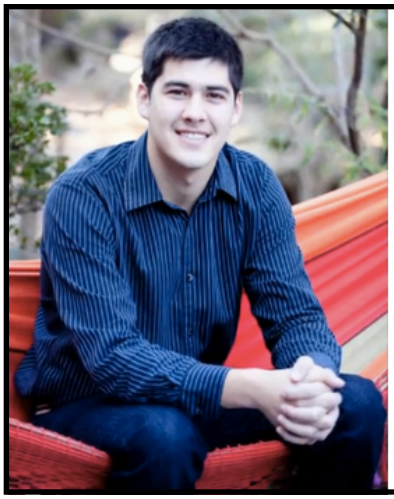


Sam Spaulding

Applying A.I. to Robotics, inspired by Cognitive Science

Resume/CV

Papers/Projects:

-The Physical Presence of A Robot Increases Cognitive Learning Gains

Initial results from a Robotic Tutor project demonstrating an 'embodiment effect': Students who received lessons from a physically present robot tutor showed increased learning gains over those who saw identical content from a video of a robot or a disembodied voice.

Presented at 34th Annual Meeting of the Cognitive Science Society (Cog Sci '12)
Attached Poster

-Assessing Cognitive Skill Proficiency with Robotic Tutors

Follow-up study to the above. We developed a simple Bayesian inference algorithm to update a model of students' understanding of the task and infer proficiency in skills. Students were then either given lessons that the model predicted would be most helpful or at random. Analysis showed that the algorithm both helped students solve puzzles faster and increased the rate at which they improved their solving times.

In Preparation for the 35th Annual Meeting of the Cognitive Science Society (Cog Sci '13)

-The Effect of Robot Agency on Trust and Decision-Making

Final project for CS 473: Intelligent Robotics. We designed a 2x2 robot interaction game to investigate how participants would react to a social robot that offered advice on a strategy card game.
Video footage of the interaction can be seen at:
http://www.youtube.com/user/SamuelLSpaulding/videos?view=0

-Making Sense of the Blogosphere: Semantic Analysis of Text Mined from the Web

Due to a NDA with the Walt Disney Company, I cannot provide a demo or any detailed media from the project until publication.

In preparation for the 2013 Disney Analytics and Optimization Summit

Back to Top

Samuel Spaulding

www.samspaulding.com
samuel.spaulding@yale.edu

Education

B.S., Computer Science — GPA 3.64/4.00 *expected May 2013*
Yale University, New Haven, CT

Relevant Coursework: Graduate courses in Artificial Intelligence and Computer Vision, Intelligent Robotics, Natural Language Processing, Algorithm Design and Analysis, Systems Programming, Vector Calculus, Linear Algebra, Statistics and Probability

Work Experience

Walt Disney Imagineering - Research, *Research Associate* *Summer 2012*
Advisor: Senior Research Scientist Jonathan Yedidia

- Worked on a Machine Learning research project. As a member of an NLP and Statistical Machine Learning research team, I was integrally involved in design, implementation, testing, and production of a prototype sentiment analysis system for Walt Disney Imagineering.
- Our project, "Making Sense of the Blogosphere: Semantic Analysis of Text Mined from the Web" won the **"Judges' Special Distinction - Methodology"** award in the company-wide 2012 Business Intelligence and Data Analytics Competition.

Amazon.com, Inc., *Software Development Engineer Intern* *Summer 2011*

- Designed and developed a website that produced dynamic client-side graphs based on internal team metrics. Responsible for the project from design, through implementation, and into production.
- Member of a four-person team whose submission, an Android app called "SmileIKnow" was a finalist at the 2011 Amazon Mobile Security Hackathon.

Yale Social Robotics Laboratory, *Undergraduate Research Assistant*, *Summer 2010 - present*
Advisor: Brian Scassellati

- Responsibilities include designing, implementing, and testing AI/Robotics systems and conducting Human-Robot Interaction research. See below for published results.

Publications

Leyzberg, D., Spaulding, S., Toneva, M. and Scassellati, B."The Physical Presence of a Robot Tutor Increases Cognitive Learning Gains," accepted for publication and presented at the 34th Annual Meeting of the Cognitive Science Society (Cog Sci '12) July 2012.

Leyzberg D., Spaulding S., and Scassellati, B. "Assessing Cognitive Skill Proficiency with Robotic Tutors," in preparation for the 35th Annual Meeting of the Cognitive Science Society (Cog Sci '13) July 2013.

"Making Sense of the Blogosphere: Semantic Analysis of Text Mined from the Web", in preparation for the Disney Analytics and Optimization Summit.

Presentations and Academic Talks

"A New Model of Cognitive Skill Assessment" Invited talk at the 2011 *Yale Engineering and Science Weekend Symposium.*

"A New Model of Cognitive Skill Assessment" Invited talk at the 2011 *Yale Undergraduate Science Symposium.*

"Social Robotics and Intelligent Cognitive Tutoring", Invited talk at the 2012 *Yale Engineering and Science Weekend Symposium.*

"Social Robotics and Intelligent Cognitive Tutoring", Invited talk at the 2012 *Yale Undergraduate Science Symposium*

Back to Top

Academic Awards

Sigma Xi Undergraduate Research Award Applied for and received funding and recognition in support of my research. Nominated to join the Sigma Xi Scientific Research Society.

Technical Skills

Programming Languages: Extensive experience with Object-Oriented (Java/C/C++), Functional (LISP), Scripting (Ruby/Python) and Assembly Languages as well as MATLAB and Mathematica. Strong Web Development and Design skills including HTML/CSS/Javascript and Ruby on Rails. Experienced with Android mobile development.

Robots: Programmed multiple robot platforms including the iRobot Create, Aldebaran Nao, and Beatbots Keepon. Experienced with Robot Operating System (ROS) and OpenCV.

Hardware: Some machine shop experience. Familiar with microcontroller programming and low-level circuit design for mobile robot control and sensing.

Teaching Experience

Undergraduate Peer Tutor, CS 201: Introduction to Computer Science *Fall 2012*
Tutored introductory Computer Science students 4 hours per week. Covered basic concepts like recursion through more advanced topics like formal language theory, logic, and computability theory.

Activities and Other Interests

Jeopardy! College Championship, First Runner-up, Won 3 of 4 games and second place at the Season 27 Jeopardy! College Championship

Treasurer, Captain of Yale Student Academic Competitions, Treasurer, Captain of four person travel team competing in national and regional academic competitions. Winner of numerous individual and team performance awards, including 1st place finish at the 2011 ACF National Championship

Other interests: Mathematical Logic, Starcraft II, Cellular Automata, Guitar Hero, Board Games, Rock Climbing, and 3D-printing.

Back to Top

The Physical Presence of a Robot Tutor Increases Cognitive Learning Gains

Daniel Leyzberg (daniel.leyzberg@yale.edu)
Samuel Spaulding (samuel.spaulding@yale.edu)
Mariya Toneva (mariya.toneva@yale.edu)
Brian Scassellati (scaz@cs.yale.edu)
Department of Computer Science, Yale University
51 Prospect St., New Haven, CT 06511, USA

Abstract

We present the results of a 100 participant study on the role of a robot's physical presence in a robot tutoring task. Participants were asked to solve a set of puzzles while being provided occasional gameplay advice by a robot tutor. Each participant was assigned one of five conditions: (1) *no advice*, (2) *robot providing randomized advice*, (3) *voice of the robot providing personalized advice*, (4) *video representation of the robot providing personalized advice*, or (5) *physically-present robot providing personalized advice*. We assess the tutor's effectiveness by the time it takes participants to complete the puzzles. Participants in the *robot providing personalized advice* group solved most puzzles faster on average and improved their same-puzzle solving time significantly more than participants in any other group. Our study is the first to assess the effect of the physical presence of a robot in an automated tutoring interaction. We conclude that physical embodiment can produce measurable learning gains.

Keywords: Robotics; Computer Science; Tutoring

Introduction

What kinds of human-robot interactions benefit from the physical embodiment of a robot? For human-robot interactions that require manipulating the physical world, a physical robot is a necessity, but for those interactions where physical embodiment is optional, when is an embodied robot more useful than an on-screen agent?

In this study, we explore the differences in task performance of participants engaged in a cognitive learning task in which a robot acts as a tutor. Participants were asked to play a puzzle game while receiving strategy advice from either: a physically-present robot, a video of the same robot, its disembodied voice, a robot giving randomized advice, or no agent at all. We use the resulting data to draw conclusions about the effect of embodiment in robot tutoring tasks.

Previous work has investigated the social influence of a robot's embodiment. Does a robot engender more trust, more compliance, more engagement, or more motivation by its physical presence, more so than an on-screen agent or a video representation of a robot would? Such questions have been explored via two methodologies: self-report measures and task-performance measures. Using self-report measures, Kidd and Breazeal (2004) found that a physically-present robot was perceived as more enjoyable, more credible, and more informative than an on-screen character in a block-moving task. In Wainer, Feil-Seifer, Shell, and Mataric (2007), an embodied robot was rated as more attentive and more helpful than both a video representation of the robot and a simulated on-screen robot-like character. Tapus, Tapus, and Mataric (2009) found that individuals suffering from cognitive impairment and/or Alzheimer's disease reported being more engaged with a robot treatment than a similar on-screen agent treatment.

Kiesler, Powers, Fussell, and Torrey (2008) used task-performance measures to find that participants who received health advice from a physically-present robot were more likely to choose a healthy snack than participants who received the same information in robot-video or on-screen agent conditions. In Bainbridge, Hart, Kim, and Scassellati (2008), a physically-present robot yielded significantly more compliance to its commands than a video representation of the same robot.

No previous work has investigated whether learning outcomes are affected by a robot's physical presence. The closest related work is in Intelligent Tutoring Systems (ITSś), which are educational computer programs that produce individualized lessons, advice, and questions usually in a workbook-style or quiz-style environment (Nkambou, Bourdeau, & Psyché, 2010). A parallel notion of embodiment called "the persona effect" exists in ITS research. (See Dehn and Van Mulken (2000) for an overview.) The persona effect is the impact, if any, that an on-screen character has on students using an ITS. The majority of research on the persona effect has shown no significant learning gains produced by on-screen agents, although many studies note that students find an ITS with an on-screen more engaging than one without (Moundridou & Virvou, 2002).

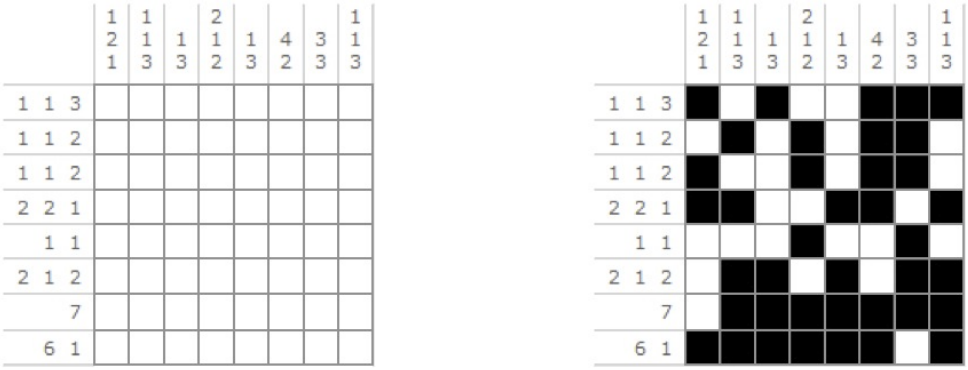
Our study is the first to assess the effect of the physical presence of a robot in an automated tutoring interaction. We use the task-performance measure of puzzle solving time in this work as well as several self-report measures.

Methodology

Participants

There were 100 participants in this study, between 18 and 40 years of age. The study was conducted in New Haven, Connecticut. Most participants were undergraduate and graduate students of Yale University. Each participant was assigned to one of five groups: (1) *no lessons*, (2) *randomized lessons from a physically-present robot*, (3) *personalized lessons from a disembodied voice*, (4) *personalized lessons from a video representation of the robot*, and (5) *personalized lessons from*

Back to Top



(a) Sample nonogram puzzle, blank.

(b) Sample nonogram puzzle, solved.

Figure 1: A sample nonogram puzzle. The objective of nonograms is, starting with a blank board (see left figure), to find a pattern of shaded boxes on the board such that the number of consecutively shaded boxes in each row and column appear as specified, in length and order, by the numbers that are printed to the left of each row and above each column (see right figure). For a more detailed explanation see the **Domain** section.

a physically-present robot. There were approximately 20 participants in each group. Exclusion criteria for participants were lack of English fluency or prior academic experience with robotics or artificial intelligence.

Apparatus

In this experiment, participants were asked to solve a series of logic puzzles. In the four experimental conditions with a tutor, the tutor interrupted participants several times per puzzle to deliver puzzle-solving strategy lessons. The lessons themselves were pre-recorded audio and synchronized visual aids, between 21 and 47 seconds in length, that explained and gave examples of the use of a single puzzle-solving strategy. In the experimental conditions with *personalized lessons*, the order of the lessons was determined by a skill assessment algorithm that identified skills in which participants were weak; see the **Skills & Lessons** section. In the *randomized lessons* condition, the tutor chose a random lesson among the same ones used in the *personalized lessons* conditions, such that it was immediately applicable to the current state of the gameboard. We compare the puzzle solving time performance between participants in these groups to evaluate the effect of the robot's physical presence on the effectiveness of the tutoring.

Domain To minimize the influence of prior experience, we chose a test domain to which participants likely had little previous exposure: a grid-based fill-in-the-blanks puzzle game called "nonograms" (or "nonogram puzzles") that resemble crossword puzzles or Sudoku. Nonogram puzzles are a difficult cognitive task, one that requires several layers of logical inferences to complete. Solving a nonogram puzzle of arbitrary size is an NP-complete problem (Nagao, Ueda, Ueda, Sato, & Watanabe, 1996), meaning that no efficient computational solution is known.

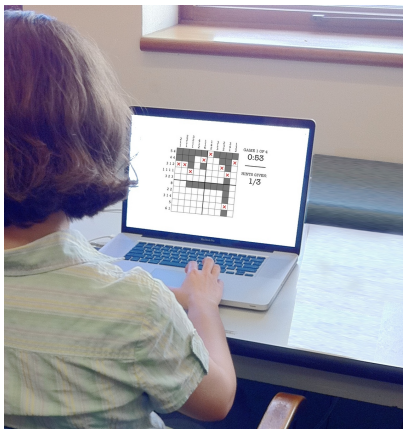
The objective of nonograms is, starting with a blank board, to shade in boxes on the board such that the number of consecutively shaded boxes in each row and column appear as specified, in length and order, by the numbers that are printed to the left of each row and above each column. (See Figures 1(a) and 1(b) for a sample puzzle and solution.) For instance, a row marked as "4 2" must have 4 adjacent shaded boxes, followed by 2 adjacent shaded boxes—in that order, with no other boxes shaded, and with at least one empty box between the sets of adjacent shaded boxes. We refer to these contiguous sets of shaded boxes as "stretches" in this paper. For instance, the row described above requires two stretches, one of length 4, the other of length 2. One solves the puzzle when one finds a pattern of blank and shaded boxes such that all of the requirements for each row and column are satisfied.

In a typical puzzle, one cannot solve many rows or columns independently. One must infer the contents of parts of rows or columns and use previous inferences as the basis of subsequent inferences. To that end, when a player has reasoned that in some box or boxes there should not be shading, they can mark such boxes with an 'X' for reference.

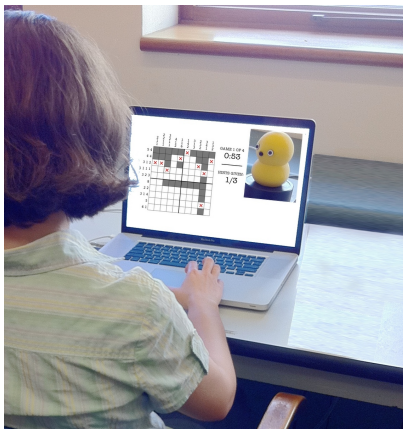
We created a full-screen nonograms computer program that participants used via mouse and keyboard. The user interface provided a timer and a count of how many lessons (called "hints" in the interface) the participant had received and how many they would receive; see Figure 2.

Participants were asked to play four puzzles on ten-by-ten grids with a time limit of fifteen minutes per puzzle. The puzzles themselves were the same across all participants. The fourth puzzle used the same board as the first, although disguised in the fourth puzzle by rotating the board 90° (such that the column stretch requirements were swapped with row stretch requirements). This means that the first puzzle and the last puzzle are of the exact same difficulty and require knowledge of the exact same set of skills to solve. This manipulation enables us to make within-subjects comparisons about the extent to which each participant improved their skills over the course of their participation in the study. There was no indication that any participant was aware of this manipulation.

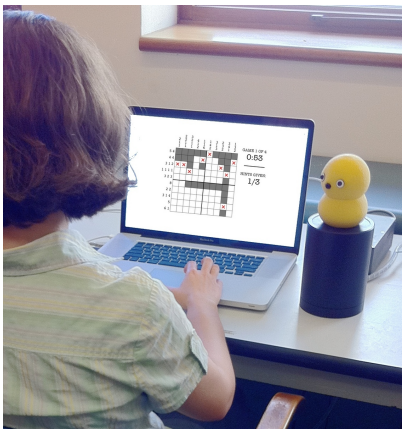
Back to Top



(a) Experiment apparatus in the *no lessons* condition and the *personalized lessons from a disembodied voice* condition.



(b) Experiment apparatus in the *personalized lessons from a video representation of the robot* condition.



(c) Experiment apparatus in the *randomized lessons from a physically-present robot* condition and the *personalized lessons from an embodied robot* condition.

Figure 2: Experiment apparatus by condition.

Skills & Lessons In the four conditions with lessons, the tutor interrupted the participant three times per puzzle, paused the puzzle, and delivered a short lesson about nonograms. The lessons ranged from 21 seconds to 47 seconds in length and consisted of a voice recording and a set of animations presented on screen during the lesson as well as a set of coordinated robot motions specific to each lesson.

When beginning a lesson the tutor would turn to face the participant (in the *video* and *physically-present robot* conditions) and say "I have an idea that might help you," or "Here's another hint for you." During the lesson, the tutor bounced subtly and looked back at the screen whenever, in the course of the lesson, it would make reference to the example presented on screen. For instance, when in the audio of the lesson the robot would say "Like in this example..." or "As you see here...," the robot would turn briefly to the screen and then back to the participant.

Ten nonogram puzzle-solving skills were identified based on the subjective experience of the authors; they are not universally identified skills or rules for nonograms. Each skill is a set of row or column states in which one can logically fill in some of the remaining empty boxes. For example, a stretch of length 9 can fit in a blank row or column of 10 boxes in only two ways. Either it fills the first box and 8 more, or it fills those same middle 8 boxes and the last box. In either case, the middle 8 boxes are shaded. One of the ten skills in this experiment is that, for an empty row or column with just one stretch requirement of n where $n > 5$, the middle $(2n - 10)$ boxes are shaded. See Figure 3 for examples and explanations of this skill and two others.

There was one recorded lesson for each skill. Three lessons were delivered per puzzle, for each of four puzzles. The number of lessons was constant for all participants regardless of how long they needed to finish the puzzle. Lessons were triggered either when a participant made no moves for 45 seconds or as he or she filled the 25^{th}, 50^{th} or 75^{th} box on the board (of 100). The user interface displayed the number of lessons remaining for each puzzle at all times.

In the *personalized lesson conditions* the lessons were chosen based on a skill assessment algorithm. For each skill, a weighted sum was calculated internally consisting of: (1) the number of recent demonstrations of that skill (weighted positively) and (2) the number of recent gameboard states in which a skill could have been applied but no action was taken (weighted negatively). These assessments were updated for each skill separately throughout the game, and the skill with the lowest assessment that was applicable to the current gameboard was the skill for which a lesson was selected. In this way, participants in the *personalized lesson* conditions received lessons based on their individual performance on the puzzles.

Alternatively, in the *randomized lesson condition*, lessons were chosen among the same ten lessons at random each time, such that the lesson chosen could be applied to the current state of the gameboard. This ensures that although the lessons were randomized, they would provide actionable information every time.

Robot The robot we used, Keepon, is a small yellow snowman-shaped robot; see Figure 2(c). Keepon has previously been used as an emotive non-threatening communication tool (Kozima, Nakagawa, & Yasuda, 2005; Leyzberg, Avrunin, Liu, & Scassellati, 2011).

The robot operated in one of three modes. First, it refereed the puzzle game: it welcomed participants when they started, told them when they had finished or when they had run out of time, and told them when the experiment was over. Second, it "observed" the board during gameplay: the robot frequently turned its head to face the location of the mouse cursor. Third, it delivered short gameplay lessons three times per puzzle: it "spoke" to the participant by turning to face him or her and "bouncing" its body subtly while playing one of several prerecorded spoken messages. If a lesson needed to be repeated,

the robot would first apologize for repeating itself (i.e., "I'm sorry to repeat this hint but I think it might help.").

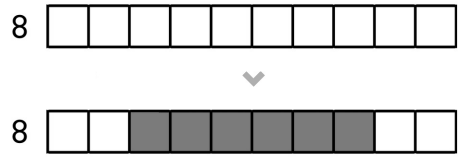
To simplify the potential perception problems inherent in real-world measurements, the robot in this study received perfect knowledge of the state of the game. We did not use a robotic vision system to detect state changes.

Procedure

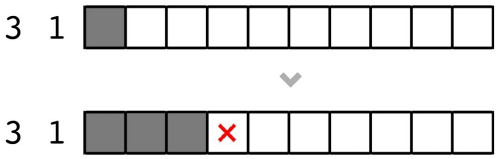
Participants were first asked to watch a five-minute instructional video and read a two-page instruction manual describing the rules of nonograms and how to use the computer interface. In the video and in the text, participants were encouraged to use logical reasoning to make moves in the puzzle rather than making moves by guessing. Potential questions about the rules of the puzzle game were answered by the experimenter after the instructions.

During the experiment, participants were alone in a room with the computer, the robot in conditions including the robot, and a video camera positioned behind them; see Figure 2. Participants would choose when they were ready to start each new puzzle; each round would end either when the participant solved the puzzle or when fifteen minutes had elapsed, whichever happened first.

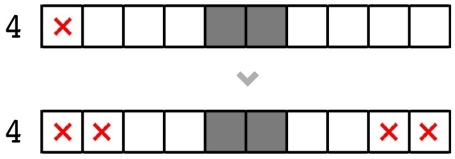
After the conclusion of the final puzzle, participants were asked to complete a survey consisting of five Likert-scale questions with open-ended follow-up questions for each. The questions were designed to assess whether the lessons were helpful, clear, and influential, as well as the user's perceptions of the robot. We asked participants to rate: how relevant the lessons were, how much the lessons influenced their gameplay, how well participants understood the lessons, and how "smart/intelligent" and "distracting/annoying" they perceived the robot to be.



(a) In this row, there must be one long stretch. By the process of elimination one can infer that this stretch must occupy at least the middle six boxes, no matter where in the row it is placed.



(b) In this row, the first box is already shaded. Given that, and that the first stretch must be 3 boxes long, one can infer that the first three boxes must be shaded and the fourth must be crossed out.



(c) In this row, there is only one short stretch and some boxes are already shaded. One can infer that regardless of where that one stretch is placed, it cannot occupy the first two or the last two boxes in that row.

Figure 3: Examples of nonograms skills. Displayed are the contents of a row before and after each skill is applied. Although only rows are shown here, all nonograms skills apply to columns as well.

Results

This study investigates the role of physical embodiment in a robot tutoring system. The behavioral measure is the time in which participants were able to solve each of the four puzzles. For the purposes of calculating a mean, puzzles in which participants ran out of time were evaluated as having solved the puzzle when time ran out, fifteen minutes from the start of each puzzle. This occurred in 12.4% of all puzzles.

Table 1: Mean Solving Time

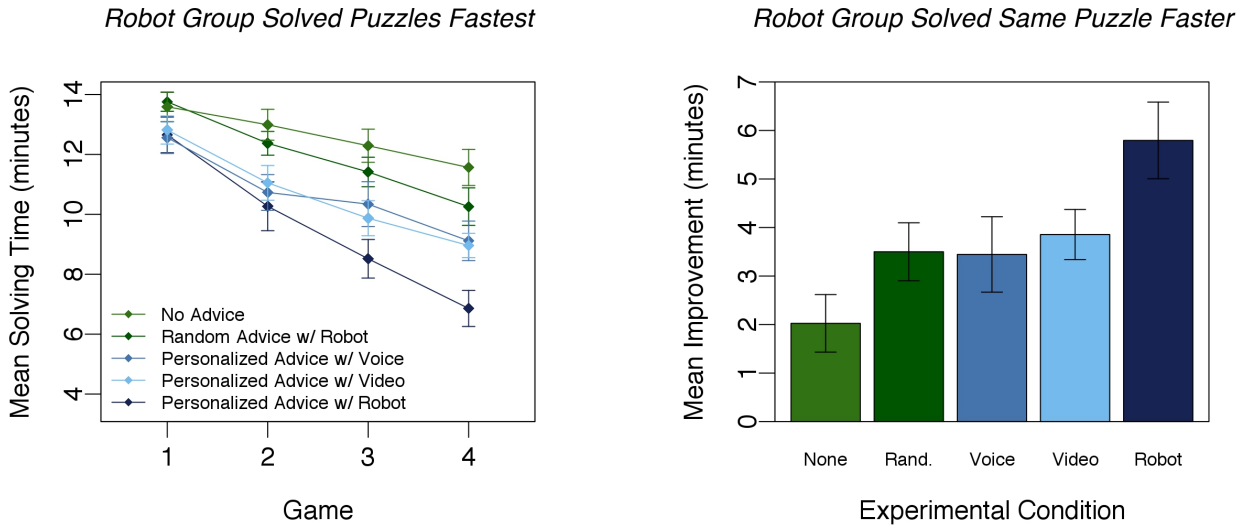
	Puzzle 1	Puzzle 2	Puzzle 3	Puzzle 4
None	13.6 ± 2.2	13.0 ± 2.3	12.3 ± 2.5	11.6 ± 2.7
Rand.	13.8 ± 1.4	12.5 ± 2.0	11.4 ± 2.3	10.3 ± 2.9
Voice	12.6 ± 2.4	10.7 ± 2.7	10.3 ± 3.3	9.1 ± 3.0
Video	12.8 ± 2.1	11.1 ± 2.6	9.9 ± 2.6	8.7 ± 2.4
Robot	12.7 ± 2.6	10.0 ± 3.5	9.4 ± 3.0	7.6 ± 3.1

Participants in the *robot group* performed better, on average, on the second, third, and four puzzles than participants in any other group. See Table 1 for means and standard deviations and see Figure 4(a). In the forth puzzle, the mean solving time in the *robot group* ($M = 7.6$ minutes, $SD = 3.1$) is significantly better than in the *video group* ($M = 8.7$ minutes, $SD = 2.4$), $t(36) = 0.03$, and in the *voice group* ($M = 9.1$ minutes, $SD = 3.0$) as well, $t(36) = 0.02$. These data indicate that the robot's physical presence made a significant learning impact on participants greater than that of an disembodied voice and a video representation of a robot.

In this experiment, the first and fourth puzzles were 90° rotated variations of the same board. Thus they required exactly the same skills to solve and the difference in their solving time is a measure of the participants' acquired knowledge over the course of the study. Participants in the *robot condition* improved ($M = 5.8$ minutes, $SD = 3.5$) their same-puzzle solving time significantly more than those in both the *video condition* ($M = 3.9$ minutes, $SD = 2.3$), $t(36) = 0.048$ and *voice condition*, ($M = 3.4$ minutes, $SD = 3.5$), $t(36) = 0.04$; see Figure 4(b). This data indicates that participants who received lessons from the robot learned more effectively than those who received only voice- or video-based lessons.

Survey results verify the following manipulation: participants in the three *personalized advice conditions* rated the lessons significantly more relevant ($M = 6.0, SD = 1.4$)

Back to Top



(a) Mean solving time per puzzle. Participants in the *robot condition* solved each puzzle faster than participants in any other condition. In the fourth puzzle, significantly faster ($p \leq 0.03$). See Table 1 for means and standard deviations.

(b) Mean improvement in solving time between puzzles #1 and #4. These two puzzles were variations of the same gameboard, disguised in the fourth puzzle by a 90° rotation. Participants in the *robot condition* improved their solving time significantly more than those in any other condition ($p < 0.05$).

Figure 4: Behavioral measure results: (a) participants who received personalized lessons from an embodied robot solved every puzzle puzzle faster on average, the fourth significantly so ($p \leq 0.03$) than participants in all other conditions; see Table 1. (b) *robot condition* participants also improved on their same puzzle solving time significantly higher more than participants in all other conditions ($p < 0.05$).

than participants in the *randomized advice condition* ($M = 3.9, SD = 1.1$), $t(33) < 0.001$. There was no significant difference in how highly participants rated their understanding of the lessons between groups: ($M = 6.0, SD = 1.4$) in the *random condition*,($M = 6.6, SD = 1.2$) in the *voice condition*, ($M = 6.6, SD = 1.5$) in the *video condition*, and ($6.4, SD = 1.2$) *robot condition*; see Figure 5. These data indicate that whatever social effect physical embodiment has on this interaction, it does not influence the participants' perception of their understanding of the lessons, despite the fact that the behavioral measure indicates better learning in the *robot condition*.

Discussion

Our results indicate that a physically-present robot tutor produces better learning gains than on-screen or voice-only tutors. Further work is needed to identify the underlying social factors and mechanisms that cause this effect.

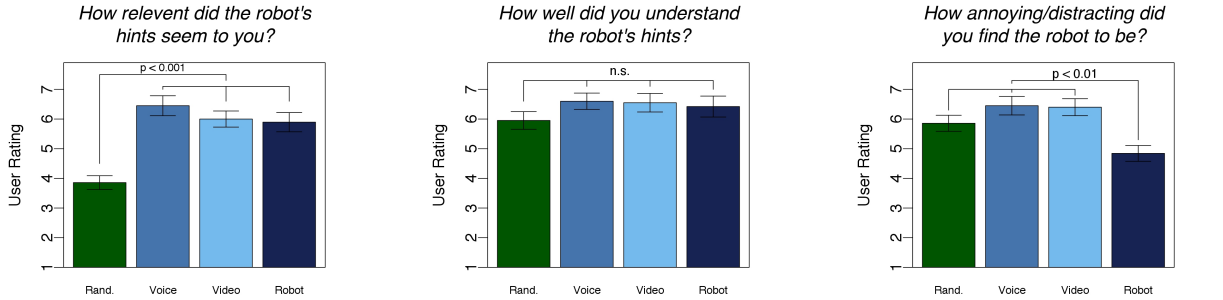
One such factor may be the novelty of the stimulus. Robots are an uncommon stimulus in the present day; we may expect participants to be more attentive to the agent in the physically-present condition. However, novelty can also be a distraction. The physical presence of the robot during the game may divert the participant's attention from the puzzle solving task. More work is needed to identify what effect, if any, a robot's novelty has on interactions such as these.

Physical presence may imbue the robot with more perceived authority than an on-screen agent. Earlier work in this area indicates that people are more likely to comply with commands given by a physically-present robot than an on-screen video of the same robot (Bainbridge et al., 2008). Embodiment may cause participants to take the robot tutor's advice more seriously. We are accustomed to receiving lessons from teachers and authority figures who have physical bodies. Perhaps a robot's physical presence increases its authority or social standing.

Participants, however, did not report having significantly more difficulty understanding the lessons in any of the three advice conditions. In fact, all four groups rated their level of understanding of the lessons fairly highly ($M = 6.3, SD = 1.3$); see Figure 5(b). This may indicate that the embodiment effect is so subtle that the participants did not notice its effect on their learning.

Another social factor is the potentially increased sense of peer pressure during the performance of the task itself. The distinction between physically-present robot and on-screen agent may parallel the way we perform tasks when we think of ourselves as alone rather than in view of another person. In person-person interactions, social presence can lead to significantly worsened task performance, especially in cognitively-demanding tasks (Short, 1976). More work is needed to compare the potential effect of peer pressure caused by a physically-present robot tutor to the peer pressure exerted by a human tutor who observes as participants perform

Back to Top



(a) Participants in the three *personalized advice conditions* rated the lessons as significantly more relevant than those in the *random condition*, $p < 0.001$.

(b) There is no significant difference between groups in self-report of level of understanding of the lessons.

(c) Participants in the *robot condition* rated the robot as significantly less "annoying/distracting" than those in the other *personalized advice conditions*, $p < 0.01$.

Figure 5: Results of self-report questionnaire measures completed after the interaction.

cognitively-demanding tasks.

Social presence effects may also be responsible for the survey result in which participants in the *physically-present robot condition* rated the robot ($M = 4.7, SD = 1.8$) as significantly less "annoying/distracting" than participants in the *other advice conditions* ($M = 6.1, SD = 1.3$), $t(33) < 0.01$; see Figure 5(c). This may indicate that physical embodiment produces a significantly greater sense of social acceptance than an on-screen agent does.

Participants in the *robot condition* became better puzzle solvers than those in the other conditions. Further research is needed to identify the underlying social factors that contribute to this empirically-observed effect.

Conclusion

This study investigates the role of physical embodiment of a robot tutor in a cognitive skill learning task. Participants who received personalized lessons from a physically-present robot outperformed participants who received the same kind of advice from a video representation of the same robot as well as participants who received the same kind of advice from a disembodied voice on the last three puzzles. Participants in the *robot condition* also improved their same-puzzle solving time significantly more than those in any other group, which is a direct measure of learning gains over the course of the experiment. From these data we conclude that physical embodiment can yield measurable learning gains in robot tutor interactions.

Acknowledgments

This material is based upon work supported by grants from the National Science Foundation under contracts No. 1139078, No. 1117801, and No. 0835767.

References

Bainbridge, W., Hart, J., Kim, E., & Scassellati, B. (2008). The effect of presence on human-robot interaction. *Robot and Human Interactive Communication, 2008. RO-MAN 2008. The 17th IEEE International Symposium on*, 701–706.

Dehn, D., & Van Mulken, S. (2000). The impact of animated interface agents: a review of empirical research. *International Journal of Human-Computer Studies*, *52*(1), 1–22.

Kidd, C., & Breazeal, C. (2004). Effect of a robot on user perceptions. *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, *4*, 3559–3564.

Kiesler, S., Powers, A., Fussell, S., & Torrey, C. (2008). Anthropomorphic interactions with a robot and robot-like agent. *Social Cognition*, *26*(2), 169–181.

Kozima, H., Nakagawa, C., & Yasuda, Y. (2005). Interactive robots for communication-care: a case-study in autism therapy. *IEEE International Symposium on Robot and Human Interactive Communication*, 341 - 346.

Leyzberg, D., Avrunin, E., Liu, J., & Scassellati, B. (2011). Robots that express emotion elicit better human teaching. *6th International Conference on Human-Robot Interaction*, 347–354.

Moundridou, M., & Virvou, M. (2002). Evaluating the persona effect of an interface agent in a tutoring system. *Journal of Computer Assisted Learning*, *18*(3), 253–261.

Nagao, T., Ueda, N., Ueda, N., Sato, C. P. T., & Watanabe, C. P. O. (1996). *Np-completeness results for nonogram via parsimonious reductions* (Tech. Rep.). Tokyo Institute of Technology.

Nkambou, R., Bourdeau, J., & Psyché, V. (2010). Building intelligent tutoring systems: An overview. *Advances in Intelligent Tutoring Systems*, 361–375.

Short, W. E. . C. B., J. (1976). *The social psychology of telecommunications*. London, England.

Tapus, A., Tapus, C., & Mataric, M. (2009). The role of physical embodiment of a therapist robot for individuals with cognitive impairments. *Robot and Human Interactive Communication, 2009. RO-MAN 2009. The 18th IEEE International Symposium on*, 103–107.

Wainer, J., Feil-Seifer, D. J., Shell, D. A., & Mataric, M. J. (2007). Embodiment and human-robot interaction: A task-based perspective. *IEEE Proceedings of the International Workshop on Robot and Human Interactive Communication*, 872-877.

Back to Top

Assessing Cognitive Skill Proficiency with Robotic Tutors

Daniel Leyzberg
Yale University
Dept. of Computer Science
daniel.leyzberg@yale.edu

Samuel Spaulding
Yale University
Dept. of Computer Science
samuel.spaulding@yale.edu

Brian Scassellati
Yale University
Dept. of Computer Science
scaz@cs.yale.edu

ABSTRACT

For a robot to be an effective tutor, it must be able to assess an individual student's strengths and weaknesses and adjust its teaching practices accordingly. To create a model of the internal cognitive processes of the student, the robot must decipher the often complex many-to-many relationship between what a student does in any given situation and what those actions indicate about the underlying skill proficiencies of the student. In this paper we present two algorithms for cognitive skill assessment and the results of an 80-participant laboratory study investigating the effectiveness of these algorithms in a robot tutoring scenario. In the experiment, a robot tutor observed participants solve a series of puzzles and infrequently interrupted the participant to deliver one of several pre-recorded puzzle-solving strategy lessons. The number of lessons each participant received was constant, but the order they received them varied in one of four ways: participants either received *no lessons*, *randomized lessons*, or one of two varieties of *personalized lessons*. Both kinds of *personalized lessons* were chosen based on an assessment of the participant's skills as an interpretation of his or her performance thus far. Participants in both *personalized lessons groups* solved three of four puzzles significantly faster than those in either the *no lessons* or *randomized lessons* groups. From this we infer that our models produced meaningful learning gains in this tutoring domain. We believe that cognitive skill assessment models such as ours will become a valuable component of human-robot interactions.

Categories and Subject Descriptors

I.2.9 [**Artificial Intelligence**]: Robotics

General Terms

Experimentation

Keywords

education, assessment, robot tutor

1. INTRODUCTION

When you teach a child to play baseball, it's easy to figure out whether he or she has learned to pitch or bat by watching him or her play the game. Baseball skills, like most physical skills, have directly observable correlates that are easy to identify; there is a one-to-one mapping between the observations made of a player and the player's skills. If, for instance, you wanted to assess how well a player pitches, you would not have him or her stand in the outfield and catch fly balls.

However, when it comes to cognitive skills, as opposed to physical skills, a one-to-one mapping between skills and observations typically cannot be achieved. For instance, a calculus teacher, observing a student incorrectly trying to solve an integral, faces the non-trivial task of assessing which skill or skills the student is lacking. More than one skill can be necessary for a single step in a calculus problem and each step taken by a student may reflect proficiency in one of several skills on the part of that student. This is true of many cognitive domains: skills and observations have a complex many-to-many relationship.

Skill assessment is a necessary tool for teaching and tutoring; it's the way a teacher or tutor identifies what kind of assignments a student is prepared for and what additional instruction a student may need. Beyond teaching, skill assessment is a crucial component of many other collaborative interactions. For instance, being able to assess a work-partner's individual strengths and weaknesses enables individuals to identify the source of problems when working together. Skill assessment is also an important tool used when making decisions about the division of labor in collaborative tasks. Many of these collaborative tasks are relevant to human-robot interaction.

In this paper we demonstrate that skill assessment modeling can be a feasible and valuable component of human-robot interaction. We implement two skill assessment algorithms in a robotic tutoring domain that has a many-to-many relationship between skills and observations (Section 2). We conduct an experiment to evaluate the algorithms' effectiveness (Section 3); the results of which indicate that our systems produce significant measurable learning gains (Section 4).

1.1 Related Work

Assessing students' knowledge and skill proficiency is a

Back to Top

topic frequently addressed in Intelligent Tutoring Systems (ITS's) research. ITS's are educational computer programs that produce individualized lessons, advice, and questions for students based on the answers students give to questions chosen by the system. Conventional ITS's are designed as interactive adaptive workbooks, in which students are presented with multiple-choice questions or fill-in-the-blanks equations, written in advance by human teachers. The answers students give to such questions are used to determine which questions are asked next [1].

A skill assessment model is a core component of many ITS's [1]. The majority of these models require a set of multiple-choice questions to be hand-written and hand-mapped to skills in advance, often by expert human teachers [2, 3, 4]. Such systems can produce significant learning gains [1], however, the task of writing questions and answers such that they map unambiguously to individual skills can be difficult. Using an ITS-like skill assessment approach in human-robot interaction is unrealistic because it requires sustained explicit testing that is too invasive and time consuming for most applications.

Some efforts in ITS skill assessment are relevant to our work, such as the Bayesian models used in an algorithm called Knowledge Tracing [5]. The goal of this algorithm is to learn which patterns of multiple-choice answers are indicative of underlying problems in the knowledge or skill competency of a student [6, 7]. We adopt a similar approach in one of our algorithms below but our work differs in that it assesses a user's skill proficiency while allowing the user to freely interact with the world.

In robotics research, there have been comparatively few projects about tutoring, and none that explicitly model a student's skill proficiency. One robot tutor is RUBI, a robot designed to interact with 18 to 24 month old children [8]. RUBI is a humanoid with articulated arms, an expressive face, and a large screen embedded in its midsection on which it displays educational content. RUBI has been used in many research projects (see [8] for an overview); the project most closely related to this work is apprenticeship learning, in which the robot builds a model of teaching based on pre-recorded observations of human teachers [9]. Our work differs in that our system operates on-line and produces an assessment of the student.

Other research on robot tutors includes a classroom-feasibility study of the iRobiQ, a humanoid similar in design to RUBI [10]. Research has also been done on tutoring robots that operate as museum guides [11, 12] or as semi-autonomous teleoperated instruments of a human teacher, such as the Huggable robot [13]. No previous robotic tutor research, however, has aimed to assess the skill proficiencies of students in order to provide more effective tutoring.

2. SKILL ASSESSMENT

Our approach to skill assessment is to allow users to behave naturally in their environment while the robot interprets observations it makes in real time. The skill assessment algorithm's job is to decide what those observations indicate about the internal cognitive processes of that user. As described above, a significant challenge in modeling these cognitive processes is that each of the user's actions can reflect the presence or absence of many skills at once.

To arrive at an overall skill assessment, it is necessary to unscramble these potentially mixed signals. The algorithms

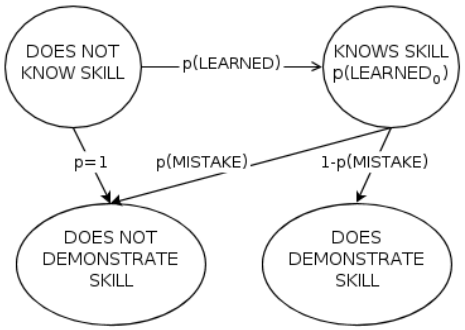


Figure 1: We propose a Bayesian network skill assessment algorithm in which the assessment of each skill is determined by an independent network of the form illustrated above. The upper nodes represent internal states of the skill assessment algorithm and the lower nodes correspond to the robot's observations of the user. For details see Section 2.2.

described below produce an online vector of skill assessments for a predefined set of skills, given a set of observations. The skills themselves are defined in advance by an expert in the subject domain and may overlap in the kinds of actions they prescribe. The responsibility of a skill assessment algorithm is to interpret and combine the output of these functions over time.

For our purposes, a skill i is defined as is defined as a function s_i that maps potential states of the world ($w \in W$) before the application of that skill to potential resulting states of the world once that skill was applied. If a skill is not applicable to the state w, then $s_i(w) = \{w\}$.

The skill functions are designed to be used in two ways:

* Skill functions are used to identify successful demonstrations of a skill. Skill i is said to be demonstrated at state w_t if $w_t \in s_i(w_{t-1})$.

* Skill functions are used to identify missed opportunities to demonstrate a skill. Skill i is said to have went undemonstrated at previous state w_t if no action was taken and $s_i(w_t) \neq \{w_t\}$.

We offer two competing algorithms for interpreting these signals below.

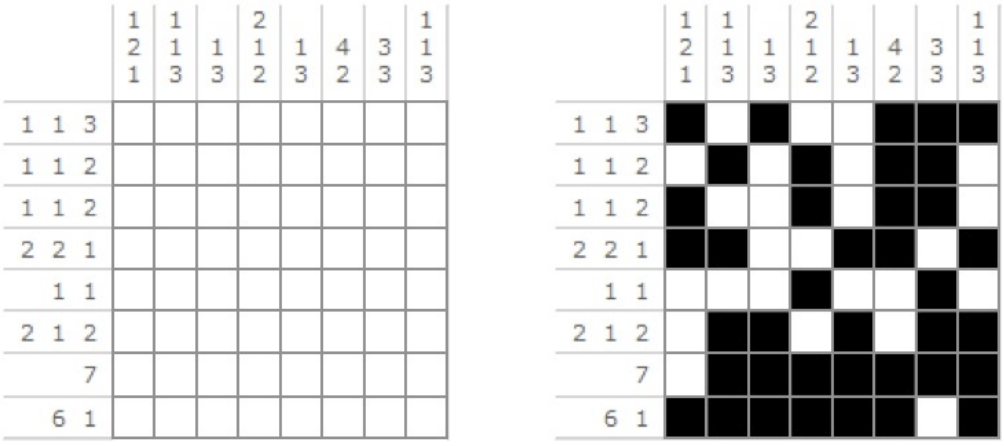
2.1 Additive Skill Assessment

In our first approach, we use a simple additive model to update a vector of skill proficiency assessments over time.

Given a set of skill definitions $s_i \in S$, this algorithm produces two internal Boolean functions for each skill i: a positive indicator p_i and a negative indicator n_i. p_i takes as input the previous and current world states and determines whether proficiency in skill i could have been responsible for this state transition ($w_t \in s_i(w_{t-1})$). n_i takes as input a world state and determines whether the i^{th} skill is applicable to that state ($s_i(w_t) \neq \{w_t\}$).

The positive indicator functions are evaluated every time the state of the world changes. The negative indicator functions are evaluated every time the state of the world does not change for a given delay, and at regular time intervals thereafter until the user changes the state. In our implementation below, the initial delay was set to 3 seconds and the

Back to Top



(a) Sample nonogram puzzle, blank.

(b) Sample nonogram puzzle, solved.

Figure 2: A sample nonogram puzzle. The objective of nonograms is, starting with a blank board (see left figure), to find a pattern of shaded boxes on the board such that the number of consecutively shaded boxes in each row and column appear as specified, in length and order, by the numbers that are printed to the left of each row and above each column (see right figure). For a more detailed explanation see Section 3.2.1.

regular interval was 1 second. These time delays are dependent on the task; the ones used in this paper were chosen based on the authors' subjective experience with the task domain.

The skill assessment $a_{i,t}$ for each skill i is produced at time t as follows:

$$a_{i,t} = d + \sum_{j=0}^{t} (\omega_p p_i(w_{j-1}, w_j) - \omega_n n_i(w_j))$$

Each skill assessment $a_{i,t}$ starts at an initial seed value of $d = 50\%$, and is incremented or decremented by a linear combination of the positive and negative indicator signals . The relative weights of positive indications (ω_p) to negative indications (ω_n) will vary with expected relative frequencies of positive to negative indicators. In this application, we expected positive indications to be rare relative to negative indications; the weights we used were $\omega_p = 50\%, \omega_n = 1\%$. A floor of 0% and a ceiling of 100% is applied to the summed value at each timestep. Generally, the weights and seed value used in this algorithm were subjectively derived and fine-tuned based on pilot studies.

2.2 Bayesian Network Skill Assessment

One weakness of the additive skill assessment algorithm is that it is very susceptible to local maxima and minima. Because individual skill assessments frequently reach floor or ceiling, the additive algorithm often makes assessments based on only the most recent observations. A good human tutor, however, does not dismiss previous successes or failures in light of more recent observations.

We addressed this weakness by offering a Bayesian network approach. Bayesian networks provide a way of modeling the relationship between observations and skill assessment in probabilistic terms. For each skill, we used the same graph structure with assumed independent variables; see graph structure in Figure 1. Skills in this representation are categorized as either learned or not learned.

The following equation is used to estimate the student's proficiency with each skill individually:

$$p(L_t) = p(L_{t-1}|w_t) + (1 - p(L_{t-1}|w_t)p(L_0) \quad (1)$$

where $p(L_t)$ is the sum of the posterior probability that the rule was already learned, regardless of the current world state and the probability that the rule will make the transition to the learned state if it is not learned. We did not choose to model the potential of forgetting a skill because our application is to an interaction that lasts less than an hour and thus we deemed forgetting a skill relatively unlikely.

The two parameters are learned based on observations via an Expectation Maximization (EM) algorithm. Expectation Maximization requires starting points for each value it estimates; in the experiment described below, EM was implemented with seed values of 0 for p(LEARNED) and .5 for p(MISTAKE). EM alternates between performing an expectation step, which creates a function for the expectation of the log-likelihood evaluated using the current estimate for the parameters, and a maximization step, which computes parameters maximizing the expected log-likelihood. This method is commonly used to estimate the unknown parameters of a Bayesian network.

For both algorithms, the result is a vector of likelihoods or beliefs that the tutor has about the users proficiency with each individual skill. That information can be used to customize a robot's behavior to suit the individual strengths and weaknesses of a user.

3. METHOD

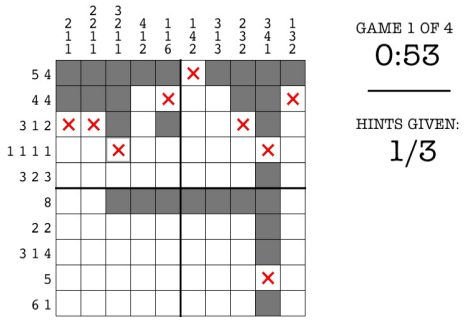
To assess the effectiveness of the models proposed above, an experiment was conducted in a sample robot tutoring domain.

3.1 Participants

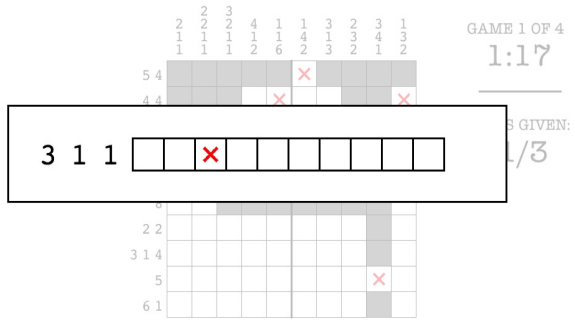
In this experiment there were 80 participants, between 18 and 40 years of age, from New Haven, Connecticut. Most participants were undergraduate and graduate students of Yale University, none of whose academic focus was com-

Back to Top

(a) Participants solve a nonogram puzzle on the computer as the robot (Keepon) analyzes the moves they make and, three times per puzzle, delivers brief lessons (21 – 47 sec.) about gameplay strategies.



(b) A screenshot of the nonogram puzzle user interface during gameplay. All boards start blank. Participants played four puzzles for a maximum of fifteen minutes per puzzle.



(c) A screenshot of the tutoring user interface. The robot 'speaks' and moves in pre-recorded spoken messages and motions and the screen displays examples synchronized to the voice and motion.

Figure 3: Experiment apparatus and user interface screenshots.

puter science. This study is a between-subjects design with four groups of 20 participants: receiving either *no lessons*, *randomized lessons*, *personalized lessons via an additive algorithm*, or *personalized lessons via a Bayesian network algorithm*. Exclusion criteria were a lack of English fluency or prior academic experience with robotics or artificial intelligence.

3.2 Apparatus

In this experiment, participants were asked to solve several logic puzzles; the robot tutor interrupted several times per puzzle to deliver puzzle-solving strategy lessons. In the two *personalized lessons conditions*, the robot used one of two skill assessment models (described in Section 2) to choose which among a set of pre-recorded lessons to give the participant, whereas in the *randomized lessons condition* the robot picked a random pre-recorded lesson among those applicable to the current state of the board, and in the *no lessons condition* participants solved the puzzles with no help. We compare the puzzle solving time performance between participants in these four groups to evaluate the contribution of the skill assessment system.

In this study, a robot delivered the gameplay lessons, though it would be reasonable to think that an on-screen agent delivering the same lessons may have produced similar learning gains. The authors have investigated this hypothesis with this same apparatus and found that not to be the case; participants who were tutored by a robot outperformed those who received the same lessons from an on-screen video version of the same robot [14]. The physical presence of a robot is beneficial in such an interaction.

3.2.1 *Domain: Cognitive Puzzle-Solving*

To minimize the influence of prior experience, we chose a test domain to which participants likely had little previous exposure: a grid-based fill-in-the-blanks puzzle game called "nonograms" (or "nonogram puzzles") that resemble crossword puzzles or Sudoku. Nonogram puzzles are a difficult cognitive task, one that requires several layers of logical inferences to complete. Solving a nonogram puzzle of arbitrary size is an NP-complete problem. In addition, although each move in the puzzle requires a set of logical inferences, observing a person's moves in this puzzle does not easily or definitively reveal the player's strategies. In this way, nonograms mimic real-world cognitive tasks, in which the intentions of a user's actions are not necessarily easily determined from observing the actions themselves.

The objective of nonograms is, starting with a blank board, to shade in boxes on the board such that the number of consecutively shaded boxes in each row and column appear

as specified, in length and order, by the numbers that are printed to the left of each row and above each column. (For an example puzzle and its solution see Figures 2(a) and 2(b).) For instance, a row marked as "4 2" must have 4 adjacent shaded boxes, followed by 2 adjacent shaded boxes—in that order, with no other boxes shaded, and with at least one empty box between the sets of adjacent shaded boxes. We refer to these contiguous sets of shaded boxes as "stretches" in this paper. For instance, the row described above requires two stretches, one of length 4, the other of length 2. One solves the puzzle when one finds a pattern of blank and shaded boxes such that all of the requirements for each row and column are satisfied.

In a typical puzzle, one cannot solve many rows or columns independently. One must infer the contents of parts of rows or columns and use previous inferences as the basis of subsequent inferences. To that end, when a player has reasoned that in some box or boxes there should not be shading, they can mark such boxes with an 'X' for reference.

We created a full-screen nonograms computer program that participants used via mouse and keyboard. The user interface provided a timer and a count of how many lessons (called "hints" in the interface) the participant had received and how many they would receive (see Figure 3(b)).

Participants were asked to play four puzzles on ten-by-ten grids with a time limit of fifteen minutes per puzzle. The puzzles themselves were the same across all participants. The fourth puzzle used the same board as the first, although disguised in the fourth puzzle by rotating the puzzle 90° (such that the column stretch requirements were swapped with row stretch requirements.). This manipulation enables us to make within-subjects comparisons about the extent to which each participant improved over the course of their participation in the study. The data revealed no indication that any participant was aware of this manipulation.

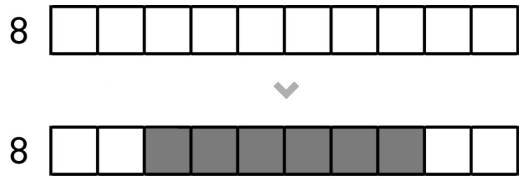
3.2.2 *Skills & Lessons*

Three times per puzzle, the robot interrupted the participant, paused the puzzle, and delivered a short lesson about nonograms. The lessons ranged from 21 seconds to 47 seconds in length and consisted of a voice recording and a set of animations presented on screen during the lesson as well as a set of coordinated robot motions specific to each lesson.

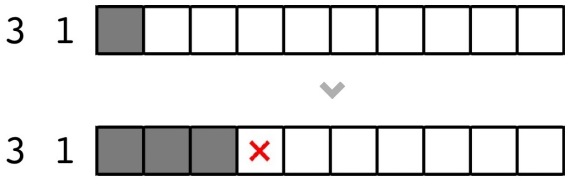
When beginning a lesson the robot would turn to face the participant and say "I have an idea that might help you," or "Here's another hint for you." During the lesson, the robot bounced subtly and looked back at the screen whenever, in the course of the lesson, it would make reference to the example presented on screen. For instance, when in the audio of the lesson the robot would say "Like in this example..." or "As you see here...," the robot would turn briefly to the screen and then back to the participant.

Ten nonogram puzzle-solving skills were identified based on the subjective experience of the authors; they are not universally identified skills or rules for nonograms. Each skill is a set of row or column states in which one can logically fill in some of remaining empty boxes. For example, a stretch of length 9 can fit in a blank row or column of 10 boxes in only two ways. Either it fills the first box and 8 more, or it fills those same middle 8 boxes and the last box. In either case, the middle 8 boxes are shaded. One of the ten skills in this experiment is that, for an empty row or column with just one stretch requirement of n where $n > 5$, the middle $(2n - 10)$ boxes are shaded. For examples of this skill and two others, see Figure 4.

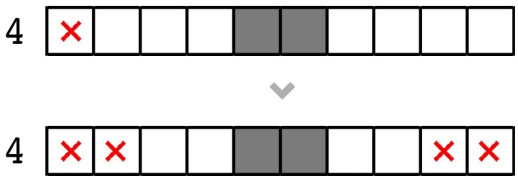
There was one recorded lesson for each skill. Three lessons were delivered per puzzle, for each of four puzzles. The number of lessons was constant for all participants regardless of how long they needed to finish the puzzle. Lessons were triggered either when a participant made no moves for 45 seconds or as he or she filled the 25^{th}, 50^{th} or 75^{th} box on the board (of 100). Participants were informed in the user interface of how many lessons were remaining for each puzzle.



(a) In this row, there must be one long stretch. By the process of elimination one can infer that this stretch must occupy at least the middle six boxes, no matter where in the row it is placed.



(b) In this row, the first box is already shaded. Given that, and that the first stretch must be 3 boxes long, one can infer that the first three boxes must be shaded and the fourth must be crossed out.

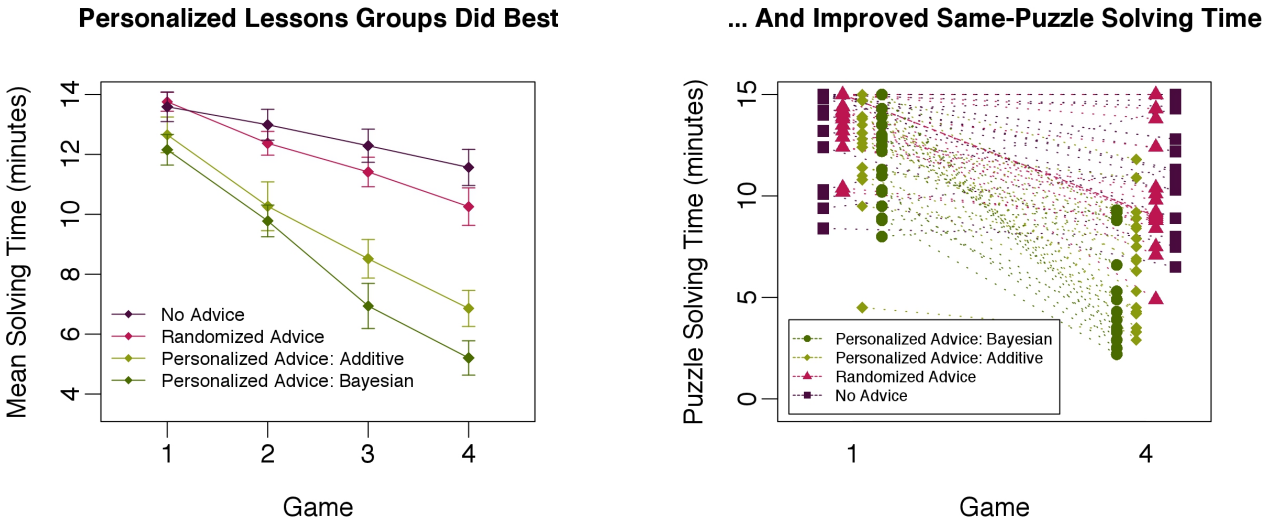


(c) In this row, there is only one short stretch and some boxes are already shaded. One can infer that regardless of where that one stretch is placed, it cannot occupy the first two or the last two boxes in that row.

Figure 4: Examples of nonograms skills. Displayed are the contents of a row before and after each skill is applied. Although only rows are shown here, all nonograms skills apply to columns as well.

The lessons were chosen based on the participant's experimental condition: either the lesson corresponding with the skill that had the lowest on-line internal skill assessment score (in either *personalized lessons condition*) or randomly chosen from among the applicable lessons to the current game board (*randomized lessons condition*). In all conditions, the only lessons eligible to be given were ones that had an available application for the current board. This ensures that each lesson provides information that is actionable at the time the lesson is given.

Back to Top



(a) Mean solving time per puzzle. Participants in both *personalized lesson groups* solved each puzzle except the first puzzle significantly faster than participants in either of the control groups. For means and standard deviations see Table 1.

(b) Individual participants' solving time on the first and fourth puzzles. Both puzzles consisted of the same game board, though disguised in the fourth game by rotating the board 90°. Participants in the *personalized lessons groups* improved their time significantly more than participants in the *control groups*, $p < 0.01$.

Figure 5: Results of the experiment: (a) Participants whose lessons were picked by either algorithm solved most puzzle faster than participants in either *control group*; see Table 1. (b) Participants in the *personalized lessons groups* also significantly improved their same-puzzle solving time over participants in the *control groups*.

3.2.3 Robot

The robot we used, Keepon, is a small yellow snowman-shaped robot (see Figure 3(a)). Keepon has previously been used as an emotive non-threatening communication tool [15, 16].

During the experiment, the robot operated in one of three modes. First, it refereed the puzzle game: it welcomed participants when they started, told them when they had finished or when they had run out of time, and told them when the experiment was over. Second, it "observed" the board during gameplay: the robot frequently turned its head to face the location of the mouse cursor. Third, it delivered short gameplay lessons three times per puzzle: it "spoke" to the participant by turning to face him or her and "bouncing" its body subtly while playing one of several pre-recorded spoken messages. If a lesson needed to be repeated, the robot would first apologize for repeating itself (i.e., "I'm sorry to repeat this hint but I think it might help.").

To simplify the potential perception problems inherent in real-world measurements, the robot in this study received perfect knowledge of the state of the puzzle. We did not use a robotic vision system to detect state changes.

3.3 Procedure

Participants were first asked to watch a five minute instructional video describing the rules of nonograms and how to use the computer interface. In this video, participants were encouraged to use logical reasoning to make moves in the game, rather than making moves by guessing. Questions about the game were answered by the experimenter after the video.

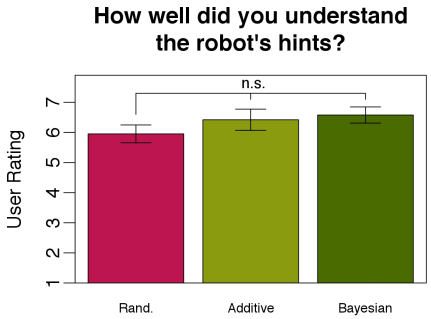
During the experiment, participants were alone in a room with the robot, the computer, and a video camera positioned behind them (see Figure 3). Participants would choose when they were ready to start each new puzzle; each game would end either when the participant solved the puzzle or when fifteen minutes had elapsed, whichever happened first.

After the conclusion of the final puzzle, participants were asked to complete a survey consisting of three open-ended questions and five Likert-scale questions. The questions were designed to assess whether the lessons were helpful, clear, and influential, as well as the user's perceptions of the robot. We asked participants to rate how relevant the lessons were, how much the lessons influenced their gameplay, how well participants understood the lessons, and how "smart/intelligent" and "distracting/annoying" they perceived the robot to be.

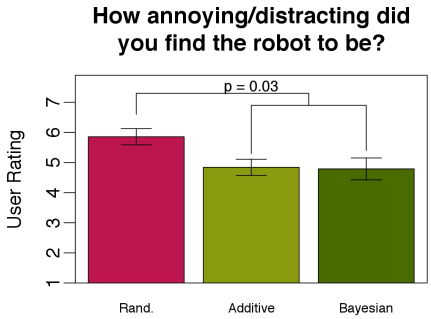
4. RESULTS

The main hypothesis of this study is that lessons chosen by our skill assessment algorithms will produce measurable learning gains compared to no lessons and compared to lessons chosen randomly from among actionable lessons. The behavioral measure is the time participants took to solve each of the four puzzles. For the purposes of calculating a mean, games in which participants ran out of time were evaluated as having finished when time ran out: fifteen minutes from the start of each game. The rate of failure was not significantly different between groups for any of the four games; it varied from highs of 29% to 38% in the first game to lows of 9% to 17% in the fourth game.

Back to Top



(a) There is no significant difference between groups in how participants rated their understanding of the lessons.



(b) Participants in both *personalized lessons groups* rated the robot as significantly less "distracting/annoying" than those in the *randomized lessons group*, $p = 0.03$.

Figure 6: Results of two survey questions answered by participants upon completing the experiment.

	Game 1	Game 2	Game 3	Game 4
None	12.7 ± 2.6	10.0 ± 3.5	9.4 ± 3.0	7.6 ± 3.1
Random	13.8 ± 1.4	12.5 ± 2.0	11.4 ± 2.3	10.3 ± 2.9
Additive	12.7 ± 2.6	10.0 ± 3.5	9.4 ± 3.0	7.6 ± 3.1
Bayesian	12.2 ± 2.3	9.8 ± 2.4	6.9 ± 3.4	5.2 ± 2.6

Table 1: Solving time means and standard deviations, in minutes. In each game except the first game, participants in both *personalized lessons groups* solved the puzzle significantly faster than participants in both the *randomized lessons* and *no lessons* groups ($p < 0.03$ for all).

Participants in both *personalized lessons groups* solved three of four puzzles significantly faster, on average, than those in either the *randomized lessons* or *no lessons* groups, $p < 0.03$ for all comparisons. See Table 1 and Figure 5(a). These results confirm the main hypothesis: both skill assessment algorithms produced significantly more learning gains than the control conditions.

Between *personalized lesson groups*, the *Bayesian group* did significantly better on the last puzzle than the *Additive group*, $t(37) = 0.05$.

In this study, the first and fourth puzzles consisted of rotated variations of the same board. The difference in completion times between the first and fourth puzzles is a within-subjects measure of an individual participant's improvement over the course of the experiment. According to this metric, participants in either *personalized lessons group* improved ($M = 5.8$ minutes, $SD = 3.3$) their same puzzle solving time significantly more than those in either *control group* ($M = 3.1$ minutes, $SD = 2.4$), $t(31) < 0.01$. See Figure 5(b).

Survey results indicate participants in the *personalizes lessons groups* rated the lessons significantly more relevant to them ($M = 4.9, SD = 1.4$) than participants in the *randomized lessons group* ($M = 2.9, SD = 1.1$), $t(33) < 0.001$. There was no significant difference, however, in how highly participants rated their understanding of the lessons between groups, ($M = 5.4, SD = 1.5$) in the *personalized lessons groups* and ($M = 5.0, SD = 1.4$) in the *random condition*, $t(36) = 0.32$. Participants in the *personalized lessons groups* rated the robot as smarter or more intelligent ($M = 4.7, SD = 1.8$) than participants in the *random condition* ($M = 3.5, SD = 1.6$), $t(36) = 0.03$.

5. DISCUSSION

The experiment presented here assesses whether our two skill assessment algorithms, based on observations of students' natural behavior, produces significant learning gains in a sample tutoring domain against two control conditions. The data indicate that both skill assessment algorithms successfully produced significant learning gains for the domain we chose. We believe these algorithms are the first step in producing a generalizable (and more sophisticated) skill assessment algorithms for human-robot interaction.

Creating a generalizable and widely applicable skill assessment algorithm is no trivial task. The algorithms presented here are just the beginning. The additive model is the simpler of the two to implement but limited in its applicability to only simple domains and short-term interactions. The Bayesian model we propose is more robust and performed better even in this test domain. One of the biggest challenges in creating a generalizable algorithm is whether it is reasonable to define skills the way we did, as transitions between potential world states. Will this notion translate pragmatically into applications with real-world measurements and more interesting task domains? We believe it might, under the right circumstances.

For this sample tutoring task, the survey ratings indicate that participants did not report having more difficulty understanding the lessons presented to them in the *randomized lessons group* than in either *personalized lessons group*. All three groups rated their level of understanding of the lessons fairly highly: a mean of 5.4 across *personalized lessons groups* and 5.0 in the *randomized lessons group* out of 7, $t(36) = 0.32$ (see Figure 6(a)). It is interesting to see that the *randomized lessons group* reported understanding the lessons, even though they did not demonstrably use that understanding. This self-report measure contradicts the performance measure results; this may simply be because people are reluctant to admit there is something they do not understand.

Judging by survey free-response data, participants in this study ranged greatly in their own evaluation of the usefulness of the lessons. In both experimental conditions, some reported very positive feedback about the lessons while others reported only frustration. An example of the free-response answers about the helpfulness of the lessons from participants in the *randomized lessons group*: "Lessons were

repetitive and a little distracting, even frustrating." An example from one of the *personalized lessons groups*, in response to whether the lessons affected his or her gameplay: "Not really. I just learned by seeing what worked & what did not." However, judging by their performance data, some of these participants who reported discounting the value of the lessons, seemed to benefit from them just as much as other participants. From this result and the previous result we conclude that a student's perceived value of a lesson does not always match the real impact of that lesson.

The free-response survey data also offers insight into the perception and impact of "easy" lessons. Most participants reported that they found some lessons "easy," "obvious," or "very obvious;" however, providing those lessons may not have been a waste. Eight participants reported a variant of the following sentiment: "Most hints I had previously explicitly figured out, though I found myself more actively seeking the pattern(s) suggested by a hint in the minutes that followed." These statements indicate that the value of this skill assessment algorithm is not only to identify skills that a user does not know, but also to reinforce skills that a user does know. The participants that thought some lessons were "obvious" perhaps unwittingly benefited from having been reminded of them at appropriate times during gameplay.

6. CONCLUSION

In this paper, we describe the cognitive skill assessment problem space in human robot interaction and we offer two skill assessment algorithms evaluated in a puzzle-game robotic tutoring domain. Participants who received lessons that were chosen based on either skill assessment algorithm outperformed participants in either control condition (randomized but applicable lessons and no lessons) in three of four puzzles. Participants in the *personalized lessons groups* also improved their same-puzzle solving time significantly more than those in the *control groups*. These data indicate that our skill assessment models produced measurable learning gains and that skill assessment modeling can be a feasible and valuable component of human-robot interaction.

7. REFERENCES

[1] R. Nkambou, J. Bourdeau, and V. Psyché, "Building Intelligent Tutoring Systems: An overview," *Advances in Intelligent Tutoring Systems*, pp. 361–375, 2010.

[2] A. Corbett, "Cognitive computer tutors: Solving the two-sigma problem," in *User Modeling 2001*, ser. Lecture Notes in Computer Science, M. Bauer, P. Gmytrasiewicz, and J. Vassileva, Eds. Springer Berlin Heidelberg, 2001, vol. 2109, pp. 137–147.

[3] K. Vanlehn, C. Lynch, K. Schulze, J. A. Shapiro, R. Shelby, L. Taylor, D. Treacy, A. Weinstein, and M. Wintersgill, "The Andes physics tutoring system: Lessons learned," *Int. J. Artif. Intell. Ed.*, vol. 15, pp. 147–204, August 2005.

[4] M. Feng, J. Beck, N. Heffernan, and K. Koedinger, "Can an intelligent tutoring system predict math proficiency as well as a standardized test?" *First International Conference on Educational Data Mining*, pp. 107–116, 2008.

[5] J. Beck, K.-m. Chang, J. Mostow, and A. Corbett, "Does help help? Introducing the Bayesian evaluation and assessment methodology," in *Intelligent Tutoring Systems*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2008, vol. 5091, pp. 383–394.

[6] R. S. Baker, A. T. Corbett, and V. Aleven, "More accurate student modeling through contextual estimation of slip and guess probabilities in Bayesian Knowledge Tracing," in *9th International Conference on Intelligent Tutoring Systems*, ser. ITS '08. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 406–415.

[7] A. T. Corbett and J. R. Anderson, "Knowledge tracing: Modeling the acquisition of procedural knowledge," *User Modeling and User-Adapted Interaction*, vol. 4, pp. 253–278, 1994, 10.1007/BF01099821.

[8] J. R. Movellan, F. Tanaka, I. R. Fasel, C. Taylor, P. Ruvolo, and M. Eckhardt, "The RUBI project: A progress report," *2nd ACM/IEEE International Conference on Human-Robot Interaction*, pp. 333–339, 2007.

[9] P. Ruvolo, J. Whitehill, M. Virnes, and J. Movellan, "Building a more effective teaching robot using apprenticeship learning," *7th IEEE International Conference on Development and Learning*, pp. 209 –214, aug. 2008.

[10] E. Hyun, H. Yoon, and S. Son, "Relationships between user experiences and children's perceptions of the education robot," *5th ACM/IEEE International Conference on Human-Robot Interaction*, pp. 199–200, 2010.

[11] K. Yamazaki, A. Yamazaki, M. Okada, Y. Kuno, Y. Kobayashi, Y. Hoshi, K. Pitsch, P. Luff, D. vom Lehn, and C. Heath, "Revealing Gauguin: Engaging visitors in robot guide's explanation in an art museum," *27th International Conference on Human Factors in Computing Systems*, pp. 1437–1446, 2009.

[12] M. Bennewitz, F. Faber, D. Joho, M. Schreiber, and S. Behnke, "Towards a humanoid museum guide robot that interacts with multiple persons," *5th IEEE-RAS International Conference on Humanoid Robots*, pp. 418 –423, dec. 2005.

[13] J. K. Lee, R. Toscano, W. Stiehl, and C. Breazeal, "The design of a semi-autonomous robot avatar for family communication and education," *Robot and Human Interactive Communication, 2008. RO-MAN 2008. The 17th IEEE International Symposium on*, pp. 166 –173, aug. 2008.

[14] D. Leyzberg, S. Spaulding, M. Toneva, and B. Scassellati, "The physical presence of a robot tutor increases cognitive learning gains," in *Proceedings of the 34th Annual Conference of the Cognitive Science Society*. Austin,TX: Cognitive Science Society, 2012.

[15] H. Kozima, C. Nakagawa, and Y. Yasuda, "Interactive robots for communication-care: A case-study in autism therapy," *IEEE International Symposium on Robot and Human Interactive Communication*, pp. 341 – 346, aug. 2005.

[16] D. Leyzberg, E. Avrunin, J. Liu, and B. Scassellati, "Robots that express emotion elicit better human teaching," in *6th International Conference on Human-Robot Interaction*. New York, NY, USA: ACM, 2011, pp. 347–354.

Back to Top

The Effects of Robotic Agency on Trust and Decision-Making

Sam Spaulding
samuel.spaulding@yale.edu

Wayne Zhu
wayne.zhu@yale.edu

Devon Balicki
devon.balicki@yale.edu

ABSTRACT

The purpose of this project is to investigate the effect of social factors on robotic trust and irrational decision-making. We recorded results from 20 participants, consisting of gameplay data from a unique card game and ratings from a post-experiment survey. During each of 12 rounds of gameplay, participants received advice from a robotic partner that they could choose to accept or ignore. We devised a 2-by-2 experiment, in which participants were split into four groups based on two factors: the robot was either social or nonsocial, and it gave either consistently bad advice or gradually transitioned from good to bad advice. We tracked the pattern in which each participant accepted the robot's advice. Furthermore, in the post-experiment survey, participants were asked to rate how human-like, trustworthy, and likeable the robot seemed. Our results suggest that social behaviors strongly influence participants' level of trust, but that these same features cause a robot to appear more fallible. Therefore, participants in the social cases were less likely to allow the advice to affect their gameplay. We further found that participants perceive a nonsocial robot's behavior as relatively static and a social robot's as relatively variable. Our results highlight the importance of a robot's social engineering on building trust with humans.

1. INTRODUCTION

What makes someone, or something, trustworthy? Can we identify these factors and use them to influence a human's decision-making process? What sorts of prejudices and stereotypes do humans hold about robots that affect the way they interact in a game setting? These were the questions that initially motivated our project. While they are undoubtedly serious questions for the field of HRI, they lack conclusive answers. Human decision-making has been extensively studied since the birth of behavioral economics in the 1970's, but that work has yet to see real integration with the HRI community. Some work has been done in the area, and it has shown that many factors—such as human personality and experience, machine failure rates and false alarms, and environmental influences and the type of task—influence the level of trust in human-robot collaboration. (Oleson, et al.)[1]

There is also a documented correlation between a human's confidence in his or her decision-making and the credibility of the robot providing him or her with advice in a game of chance. (Weiss, et al.)[2] There has also been an attempt to construct HRI trust scales based on the robot's mechanisms being reliable, dependable, understandable, consistent, and timely. (Yagoda, Gillan)[3]

2. METHODOLOGY

2.1 Experimental Design

To investigate the effect of affective behaviors on decision-making, we devised a game, called "11", which is a derivative of the game Black Jack. The 20 participants were divided evenly into 4 groups. One half faced a nonsocial robot with stoic speech while the other half faced a social robot with emotional speech and hand gestures. We chose Nao, an anthropomorphic and expressive robot for this experiment. The social and nonsocial subsets were further divided into two conditions: one in which Nao gives only bad advice, and another in which Nao gradually transitions from good to bad advice. The participants in the former condition represent the control group, and those in the latter represent the experimental group.

The participants played the same 12 rounds of our game and were told that they would be part of a machine learning study. We stressed the importance of playing to the best of one's ability by telling the participants that their results were crucial for providing Nao with good training data. We recorded each participant's pattern of play, whether they accepted or ignored the robot's advice each round, and whether they won or lost each round.

The participants were instructed to fill out a brief survey upon completion of the game. Each assigned a rating on a scale of 1 to 7 (7 being the best) to the following facets of his or her interaction with the robot: the quality of its advice, its likability, its trustworthiness, and how human it appeared. Furthermore we asked the participants to rate their own performance at the game, again on a scale of 1 to 7, and to list as many media references (e.g., books, movies, etc.) to robots or artificial intelligence they could think of.

2.2 Description of the Game

The game of "11" works in the following way. It is a text-based simulation of a game involving a standard deck of cards with two jokers. The players' hands begin with two

Back to Top

cards (one face-up, and one face-down), and each turn the player has the option to draw a card (hit) up to three times per round. Alternatively, the player can opt not to hit (stay), but after choosing to stay, for the rest of the hand the player cannot hit. The player's total is a running sum of cards modulo 12, i.e., 12 is subtracted from the total whenever it exceeds 11. A player's final score at the end of the game is the player's total minus the number of the times the player chose to hit. In effect, each hit "costs" the player one point from his or her score. After both players have finished drawing, the face-down cards are revealed, and the player with the highest score wins. The scoring is tallied as follows: number cards are worth their corresponding value, face cards are worth 10, and aces are worth 11. A joker makes a player's total worth 11, regardless of the other cards the player holds. In our design, each participant faced the same predetermined order of cards for both themself and the dealer.

We kept the order constant because otherwise, the number of test runs necessary to reduce the variability of hands exceeded our resources. Before beginning the game, the human subject was provided with a detailed explanation to the game and was asked to begin once the subject declares that he or she is comfortable with the explanation and is ready to play. The subject is then told that the experiment is meant to teach the robot how to play the game, and that it was important that the participant play to the best of his or her ability so as to give the robot the best opportunity to learn.

The subject played 12 rounds of "11" against a computer dealer, and for each step within each round, Nao offered advice in the form of "hit" or "stay." The condition in which the subject had been assigned determined whether the Nao advised the subject to hit or stay, and the quality of this advice. A piece of good advice was defined as a choice that would increase the subject's probability of winning given his or her current information. A piece of bad advice was defined as the opposite. In the control case, Nao's advice was bad every round. In the experimental case, Nao's advice was good at first but gradually converged to being as bad as in the control case. Nao's advice for each round was predetermined and constant across all trials within each case.

2.3 Robot Behavior

In the nonsocial case, the robot remained in a sitting position and only spoke. It used quantitatively phrased advice such as "There is a high probability you will win if you hit" or "My calculations say that there is an *X* percent chance you will win if you hit." X was a randomly generated number from 55 to 90.

In the social case, the robot's arms made various gestures as it spoke. In addition to the quantitative speech, the robot provided qualitative commentary as well. Each time the player had the option to hit or stay, the robot said, "Let me think about it." The robot then made a gesture that resembled scratching its head. After putting its hand down, the Nao said, for example, "I think I have the answer. There is a 60% chance you will win if you stay. You have a good score already."

In the social condition, the manner in which Nao reacted to results was also socially appropriate. If the player took Nao's advice but lost the round, the Nao responded with slouching its shoulders, shaking its head, and saying, "Wow. I guess I was wrong." Conversely, if the player took the advice and won the round, the Nao lifted its arms and said, "Yes! I knew it was a good idea." The reactions to the player refusing to follow advice were also distinct and socially appropriate. If the player won, the Nao responded by making a clapping motion and saying "Oh, you won. That was a good choice." If the player lost, the robot shook its fist and said, "You should have listened to me. I am very good at this game."

2.4 Robot Advice

In the control condition, the robot gave clearly bad advice throughout the game. In the experimental condition the robot initially gave good advice and transitioned to bad advice. The transition is nuanced and works in the following way. For the first 3 rounds, the robot maximized the probability of winning given the participant's available information. Throughout the rest of the game, the point at which the robot advised the player to hit steadily decreased, and conversely, the point for staying increased. Finally, in the last 3 rounds, the robot's advice became glaringly wrong.

3. RESULTS

	Advice Quality	Likability	Trust	Human-like	Performance	Number of Refs.
Social Experimental	3.83	4.33	2.83	3.17	5.50	3.50
Social Control	2.00	4.75	2.00	3.25	4.50	2.50
Nonsocial Experimental	4.20	5.40	3.40	2.00	5.20	5.40
Nonsocial Control	2.00	4.60	1.60	2.80	5.40	3.60

Figure 1: Average values of survey responses across groups

Our results begin with a basic analysis of this survey data. The average ratings for each group along with the average number of relevant media references are displayed in Figure 1. For each test group we averaged the ratings for each of the values listed in Figure 1. This provides some intuitive results such as the fact that the participants in the

experimental cases tended to rate both the advice quality and trustworthiness of the robot higher than those in the control case. Also, we see that the social cases tended to rate how human the robot was higher than the nonsocial cases. While these numbers do not produce any deep insights, they do confirm that the experiment was carried out smoothly and that we have enough data to discern existing differences between the test groups.

3.1 Win-rate Differential Between Experimental and Control Groups

Win Rates	Experimental	Control
Social	62.7%	62.2%
Nonsocial	66.7%	61.8%

Figure 2: Percent of rounds won across different conditions

As can be seen from Figure 2, the win-rates across the four groups are fairly close. Part of this clustering can be attributed to the fact that we used the same slate of cards for each player. In many of the rounds, the participants were presented with fairly easy decisions, causing some rounds to be won or lost almost every time. However, it was necessary to carry out the experiment in this manner, because our relatively low sample size could not handle the variability that would have come from completely random rounds for each player. Given the resources of a full-scale experiment, it would be more apt to present entirely random rounds in the gameplay, thereby reducing how clustered the win percentages are.

While there is evidently no statistical significance between the two social cases, we can see that there is nearly a five percent difference between the two nonsocial cases. Upon running a Z-test on the population of the nonsocial experimental case against the average win percentage of the nonsocial control case, we were able to find a significant difference with p-value of .077. While this value is not under the widely accepted threshold for true significance (.05), we posit that it would be if the sample size were increased.

3.2 Patterns of Deviation from Robotic Advice

We then examined the rates at which the participants accepted the robot's advice in each of the four test groups. Displayed in the charts in Figures 3a and3b, we plotted the advice acceptance rates across the twelve rounds of play. Within each round, a player has up to three opportunities to make decisions (labeled *a*, *b*, and *c*), and thus there are up to 36 data points per player. As a caveat, we note that when a player stays early in the round, he never reaches the later decision nodes. Thus, as we go from nodes *a* to *b* to *c* in each round, there are fewer and fewer data points. This limitation of the game design causes large spikes upward and downward in both graphs as one moves towards later decision nodes. In Figure 3a, the section shaded in yellow indicates the rounds that the Nao gave questionable advice, and the section shaded in purple indicates when the Nao gave clearly wrong advice.

The graphs of the experimental cases in Figure 3a along with their corresponding trending lines demonstrate a stark result. Directly opposing our hypothesis, we see that the participants in the social case actually begin to deviate from the robot's advice earlier than those in the nonsocial case. Adding to the magnitude of this finding is the corresponding chart for the control case. Comparing the two cases in Figure 3b, we notice that the rate of acceptance trends downward faster in the nonsocial case. In considering both of these differences, we see that the difference in behavior in the experimental and control cases is quite large for the nonsocial test group. Further discussion of the implications of these graphs lies in the following section.

3.3 Correlation Between Perception of Trust and Experimental Condition

Going beyond the simple averages presented in subsection 3.1, we analyzed the correlations across the various qualities of the robot that were asked about in the post-experiment survey. One striking result arose in examining how correlated trust is with the experimental case. We found markedly different correlations in the social and nonsocial cases, with correlation values of .285 and .805, respectively. Because the advice quality is the central factor that the participant considers when rating the trustworthiness of the robot in the nonsocial case, it can largely predict how trustworthy the participant will view the robot. However, in the social case, the social actions of the robot serve as an emotional appeal to the participant, so these behaviors influence the participant's trust, thus causing the advice quality to be a poorer explanatory variable than in the nonsocial case.

Back to Top

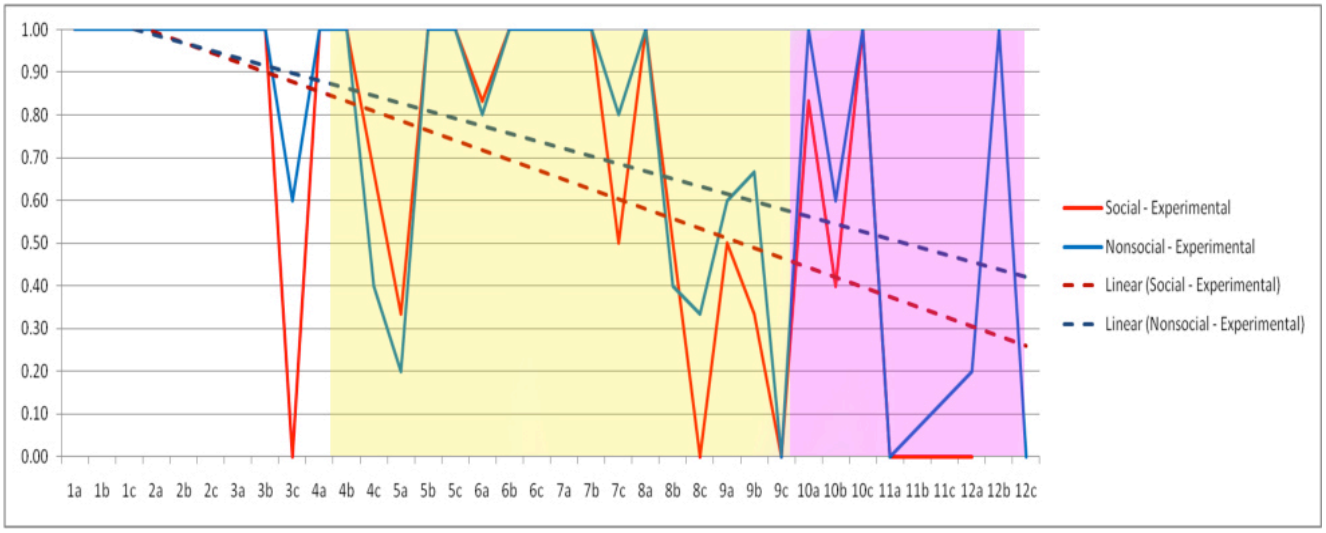


Figure 3a: Advice acceptance and deviance rates in experimental condition

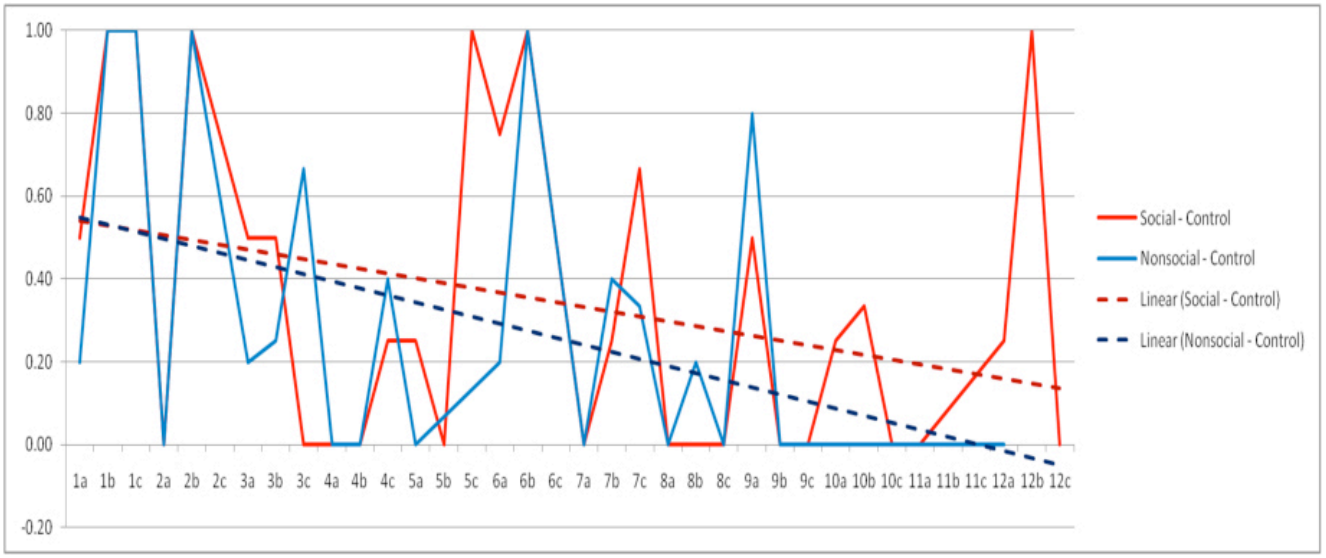


Figure 3b: Advice acceptance and deviance rates in control condition

4. DISCUSSION

4.1 Participants in Nonsocial Condition Let Advice Affect Gameplay More Often

As Figure 2 shows, participants in the social condition do not differ much in win-rate across experimental conditions. Because the only difference between these cases is the quality of advice, this implies that they do not let the robot's advice affect their gameplay very much. Analyzing the survey data (Figure 1), we found a negative correlation between ratings of "how human-like" Nao was and the quality of Nao's advice. This leads us to believe that acting humanly causes participants to attribute more fallibility to the robot and thus take its advice less seriously.

This effect also explains why the nonsocial case was capable of producing a significantly higher win-rate in the experimental condition. Essentially, a nonsocial robot that gives correct answers seems somewhat infallible, thus participants take its advice more often, achieving a higher win-rate than the other three cases.

4.2 Participants Attribute More Variability to a Social Robot's Behavior

Figures 3a and 3b depict the patterns of advice acceptance and rejection among groups of participants. We found some interesting patterns in the rates at which participants deviated from the robot's advice. While we had hypothesized that participants in the social cases would build up more trust in the robot and thus continue to take its advice in the experimental case even as the advice grew worse, instead we found that participants in the social case began to deviate from the robot's advised actions *faster* in

the social experimental case than in the nonsocial experimental case.

We saw the opposite situation in the control case. Participants in the nonsocial condition deviated very quickly from the robot's advice in the control case and continued to deviate from its advice for the rest of the game. In the social control case, participants took longer to realize that the advice was bad, and so deviated later. Furthermore, participants in the social control case exhibited a willingness to try taking the robot's advice again later on in the game. We did not see this in the nonsocial control condition.

We believe this data is explained by the fact that people tend to attribute bad outcomes produced by a nonsocial robot to a malfunction, and attribute bad outcomes produced by a social robot to a decision by the robot. A nonsocial robot is perceived to be using a single computation and thus has two modes to a user: working and broken. If the computation works, then people believe it will continue to work. This is why in the experimental condition, when the nonsocial robot gives good advice at the start, people tend to deviate from its advice more slowly. If the computation does not work, as participants in the control case believed, then it is permanently broken and participants deviate rapidly from its advice.

4.3 Social Factors, When Present, Are Better Predictors of Trust Than Advice Quality

We saw a very significant difference in the correlation between perceived trust and experimental condition across the social and nonsocial cases. Figure 5 shows a strong positive correlation between trust and experimental condition in the nonsocial groups, but a relatively weak correlation in the social groups. Because the only difference in experimental conditions was the quality of advice, this tells us that in social settings, advice quality is less important for building trust than other social, emotional factors.

One possible confound to this data is that people may interpret the pragmatics of the question differently in a social vs. nonsocial setting. Clearly, in a social environment a large component of 'trust' is based on general affective feeling. In a nonsocial environment, when such affective feelings may not be quite so pronounced, 'trust' might be considered synonymous with 'accuracy,' which would explain the strong correlation. Still, we are confident that, had we controlled for this issue, we would still see the same sort of effect, and in any further research this should be kept in mind.

5. CONCLUSION

We established that while social factors do affect trust, they also make a robotic agent seem more fallible. Thus, people tend not to let a social robot's advice affect the way they play the game. The flip side of this effect is that presenting a robot as a nonsocial being plays into people's perception of computers as mechanical automatons that are good at calculations, and therefore people do let its advice affect their in-game decisions. Although we could not establish a significant correlation between social / nonsocial behavior and perception of "humanness" we believe that a follow-up study with more participants that more directly sought to explore this relationship would complement our own research well.

In addition, we found another interesting pattern in participants' patterns of play and their perceptions of the robot. When people perceive a robotic agent as nonsocial, bad advice decisions were seen as symptomatic of a larger problem. A robot that people perceive as "broken" is going to remain "broken," thus people diverged very quickly from following the nonsocial robot's advice in the control condition, in which advice was uniformly bad. One participant, stopped after the second round of gameplay to tell us that he thought the robot was broken or that we had a bug in our code. However, when the advice started out good, this attribution worked the other way. Participants took longer to diverge from a nonsocial robot's advice in the experimental condition, when the advice switched from good to bad.

That we were able to obtain these results from such a small group of participants is highly noteworthy. If we had been able to run the experiment on a larger population, we are confident that additional patterns would have emerged from the data. Several of these trends we noticed, although the effect was not nearly significant enough for us to rule out chance as the underlying cause. However, this gives us additional confidence that the trends we did manage to find significant are genuine and that the effects would likely be even more pronounced in a larger population.

6. REFERENCES

[1] Oleson, Kristen E., D. R. Billings, Kocsis Viven, Jessie Y. Chen, and P. A. Hancock. "Antecedents of Trust in Human-Robot Collaboration." IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support. Web. DOI=http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=05753439

[2] Weiss, Astrid, Roland Buchner, Thomas Scherndl, and Manfred Tscbeligi. "Proceedings of the 4th ACM." HRI '09 Proceedings of the 4th ACM. 259-60. ACM Digital Library. Web. DOI=http://dl.acm.org/citation.cfm?id=1514165.

[3] Yagoda, Rosemarie E., and Douglas J. Gillan. "You Want Me to Trust a ROBOT? The Development of a Human-Robot Interaction Trust Scale." Int J Soc Robot (2012). Springer Science & Business Media. Web. <http://www.springerlink.com/content/p5rhk43w32nh10r3/fulltext.pdf>.Tavel, P. 2007. *Modeling and Simulation Design*. AK Peters Ltd., Natick, MA.

Yale UNIVERSITY

The Physical Presence of a Robot Tutor Increases Cognitive Learning Gains

Daniel Leyzberg, Samuel Spaulding, Mariya Toneva, Brian Scassellati

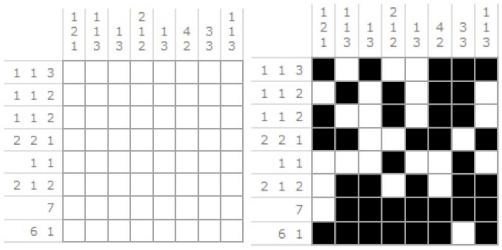
Cognitive Sciences Conference August 1-4th 2012 Sapporo, Japan

Abstract

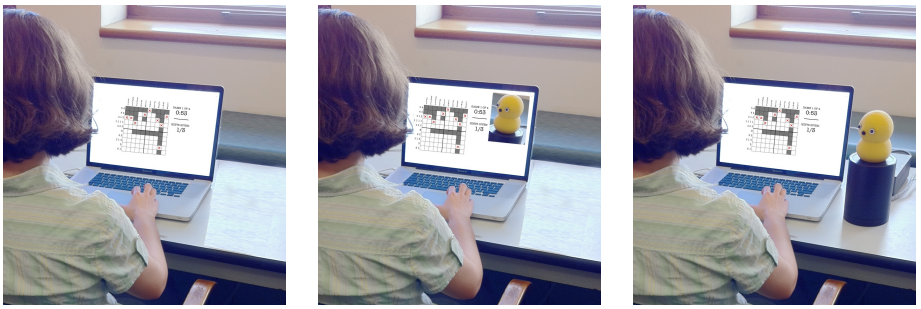
We present the results of a **100 participant study** on the role of a **robot's physical presence** in a robot tutoring task. Participants were asked to solve a set of Sodoku-like puzzles while being provided occasional gameplay advice by one of three automated tutors: physical robot, video of a robot, or robot voice only. Participants in the **physical robot group solved puzzles faster** on average and improved their same-puzzle solving time significantly more than participants in any other group. We conclude that the physical embodiment of a robot can produce measurable learning gains.

Apparatus

"NONOGRAMS"



Sample "nonogram" game, a grid-based fill-in-the-blanks constraint-satisfaction puzzle game resembling crossword puzzles or Sudoku.



(A) DISEMBODIED VOICE (B) VIDEO OF ROBOT (C) PHYSICAL ROBOT

Experiment apparatus varied by condition. There were two control conditions: no advice given (A) and random advice with robot (C). There were three experimental conditions: personalized advice via disembodied voice (A), on-screen video of robot (B), or physically present robot (C).

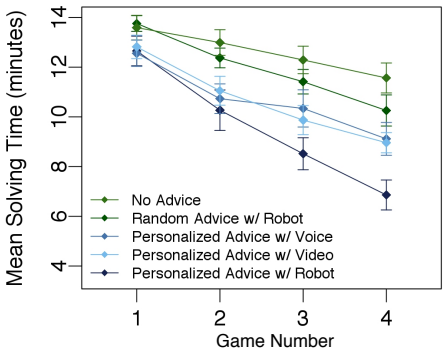
Results

ROBOT GROUP SOLVED PUZZLES FASTEST

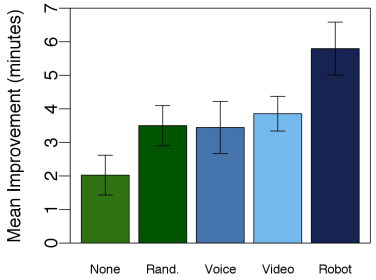
Mean Solving Time (minutes)

	Puzzle 1	Puzzle 2	Puzzle 3	Puzzle 4
None	13.6±2.2	13.0±2.3	12.3±2.5	11.6±2.7
Rand.	13.8±1.4	12.5±2.0	11.4±2.3	10.3±2.9
Voice	12.6±2.4	10.7±2.7	10.3±3.3	9.1±3.0
Video	12.8±2.1	11.1±2.6	9.9±2.6	8.7±2.4
Robot	12.7±2.6	10.0±3.5	9.4±3.0	7.6±3.1

Mean solving time per puzzle. Participants in the robot condition solved each puzzle faster than participants in any other condition. In the fourth puzzle, significantly faster, $p \leq 0.03$.



ROBOT GROUP IMPROVED MOST



The first and last puzzles were **variations of the same gameboard**, disguised by a 90° rotation. Participants in the robot condition improved their **same-puzzle** solving time significantly more than other groups, $p < 0.05$.

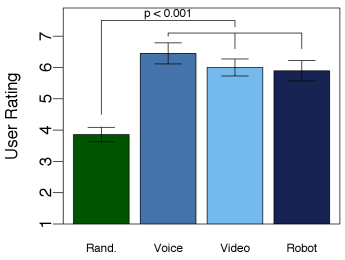
Discussion

Why did the robot's physical presence lead to learning gains?

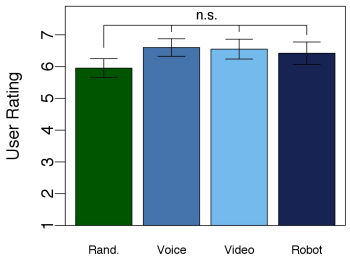
- **Novelty**: New stimulus may have increased attention, but may also have a distraction.
- **Authority:** Physical presence may imbue a robot with more credibility.
- **Peer pressure**: Compliance with a physical agent may be greater than an on-screen agent.

More investigation is needed. See paper for related work.

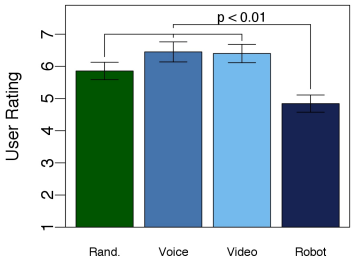
How relevent did the robot's hints seem to you?



How well did you understand the robot's hints?



How annoying/distracting did you find the robot to be?



Participants completed self-report Likert-scale questionnaires after the interaction. Participants in the three experimental conditions rated the hints as significantly more relevant to them than those in the random condition, but there was no significant difference between groups in the self-reported level of understanding of the hints. Participants in the robot group rated the robot as less "annoying/distracting" than participants in any other group.

This material is based upon work supported by grants from Microsoft Research and the National Science Foundation under contracts No. 0835767, No. 0968538, and No. 1117801.

For more information, including references and background, please see www.danleyzberg.com.