Ask or Tell: Balancing questions and instructions in intuitive teaching

Pamela J. Osborn Popp (pamop@nyu.edu)
Center for Neural Science, 4 Washington Pl
New York, NY 10003 USA

Todd M. Gureckis (todd.gureckis@nyu.edu)
Department of Psychology, 6 Washington Pl
New York, NY 10003 USA

Abstract

Teaching is an intuitive social activity that requires reasoning about and influencing the mind of others. A good teacher forms a belief about the knowledge of their student, asks clarifying questions, and gives instructions or explanations to try to induce a target concept in the student's mind. We propose Partially Observable Markov Decision Processes (POMDPs) as a model of intuitive human teaching. According to this account, teachers make pedagogical decisions with uncertainty about the knowledge state of their student. In two behavioral experiments, human participants were tasked with balancing assessments (asking questions) and instructions to help teach a student to build a tower of colored blocks. Human behavior in the task was compared to the performance of a computerized teaching algorithm optimized to solve the equivalent POMDP. Our results show that humans favor asking questions and establishing common ground during teaching even at an economic cost and increase question asking as uncertainty grows.

Keywords: teaching; machine teaching; POMDPs; question asking; instruction; education

Introduction

The simplest view of teaching is that the teacher knows something, and tells it to the pupils. If I know the way to the station, and you do not, I can tell you. What can be plainer? Teaching, then, whatever else it may be, includes the communication of information. But let us look at the matter a little more closely. I utter words. You hear them. All that you thus obtain is a number of sounds. I cannot transfer my knowledge from my mind to yours. Your mind is a closed book to me; I can never get into direct contact with it. - Dumville (1915)

Traditionally, cognitive science research focuses on learning without accounting for the teacher present in many learning scenarios. However, teaching is an incredibly rich and important behavior which forms the basis of much of our cultural knowledge. While there is considerable research in education on formal instruction (Lepper, Aspinwall, Mumme, & Chabay, 1990), we focus here on "intuitive teaching" – the types of everyday pedagogy we engage in when teaching a child how to tie their shoes, a colleague how to open the file cabinet, or a friend how to use an app on their phone.

Teaching is an inherently social behavior that involves the transfer of information from a knowledgeable individual to a naive one. In order to teach someone something, a good teacher must rely on an internal model of how the learner learns. For example, young children improve their teaching skills around the same time they develop theory of mind (Davis-Unger & Carlson, 2008; Bridgers, Jara-Ettinger, & Gweon, 2020). However, as Dumville (1915) points out, this is only part of the problem. We never know exactly what a student is thinking and thus teaching is fundamentally an act of decision making under uncertainty. We aim to influence the minds of our students without knowing exactly what they know or what they are thinking.

The goal of a teacher is to embed information in the pedagogical data they provide the student, such that by interacting with the data, the student acquires the desired information. Recently introduced Bayesian models of teaching and demonstration follow this framework, characterizing how the teacher selects helpful material based on their theory of the student's mind and how the student interprets that material based on their theory of the teacher's intentions (Shafto, Goodman, & Griffiths, 2014; Ho, Littman, MacGlashan, Cushman, & Austerweil, 2016). These models can be applied to describe active learning as a form of self-teaching (Yang, Vong, Yu, & Shafto, 2019; Coenen, Rehder, & Gureckis, 2015; Markant & Gureckis, 2014). Additionally, utilizing models of teaching to improve intelligent tutoring system technology is of increasing interest. In "machine teaching," a play on the more well known concept of machine learning, the goal is to design a data set that will convey or induce a particular model in the student (Zhu, Singla, Zilles, & Rafferty, 2018). The present paper builds upon this work to understand how teachers balance learning about the minds of their students while giving instructions to move those minds closer to a target concept.

Intuitive teaching as "debugging" the mind of someone else under uncertainty

It has long been recognized in formal analyses of teaching that students often have misconceptions or incorrect models of a domain. For example, Benson, Wittrock, and Baur (1993) explored students' preconceptions of the nature of gases and showed that prior to a chemistry course, students came with a broad array of (often subtly incorrect) ideas about how an ideal gas would behave in a chamber under various manipulations. Through formative assessment, the role of a teacher is to garner and confront the student's current understanding to repair misconceptions and build on existing knowledge (Bransford, Brown, Cocking, et al., 2000; Kane, 2006).

Starting from the premise that we never know exactly what a student is initially thinking, teaching naturally can be framed as a special class of well known decision problems known as Partially Observable Markov Decision Processes (or POMDPs). In fact, POMDPs have recently been applied to the problem of teaching for the purpose of developing intelligent tutoring systems, demonstrating their ability to capture critical features of teaching (Rafferty, Brunskill, Griffiths, & Shafto, 2016; Brunskill & Russell, 2011). However, the POMDP framework has not yet been used to model how *humans* decide to teach one another. The goal of the present paper is to evaluate POMDPs as a framework for thinking about intuitive human teaching.

In particular, we present the results of two experiments which asked human participants to play the role of an informed teacher instructing a less knowledgeable student. The task mimics the formal teaching setting of a teacher identifying and addressing a student misconception as in a one-on-one tutoring interaction (Vanlehn, 2006). In the task, the student (a computer agent) needs to be guided by the teacher to configure a set of blocks in a particular arrangement[1]. Human subjects (i.e., the teachers) can ask questions and give instructions to guide the students to a target building-block configuration which they cannot directly observe. This is similar to the collaborative block game in Wang, Liang, and Manning (2016), where humans are isolated to providing instructions and only the computer can control the movement of blocks. Critically, in our version of the task the teacher is never quite certain of the arrangement of the student's blocks but can either give instructions or ask questions to clarify that student's current "state." The goal of the teacher is to efficiently guide the student to the appropriate task state and then end the task for a monetary reward.

We had two hypotheses about human teaching that were informed in part by pilot analyses of the task. First, we hypothesized that humans have a preference for first establishing common ground with their partner before providing instructions. As a result our human teacher would, in certain circumstances, perform sub-optimally at the task by asking too many questions compared to the optimal analysis (Experiment 1). The second hypothesis was that participants would alter their question asking and instruction behavior systematically in response to changes in the uncertainty in the task. Thus it is not just that they first establish common ground with their partner, but that they selectively do this when there is uncertainty about the student's mental state (Experiment 2).

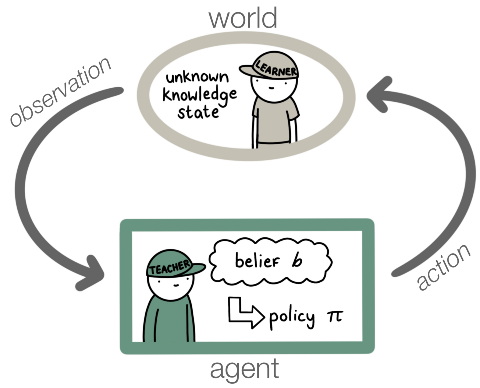


Figure 1: In a teaching POMDP, the teacher takes pedagogical actions that influence the knowledge state of the student.

Modeling Approach

Our model of teaching takes inspiration from computational models of reinforcement learning (RL) (Sutton, Barto, et al., 1998). Traditional RL research analyzes learning in reward-based decision making problems, using Markov Decision processes (MDPs) to characterize the *states*, *actions*, and *rewards* in an environment. An agent in an MDP takes actions to move between world states and earn rewards, like a slot-machine player pulling levers in an attempt to reach the jackpot "all-7s" state. POMDPs are a generalization of MDPs where the agent does not know the current state of the world with certainty. Instead, an agent solving a POMDP forms a *belief* about what the state may be, and uses *observations* of their environment to improve their belief and act accordingly.

POMDPs capture a wide variety of decision-making problems facing human and, for our purposes, provide a compelling computational account of the many features that make teaching interactions so challenging. Specifically, teachers act under uncertainty about various aspects of their students including the student's background knowledge, how the student responds to different types of instruction, and how a student's behaviors reflect their underlying knowledge. Modeling teaching as a POMDP thus enables us to formulate concrete predictions about how different forms of uncertainty affect teacher behavior. This approach is similar to that of (Ho, Littman, Cushman, & Austerweil, 2018), who model teaching as an MDP with deterministic meta-belief changes. The formulation of the teaching problem as a POMDP is an intuitive expansion with more flexible handling of uncertainty and the teacher's belief state.

POMDP Formalism Partially Observable Markov Decision Processes (POMDPs) describe a class of planning problems where an agent makes decisions under uncertainty (Kaelbling, Littman, & Cassandra, 1998). A POMDP is represented as a tuple $\langle S, A, T, R, \Omega, O \rangle$, where $s \in S$ are all possible states of the environment, $a \in A$ are all possible actions, $T(s, a, s')$ describes the transition probabilities between states

[1] While the block building task might appear trivial, using a physically embodied task means that the knowledge states of the student change in predictable ways following an instruction (i.e., if the teacher says "Move a block" the resulting effect on the student is clear). Additionally, the task is incredibly intuitive and easy to explain in a short session. Ultimately the task simply acts as a temporary stand-in for a more elaborated model of student learning in a particular domain.

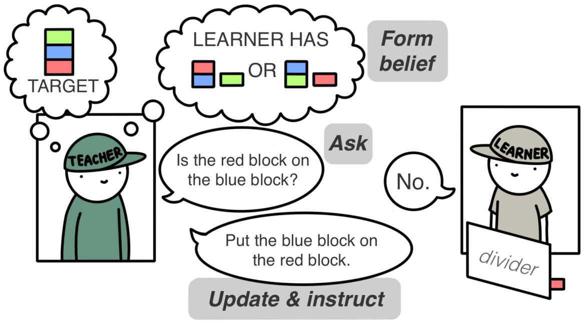


Figure 2: In the block-building task, the subject interacts with the computer to instruct it to build the tower. In our experiments, the subjects wear the "teacher" hat and the computer wears the "learner" hat.

s and s' after taking action a, and $R(s,a)$ describes the reward function for taking action a in state s. The former terms define a Markov Decision process, but a POMDP also includes a set of observations $o \in \Omega$. Here, Ω is the space of all possible observations, and the observation function $O(s',a,o)$ defines the probability of the agent receiving observation o after taking action a and ending up in state s'.

In a teaching POMDP, transitions between states S reflect the changing knowledge state of the learner, actions A are pedagogical choices by the teacher (such as exercises, opportunities to study particular material, or the presentation of facts), and observations Ω are the data a teacher receives about their student, such as test scores or classroom attitudes (see Figure 1 for an illustration). The reward structure of the environment might encourage the teacher to attempt to improve test scores, or maximize long-term retention of material, or minimize teaching time required to convey a concept.

Partially Observable Monte Carlo Planning. POMDPs define the problem of decision making under uncertainty but there are many possible methods for solving these problems. For example, Monte Carlo Tree Search, or MCTS (Browne et al., 2012), is a popular solution method for standard MDPs based on estimating the value of actions by simulation. Expanding this algorithm to partially observable domains yields a solution technique called Partially Observable Monte Carlo Planning (POMCP) (Silver & Veness, 2010). POMCP is an on-line, best-first tree search algorithm for POMDPs. Whereas in MCTS, the agent builds a tree of simulated future states given the actions it might take, in POMCP, the agent is uncertain about the true state of the world. As a result, a POMCP agent builds a tree indexed by the possible histories of actions and observations it may encounter, which contribute to the agent's beliefs about the latent world state.

The POMCP model requires an exploration hyperparameter c and a search-depth hyperparameter γ. During simulation, actions are selected by the Upper Confidence Bound algorithm (UCB), which augments tree node values to provide advantage to under-sampled actions with larger values of c.

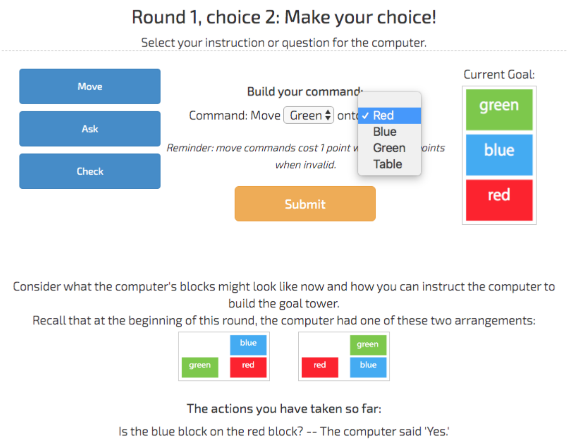
One interesting feature of POMCP is that it plans and learns simultaneously. The belief of the agent is represented by a particle filter that trickles down through the tree during simulations. As a result, POMCP tree search builds a tree of not only valuations of prospective actions, but also of the belief state it should acquire given certain observations. When the POMCP agent selects an action and receives a real-world observation, provided enough simulations have been performed, the POMCP arrives at a tree node which already contains the appropriate new belief state. While POMCP simulates forward to choose a valuable action, it simultaneously builds a tree of future beliefs.

Student policy. In the experiment, subjects play the role of the teacher and attempt to convey a target block tower to a student, as illustrated in Figure 2. The computer plays the role of the artificial student and acts following a defined set of rules. The computer-student's policy is defined by the fact that the block-building takes place in a world with normal physics and the restriction that only one block can move at a time. However, when we manipulate the transition or observation functions of the teaching POMDP, we introduce separate types of students with which the subjects can interact. These manipulations lead to emergent behavioral changes in the computer-student that approximate features of real world students such as struggling with acquiring and storing new information (transition noise), or weakness in test-taking and reporting one's own knowledge (observation noise). In the current paper there are three different student policies, outlined below.

Computer-Student A: Partner A is the most faithful and trustworthy of the three computer student agents. There is no noise in either the transition or observation functions, meaning that Partner A always answers questions with the truth and always follows block-moving instructions precisely.

Computer-Student B: Partner B is worse at following instructions than Partner A. When the teacher gives a block-moving instruction, Partner B will execute the movement only 80% of the time. The other 20% of the time, Partner B will refuse to move any blocks and the current block arrangement will stay the same. This behavior is implemented in the learner model as a transition probability of 80% for all Legal Move actions. Partner B is a reliable reporter, however, and always answers questions with the truth.

Computer-Student C: Partner C is an unreliable reporter, akin to a student whose test scores do not line up with their true knowledge. In response to a question, Partner C sometimes answers with the incorrect (untruthful) answer. Specifically, Partner C responds to questions with the truth 80% of the time, and responds with the opposite of the truth 20% of the time. This behavior is implemented in the observation function of the learner model with an 80% probability of making a correct observation on a question. Like Partner A, Partner C always follows block-moving instructions precisely as given.



Round 1, choice 2: Make your choice!

Select your instruction or question for the computer.

Build your command:

Command: Move Green onto ✓ Red

Reminder: move commands cost 1 point w... ...oints when invalid.

Consider what the computer's blocks might look like now and how you can instruct the computer to build the goal tower.

Recall that at the beginning of this round, the computer had one of these two arrangements:

The actions you have taken so far:

Is the blue block on the red block? -- The computer said 'Yes.'

Figure 3: Screen shot from the task. Subjects choose between "Move," "Ask," and "Check" actions. Drop-down menus allow specification of block colors. The goal tower of the round is shown at right. The possible initial block arrangements are displayed as a reminder at the bottom, as well as a list of the actions the subject has taken so far and any answers from the computer-student.

Experimental Methods

Task design. Participants were recruited on Amazon Mechanical Turk to perform the task. Following a series of instructions and a comprehension check, subjects played ten rounds of the block building game.

In the task, participants are instructed to teach a computer how to build a particular block tower. Figure 3 presents the computer display during the task. Participants are told that in front of the automated computer player are three blocks: one red, one blue, and one green. The participant is naive to the computer's arrangement of blocks. At the beginning of each round, the game displays two possible block arrangements to the participant, one of which is the true arrangement of blocks in front of the computer. Both arrangements are equally likely to be the true initial arrangement. The screen also displays a goal block tower (that is, an ordered stack of the three colored blocks) that the participant aims to help the computer build with the limitation that only one block can be moved at a time. The computer's blocks always begin in a non-tower arrangement.

The participant chooses from a series of drop down menus to specify the action they want to take. There are three classes of actions: Ask, Move, and Check. Ask actions are in the form of a question, requesting information from the computer in the format "Is the [color a] block on the [color b] block?" or "Is the [color a] block on the table?" Move actions give the computer a command of which block to move in the format "Move the [color a] block onto the [color b] block," or "Move the [color a] block onto the table." The final type of action is the Check action, which ends the round, tests whether the block arrangement is correct, and brings up a screen displaying the subject's point score for the round. Subjects have a chance to review and edit the action they have selected before pressing the "submit" button, and there is no time limit.

Costs and penalties associated with each action incentivized participants to be judicious about action selection to maximize their earnings as follows: Ask actions and Legal Move actions cost one point. Illegal Move actions, such as attempting to move a block that is obstructed by another block on top of it, cost two points and result in no block movement. This scoring was selected to convey the innate temporal and opportunity costs of teaching actions, with an additional cost for giving the student an instruction which they cannot physically follow. If the participant guides the computer to build the correct tower, they earn ten points, but if the completed tower is incorrect, the participant loses ten points. Subjects were informed that their bonus payment would be calculated at the end based on the score earned in one randomly selected game round. Specific payment policies are described by experiment below.

Experiment 1. The experimental design and analyses for Experiment 1 were preregistered online[2]. We gathered data until we reached N=50, not including 6 subjects who met the exclusion criteria of earning zero or fewer points on more than half of the game rounds. The participants' mean age was 36.6 years with a range of 20 to 69 years old. The task took approximately 15 minutes ($M = 13.7$ minutes, $SD = 4.6$) and subjects were paid between $2.00 and $2.50 for their time depending on bonus.

The student policy in Experiment 1 did not incorporate any transition or observation noise (Computer-student A). To form our stimuli and hypotheses, we ran the POMCP algorithm on many problems (e.g., sets of priors and target states) to find specific stimuli that yielded diverse results. Because we wanted to analyze how subjects balance assessment and instruction, we selected 5 contexts where the model predicts that the optimal first action should be a question ("Ask" type action) and five contexts where the model predicts that the optimal first action is an instruction ("Move" type action). We were thus particularly interested in the action type of the first action subjects selected within a round. Based on pilot data, we predicted that subjects would choose a question action as their first action in more than five out of the ten contexts; that is, they would ask questions more frequently than the optimal POMCP model.

The results of the simulations show the averaged behavior of the POMCP agent initialized with 10 random seeds. The tree search depth was set to a maximum of 20 steps and the UCB exploration parameter was set to $c = 0.5$. The POMCP agent performed 10^6 simulations per action. Additionally, the algorithm was designed with preferred actions to improve performance (reduce computation time). Preferred actions primed the agent to choose an instruction that would decrease the distance to the goal state. Additionally, when the

[2] Preregistration at https://aspredicted.org/blind.php?x=yc6pe3.

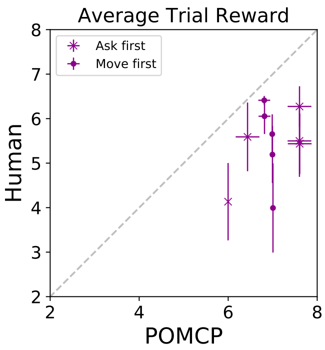


Figure 4: Human and POMCP performance (with standard error bars) compared for the ten Experiment 1 contexts. The x markers denote the five "Ask" first contexts, and the dot markers denote the five contexts where POMCP "Moves" first.

agent believed it was in the goal state, it was encouraged to choose the "Check" action to end the round.

Experiment 2. Subjects were recruited via Amazon mTurk as in Experiment 1. The expected points earned per round was lower in Experiment 2 because of the increased difficulty (noise), and so the alignment between point score and bonus was adjusted. The time duration of Experiment 2 was also slightly longer ($M = 13.9$ minutes, $SD = 8.1$) than in Experiment 1, and subjects were paid between \$2.50 and \$3.00 for their time depending on bonus. Subjects would have been excluded if they scored zero points or fewer on all rounds, but no subjects met this criteria. Experiment 2 had two between-subject conditions, both of which involved manipulating the teaching POMDP subjects were tasked with solving. In condition 1, subjects (N=29; mean age 36.7, range 23-55) taught Computer-student B (noise in the POMDP transition function). In condition 2, subjects (N=24; mean age 38.7, range 23-66) taught Computer-student C (noise in the POMDP observation function). To better convey the probabilistic dynamics of Computer-students B and C, subjects had an opportunity to sample the stochasticity with a practice button during the instructions that demonstrated an 80% probability of correct instruction-following or question-answering.

In Experiment 2, our goal was to examine how manipulation of specific features of a teaching POMDP affect human behavior. We predicted that when handling transition noise (Student B), subjects would increase the amount of questions they asked throughout a round. We also expected that when handling observation noise (Student C), subjects would limit question asking to the beginning of the round, perhaps asking multiple questions before executing only "Move" actions for the rest of the round. Five of the ten contexts were the same as the POMCP-predicted "ask-first" contexts of Experiment 1, referred to as "Unknown Initial State" trials. The other five contexts were "Known Initial State" trials that displayed the exact beginning block arrangement of the student, removing the uncertainty over initial state compared to the rounds displaying two possible beginning arrangements. Without initial state uncertainty, asking a question first is unnecessary; however, in the condition with transition noise, uncertainty could still grow throughout the round.

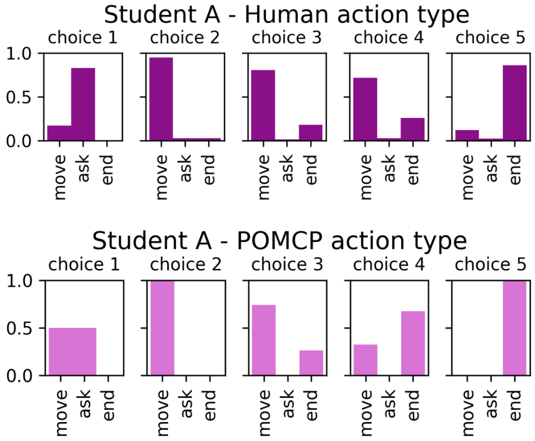


Figure 5: Experiment 1 sequences of action types across all rounds. Top, subject data; bottom, POMCP teaching agent simulation results.

Results

Experiment 1 was designed so that the optimal POMCP teaching policy endorsed an "ask-first" policy for half of the trials, and a "move-first" policy for the others. This element of the design is visualized in the bottom row of Figure 5, which depicts the proportion of action types the POMCP algorithm chose consecutively within rounds. In all of the game rounds, the teacher was presented with two possible initial block arrangements, and the teacher was aware that the student followed a completely deterministic policy (Student A). In half of the rounds, POMCP chooses an "Ask" action first, while it chooses a "Move" action first in the other rounds. Examining human action types in the same way (Figure 5), we notice that humans choose an "Ask" action as their first choice much more than 50% of the time. We performed a one-tailed, one-sample t-test to compare human question asking preference to POMCP behavior. As hypothesized in our preregistration, we found that subjects asked questions on the first choice of the trial ($M = 82.6\%$ of trials, $SD = 19.7\%$) significantly more than the optimal POMCP model ($t(49) = 11.6$, $p < .001$).

As a result, humans scored fewer points on average than the POMCP model. Figure 4 shows the average number of points earned across humans and across runs of the POMCP model for the ten Experiment 1 trials. All of the points lie to the right of the unity line, meaning that POMCP received on average a greater number of points on every trial. In Figure 4 the marker style indicates whether it was optimal to "Ask" or "Move" first in that context. Because of their preference to ask questions at the beginning of trials where a question is unnecessary, subjects sacrificed points and thereby monetary

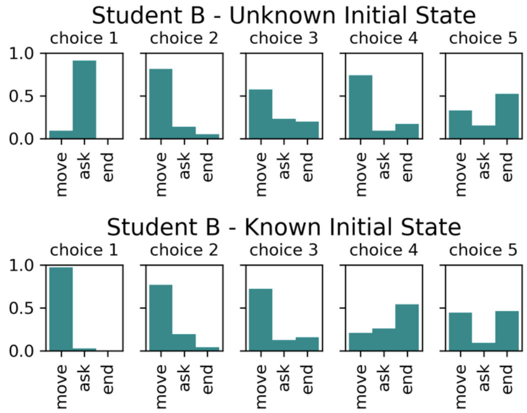


Figure 6: Experiment 2, Student B condition. Histograms of subject action type frequency by choice within game round. Top, Unknown Initial State contexts; bottom, Known Initial State contexts.

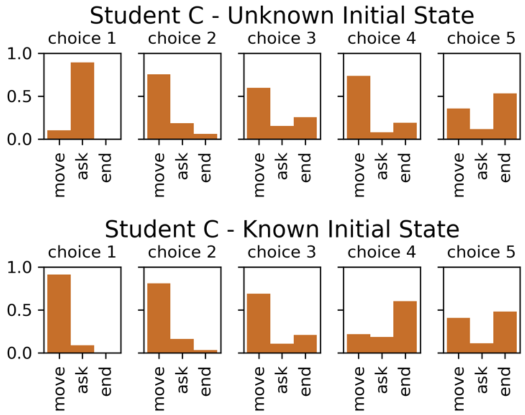


Figure 7: Experiment 2, Student C condition. See Figure 6 caption for details.

reward in the task.

Experiment 2 introduced stochasticity into the behavior of the student. Subjects assigned to condition 1 interacted with Student B, who unreliably followed instructions. Subjects assigned to condition 2 interacted with Student C, who unreliably answered questions. Subjects in Experiment 2 found the task more difficult; in a post-task questionnaire, Experiment 2 subjects ($M = 5.0$, $SD = 3.2$) reported a significantly higher difficulty rating on a scale of 1-10 than did Experiment 1 subjects ($M = 3.1$, $SD = 2.2$) according to an independent samples t-test ($t(100) = -3.6$, $p < .001$).

As we predicted, introducing transition noise increases the amount of questions subjects took throughout each round. When teaching Student B, subjects still frequently asked questions throughout the round even in Known Initial State contexts (Figure 6). Alternatively, observation noise led to more front-loading of questions, as seen in Figure 7. In particular, subjects teaching Student C during Unknown Initial State trials were likely to ask a question on the second action in addition to the first action, indicating they were seeking reassurance that the student had answered correctly. Interestingly, subjects teaching Student C still asked questions in Known Initial State trials, even though no questions were required to assess the current state and the Move actions would be completely deterministic.

Discussion

This study examined how people intuitively teach another agent. We designed an interactive block building task which allowed us to assess how a teacher tracked the mental state of a student to reach a goal. The task required teachers to balance asking questions (to establish common ground) and providing instruction. The optimal sequence and balance of these actions was determined by considering the optimal solution to an equivalent POMDP.

Our results show that in Experiment 1, teachers overutilized questions even when there was no economic benefit in the task from doing so. We saw the same trend in the Experiment 2 Student C condition on Known Initial State trials, where question-asking is unnecessary to the task goal. While more work is needed to understand the nature of this suboptimality, we suggest that it stems from a bias that human teachers have to ask questions to establish what their partner knows and then provide corrective instruction. It is quite unnatural to begin giving instructions to another agent when you are unclear about their goals or mental model. It is possible this bias stems from a more general tendency of humans to avoid uncertainty (Epstein, 1999; Bradac, 2001; Halevy & Feltkamp, 2005). However, humans may also have an intuition that question asking is a necessary part of a teaching interaction. Future work should analyze question-asking preferences between social tasks such as teaching and non-social tasks.

That said, in Experiment 2 we showed that people do ask questions selectively – they ask questions predominantly when there is uncertainty about their partner, and they ask more questions as the task proceeds in cases where a poor performing student has increasing chance of entering the wrong state (i.e., becoming confused). Thus, in a broader sense non-expert teachers seem to intuitively recognize the value of asking questions when teaching.

Overall this is a first attempt to model human teaching as a planning problem that can be articulated by the POMDP framework. One aspect we neglected in the current study is more realistic and reactive behavior on the part of the student. For example, Shafto et al. (2014) explore not only how a teacher adapts to the student but also how the student adapts to the instructions of the teacher. In our task this was less of a concern because the instructions and task were otherwise unambiguous. In addition, our preliminary modeling results with POMCP need to be extended to consider more psychologically realistic mechanisms for how an agent would approximate the POMDP solution.

Acknowledgements

We thank the James S. McDonnell Foundation (Scholar Award to T.M.G.) for financial support.

References

Benson, D. L., Wittrock, M. C., & Baur, M. E. (1993). Students' preconceptions of the nature of gases. *Journal of research in science teaching*, *30*(6), 587–597.

Bradac, J. J. (2001). Theory comparison: Uncertainty reduction, problematic integration, uncertainty management, and other curious constructs. *Journal of Communication*, *51*(3), 456–476.

Bransford, J. D., Brown, A. L., Cocking, R. R., et al. (2000). *How people learn* (Vol. 11). Washington, DC: National academy press.

Bridgers, S., Jara-Ettinger, J., & Gweon, H. (2020). Young children consider the expected utility of others' learning to decide what to teach. *Nature Human Behaviour*, *4*(2), 144–152.

Browne, C. B., Powley, E., Whitehouse, D., Lucas, S. M., Cowling, P. I., Rohlfshagen, P., ... Colton, S. (2012). A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, *4*(1), 1–43.

Brunskill, E., & Russell, S. (2011). Partially observable sequential decision making for problem selection in an intelligent tutoring system.

Coenen, A., Rehder, B., & Gureckis, T. M. (2015). Strategies to intervene on causal systems are adaptively selected. *Cognitive psychology*, *79*, 102–133.

Davis-Unger, A. C., & Carlson, S. M. (2008). Development of teaching skills and relations to theory of mind in preschoolers. *Journal of Cognition and Development*, *9*(1), 26–45.

Dumville, B. (1915). *Teaching: Its nature and varieties*. University Tutorial Press.

Epstein, L. G. (1999). A definition of uncertainty aversion. *The Review of Economic Studies*, *66*(3), 579–608.

Halevy, Y., & Feltkamp, V. (2005). A bayesian approach to uncertainty aversion. *The Review of Economic Studies*, *72*(2), 449–466.

Ho, M. K., Littman, M., MacGlashan, J., Cushman, F., & Austerweil, J. L. (2016). Showing versus doing: Teaching by demonstration. In *Advances in neural information processing systems* (pp. 3027–3035).

Ho, M. K., Littman, M. L., Cushman, F., & Austerweil, J. L. (2018). Effectively learning from pedagogical demonstrations. In *Proceedings of the annual conference of the cognitive science society.*

Kaelbling, L. P., Littman, M. L., & Cassandra, A. R. (1998). Planning and acting in partially observable stochastic domains. *Artificial intelligence*, *101*(1-2), 99–134.

Kane, M. T. (2006). Validation. In R. L. Brennan (Ed.), *Educational measurement* (4th ed., pp. 17–64). Westport, CT: Praeger Publishers.

Lepper, M., Aspinwall, L., Mumme, D., & Chabay, R. (1990). W.(1990). self-perception and social-perception processes in tutoring: Subtle social control strategies of expert tutors. In *Self-inference processes: The ontario symposium* (pp. 217–237).

Markant, D. B., & Gureckis, T. M. (2014). Is it better to select or to receive? learning via active and passive hypothesis testing. *Journal of Experimental Psychology: General*, *143*(1), 94.

Rafferty, A. N., Brunskill, E., Griffiths, T. L., & Shafto, P. (2016). Faster teaching via pomdp planning. *Cognitive science*, *40*(6), 1290–1332.

Shafto, P., Goodman, N. D., & Griffiths, T. L. (2014). A rational account of pedagogical reasoning: Teaching by, and learning from, examples. *Cognitive psychology*, *71*, 55–89.

Silver, D., & Veness, J. (2010). Monte-carlo planning in large pomdps. In *Advances in neural information processing systems* (pp. 2164–2172).

Sutton, R. S., Barto, A. G., et al. (1998). *Introduction to reinforcement learning* (Vol. 2) (No. 4). MIT press Cambridge.

Vanlehn, K. (2006). The behavior of tutoring systems. *International journal of artificial intelligence in education*, *16*(3), 227–265.

van Opheusden, B., Galbiati, G., Bnaya, Z., Li, Y., & Ma, W. J. (2017). A computational model for decision tree search. In *Cogsci.*

Wang, S. I., Liang, P., & Manning, C. D. (2016). Learning language games through interaction. *arXiv preprint arXiv:1606.02447*.

Yang, S. C.-H., Vong, W. K., Yu, Y., & Shafto, P. (2019). A unifying computational framework for teaching and active learning. *Topics in cognitive science*, *11*(2), 316–337.

Zhu, X., Singla, A., Zilles, S., & Rafferty, A. N. (2018). An overview of machine teaching. *CoRR*, *abs/1801.05927*.