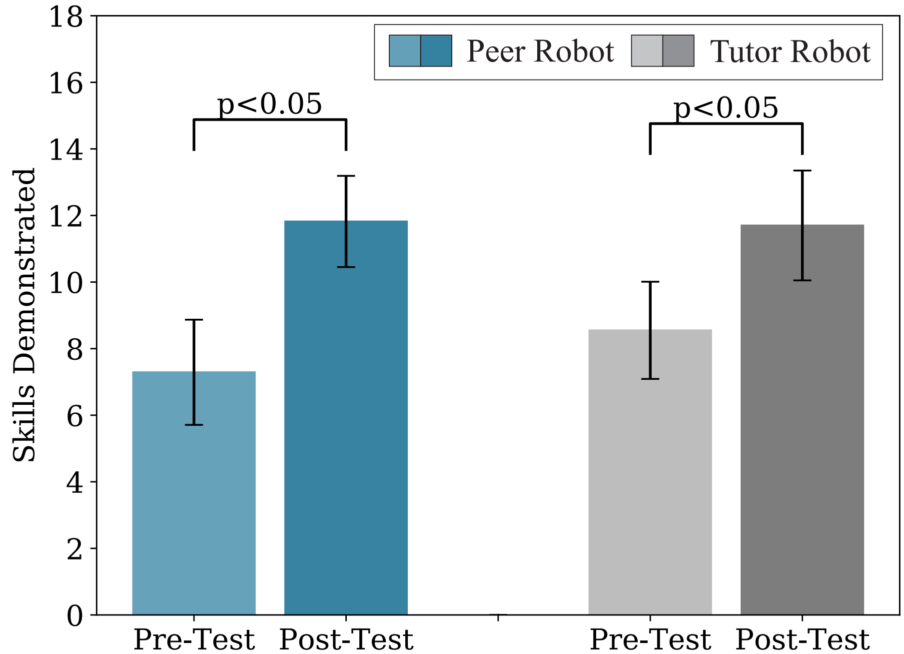


Figure 5.6: (a) Participants significantly improved their circuit knowledge skills from pre-test to post-test in both conditions. (b) There were no significant differences in number of skills learned between conditions.

believe our manipulation check was successful.

5.4.2 Test Results

We compared the skill increase in each condition from pre-test to post-test on the 17 skills. A skill is attributed as known when the participant has correctly applied it at least half of the time. On average, participants in the peer robot condition scored 7.29(SD = 3.16) on the pre-test and 11.82(SD = 2.74) on the post-test. Participants in the tutor condition scored on average 8.55($SD = 2.93$) on the pre-test and 11.70(SD = 3.31) on the post-test. An ANOVA comparing moment (pre-test and post-test) and condition found significant differences $F(3, 74) = 10.01, p < 0.001$. A Tukey HSD test revealed that both the peer condition ($p = 0.001$) and the tutor condition ($p = 0.009$) significantly improved from pre-test to post-test, as seen in Figure 5.6. There were no significant differences between conditions for the pre-test ($p = 0.587$) or the post-test ($p = 0.900$).

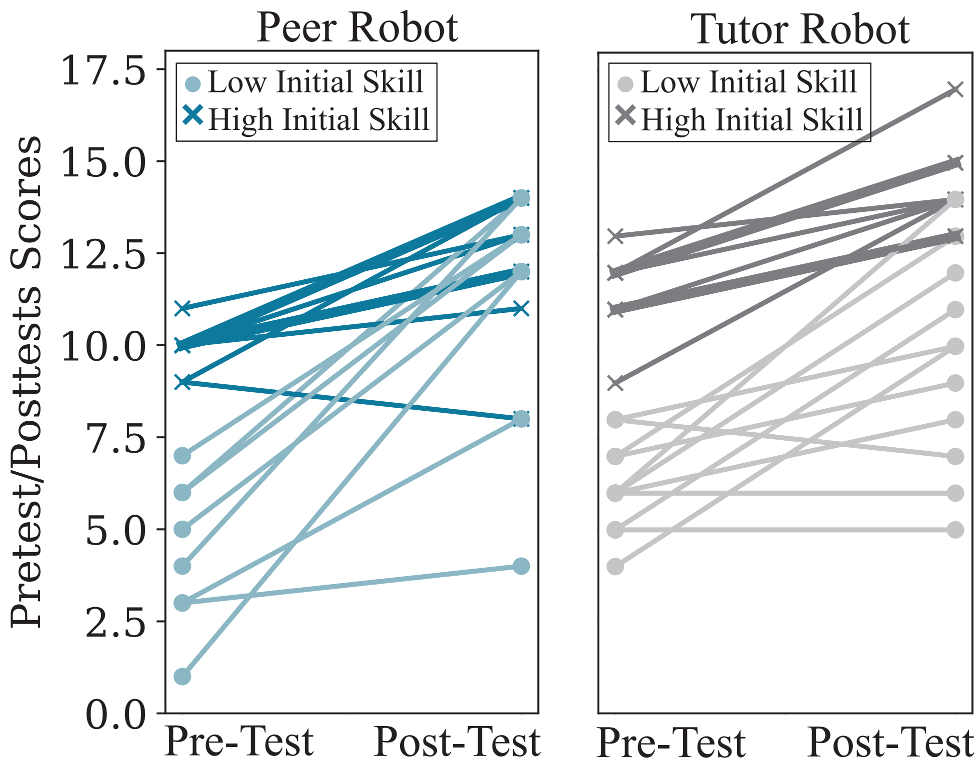


Figure 5.7: The pre-test and post-test scores for the peer and tutor conditions. There were no significant differences in skills gained between conditions. However, participants with low skill knowledge improved their skills significantly more with the peer robot than the tutor robot.

Participants in the peer condition learned on average 4.53 ($SD = 3.22$) new skills, whereas participants in the tutor condition learned on average 3.15 ($SD = 2.37$) new skills. These differences were not significant $t(37) = 1.46, p = 0.154$. Next, we compare participants with prior low electronic circuit knowledge and participants with high prior electronic circuit knowledge. We compared the number of learned skills between condition and prior knowledge using an ANOVA and found significant differences $F(3, 37) = 5.47, p = 0.004$. A Tukey HSD test revealed that participants with low prior knowledge learned significantly more in the peer condition than the tutor condition (p=0.023), but no significant differences were found for high prior knowledge participants (p=0.900). These results can be seen in Figure 5.7.

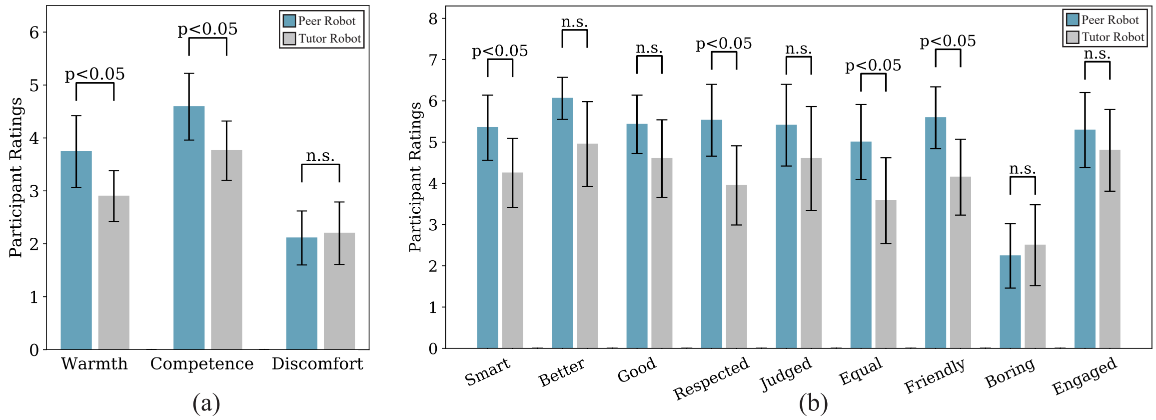


Figure 5.8: Questionnaire results. (a) The peer robot was perceived as significantly more social and intelligent than the tutor robot. (b) The peer robot was perceived as significantly smarter, more respectful, friendly than the tutor robot, in addition to participants feeling more like they were treated as an equal.

5.4.3 Behavioral Results

In the peer condition, there were six participants who engaged in conversation with the robot, and nine participants who did not. There were two participants whose audio data was corrupted. In the tutor condition there were eight participants who engaged in conversation with the robot and twelve who did not. Using a Chi-Squared test, these results were not statistically significantly different from each other $X^2(1, N = 35) = 0, p = 1.000$.

5.4.4 Questionnaire Results

On the RoSAS questionnaire, participants rated the robot as more warm in the peer condition ($M = 3.74, SD = 1.37$) compared to the tutor condition ($M = 2.90, SD = 0.96$). Participants also rated the robot as more competent in the peer condition ($M = 4.59, SD = 1.27$) than the tutor condition ($M = 3.76, SD = 1.12$). Lastly, participants rated the robot similarly in regards to discomfort between the peer ($M = 2.11, SD = 1.03$) and tutor conditions ($M = 2.20, SD = 1.19$). Their ratings were significantly different for warmth $t(37) = 2.18, p = 0.036$, and for competence $t(37) = 2.09, p = 0.044$, but not for discomfort $t(37) = -0.25, p = 0.804$. The RoSAS questionnaire results are seen in Figure 5.8(a).

On the post-experiment questionnaire, participants in the peer condition ($M = 5.35, SD = 1.58$) rated the robot significantly smarter than the tutor condition ($M = 4.25, SD = 1.68$), $t(37) = 2.04, p = 0.049$. Participants rated the robot as being better than them at electronic circuits in the peer condition ($M = 6.06, SD = 1.03$) than the tutor conditions ($M = 4.95, SD = 2.06$), but these differences were not quite significant $t(37) = 2.01, p = 0.052$. Lastly, participants rated the robot a $5.42(SD = 1.42)$ on being good at circuits in the peer condition, and a $4.60(SD = 1.88)$ in the tutor condition. These differences were not significant $t(37) = 1.46, p = 0.152$.

Participants felt more respected by the peer robot ($M = 5.53, SD = 1.74$) than the tutor robot ($M = 3.95, SD = 1.93$), and this difference was significant $t(37) = 2.59, p = 0.014$. Participants in the peer condition rated the robot a 3.06 (SD = 1.98) for feeling judged and a 3.60 (SD=2.52) in the tutor condition, $t(37) = -0.72, p = 0.479$. Lastly, participants in the peer condition ($M = 5.00, SD = 1.83$) perceived the robot as treating them as an equal significantly more than participants in the tutor condition ($M = 3.58, SD = 2.09$), $t(35) = 4.50, p = .041$.

Participants viewed the robot as more friendly in the peer condition ($M = 5.59, SD = 1.50$) than the tutor condition ($M = 4.15, SD = 1.84$), $t(37) = 2.57, p = 0.015$. Participants did not think the robot was more boring in one condition than the other (peer: $M = 2.24, SD = 1.56$; tutor: $M = 2.50, SD = 1.96$), $t(37) = -0.45, p = 0.657$, nor did they feel more engaged in one condition than another (peer: $M = 5.29$, $SD = 1.83$; tutor: $M = 4.80$, $SD = 1.99$; $t(37) = 0.77$, $p = 0.447$). These results can be seen in Figure 5.8(b).

5.5 Discussion

This work is the first to show multiple benefits in peer tutoring while only manipulating minor aspects of the robot (small differences in its utterances and of how the experimenter presented the robot). Most research in peer robot tutoring was conducted with children and was either not focused on peer vs. tutor [147, 8] or had many differences between conditions [50, 268]. Additionally, we believe this is the first HRI work comparing peers and tutors that shows significant differences in learning using a pre-test and post-test rather than other measures (such as engagement and facial expressions).

5.5.1 Hypotheses

Participants in both the peer condition and the tutor condition significantly improved their electronic circuit skills from pre-test to post-test. This shows that the robot in both conditions successfully taught the adults. Therefore Hypothesis 1a is true: *Adults in both conditions showed significant improvement in electronic circuit skills from pre-test to post-test.* Participants did not learn more skills in the peer condition compared to the tutor condition. Therefore we cannot confirm Hypothesis 1b; *Adults did **not** learn more from a peer robot than a tutor robot.* Participants with high skill knowledge did not have significantly different skill increase between conditions. Therefore in regards to Hypothesis 1c, *Adults with high initial knowledge did **not** especially benefit from a peer robot, compared to adults with high initial knowledge interacting with the tutor robot.* On the contrary, participants with low circuit knowledge learned significantly more with the peer robot. These results differ from those seen in human adult peer-to-peer tutoring. Therefore, the robot taking on the role of a peer should be especially considered in scenarios where the person likely has low prior knowledge in the domain.

Participants did rate the peer robot more positively in several dimensions. On the post-experiment questionnaire, participants rated the peer robot as significantly more social and as significantly friendlier than the tutor robot. Therefore we believe that Hypothesis 2 is true: *Adults viewed a peer robot more positively than a tutor robot.* It is important that people view the robot positively when interacting with it, as they will likely be more engaged and learn from the robot in the long term.

In most human-robot interaction studies, engagement is assessed using gaze patterns [207]. However, due to the COVID-19 pandemic, participants wore masks during the interactions, which made computer vision systems that tracked participant faces unreliable. Therefore we measured engagement using the amount participants talked to the robot and their self-assessed engagement on the questionnaire. Participants did not significantly view the robot as being more boring in one condition than another. Neither did they report being more engaged. There were also no significant differences between engaging in conversation with the robot between conditions. Therefore, we do not support Hypothesis 3; *Adults were **not** more engaged with a peer robot than a tutor robot.*

Participants reported feeling significantly more respected by the peer robot compared to the tutor robot. Additionally, participants felt that they were treated significantly more as an equal when interacting with the peer robot. Therefore, we confirm Hypothesis 4: *Adults felt more respected from a peer robot than a tutor robot.* This is important, as feeling respected is an essential factor in learning success [58].

Participants in the peer condition viewed the peer robot as significantly smarter than participants in the tutor condition. Additionally, participants rated the peer robot as significantly more competent than the tutor robot. Therefore we confirm Hypothesis 5: *Adults saw a peer robot as more intelligent than a tutor robot.*

5.5.2 Expectations of A Tutoring Robot

One possible reason the peer robot was rated as more intelligent than the tutor robot was because participants had lower expectations of a peer than a tutor. A tutor is presented as an expert, whereas there is more uncertainty involved in the capabilities of a peer-teacher. An open-ended question asking participants whether they wished the robot had done anything differently confirmed that many wished the robot had additional capabilities. Participants wished the robot had given examples of completed circuits, had the ability to answer participants' questions, and had given step-by-step instructions for the more complicated circuits.

Domains in adult tutoring are often more complex than those seen in children's tutoring, with many requiring computer vision systems to model the interactions. Therefore giving personalized advice to adults is often not as straightforward as providing help during child-robot interactions. Presenting the robot as a peer could lower expectations. As a consequence, people might be more willing to receive advice from it than they would from a tutor robot whose expectations are not met.

5.5.3 In-group/Out-group effects

In the peer condition, the robot presents itself as being in-group with the participant when using the pronouns "we/us". Whereas in the tutor condition, the robot presents itself as an authority figure by placing itself in the out-group when using the "you" pronoun. People evaluate robots more positively when they are in-group than when they are out-group [110]. This is one potential confound in our work, where part of the results could be due to in-group/out-group membership.

5.6 Summary

This chapter explored different roles a robot can take when teaching people about electronic circuits. The robot would either take on the role of a peer or the role of a tutor. Participants with low prior circuit knowledge learned significantly more with the peer robot than with the tutor robot. This shows the benefits of peer robots, especially in domains where the user is likely lower-skilled. Additionally, participants who interacted with the peer robot viewed it as more friendly, more social, more intelligent, and felt more respected than participants who interacted with the tutor robot, independent of prior knowledge. These are all essential qualities for a robot to have for enabling long-term user learning.

This is the first work that shows significant learning differences when comparing a robot that acts like a peer versus when it acts like a traditional teacher. However, according to a review on robot tutoring, 86% of studies still have a robot teaching as a tutor [31]. Based on our findings, we recommend that future robot tutoring studies, especially those targeting adults, consider using robots that interact as if they were a peer.

Chapter 6

Social Robot Characteristics: Investigating the Ability of Robots to Indirectly Influence People*

For robots to be effective tutors, they need the ability to influence those around them. People need to be willing to receive information from a robot and trust the robot to be providing good teaching [71]. However, we still have a limited understanding of how robots influence those around them, especially when they are not trying to be explicitly persuasive. Therefore, this chapter investigates one dimension of persuasion called conformity. Conformity is when a person changes their behaviors or answers to match those of a group of agents. We investigate whether a group of robots cause conformity and the reasons people conform to robots.

Psychology literature has shown that people conform their answers to match those of human group members even when they believe the group's answer to be wrong [15].

*Parts of this chapter were originally published as: Nicole Salomons, Sarah Strohkorb Sebo, Meiying Qin, Brian Scassellati. (2021). A Minority of One against a Majority of Robots: Robots Cause Normative and Informational Conformity. In *ACM Transactions on Human-Robot Interaction (THRI)*[222] **and** Nicole Salomons, Sarah S. Sebo, Michael van der Linden, Brian Scassellati. (2018). Humans Conform to Robots: Disambiguating Trust, Truth and Conformity. In *Proceedings of the Thirteenth Annual ACM/IEEE International Conference on Human-Robot Interaction (HRI)* [223].

We tested whether robots can similarly cause humans to conform to them during a subjective task. We also analyze whether robots cause informational conformity (believing the group to be correct), normative conformity (feeling peer pressure), or both. We conducted an experiment where participants (N = 63) played a subjective game with three robots. We measured humans' conformity to robots by how many times participants changed their preliminary answers to match the group of robots' in their final response.

Our results show that participants change their answers to match the robots more frequently when they are aware of the robots' answers than when they are not. Therefore we show that robots can cause humans to conform to them. In sequence, we investigate what types of conformity were at play. Participants in conditions that were given more information about the robots' answers conformed significantly more than those given less, indicating that informational conformity is present. Participants in conditions where they were aware they were a minority in their answers conformed more than those unaware they were a minority. Additionally, they also reported feeling more pressure to change their answers from the robots. The amount of pressure they reported was correlated to the frequency they conformed, indicating normative conformity. Therefore, we conclude that robots can cause both informational and normative conformity in people.

6.1 Introduction

"Conformity refers to the act of changing one's behavior to match the responses of others" (page 606)[55]. One of the most foundational psychological studies measuring conformity was performed by Asch in the 1950's [15]. When individually asked to answer a very simple perceptual question (identifying which line out of a set of lines matched another line in length), participants responded correctly 99% of the time.

However, when placed second to last in a group of six to eight confederates, if the confederates unanimously verbalized an incorrect answer before them, the participant would choose the same incorrect answer 37% of the time.

Shiomi et al. and Brandstetter et al. attempted to replicate the Asch paradigm with robot confederates instead of humans [234, 37]. Still, neither study was able to show conformity to a group of robots. Possible reasons suggested for the lack of conformity included lack of a social relationship between the participant and robots, participants not viewing the robots as authoritative entities, and the robots not being human-like enough [234, 37].

These results are somewhat surprising, as research has shown that individual robots can persuade people. Siegel et al. shows that participants voluntarily donated money to a persuasive robot [237]. Another example of persuasive robots can be seen in research conducted by Chidambaram et al., in which participants comply with a robot's suggestions when playing a game [51]. Additionally, in cases such as lexical entrainment, the influence of robots can persist even after the interaction ends [36]. Therefore, it is unexpected that groups of robots fail to provoke conformity in their interactions with people.

Hodges and Geyer [115] offer one possible alternative interpretation of Asch's conformity experiment, in which they propose that participants were constrained by multiple influencing factors, including trust, truth, and conformity. They argue that participants were placed in a challenging situation in which they had to trade off their trust in the group members, their desire to give a truthful answer, and the pressure to conform to the group. They support this argument by pointing out the time-varying nature of participant responses, often interweaving correct (non-conforming) answers with false (conforming) answers, and the overall low conformity rate (37%).

Robots cause less social pressure than humans, and their trustworthiness level is unclear compared to a person. We propose to unravel the interplay between truth, conformity, and trust in human-robot groups by changing the task to one where there is no apparent objective truth. This will allow us to investigate whether a group of robots cause conformity when the answer is less clear.

Furthermore, as previous studies focused on tasks where the correct answer was clear, this only allowed for one type of conformity: normative conformity. However, psychology literature identifies two main types of conformity: normative conformity and informational conformity [70].

Normative conformity is conformity due to feeling pressure to change one's answer or behavior to be equal to the rest of the group. One classic example of normative conformity is shown in Asch's 1950s line task [15]. In his studies, a participant was shown a line and was asked which of three other lines was the same length. When answering alone, only 1% of people chose the wrong answer, but when a group of confederates all stated the wrong answer, the participants conformed to the group, answering incorrectly in 37% of the trials. His experiments were a clear example of conformity due to peer pressure: the answer was obvious, but participants still chose the incorrect answer a significant number of times to not differ from the rest of the group.

Informational conformity is conforming due to believing the other group members know the correct answer or behavior. Informational conformity frequently occurs when one decides which product to purchase. For example, Cohen and Golden show that participants were influenced by supposed peer ratings when rating a coffee product [57]. Participants rated the coffee product after tasting it and observing a board with other people's product ratings. Participants were more likely to score the product higher when observing others rate it higher than when they had no information about the others' ratings. No significant differences were found in whether the participant was told that their rating would be visible to future raters. Their results suggest that participants were incorporating the ratings of others' into their own ratings because they believed them to be accurate.

This chapter analyzes whether 1) participants conform to a group of robots when the task is more subjects, and 2) whether they are being led by informational conformity, normative conformity, or both. Therefore we have two hypotheses for this study:

Hypothesis 1: *When provided with the opportunity to conform without providing an obviously false answer, participants will conform with a group of robots at a rate similar to the original Asch participants.*

Hypothesis 2: *Participants conform to the group of robots due to both a) informational conformity and b) normative conformity.*

6.2 Related Work

This section presents a literature overview of previous work conducted on conformity in both human groups and mixed human-robot groups.

6.2.1 Conformity in Human Groups

Conformity is when one changes their own behavior or choices to match the behavior or choices of those surrounding them [55]. Asch's line experiments show that participants will often change their answers to match a group of confederates answers even when the group is clearly incorrect [15]. Asch conducted several further experiments [15] in which he varied the number of trials, the number of human confederates, and the amount of ambiguity of the lines, among other factors. He concluded that in all the different variations, people consistently conformed to the answers of the group. In one of the additional experiments, the participant would write their answer on paper rather than stating the answer, after hearing the confederates verbalize their answer. In this experiment, the conformity rate decreased to only 12.5%. The fact that conformity was much more frequent when giving public answers indicates that participants were acting due to feeling peer pressure from the confederates to give the same answer as the rest of the group.

Research has shown two main reasons people conform to a group: normative conformity and informational conformity [70]. In normative conformity, the person is conforming to the expectations of others, usually because they feel peer pressure to do so. Social impact theory predicts that normative conformity increases due to three factors: group size, immediacy, and social importance [150]. Other factors that increase normative conformity are giving answers publicly rather than privately [70] and having a unanimous group [15]. Asch's study [15] was a clear example of normative conformity: the participants did not think the majority was correct but conformed to the group nonetheless when verbalizing their answer.

Informational conformity is behaving or answering according to the group due to gaining and accepting information from them. Toelch and Dolan define it as "information that is acquired through sampling of the environment with the goal to make adaptive decisions that are optimized for the current context." [252]. In informational conformity, people change their answer/behavior to match the group not because they are feeling pressured to do so, but because they are adopting that information as their own. Cialdini stated that the main factors that influence informational conformity are uncertainty in the correct behavior/answer and similarity with the group [54]. Other factors in informational conformity are group size and expertise of the group. An example of informational conformity can be observed in work by Lucas et al. [166], in which they studied conformity in a math task. Their results show that participants were more likely to conform to a group when the math problems were difficult and when their self-efficacy in that particular skill was low.

Normative and Informational conformity are the two main types of conformity. However, they are not always easily distinguishable, as people are frequently influenced by both at the same time [127]. For example, when buying clothes, pre-adolescents are influenced by their friends' opinions on style and what their friends like to wear [175], which are a mix of information and normative reasons. Furthermore, there may be other reasons to conform to a group [128, 165]. In this chapter, we focus on normative and informational conformity and try to disambiguate between them in a subjective task.

6.2.2 Conformity with Non-Human Agents

Similar experiments to Asch's conformity experiment were conducted by Beckner et al. [28] and Brandstetter et al. [37], where conformity was tested with both a group of robots and a group of human actors. They administered Asch's conformity line test and also tested a verbal task where participants determined verb tenses. There were four NAO robots present, which all stated the same wrong answer in some of the rounds. Participants conformed to human confederates during the experiment but did not conform to the robots. Rather, the amount of conformity to robots was not significantly different from a baseline condition where participants pressed a button instead of verbalizing their answer. Shiomi and Hagita also tested conformity with robots using Asch's line task [234]. Conformity was tested in 12 out of 18 rounds with two conditions: one where the robots synchronized their actions (by first looking at the previous agents and then the next agent before answering), and one condition where the actions were not synchronized. Neither condition demonstrated conformity compared to a condition where no robots were present.

Children are more likely to conform to robots compared to adults. Vollmer et al. [255] tested Asch's study with robots as the confederates on children seven to nine years old in addition to adults. Similar to previous studies, adults did not conform to the robots. However, children did demonstrate a significant frequency of conformity to robots, suggesting that children are more susceptible to the influence of robotic agents. Williams et al. [262] also show child-robot conformity (ages 4-10) when answering socio-conventional and moral questions with a robotic doll.

Hertz and Wiese also studied Asch's line task with robots in addition to humans and computers, where they compared high ambiguity and low ambiguity lines [113]. Participants observed videos of either one robot, one person, or one computer answering the line task before answering themselves. Ambiguity was introduced by how long participants saw the lines: 1000ms (low) vs. 400ms (high). Their results show that there was a low overall conformity rate of 22.3%, but that there were significantly higher differences in conformity with high ambiguity lines compared to low ambiguity lines. Further experiments by Hertz and Wiese analyzed different tasks than Asch's line experiment [114]. They compared conformity towards three different groups of agents: humans, robots, and computers. They tested two different task types; the first was a social task in which participants observed images of people's eyes and selected the emotion they believed the eyes were expressing and a second analytical task in which participants conducted addition and subtraction with a series of dots on the screen. Before participants selected their own answer, they observed either a robot hand, a human hand, or a computer code select an answer, depending on condition. Similar to Asch, in 24 out of 36 rounds, the agents would unanimously select the incorrect answers. Their results show that overall there is no significant difference between conformity rates in the different agents, showing that robots and computers are capable of causing conformity similar to humans. However, there was a difference in conformity for the different types of tasks: participants conformed less to robots and computers in the social task, while in the analytical tasks, the conformity rates were very similar. There are several limitations to the studies conducted by Hertz and Wiese [114, 113]. The robots were not present in person but only shown through videos; there was only one other agent in the videos, but conformity is usually related to groups; the second study [114] only showed videos of hands and not of the whole

Figure 6.1: In this experiment, participants sat around a table playing a game with three myKeepon robots. In certain rounds of the game, the three robots choose a different answer than the one the participant chose. Participants often changed their answers to match the answer of the robots, demonstrating conformity to the group of robots.

robot; and there was no feeling of perception that the agents or robots observed the participants' answer, making normative conformity unlikely.

6.3 Methodology

In this study we aim to study whether groups of robots can cause conformity in human participants. And if participants conform to robots are they conforming to the robots to have the correct answer (informational conformity) or because they are feeling peer pressure from the robots (normative conformity).

In our experiment, participants sat around a table with three robots playing a subjective game that did not have a clear, correct answer. A subjective game meant that prior knowledge of the game was not a main factor. Participants played 20 rounds of the game, and each round was composed of two stages. In the first stage, the participant gave a preliminary answer, and in the second stage, they gave their final answer. This mechanic permitted us to measure when they changed their mind directly and allowed us to manipulate the robots' answers depending on the participant's answer in order to maintain consistency across participants. In some of the rounds of the game, which we call **critical rounds**, all of the robots chose a different answer than the participant. It was in these rounds that we tested whether the participant conformed to the robots.

We devised four different experimental conditions to test the degrees to which people conform to robots. In three of these conditions, we varied the amount of information participants received about the robots' answers. People seek information from the environment when uncertain [252]. Therefore, we tested whether participants believed the robots to have the correct answer by varying the visibility of the robots' answers. If participants trust in the answers of the robots, when provided sufficient information about the robots' answers, they will conform to them. However, when provided limited information about the robots' answers, they will not be able to conform. If participants do not trust in the answers of the robots, then they will not conform due to informational reasons independent of the amount of information provided by the robots.

The three conditions with varying amounts of information could also lead to a change in normative conformity, with more information leading to more conformity as the participant would be aware they were a minority in their answer. In the conditions with more information, if the participant does not conform, it would be visible to the whole group that they chose a different answer. Thus having increased information may lead to feeling more peer pressure. The increase of information about the robots' answers influences both normative and informational conformity. Completely separating information and normative conformity is difficult as they are both frequently present [252, 127].

The fourth condition was similar to the condition with the highest amount of

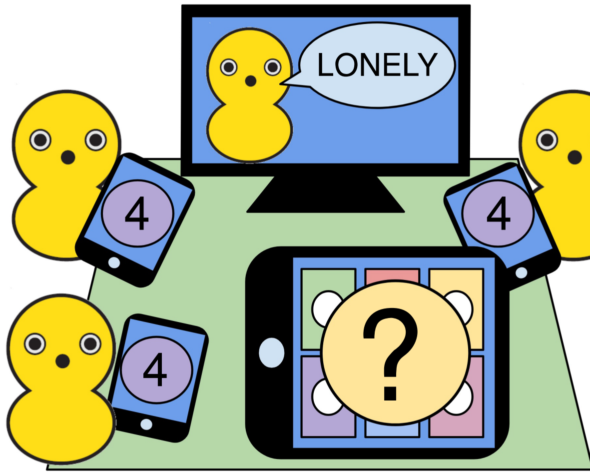


Figure 6.2: Participants were given a word by a robot on a screen and then chose out of six pictures the one they best believed corresponded to the word. In certain rounds, the three robots would select an opposing answer than them, and the participant had to decide whether to conform to the robots or to continue with their initial answer.

information, but a peer pressure behavior was included in the form of staring. Eye gaze is a tactic often adopted when trying to persuade [136] and has been shown to increase compliance [233]. Therefore, we tested whether staring increased the frequency of normative conformity.

To further disambiguate between normative and informational conformity, participants completed post-experiment questionnaires assessing the reasons they conformed and whether they felt pressure to change their answers because of the robots.

6.3.1 Procedure

Participants were seated around a table with three myKeepon robots [142]. MyKeepon robots are small yellow robots, which were dressed in colored hats to give each robot a unique personality. The robots were present in the same room as the participant, as it has been shown that being physically close to the group increases conformity [150]. Additionally, the number of robots was chosen to be three because previous studies have shown that conformity increases with the number of agents but that there are no significant differences after having more than three agents [15].

Each of the robots and the participant were given a personal tablet, and there was a shared screen that all the agents could observe.

Participants played a game with the robots in which, in each round, they were shown a set of six cards with drawings. Along with the six cards, a fourth robot (called the "game master") gave the group a word for the six cards, such as "Irony" or "Leader" (as seen in Figure 6.2). They were asked to select the card they believed the game master had selected as the correct answer. The cards used were digitalized images from a commercial board game called Dixit manufactured by Libellud [160]. * They contain detailed drawings, which often did not have one singular interpretation. After having chosen a preliminary answer on a personal tablet, the participant was given some information about the robots' answers on a shared screen. In sequence, the participant selected the final answer, and then the game master robot gave the group the "correct" answer. As the game is subjective, there is no absolute correct answer, so it was defined as the one the game master thought was the correct answer. Participants played a total of 20 rounds of the game with the robots, in which the word, the specific cards, and the answers for the round were chosen beforehand.

The sequence of the task

After completion of the consent form, the experimenter led the participant into the testing room and greeted the robots and the participant. The experimenter explained the game, and then the participants and the robots completed a practice round. After the practice round, the experimenter gave the robots and the participants a chance to ask questions. One of the robots was pre-programmed to ask whether it was OK to change their answer, and the experimenter said that they could all change their answers as many times as they wished. If the participants had any further questions, they were answered. Afterward, the experimenter left the room, and the participant

*Participants might have been familiar with the cards because of the board game, however the task they did with the robots was novel to them.

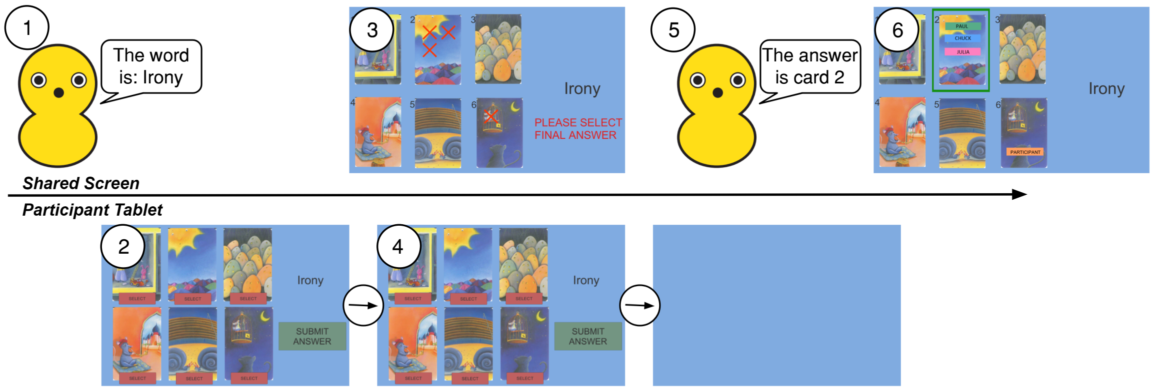


Figure 6.3: The sequence of each round. (1) A "game master" robot gave a word on the shared screen. (2) The participants chose the card that they believed best corresponded to the word on their tablets. (3) Information about the robots' answers was given. In this case, the quantitative condition is shown where red "X's" were shown for the chosen cards on the shared screen. (4) The participant chose their final answers on their tablets. (5) The game master gave the correct answer on the shared screen. (6) Each agents' final answer was shown by displaying their name on top of the card they chose on the shared screen.

played 20 rounds of the game with the group of robots (the sequence of what happened in each round can be seen in Figure 6.3). Each round proceeded as follows:

1. A word was announced by a video of a fourth robot (the "game master") on the shared screen.
2. The participant and the three robots individually chose one of the six cards that they felt best represented the word out of six cards on their personal tablets.
3. Participants were given full, partial, or no information on the robots' answers depending on the condition they were in. Robots directed their gaze as determined by the experimental condition.
4. The participant and the robots were given the opportunity to change their answer to one of the other cards (for the same word).
5. The correct answer was given by a video of the "game master" robot on the shared screen.
6. The robots' and participant's answers were shown publicly to everyone by displaying their names on top of the cards they chose for their final answer.

After the participant had played 20 rounds with the robots, they completed a post-experiment questionnaire, which is detailed in Section 6.3.5.

User Interfaces: Tablet and Shared Screen

There were two different user interfaces used in the study: the tablet user interface and the shared screen. Each of the agents (the participant and the three robots) had their own tablet through which they gave their preliminary and their final answers. The tablet showed a blue screen when the participant was not selecting an answer. A tablet was used instead of physical cards, as the robots did not have the dexterity to manipulate cards. The tablets also provided a mechanism for each agent to provide their answer without it being visible to the other agents.

There was a shared interface that was visible to the participant and to all the robots, called the shared screen. The shared screen showed videos of a fourth robot called the "game master" which gave the word at the start of each round and also provided the correct answers at the end of each round. Additionally, the shared screen showed the preliminary answers (Figure 6.4a) and the final answers publicly (Figure 6.4b).

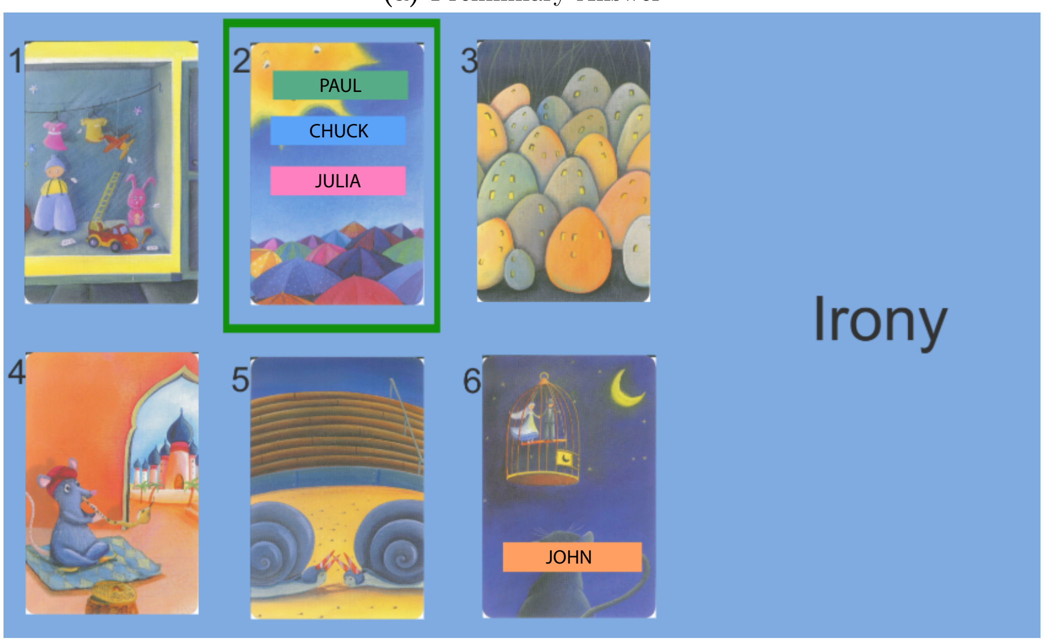
The shared screen facilitated social pressure on the participant as the participant and the robots all looked at the same screen. The participant felt as if the robots clearly could see when the participant chose a different answer than them; thus they may have felt peer pressure from the group.

6.3.2 Conditions

In this study, we present a between-subjects study in which 63 participants were spread across four different conditions.

- **Blind Condition**: Participants were given no information about the robots' preliminary answers. Instead, they were just shown a screen with the cards

(a) Preliminary Answer



(b) Final Answer

Figure 6.4: In this example of the quantitative condition, the preliminary answer (a) of all of the robots and the participant was publicly shown on the shared screen. After having chosen their final answer, each of the answers of the agents was shown on the shared screen. The names of each of the robots (Paul, Chuck, and Julia) were shown on top of their chosen cards, and the participants' answer was also marked with their name (John in this case).

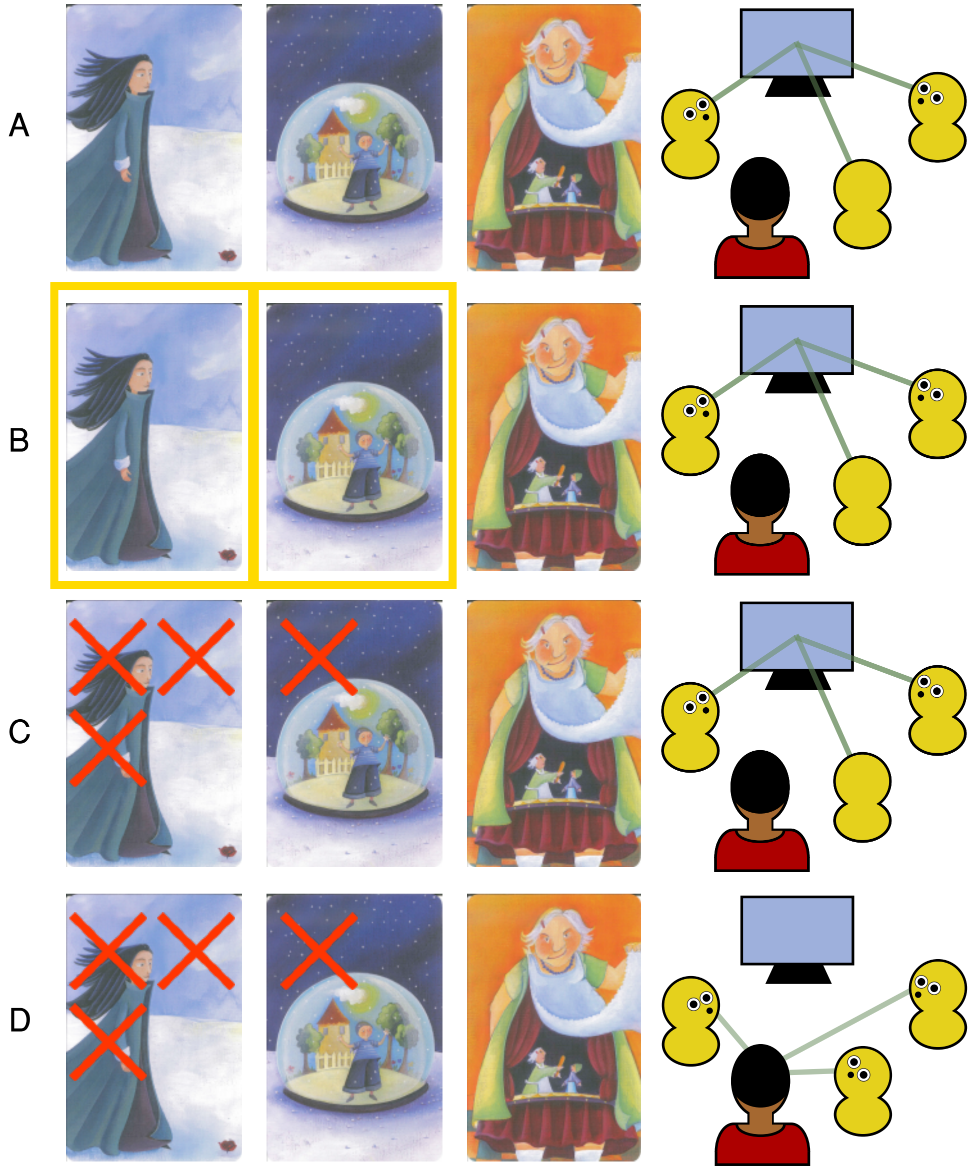


Figure 6.5: Examples of the preliminary answers of the robots in the four conditions. In the **blind condition** (A), no information was shown about the robots' preliminary answers, and the robots looked at the shared screen after the participant had selected their answer. In the **selected condition** (B), the selected cards (that were chosen by at least one robot or the participant) were shown with yellow squares around them, and the robots looked at the screen after the participant had selected their answer. In the **quantitative condition** (C), each robot's answer and the participants' answer was represented with a red X on top of their chosen card, and the robots looked at the screen after the participant had selected their answer. In the **staring condition** (D), the robots' and participants' answers were represented with a red X on top of the chosen card. During critical rounds, all three robots first looked at the screen briefly and then turned around and stared at the participant for several seconds. During non-critical rounds, the robots looked at the screen after the participant had selected their answer.

again (Figure 6.5A). During each round, the robots looked at the screen when the screen showed the cards again.

- **Selected Condition**: Participants could see which preliminary answers were selected by at least one robot but not how many robots chose each card. Figure 6.5B shows that both the first and second cards were chosen by at least one person/robot, but no information was given about how many chose each of the answers. The robots looked at the screen when the preliminary answers were shown.

- **Quantitative Condition**: Participants were shown an "X" on top of each card chosen by a robot. An example can be seen in Figure 6.5C, where three players chose the first card, one player chose the second card, and none chose the last card, but no information was given about which red "X" corresponded to whom. The robots looked at the screen when the preliminary answers were shown.

- **Staring Condition**: Participants also saw the robots' answers in the form of red "X"s on the screen (identical to the quantitative condition). However, whenever the robots all chose a different preliminary answer than the participant in the critical rounds, they all first looked at the screen briefly and then stared at the participant for several seconds. If the participant continued choosing a different answer than the robots for their final answer, the robots would stare again at the participant after the final answers were revealed. During all the non-critical rounds, the robots did not exhibit the staring behavior.

We only created the condition that included the staring behavior in the case where the red X's appeared for each answer. In the other two cases, the robots would not have sufficient information to know there was a minority present, and therefore the staring behavior would not make sense.

6.3.3 Rounds

Participants played 20 rounds of the game with the robots. For each round, the cards that were shown and the word were chosen beforehand. There was a variety in the types of rounds to make the game feel more realistic. For example, there were rounds where the correct answer was very clear, and all three robots chose the same card. There were rounds where all their answers diverged. Furthermore, there were rounds where one or more of the robots changed their answers, and there were rounds where they did not change their answers. This showed that it was permissible to either change or keep your original answer. The "correct" answers for the rounds were also chosen beforehand. For the easier rounds, the most plausible answer was usually correct. For some of the rounds, the answer that the participant chose was correct, and for others, the answer that the robots chose was correct.

The different rounds were based on Table 6.1. Because the robots adapted to the choices of the participant, the robots' choices did not always perfectly follow the pre-planned rounds.

- **Unanimous Rounds** - There was one picture that seemed more correct than the others. All three of the robots chose the same preliminary answer and did not change their answer for their final answer. The participant was also expected to choose this answer.

- **One Robot Converges** - During the preliminary round, two robots chose one answer, and one chose a different answer. The robot with the different answers converged to match the group for its final answer.

- **Two Robots Converge** - During the preliminary round, one robot chose the same answer as the participant, and the two other robots chose a different answer. The two differing robots changed their answer to be the same as the participant.

- **One Robot different** - One robot was different from the other robots. The participant could have aligned their answer with the two robots or with the one robot. The different robot did not change its answer.

- **All different** - All the robots chose a different answer from each other. In some of the rounds, one of the robot's answer might overlap with the participant's answer. In some of these rounds, one of the robots changes their answer to either match a participant or to match another robot.

- **Critical Rounds** - All the robots chose the same opposing answer from the participant. These are the rounds in which we were testing conformity.

The answers of the robots were adjusted to the answers of the participant to make the experience of the participants as similar as possible. For example, in the "two robots converge" case, when the robot chose a particular answer, one of the other robots also chose it. The two remaining robots chose a separate reasonable answer. And in the "One robot different" case, two of the robots were assigned the same answer that the participant chose, and the third robot was assigned a different answer. By adjusting the behavior of the robots, participants experienced the same scenarios independent of their own preliminary answers, and they observed the robots get the answers right/wrong the same amount of times and on the same rounds. The exception to this was on the unanimous rounds where the answer was very apparent, and the robots consistently chose the correct answer as a clear wrong answer might make the participants question the robots' capabilities.

Critical Rounds

Out of the 20 rounds, six were **critical rounds**. In these rounds, the three robots were programmed to unanimously choose a different plausible answer opposing the participant's answer. For example, in one of the rounds, the word was "Immense," and

Round Numbers	Type of Rounds
1,3,6,8,18	Unanimous Round
2,4	One Robot converges
11,19	Two Robots Converge
12	One robot different
7,14,15,17	All different
5, 9, 10, 13, 16, 20	Critical Round

Table 6.1: The round number with the type of round.

one of the options was a large landscape with a night sky above it, and another option was a monster with its mouth open. When the participant chose the landscape, then all three robots would choose the monster and vice versa. The six critical rounds where the robots unanimously diverged their answers were the rounds in which we observed whether participants conformed to the robots. In three of the six rounds, the participants' initial guess was right, and in the other three rounds, the robots' preliminary guess was correct. This was kept balanced to prevent participants from believing that the robots were always correct or always incorrect.

6.3.4 MyKeepon Robots

During the experiment, three myKeepon robots were used. MyKeepon robots have four degrees of freedom and are commercialized versions derived from a research robot called Keepon Pro [142]. The robot is a 15cm tall robot composed of two spheres giving it a snowman-like appearance with a soft yellow exterior foam. The three robots used are shown in Figure 6.7. Each robot was capable of moving to look at the different robots and also to look at the different screens. Additionally, the robot was programmed to sway side-to-side when an audio file played to simulate talking. Each of the robots had a unique name, a different recorded voice, was dressed differently, and had different styles of utterances (for example, one made some jokes, and one was a bit shy), so they appeared to have different personalities. Previous studies have

Figure 6.6: Examples of several of the critical rounds, with their words and images. The pictures highlighted in yellow were the two most reasonable answers. Whichever of the two that the participant chose for their preliminary answer, the three robots unanimously selected the other one. When a participant chose neither of the two, the robots had a predetermined one of the two that they chose.

shown that diverse human groups are attributed more agency [177] and that diverse robots are perceived as more intelligent [97] than entitative robots. Therefore we chose to have the robots look and behave slightly different from each other. They performed utterances during the game such as "I am not sure about this one," "hmmm," or "this one I think I know." During the critical rounds, the robots did not say anything to avoid confounds created by verbal persuasion.

We chose to have three robots in the experiment, as three robots are the minimum necessary for them to be considered a group [264]. In Asch's studies, he founds that more confederates present increased conformity, but few differences were found when the group was larger than three confederates [15]. Two of the robots were assigned

Figure 6.7: In this human subjects study, a human participant interacted with a group of MyKeepon robots. Each of the robots was dressed uniquely and had a different voice.

to be male, and one of the robots was assigned to be female. The game master robot was also assigned to be female to keep the genders balanced during the interaction. The three robots were on the table around the participant, and each had a tablet in front of it to play the game with the participant.

6.3.5 Measures

During the interaction, we collected both behavioral (mainly what participants chose as their final answer during critical and non-critical rounds) and questionnaire data.

Answer Changes

Our primary measurement was whether or not participants changed their answers to the answer of the group of robots in the six critical rounds. We measured how often participants continue conforming to the group in the next critical round, depending on if they got the answer right or wrong when conforming in the previous critical round. We also measured how frequently participants changed their answers to match at least one of the robots in the rounds that are not critical rounds. Additionally, we measured how frequently participants conformed in the round following the critical round. We did this to test whether participants in the selected and blind conditions who did not know they are in the minority in the current critical round would attempt to be similar to the robots in the following round.

Questionnaire

After playing 20 rounds of the game with the robots, the participants completed a post-experiment survey. The survey included the Godspeed questionnaire with questions on the perceived animacy, likeability, and intelligence of the robots [24]. The survey also asked the participants to rate the following questions on a Likert scale from 1 (disagree) to 5 (agree): "I felt pressure to change my answers because of the robots" and "The robots were better at playing the game than me." The last question asked for an open-ended response: "Did you ever change your answer because of the robot and why?".

6.3.6 Participants

A total of 66 participants were recruited, out of which three participants were excluded due to technical problems. Of the remaining 63, 27 were male, and 37 were female, with an average age of 26.3 years old (SD = 8.8). Most of the participants were students from a local university and people from its surrounding community. There were no significant differences in age and gender between conditions. Participants were randomly assigned to conditions: 15 participants were in the staring condition (6 male and 9 female), 17 participants (6 male and 11 female) were in the quantitative condition, 16 participants (9 male and 7 female) were in the selected condition, and 15 participants were in the blind condition (6 male and 9 female). The University Institutional Review Board approved this study. Participants signed a consent form agreeing to participate in the study and received five dollars compensation for

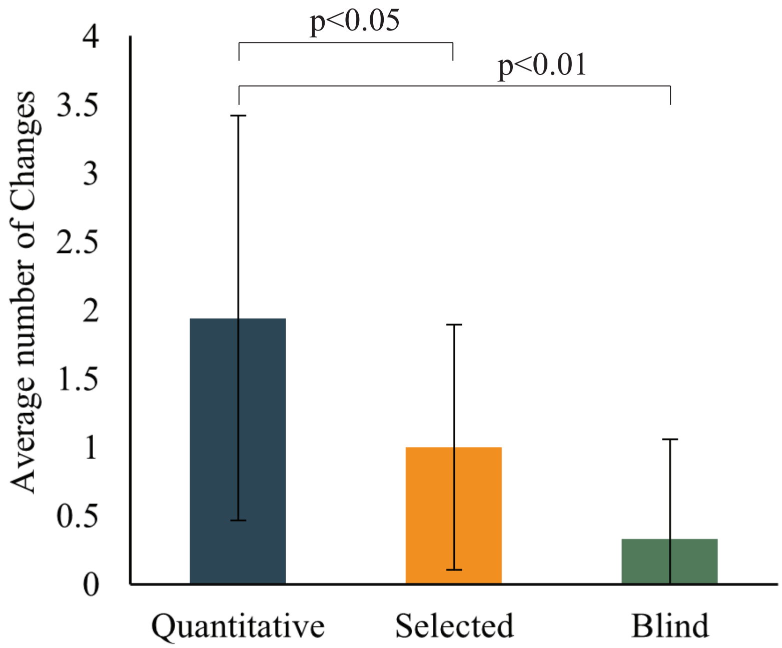


Figure 6.8: (a) - Participants were significantly more likely to conform to the robots' answers when they were aware they were a minority compared to when they only knew at least one robot had chosen a different answer than them. They were also significantly more likely to conform than the participants who had no information about the robots' answers.

their time. The game with the robots and the questionnaire took approximately 30 minutes.

6.4 Results

In this section, the findings on the conformity rates for the different conditions and post-experiment questionnaire results are presented.

6.4.1 Behavioral Results

First, we tested whether there was a difference in the conditions for the number of times people changed their answers to be the same as at least one of the robots throughout all the rounds. A logistic regression was conducted with condition (quan-

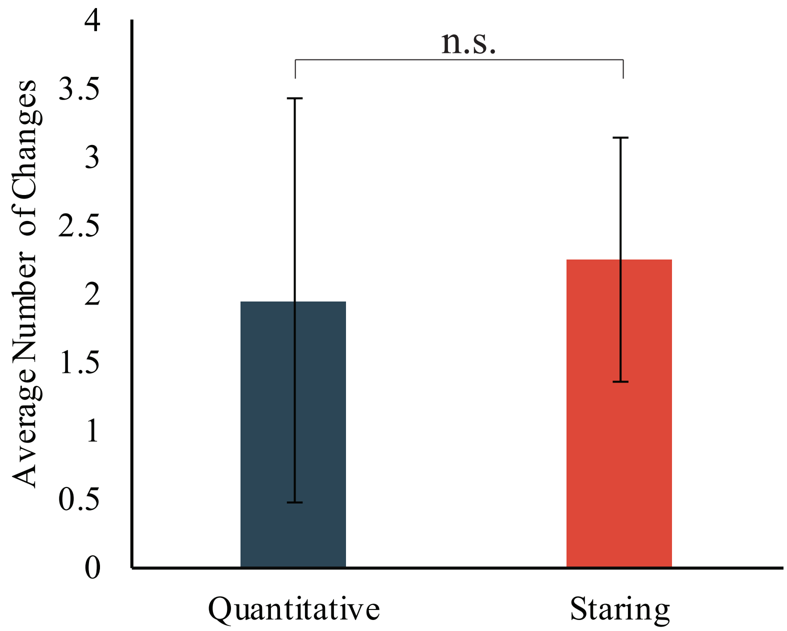


Figure 6.9: (b) Adding the staring behavior to the quantitative condition did not significantly increase conformity in the critical rounds.

titative+staring†, selected and blind) and whether it was a critical round as independent variables; the independent variable was whether they conformed during that round. The logistic regression showed that there was an effect on the amount of information about the robots' answers shown to the participants ($logodds : 0.61, SE : 0.16, Z : 3.84, p < .001$). It also showed that there was a significant difference for the conditions in the conformity rate depending on if it was a critical trial or a neutral trial ($logodds : 0.60, SE : 0.16, Z : 3.84, p < .001$), indicating that people were changing their answers more frequently to match the robots in the critical trials than in the neutral trials. Lastly, a logistic regression with staring as an independent variable shows that the staring behavior had no significant influence on the interaction ($logodds : 0.19, SE : 0.20, Z : 0.95, p = 0.340$).

Since there was a significant difference in the conditions on the conformity rate during the critical trials, we further investigated how the varying amounts of information

†quantitative+staring encompassed both the pure quantitative and the staring conditions as they both provide the same amount of information of the robots answers

given to the participants influenced their conformity rates. On average, participants conformed to the robots less in the blind condition ($M = 0.33, SD = 0.72$) and the selected condition ($M = 1.00, SD = 0.89$) compared to the quantitative condition ($M = 1.94, SD = 1.48$). An ANOVA test with the conditions as the independent variable and number of changes as the dependent variable showed that there was a significant difference in conformity between conditions during the critical rounds [$F(2, 48) = 8.70, p < 0.001$]. A post-hoc Tukey HSD test showed that there was a significant difference between the quantitative and selected conditions ($p = 0.046$), a significant difference between the selected and blind conditions ($p = 0.001$), but no significant difference between the selected and blind conditions ($p = 0.221$). These results showed that participants conformed significantly more to the robots when they were aware they were a minority in their answer. These results are presented in Figure 6.8. In the quantitative conditions, the robot would either stare at the participant or not when their answers differed. However, the staring behavior did not cause a significant increase in frequency of conformity in the critical trials [$F(1, 30) = 0.22, p = 0.640$]. These results are presented in Figure 6.9.

Participants changed their answers (on critical and non-critical rounds) to match at least one robot a similar number of times on average in the three conditions where information was provided: staring ($M = 4.2, SD = 2.018$), quantitative ($M = 3.53, SD = 2.29$), and selected ($M = 3.94, SD = 3.21$). However, in the staring and quantitative condition, participants were making a large number of these changes in critical rounds (52.8% and 55.0%, respectively), whereas most of the changes in the selected condition were not in critical rounds (25.40% in critical rounds). Using Chi-Squared with Bonferroni corrections, we compared the number of critical and non-critical round changes in the three conditions: the difference was significant between the staring and selected conditions X^2 $(2, N = 126) = 9.6512, p = 0.006$, the difference was also significant between the quantitative and the selected conditions

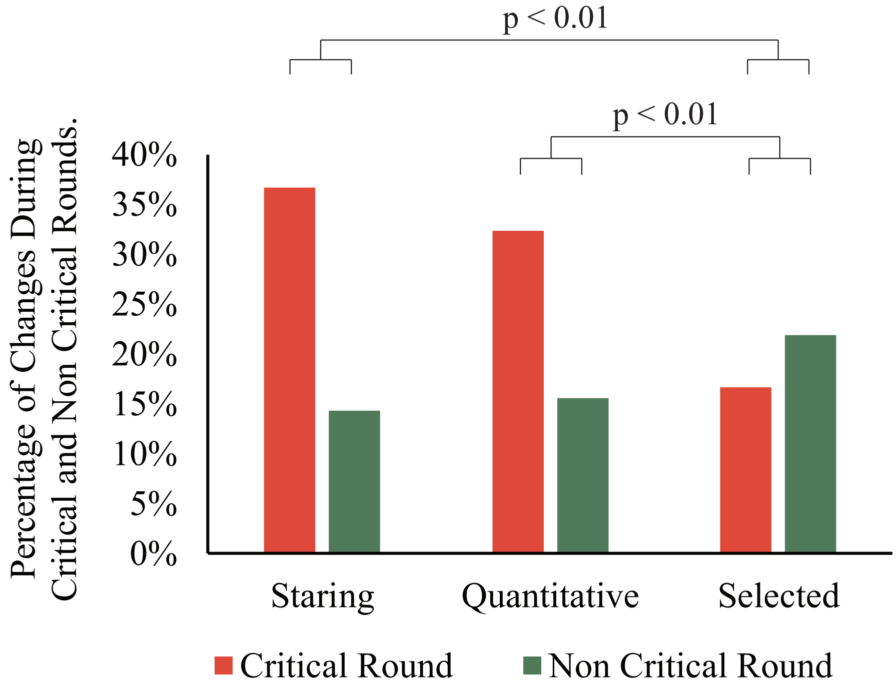


Figure 6.10: (a) - Despite participants having changed their answers a similar number of times to match the robots in the three conditions with information, participants in the selected condition significantly changed their answers more in non-critical rounds than in critical rounds compared to the staring and quantitative conditions.

X^2 $(2, N = 123) = 11.24, p = 0.002$. There was no significant difference between the staring and quantitative conditions. These results are presented in Figure 6.10.

Further investigation of which rounds participants in the selected and blind conditions changed their answers to match at least one of the robots showed that they were frequently changing their answers in the rounds immediately after the critical rounds [‡]. A logistic regression showed that there was a difference in conformity between the round immediately after the critical round and the remaining neutral rounds ($logodds : 0.50428, SE : 0.17, Z : 2.90, p = 0.004$). Participants in the selected and blind conditions were frequently making changes in the round immediately after the critical round, compared to the quantitative and staring conditions: 34.4% of the

[‡]One subsequent round was excluded because it was both a critical round and the round after the critical round.

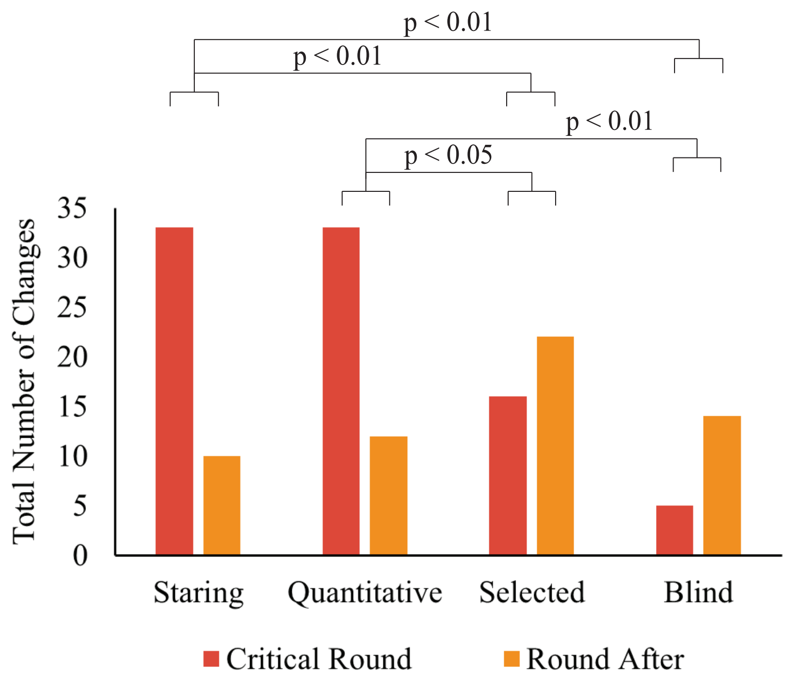


Figure 6.11: (b) - Participants in the selected and blind conditions significantly more frequently changed their answers in the round right after the critical round than in the critical rounds, compared to the staring and quantitative conditions.

time in the selected condition, 23.3% of the time in the blind condition, 19.1% in the quantitative condition, and 16.7% in the staring condition. Performing a chi-squared analysis with Bonferroni corrections showed that participants were significantly more likely to change their answer in the critical round than in the round immediately after the critical round in the quantitative condition compared to the selected condition X^2 $(2, N = 83) = 8.31, p = 0.024$ and compared to the blind condition X^2 $(2, N = 64) = 12.24, p = 0.003$. They were also significantly more likely to change in the critical round rather than in the round after the critical round in the staring condition compared to the selected condition X^2 $(2, N = 81) = 10.13, p = 0.009$, and the blind condition X^2 $(2, N = 62) = 14.12, p = 0.001$. In summary, participants in the blind and selected conditions were frequently changing their answers to match at least one of the robots in the round immediately after the critical round, while in the quantitative and staring conditions, they were more frequently changing their

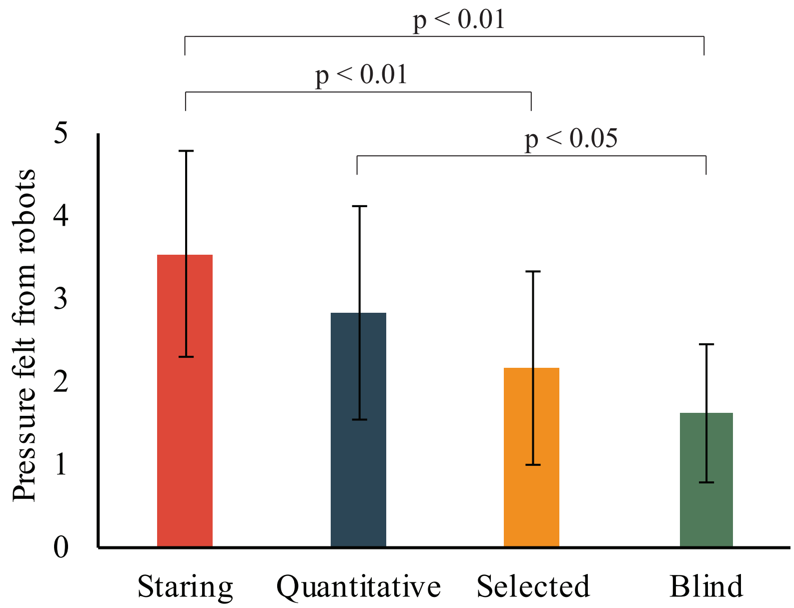


Figure 6.12: (a) - Participants felt significantly more pressure to change their answers because of the robots in the Staring conditions than they did in the Selected and Blind conditions. Additionally, participants in the quantitative condition felt more pressure than the blind condition.

answers in the critical round. These results can be seen in Figure 6.11. Whether the robot achieved the right or wrong answer did not appear to play a significant role in the participant deciding to change their answer in following rounds (12 and 10 times respectively after observing the robots' answers were right and wrong in the selected condition, and 8 and 6 in the blind condition).

Another element to consider is how the round number influenced the number of times participants changed their answers. There was not a significant correlation between the round number and the number of changes for all rounds ($R = -0.02, N = 20, p = .93$). Neither was there a significant correlation between the round number and the number of changes during critical rounds ($R = -0.48, N = 6, p = .34$). Therefore, the round's number did not play a strong role in the number of changes during the interaction.

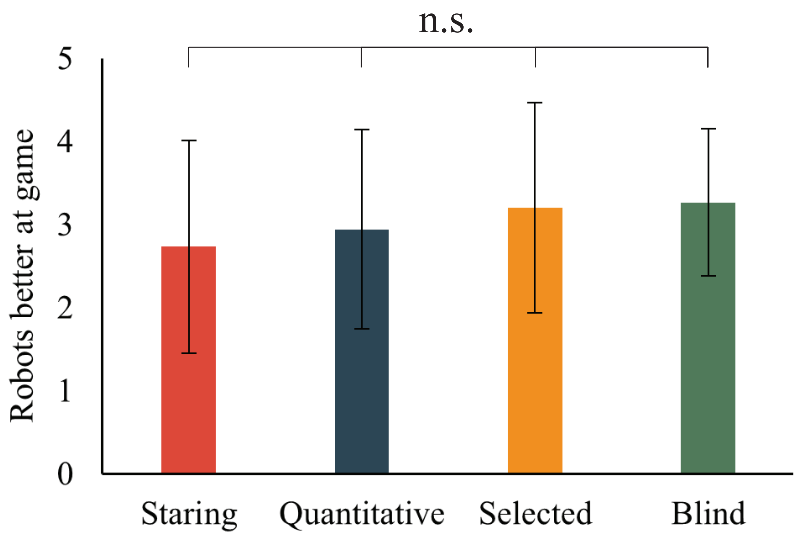


Figure 6.13: (b) - Participants in all four conditions viewed robots similarly in terms of if they were better at the game than them.

6.4.2 Questionnaire Results

In the post-experimental questionnaire, participants reported on a scale from 1-5 on how much pressure they felt to change their answers because of the robots. In the staring condition, participants reported feeling the highest amount of pressure to change ($M = 3.53, SD = 1.25$), followed by participants in the quantitative condition ($M = 2.82, SD = 1.29$), then participants in the selected condition ($M = 2.16, SD = 1.18$), and lastly participants in the blind condition ($M = 1.6, SD = 0.83$). Performing an ANOVA with condition as the independent variable showed that there were differences between conditions in the amount of pressure they felt from the robots [$F(4, 63) = 7.94, p < 0.001$]. Post-hoc Tukey tests showed that participants in the staring conditions felt a significantly higher amount of pressure to change compared to the blind ($p = 0.001$) and selected conditions ($p = 0.008$). Participants in the quantitative condition also felt a significantly higher amount of pressure compared to the blind condition ($p = 0.021$). There were no significant differences between the remaining conditions. These results are presented in Figure 6.12. There was a correlation in the staring condition and the quantitative condition between reporting pressure to change and critical round changes. Using Pearson correlation, in the staring condition there was a moderately positive correlation between critical round changes and pressure to change ($R = 0.52, N = 15, p = 0.048$), and also in the quantitative condition there was a positive correlation ($R = 0.65, N = 17, p = 0.005$). There were no significant correlations between feeling pressure and critical round changes for the selected and blind conditions.

Lastly, participants were asked whether the robots were better at playing the game than them on a 5-point Likert scale. On average, participants across the three conditions responded similarly that the robots were on par as them at the game (staring: $M = 2.73, SD = 1.28$; quantitative: $M = 2.94, SD = 1.20$; selected : $M = 3.2, SD = 1.26$; blind: $M = 3.27, SD = 0.88$). An ANOVA showed no differences between all four conditions in how participants rated the robots in if they were better at the game than them [$F(4, 63) = 0.81, p = 0.495$]. These results are shown in Figure 6.13.

Although prior work has shown females to conform more than males [76, 35], in our study, there were no significant differences in gender for the number of times participants conformed. Additionally, prior work has shown that people are more influenced by those they perceive as more likeable and intelligent [53]; however, in our study, there were no significant correlations between conformity and the perceived animacy, intelligence, or likeability of the robots.

6.5 Discussion

In this section we discuss how robots can cause people to conform to them, and the different types of conformity that are involved. We also discuss how different tasks affect the conformity rates in human-robot interaction. Last we discuss some potential future directions.

6.5.1 People Conform to Robots

The main metric was whether participants conformed to the consensus of the group of robots during the six critical rounds. On average participants conformed more frequently to the answers of the robots in the staring and quantitative conditions than in the selected and blind conditions. In these results we see strong indications that people do conform their answers to match those of the robots in the critical rounds. They change their answers significantly more when knowing the robots' answers, than when they do not know the answers. Asch's participants conformed on average in 37% of the rounds, and participants in our study conform on average in 32% (quantitative) and 38% (staring) of the rounds. Therefore we have evidence to support that **Hypothesis 1** is true: When provided with the opportunity to conform without providing an obviously false answer, participants will conform with a group of robots at a rate similar to the original Asch participants.

6.5.2 Informational Conformity

We believe robots were causing informational conformity due to three main reasons. First, when participants were given more information about the robots' answers, they conformed significantly more. Second, participants viewed the robots as being capable of this task, which is an element that enables informational conformity. And third, the subjective nature of the task increases the willingness to accept information from the robots, which was confirmed by many participants in the open-ended question.

To measure whether robots were causing informational conformity, the current analysis focused on the quantitative, selected, and blind conditions, where the amount of information given to the participant varied. Participants conformed significantly more in the quantitative conditions than they did in the blind and selected conditions.

Even though participants in the selected condition did conform a higher number of times on average than the blind condition, this difference was not significant, implying that being aware that at least one robot chose a different answer was not sufficient to sway the participant to change their answer. However, when the participant was aware that all the robots chose a different answer than they did, they more frequently conformed to the robots. The results suggest that having the information that one was in the minority in their answer increased the likelihood of accepting information from the majority as they had more information from the environment to make decisions [252].

One of the factors that is believed to influence informational conformity is the expertise of the group [263, 53]. Participants viewed the robots as performing well at this particular task: in the questionnaire, participants rated the robots as being similar to themselves at how good the robots were at the game. These results are surprising, considering robots do not usually perform well at high-level tasks such as understanding the meanings of images. Additionally, participants gave similar scores to the robots' capabilities across the conditions, indicating that they are not viewing the robots as better at the game in one condition compared to another. However, participants conformed more in the quantitative condition compared to the selected and blind conditions. Once participants were given more information (such as how many robots chose each answer), they utilized this information and conformed to the robots. This indicated that participants believed the robots to have the correct answer but only had sufficient information to conform in the conditions with more information.

One of the main factors that influences informational conformity is uncertainty in the answer [54]. Individuals are more likely to copy others when they are uncertain [251]. Therefore it was likely that the subjective nature of the game increased the participants' likelihood of accepting information from the robots. Our results are in line with how participants responded to the open-ended question, where they frequently state they were using the robots' answers to decide their own final answer. For example, one participant in the quantitative condition wrote: "Yes, from life experiences, majority is usually correct." Another participant in the quantitative condition wrote: "Yes, because I thought the way they decided was going to be right." There were also participants in the selected condition who responded that they changed their answer due to information of the robots' answers: "when they highlighted a different option, and then I felt that it was more apt than the one I chose."

Therefore we believe **Hypothesis 2a** is true: Participants conform to the group of robots due to informational conformity.

6.5.3 Normative Conformity

Our results show that normative conformity was playing a role in the participants' decisions to conform to the robots in the staring and quantitative conditions: participants conformed significantly more in critical rounds in the staring and quantitative conditions compared to the other rounds. Participants reported feeling pressure to change because of the robots and acted upon it. Additionally, participants in the selected and blind conditions that did not have the information to conform during critical rounds changed their answers frequently to the answer of at least one of the robots in the next round.

To measure whether robots were causing normative conformity, this analysis was focused on the comparison of the staring, quantitative, and selected conditions (as in the blind condition, participants had no information on the robots' answers, and therefore normative conformity was highly unlikely). In the quantitative condition, there were significantly more changes in participant answers during critical rounds than in the selected condition, demonstrating that being aware of the number of robots choosing certain answers influenced participant's decisions to conform. Therefore being aware that one was in the minority in a group of robots increased the likelihood of conforming to them, compared to only knowing that at least one robot chose a different answer.

Participants in the staring, quantitative, and selected conditions were making, on average, a similar number of changes across all the rounds (critical and non-critical). However, participants in the staring and quantitative conditions were making many more of these changes in critical rounds. Providing participants with the information of how many robots chose each card did not increase overall changes but specifically increased the number of changes to the robots in the rounds where they were the minority. This is in line with previous research showing that having a unanimous group increases normative conformity [15].

There were no significant differences in the frequency of conformity between the staring and quantitative conditions. Therefore adding the staring behavior did not significantly increase conformity. There are multiple possible interpretations of this. The first being that participants did not feel additional peer pressure because of the staring behavior either because it was not very observable or because they did not perceive it as a persuasive behavior. Another interpretation is that the quantitative behavior alone was already causing a large amount of peer pressure, and adding the staring behavior did not increase the frequency of conformity significantly. A previous study has shown that eye contact can actually create resistance to the person who is trying to persuade [48]; therefore, the staring behavior might have caused some participants to conform less. An additional possibility is that the staring behavior is causing psychological reactance in some of the participants towards the robots. Studies have shown that very apparent persuasive behaviors can decrease the amount of compliance [98]. Additional studies should be conducted to determine which social behaviors of robots cause increased peer pressure.

Participants in the staring and quantitative conditions were making most of their changes in critical rounds, whereas participants in the selected and blind conditions were frequently changing their answers in the round right after the critical round. Participants in the selected and blind conditions did not have the necessary information to see they were a minority in the critical rounds in time for them to change their answers. However, when the final answer was shown, the participants observed that all the robots had chosen a different answer than they did. We believe this caused the participants in these conditions to change their answers in the following round, attempting to choose the same answer as the robots. Additionally, the robots getting the answer right or wrong in the critical rounds did not appear to play a role in deciding to change their answer in the subsequent round. Therefore the main reason they were changing their answer was not necessarily because they thought accuracy would be increased. Instead, we believe this was an indication of normative conformity where participants wanted to be in-group with the robots.

Participants in the staring and quantitative conditions reported higher pressure to change their answers because of the robots than the selected and blind conditions. Additionally, the amount of pressure to change was correlated with the number of critical round changes. This was an indication of normative conformity, where participants were feeling pressure to change and acting upon that pressure. Participants in the staring and quantitative conditions also mentioned feeling peer pressure in the open-ended question. Several participants commented that they changed their answers to match the robots' answers when they were part of the minority, indicating that participants were changing to become part of the majority. For example, one participant in the staring condition wrote: "Yes, because they'd look at me judgmentally when I had a different answer, so it made me doubt myself." Another participant in the staring condition commented: "Sometimes when they all chose the rose field, I felt dumb for picking the ballet shoes." A participant in the quantitative condition wrote: "Yes (I changed) if they outnumbered me on one particular picture."

Therefore we believe **Hypothesis 2b** is true: Participants conform to the group of robots due to normative conformity.

6.5.4 Influence of Task in Conformity

Several studies have been conducted attempting conformity with robots, of which some observed conformity [223, 262, 114] and some did not [37, 28, 234]. The main difference between the experiments which observed conformity and those which did not was the task being tested. It is necessary to have a task where the participant is not certain of the correct answer. The robotic studies which failed to show conformity mostly tested Asch's line task, which has a clear, correct answer. Whereas the studies which did show conformity with robots had a task in which the answer was not as clear. Our study used a subjective word-card matching task. Hertz and Weise presented the questions to the participants for only 2.5 seconds, and the accuracy rate of responding solely was 63% for the analytical task and 68% for the social task [114]. And Williams et al. tasks used socio-conventional and moral questions [262]. Therefore we believe it is necessary to have a more subjective task to cause conformity. This is in line with human psychology, where more difficult and subjective tasks have higher rates of conformity [23].

Other characteristics of our task that could have influenced the number of times participants conformed were that they were all sat at the same table close together [150], the answers were publicly shown on a shared screen [70]. Additionally, the answers of the robots were highlighted on the screen, focusing the participants' attention on those answers, which could have influenced conformity [53].

6.5.5 Future Work

There are several different areas of potential future work, following our results. The robots used in the study were very simple, but despite their size and simplicity, they caused both informational and normative conformity. Factors that increases informational conformity are the similarity with the group [54] and the expertise of the group [263]. Future studies should analyze whether having robots with increased similarity to humans or with higher appearance of capabilities will also lead to more informational conformity.

Factors that increase normative conformity are group size, the immediacy of the group, and their social importance [150]. Future studies could analyze how changing the perceived social importance of the group, changing the number of robots, and changing how close the robots are will influence the frequency of normative conformity. Another factor that influences normative conformity is whether the other members are considered in-group or out-group [3]. Several studies have shown that in-group robots are rated more anthropomorphic and are favored over out-group members [84, 83]. Therefore it should be studied how group membership and anthropomorphism influence conformity.

Our results indicate that conformity is directly linked with the type of task being tested. Future work could analyze how conformity changes depending on the type of task. And to further investigate if conformity to robots can be used in pro-social ways [60]. Lastly, conformity is influenced by individual characteristics [251]. Culture [35], age [63], gender [76, 35], and other personal factors have been shown to influence the decision to conform in human groups. More studies on different personal factors should be studied to see how they influence conformity to robot groups.

6.6 Summary

In this chapter, we showed that robots cause people to conform to them in a subjective task. Participants played a card game with three robots in which they were given varying amounts about the robot's answers. Additionally, in one condition, the robots stared at them to cause peer pressure. Informational conformity was shown to be at play because participants conformed more when they were given more information about the robots' answers, and they considered the robots on par with them at the game. Normative conformity was shown to be at play because participants conformed significantly more when they were aware they were a minority and because they reported feeling pressure by the robots in the questionnaire. We conclude that adults conform to robots due to both informational conformity and normative conformity.

Informational influence plays an essential role during robotic tutoring sessions, as users need to be willing to accept and believe the information provided by the robot. If the user does not trust the robot to be providing the correct information, they will not trust it to learn from them. Normative influence can also play an important role as the robot might need to influence the user to study the material or to pay attention during the lesson. Normative influence can be especially significant if a robot needs to create long-term behavior change.

In the following two chapters, we build in-home robotic systems that promote behavior change in their users over several weeks. In Chapter 7, we investigate how a robot influences the motivation and exercise accuracy of a user that does daily exercises with it. In Chapter 8 we show how a robot can encourage children with Autism Spectrum Disorder to learn several positive social skills by exemplifying different emotions and demonstrating joint attention.

Chapter 7

Long-term In-Home Robots: Robotic Coach to Guide Users While Doing Dumbbell Exercises*

In this chapter, we present our first robotic system that shapes people's interactions in the home. There are several challenges in having a robot that operates in the home compared to a structured lab setting. The first is that it needs to be able to model a wide range of different and often complex tasks. The system needs to model these tasks without making significant changes to the environment as that would be an imposition on the user. A second challenge is that the user needs to want to interact with such a system on a long-term base. Therefore, we need to give the robot a personality that maintains engagement even after the novelty effect wears off. Lastly, it needs to be given the capabilities to influence the user to complete daily exercises with it. And to be persuasive for the user to correct any mistakes they perform while exercising.

Parts of this chapter were originally published as: Nicole Salomons, Tom Wallenstein*, Debasmita Ghose*, Brian Scassellati. (2022). The Impact of an In-Home Co-Located Robotic Coach in Helping People Make Fewer Exercise Mistakes. In *International Conference on Robot and Human Interactive Communication* [224].

We present a robotic system that motivates people to complete daily weight training with it. We created a computer vision system that tracks users while they exercise and classifies in real-time whether each repetition of an exercise was performed correctly. The robot provides personalized feedback when the user makes mistakes while exercising. We designed the robot to take on the role of a peer while exercising, giving it the backstory that it wanted to be "big and strong" just like its older brother. The backstory was designed to be engaging and interactive. This study focused on whether a robot would be more effective at correcting exercise mistakes when being physically present or whether it would be just as effective when on a tablet screen.

Other domains have shown several advantages in having a robot co-located with its user compared to having it on a screen. Some benefits of co-located robots include higher learning gains [158] and increased user motivation [132]. This study investigates whether a physically co-located robot generates fewer exercise mistakes. We also assess the different perceptions users have of different robot embodiments. Participants (n=25) had a robotic system in their homes for two weeks and were asked to exercise with the robot daily for around 20 minutes. Our results show that participants who exercised with the co-located robot made fewer mistakes than those who exercised with the video-displayed robot. Furthermore, participants in the robot condition reported a higher fitness increase and were more motivated to exercise than participants in the tablet condition.

7.1 Introduction

An large number of studies have outlined the benefits of exercise [215, 216] including improved cardio-respiratory fitness [184], improved mental health [168], and prevention of lifestyle diseases such as diabetes [52] and hypertension [260]. Despite the many benefits of exercising, there are risks associated with performing exercises with the wrong body posture or movement patterns [253, 266]. Studies have shown that weight-training-related exercises, when performed without supervision and corrective feedback from a trained professional, can put one at risk of sustaining musculoskeletal injuries [209]. Therefore, many choose to exercise with a personal trainer, who helps them learn correct techniques and tailors their regime according to their body type and strength levels, thus mitigating the risks associated with performing weight-training exercises without supervision [81].

However, in the COVID-19 pandemic, when access to gyms and personal trainers became limited, many people resorted to exercising at home with limited access to equipment and exercise partners [75, 183]. Additionally, as the pandemic accelerated the growth of virtual fitness programs, videos created by trainers from across the world gained popularity among a broader audience. When people started learning to exercise with trainers virtually, they risked sustaining exercise-related injuries as they received little to no feedback on their posture or technique while learning new exercises.

One potential enhancement to virtual fitness training systems could be robotic coaches. Social robots have been shown to be effective in providing corrective feedback to help people learn a task effectively while motivating them to complete tasks [172, 17]. Therefore, we designed a physically present social robot system that helps people adhere to an exercise routine while providing corrective feedback on their form. We deployed this system in homes over a 14-day study while providing corrective feedback using a machine learning algorithm. We compare our co-located robotic coach to a video of the same robot displayed on a tablet screen. To the best of our knowledge, this is the first in-home study to analyze the effects of a robot's physical presence in helping people maintain correct exercising techniques.

Our results show that participants in the robot condition made significantly fewer mistakes while exercising than participants in the tablet condition. Additionally, our

Figure 7.1: Participants completed dumbbell exercises with a robotic coach during a two week in-home study

questionnaire results show that participants in the robot condition reported finding the workouts less difficult and reported a higher fitness increase than tablet condition participants. Finally, participants in the robot condition felt more motivated to exercise and found the system more entertaining.

7.2 Related Work

This section provides a literature overview on the efficacy of robots in exercise training and the impact of physically present robots in helping people learn a task correctly.

7.2.1 Use of Robots in Physical Exercise Training

There have been a growing number of studies that show robots as personal coaches [17, 141], including assisting during repetitive, self-directed exercises in rehabilitative therapies [108, 89]. Other studies have deployed robot systems to engage the elderly in physical exercise [91, 86].

Several studies show that a robot can effectively help people learn correct movement patterns and exercise postures by providing real-time corrective feedback to the participant [17, 107]. However, most studies in this domain asked the participants to perform simple, rehabilitative, or injury-preventive exercises without external weights. Furthermore, previous robotic coaches were mainly conducted in a controlled lab setting during one session. For robots to be effective coaches, they need to operate in unstructured environments for more extended periods and provide guidance during mainstream fitness exercises. Thus, we explore the impact of an in-home co-located robot in helping people practice weight-training exercises done typically at the gym.

7.2.2 Benefits of Physically Present Robots

A robot being physically located with the participant has many advantages. For example, research has shown that a physically present robot led to greater compliance [20] than a robot on a screen in a book moving task. Participants that interacted with an in-person robot had higher learning gains than a robot that was on a video [158] in a tutoring task. Studies have also shown that corrective feedback provided by co-located robots is more effective in helping people learn a given task correctly over video-displayed robots [258, 244].

This work studies if a physically present robot will also show benefits when acting like an exercise coach. Specifically, we aim to investigate if the feedback provided by a co-located robotic coach would impact the number of exercise mistakes people would make as compared to people exercising with videos of a robot on a tablet.

(a) (b)

Figure 7.2: (a) The co-located robot as part of the system. (b) The robot displayed on a tablet screen

7.3 Methodology

In our study, participants interacted with a system in their homes over 14 days and the participants were expected to do one coaching session every day. Each coaching session took between 15-25 minutes, to complete five upper or lower body exercises. We compared a co-located robot (*Robot Condition*) to the video of same robot displayed on a tablet screen (*Tablet Condition*).

We had three hypotheses based on prior work showing that physically present robots can lead to higher learning gains [158], and maybe seen as more perceptive and helpful [258], and can be more motivating [132] over a video-displayed robot.

Hypothesis 1: *Participants will perform fewer exercise mistakes with the co-located robot than with the video-displayed robot.*

Hypothesis 2: *Participants will perceive the co-located robot as smarter and more helpful than the video-displayed robot.*

Figure 7.3: Participants were shown how to complete each exercise correctly via demonstrations of an expert on the tablet.

Hypothesis 3: *Participants will be more motivated to exercise with the co-located robot than the video-displayed robot.*

7.3.1 Conditions

We compared our exercise system with a physically present robot which we call the **Robot Condition**, to a similar system which displays a video of the same robot on a tablet, which we call the **Tablet Condition**. The video-displayed robot was recorded using a high-resolution camera. The physically present robot and the video-displayed robot appeared to be of similar size and performed the same exercise movements and utterances across the two conditions. The robots in the two conditions are shown in Figure 7.2.

7.3.2 Exercises

A professional coach helped design the exercise routine, including a good sequence of exercises, the number of repetitions, appropriate break times and helped identify

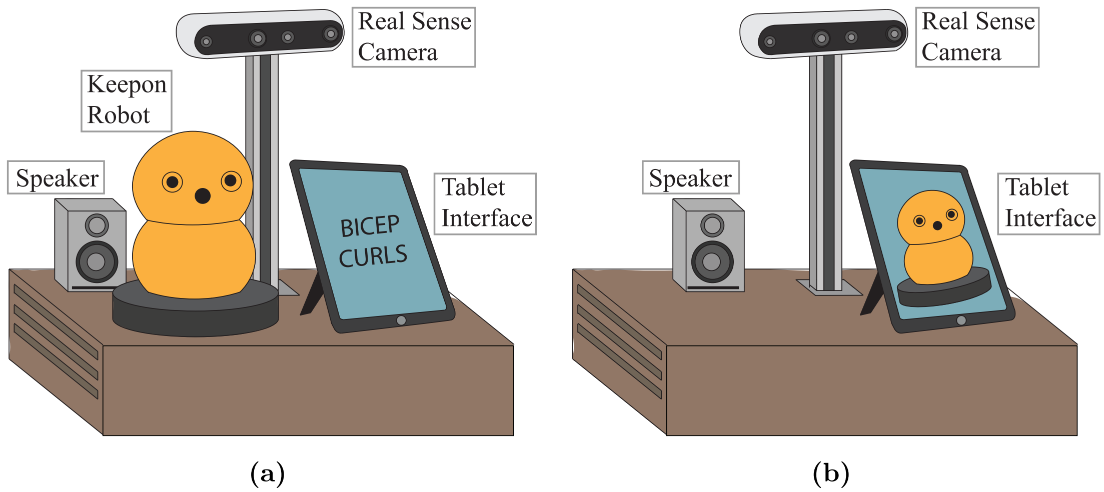


Figure 7.4: a) System for the *Robot Condition* was composed of a Keepon Robot, a speaker, a RealSense camera, and a tablet interface. b) System for the *Tablet Condition* was composed of a tablet interface, a speaker, and a RealSense camera

common mistakes. These are seen in Figure 7.5. The number of common mistakes varied between the different exercises. We classified two mistakes for bicep curls front raises and one mistake for shoulder presses, single-arm triceps extensions, squats, and lunges. Calf raises, and single-leg raises had no possible mistakes because of the simplicity of these exercises. Over a 14-day study period, the system guided participants through upper-body exercises on odd days and lower-body exercises on even days. An example of a demonstration of bicep curls given to participants via video can be seen in Figure 7.3. For each exercise, the participants could choose to use an appropriate dumbbell set provided weighing between 2-10 lbs or perform the exercises without weights. The entire interaction was expected to take 15-25 minutes each day.

7.3.3 System Design

We used the MyKeepon robot, a 4-DOF, 15cm tall yellow-colored desk robot derived from a commercialized robot called the KeeponPro [142], as seen in Figure 7.2a. The robot does not have limbs, but it can move up and down, side to side, or front and back during the exercise to give the participant the appropriate speed of each repetition. The MyKeepon robot was chosen due to its low cost and small form factor, making it an appropriate robot coach in people's homes. Also, there might be benefits of having a robot that is not perfectly human-like. When interacting with an anthropomorphic robot, users might assume the robot can perfectly demonstrate an exercise, which it often can't due to joint constraints. Therefore, it might induce mistakes if the user tries to replicate its movement.

To deploy the robotic system in participants' homes in the *Robot Condition*, we built a compact 16in x 12in junction box to house a mini-computer, a router, and some support equipment as shown in Figure 7.4a. Outside the box, we had the MyKeepon robot, a 12-inch tablet for the participant to interact with the robot, an Intel RealSense camera [131] to track the participant's pose while exercising, and an external speaker. The setup for the participants in the *Tablet Condition* was identical except that the video of the robot was shown on the tablet instead of the robot being physically present as shown in Figure 7.4b. Additionally, in the Tablet Condition, the same tablet was used to display the video of the human trainer demonstrating the exercises and the video of the robot exercising. Notably, when the video of the human trainer was displayed for demonstrating an exercise, the video-displayed robot was not shown to the participants.

7.3.4 Procedure

After delivering the system to the participants' home, they filled out a consent form and a pre-study demographics questionnaire. The participants were asked to interact with the system every day for 14 days. Each day a given participant experienced the following interaction sequence:

1. The participant would turn on the system and start an application on the tablet.

Day	Exercise Name	Mistakes	Reps	Mistake Description	Classifier	Validation Accuracy
Upper Body	Bicep Curls	Arm Swinging	12	Arm swings instead of moving only lower arm and keeping upper arm attached to body	SVC	83.7%
		Arm Half-Down		Movement doesn't cover the full range of the arm's motion		
	Front Raises	Arm Swinging	12	Arm swings to use momentum instead of controlled movement	FNN	86.2%
		Arm Above Shoulder		Weights raised above the shoulder		
	Shoulder Presses	Elbows Out	12	Elbows point outward instead of keeping them at shoulder-width	KNN	92.3%
	Single-Arm Tricep Extensions (Right/Left)	Elbows Out	10	Elbows point outward not upward, upper arm far away from head	KNN	right: 91.6% left: 87.8%
Lower Body	Squats	Knees Unstable	15	Knees move around or point inward while going down	FNN	98.1%
	Lunges	Knees Unstable	10	Struggles to keep balance, knee moves around while stepping back	KNN	73.5%
	Calf Raises		12			
	Single-Leg Raises (Right/Left)		20s			

Figure 7.5: Participants completed upper body exercises on odd days and lower body exercises on even days. Upper body days had several different mistakes that were classified using our computer vision system. Whereas some lower body days had mistakes, and other ones did not. Each exercise had an appropriate number of repetitions that were completed according to the advice of professional coaches. Additionally we present which machine learning classifier was used to classify each exercise, and their accuracy on the validation set.

2. The robot would introduce itself on the first day and explain how the interaction is expected to proceed briefly. Each subsequent day, the robot would begin with a 1-2 minute long motivational greeting.

3. The robot would then guide the person to position themselves at an appropriate distance from the camera with the help of prompts on the tablet.

4. The robot would guide the participant through two sets of exercises, each set containing five exercises. For each of the exercises:

   (a) A video of a human trainer performing the exercise was shown on the tablet for 15 seconds.

   (b) After the participant is prompted to begin exercising, the robot would either move up and down, side to side, or front and back indicating the primary body movement in a given exercise.

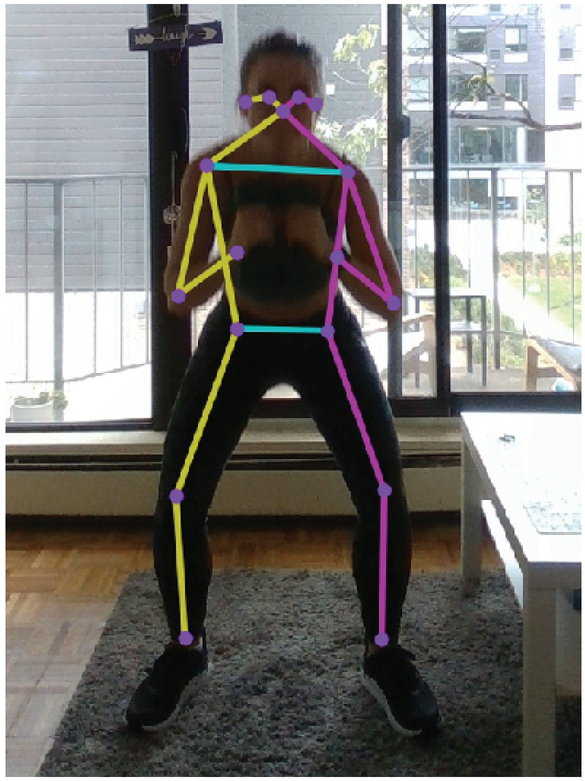


Figure 7.6: an example of keypoints predicted by the pre-trained MoveNet [1] model on the images captured by the Intel Real Sense Camera.

(c) The robot would then instruct the participant to perform the same exercise following the pace of its movements.

(d) The robot would perform a short celebratory dance after the participant completed an exercise, and that would be followed by a minute-long rest period.

5. The robot would bid goodbye for the day and shut off the system automatically.

7.3.5 Mistake Correction

During the interaction, the images captured by the camera were used to track the pose of the participant. We used MoveNet [1] based on TensorFlow.js's [240] pose detection API. An example of our system doing body tracking for squats can be seen in Figure 7.6. Given a two-dimensional image, we run inference on a pre-trained

MoveNet model to predict 17 keypoints on the human body with high accuracy in real-time. These keypoints were used in two ways:

Evaluating the participant's position with respect to the camera

To evaluate if the person's positioning was valid, we observed whether all 17 keypoints were present in the camera's field of view with high confidence. If the participant's position was not valid, they were asked to adjust their position appropriately until it was valid.

Evaluating the participant's form while exercising

Participants were given corrective feedback on their form during most exercises. We designed an algorithm that separated the participants' movements into repetitions and classified those repetitions according to the pre-defined mistakes in real-time. Given the predicted keypoints, the algorithm used the following steps to identify the appropriate feedback:

Preprocessing: We normalized the predicted keypoints to reduce the dependence of the analysis on the position and person's height with respect to the camera. First, we translated each of the keypoints with respect to the center of the body, which we defined to be the middle point of the quadrangle formed by the shoulder and hip keypoints. Then, we divided the translated keypoint positions by the body's torso length (distance between the shoulder and hip keypoints) to account for different people's heights. Afterward, we used a Kalman filter [32] to smooth out jitters between predicted keypoints for different frames of the video.

Repetition Detection: To identify when a person had completed a valid repetition of an exercise, we calculated the minima and maxima for each movement by analyzing in real-time the trajectory of a chosen keypoint whose value changed prominently along the y-axis during a single repetition. We focused on the value of the wrist keypoint for all upper body exercises and the nose keypoint for all lower body exercises. For example, in a correct bicep curl movement, the y-values of the wrist in the trajectory must first strictly increase, then strictly decrease in the y-dimension. Therefore, a bicep-curl repetition is considered valid when a minimum is followed by a maximum and then another minimum.

Mistake Classification: After a valid repetition has been detected, we use machine learning classifiers to detect if it was performed correctly. We detect mistakes for seven of the ten exercises. For each, we trained a different machine learning classifier. We collected data from twelve people under the supervision of a trainer. We asked each person to perform 10-15 repetitions of each exercise correctly and an additional 10-15 repetitions purposefully performing the pre-defined exercise mistakes. We experimented with three different classifiers for each exercise: support vector classifier (SVC) [182], k-nearest neighbor time series classifier (KNN) with $k = 5$ [249], and a feedforward neural network (FNN) [243]. The FNN consisted of two hidden layers, with 64 and 32 neurons respectively, used a sliding window approach with a window size of $n = 15$, and was trained using cross-entropy loss. The chosen classifier for each exercise and its performance for leave-one-subject-out cross-validation on the training data is reported in Figure 7.5.

Providing Feedback: If the participant performed a given exercise correctly for at least three repetitions, the robot would provide a motivating utterance like "*Keep going! You are doing well!*". However, if a specific exercise mistake is detected twice or more per set, the robot would provide a corrective utterance to the participant. An example utterance for the arm swinging mistake during biceps curls was "*Don't swing your arms so much! Keep your upper arm attached to the sides of your body*".

7.3.6 Measures

We collected both behavioral and questionnaire measures. **Behavioral Measures:** Our behavioral measures included the percentage of days the participant exercised with the robot and the percentage of exercises performed correctly. To measure correct exercise execution, three people coded the first two days and the last two days the system was in each person's home. One person coded front raises and shoulder presses; one coded right triceps and left triceps; one coded squats and lunges. Bicep curls were not coded due to the difficulty of detecting swinging due to the low video frame rate. The coders were blind to condition. All three coders coded all exercises done by two participants for the first and the last two days of system deployment to measure the coders' agreement with each other. The three coders had moderate agreement (Fleiss' Kappa $= 0.44, p < 0.001$). There were several difficulties in coding, including the variability in camera angles and positions. The percentage of correct exercises is only relating to the six exercises that were coded.

Questionnaire Measures: Our questionnaire measures included a demographics questionnaire asking age, gender; a RoSAS questionnaire [42], assessing robot's perceived warmth, competence, and discomfort; and a post-experiment survey that asked participants to answer the following questions regarding their perceptions of the interaction using a 1-7 Likert responding format: *How difficult did you find the workouts? Do you feel an increase in strength and fitness after the last two weeks? How helpful did you view the instructions the robot gave you for doing the exercises? How helpful did you view the exercise corrections the robot gave you while doing the exercises? How important were the workouts in your daily routine? Did the robot motivate you to work out? Do you feel more motivated to continue exercising on a regular basis after the last two weeks? Did you exercise because the robot made you feel guilty if you did not?*

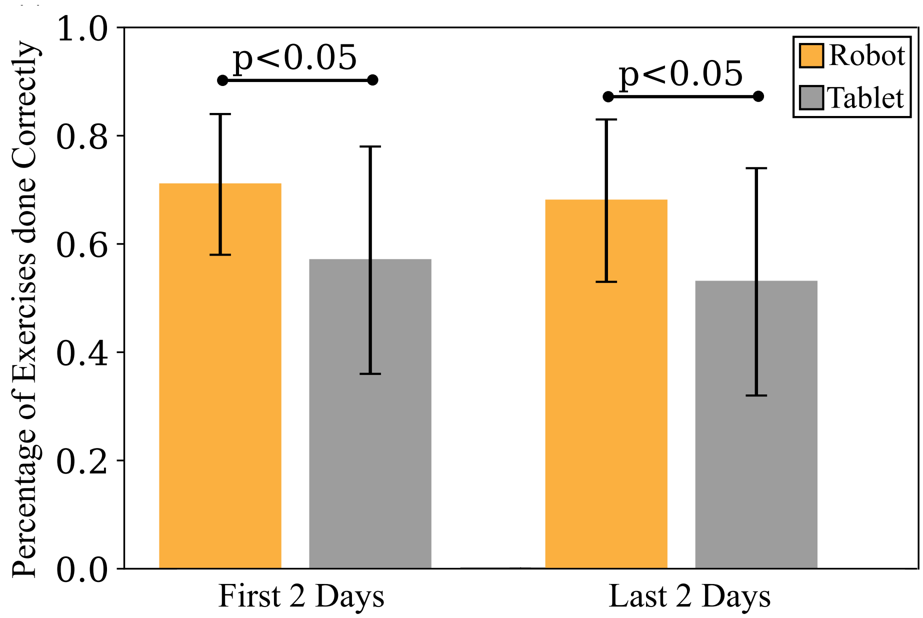


Figure 7.7: Participants in the robot condition on average performed the exercises significantly more correctly than participants in the tablet condition.

7.3.7 Participants

There were 25 total participants in our study. 14 participants were in the robot condition, of which five were male, eight were female, and one was non-binary. Their average age was 21.91 years (SD=2.84). 11 participants were in the tablet condition, of which four were male, and seven were female. Their average age was 23.56 years (SD=5.85). There were no significant differences regarding robot familiarity (Robot Condition: M=3.64, SD=1.96; Tablet Condition: M=3.33, SD=1.00; p=.340).

7.4 Results

We first present our behavioral results, followed by our post-experiment questionnaire results.

7.4.1 Behavioral Results

Participants in the robot condition completed on average 68.65% (SD = 11.41%) of the coded exercises correctly, while participants in the tablet condition completed 57.46% (SD = 16.78) of the coded exercises correctly. These differences were statistically significant $t(25) = 1.98, p = .03$. On the first two days participants in the robot condition completed more exercises correctly than the tablet condition (Robot - M:71.25%, SD:12.82%; Tablet - M:56.90%, SD: 20.54%; $t(25) = 2.14, p = .022$). Participants in the robot condition also completed more exercises correctly during the last two days than the tablet condition (Robot - M:68.11%, SD:15.25%; Tablet - M:52.90%, SD: 21.15%; $t(25) = 2.05, p = .026$). These results are presented in Figure 7.7.

On average, participants in the robot condition exercised 71.74% ($SD = 19.79$) days out of the days the system was in their home. Participants in the tablet conditions exercised on average 66.46% ($SD = 23.48$) days. These differences were not significant using a t-test $t(25) = 0.61, p = 0.273$.

7.4.2 Post-Experiment Questionnaire

Regarding the RoSAS questionnaire, there were no significant differences in warmth (Robot: $M = 4.56, SD = 0.98$; Tablet: $M = 3.82, SD = 1.38; t(25) = 1.58, p = 0.064$), competence (Robot: $M = 4.73, SD = 1.26$; Tablet: $M = 3.98, SD = 1.36; t(25) = 1.41, p = 0.086$), or discomfort (Robot: $M = 1.71, SD = 0.51$; Tablet: $M = 2.17, SD = 1.30; t(25) = -1.19, p = 0.123$).

On the post-experiment questionnaire, participants in the robot condition found the exercises less difficult (Robot: $M = 2.43$, $SD = 1.02$; Tablet: $M = 3.27$, $SD = 1.42$; p=0.048) and felt a larger fitness increase (Robot: $M = 4.86$, $SD = 0.86$; Tablet: $M = 3.91$, $SD = 1.51$; p=0.030) than participants in the tablet condition. There were no significant differences between conditions in how helpful they found the instructions (Robot: $M = 5.57$, $SD = 1.40$; Tablet: $M = 4.73$, $SD = 1.79$; p=0.099) or the corrections (Robot: $M = 3.93$, $SD = 1.90$; Tablet: $M = 3.64$, $SD = 1.86$; p=0.352) given by the robot.

Participants felt more motivated to workout in the robot condition than the tablet conditions (Robot: $M = 4.64$, $SD = 1.65$; Tablet: $M = 3.27$, $SD = 2.05$; p=0.038), but there were no significant differences in motivation to continue exercising post-experiment (Robot: $M = 5.57$, $SD = 1.40$; Tablet: $M = 4.73$, $SD = 1.79$; p=0.099). Participants in the robot condition reported seeing the workouts as more important in their daily routine (Robot: $M = 3.93$, $SD = 1.21$; Tablet: $M = 2.55$, $SD = 1.29$; p=0.006), and that they felt guiltier when they did not (Robot: $M = 5.21$, $SD = 2.02$; Tablet: $M = 2.73$, $SD = 1.49$; p=0.039).

7.5 Discussion

We first discuss whether our results confirm our different hypotheses. In sequence we discuss the impact of robot embodiment on exercising mistakes.

7.5.1 Hypotheses

Participants in the robot condition made fewer mistakes overall than participants in the tablet condition. Furthermore, participants in the robot condition made fewer mistakes in the first two days and the last two days than in the tablet condition. The in-home systems were not deployed for a long-enough duration to see any reductions in the number of mistakes people made in either condition over the study period. Both conditions were consistent in the number of mistakes shown across the two weeks. On the questionnaire, Robot Condition participants reported finding the workouts less difficult and felt a higher strength and fitness increase. These results support Hypothesis 1: *Participants performed fewer exercise mistakes with the co-located robot* *than with the video-displayed robot.*

There were no significant differences regarding competence on the ROSaS questionnaire. Participants also did not report any significant differences regarding the helpfulness of the instructions or the exercise corrections. Therefore we do not believe Hypothesis 2 to be true: *Participants* ***did not*** *perceive the co-located robot as smarter and more helpful than the video-displayed robot.*

There were no significant differences between conditions regarding the percentage of days the participants exercised with the robot. However, participants in the robot condition did report feeling more motivated to work out, placed more importance on exercising with the robot, and felt guiltier when they did not. Therefore we have partial support for Hypothesis 3: *Participants felt more motivated to exercise with the co-located robot than the video-displayed robot.*

7.5.2 Impact of Robot Co-Location on Exercising Mistakes

Having a co-located robot significantly reduced the number of mistakes people made while exercising. On average, participants in the tablet condition performed 43% of their exercises incorrectly. This means, if a person exercising with a co-located robot performed two days of exercises (one lower-body day and one upper-body day) and did two sets each day, they would have performed 61 more incorrect repetitions as compared to a person in the tablet condition. Performing this large amount of incorrect repetitions could lead to injuries and sub-optimal strength improvements. This was confirmed by questionnaire results where participants in the tablet condition found the exercises more difficult and felt lower fitness and strength increase than participants in the robot condition.

There are multiple reasons why a person would have performed fewer mistakes with the co-located robot. One possibility is that they felt more engaged and entertained by the physically present robot and therefore were paying more attention to the exercise demonstrations and corrections. Literature also shows that physically present robots increase learning gains [158], and therefore people might have learned more from the robot's corrections. Lastly, research shows that co-located robots cause higher amounts of compliance [20]. Thus, participants in the robot condition could have been more willing to receive corrections from the robot.

One potential confound of the study is that participants in the robot condition had more time with the robot, as in the tablet condition, the robot was shortly not visible during the exercising demonstrations. However, we do not believe this significantly impacted the study, as the demonstrations were short and the robot was mostly static during them. A second potential confound is that the movements of the robot representing repetition speed might have been more visible in 3D rather than in 2D on the tablet screen. However, this would have a minimal effect as most exercise speed movements were from left to right or up and down which were equally visible in both conditions.

7.6 Summary

This chapter demonstrated how a low-cost peer robot can create behavior change in users during a two-week-long study. We investigated the effects of robot embodiment on exercising mistakes and motivation to exercise. Our results show that people make fewer exercising mistakes both at the beginning of the two weeks and at the end of the two weeks when interacting with an in-person robot versus a robot on a tablet screen. Participants also reported feeling more motivated to exercise with the co-located robot.

This study highlights the benefits of having a low-cost co-located robot present when exercising. Even if the robot cannot demonstrate the exercises themselves - due to the lack of a humanoid form - the presence alone had people making fewer mistakes while exercising. Reducing errors increases exercise gains and reduces the potential for injuries. Throughout this study, we have also demonstrated how a peer robot can positively influence people's daily habits and shows the promise for such systems in the future.

Chapter 8

Long-term In-Home Robots: Socially Assistive Robotics for Children with Autism Spectrum Disorder*

This chapter presents our second long-term in-home deployment of a robotic system. We created an intervention system for children with autism spectrum disorder (ASD). Although numerous prior studies have investigated robots for children with ASD, most of them have used short, isolated encounters in controlled laboratory settings. Our study focused on a 1-month, home-based intervention for increasing social communication skills of 12 children with ASD between 6 and 12 years old, using an autonomous social robot. The children engaged in a triadic interaction with a caregiver and the robot for 30 minutes every day to complete activities on emotional storytelling, perspective-taking, and sequencing. The robot encouraged engagement, adapted the difficulty of the activities to the child's past performance, and modeled positive social skills. The system maintained engagement over the 1-month deploy-

*Parts of this chapter were originally published as: Brian Scassellati, Laura Boccanfuso, Chien-Ming Huang, Marilena Mademtzi, Meiying Qin, Nicole Salomons, Pamela Ventola, Frederick Shic. (2018). Improving Social Skills in Children with ASD Using a Long-Term, In-Home Social Robot. In *Science Robotics* [229].

ment, and children showed improvement on joint attention skills with adults when not in the presence of the robot. These results were also consistent with caregiver questionnaires; caregivers reported less prompting over time and overall increased communication

8.1 Introduction

Autism spectrum disorder (ASD) is a neurodevelopmental condition characterized by social interaction and communication deficits and the presence of restricted, repetitive patterns of behavior [77]. Children and adults with ASD often have difficulty in responding to social overtures, recognizing the emotional states of others from visual or auditory cues, and understanding the importance of gaze as a social cue [124]. Therapies are diverse, but are typically time-, resource- and labor-intensive, and can put significant strain on families and caregivers [167].

Technology-based interventions for ASD, and robotics in particular, have been seen as a potential approach for augmenting the efforts of families and clinicians to provide on-demand, personalized, social skills training [227]. The robots envisioned by these efforts are part of a new field called socially assistive robotics (SAR), which aims to construct systems that support social and cognitive growth using social rather than physical means [88, 169, 248]. These robots share characteristics of educational robots, which attempt to convey information typically via a tutor-student relationship [31], and rehabilitation robots, which provide structured physical therapy for deficits such as stroke and paralysis [250].

Exploratory studies from dozens of research groups have shown that many individuals with ASD enjoy interacting with robots and in many cases even demonstrate more appropriate social behaviors with robots than they do with peers or caregivers [228, 134]. These initial exploratory studies focused on short interactions, spanning

Figure 8.1: Robotic system. The robot provided personalized social skills training to children with ASD over 30 days.

tens of minutes or less, under controlled laboratory or clinic conditions, often involving sample sizes of five children or fewer, and focused exclusively on robot-directed behavior [248]. While these studies generated considerable excitement, they held little clinical value. Results tended to fade with repeated exposures and may have been the result of novelty, appropriate control conditions were rarely considered, and experiments failed to demonstrate learning that generalized to human-directed actions [72]. While a few studies did examine longer-term interactions [259], or demonstrated improved adult-directed social behavior [135], none were able to demonstrate skill acquisition that could be considered clinically meaningful that generalized beyond the specific robot encounter.

We report here a demonstration of directly assessed improvements in social skills in children with ASD after an in-home, 1-month intervention in which daily social skills games were conducted by an autonomous, socially assistive robot (Figure 8.1). This study differs from previous work in this domain in four important aspects. First, this study used a fully autonomous robot system that operates for a 1-month deployment duration with no adjustments made by clinical or research staff. Many socially assistive robots still operate under teleoperative control, because autonomous operation for this duration is a substantial challenge in the robotics community even when static program requirements are used throughout the deployment [169]. Second, unlike previous work where predefined protocols are followed explicitly [269], the system used here must adapt to the strengths and weaknesses of the individual child by changing the difficulty of individual tasks based on the child's preferences and performance. Because individuals with ASD have substantial individual differences in the type and severity of their social skill deficits, the need to adapt to an individual child is essential to enabling a positive learning outcome. Further, the interaction between the need for autonomy and the need for adaptation creates additional technical challenges. Third, this study provided therapy directly in homes with a fully autonomous robot. Whereas clinical and laboratory spaces represent known environmental conditions that can be controlled or explicitly planned for, the unconstrained home environment requires more complex sensing and behavioral routines to deal with greater variation in environmental conditions. Last, this study focused primarily on demonstrations of clinically meaningful measures of performance using standard evaluation metrics that are conducted by an independent assessor away from the robot. This represents a challenging evaluation standard, because a child must not only learn a skill while practicing with the robot but also be capable of generalizing that skill to interactions with an adult in an environment that differs from the practice games used by the robot.

8.2 Methodology

Our robot-assisted intervention included a 30-minute session everyday for 30 days and involved triadic interactions among the social robot, the child, and the caregiver, providing opportunities for the child to interact and share experience with the caregiver

Figure 8.2: A typical interaction between the robot, the child, and the caregiver during our deployment. Our robot system was design to engage and facilitate interactions between the child and the caregiver, therefore providing opportunities for the child to practice social skills in a fun, natural way.

(Figure 8.2). The robot modeled social gaze behaviors such as making eye contact (Figure 8.3) and sharing attention throughout the sessions, and provided feedback to and guided the participants in six interactive games. The six games targeted different social skills including social and emotional understanding, perspective taking, and ordering and sequencing (Figure 8.6). Each session began with the robot telling a daily story to engage the participants. The session continued with three games, which varied from day to day, and concluded with a caregiver survey, where the caregivers rated their observations of the child's social communication skills.

8.2.1 Objectives and study design

The objectives of this study were to investigate how a social robot may deliver behavioral intervention to children with ASD both autonomously and effectively outside clinical settings, as well as how such robot-assisted intervention can improve these children's social communicative abilities. This study was modeled after single-subject withdrawal (ABA) designs [41, 241]. This design included pretest (A), test (B), and

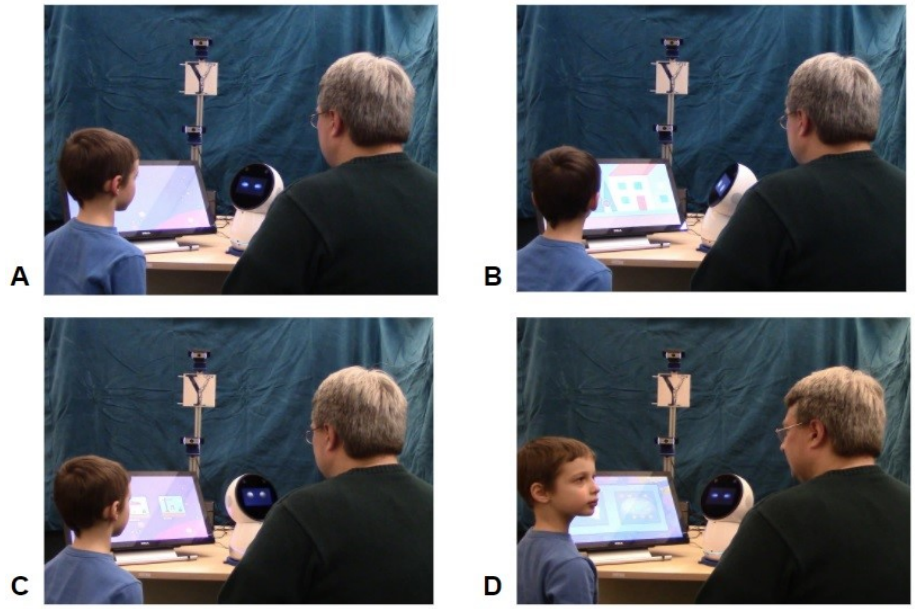


Figure 8.3: Robot-initiated joint attention. The robot models appropriate social gaze behavior by demonstrating context-contingent gaze and facilitates mutual gaze and experience sharing between the child and the caregiver. When the child is engaged with the robot (A), the robot directs the child's attention to relevant task content on the screen (B). As the child's attention shifts to the robot-directed focus on the screen, the robot then attempts to redirect gaze to the caregiver (C) in the hope of redirecting the child's visual attention to the caregiver (D). (These demonstration images were recreated in the laboratory to show both robot and child behavior, as this perspective was not recorded by the deployed system.)

posttest (A) phases, each phase lasting for about 30 days. The pretest phase served as a comparison baseline, capturing possible maturation of social communication abilities and the effectiveness of any other therapies or interventions that the family may have been using. The test phase involved the in-home deployment of a socially assistive robot system that engaged the participating child in our intervention program, which was based on intervention activities commonly used in clinical settings. The posttest phase sought to explore whether the benefits provided by our robot-assisted intervention would be sustained after the removal of the system. This study design is suitable for investigating the effects of a single intervention and for when there are wide variances in participants' characteristics and responses to the intervention. Informed consent from families and assent from minors were obtained in all cases, as approved by the Yale University Institutional Review Board.

8.2.2 Assessment

To assess a child's ability to respond to joint attention bids in their familiar environments, we employed the validated, naturalistic joint attention assessment of Bean and Eigsti [27]. This assessment includes six naturalistic prompts that can be delivered at any point during an interaction with the child and is designed particularly for school-age children and adolescents. The six prompts examine different aspects of joint attention, including gaze following, response to name and a greeting opportunity, and recognition of the other person's current interest. This assessment of joint attention was administered four times throughout the study while a researcher was interacting with a child in play-based activities.

To understand whether or not a child's behaviors of social communication changed over time outside intervention sessions, we asked the child's caregiver to fill out a survey regarding his/her own observations of the child's communicative behaviors at the end of each daily session. The survey questions sought to measure the broader influence of our robot-assisted intervention outside of intervention sessions, focusing on the child's ability to make eye contact with, initiate communication with, and respond to communication bids from the caregiver and others.

8.2.3 Robot-Assisted Intervention System

Our intervention system consisted of a social robot, a 24-inch touch screen, two external RGB cameras, and two computers (Figure 8.4). The social robot used was an early prototype of the Jibo robot [2]. The Jibo robot is a 12-inch table-top robot with three degrees of freedom, capable of turning its head and body around 360 degrees. The robot can exhibit expressive behaviors through body movements, a ring of color-changing LED lights, and a pair of animated eyes (e.g., blinking and dilation).

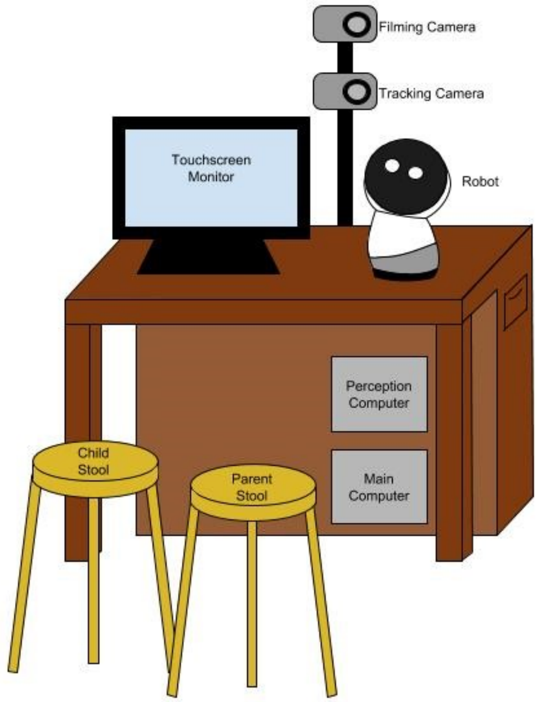


Figure 8.4: Robot-assisted intervention system. Our system consists of a social robot, touchscreen monitor, and two RGB cameras. The system supports triadic interactions between the robot, the child, and the caregiver. Software running on the perception computer uses an elevated camera to track both the child's and caregiver's attentional foci, while the other camera records the intervention session (Fig. 2). The main computer controls the flow of the intervention as well as the robot's behavior to ensure presentation of coherent, meaningful intervention.

These capabilities allow the robot to make eye contact with the participants and signal shared attention. Additionally, the robot can deliver information verbally to the participants through its internal speakers. The 24-inch touch screen presented educational content and served as a shared medium where the robot and the participants could all interact with and reference to. One of the cameras tracked both the child's and the caregiver's attentional foci as approximated by head orientations, while the other camera recorded the intervention session.

Our software system (Figure 8.5), which involved attention tracking and intervention presentation, was implemented in the Robot Operating System (ROS) framework [196]. The attention tracking subsystem, running on one of the computers, continuously approximated users' attentional targets in the environment. Using RGB camera

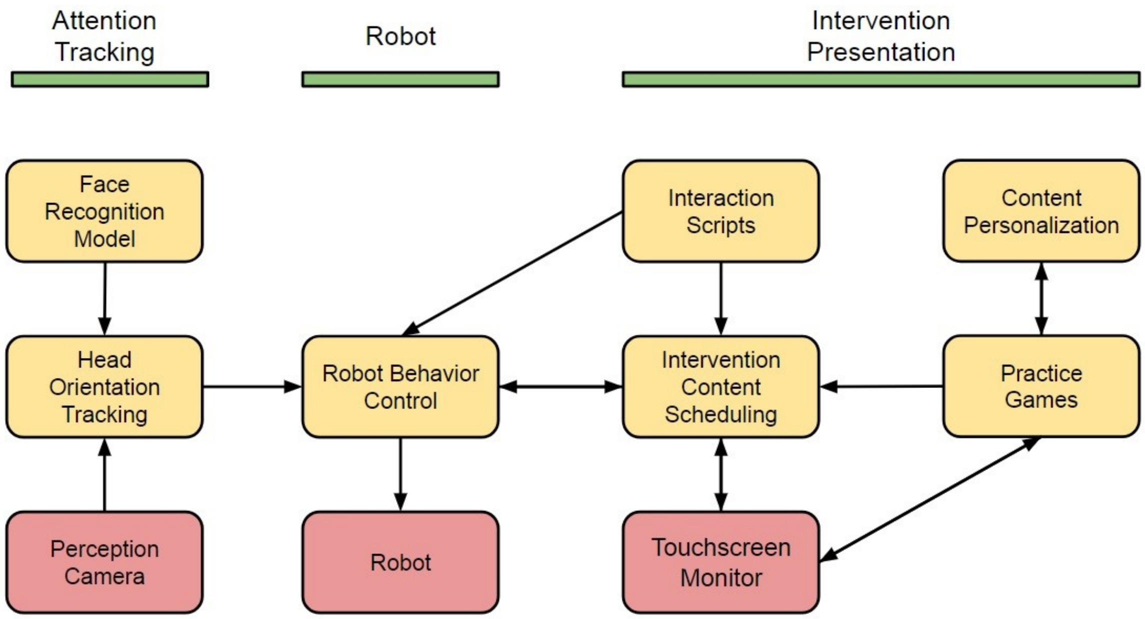


Figure 8.5: Diagram of Software Components. Our software system consists of several components responsible for attention tracking of the participants, robot behavior control, and intervention presentation. These components together create rich, engaging interactions for our robot-assisted autism therapy. These components operate within the ROS framework.

stream input, the system estimated and tracked head poses and orientations using Constrained-Local-Model (CLM) face tracking and landmark detection algorithms [21], and approximated attentional targets according to the estimated head poses and orientations. The intervention presentation subsystem, running on the other computer, ensured smooth delivery of curricular content. It controlled the robot's behaviors, scheduled intervention content, and adjusted difficulty levels of the social skills games.

In our implementation, we manually prepared interaction scripts that specified predefined behavioral animations for the robot, daily opening stories, and verbal encouragement and feedback to the participants. When the robot was not displaying pre-specified behaviors in a prepared interaction script, it maintained eye contact with the child to show engagement. The robot also shared attention with the child by looking toward the visual content on the screen from time to time throughout each session. These behaviors of making eye contact and sharing visual attention were meant to model social gaze behaviors for the child.

8.2.4 Interactive Games

In addition to targeting the core social skill of joint attention, we designed and developed six interactive games that provide opportunities for the child to practice social and emotional understanding, perspective taking, and ordering and sequencing while interacting with the robot and the caregiver (Figure 8.6). The six games are an emotional understanding game ("Story"), two barrier games that facilitate perspective taking ("House" and "Rocket"), and three ordering and sequencing games ("Train," "Spaceship," and "Traveler"). Each game involves multiple levels of difficulty, ranging from 1-4 to 1-8, except for "Spaceship" and "Traveler" which had only one difficulty setting. Depending on the child's performance in the game, the system adjusts the difficulty level accordingly. As inspired by the challenge point theory [106], our personalization module was focused on delivering learning contents with appropriate levels of difficulty to increase learning gains while reducing frustration. The personalization module kept track of the child's performance in game activities, providing approximate measures of their abilities of social and emotional understanding, perspective-taking, and ordering and sequencing. Using these performance measures, the module followed a simple decision tree mechanism to decide the difficulty level of the game for the next round of interaction. In our implementation, we used 25% and 75% as criteria for decreasing and increasing a difficulty level, respectively. Similar approaches of discrete adaptation have been used in robot-child tutoring applications [199, 200, 202].

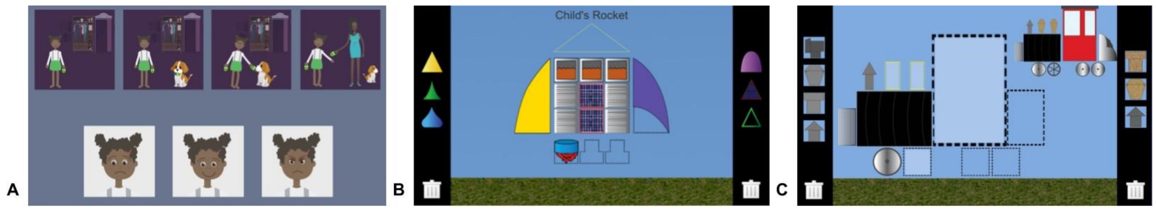


Figure 8.6: Screenshots of social skills games. A set of interactive games were developed to allow children with ASD to practice social skills through play. The games were designed to support interactions between the caregiver and the child as well as between the robot and the child. The games targeted three social skills, including social and emotional understanding (A) ("Story"), perspective taking (B) ("Rocket"), and ordering and sequencing (C) ("Train")

Social and emotional understanding

The Story game targets the skills of social and emotional understanding. A typical example of this game is as follows. The robot provides a social situation, displayed as cartoon-like images on the touchscreen, and asks the child to choose what he/she thinks the story character is feeling at different points in the story by selecting one of multiple options displayed on the screen. As the child progresses, the social stories become longer and more complex. To succeed in this game, the child needs to understand the social situations and emotional states of the characters.

Perspective taking

Two virtual barrier games, Rocket and House, target the ability of taking the other person's perspective on a joint task. Modeled after physical barrier games commonly used in clinical interventions, these games provide spatial information to one of either the child or caregiver and ask them to relay that information to the other verbally. In both games, the robot facilitates interactions between the child and their caregiver and acts as a game moderator by keeping time and providing motivational support. In Rocket, the child and the caregiver take turns building a rocket ship. The first player builds a rocket by dragging modular component parts onto a rocket template while the second player looks away. The screen is then reset to hide this design and the first player must explain to the second player how to recreate the design. If the two designs are identical, the players have succeeded and win the game. Similarly, in House, the child and the robot take turns in the roles of builder and guesser. The builder builds a virtual house that comprises of various designs and materials while the guesser looks away. The game then shows six possible designs, one of which was constructed by the builder. The guesser then asks questions about whether the builder's house has a particular design or material in order to guess which of the shown designs is the builder's. These games provide opportunities for the child not only to understand that the caregiver or the robot has a different perspective but also to practice turn taking and verbal communication.

Ordering and sequencing

The Train sequencing game targets the skills of ordering and sequencing. In this game, the robot instructs the child to build a train by dragging parts onto a template. To succeed in this game, the child needs to follow the robot's instructions carefully in sequence. Two additional games, Spaceship and Traveler, involve various tasks such as sorting objects in order. In an ordering task, the child needs to place objects in the right order to complete the task successfully.

8.2.5 Participants Information

Fourteen families with a child with ASD enrolled in this study. Two families withdrew, one due to unrelated health problems of a caregiver and one due to technical difficulties with the robot installation. Among the 12 families who finished the study, 5 of the children with ASD were females and 7 were males. These participants' age ranged from 6 years to 12 years old ($M = 9.02, SD = 1.41$). All had nonverbal IQ scores $\geq$ 70 as determined by the Differential Ability Scales [79] ($M = 94.17, SD = 20.06$). Diagnosis of ASD was based on standard-in-field clinical best-estimate (CBE) diagnosis by licensed clinical psychologists and/or speech-language pathologists with extensive experience in autism diagnosis. Measures used in the diagnostic process included the Autism Diagnostic Interview - Revised (ADI–R; [164]) caregiver interview and the Autism Diagnostic Observation Schedule (ADOS; [163]) semi-structured play observation.Scores on the ADI-R and the ADOS reflect the presence of autism symptoms, with higher scores reflecting greater autism severity. The ADI-R is broken down into 4 domains: reciprocal social interactions (M = 17.64, SD = 6.98; cutoff for ASD = 10); communication ($M = 16.36, SD = 4.74$; cutoff for ASD = 8); restricted, repetitive, and stereotyped behaviors ($M = 6.00, SD = 1.41$; cutoff for ASD = 3); and history of early abnormal development ($M = 3.44, SD = .73$; cutoff =1). The ADOS yields outputs including a calibrated severity score ($M = 7.08, SD = 2.02$; scale from 1 to 10, cutoff for ASD = 4). All participants, in addition to receiving a CBE of ASD, scored above the ASD-cutoff on either the ADOS or the ADI-R.

All participants were recruited from a large database of children with ASD who have either participated in previous research studies with our laboratory or expressed interest in participation. Eligible families were contacted via email to inquire about their interest in participating. Given the scope of the project, the first 12 eligible families were enrolled. Inclusionary criteria were (i) age between 4 and 12 years old, (ii) good medical health, (iii) cooperative with testing, (iv) English is a language spoken in the family, and (v) having been diagnosed with ASD and meet the characterization cutoffs described above. Exclusionary criteria were (i) a fragile health status and (ii) suspected or diagnosed hearing loss or visual impairment or diagnosed neurological abnormality significantly affecting visual or auditory acuity.

All children in the study were enrolled in school programming full time and received intensive special education services as consistent with the state standards for educating children with ASD. Because the scope and form of these services and therapies varied substantially across participants based not only on their individual needs but also on family preferences and local resource availability, we used a single-subject withdrawal design (ABA) that allowed each child to serve as their own control (see Materials and Methods for details). Caregivers were instructed to maintain consistent intervention services during their participation in the study.

8.3 Results

In this section, we first present our results regarding the improvements the children demonstrated in the interactive games and their engagement results. In sequence, we give our joint attention results. Last, we provide our results from the daily caregiver surveys.

8.3.1 Engagement and skills performance

A total of 127 hours of data was collected from the interaction between the 12 children, the robot, and their caregivers. These data included video and audio data, head orientation of both child and caregiver, interaction logs containing the robot utterances and actions, game logs for the tablet-based games, and caregiver survey responses. Because our primary study design was focused on showing the efficacy of this intervention, we focus in this chapter on the analysis of child social performance as measured by game performance, caregiver reports, and clinical measures.

The children combined initiated a total of 653 games with the robot, which resulted in 540 complete games for analysis. (Games that were shortened because of the end of time in the session were not considered for analysis.) On average, each child performed 23.25 sessions with the robot across the month, and each session lasted, on average, for 27 min and 42 s. After a month of interacting with the robot on a daily basis, the robot was able to maintain engagement with the child during the interactions: Children played with the robot for an average of 27 min during the first five sessions

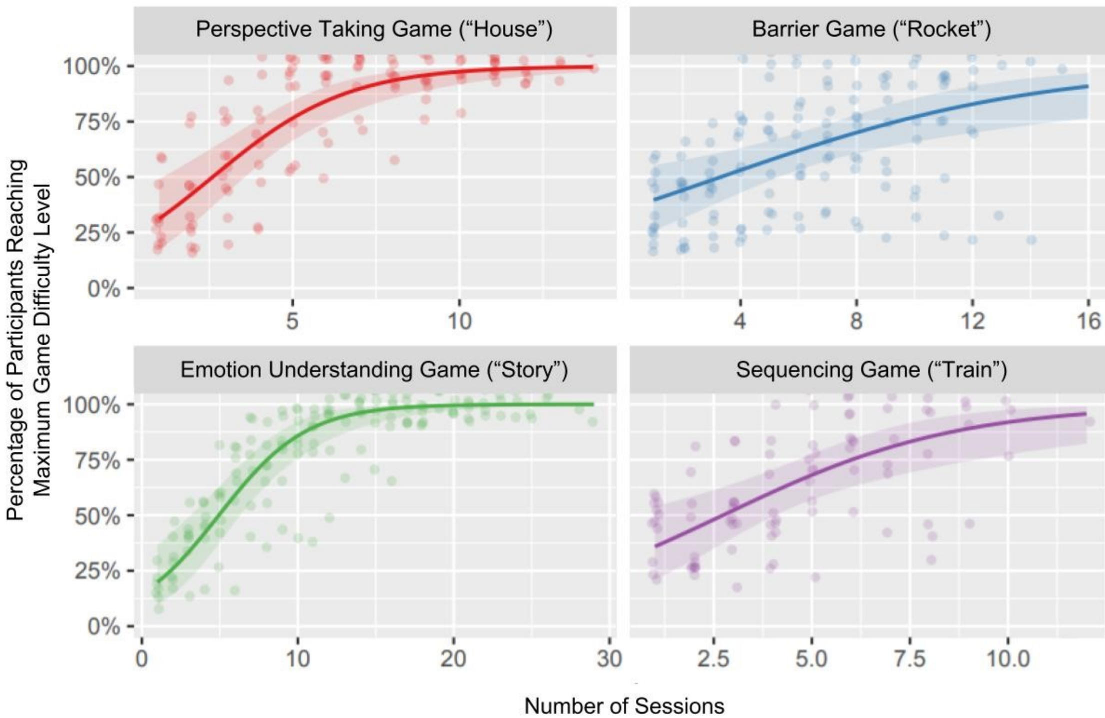


Figure 8.7: Proportion of maximum level achieved as a function of game session. Curves were modeled in a binomial generalized linear mixed model with session and game as fixed and random effects. 95% confidence intervals are shown. Children advanced in the level of each game when they achieved over 75% of correct answers, and regressed a level when giving less than 25% correct answers. When achieving between 25% and 75% of the correct answers, the children would remain at the same level.

and an average of 25 min during the last five sessions.

The robot adapted the difficulty of each individual game based on the child's history of performance in each skill set. On the emotion understanding game "Story", 86% of children reached the most difficult level of the game by the last session. On the perspective taking games, 58% and 92% of children reached the highest level on "Rocket" and "House" respectively. On the sequencing and ordering game "Train", 67% of the children reached the highest level. The "Spaceship" and "Traveler" games used only a single difficulty level and are excluded from this analysis.

Binomial generalized linear mixed models (Figure 8.7) were used to model the level attained by children as a proportion of the maximum possible level as a function of the specific game and session number. Game and session number were included as

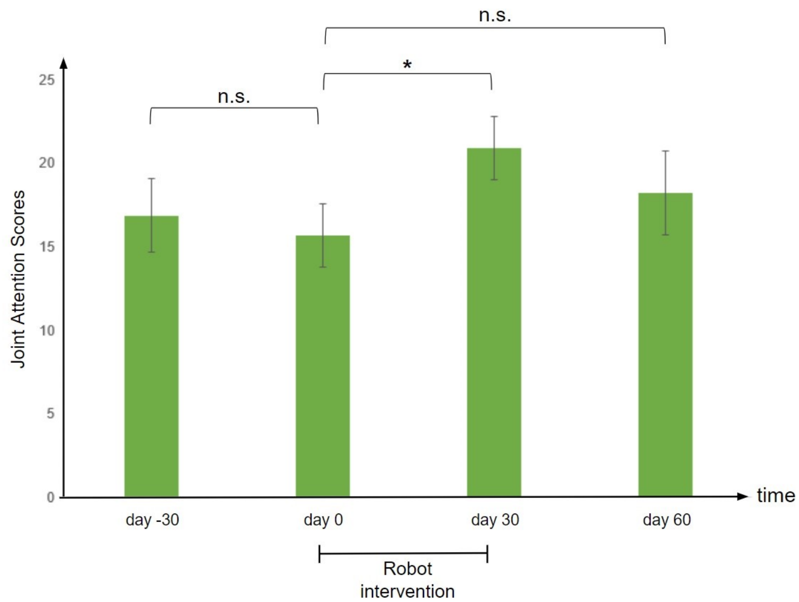


Figure 8.8: Result of joint attention assessment. Probe Scores for the child at four different time points: 30 days before the robot intervention started, on the start day of the robot intervention, on the last day of the robot intervention, and 30 days after the end of the robot intervention. There was a significant increase in joint attention scores when comparing before the robot intervention and after it.

both fixed and random effects. Likelihood ratio tests on the resultant model indicated significant main effects of game, session, and their interaction (all $p < 0.001$). In terms of overall performance (i.e., intercept) and gains over sessions (i.e., slope), the House game was easier than other games [intercept, slope: $p = 0.001, p = 0.030$ (versus Story); $p < 0.001, p < 0.001$ (versus Rocket); $p = 0.014, p = 0.030$ (versus Train)].

8.3.2 Joint Attention

Performance on the joint attention probe was measured and recorded at four time points: (i) T_0, 30 days before intervention began; (ii) T_1, on the first day of robot intervention; (iii) T_2, on the last day of intervention; and (iv) T_3, 30 days after the end of the intervention. The difference between time points T_0 and T_1 was computed to measure change in joint attention during a period of time with no robot intervention and is denoted as the pretest. The difference between time points T_1 and T_2 was calculated to measure joint attention changes resulting from the robot-administered intervention and is denoted as the test phase. Last, the difference between time points T_1 and T_3 was evaluated to measure the stability of any changes recorded during the robot-administered intervention and is denoted as the posttest.

Two participants were excluded for lack of data at one or more time points. Another participant was excluded for being out of the age range in which the task was normed (7 to 12 years of age). Group means were as follows: for T_0, $M = 16.89$ and $SD = 4.46$; for T_1, $M = 15.67$ and $SD = 3.81$; for T_2, $M = 20.89$ and $SD = 3.79$; and for T_3, $M = 18.22$ and $SD = 5.02$. A linear mixed model with compound symmetry repeated covariance effects indicated a significant time point effect $[F(3, 24) = 5.03, p = 0.008]$. Planned comparisons showed that, although no pretest or posttest effect was observed ($T_1 - T_0$, $p = 0.395$; $T_3 - T_1$, $p = 0.083$), joint attention improvements occurred in the test phase ($T_2 - T_1$, $p = 0.001$; Figure 8.8). Test phase changes were negatively associated with nonverbal reasoning performance on the DAS $[r(9) = -0.750, p = 0.020]$. These results are consistent with greater joint attention gains made by children with lower nonverbal ability. Exploration of relationships between baseline nonverbal ability and average baseline (T_0 and T_1) joint attention performance indicated a strong positive relationship $[r(9) = 0.831, p = 0.005]$, suggesting that children with lower nonverbal reasoning skills had more capacity to grow in terms of joint attention skills. Joint attention performance at T_1 was also positively Pearson's correlated with modeled participant overall performance on the House $[r(9) = 0.702, p = 0.035]$ and Story $[r(9) = 0.705, p = 0.034]$ games, suggesting shared variance in performance.

8.3.3 Caregivers Survey

Caregivers completed an on-screen survey immediately after each day's intervention session during the test phase. In all but one family, these interactions were conducted

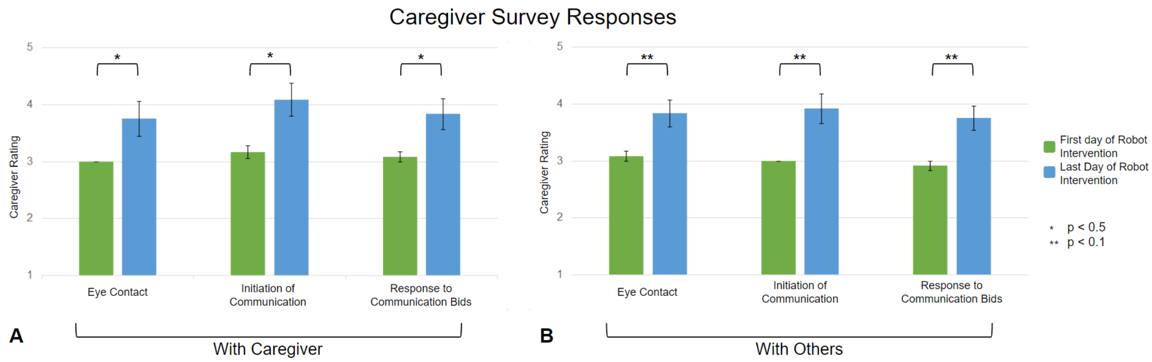


Figure 8.9: Result of caregiver survey. Caregivers reported increased eye contact, increased initiation of communication, and increased response to communication bids with them (A) and with other people (B). Based on comparisons of ratings from the last day of the robot intervention (T2) to the first day of the intervention (T1), these results showed that caregivers were able to observe improved communication abilities of the children beyond our robot-assisted intervention sessions over the period of 30 days.

with the same caregiver (one father, one grandmother, and nine mothers).

The survey consisted of five-point Likert scale ratings. The questions were grouped into two categories: questions on how children interacted with caregivers during the past 24 hours, parallel questions about interactions with other people, and one final question regarding engagement. We compared the ratings scored by the caregivers on the first day and the last day of interventions with paired sample t tests. All 12 caregivers' responses were included in the analysis

Caregivers reported increased social skill performance between their child and themselves, including more eye contact $[t(11) = -2.462, p = 0.03]$ with them on the last day of the intervention $(M = 3.75, SD = 1.06)$ compared with the first day $(M = 3.00, SD = 0.00)$, more attempts to initiate communication $[t(11) = --2.930, p = 0.014]$ with them on the last day $(M = 4.08, SD = 1.00)$ than on the first day $(M = 3.17, SD = 0.39)$, and more frequent responses to communication bids from the caregiver $[t(11) = -3.000, p = 0.012]$ on the last day $(M = 3.83, SD = 0.94)$ than on the first day $(M = 3.08, SD = 0.29)$. These results can be seen in Figure 8.9A.

Caregivers also reported increased social skill performance between their child and other people, including more eye contact $[t(11) = -3.447, p = 0.005]$ with other people on the last day of the intervention $(M = 3.83, SD = 0.83)$ when compared with the first day $(M = 3.08, SD = 0.29)$, more attempts to initiate communication $[t(11) = -3.527, p = 0.005]$ with other people on the last day $(M = 3.91, SD = 0.90)$ than on the first day $(M = 3.00, SD = 0.00)$, and more frequent responses to communication bids from other people $[t(11) = -3.458, p = 0.005]$ on the last day $(M = 3.75, SD = 0.75)$ than on the first day $(M = 2.91, SD = 0.29)$. These results can be seen in Figure 8.9B.

Last, caregivers were asked daily to rate how easy it was to engage their child with the robot therapy session. To confirm that the continued length of engagement was not solely a result of compliance to the protocol instruction, we modeled the engagement rating with a cumulative link mixed model fitted with an adaptive Gauss-Hermite quadrature approximation as a function of day with random participant effects. This model revealed no significant effect of day on engagement ($p = 0.822$). This suggests that participant engagement did not change in a systematic fashion throughout the study.

8.4 Discussion

The potential benefit of a socially assistive robot lies in the ability to provide personalized, on-demand, and structured cognitive or social support to augment the efforts of clinicians, teachers, and families. In the ideal case, robots could provide personalized support, whenever and wherever needed, and could be capable of producing lasting enhancements in social and communicative skills not only in human-robot interactions but also in human-human interactions [227]. The system presented here takes steps in this direction beyond the current state of the art, but also does not yet live up to all of these grand visions. We focus our discussion around the points in which the current work makes substantial improvements and also describe the limitations and areas requiring continued focus as this field progresses.

8.4.1 Autonomous interaction

Our deployed robots operated autonomously without any experimenter intervention for a total of 127 hours over 279 sessions. Caregivers contacted our 24/7 help line a total of eight times: six times for confirmation that they were using the system correctly, which required no action from our team, and two times for a technical issue that was prompted by the sudden disappearance of an online software library, which required a software update and was resolved quickly. Robot assisted autism intervention in previous studies was mostly short episodic interactions that rarely lasted more than 30 min [34, 135] and usually required experimenters to supervise robot-directed actions (although see [143] and [212] for exceptions). Moving from teleoperated to autonomous interactions presents substantial challenges in computational perception and robot control to create meaningful therapeutic training. Although challenging, increasing robot autonomy in assisted therapy has potential to reduce therapists' cognitive load and ensures consistent therapy for the children with autism (21). Our system demonstrated the possibility and potential of autonomous robot interventions for autism, which would enable the implementation and application of robot-assisted intervention at a large scale in various environments, accelerating us toward the goal of achieving clinical significance.

8.4.2 Adaptive intervention

Sustaining engagement with participants is key to effective interventions. Repetitive and unchallenging tasks are likely to bore participants, who then would disengage from the intervention and miss opportunities to practice and improve on targeted skills. As informed by the challenge point theory [106], optimal learning occurs when the task is neither too easy nor too difficult. Our system sought to keep the participating children challenged and adapted the difficulty level of practice games to the children's skill performances as measured in the games. This adaptation allowed the children to practice and improve the target skills at their own pace. Our results confirmed that the children continued to engage with our system throughout the test phase. We speculate that such engagement with our robot-assisted intervention was crucial to the observed improvements in the children's social skills.

8.4.3 Deployment in uncontrolled environments

Deployment of robotic systems outside controlled, laboratory settings is challenging. Our deployment needed to address various environmental constraints and meet different human considerations. For example, the setup location of our system was constrained by electrical power, network connectivity, and family preferences. For instance, one child was particularly sensitive to light; therefore, our system had to be set up in a dimmed room, which created additional challenges for our perception system. Furthermore, our deployment needed to accommodate other family members' needs, especially the participating child's siblings. We provided robotic toys to the siblings, so that they would not interrupt daily intervention sessions. We also made the operation of the system user-friendly by automating startup procedures and by providing a simple checklist to ensure that caregivers would feel comfortable operating the systems on their own on a daily basis. These challenges, constraints, and considerations are unique to field deployment of robotic systems aiming to interact with non-specialist users every day over a long period of time. Yet, meeting these requirements are a practical necessity for the integration of robots into our environments to provide daily support.

8.4.4 Contributions of the social robot

Although the focus of this study is not to understand the specific influence that any of the individual components of our system (including the robot, tablet-based games, perception system, etc.) have on our robot-mediated intervention, we believe that the social robot contributed positively to the observed behavior based on three converging lines of research. First, in triadic interactions between a child, an adult, and a third interaction partner, children with ASD demonstrate more social behavior toward the adult when the third interaction partner is a robot rather than a tablet-based game or another adult [135]. Second, the embodiment of the robot provides necessary affordance to convey gaze cues that are central to our behavioral intervention. Known as the Mona Lisa gaze effect [178], agents on a flat screen are limited in accurate communication of gaze directionality. Third, in tutoring interactions with both adults and typically developing children, physically embodied robots increase learning outcomes [158, 199], increase compliance to instructions [19], and increase user engagement during the interaction [195, 191] over screen-based agents. Nonetheless, we acknowledge that, in this study, the impact of the robot (or any other system component) cannot be measured independently. We present this as a limitation of this study and an area for future work.

8.4.5 Improvements in caregiver-reported social behavior

Over the month of the robot-based intervention, caregivers reported improved social behavior directed both toward themselves and toward others in areas including eye contact, initiation of communication, and responses to communication. The change in reported behavior on the caregiver survey could be, in part, related to the caregiver attending more to the child's social communication skills. It is unlikely that the change is due solely to this increased focus given the nature of the sample; caregivers of children with ASD provide ongoing support to their children in this area and generally monitor social communication development

8.4.6 Improvements in clinical measures

Our results showed improvements of children's joint attention in the absence of the robot, indicating that the children were able to demonstrate greater skill in the context of human interactions. These results advanced and differed significantly from prior research in robot-assisted autism therapy, where behavioral improvements in children with ASD were mostly documented in the context of robot-directed interactions [143]. Successful demonstration of improvement in human interactions is the ultimate goal of robot-assisted intervention, evolving beyond the mediation and scaffolds that assistive robots offer during interactions with other people. Our results provide evidence illustrating possible transferable social skills beyond robot-mediated interactions in naturalistic human interactions.

The present results have multiple clinical implications. Joint attention is the critical foundation for many higher-level social communication skills, including reciprocal exchanges and perspective-taking. Therefore, with improvements in joint attention following this intervention, in time, we may see downstream effects on other higher-level skills. In fact, we did see broader gains in the context of the current study, even in this 1-month intervention. Future work with larger and longer trials will clarify this promising, yet preliminary, result. These results give promise to the potential for robot intervention studies in group treatment to facilitate interactions between peers and improve both foundational and high-level social skills in this context.

The specific developmental growth seen in the children during their participation in this study is likely due to our system, as opposed to other treatments they received, as the children did not show the same magnitude of gains during the pretest phase, just the test phase, and the children's concomitant treatments remained stable throughout their participation in the study. That said, from a clinical intervention perspective, our study is an open-label pilot. Future studies extending the duration of the study and with randomization with appropriate control groups are necessary to verify the gains we have observed and attributed to our intervention. Also, future randomized controlled studies will better control for practice effects of the tasks.

Although our results provide evidence of benefits and the possibility of using robot-assisted autism therapy for clinical intervention, limitations of our system motivate future research on the development of effective robot-based interventions. In particular, our system relied on prespecified interaction content, which included daily opening stories and a fixed set of behavioral responses. This approach was appropriate for our target scenarios, but it would not scale well for interventions that span a longer period of time (e.g., over 30 min per day and over 30 days). How to generate coherent, engaging interaction content automatically is a core challenge for realizing a long-term, autonomous robot-assisted intervention or human-robot interaction in general. Second, our intervention personalization was focused on adjusting difficulty levels of the practice games, analogous to personalization of educational contents in intelligent tutoring systems (ITS). Our personalization algorithm was simplistic, although it matched the complexity of personalization algorithms successfully used to demonstrate learning gains in other ITS systems (e.g., [157]). More complex and detailed modeling of a child's capabilities would likely provide a more substantial impact. Furthermore, to effectively support the wide variety of behavioral characteristics of individuals with ASD, adaptive models that prioritize and personalize needs and preferences in addition to skill performance are necessary to maximize the potential of robot-assisted interventions. Third, our system was designed to provide targeted interventions, involving interactions between the robot, child, and caregiver for about 30 min each day. Although this design provided structures for targeted intervention, it missed naturalistic intervention opportunities outside of the intended sessions. These three limitations necessitate smart, adaptive systems that can provide personalized, engaging interventions to children with ASD in a variety of situations over long periods of time.

8.4.7 Long-term In-Home Deployments

Substantial effort was placed into making the system robust and easy to use. Before the deployment described in this chapter, we conducted multiple pilot tests of the system and the installation process in the homes of the research team. We attempted to make the system easy for families to use by limiting the startup required to four button presses, providing in-home training on the first day, and continuous system state logging to allow for most troubleshooting to require only powering the system off and then on again with no loss of data. Multiple efforts were made to minimize disruptions to normal operations after installation: Backup power supplies in the system base guarded against short power failures; hardware components (including the cameras, robots, and tablets) were secured in place to the table; and a troubleshooting sheet and a 24/7 technical support line (via email and phone) were provided to participating families. Last, the system limited play use to conform to the study design; the robot would play games only for one session each day and only for a maximum of 30 min.

8.5 Summary

In this chapter, we presented a novel in-home long-term autonomous system where a socially assistive robot tutored children with ASD (Autism Spectrum Disorder). Children and their caregivers played daily games with the robot that trained social, perspective-taking, and sequencing skills. The system modeled user skills and adapted to the strengths and weaknesses of the individual child by changing the difficulty of individual tasks based on the child's performance. Children improved their skills in the games throughout the one month the robot was in their home. This was the first study that showed that robots could significantly improve the social skills of children with ASD using clinically verified measures. Furthermore, parents reported seeing significant improvements in their children's social skills due to the system's training.

These last two chapters presented demonstrations of how robots can positively influence people in their homes. They were each several weeks long, and were able to keep user's engaged during the deployment. By creating personalized models of users' skills, the robotic systems could personalize their actions to maximize the benefits for the user. In the next chapter, we further discuss common themes within this thesis. We also present several open challenges that are still present when bringing robotic systems into the wild.

Chapter 9

Discussion

In this dissertation, we sought to enhance current robotic systems so they could tutor not only for a few sessions but also for the long term. We also analyzed several needed steps so that robots could teach in the wild instead of only in laboratory settings. We conducted two studies that examined how robots could tutor and create user skill models for more complex tasks. Next, we studied how different robot and setup characteristics influenced the interactions, focusing on robot roles and how robots influence people. Lastly, we presented two long-term in-home studies where a robot provided autonomous and personalized help to users.

In this chapter, we first review our main contributions. Next, we discuss several common themes. Lastly, we discuss open areas of research to be explored in future work.

9.1 Contributions

The main contributions of this thesis are:

- An algorithm that creates user skill models during complex tasks, called C-BKT. We demonstrate how our algorithm is more effective than baselines in both simulations and during an electronic circuit building task with human participants.

- A novel task selection mechanism (BKT-POMDP) that considers which task to assign a user when multiple skills are present. We show that our policy performs better than baselines in quickly creating a user skill model in both simulations and with participants. We also demonstrate in simulation that our policy selects better tasks to teach a user than baselines and performs nearly on par with the optimal algorithm.

- A controlled study comparing robot roles when tutoring adults. We compared how much a person learned and their views of robots when the robot took on the role of a traditional teacher and when they took on the role of a peer. We demonstrated that a peer robot is viewed more favorably (friendlier, smarter, and more respectful) than the instructor robot. Furthermore, participants with low prior domain knowledge learned significantly more from the peer robot than the instructor robot. This study highlights the importance of more robots taking on the role of peers rather than instructors when tutoring adults.

- The first study showing that a group of robots can cause humans to conform to them. We also show that robots cause two types of conformity: normative and informational conformity. In normative conformity, participants switched their answers to match the robots because they felt peer pressure. In informational conformity, participants believed the robots' answers to be correct and therefore changed to match them.

- We built a robotic system that provided personalized feedback to users on their exercise form in their homes for two weeks. We show that an embodied robot caused participants to display significantly fewer exercising mistakes than a video of the robot on a tablet screen. We also show that participants interacting with the physically present robot reported a higher fitness increase and were more motivated to exercise than participants interacting with the robot's video on the tablet.

- A long-term, in-home study that analyzed the effect of a robot that provided social skills training for children with Autism Spectrum Disorder (ASD). This study was the first to demonstrate clinically verified improvements in social skills in children with ASD. We also show that the robot maintained engagement throughout the month-long study.

9.2 Common Themes

In this section we will discuss several common themes and contributions that span multiple chapters throughout the thesis.

9.2.1 The importance of modeling ill-defined tasks

We reviewed prior work on intelligent tutoring systems and on robotic tutoring systems in Chapter 2. Most previous studies focused on tutoring users in domains with a single correct answer, such as mathematics, or where multiple-choice questions were asked. Most robotic applications have also concentrated their user modeling on more simple skills.

Many opportunities arise when tutoring over multiple sessions rather than a single session. Additional time allows the robot to tutor more complex skills. For example, a robot, when provided sufficient sessions, could teach a person how to play the piano, play golf, or program a computer. However, there are several limitations in user modeling algorithms. These have lacked the ability to model multiple skills per task and the ability to create a model during task execution. In Chapters 3 and 4, we have tackled some of these challenges to allow a robotic system to model a user during ill-defined tasks that often appear in dynamic environments.

During this thesis, we have modeled several ill-defined domains. We have focused on creating computer vision systems that could correctly understand what the user was doing throughout each task. Computer vision systems do not give perfect observations; therefore, we accounted for the uncertainty they create. Below we describe some of the tasks we modeled throughout the thesis.

- **Snap circuits** - In Chapters 3, 4, and 5, participants completed electronic circuit tasks. There are multiple ways to build each circuit, and the robot needs to detect whether each user solution is valid. As each circuit task can take several minutes to complete, we modeled the user throughout the whole task. A computer vision system detected every time the user added, removed, or moved a piece on the circuit. We created several different subskills to represent different components that were needed for each task to be completed correctly. For the more complex subskills (such as whether a circuit is closed), we created a graph algorithm that simulated how current passed through the system. This allowed us to detect whether the pieces were connected correctly and which pieces were powered. The robot provided personalized help by detecting the skills the user lacked and provided varied help actions such as handing the user a piece, or giving a verbal description of which pieces needed to be connected.

- **Exercising form** - In Chapter 7 we built a system for in-home exercise correction. When analyzing exercising form, there is no single correct perfect solution. We must consider whether the user's form falls within a range of valid forms. We also need to consider user characteristics, such as height and weight. During this study, we used a depth camera and body tracking algorithms to detect whenever a user had completed a repetition of an exercise. We assumed the user had completed a repetition whenever they changed their body tracking point in a particular direction. Additionally, we trained machine learning algorithms to detect if the repetition of an exercise was correct by comparing it to many different demonstrations collected as training data. While designing the machine learning algorithm, we needed to consider different factors such as different user body types, backgrounds, and lighting conditions.

- **Social skills** - In Chapter 8, the focus was on teaching social skills to children with Autism Spectrum Disorder (ASD). We taught several social skills, including perspective-taking, joint attention, emotional understanding, and sequencing. For most of these skills, the child's inputs into a large touchscreen was used to model their skills. For example, during the emotion understanding task, the child answered multiple-choice questions about how a character felt during different stories the robot told. For the perspective-taking skills, we compared where a child placed pieces onto a spaceship and if their resulting spaceship was similar to the one their caregiver built. We also modeled the child's joint attention by creating a computer vision system that detected where the child looked (at the robot, the screen, or the caregiver).

During this thesis, we modeled several less traditional skills that required computer vision systems to detect what the user knows and what they do not. However, user skill modeling is not a solved problem. The skills we did model have room for improvement and there are many other tasks that we cannot yet model accurately. Future work should continue working on improved algorithms that can vary all types of tasks under many different conditions.

9.2.2 A robot interacting with the user as a peer

Most robotic tutoring studies have the robot interacting with the user as if it were a teacher or a tutor [31]. However, in Chapter 5, we showed the many advantages of a robot interacting with the user as a peer, including that participants viewed the robot more favorably (friendlier, smarter, and more respectful) and that a subset of participants learned more with the peer robot than the tutor robot. Therefore, throughout this thesis, the robot often took on the role of a peer when interacting with users. Below, we detail how the robot interacted as a peer or colleague in several instances.

- **Peer robots inducing conformity** - In Chapter 6, participants interacted with three robots at a table. The robots were introduced to the participants as peers who would play a game with them. This allowed us to analyze the different types of conformity peer robots cause. One study showed that a peer robot is also more persuasive than an authority figure robot [226]. Furthermore, the robot being introduced as a peer would give the impression that they had the same amount of prior information about the game as the participant, allowing us to study the amount of informational conformity robots cause. Future studies should analyze how robot role influences conformity.

- **Peer robotic coach** - In Chapter 7, the robot was introduced as a peer that would exercise alongside the person for a week. The robot was given a backstory in which it wanted to exercise to get big and strong like its older brother. There are several reasons why we chose to have the robot take on the peer role. The first is that participants might feel less judged while exercising with a peer than with an authoritative trainer. The second is that we could not have the robot demonstrate the exercises due to the physical limitations of the robot we chose. Our peer robot would redirect their gaze to videos on a tablet where the exercises were demonstrated.

- **Peer robot for children with ASD** - In Chapter 8, the robot was introduced to the children and the caregiver as being a peer child robot from another planet. The robot was visiting earth to learn more about "earth people". The backstory allowed the children to welcome the robot into their home. It also gave the robot a good excuse that it would leave after a limited amount of time, making the separation process easier on the child. Additionally, children with ASD are often in therapy for multiple hours a day with human therapists making them wary of additional therapy. A robot presented as a peer who would play games with them might would have seemed more fun and engaging compared to a robot who was presented as therapist. Our study shows that the peer robot was able to maintain engagement over the one-month-long study.

9.3 Open Challenges

We identify several open challenges and research opportunities in building robots that tutor for the long term in dynamic environments, including multi-person tutoring, broadening the spaces in which robots can tutor, and ethical considerations.

9.3.1 Multi-person Tutoring

The great majority of robot tutoring studies have focused on one-on-one interactions (such as [201] and [170]). Although several robotic studies have been conducted with either small groups of children [154] or with classrooms of children [125], they are part of the minority of studies. When interacting in the home or other in-the-wild scenarios, the robot needs to be prepared to provide personalized tutoring to each group member.

To the best of the author's knowledge, there has been no work done to create skill models of users while multiple people collaborate on a task. Several people might be working in the same workspace and sometimes working together on particular tasks. The perception system needs to account for the complexity of tracking multiple people and the increased noise as team members might partially block the camera's view of other members. User skill modeling algorithms use the perception system to understand individual and joint member contributions to parts of the tasks so that individualized models can be created for each person.

There are also open questions about providing optimal help to each member. The optimal help action for one person might not be the best help for another. Therefore, algorithms need to be developed to maximize the learning gains for all users while ensuring that no single user is left behind. Additionally, there are questions about how to do optimal task assignment [214], that is, dividing the tasks for each team member to complete. Task assignments can be done either to maximize the learning of each member or to ensure quick task completion by handing each person tasks according to their strengths.

9.3.2 Novel Spaces

The focus of this work has been on bringing long-term tutoring systems that operate in the home. Future work should investigate how to continue getting all different types of robotic platforms into the home and other spaces such as industrial settings, classrooms, and the wild. Each space provides novel opportunities and domains for the robot to teach, increasing the users' knowledge around it.

With the spread of collaborative robots in industry, many opportunities arise for these robots to teach while collaborating. There are opportunities to broaden the capabilities of each employee and increase work safety by detecting who is ready for each task and having the robot teach those who are not. Future work should develop systems that can create accurate models of a user's skill state and additionally model the user's emotional state, such as whether they are tired or stressed, to increase workplace satisfaction and safety. Another element to consider is how robots are perceived when interacting in the workplace. We should explore how different robot roles make employees feel as that will affect trust and acceptance within the workplace environment.

Several pieces of work have brought robots into schools, but most were shorter term and always had a researcher present [198, 119]. There are many advantages of having a robotic system in the school. Children perceive schools as learning environments, so they will be more disposed to study. Additionally, one robotic system can help many children throughout the day, as opposed to the home where they are limited to a family. There are many open challenges in building long-term systems for schools, such as facilitating content creation between the teacher and the system and learning how a robot should individualize its teaching for each student.

Lastly, we should also create robotic systems that are not constrained to a particular environment but can be placed anywhere, including outside of buildings. These systems could focus on encouraging positive behaviors instead of solely on tutoring. For example, a robot in a school playground can discourage bullying behavior and encourage the inclusion of children who look left out of the group. A robot placed on a street can promote positive environmental attitudes, such as recycling or reducing car usage. With the increase of robots in novel spaces, we should study how to use robots to benefit society.

9.3.3 Ethical Considerations

There are several ethical questions to consider when creating long-term robotic systems. These concerns might be heightened when a researcher or developer is not always present when the robot is interacting with its user. In this section, we will discuss some main concerns, including those about robots replacing jobs, the increase in education inequality, and building systems that consider diverse students.

One common concern about robotic tutoring systems is the fear that they will start replacing teachers. Our goal when building robotic systems is not to replace teachers but to provide additional support to students who are behind in the content compared to their peers. A human teacher cannot provide individual feedback and personalize the content to each student during class. They also frequently lack the time or resources to do so after class time. A robot can provide additional personalized support to students. Nonetheless, teachers still have many fears that researchers are building these systems as their substitutes. Therefore, research should analyze how to engage instructors to be part of the design process to build robots that are complementary to teachers rather than replacements.

A critical ethical question is how to build long-term robotic systems that do not increase the education inequality between the privileged and the less privileged. Robotic systems are still costly and only in the purchasing power of the minority. If only the elite purchase private robotic tutors for their children, it could increase the disparity in education. Future research should consider making robotic platforms accessible to everyone by lowering the prices of the components. Additionally, when researchers deploy and test tutoring robots in schools, they need to include schools in low- and high-income neighborhoods.

A common problem we encountered when using pre-built computer vision packages for our research was that although it often worked well for most of the population, it frequently failed for minorities. This is because the data was mostly trained on researchers and college students who are not representative of the general population. When creating user modeling systems, we need to consider how to build algorithms that work well for diverse people. It is also vital that robotic systems are deployed and tested in varied populations, so the results are not relevant to only a subset of the population.

9.4 Summary

This dissertation provides findings on how we can build robots that can tutor for multiple weeks in a dynamic environment. We present several contributions, such as creating systems that can model ill-defined domains and exploring robot characteristics that improve tutoring. In two long-term, in-home studies, we demonstrate how a robot can successfully tutor the user and improve their skills. We have discussed common themes throughout the thesis, including the importance of modeling ill-defined tasks and the advantages of having the robot interact with the user as a peer. We also presented several open challenges that identify future research areas to increase the number of robots that tutor in dynamic environments for long periods of time.

Chapter 10

Conclusion

Robots have shown great promise to be effective tutors in many different scenarios, whether it is teaching toddlers new words [246], teaching children math [202], or teaching adults a new language [111]. However, most studies to date have been conducted in highly controlled laboratory settings. The remaining have primarily focused on tutoring in schools, with a researcher always present. In this thesis, we took several steps to allow robots to autonomously tutor in unstructured environments.

Throughout the thesis, we have focused on conducting numerous user studies to better understand different aspects of robot tutoring. Some of our studies were single session and conducted in laboratory settings (Chapters 3 - 6), while others had an autonomous robot tutoring in the home for multiple weeks (Chapters 7 and 8).

The results of our user studies provided novel algorithms and insight into developing systems that can tutor in the wild. We built an algorithm (C-BKT) that can model users' skills during complex tasks, allowing the robot to provide more precise help (Chapter 3). We also created an algorithm (BKT-POMDP) to select the optimal action (whether the right task or the correct type of help) during complex tasks with multiple skills (Chapter 4). Next, we designed a user study to analyze different robot roles and demonstrated that when tutoring adults, a robot should likely take on the role of a peer rather than the role of an instructor. The peer robot was seen more positively, and participants with low prior knowledge learned more from it (Chapter 5). In Chapter 6 we show that peer robots can indirectly influence people by causing people to conform to them during a game setting.

Lastly, we presented two robotic systems where an autonomous peer robot provided personalized tutoring in the home for several weeks. In the first system, a robot detected when a user was exercising incorrectly. We show that a physically present robot caused participants to demonstrate fewer exercise mistakes than the same robot shown in a video on a tablet (Chapter 7). In the second study, we created a robotic system that provided social skills training to children with ASD. We validate the effectiveness of our robot using clinical measures, showing that children improved their social probe scores after a month with the robot (Chapter 8). These chapters provided frameworks for creating long-term robotic systems that interacted autonomously with users for long periods.

Appendix A

User Study Data

Below we present the results from our user studies. We include both pre-test and post-test scores where these were given to participants. We also present our questionnaire answers for each participant, where relevant to the results presented throughout the thesis.

Table A.1: Participant pre-test and post-test scores, along with their difference for the user study referenced in Chapters 3 and 5.

PID	Condition	Pre-test	Post-test	Difference
P1	Peer	9	14	5
P2	Tutor	6	6	0
P3	Tutor	8	10	2
P4	Peer	3	8	5
P5	Tutor	7	13	6
P6	Peer	10	13	3
P7	Peer	6	14	8
P8	Tutor	12	17	5
P9	Tutor	6	8	2
P10	Peer	3	4	1

PID	**Condition**	**Pre-test**	**Post-test**	**Difference**
P11	Tutor	12	15	3
P12	Peer	9	8	-1
P13	Peer	10	14	4
P14	Tutor	7	9	2
P15	Tutor	6	12	6
P16	Peer	10	12	2
P17	Tutor	5	5	0
P18	Peer	10	11	1
P19	Tutor	5	11	6
P20	Tutor	13	14	1
P21	Peer	10	14	4
P22	Peer	10	12	2
P23	Peer	6	13	7
P24	Peer	5	12	7
P25	Tutor	11	14	3
P26	Tutor	9	14	5
P27	Peer	11	13	2
P28	Tutor	11	13	2
P29	Peer	4	14	10
P30	Tutor	4	10	6
P31	Tutor	12	14	2
P32	Tutor	8	7	-1
P33	Peer	7	13	6
P34	Tutor	6	14	8
P35	Peer	1	12	11
P36	Tutor	11	13	2
P37	Tutor	12	15	3

Table A.2: The participant study data related to Chapter 6. We present the Condition, the number of critical round changes (CR changes), the number of non-critical Round Changes (non-CR Changes). We also provide 1-5 Likert Scale questionnaire answers for two statements: "I felt pressure to change my answers because of the robots" (Pressure), and "The robots are better at playing this game than me" (Better).

PID	Condition	CR Changes	non-CR Changes	Pressure	Better
P1	Quantitative	2	0	4	3
P2	Quantitative	3	6	2	3
P3	Quantitative	0	0	1	5
P4	Quantitative	2	0	2	2
P5	Quantitative	0	2	1	5
P6	Quantitative	3	2	2	4
P7	Quantitative	3	1	4	2
P8	Quantitative	2	2	4	3
P16	Quantitative	2	4	4	3
P17	Quantitative	2	2	3	1
P20	Quantitative	0	1	2	2
P23	Quantitative	5	3	5	4
P26	Quantitative	1	0	4	3
P27	Quantitative	1	5	2	1
P30	Quantitative	1	4	1	4
P47	Quantitative	3	2	3	3
P48	Quantitative	4	3	4	2
P9	Blind	1	1	1	3
P10	Blind	0	1	1	5
P11	Blind	0	2	1	3
P12	Blind	3	3	2	2
P13	Blind	3	6	3	3
P14	Blind	0	1	2	2
P15	Blind	1	0	1	3
P18	Blind	0	3	3	4
P19	Blind	0	2	1	3
P21	Blind	2	1	2	4
P22	Blind	0	0	1	3
P24	Blind	1	0	1	4
P25	Blind	1	3	3	4
P28	Blind	1	0	1	2
P29	Blind	0	0	1	4
P31	Selected	0	0	1	
P32	Selected	2	5	1	3
P33	Selected	1	3	1	5

PID	Condition	CR Changes	non-CR Changes	Pressure	Better
P34	Selected	2	2	4	2
P35	Selected	0	3	2	2
P36	Selected	0	2	4	5
P37	Selected	1	6	3	5
P38	Selected	1	3	2	4
P39	Selected	1	2	2	2
P40	Selected	0	1	1	2
P41	Selected	3	4	2	3
P42	Selected	1	2	2	3
P43	Selected	5	12	1	3
P44	Selected	1	1	4	4
P45	Selected	0	1	1	1
P46	Selected	1	2	3.5	4
P50	Staring	3		4	2
P52	Staring	0	3	5	2
P54	Staring	3	2	4	4
P55	Staring	2	1	4	1
P56	Staring	2	4	5	4
P57	Staring	4	2	4	4
P58	Staring	2	1	3	3
P59	Staring	0	2	1	1
P60	Staring	1	2	1	2
P61	Staring	4	2	4	3
P62	Staring	2	2	3	1
P63	Staring	6	3	5	5
P64	Staring	2	0	4	4
P65	Staring	2	4	3	3
P66	Staring	1	2	3	2
P67	Staring	3	0	4	2

Table A.3: The participant study data related to Chapter 7. We present the Condition, the percentage of days they exercised with the robot while it was with them in their home (Exercised), the percentage of exercises done correctly in the first two days (C- First 2), and the percentage of exercises done correctly in the last two days (C - Last 2).

PID	**Condition**	**Exercises**	**C - First 2**	**C - Last 2**
P1	Robot	0.77	0.79	0.84
P2	Robot	1.00	0.84	0.66
P3	Robot	0.69	0.65	0.74
P4	Robot	1.00	0.81	0.70
P13	Robot	0.53	0.55	0.64
P16	Robot	0.40	0.77	0.58
P17	Robot	0.42	0.63	0.71
P20	Robot	0.77	0.86	0.82
P21	Robot	0.64	0.56	0.64
P23	Robot	0.76	0.79	1.00
P25	Robot	0.79	0.64	0.61
P29	Robot	0.71	0.94	0.72
P35	Robot	1.00	0.61	0.39
P36	Robot	0.56	0.54	0.47
P6	Tablet	1.00	0.67	0.56
P7	Tablet	0.86	0.58	0.21
P18	Tablet	0.38	0.68	
P19	Tablet	0.57	0.71	0.57
P22	Tablet	0.69	0.52	0.81
P24	Tablet	0.71	0.17	0.41
P26	Tablet	0.33	0.42	0.54
P30	Tablet	0.93	0.72	0.54
P33	Tablet	0.73	0.28	0.25
P31	Tablet	0.33	0.85	0.89
P32	Tablet	0.78	0.66	0.51

Table A.4: The participant score data relating to Chapter 8. In this table, we have the age of each participant and their nonverbal reasoning performance on the DAS. In sequence, we have the joint attention scores for thirty days before the intervention began (JA - 1), the first day of the intervention (JA - 2), the last day of the intervention (JA - 3), and thirthy days after the end of the intervention (JA - 4).

PID	**age**	**DAS**	**JA - 1**	**JA - 2**	**JA - 3**	**JA - 4**
EXP01	8.7	104	22	12	25	23
EXP03	8.3	89	18	16	21	22
EXP04	7.3	85	9	12	13	10
EXP05	10.3	71	12	11	25	16
EXP06	8.11	111	22	20	22	22
EXP08	9.2	111	18	17	17	12
EXP11	12	102	20	21	21	22
EXP12	10.3	100	17	19	22	22
EXP14	8.6	78	14	13	22	15

Table A.5: The caregiver scores for the first day and last day of the daily surveys relating to Chapter 8. Q2 - How easy was it to engage your child with the robot today? Q8 - Have you noticed any changes in your son/daughter's eye contact with you? Q9 - Have you noticed any changes in your son/daughter's initiation of communication with you? Q10 - Have you noticed any changes in your son/daughter's responding to communication bids from you? Q12 - Have you noticed any changes in your son/daughter's eye contact with others? Q13 - Have you noticed any changes in your son/daughter's initiation of communication with others? Q14 - Have you noticed any changes in your son/daughter's response to communication bids from others?

		Q2	**Q8**	**Q9**	**Q10**	**Q12**	**Q13**	**Q14**
Participant 1	first day	3	3	3	3	3	3	3
	last day	5	3	3	4	4	4	4
Participant 2	first day	5	3	3	3	3	3	3
	last day	5	5	5	5	5	5	5
Participant 3	first day	5	3	3	3	3	3	3
	last day	5	4	5	5	4	5	4
Participant 4	first day	4	3	3	3	3	3	3
	last day	5	5	5	5	5	5	5
Participant 5	first day	5	3	3	3	3	3	3
	last day	4	4	4	4	4	3	4
Participant 6	first day	3	3	4	4	4	3	2
	last day	5	5	5	5	5	4	4
Participant 7	first day	5	3	3	3	3	3	3
	last day	5	3	5	3	3	3	3
Participant 8	first day	5	3	3	3	3	3	3
	last day	5	3	3	3	3	3	3
Participant 9	first day	5	3	3	3	3	3	3
	last day	3	5	5	3	4	5	3
Participant 10	first day	5	3	3	3	3	3	3
	last day	5	3	3	3	3	3	3
Participant 11	first day	5	3	4	3	3	3	3
	last day	3	3	3	3	3	4	4
Participant 12	first day	5	3	3	3	3	3	3
	last day	5	2	3	3	3	3	3

Appendix B

Snap Circuit Task

In this appendix we present details regarding the Snap Circuit task that was used in Chapters 3 and 5.

B.1 Skills

We tested the following skills in the Snap Circuits domain:

- **1-LED** - When a LED is required for the task.
- **2-LED Directionality** - To understand which direction the LED should be facing within the circuit.
- **3-LED needs resistor** - When an LED is placed on the board, it needs a resistor.
- **4-Resistor** - How a resistor can be used to lower the light or music.
- **5-Motor** - When a motor should be used.
- **6-Switch** - When a switch should be used.
- **7-Button** - When a button should be used.

- **8-Speaker** - When a speaker should be used.
- **9-MC** - When a music circuit should be used.
- **10-Connect** - How to correctly connect two pieces together.
- **11-Closed** - A circuit must be closed, forming a complete loop from the positive side of the battery to the negative of the battery.
- **12-Power MC** - How to connect the Music Circuit (MC) to power. The positive of the MC should be connected to the positive of the batter, and the negative of the MC should be connected to the negative of the battery.
- **13-Signal MC** - How to connect the positive of the battery to the signal of the MC (which enables music to be played in a rhythm).
- **14-Trigger MC** - How to correctly connect the positive side of the battery to the Trigger input on the MC.
- **15-Hold MC** - How to correctly connect the positive side of the battery to the Hold input on the MC.
- **16-AND gate** - How to create an AND gate using pieces. This was for tasks that asked a switch to be on and a button pressed for the circuit to work.
- **17-OR gate** - How to create an OR gate using pieces. This was for tasks that asked a switch to be on or a button pressed for the circuit to work.

B.2 Tasks

Below are presented all the tasks that could be given to participants.

- **Task 1** - "Build a circuit that you can turn a light on and off using a switch."

- **Task 2** - "Build a circuit that you can turn a light on and off using a button."
- **Task 3** - "Build a circuit that you can turn a motor on and off using a switch."
- **Task 4** - "Build a circuit that you can turn a motor on and off using a button."
- **Task 5** - "Build a circuit that constantly will have a motor spinning."
- **Task 6** - "Build a circuit that has a constant light on."
- **Task 7** - "Build a circuit that plays music."
- **Task 8** - "Build a circuit that plays music when a button is pressed."
- **Task 9** - "Build a circuit that plays music when a switch is turned on."
- **Task 10** - "Build a circuit that plays low sounding music."
- **Task 11** - "Build a circuit that plays low music when a button is pressed."
- **Task 12** - "Build a circuit that plays low music when a switch is turned on."
- **Task 13** - "Build a circuit that blinks a light in a rhythm of a song."
- **Task 14** - "Build a circuit that blinks a light in the rhythm of a song when a button is pressed."
- **Task 15** - "Build a circuit that blinks a light in the rhythm of a song when a switch is turned on."
- **Task 16** - "Build a circuit that blinks a low light in the rhythm of a song."
- **Task 17** - "Build a circuit that blinks a low light in the rhythm of a song when a button is pressed."
- **Task 18** - "Build a circuit that blinks a low light in the rhythm of a song when a switch is turned on."

- **Task 19** - "Build a circuit that plays music while the button is pressed. But stops playing when it is not pressed."
- **Task 20** - "Build a circuit that triggers a song being played each time you spin a motor."
- **Task 21** - "Build a circuit that blinks a light in the rhythm of a song while the button is pressed. But stops playing when it is not pressed."
- **Task 22** - "Build a circuit that triggers a light to blink in the rhythm of a song each time you spin a motor."
- **Task 23** - "Build a circuit that turns on a light when a switch is turned on AND a button is pressed."
- **Task 24** - "Build a circuit that turns on a light when a switch is turned on OR a button is pressed."
- **Task 25** - "Build a circuit that spins a motor when a switch is turned on AND a button is pressed."
- **Task 26** - "Build a circuit that spins a motor when a switch is turned on OR a button is pressed."
- **Task 27** - "Build a circuit that plays music when a switch is turned on AND a button is pressed."
- **Task 28** - "Build a circuit that plays music when a switch is turned on OR a button is pressed."
- **Task 29** - "Build a circuit that blinks a light in a rhythm of a song when a switch is turned on AND a button is pressed."

- **Task 30** - "Build a circuit that blinks a light in a rhythm of a song when a switch is turned on OR a button is pressed."

B.3 Robot Utterances and Actions

Below are presented all the utterances for both conditions.

Reinforcement Question 1

Peer - "An LED seems like the right piece, can you explain to me what it does?"

Peer - "Does the LED emit a light?"

Peer - "What does an LED do?"

Tutor - "An LED seems like the right piece, can you explain to me what it does?"

Tutor - "Does the LED emit a light?"

Tutor - "What does an LED do?"

Reinforcement Question 2

Peer - "Why did you choose to orient the LED like that?"

Peer - "What direction should the LED be pointing from compared to the positive side of the battery?"

Tutor - "Why did you choose to orient the LED like that?"

Tutor - "What direction should the LED be pointing from compared to the positive side of the battery?"

Reinforcement Question 3

Peer - "Ah cool, I see you added both a resistor and an LED, why did we add the resistor?"

Peer - "When we have an LED in the circuit, should we always have a resistor?"

Peer - "How come we added a resistor to the circuit?"

Tutor - "Ah cool, I see you added both a resistor and an LED, why did you add the resistor?"

Tutor - "When you have an LED in the circuit, should you always have a resistor?"

Tutor - "How come you added a resistor to the circuit?"

Reinforcement Question 4

Peer - "Is the resistor the piece that makes the music lower?"

Peer - "I see a resistor in the circuit, how does it affect the circuit?"

Tutor - "Is the resistor the piece that makes the music lower?"

Tutor - "I see a resistor in the circuit, how does it affect the circuit?"

Reinforcement Question 5

Peer - "Can you explain to me what a motor does?"

Peer - "The motor seems like a good idea, can you explain to me how it works?"

Peer - "Interesting. In general, when should a motor be used?"

Tutor - "Can you explain to me what a motor does?"

Tutor - "The motor seems like a good idea, can you explain to me how it works?"

Tutor - "Interesting. In general, when should a motor be used?"

Reinforcement Question 6

Peer - "Nice idea with the switch. What happens when we turn it on?"

Peer - "When should a switch be used?"

Peer - "I see a switch! How does the switch work?"

Tutor - "Nice idea with the switch. What happens when you turn it on?"

Tutor - "When should a switch be used?"

Tutor - "I see a switch! How does the switch work?"

Reinforcement Question 7

Peer - "Good choice to add a button to the circuit. What happens if we press it?"

Peer - "Can you explain to me what a button does?"

Peer - "When should we use a button in a circuit?"

Tutor - "Good choice to add a button to the circuit. What happens if you press it?"

Tutor - "Can you explain to me what a button does?"

Tutor - "When should you use a button in a circuit?"

Reinforcement Question 8

Peer - "I like the speaker. How does it work?"

Peer - "Is the speaker the piece that will play music?"

Peer - "When should we add a speaker to the circuit?"

Tutor - "I like the speaker. How does it work?"

Tutor - "Is the speaker the piece that will play music?"

Tutor - "When should you add a speaker to the circuit?"

Reinforcement Question 9

Peer - "What is the difference between the speaker and the music circuit?"

Peer - "What is the functionality of the music circuit?"

Peer - "Music circuits are tricky. Can you explain to me what it does?"

Tutor - "What is the difference between the speaker and the music circuit?"

Tutor - "What is the functionality of the music circuit?"

Tutor - "Music circuits are tricky. Can you explain to me what it does?"

Reinforcement Question 10

Peer - "Do we use the blue wires to connect pieces together?"

Peer - "How do the blue wires affect the circuit?"

Peer - "Why do we need to connect pieces together?"

Tutor - "Do you use the blue wires to connect pieces together?"

Tutor - "How do the blue wires affect the circuit?"

Tutor - "Why do you need to connect pieces together?"

Reinforcement Question 11

Peer - "How come you connected all the pieces together like that?"

Peer - "I see that all the pieces are connected to form a circuit. Should we always do that?"

Peer - "Why did you form a circuit out of all the pieces?"

Tutor - "How come you connected all the pieces together like that?"

Tutor - "I see that all the pieces are connected to form a circuit. Should you always do that?"

Tutor - "Why did you form a circuit out of all the pieces?"

Reinforcement Question 12

Peer - "I see that the positive signal of the music circuit is connected to the positive in the battery. How come?"

Peer - "I see that the negative signal of the music circuit is connected to the negative in the battery. Should we always do that?"

Peer - "Can you explain to me, how we power a music circuit?"

Tutor - "I see that the positive signal of the music circuit is connected to the positive in the battery. How come?"

Tutor - "I see that the negative signal of the music circuit is connected to the negative in the battery. Should you always do that?"

Tutor - "Can you explain to me, how you power a music circuit?"

Reinforcement Question 13

Peer - "I see that we connected the output port of the music circuit to the positive side of the battery. Why did you do that?"

Peer - "Can you explain to me how we connect the output port of the music circuit?"

Peer - "What does the output port of the music circuit do?"

Tutor - "I see that you connected the output port of the music circuit to the positive side of the battery. Why did you do that?"

Tutor - "Can you explain to me how you connect the output port of the music circuit?"

Tutor - "What does the output port of the music circuit do?"

Reinforcement Question 14

Peer - "What is the trigger port of the music circuit?"

Peer - "What do we connect to the trigger port in the music circuit?"

Tutor - "What is the trigger port of the music circuit?"

Tutor - "What should you connect to the trigger port in the music circuit?"

Reinforcement Question 15

Peer - "What is the hold port of the music circuit?"

Peer - "What do we connect to the hold port in the music circuit?"

Tutor - "What is the hold port of the music circuit?"

Tutor - "What do you connect to the hold port in the music circuit?"

Reinforcement Question 16

Peer - "What happens if we press the button and turn on the switch?"

Tutor - "What happens if you press the button and turn on the switch?"

Reinforcement Question 17

Peer - "What happens if we press the button or turn on the switch?"

Tutor - "What happens if you press the button or turn on the switch?"

Reinforcement Good 1

Peer - "Interesting, yes, an LED should emit a light"

Peer - "Adding the LED was a good idea!"

Peer - "The LED created a light in the circuit"

Tutor - "Interesting, yes, an LED will emit a light"

Tutor - "Adding the LED was a good idea!"

Tutor - "The LED created a light in the circuit"

Reinforcement Good 2

Peer - "The LED seems to be pointing in the correct direction"

Peer - "Yes, I agree that the battery positive port should be connected to the positive port of the LED"

Tutor - "The LED seems to be pointing in the correct direction"

Tutor - "Yes, I agree that the battery positive port should be connected to the positive port of the LED"

Reinforcement Good 3

Peer - "Yes. We should always add a resistor when there is a LED in the circuit."

Peer - "I see both a resistor and an LED, I will remember that."

Peer - "I have learned that a resistor should be added when there is an LED."

Tutor - "Yes. You should always add a resistor when there is a LED in the circuit."

Tutor - "I see both a resistor and an LED, That is correct."

Tutor - "You have learned that a resistor should be added when there is an LED."

Reinforcement Good 4

Peer - "The resistor lowers the music."

Peer - "The resistor will do the job of lowering volume of the music in this circuit."

Peer - "I can see that a resistor should be added when we want to lower the music."

Tutor - "The resistor lowers the music."

Tutor - "The resistor will do the job of lowering volume of the music in this circuit."

Tutor - "I can see you know that a resistor should be added when you want to lower the music."

Reinforcement Good 5

Peer - "I see a motor in the circuit just like the task asked."

Peer - "Adding a motor was a good idea."

Peer - "Yes, we needed to have the motor in the circuit."

Tutor - "I see a motor in the circuit just like the task asked."

Tutor - "Adding a motor was a good idea."

Tutor - "Yes, you needed to have the motor in the circuit."

Reinforcement Good 6

Peer - "I agree that a switch was needed."

Peer - "There is a switch just like the task asked."

Peer - "Now we can turn the circuit on and off using the switch."

Tutor - "I agree that a switch was needed."

Tutor - "There is a switch just like the task asked."

Tutor - "Now you can turn the circuit on and off using the switch."

Reinforcement Good 7

Peer - "Clicking the button will let us turn the circuit on and off."

Peer - "Adding a button was a good idea."

Tutor - "Clicking the button will let you turn the circuit on and off."

Tutor - "Adding a button was a good idea."

Reinforcement Good 8

Peer - "The speaker will play music in the circuit."

Peer - "There is a speaker on the circuit just like the task asked."

Tutor - "The speaker will play music in the circuit."

Tutor - "There is a speaker on the circuit just like the task asked."

Reinforcement Good 9

Peer - "Good job on adding the music circuit, that is a hard piece."

Peer - "Yea, having a music circuit is a good idea."

Peer - "The music circuit will send a signal to the speaker to play music."

Tutor - "Good job on adding the music circuit, that is a hard piece"

Tutor - "Yea, having a music circuit is a good idea."

Tutor - "The music circuit will send a signal to the speaker to play music."

Reinforcement Good 10

Peer - "I like how we connected several of the pieces using wires."

Peer - "The blue wires connect several of the pieces together."

Peer - "Good job, The pieces can all send signals to each other when they are connected with the wires."

Tutor - "I like how you connected several of the pieces using wires."

Tutor - "The blue wires connect several of the pieces together."

Tutor - "Good job, The pieces can all send signals to each other when they are connected with the wires."

Reinforcement Good 11

Peer - "Impressive, we built a complete circuit together."

Peer - "Yay, I think we did a good job having all the pieces connected together to the battery."

Peer - "It looks like we build a good circuit."

Tutor - "Impressive, you built a complete circuit."

Tutor - "Yay, I think you did a good job having all the pieces connected together to the battery."

Tutor - "It looks like you build a good circuit."

Reinforcement Good 12

Peer - "The music circuit seems to be powered correctly."

Peer - "Yay, the music circuits positive and negative ports are connected to the battery."

Peer - "I like how we connected the music circuit."

Tutor - "The music circuit is powered correctly."

Tutor - "Yay, the music circuits positive and negative ports are connected to the battery."

Tutor - "I like how you connected the music circuit."

Reinforcement Good 13

Peer - "Awesome, the music circuit can now send signals to play a rhythm of a song."

Peer - "We are doing a good job connecting the out port of the music circuit."

Tutor - "Awesome, the music circuit can now send signals to play a rhythm of a song."

Tutor - "You are doing a good job connecting the out port of the music circuit."

Reinforcement Good 14

Peer - "The trigger port of the music circuit it connected correctly."

Peer - "Lets try spinning that motor, and see if it triggers the music circuit to start playing."

Peer - "The motor on the trigger port was a good idea."

Tutor - "The trigger port of the music circuit it connected correctly."

Tutor - "Try spinning that motor, and see if it triggers the music circuit to start playing."

Tutor - "The motor on the trigger port was a good idea."

Reinforcement Good 15

Peer - "The hold port of the music circuit it connected correctly."

Peer - "Lets try pressing the button, while its pressed it should play music."

Peer - "The button on the hold port was a good idea."

Tutor - "The hold port of the music circuit it connected correctly."

Tutor - "Try pressing the button, while its pressed it should play music."

Tutor - "The button on the hold port was a good idea."

Reinforcement Good 16

Peer - "Lets press the button and turn on the switch to see what happens."

Peer - "Yes, the switch and the button seem to be correct."

Tutor - "Press the button and turn on the switch to see what happens."

Tutor - "Yes, the switch and the button are correct."

Reinforcement Good 17

Peer - "Lets press the button and turn on the switch to see what happens."

Peer - "Yes, the switch and the button seem to be correct."

Tutor - "Press the button and turn on the switch to see what happens."

Tutor - "Yes, the switch and the button seem to be correct."

Wrong Piece [Robot Points at Piece]

Peer - "I am not so sure about this [piece_name] over here, should it be on the board?"

Peer - "Let's take another look at the task. Do you think it needs this [piece_name]?"

Peer - "This [piece_name] over here might be useful in other moments, but I don't think its needed this time."

Peer - "Over here we have a [piece_name]. I don't think its needed right now."

Tutor - "I am not so sure about this [piece_name] over here, should it be on the board?"

Tutor - "Take another look at the task. Do you think it needs this [piece_name]?"

Tutor - "This [piece_name] over here is useful in other moments, but its not needed this time."

Tutor - "Over here you have a [piece_name]. It is not needed right now."

Wrong Piece [No Pointing]

Peer - "I see we have added a [piece_name]. Let's think about if it is needed for the current task."

Peer - "Can you explain to me what a [piece_name] does?. Do you think we need it for the current task?"

Peer - "I am pretty sure the [piece_name] should not be on the board for the task we are doing right now."

Tutor - "I see you have added a [piece_name]. Think about if it is needed for the current task."

Tutor - "Can you explain to me what a [piece_name] does?. Do you think you need it for the current task?"

Tutor - "I am pretty sure the [piece_name] should not be on the board for the task you are doing right now."

Help Utterance 1

Peer - "An LED might be needed on the board."

Peer - "Looks like we dont have an LED on the board yet, how do you feel about adding one?"

Peer - "Adding an LED, will satisfy having something that gives us a light."

Peer - "An LED creates a little light when it is connected correctly."

Peer - "The task is mentioning having something that creates a light. Do you think the LED might do that?"

Tutor - "An LED might be needed on the board."

Tutor - "Looks like you dont have an LED on the board yet, how do you feel about adding one?"

Tutor - "Adding an LED, will satisfy having something that gives you a light."

Tutor - "An LED creates a little light when it is connected correctly."

Tutor - "The task is mentioning having something that creates a light. Do you think the LED might do that?"

Help Utterance 2

Peer - "We need to make sure we have the LED facing the correct direction. It does not seem to be right now."

Peer - "The LED positive side should be connected via wires to the batteries positive side."

Peer - "There is a little positive symbol on one of the LED sides. Lets try connecting that to the positive side of the battery using wires."

Tutor - "You need to make sure you have the LED facing the correct direction. It does not seem to be right now."

Tutor - "The LED positive side should be connected via wires to the batteries positive side."

Tutor - "There is a little positive symbol on one of the LED sides. Try connecting that to the positive side of the battery using wires."

Help Utterance 3

Peer - "The LED can not take all the current passing through it, so we must add a resistor to lower it."

Peer - "I have learned that anytime we add an LED to the board, we must also add something that lowers the amount of current going through."

Peer - "Because we have added an LED, we should also add a resistor."

Tutor - "The LED can not take all the current passing through it, so you must add a resistor to lower it."

Tutor - "Anytime you add an LED to the board, you must also add something that lowers the amount of current going through."

Tutor - "Because you have added an LED, you should also add a resistor."

Help Utterance 4

Peer - "The task wants us to play low music. The resistor lowers the amount of current passing through the circuit."

Peer - "So we need to play low music. What piece do you think could lower the music?"

Peer - "We need to add a resistor. It would lower the music."

Tutor - "The task wants you to play low music. The resistor lowers the amount of current passing through the circuit."

Tutor - "So you need to play low music. What piece do you think could lower the music?"

Tutor - "You need to add a resistor. It would lower the music."

Help Utterance 5

Peer - "So the task requires a motor. The only piece that looks like a motor is the yellow one with the spinny thing on top."

Peer - "We need to add the motor to the circuit."

Peer - "It looks like the task wants a motor, lets try adding it to the circuit."

Tutor - "So the task requires a motor. The only piece that looks like a motor is the yellow one with the spinny thing on top."

Tutor - "You need to add the motor to the circuit."

Tutor - "It looks like the task wants a motor, try adding it to the circuit."

Help Utterance 6

Peer - "So we need to turn the circuit on and off. How about using the switch?"

Peer - "Lets try adding the switch to the circuit."

Peer - "How about the switch? It would turn the circuit on and off."

Tutor - "So you need to turn the circuit on and off. How about using the switch?"

Tutor - "Try adding the switch to the circuit."

Tutor - "How about the switch? It would turn the circuit on and off."

Help Utterance 7

Peer - "So we need to turn the circuit on by pressing the button. Lets add the button?"

Peer - "Lets try adding the button to the circuit."

Peer - "How about the button? It would turn the circuit on when we press on it."

Tutor - "So you need to turn the circuit on by pressing the button. Add the button?"

Tutor - "Try adding the button to the circuit."

Tutor - "How about the button? It would turn the circuit on when you press on it."

Help Utterance 8

Peer - "We need a piece that can play some music. How about the speaker?"

Peer - "The speaker can play music. Lets add it to the circuit."

Peer - "Can we try adding the red speaker to the circuit?"

Tutor - "You need a piece that can play some music. How about the speaker?"

Tutor - "The speaker can play music. Add it to the circuit."

Tutor - "Can you try adding the red speaker to the circuit?"

Help Utterance 9

Peer - "The music circuit will create the song for the speaker to play."

Peer - "Lets try adding the music circuit to the board. It will create the song waves."

Peer - "We still need to add a music circuit to the board."

Tutor - "The music circuit will create the song for the speaker to play."

Tutor - "Try adding the music circuit to the board. It will create the song waves."

Tutor - "You still need to add a music circuit to the board"

Help Utterance 10

Peer - "Some of the pieces on the board aren't connected to each other. Can we try connecting them together using the blue wires."

Peer - "We can connect pieces together using the blue wires. They snap to each other at the ends."

Peer - "The pieces need to be connected to each other and to the battery."

Tutor - "Some of the pieces on the board aren't connected to each other. Can you try connecting them together using the blue wires."

Tutor - "You can connect pieces together using the blue wires. They snap to each other at the ends."

Tutor - "The pieces need to be connected to each other and to the battery."

Help Utterance 11

Peer - "Lets make sure all the pieces are connected together in a loop. The battery needs to be part of the loop."

Peer - "Some pieces are not connected to the battery. Lets connect them all together using wires."

Peer - "Using the blue wires, we should connect all the pieces together in a circle, and at the end of the circle we should have the battery."

Tutor - "Make sure all the pieces are connected together in a loop. The battery needs to be part of the loop."

Tutor - "Some pieces are not connected to the battery. Connect them all together using wires."

Tutor - "Using the blue wires, you should connect all the pieces together in a circle, and at the end of the circle you should have the battery."

Help Utterance 12

Peer - "To power the music circuit we need to make sure that its positive port is connected to the positive port of the battery."

Peer - "Connect the negative port of the music circuit to the negative port of the battery."

Peer - "Lets connect the positives and the negatives of the music circuit and the battery together using wires."

Tutor - "To power the music circuit you need to make sure that its positive port is connected to the positive port of the battery."

Tutor - "Connect the negative port of the music circuit to the negative port of the battery."

Tutor - "Connect the positives and the negatives of the music circuit and the battery together using wires."

Help Utterance 13

Peer - "Lets Connect the speaker to the output port of the music circuit, and the other end of the speaker to the battery."

Peer - "The speaker should be in between the output port of the music circuit and the positive side of the battery."

Peer - "Using wires, lets connect the speaker to the output port of the battery."

Peer - "Using wires, connect one side of the speaker to the positive side of the battery."

Tutor - "Connect the speaker to the output port of the music circuit, and the other end of the speaker to the battery."

Tutor - "The speaker should be in between the output port of the music circuit and the positive side of the battery."

Tutor - "Using wires, connect the speaker to the output port of the battery."

Tutor - "Using wires, connect one side of the speaker to the positive side of the battery."

Help Utterance 14

Peer - "We need to connect the motor to the trigger input of the music circuit."

Peer - "If the motor is connect to the trigger input of the music circuit, spinning the motor will start the music."

Peer - "Lets add a motor between the trigger of the music circuit and the positive side of the battery."

Tutor - "You need to connect the motor to the trigger input of the music circuit."

Tutor - "If the motor is connect to the trigger input of the music circuit, spinning the motor will start the music."

Tutor - "Add a motor between the trigger of the music circuit and the positive side of the battery."

Help Utterance 15

Peer - "While we are holding down the button, the music needs to play."

Peer - "Lets add the button to the hold port of the music circuit, the other side of the button we can connect to the battery."

Peer - "Lets have a button in between the hold port of the music circuit and the positive side of the battery."

Tutor - "While you are holding down the button, the music needs to play."

Tutor - "Add the button to the hold port of the music circuit, the other side of the button you can connect to the battery."

Tutor - "Have a button in between the hold port of the music circuit and the positive side of the battery."

Help Utterance 16

Peer - "For this task the circuit will only work when both the button is pressed and the switch is on."

Peer - "There should be a button and switch in the main loop of the circuit."

Peer - "For this task, lets have a button and a switch connected together."

Tutor - "For this task the circuit will only work when both the button is pressed and the switch is on."

Tutor - "There should be a button and switch in the main loop of the circuit."

Tutor - "For this task, have a button and a switch connected together."

Help Utterance 17

Peer - "In this task either the button needs to be pressed, or, the switch needs to be turned on for the circuit to work."

Peer - "For this one, we need to split one of the wires, into two wires, and connect the button and the switch to each end. Afterwards you can bring the wires together again."

Peer - "One way to do this, is to connect both the ends of switch and the button together using wires. That way as long as one is pressed, current can pass through."

Peer - "Lets place the switch and button side by side, and connect them using wires at both ends. Now connect each end to the positive and negative sides of the battery."

Tutor - "In this task either the button needs to be pressed, or, the switch needs to be turned on for the circuit to work."

Tutor - "For this one, you need to split one of the wires, into two wires, and connect the button and the switch to each end. Afterwards you can bring the wires together again."

Tutor - "One way to do this, is to connect both the ends of switch and the button together using wires. That way as long as one is pressed, current can pass through."

Tutor - "Lets place the switch and button side by side, and connect them using wires at both ends. Now connect each end to the positive and negative sides of the battery."

Help Movement 1 [Robot Grabs Piece and Hands Over]

Peer - "Here, lets try to add the LED to the circuit."

Peer - "How about adding this LED to the circuit to create the light."

Peer - "This LED is needed on the board."

Tutor - "Here, try to add the LED to the circuit."

Tutor - "How about adding this LED to the circuit to create the light."

Tutor - "This LED is needed on the board."

Help Movement 2 [Robot Points to LED]

Peer - "We want to have the positive side of the battery connected to the positive side of the LED, and we want to have the negative side of the battery connected to the negative side of the LED"

Tutor - "You want to have the positive side of the battery connected to the positive side of the LED, and you want to have the negative side of the battery connected to the negative side of the LED."

Help Movement 3 [Robot Grabs Piece and Hands Over]

Peer - "Here, lets try to add the resistor to the circuit."

Peer - "Whenever there is an LED, we also need a resistor."

Peer - "This resistor is needed on the board."

Tutor - "Here, try to add the resistor to the circuit."

Tutor - "Whenever there is an LED, you also need a resistor."

Tutor - "This resistor is needed on the board."

Help Movement 4 [Robot Grabs Piece and Hands Over]

Peer - "Here, lets try to add the resistor to the circuit."

Peer - "How about adding this resistor to the circuit as it will make the music play softer."

Peer - "This resistor is needed on the board."

Tutor - "Here, try to add the resistor to the circuit."

Tutor - "How about adding this resistor to the circuit as it will make the music play softer."

Tutor - "This resistor is needed on the board."

Help Movement 5 [Robot Grabs Piece and Hands Over]

Peer - "Here, lets try to add the motor to the circuit."

Peer - "How about adding this motor to the circuit as the task is asking for a motor."

Peer - "This motor is needed on the board."

Tutor - "Here, try to add the motor to the circuit."

Tutor - "How about adding this motor to the circuit as the task is asking for a motor."

Tutor - "This motor is needed on the board."

Help Movement 6 [Robot Grabs Piece and Hands Over]

Peer - "Here, lets try to add the switch to the circuit."

Peer - "How about adding this switch to the circuit so we can turn it off and on."

Peer - "This switch is needed on the board."

Tutor - "Here, try to add the switch to the circuit."

Tutor - "How about adding this switch to the circuit so you can turn it off and on."

Tutor - "This switch is needed on the board."

Help Movement 7 [Robot Grabs Piece and Hands Over]

Peer - "Here, lets try to add the button to the circuit."

Peer - "How about adding this button to the circuit so the circuit is on while we are pressing it."

Peer - "This button is needed on the board."

Tutor - "Here, try to add the button to the circuit."

Tutor - "How about adding this button to the circuit so the circuit is on while you are pressing it."

Tutor - "This button is needed on the board."

Help Movement 8 [Robot Grabs Piece and Hands Over]

Peer - "Here, lets try to add the speaker to the circuit."

Peer - "How about adding this speaker to the circuit as it can play us some music."

Peer - "This speaker is needed on the board."

Tutor - "Here, try to add the speaker to the circuit."

Tutor - "How about adding this speaker to the circuit as it can play you some music."

Tutor - "This speaker is needed on the board."

Help Movement 9 [Robot Grabs Piece and Hands Over]

Peer - "Here, lets try to add the music circuit to the circuit."

Peer - "How about adding this music circuit to the circuit and it can send a song for the speaker to play."

Peer - "This music circuit is needed on the board."

Tutor - "Here, try to add the music circuit to the circuit."

Tutor - "How about adding this music circuit to the circuit and it can send a song for the speaker to play."

Tutor - "This music circuit is needed on the board."

Help Movement 10 [Points to Board]

Peer - "We have several pieces on the board. But they aren't connect to each other. Can we try connecting them together using the blue wires."

Peer - "We can connect pieces together using the blue wires. They snap to each other at the ends."

Peer - "The pieces need to be connected to each other and to the battery."

Tutor - "You have several pieces on the board. But they aren't connect to each other. Can you try connecting them together using the blue wires."

Tutor - "You can connect pieces together using the blue wires. They snap to each other at the ends."

Tutor - "The pieces need to be connected to each other and to the battery."

Help Movement 11 [Robot Creates a Loop Above the Board with Gripper]

Peer - "For a circuit to work, all the pieces need to be connected together to form a loop like this. And the battery needs to be part of the loop."

Peer - "We need to connect all the pieces together in a circle such that they form a loop with the battery included in the loop."

Peer - "Using the blue wires, we should connect all the pieces together in a circle, and at the end of the circle we should have the battery."

Tutor - "For a circuit to work, all the pieces need to be connected together to form a loop like this. And the battery needs to be part of the loop."

Tutor - "You need to connect all the pieces together in a circle such that they form a loop with the battery included in the loop."

Tutor - "Using the blue wires, you should connect all the pieces together in a circle, and at the end of the circle you should have the battery."

Help Movement 12 [Robot points from Music Circuit to Positive Side of Battery]

Peer - "To power the music circuit, lets connect its positive to the positive on the battery, and its negative to the negative on the battery."

Peer - "The music circuit does not have power. Lets connect the positives and the negatives of the music circuit to the battery."

Peer - "For the music circuit to work, the positive of the music circuit should be connected to the positive of the battery."

Peer - "For the music circuit to work, the negative of the music circuit should be connected to the negative of the battery."

Tutor - "To power the music circuit, connect its positive to the positive on the battery, and its negative to the negative on the battery."

Tutor - "The music circuit does not have power. Connect the positives and the negatives of the music circuit to the battery."

Tutor - "For the music circuit to work, the positive of the music circuit should be connected to the positive of the battery."

Tutor - "For the music circuit to work, the negative of the music circuit should be connected to the negative of the battery."

Help Movement 13 [Robot points from Speaker to the Positive Side of Battery]

Peer - "For the speaker to work it should be connected on one side to the positive of the battery and the other side to the out port of the music circuit."

Peer - "Lets have the speaker in between the positive side of the battery and the out port of the music circuit."

Peer - "The speaker plays the music that the music circuit created. Lets connect the speaker to the out port of the music circuit."

Tutor - "For the speaker to work it should be connected on one side to the positive of the battery and the other side to the out port of the music circuit."

Tutor - "You should have the speaker in between the positive side of the battery and the out port of the music circuit."

Tutor - "The speaker plays the music that the music circuit created. Connect the speaker to the out port of the music circuit."

Help Movement 14 [Robot Points from Music Circuit to the Positive Side of the Battery]

Peer - "When you use your finger to spin the motor, the speaker should start playing music."

Peer - "The motor should be connected to the trigger port of the music circuit."

Peer - "Lets connect one side of the motor to the positive side of the battery, and the other side to the trigger port of the music circuit."

Tutor - "When you use your finger to spin the motor, the speaker should start playing music."

Tutor - "The motor should be connected to the trigger port of the music circuit."

Tutor - "Connect one side of the motor to the positive side of the battery, and the other side to the trigger port of the music circuit."

Help Movement 15 [Robot Points from Music Circuit to the Positive Side of the Battery]

Peer - "While we are holding the button down, the music should play. When we let it go it should stop."

Peer - "The button should be placed on the hold port of the music circuit."

Peer - "Lets have the button between the positive side of the battery and the hold port of the music circuit."

Tutor - "While you are holding the button down, the music should play. When you let it go it should stop."

Tutor - "The button should be placed on the hold port of the music circuit."

Tutor - "You should have the button between the positive side of the battery and the hold port of the music circuit."

Help Movement 16

Peer - "The circuit should only work if the button is pressed and the switch is turned on."

Peer - "Lets connect the switch and button together in the loop of the circuit."

Peer - "We should connect the end of the switch to one end of the button in this circuit."

Tutor - "The circuit should only work if the button is pressed and the switch is turned on."

Tutor - "Connect the switch and button together in the loop of the circuit."

Tutor - "You should connect the end of the switch to one end of the button in this circuit."

Help Movement 17

Peer - "In this task the circuit should work if either the button is pressed or the switch is on."

Peer - "To create the or behavior, we should split a wire in two, and connect the button and switch to each end, and then bring them together using wires."

Peer - "Lets try placing the button and switch side by side, and connecting their ends together. Then we can connect them to the battery and the rest of the circuit."

Tutor - "In this task the circuit should work if either the button is pressed or the switch is on."

Tutor - "To create the or behavior, you should split a wire in two, and connect the button and switch to each end, and then bring them together using wires."

Tutor - "Try placing the button and switch side by side, and connecting their ends together. Then you can connect them to the battery and the rest of the circuit."

Start of New Task

Peer - "Hmmm, lets see what we need to do for our next task."

Peer - "Looks like we have the next task, we can do this!"

Peer - "Next task is up!"

Peer - "We have our next task."

Peer - "Let's see what we need to do next."

Tutor - "Hmmm, lets see what you need to do for your next task."

Tutor - "Looks like you have the next task, you can do this!"

Tutor - "Next task is up!"

Tutor - "You have your next task."

Tutor - "Let's see what you need to do next."

Finished Task (correct)

Peer - "Yay, we did a great job. Lets clear up the board for the next task."

Peer - "Awesome, looks like we did that perfectly. Can you help me removing the pieces back onto the styrofoam?"

Peer - "Nice, we did the task just like it was meant. Lets clear up the pieces."

Peer - "Yay, go us. Can you take each piece and put it back onto the styrofoam?"

Peer - "Awesome, we make a good team."

Tutor - "Yay, you did a great job. Clear up the board for the next task."

Tutor - "Awesome, looks like you did that perfectly. Could you remove the pieces back onto the styrofoam?"

Tutor - "Nice, you did the task just like it was meant. Clear up the pieces."

Tutor - "Yay, go you!Can you take each piece and put it back onto the styrofoam?"

Tutor - "Awesome, you are a good student."

Finished Task (incorrect)

Peer - "Oh no, we were not able to finish it correctly in time. Lets clear up the board for the next one."

Peer - "We ran out of time. So we did not do this one, but hopefully we can do the next one."

Peer - "Times up. That was a hard one, hopefully the next one is easier. Please help me remove all the pieces back on to the styrofoam."

Peer - "Looks like we did that wrong and that we ran out of time."

Tutor - "Oh no, you were not able to finish it correctly in time. Clear up the board for the next one."

Tutor - "You ran out of time. So you did not do this one, but hopefully you can do the next one."

Tutor - "Times up. That was a hard one, hopefully the next one is easier. Please remove all the pieces back on to the styrofoam."

Tutor - "Looks like you did that wrong and that you ran out of time."

Out of Time

Peer - "It looks like there still is a mistake that we need to fix."

Peer - "We are not doing what the task asks."

Peer - "There are still some mistakes."

Peer - "Let's look at what the task is asking again."

Peer - "I don't think we are done yet."

Peer - "It's not fully working yet."

Peer - "Let's continue trying a bit more."

Tutor - "It looks like there still is a mistake that you need to fix."

Tutor - "You are not doing what the task asks."

Tutor - "There are still some mistakes."

Tutor - "Look at what the task is asking again."

Tutor - "I don't think you are done yet."

Tutor - "It's not fully working yet."

Tutor - "Continue trying a bit more."

Start of Session

Peer - "Hello, nice to meet you! I am excited to work with you today to create some electronic circuits. I am sure we will be a great team!"

Tutor - "Hello, nice to meet you! I am excited to teach you today on how to create some electronic circuits. I am sure you will be a great student!"

End of Session

Peer - "Looks like we are finished for today. Thanks so much for building these circuits with me, it was a lot of fun! Shall we call the experimenter back?"

Tutor - "Looks like you are finished for today. Thanks so much for building these circuits for me, it was a lot of fun teaching you! Shall we call the experimenter back?"

B.4 Pre-test and Post-test

The pre-test (1-6) and the post-test (7-12) questions were identical. The differences between the pre- and post-test were regarding the arrangement of pieces on the board.

- 1/7) Create a circuit that shines a constant light. [On Empty Board]
- 2/8) Create a circuit that will play music. [On Empty Board]
- 3/9) Add pieces to the circuit such that the circuit will play low music only while a button is being held down. [Figure B.1]

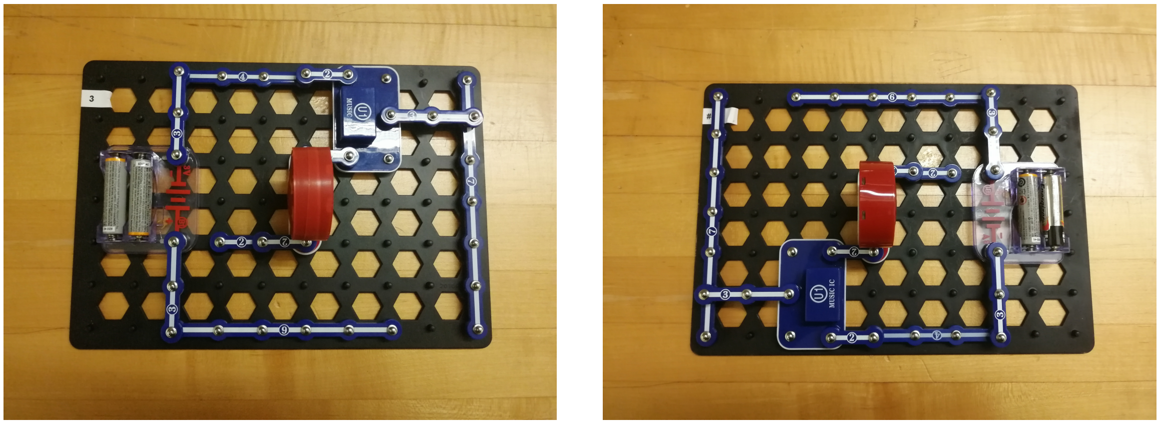


Figure B.1: The pre-test and post-test boards for tasks 3 and 9.

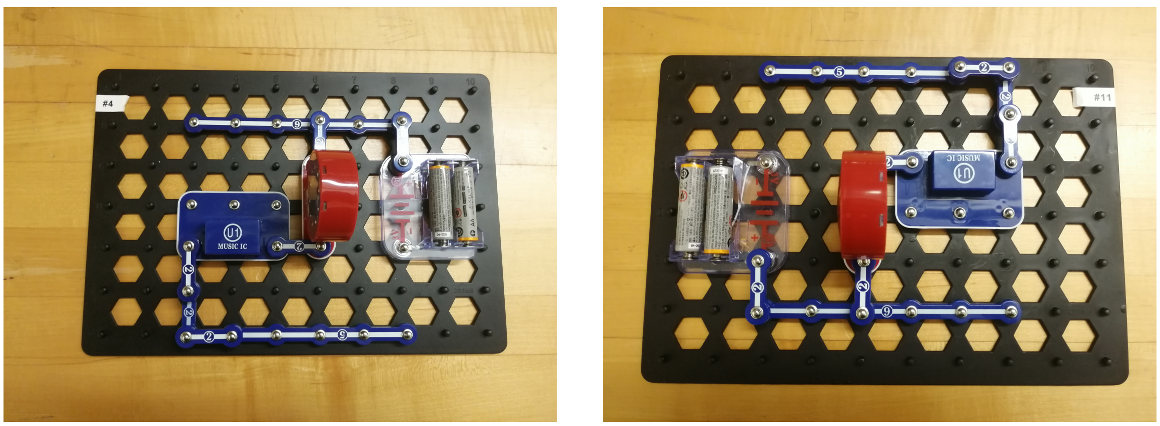


Figure B.2: The pre-test and post-test boards for tasks 4 and 10.

- 4/10) Add pieces to the circuit such that the circuit will play music whenever you spin a motor. [Figure B.2]

- 5/11) Please identify what the circuit does: [Picture of circuits with an AND gate.]

- 6/12) Please identify what the circuit does: [Picture of circuits with an OR gate.]

Bibliography

[1] Next-generation pose detection with movenet and tensorflow.js. https://blog.tensorflow.org/2021/05/next-generation-pose-detection-with-movenet-and-tensorflowjs.html.

[2] Jibo, Aug 2020.

[3] Dominic Abrams, Margaret Wetherell, Sandra Cochrane, Michael A Hogg, and John C Turner. Knowing what to think by knowing who you are: Self-categorization and the nature of norm formation, conformity and group polarization. *British journal of social psychology*, 29(2):97–119, 1990.

[4] Samy S Abu-Naser. Developing an intelligent tutoring system for students learning to program in c++. 2008.

[5] Timothy Adamson, Debasmita Ghose, Shannon C Yasuda, Lucas Jehu Silva Shepard, Michal A Lewkowicz, Joyce Duan, and Brian Scassellati. Why we should build robots that both teach and learn. In *Proceedings of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*, pages 187–196, 2021.

[6] Bashar G Al-Bastami and Samy S Abu Naser. Design and development of an intelligent tutoring system for c# language. *European academic research*, 4(10), 2017.

[7] Minoo Alemi, Ali Meghdari, and Maryam Ghazisaedy. Employing humanoid robots for teaching english language in iranian junior high-schools. *International Journal of Humanoid Robotics*, 11(03):1450022, 2014.

[8] Safinah Ali, Tyler Moroso, and Cynthia Breazeal. Can children learn creativity from a social robot? In *Proceedings of the 2019 on Creativity and Cognition*, pages 359–368. ACM, 2019.

[9] Katrin Allmendinger. Social presence in synchronous virtual learning situations: The role of nonverbal signals displayed by avatars. *Educational Psychology Review*, 22(1):41–56, 2010.

[10] Fehaid Alqahtani, Stamos Katsigiannis, and Naeem Ramzan. On the use of ecg and emg signals for question difficulty level prediction in the context of

intelligent tutoring systems. In *2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE)*, pages 392–396. IEEE, 2019.

[11] Patrícia Alves-Oliveira, Tiago Ribeiro, Sofia Petisca, Eugenio Di Tullio, Francisco S Melo, and Ana Paiva. An empathic robotic tutor for school classrooms: Considering expectation and satisfaction of children as end-users. In *International Conference on Social Robotics*, pages 21–30. Springer, 2015.

[12] John R Anderson, C Franklin Boyle, and Brian J Reiser. Intelligent tutoring systems. *Science*, 228(4698):456–462, 1985.

[13] John R Anderson, Albert T Corbett, Kenneth R Koedinger, and Ray Pelletier. Cognitive tutors: Lessons learned. *The journal of the learning sciences*, 4(2):167–207, 1995.

[14] Linda F Annis. The processes and effects of peer tutoring. *Human Learning: Journal of Practical Research & Applications*, 1983.

[15] Solomon E Asch. Studies of independence and conformity: I. a minority of one against a unanimous majority. *Psychological monographs: General and applied*, 70(9):1, 1956.

[16] Karl J Astrom. Optimal control of markov processes with incomplete state information. *Journal of mathematical analysis and applications*, 10(1):174–205, 1965.

[17] Omri Avioz-Sarig, Samuel Olatunji, Vardit Sarne-Fleischmann, and Yael Edan. Robotic system for physical training of older adults. *International Journal of Social Robotics*, pages 1–16, 2020.

[18] Omer A. Awan. Peer to peer learning: Its importance and benefits. *Academic Radiology*, 28(5):747–748, 2021.

[19] Wilma A Bainbridge, Justin Hart, Elizabeth S Kim, and Brian Scassellati. The effect of presence on human-robot interaction. In *RO-MAN 2008-The 17th IEEE International Symposium on Robot and Human Interactive Communication*, pages 701–706. IEEE, 2008.

[20] Wilma A Bainbridge, Justin W Hart, Elizabeth S Kim, and Brian Scassellati. The benefits of interactions with physically present robots over video-displayed agents. *International Journal of Social Robotics*, 2011.

[21] Tadas Baltrusaitis, Peter Robinson, and Louis-Philippe Morency. Constrained local neural fields for robust facial landmark detection in the wild. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 354–361, 2013.

[22] John A Bargh and Yaacov Schul. On the cognitive benefits of teaching. *Journal of Educational Psychology*, 72(5):593, 1980.

[23] Robert S Baron, Joseph A Vandello, and Bethany Brunsman. The forgotten variable in conformity research: Impact of task importance on social influence. *Journal of personality and social psychology*, 71(5):915, 1996.

[24] Christoph Bartneck, Elizabeth Croft, and Dana Kulic. Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots. *International Journal of Social Robotics*, 1(1):71–81, 2009.

[25] R Barker Bausell, William B Moody, and F Neil Walzl. A factorial study of tutoring versus classroom instruction. *American Educational Research Journal*, 9(4):591–597, 1972.

[26] Paul Baxter, Emily Ashurst, Robin Read, James Kennedy, and Tony Belpaeme. Robot education peers in a situated primary school study: Personalisation promotes child learning. *PLOS ONE*, 12(5):1–23, 2017. Publisher: Public Library of Science.

[27] Jessica L Bean and Inge-Marie Eigsti. Assessment of joint attention in school-age children and adolescents. *Research in Autism Spectrum Disorders*, 6(4):1304–1310, 2012.

[28] Clay Beckner, Péter Rácz, Jennifer Hay, Jürgen Brandstetter, and Christoph Bartneck. Participants conform to humans but not to humanoid robots in an english past tense formation task. *Journal of Language and Social Psychology*, 35(2):158–179, 2016.

[29] Richard Bellman. Dynamic programming. *Science*, 153(3731):34–37, 1966.

[30] Tony Belpaeme, Paul Baxter, Robin Read, Rachel Wood, Heriberto Cuayáhuitl, Bernd Kiefer, Stefania Racioppa, Ivana Kruijff-Korbayová, Georgios Athanasopoulos, Valentin Enescu, et al. Multimodal child-robot interaction: Building social bonds. *Journal of Human-Robot Interaction*, 1(2), 2012.

[31] Tony Belpaeme, James Kennedy, Aditi Ramachandran, Brian Scassellati, and Fumihide Tanaka. Social robots for education: A review. *Science robotics*, 3(21):eaat5954, 2018.

[32] Gary Bishop, Greg Welch, et al. An introduction to the kalman filter. *Proc of SIGGRAPH, Course*, 8:41, 2001.

[33] Benjamin S Bloom. The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational researcher*, 13(6):4–16, 1984.

[34] Laura Boccanfuso, Elizabeth S Kim, James C Snider, Quan Wang, Carla A Wall, Lauren DiNicola, Gabriella Greco, Lilli Flink, Sharlene Lansiquot, Pamela Ventola, et al. Autonomously detecting interaction with an affective robot to

explore connection to developmental ability. In *2015 International Conference on Affective Computing and Intelligent Interaction (ACII)*, pages 1–7. IEEE, 2015.

[35] Rod Bond and Peter B Smith. Culture and conformity: A meta-analysis of studies using asch's (1952b, 1956) line judgment task. *Psychological bulletin*, 119(1):111, 1996.

[36] Jürgen Brandstetter, Clay Beckner, Eduardo Benitez Sandoval, and Christoph Bartneck. Persistent lexical entrainment in hri. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 63–72. ACM, 2017.

[37] Jürgen Brandstetter, Péter Rácz, Clay Beckner, Eduardo B Sandoval, Jennifer Hay, and Christoph Bartneck. A peer pressure experiment: Recreation of the asch conformity experiment with robots. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1335–1340. IEEE, 2014.

[38] LaVonda Brown, Ryan Kerwin, and Ayanna M Howard. Applying behavioral strategies for student engagement using a robotic educational agent. In *2013 IEEE international conference on systems, man, and cybernetics*, pages 4360–4365. IEEE, 2013.

[39] Carolien Bulte, Aaron Betts, Kathryn Garner, and Steven Durning. Student teaching: views of student near-peer teachers and learners. *Medical teacher*, 29(6):583–590, 2007.

[40] Cory J Butz, Shan Hua, and R Brien Maguire. A web-based bayesian intelligent tutoring system for computer programming. *Web Intelligence and Agent Systems: An International Journal*, 4(1):77–97, 2006.

[41] Breanne J Byiers, Joe Reichle, and Frank J Symons. Single-subject experimental design for evidence-based practice. *American Journal of Speech Language Pathology*, 2012.

[42] Colleen M Carpinella, Alisa B Wyman, Michael A Perez, and Steven J Stroessner. The robotic social attributes scale (rosas) development and validation. In *Proceedings of the 2017 ACM/IEEE International Conference on human-robot interaction*, pages 254–262, 2017.

[43] Ginevra Castellano, André Pereira, Iolanda Leite, Ana Paiva, and Peter W McOwan. Detecting user engagement with a robot companion using task and social interaction-based features. In *Proceedings of the 2009 international conference on Multimodal interfaces*, pages 119–126, 2009.

[44] Hao Cen, Kenneth Koedinger, and Brian Junker. Learning factors analysis–a general method for cognitive model evaluation and improvement. In *International Conference on Intelligent Tutoring Systems*, pages 164–175. Springer, 2006.

[45] Hao Cen, Kenneth Koedinger, and Brian Junker. Comparing two irt models for conjunctive skills. In *International Conference on Intelligent Tutoring Systems*, pages 796–798. Springer, 2008.

[46] Suleyman Cetintas, Luo Si, Yan Ping Ping Xin, and Casey Hord. Automatic detection of off-task behaviors in intelligent tutoring systems with machine learning techniques. *IEEE Transactions on Learning Technologies*, 3(3):228–236, 2009.

[47] Shruti Chandra, Raul Paradeda, Hang Yin, Pierre Dillenbourg, Rui Prada, and Ana Paiva. Do children perceive whether a robotic peer is learning or not? In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, pages 41–49. ACM, 2018.

[48] Frances S Chen, Julia A Minson, Maren Schöne, and Markus Heinrichs. In the eye of the beholder: Eye contact increases resistance to persuasion. *Psychological science*, 24(11):2254–2261, 2013.

[49] Huili Chen, Hae Won Park, and Cynthia Breazeal. Teaching and learning with children: Impact of reciprocal peer learning with a social robot on children's learning and emotive engagement. *Computers & Education*, 150:103836, 2020.

[50] Huili Chen, Hae Won Park, Xiajie Zhang, and Cynthia Breazeal. Impact of interaction context on the student affect-learning relationship in child-robot interaction. In *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, pages 389–397, 2020.

[51] Vijay Chidambaram, Yueh-Hsuan Chiang, and Bilge Mutlu. Designing persuasive robots: how robots might persuade people using vocal and nonverbal cues. In *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, pages 293–300. ACM, 2012.

[52] Stuart R Chipkin, Serena A Klugh, and Lisa Chasan-Taber. Exercise and diabetes. *Cardiology clinics*, 19(3):489–505, 2001.

[53] Robert Cialdini. *Pre-suasion: A revolutionary way to influence and persuade*. Simon and Schuster, 2016.

[54] Robert B Cialdini. *Influence*, volume 3. A. Michel Port Harcourt, 1987.

[55] Robert B Cialdini and Noah J Goldstein. Social influence: Compliance and conformity. *Annu. Rev. Psychol.*, 55:591–621, 2004.

[56] Benjamin Clement, Didier Roy, Pierre-Yves Oudeyer, and Manuel Lopes. Multi-armed bandits for intelligent tutoring systems. *arXiv preprint arXiv:1310.3174*, 2013.

[57] Joel B Cohen and Ellen Golden. Informational social influence and product evaluation. *Journal of applied Psychology*, 56(1):54, 1972.

[58] Jannette Collins. Education techniques for lifelong learning: principles of adult learning. *Radiographics*, 24(5):1483–1489, 2004.

[59] Cristina Conati, Abigail Gertner, and Kurt Vanlehn. Using bayesian networks to manage uncertainty in student modeling. *User modeling and user-adapted interaction*, 12(4):371–417, 2002.

[60] Joe Connolly, Viola Mocz, Nicole Salomons, Joseph Valdez, Nathan Tsoi, Brian Scassellati, and Marynel Vázquez. Prompting prosocial human interventions in response to robot mistreatment. In *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, pages 211–220. ACM, 2020.

[61] Albert T Corbett and John R Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4(4):253–278, 1994.

[62] Albert T Corbett, Kenneth R Koedinger, and John R Anderson. Intelligent tutoring systems. In *Handbook of human-computer interaction*, pages 849–874. Elsevier, 1997.

[63] Philip R Costanzo and Marvin E Shaw. Conformity as a function of age level. *Child development*, pages 967–975, 1966.

[64] Arturo Cruz-Maya and Adriana Tapus. Learning users' and personality-gender preferences in close human-robot interaction. In *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 791–798. IEEE, 2017.

[65] Yossi Ben David, Avi Segal, and Ya'akov Kobi Gal. Sequencing educational content in classrooms using bayesian knowledge tracing. In *Proceedings of the sixth international conference on Learning Analytics & Knowledge*, pages 354–363. ACM, 2016.

[66] Maartje MA de Graaf, Somaya Ben Allouch, and Jan AGM van Dijk. Long-term evaluation of a social robot in real homes. *Interaction studies*, 17(3):462–491, 2016.

[67] Jan de Wit, Thorsten Schodde, Bram Willemsen, Kirsten Bergmann, Mirjam de Haas, Stefan Kopp, Emiel Krahmer, and Paul Vogt. The effect of a robot's gestures and adaptive tutoring on children's acquisition of second language vocabularies. In *2018 13th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 50–58. IEEE, 2018.

[68] Eric Deng, Bilge Mutlu, and Maja Mataric. Embodiment in socially interactive robots. *Foundations and Trens in Robotics*, 2019.

[69] Michel C Desmarais and Ryan S Baker. A review of recent advances in learner and skill modeling in intelligent learning environments. *User Modeling and User-Adapted Interaction*, 22(1-2):9–38, 2012.

[70] Morton Deutsch and Harold B Gerard. A study of normative and informational social influences upon individual judgment. *The journal of abnormal and social psychology*, 51(3):629, 1955.

[71] Cinzia Di Dio, Federico Manzi, Giulia Peretti, Angelo Cangelosi, Paul L Harris, Davide Massaro, and Antonella Marchetti. Shall i trust you? from child–robot interaction to trusting relationships. *Frontiers in Psychology*, 11:469, 2020.

[72] Joshua J Diehl, Lauren M Schmitt, Michael Villano, and Charles R Crowell. The clinical use of robots for individuals with autism spectrum disorders: A critical review. *Research in autism spectrum disorders*, 6(1):249–262, 2012.

[73] Dimiter M Dimitrov and Phillip D Rumrill Jr. Pretest-posttest designs and measurement of change. *Work*, 20(2):159–165, 2003.

[74] Sidney D'Mello, Andrew Olney, Claire Williams, and Patrick Hays. Gaze tutor: A gaze-reactive intelligent tutoring system. *International Journal of human-computer studies*, 70(5):377–398, 2012.

[75] Michael John Dwyer, Margherita Pasini, Stefano De Dominicis, and Elda Righi. Physical activity: Benefits and challenges during the covid-19 pandemic. *Scandinavian journal of medicine & science in sports*, 30(7):1291, 2020.

[76] Alice H Eagly and Carole Chrvala. Sex differences in conformity: Status and gender role interpretations. *Psychology of Women Quarterly*, 10(3):203–220, 1986.

[77] Fifth Edition et al. Diagnostic and statistical manual of mental disorders. *Am Psychiatric Assoc*, 21:591–643, 2013.

[78] Elenco. Snap circuits. `https://www.elenco.com/brand/snap-circuits/`, 2021. Accessed: 2019-9-10.

[79] CD Elliott. Das administration and scoring manual. *San Antonio, TX: Psychological Corporation*, 1990.

[80] Mark Elsom-Cook. Student modelling in intelligent tutoring systems. *Artificial Intelligence Review*, 7(3):227–240, 1993.

[81] Bryn Evans and Edward Reynolds. The organization of corrective demonstrations using embodied action in sports coaching feedback. *Symbolic Interaction*, 39(4):525–556, 2016.

[82] Martha Evens, Ru-Charn Chang, Yoon Hee Lee, Leem Seop Shim, Chong Woo Woo, and Yuemei Zbang. Circsim-tutor: An intelligent tutoring system using natural language dialogue. In *Fifth Conference on Applied Natural Language Processing: Descriptions of System Demonstrations and Videos*, pages 13–14, 1997.

[83] Friederike Eyssel, Laura De Ruiter, Dieta Kuchenbrandt, Simon Bobinger, and Frank Hegel. ‘if you sound like me, you must be more human’: On the interplay of robot and user features on human-robot acceptance and anthropomorphism. In *2012 7th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 125–126. IEEE, 2012.

[84] Friederike Eyssel and Dieta Kuchenbrandt. Social categorization of social robots: Anthropomorphism as a function of robot group membership. *British Journal of Social Psychology*, 51(4):724–731, 2012.

[85] John W Fantuzzo, Ronald E Riggio, Sharon Connelly, and Linda A Dimeff. Effects of reciprocal peer tutoring on academic achievement and psychological adjustment: A component analysis. *Journal of educational psychology*, 81(2):173, 1989.

[86] Juan Fasola and Maja J Mataric. Using socially assistive human–robot interaction to motivate physical exercise for older adults. *Proceedings of the IEEE*, 100(8):2512–2526, 2012.

[87] Juan Fasola and Maja J Matarić. A socially assistive robot exercise coach for the elderly. *Journal of Human-Robot Interaction*, 2(2):3–32, 2013.

[88] David Feil-Seifer and Maja J Mataric. Defining socially assistive robotics. In *9th International Conference on Rehabilitation Robotics, 2005. ICORR 2005.*, pages 465–468. IEEE, 2005.

[89] Ronit Feingold Polak and Shelly Levy Tzedek. Social robot for rehabilitation: Expert clinicians and post-stroke patients’ evaluation following a long-term intervention. In *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, pages 151–160, 2020.

[90] Logan Fiorella and Richard E Mayer. The relative benefits of learning by teaching and teaching expectancy. *Contemporary Educational Psychology*, 38(4):281–288, 2013.

[91] Naomi T Fitter, Mayumi Mohan, Katherine J Kuchenbecker, and Michelle J Johnson. Exercising with baxter: preliminary support for assistive social-physical human-robot interaction. *Journal of neuroengineering and rehabilitation*, 17(1):1–22, 2020.

[92] Jeremiah T Folsom-Kovarik, Gita Sukthankar, and Sae Schatz. Tractable pomdp representations for intelligent tutoring systems. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4(2):1–22, 2013.

[93] Markus Forsberg. Why is speech recognition difficult. *Chalmers University of Technology*, 2003.

[94] Philippe Fournier-Viger, Roger Nkambou, and Engelbert Mephu Nguifo. Building intelligent tutoring systems for ill-defined domains. In *Advances in intelligent tutoring systems*. Springer, 2010.

[95] Barbara A Fox. Cognitive and interactional aspects of correction in. *Teaching knowledge and intelligent tutoring*, page 149, 1991.

[96] Dorothée François, Stuart Powell, and Kerstin Dautenhahn. A long-term study of children with autism playing with a robotic pet: Taking inspirations from non-directive play therapy to encourage children's proactivity and initiative-taking. *Interaction Studies*, 10(3):324–373, 2009.

[97] Marlena R Fraune, Yusaku Nishiwaki, Selma Sabanović, Eliot R Smith, and Michio Okada. Threatening flocks and mindful snowflakes: How group entitativity affects perceptions of robots. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 205–213. ACM, 2017.

[98] Aimi S Ghazali, Jaap Ham, Emilia I Barakova, and Panos Markopoulos. Poker face influence: Persuasive robot with minimal social cues triggers less psychological reactance. In *2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 940–946. IEEE, 2018.

[99] Victor Girotto, Cecil Lozano, Kasia Muldner, Winslow Burleson, and Erin Walker. Lessons learned from in-school use of rtag: A robo-tangible learning environment. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 919–930, 2016.

[100] Rachel Gockley, Allison Bruce, Jodi Forlizzi, Marek Michalowski, Anne Mundell, Stephanie Rosenthal, Brennan Sellner, Reid Simmons, Kevin Snipes, Alan C Schultz, et al. Designing robots for long-term social interaction. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1338–1343. IEEE, 2005.

[101] Yue Gong, Joseph E Beck, and Neil T Heffernan. Comparing knowledge tracing and performance factor analysis by using multiple model fitting procedures. In *International conference on intelligent tutoring systems*, pages 35–44. Springer, 2010.

[102] José González-Brenes, Yun Huang, and Peter Brusilovsky. General features in knowledge tracing to model multiple subskills, temporal item response theory, and expert knowledge. In *The 7th International Conference on Educational Data Mining*, pages 84–91. University of Pittsburgh, 2014.

[103] Goren Gordon and Cynthia Breazeal. Bayesian active learning-based robot tutor for children's word-reading skills. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

[104] Goren Gordon, Samuel Spaulding, Jacqueline Kory Westlund, Jin Joo Lee, Luke Plummer, Marayna Martinez, Madhurima Das, and Cynthia Breazeal. Affective personalization of a social robot tutor for children's second language skills. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

[105] Arthur C Graesser, Patrick Chipman, Brian C Haynes, and Andrew Olney. Autotutor: An intelligent tutoring system with mixed-initiative dialogue. *IEEE Transactions on Education*, 48(4):612–618, 2005.

[106] Mark A Guadagnoli and Timothy D Lee. Challenge point: a framework for conceptualizing the effects of various practice conditions in motor learning. *Journal of motor behavior*, 36(2):212–224, 2004.

[107] Arzu Guneysu and Bert Arnrich. Socially assistive child-robot interaction in physical exercise coaching. In *2017 26th IEEE international symposium on robot and human interactive communication (RO-MAN)*, pages 670–675. IEEE, 2017.

[108] Arzu Guneysu Ozgur, Maximilian Jonas Wessel, Wafa Johal, Kshitij Sharma, Ayberk Özgür, Philippe Vuadens, Francesco Mondada, Friedhelm Christoph Hummel, and Pierre Dillenbourg. Iterative design of an upper limb rehabilitation game with tangible robots. In *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2018.

[109] Marc Hanheide, Denise Hebesberger, and Tomáš Krajník. The when, where, and how: An adaptive robotic info-terminal for care home residents. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 341–349, 2017.

[110] Markus Häring, Dieta Kuchenbrandt, and Elisabeth André. Would you like to play with me? how robots' group membership and task features influence human–robot interaction. In *2014 9th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 9–16. IEEE, 2014.

[111] J Hemminki and A Erkinheimo-Kyllönen. A humanoid robot as a language tutor. *Proceedings of R4L@ HRI2017*, 2017.

[112] Olivier A Blanson Henkemans, Bert PB Bierman, Joris Janssen, Mark A Neerincx, Rosemarijn Looije, Hanneke van der Bosch, and Jeanine AM van der Giessen. Using a robot to personalise health education for children with diabetes type 1: A pilot study. *Patient education and counseling*, 92(2):174–181, 2013.

[113] Nicholas Hertz and Eva Wiese. Influence of agent type and task ambiguity on conformity in social decision making. In *Proceedings of the Human Factors*

and Ergonomics Society Annual Meeting, volume 60, pages 313–317. SAGE Publications Sage CA: Los Angeles, CA, 2016.

[114] Nicholas Hertz and Eva Wiese. Under pressure: examining social conformity with computer and robot groups. *Human factors*, 60(8):1207–1218, 2018.

[115] Bert H Hodges and Anne L Geyer. A nonconformist account of the asch experiments: Values, pragmatics, and moral dilemmas. *Personality and Social Psychology Review*, 10(1):2–19, 2006.

[116] Deanna Hood, Séverin Lemaignan, and Pierre Dillenbourg. When children teach a robot to write: An autonomous teachable humanoid which uses simulated handwriting. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*, HRI '15, pages 83–90. Association for Computing Machinery, 2015.

[117] Stephen Hutt, Caitlin Mills, Shelby White, Patrick J Donnelly, and Sidney K D'Mello. The eyes have it: Gaze-based detection of mind wandering during learning with an intelligent tutoring system. *International Educational Data Mining Society*, 2016.

[118] Alexis Jacq, Séverin Lemaignan, Fernando Garcia, Pierre Dillenbourg, and Ana Paiva. Building successful long child-robot interactions in a learning context. In *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 239–246, 2016. ISSN: 2167-2148.

[119] Joris B Janssen, Chrissy C Wal, Mark A Neerincx, and Rosemarijn Looije. Motivating children to learn arithmetic with an adaptive robot game. In *International conference on social robotics*, pages 153–162. Springer, 2011.

[120] Sooyeon Jeong, Sharifa Alghowinem, Laura Aymerich-Franch, Kika Arias, Agata Lapedriza, Rosalind Picard, Hae Won Park, and Cynthia Breazeal. A robotic positive psychology coach to improve college students' wellbeing. In *2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, pages 187–194. IEEE, 2020.

[121] Aidan Jones, Susan Bull, and Ginevra Castellano. "i know that now, i'm going to learn this next" promoting self-regulated learning with a robotic tutor. *International Journal of Social Robotics*, 10(4):439–454, 2018.

[122] Aidan Jones and Ginevra Castellano. Adaptive robotic tutors that support self-regulated learning: A longer-term investigation with primary school children. *International Journal of Social Robotics*, 10(3):357–370, 2018.

[123] Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2):99–134, 1998.

[124] Chieko Kanai, Gabor Toth, Miho Kuroda, Atsuko Miyake, and Takashi Itahashi. Social skills in autism spectrum disorders. In *Handbook of social behavior and skills in children*, pages 217–248. Springer, 2017.

[125] Takayuki Kanda, Takayuki Hirano, Daniel Eaton, and Hiroshi Ishiguro. Interactive robots as social partners and peer tutors for children: A field trial. *Human–Computer Interaction*, 19(1-2):61–84, 2004.

[126] Takayuki Kanda, Rumi Sato, Naoki Saiwaki, and Hiroshi Ishiguro. A two-month field trial in an elementary school for long-term human–robot interaction. *IEEE Transactions on robotics*, 23(5):962–971, 2007.

[127] Martin F Kaplan and Charles E Miller. Group decision making and normative versus informational influence: Effects of type of issue and assigned decision rule. *Journal of Personality and social psychology*, 53(2):306, 1987.

[128] Herbert C Kelman. Processes of opinion change. In *Attitude Change*, pages 205–233. Routledge, 2017.

[129] James Kennedy, Paul Baxter, and Tony Belpaeme. The robot who tried too hard: Social behaviour of a robot tutor can negatively affect child learning. In *2015 10th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 67–74. IEEE, 2015.

[130] James Kennedy, Paul Baxter, Emmanuel Senft, and Tony Belpaeme. Higher nonverbal immediacy leads to greater learning gains in child-robot tutoring interactions. In *International conference on social robotics*, pages 327–336. Springer, 2015.

[131] Leonid Keselman, John Iselin Woodfill, Anders Grunnet-Jepsen, and Achintya Bhowmik. Intel realsense stereoscopic depth cameras. *CoRR*, abs/1705.05548, 2017.

[132] Cory D Kidd and Cynthia Breazeal. Effect of a robot on user perceptions. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2004.

[133] Cory D Kidd and Cynthia Breazeal. Robots at home: Understanding long-term human-robot interaction. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3230–3235. IEEE, 2008.

[134] Elizabeth Kim, Rhea Paul, Frederick Shic, and Brian Scassellati. Bridging the research gap: Making hri useful to individuals with autism. *Journal of Human-Robot Interaction*, 2012.

[135] Elizabeth S Kim, Lauren D Berkovits, Emily P Bernier, Dan Leyzberg, Frederick Shic, Rhea Paul, and Brian Scassellati. Social robots as embedded reinforcers of social behavior in children with autism. *Journal of autism and developmental disorders*, 43(5):1038–1049, 2013.

[136] Chris L Kleinke. Gaze and eye contact: a research review. *Psychological bulletin*, 100(1):78, 1986.

[137] Kheng Lee Koay, Dag Sverre Syrdal, Michael L Walters, and Kerstin Dautenhahn. Living with robots: Investigating the habituation effect in participants' preferences during a longitudinal human-robot interaction study. In *RO-MAN 2007-The 16th IEEE International Symposium on Robot and Human Interactive Communication*, pages 564–569. IEEE, 2007.

[138] Ekaterina Kochmar, Dung Do Vu, Robert Belfer, Varun Gupta, Iulian Vlad Serban, and Joelle Pineau. Automated data-driven generation of personalized pedagogical interventions in intelligent tutoring systems. *International Journal of Artificial Intelligence in Education*, pages 1–27, 2021.

[139] Jacqueline M Kory-Westlund and Cynthia Breazeal. A long-term study of young children's rapport, social emulation, and language learning with a peer-like robot playmate in preschool. *Frontiers in Robotics and AI*, 6:81, 2019.

[140] Hatice Kose-Bagci, Ester Ferrari, Kerstin Dautenhahn, Dag Sverre Syrdal, and Chrystopher L Nehaniv. Effects of embodiment and gestures on social interaction in drumming games with a humanoid robot. *Advanced Robotics*, 23(14):1951–1996, 2009.

[141] Austin Kothig, John Munoz, Sami Alperen Akgun, Alexander M Aroyo, and Kerstin Dautenhahn. Connecting humans and robots using physiological signals–closing-the-loop in hri. In *2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN)*, pages 735–742. IEEE, 2021.

[142] Hideki Kozima, Marek P Michalowski, and Cocoro Nakagawa. Keepon. *International Journal of Social Robotics*, 1(1):3–18, 2009.

[143] Hideki Kozima, Cocoro Nakagawa, and Yuriko Yasuda. Children–robot interaction: a pilot study in autism therapy. *Progress in brain research*, 164:385–400, 2007.

[144] James A Kulik and JD Fletcher. Effectiveness of intelligent tutoring systems: a meta-analytic review. *Review of educational research*, 86(1):42–78, 2016.

[145] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[146] Sonya S Kwak, Yunkyung Kim, Eunho Kim, Christine Shin, and Kwangsu Cho. What makes people empathize with an emotional robot?: The impact of agency and physical embodiment on human empathy for a robot. In *2013 IEEE RO-MAN*, 2013.

[147] Minae Kwon, Malte F Jung, and Ross A Knepper. Human expectations of social robots. In *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 463–464. IEEE, 2016.

[148] Andrew S Lan and Richard G Baraniuk. A contextual bandits framework for personalized learning action selection. In *EDM*, pages 424–429, 2016.

[149] Nathaniel Lasry, Eric Mazur, and Jessica Watkins. Peer instruction: From harvard to the two-year college. *American journal of Physics*, 76(11):1066–1069, 2008.

[150] Bibb Latane, Steve Nida, et al. Social impact theory and group influence: A social engineering perspective. *Psychology of group influence*, pages 3–34, 1980.

[151] Iolanda Leite, Ginevra Castellano, André Pereira, Carlos Martinho, and Ana Paiva. Empathic robots for long-term interaction. *International Journal of Social Robotics*, 6(3):329–341, 2014.

[152] Iolanda Leite, Carlos Martinho, Andre Pereira, and Ana Paiva. As time goes by: Long-term evaluation of social presence in robotic companions. In *RO-MAN 2009-the 18th IEEE international symposium on robot and human interactive communication*, pages 669–674. IEEE, 2009.

[153] Iolanda Leite, Marissa McCoy, Monika Lohani, Nicole Salomons, Kara McElvaine, Charlene Stokes, Susan Rivers, and Brian Scassellati. Autonomous disengagement classification and repair in multiparty child-robot interaction. In *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 525–532. IEEE, 2016.

[154] Iolanda Leite, Marissa McCoy, Monika Lohani, Daniel Ullman, Nicole Salomons, Charlene Stokes, Susan Rivers, and Brian Scassellati. Emotional storytelling in the classroom: Individual versus group interaction between children and robots. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*, pages 75–82, 2015.

[155] Iolanda Leite, Marissa McCoy, Daniel Ullman, Nicole Salomons, and Brian Scassellati. Comparing models of disengagement in individual and group interactions. In *2015 10th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 99–105. IEEE, 2015.

[156] James C Lester, Sharolyn A Converse, Susan E Kahler, S Todd Barlow, Brian A Stone, and Ravinder S Bhogal. The persona effect: affective impact of animated pedagogical agents. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*, pages 359–366, 1997.

[157] Daniel Leyzberg, Samuel Spaulding, and Brian Scassellati. Personalizing robot tutors to individuals' learning differences. In *2014 9th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 423–430. IEEE, 2014.

[158] Daniel Leyzberg, Samuel Spaulding, Mariya Toneva, and Brian Scassellati. The physical presence of a robot tutor increases cognitive learning gains. In *Proceedings of the annual meeting of the cognitive science society*, volume 34, 2012.

[159] Jamy Li. The benefit of being physically present: A survey of experimental works comparing copresent robots, telepresent robots and virtual agents. *International Journal of Human-Computer Studies*, 77:23–37, 2015.

[160] Libellud. Dixit. Card game, 2008.

[161] Diane Litman and Scott Silliman. Itspoke: An intelligent tutoring spoken dialogue system. In *Demonstration papers at HLT-NAACL 2004*, pages 5–8, 2004.

[162] Alexandru Litoiu and Brian Scassellati. Robotic coaching of complex physical skills. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction Extended Abstracts*, pages 211–212, 2015.

[163] Catherine Lord, Michael Rutter, Pamela C DiLavore, Susan Risi, Katherine Gotham, Somer L Bishop, et al. Ados. *Autism diagnostic observation schedule. Manual. Los Angeles: WPS*, 1999.

[164] Catherine Lord, Michael Rutter, and Ann Le Couteur. Autism diagnostic interview-revised: a revised version of a diagnostic interview for caregivers of individuals with possible pervasive developmental disorders. *Journal of autism and developmental disorders*, 24(5):659–685, 1994.

[165] Kenneth R Lord, Myung-Soo Lee, and Peggy Choong. Differences in normative and informational social influence. *ACR North American Advances*, 2001.

[166] Todd Lucas, Sheldon Alexander, Ira J Firestone, and Boris B Baltes. Self-efficacy and independence from social influence: Discovery of an efficacy–difficulty effect. *Social Influence*, 1(1):58–80, 2006.

[167] Virginia H Mackintosh, Robin P Goin-Kochel, and Barbara J Myers. "what do you like/dislike about the treatments you're currently using?" a qualitative study of parents of children with autism spectrum disorders. *Focus on Autism and Other Developmental Disabilities*, 27(1):51–60, 2012.

[168] Egil W Martinsen. Benefits of exercise for the treatment of depression. *Sports Medicine*, 9(6):380–389, 1990.

[169] Maja J Matarić and Brian Scassellati. Socially assistive robotics. *Springer handbook of robotics*, pages 1973–1994, 2016.

[170] Shizuko Matsuzoe and Fumihide Tanaka. How smartly should robots behave?: Comparative investigation on the learning ability of a care-receiving robot. In *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*, pages 339–344. IEEE, 2012.

[171] David A McAllester and Satinder Singh. Approximate planning for factored pomdps using belief state simplification. *arXiv preprint arXiv:1301.6719*, 2013.

[172] Monica M McGill. Learning to program with personal robots: Influences on student motivation. *ACM Transactions on Computing Education (TOCE)*, 12(1):1–32, 2012.

[173] Douglas C Merrill, Brian J Reiser, Michael Ranney, and J Gregory Trafton. Effective tutoring techniques: A comparison of human tutors and intelligent tutoring systems. *The Journal of the Learning Sciences*, 2(3):277–305, 1992.

[174] Janet Metcalfe and Nate Kornell. A region of proximal learning model of study time allocation. *Journal of Memory and Language*, 52(4):463–477, 2005.

[175] Deborah JC Meyer and Heather C Anderson. Preadolescents and apparel purchasing: Conformity to parents and peers in the consumer socialization process. *Journal of Social Behavior and Personality*, 15(2):243, 2000.

[176] François Michaud, Tamie Salter, Audrey Duquette, Henri Mercier, Michel Lauria, Helene Larouche, and Francois Larose. Assistive technologies and child-robot interaction. In *AAAI spring symposium on multidisciplinary collaboration for socially assistive robotics*, 2007.

[177] Carey K Morewedge, Jesse J Chandler, Robert Smith, Norbert Schwarz, and Jonathan Schooler. Lost in the crowd: Entitative group membership reduces mind attribution. *Consciousness and cognition*, 22(4):1195–1205, 2013.

[178] Samer Al Moubayed, Jens Edlund, and Jonas Beskow. Taming mona lisa: communicating gaze faithfully in 2d and 3d facial projections. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 1(2):1–25, 2012.

[179] Javier Movellan, Micah Eckhardt, Marjo Virnes, and Angelica Rodriguez. Sociable robot improves toddler vocabulary skills. In *Proceedings of the 4th ACM/IEEE international conference on Human robot interaction*, pages 307–308, 2009.

[180] R Charles Murray and Kurt VanLehn. A comparison of decision-theoretic, fixed-policy and random tutorial action selection. In *International conference on intelligent tutoring systems*, pages 114–123. Springer, 2006.

[181] James Michael Niehaus, Boyang Li, and Mark Riedl. Automated scenario adaptation in support of intelligent tutoring systems. In *Twenty-Fourth International FLAIRS Conference*, 2011.

[182] William S Noble. What is a support vector machine? *Nature biotechnology*, 24(12):1565–1567, 2006.

[183] Sharmilee M Nyenhuis, Justin Greiwe, Joanna S Zeiger, Anil Nanda, and Andrew Cooke. Exercise and fitness in the age of social distancing during the covid-19 pandemic. *The journal of allergy and clinical immunology. In practice*, 8(7):2152, 2020.

[184] Matthew A Nystoriak and Aruni Bhatnagar. Cardiovascular effects and benefits of exercise. *Frontiers in cardiovascular medicine*, 2018.

[185] Angela M O'Donnell and Alison King. *Cognitive perspectives on peer learning*. Routledge, 2014.

[186] Eleanor O'Rourke, Eric Butler, Armando Díaz Tolentino, and Zoran Popović. Automatic generation of problems and explanations for an intelligent algebra tutor. In *International Conference on Artificial Intelligence in Education*, pages 383–395. Springer, 2019.

[187] Zachary Pardos, Neil Heffernan, Carolina Ruiz, and Joseph Beck. The composition effect: Conjuntive or compensatory? an analysis of multi-skill math questions in its. In *Educational Data Mining 2008*, 2008.

[188] Hae Won Park, Rinat Rosenberg-Kima, Maor Rosenberg, Goren Gordon, and Cynthia Breazeal. Growing growth mindset with a social robot peer. *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 137–145, 2017.

[189] Philip I Pavlik Jr, Hao Cen, and Kenneth R Koedinger. Performance factors analysis–a new alternative to knowledge tracing. *Online Submission*, 2009.

[190] Radek Pelánek. Bayesian knowledge tracing, logistic models, and beyond: an overview of learner modeling techniques. *User Modeling and User-Adapted Interaction*, 27(3):313–350, 2017.

[191] André Pereira, Carlos Martinho, Iolanda Leite, and Ana Paiva. icat, the chess player: the influence of embodiment in the enjoyment of a game. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 3*, pages 1253–1256, 2008.

[192] Chris Piech, Jonathan Spencer, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas Guibas, and Jascha Sohl-Dickstein. Deep knowledge tracing. *arXiv preprint arXiv:1506.05908*, 2015.

[193] Peter Pirolli and Margaret Recker. Learning strategies and transfer in the domain of programming. *Cognition and instruction*, 12(3):235–275, 1994.

[194] Leo Porter, Cynthia Bailey Lee, and Beth Simon. Halving fail rates using peer instruction: a study of four computer science courses. In *Proceeding of the 44th ACM technical symposium on Computer science education*, pages 177–182, 2013.

[195] Aaron Powers, Sara Kiesler, Susan Fussell, and Cristen Torrey. Comparing a computer agent with a humanoid robot. In *Proceedings of the ACM/IEEE international conference on Human-robot interaction*, pages 145–152, 2007.

[196] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, Andrew Y Ng, et al. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, Japan, 2009.

[197] Anna N Rafferty, Emma Brunskill, Thomas L Griffiths, and Patrick Shafto. Faster teaching via pomdp planning. *Cognitive science*, 40(6):1290–1332, 2016.

[198] Aditi Ramachandran. *Building Effective Robot Tutoring Interactions for Children*. PhD thesis, Yale University, 2018.

[199] Aditi Ramachandran, Chien-Ming Huang, Edward Gartland, and Brian Scassellati. Thinking aloud with a tutoring robot to enhance learning. In *Proceedings of the 2018 ACM/IEEE international conference on human-robot interaction*, pages 59–68, 2018.

[200] Aditi Ramachandran, Chien-Ming Huang, and Brian Scassellati. Give me a break! personalized timing strategies to promote learning in robot-child tutoring. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 146–155, 2017.

[201] Aditi Ramachandran, Chien-Ming Huang, and Brian Scassellati. Toward effective robot–child tutoring: Internal motivation, behavioral intervention, and learning outcomes. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 9(1):1–23, 2019.

[202] Aditi Ramachandran, Alexandru Litoiu, and Brian Scassellati. Shaping productive help-seeking behavior during robot-child tutoring interactions. In *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 247–254. IEEE, 2016.

[203] Aditi Ramachandran, Sarah Strohkorb Sebo, and Brian Scassellati. Personalized robot tutoring using the assistive tutor pomdp (at-pomdp). In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8050–8057, 2019.

[204] Rebecca Ramnauth, Emmanuel Adéníran, Timothy Adamson, Michal A Lewkowicz, Rohit Giridharan, Caroline Reiner, and Brian Scassellati. A social robot for improving interruptions tolerance and employability in adults with asd. In *Proceedings of the 2022 ACM/IEEE International Conference on Human-Robot Interaction*, pages 4–13, 2022.

[205] Céline Ray, Francesco Mondada, and Roland Siegwart. What do people expect from robots? In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3816–3821. IEEE, 2008.

[206] Natalia Reich-Stiebert and Friederike Eyssel. (ir) relevance of gender? on the influence of gender stereotypes on learning with a robot. In *2017 12th ACM/IEEE International Conference on Human-Robot Interaction (HRI*, pages 166–176. IEEE, 2017.

[207] Charles Rich, Brett Ponsler, Aaron Holroyd, and Candace L Sidner. Recognizing engagement in human-robot interaction. In *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 375–382. IEEE, 2010.

[208] Laurel D Riek. Wizard of oz studies in hri: a systematic review and new reporting guidelines. *Journal of Human-Robot Interaction*, 1(1):119–136, 2012.

[209] William L Risser. Musculoskeletal injuries: Caused by weight training guidelines for prevention. *Clinical Pediatrics*, 29(6):305–310, 1990.

[210] Michael B Robb, Rebekah A Richert, and Ellen A Wartella. Just a talking book? word learning from watching baby videos. *British Journal of Developmental Psychology*, 27(1):27–45, 2009.

[211] Ben Robins, Kerstin Dautenhahn, R Te Boekhorst, and Aude Billard. Robotic assistants in therapy and education of children with autism: can a small humanoid robot help encourage social interaction skills? *Universal access in the information society*, 4(2):105–120, 2005.

[212] Ben Robins, Kerstin Dautenhahn, and Paul Dickerson. From isolation to communication: a case study evaluation of robot assisted play for children with autism with a minimally expressive humanoid robot. In *2009 second international conferences on advances in computer-human interactions*, pages 205–211. IEEE, 2009.

[213] Cynthia A Rohrbeck, Marika D Ginsburg-Block, John W Fantuzzo, and Traci R Miller. Peer-assisted learning interventions with elementary school students: A meta-analytic review. *Journal of educational Psychology*, 95(2):240, 2003.

[214] Alessandro Roncone, Olivier Mangin, and Brian Scassellati. Transparent role assignment and task allocation in human robot collaboration. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1014–1021. IEEE, 2017.

[215] Matthew B Ruby, Elizabeth W Dunn, Andrea Perrino, Randall Gillis, and Sasha Viel. The invisible benefits of exercise. *Health Psychology*, 2011.

[216] Gregory N Ruegsegger and Frank W Booth. Health benefits of exercise. *Cold Spring Harbor perspectives in medicine*, 2018.

[217] Alessandra Maria Sabelli, Takayuki Kanda, and Norihiro Hagita. A conversational robot in an elderly care center: an ethnographic study. In *2011 6th*

ACM/IEEE international conference on human-robot interaction (HRI), pages 37–44. IEEE, 2011.

[218] Martin Saerbeck, Tom Schut, Christoph Bartneck, and Maddy D Janse. Expressive robots in education: varying the degree of social supportive behavior of a robotic tutor. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1613–1622, 2010.

[219] Nicole Salomons, Emir Akdere, and Brian Scassellati. Bkt-pomdp: Fast action selection for user skill modelling over tasks with multiple skills. *International Joint Conference on Artificial Intelligence*, 2021.

[220] Nicole Salomons, Kaitlynn Taylor Pineda, Adérónké Adéjàre, and Brian Scassellati. " we make a great team!": Adults with low prior domain knowledge learn more from a peer robot than a tutor robot. In *Proceedings of the 2022 ACM/IEEE International Conference on Human-Robot Interaction*, pages 176–184, 2022.

[221] Nicole Salomons and Brian Scassellati. Continuous bayesian knowledge tracing - modelling user skills over time. *[Under Review]*, 2022.

[222] Nicole Salomons, Sarah Strohkorb Sebo, Meiying Qin, and Brian Scassellati. A minority of one against a majority of robots: Robots cause normative and informational conformity. *ACM Transactions on Human-Robot Interaction (THRI)*, 10(2):1–22, 2021.

[223] Nicole Salomons, Michael van der Linden, Sarah Strohkorb Sebo, and Brian Scassellati. Humans conform to robots: Disambiguating trust, truth, and conformity. In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, pages 187–195. ACM, 2018.

[224] Nicole Salomons, Tom Wallenstein, Debasmita Ghose, and Brian Scassellati. People perform less exercising mistakes with an embodied robot than a disembodied robot. In *International Symposium on Robot and Human Interactive Communication (RO-MAN)*, 2022.

[225] Tamie Salter, Kerstin Dautenhahn, and R Bockhorst. Robots moving out of the laboratory-detecting interaction levels and human contact in noisy school environments. In *RO-MAN 2004. 13th IEEE International Workshop on Robot and Human Interactive Communication (IEEE Catalog No. 04TH8759)*, pages 563–568. IEEE, 2004.

[226] Shane P Saunderson and Goldie Nejat. Persuasive robots should avoid authority: The effects of formal and real authority on persuasion in human-robot interaction. *Science Robotics*, 6(58):eabd5186, 2021.

[227] Brian Scassellati. How social robots will help us to diagnose, treat, and understand autism. In *Robotics research*, pages 552–563. Springer, 2007.

[228] Brian Scassellati, Henny Admoni, and Maja Matarić. Robots for use in autism research. *Annual review of biomedical engineering*, 14:275–294, 2012.

[229] Brian Scassellati, Laura Boccanfuso, Chien-Ming Huang, Marilena Mademtzi, Meiying Qin, Nicole Salomons, Pamela Ventola, and Frederick Shic. Improving social skills in children with asd using a long-term, in-home social robot. *Science Robotics*, 3(21):eaat7544, 2018.

[230] Bob R Schadenberg, Mark A Neerincx, Fokie Cnossen, and Rosemarijn Looije. Personalising game difficulty to keep children motivated to play with a social robot: A bayesian approach. *Cognitive systems research*, 43:222–231, 2017.

[231] Sebastian Schneider and Franz Kummert. Comparing the effects of social robots and virtual agents on exercising motivation. In *International Conference on Social Robotics*, pages 451–461. Springer, 2018.

[232] Thorsten Schodde, Kirsten Bergmann, and Stefan Kopp. Adaptive robot language tutoring based on bayesian knowledge tracing and predictive decision-making. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 128–136, 2017.

[233] Chris Segrin. The effects of nonverbal behavior on outcomes of compliance gaining attempts. *Communication Studies*, 44(3-4):169–187, 1993.

[234] Masahiro Shiomi and Norihiro Hagita. Do synchronized multiple robots exert peer pressure? In *Proceedings of the Fourth International Conference on Human Agent Interaction*, pages 27–33. ACM, 2016.

[235] Elaine Short, Katelyn Swift-Spong, Jillian Greczek, Aditi Ramachandran, Alexandru Litoiu, Elena Corina Grigore, David Feil-Seifer, Samuel Shuster, Jin Joo Lee, Shaobo Huang, et al. How to train your dragonbot: Socially assistive robots for teaching children about nutrition through play. In *The 23rd IEEE international symposium on robot and human interactive communication*, pages 924–929. IEEE, 2014.

[236] K Dmello Sidney, Scotty D Craig, Barry Gholson, Stan Franklin, Rosalind Picard, and Arthur C Graesser. Integrating affect sensors in an intelligent tutoring system. In *Affective Interactions: The Computer in the Affective Loop Workshop at*, pages 7–13, 2005.

[237] Mikey Siegel, Cynthia Breazeal, and Michael I Norton. Persuasive robotics: The influence of robot gender on human behavior. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2563–2568. IEEE, 2009.

[238] Beth Simon, Sarah Esper, Leo Porter, and Quintin Cutts. Student experience in a student-centered peer instruction classroom. In *Proceedings of the ninth annual international ACM conference on International computing education research*, pages 129–136, 2013.

[239] Robert E Slavin. Cooperative learning. *Review of educational research*, 50(2):315–342, 1980.

[240] Daniel Smilkov, Nikhil Thorat, Yannick Assogba, Ann Yuan, Nick Kreeger, Ping Yu, Kangyi Zhang, Shanqing Cai, Eric Nielsen, David Soergel, Stan Bileschi, Michael Terry, Charles Nicholson, Sandeep N. Gupta, Sarah Sirajuddin, D. Sculley, Rajat Monga, Greg Corrado, Fernanda B. Viégas, and Martin Wattenberg. Tensorflow.js: Machine learning for the web and beyond. *CoRR*, abs/1901.05350, 2019.

[241] Tristram Smith, Lawrence Scahill, Geraldine Dawson, Donald Guthrie, Catherine Lord, Samuel Odom, Sally Rogers, and Ann Wagner. Designing research studies on psychosocial interventions in autism. *Journal of autism and developmental disorders*, 37(2):354–366, 2007.

[242] Bailin Song and Eva Richter. Tutoring in the classroom: A quantitative study. *The Writing Center Journal*, 18(1):50–60, 1997.

[243] Daniel Svozil, Vladimir Kvasnicka, and Jiri Pospichal. Introduction to multi-layer feed-forward neural networks. *Chemometrics and intelligent laboratory systems*, 39(1):43–62, 1997.

[244] Katelyn Swift-Spong, Elaine Short, Eric Wade, and Maja J Matarić. Effects of comparative feedback from a socially assistive robot on self-efficacy in post-stroke rehabilitation. In *2015 IEEE International Conference on Rehabilitation Robotics (ICORR)*, pages 764–769. IEEE, 2015.

[245] Daniel Szafir and Bilge Mutlu. Pay attention! designing adaptive agents that monitor and improve user engagement. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 11–20, 2012.

[246] Fumihide Tanaka, Aaron Cicourel, and Javier R Movellan. Socialization between toddlers and robots at an early childhood education center. *Proceedings of the National Academy of Sciences*, 104(46):17954–17958, 2007.

[247] Fumihide Tanaka and Shizuko Matsuzoe. Children teach a care-receiving robot to promote their learning: Field experiments in a classroom for vocabulary learning. *Journal of Human-Robot Interaction*, 1(1):78–95, 2012.

[248] Adriana Tapus, Maja J Mataric, and Brian Scassellati. Socially assistive robotics [grand challenges of robotics]. *IEEE robotics & automation magazine*, 14(1):35–42, 2007.

[249] Romain Tavenard, Johann Faouzi, Gilles Vandewiele, Felix Divo, Guillaume Androz, Chester Holtz, Marie Payne, Roman Yurchak, Marc Rußwurm, Kushal Kolar, and Eli Woods. Tslearn, a machine learning toolkit for time series data. *Journal of Machine Learning Research*, 21(118):1–6, 2020.

[250] Noriyuki Tejima. Rehabilitation robotics: a review. *Advanced Robotics*, 14(7):551–564, 2001.

[251] Ulf Toelch, Matthew J Bruce, Lesley Newson, Peter J Richerson, and Simon M Reader. Individual consistency and flexibility in human social information use. *Proceedings of the Royal Society B: Biological Sciences*, 281(1776):20132864, 2014.

[252] Ulf Toelch and Raymond J Dolan. Informational and normative influences in conformity from a neurocomputational perspective. *Trends in cognitive sciences*, 19(10):579–589, 2015.

[253] Luca van Breda, Niels van Gils, Tim Muyrers, and Yinyuan Peng. Correcting incorrect form and posture within weightlifting exercises through electrical muscle stimulation (ems). 2021.

[254] Kurt VanLehn. The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational Psychologist*, 46(4):197–221, 2011.

[255] Anna-Lisa Vollmer, Robin Read, Dries Trippas, and Tony Belpaeme. Children conform, adults resist: A robot group induced peer pressure on normative social conformity. *Sci Robotics*, 2018.

[256] Lev Semenovich Vygotsky and Michael Cole. *Mind in society: Development of higher psychological processes*. Harvard university press, 1978.

[257] Kazuyoshi Wada and Takanori Shibata. Robot therapy in a care house-its sociopsychological and physiological effects on the residents. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 3966–3971. IEEE, 2006.

[258] Joshua Wainer, David J Feil-Seifer, Dylan A Shell, and Maja J Mataric. Embodiment and human-robot interaction: A task-based perspective. In *RO-MAN 2007-The 16th IEEE International Symposium on Robot and Human Interactive Communication*, pages 872–877. IEEE, 2007.

[259] Joshua Wainer, Ben Robins, Farshid Amirabdollahian, and Kerstin Dautenhahn. Using the humanoid robot kaspar to autonomously play triadic games and facilitate collaborative play among children with autism. *IEEE Transactions on Autonomous Mental Development*, 6(3):183–199, 2014.

[260] Janet P Wallace. Exercise in hypertension. *Sports Medicine*, 2003.

[261] Jacob Whitehill, Marian Bartlett, and Javier Movellan. Automatic facial expression recognition for intelligent tutoring systems. In *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–6. IEEE, 2008.

[262] Randi Williams, Christian Vázquez Machado, Stefania Druga, Cynthia Breazeal, and Pattie Maes. My doll says it's ok: a study of children's conformity to a talking doll. In *Proceedings of the 17th ACM Conference on Interaction Design and Children*, pages 625–631. ACM, 2018.

[263] Paul Williamson, Nathan Weber, and Marie-Therese Robertson. The effect of expertise on memory conformity: a test of informational influence. *Behavioral sciences & the law*, 31(5):607–623, 2013.

[264] Ricarda Wullenkord and Friederike Eyssel. The influence of robot number on robot group perception—a call for action. *ACM Transactions on Human-Robot Interaction (THRI)*, 9(4):1–14, 2020.

[265] Yanbo Xu and Jack Mostow. Using logistic regression to trace multiple subskills in a dynamic bayes net. In *EDM*, pages 241–246. Citeseer, 2011.

[266] Ammar Yasser, Doha Tariq, Radwa Samy, M Allah, and Ayman Atia. Smart coaching: Enhancing weightlifting and preventing injuries. *International Journal of Advanced Computer Science and Applications*, 10:01, 2019.

[267] Michael V Yudelson, Kenneth R Koedinger, and Geoffrey J Gordon. Individualized bayesian knowledge tracing models. In *International conference on artificial intelligence in education*, pages 171–180. Springer, 2013.

[268] Cristina Zaga, Manja Lohse, Khiet P. Truong, and Vanessa Evers. The effect of a robot's social character on children's task engagement: Peer versus tutor. In Adriana Tapus, Elisabeth André, Jean-Claude Martin, François Ferland, and Mehdi Ammi, editors, *Social Robotics*, Lecture Notes in Computer Science, pages 704–713. Springer International Publishing, 2015.

[269] Zhi Zheng, Esubalew Bekele, Amy Swanson, Amy Weitlauf, Zachary Warren, and Nilanjan Sarkar. The impact of robots on children with autism spectrum disorder. In *Autism Imaging and Devices*, pages 413–434. CRC Press, 2017.