Help Me Help You: Bridging the Gaps in Human-Agent Collaboration — Papers from the AAAI 2011 Spring Symposium (SS-11-05)

Human Natural Instruction of a Simulated Electronic Student

Tasneem Kaochar, Raquel Torres Peralta, Clayton T. Morrison, Thomas J. Walsh, Ian R. Fasel, Sumin Beyon, Anh Tran, Jeremy Wright and **Paul R. Cohen**
The University of Arizona, Department of Computer Science, Tucson, AZ 85721-0077
{tkaochar,clayton,twalsh,ianfasel}@cs.arizona.edu

Abstract

Humans naturally use multiple modes of instruction while teaching one another. We would like our robots and artificial agents to be instructed in the same way, rather than programmed. In this paper, we review prior work on human instruction of autonomous agents and present observations from two exploratory pilot studies and the results of a full study investigating how multiple instruction modes are used by humans. We describe our Bootstrapped Learning User Interface, a prototype multi-instruction interface informed by our human-user studies.

Introduction

Humans are remarkably facile at imparting knowledge to one another. Through the lens of the various kinds of state of the art machine learning algorithms we can identify multiple modes of natural human instruction: we *define* concepts, we *describe* and provide *demonstrations* of procedures, we give *examples* of rules and conditions, and we provide various kinds of *feedback* to student behavior. These methods are used depending on the kind of concept that is being taught (concept definitions, conditions, rules, procedures) and the conditions under which the teacher and student find themselves. Just as important, the student is an equal participant in the lesson, able to learn and recognize conventions, and actively observes and constructs a model of the situation the teacher is presenting in the lesson.

The familiar and readily available systems of instruction that are used in human-to-human teaching stand in sharp contrast with how we currently get computers to do what we want, despite the fact that computers appear to share with humans a kind of universal flexibility: we have been able to make computers perform an enormous variety of complex tasks. These capabilities are currently only achieved as the result of often costly and labor intensive programming by human engineers—we get computers to do what we want through a process that is closer to brain surgery than *instruction*. This is true even for state of the art machine learning, where algorithms are capable of extracting patterns, classifying noisy instances, and learning complex procedures. But human engineers must provide data in just the right form, with the correct training data, and even then must often explore through trial and error to get the agent to learn what is desired.

The goal of *human-instructable computing* is to build an "electronic student" that can be taught using the same natural instruction methods humans (specifically non-programmers) use. An electronic student architecture will naturally include a variety of state of the art machine learning algorithms, but the key challenge is to provide the interface between them and the natural instruction methods used by humans. There is now a growing body of literature from researchers studying the intersection between human-computer interfaces and machine learning. However, to date, the focus of this work has been on particular individual modes for human instruction rather than the full gamut. The next step is to understand how to provide several modes of instruction in the interface, and to better understand how humans might use such an interface to teach.

In this paper, we describe three investigations intended to uncover how humans might naturally instruct a *capable* electronic student, using a broad spectrum of instruction types. We aim to design an interface that can accommodate the variety of natural instruction modes that humans appear to use, and understand when and how these modes are used.

Prior Work

Many prior efforts have described agents that interact with human teachers. However, most of these works used only one mode of teacher-student interaction (e.g. teaching by demonstration) over the agent's lifetime. We can roughly classify this existing work into three categories based on the kind of feedback the teacher can pass to the student: teaching by *demonstration*, teaching *concepts by example*, and teaching through *reinforcement*. We now describe these interactions in more detail and provide examples from the literature.

In teaching by demonstration, a teacher has the same control interface to a dynamical system as the student does, and is able to provide *traces* of proper behavior. The learning in this case does not need to be pure mimicry, and instead enables the acquisition of a higher-level policy from the traces, allowing the student to perform correctly in new situations. For instance, after seeing the teacher navigate around a crate, the agent may be able to navigate around similar obstacles in different locations. Prior work in learning from demonstration has appeared in several branches of the reinforcement learning literature, including using traces in a batch setting to bootstrap robot behaviors (Smart and Kaelbling 2002; Argall et al. 2009) and in extended apprenticeship learning over the agent's lifetime (Walsh et al. 2010). These works often espouse the use of autonomous learning in concert with the teacher-provided traces. In our case, we will be accompanying the demonstration traces by other forms of teacher interaction that will allow for the same fine grained tuning of behavior.

Another general form of teacher-student interaction is teaching by examples of concepts. This protocol shadows standard supervised-learning interaction in that the teacher is responsible for providing labeled examples of a concept to be learned. However, the teacher may also provide explanations or hints as to *why* a specific example was classified a certain way. An example of this framework is the WILL system (Natarajan et al. 2010) for inductive logic programming, which combines traditional concept learning machinery with users' indications of relevant and important definitional components. Other systems for more traditional classification problems, such as training support vector machines (Chernova and Veloso 2009) and email classification (Stumpf et al. 2009) have used this paradigm, with the latter focussing explicitly on natural ways that humans provide reasons for categorization decisions. Similar techniques can be helpful in learning conditional dynamics (e.g. you cannot walk through walls), as was the case in work on learning object categories and affordances (Thomaz and Cakmak 2009).

While learning concepts is often an essential component of a task, this framework does not allow for specific actions (other than labeling) to be designated as good or bad, and while demonstration provides an indirect channel for such guidance (by only showing good behavior), we now consider a third channel of teacher interaction for more fine-grained behavioral refinement. In teaching through reinforcement, the teacher is able to give a feedback signal indicating a degree of happiness (or unhappiness) with the agent's behavior, either at specific timesteps or when an episode has ended. This form of feedback has been shown to be moderately successful in complex tasks such as a simulated cooking scenario (Thomaz, Hoffman, and Breazeal 2006) and tactical battles in a real-time strategy game (Judah et al. 2010). However, recent work (Knox and Stone 2010; Thomaz and Breazeal 2008) has indicated that many "natural" ways of incorporating numeric or ordinal feedback from humans into a reinforcement learning problem can be perilous, as humans often provide incompatible feedback or do not follow "standard" definitions of reward and value. As such, this channel is usually best suited for fine-grained refinements that are not easily teachable through the protocols discussed above.

Methodology

In contrast to these works, the interface we are designing is built around allowing the human teacher to use instantiations of all of these instruction types during learning. This allows for both the teaching of high-level concepts and for fine-grained adjustments of undesired behavior based on direct reinforcement or demonstrations of specific situations.

However, the design of such an interface poses a methodological challenge: we do not yet understand how humans might naturally teach an automated agent using a multimodal instruction interface. The ideal situation would be to take an existing electronic student and see how humans instruct it. However, such a student does not yet exist. Furthermore, prior work (Perry 2008), in which transcripts were collected of the interaction between a human teacher and a human serving as the interface between the teacher and an early version of the Mable electronic student (Mailler et al. 2009), found that as soon as the teacher recognized that Mable is limited in the kind of interaction it can accommodate, the participant tended to reduce instruction to a style more like direct programming than natural instruction. To avoid this, in the following studies we employed a *Wizard of OZ* (WOZ) protocol in which the student is actually controlled by a human without the teacher's knowledge. This allowed us to provide human teachers a high degree of freedom in how they choose to instruct the student while believing they are interacting with a capable student. We begin by reviewing two pilot studies aimed at exploring different interface configurations for eliciting multiple instruction methods during a teaching episode.

Pilot Study 1 - Wubble World: Free Text Interaction

In our first exploratory pilot study, we wanted to elicit as close to fully natural instruction behavior as possible without constraining how the Teacher might teach, but also under the condition that the Teacher believed they were interacting with a capable automated agent. We used Wubble World (hereafter, *WW*) (Hewlett et al. 2007), a three dimensional simulated environment in which agents called *wubbles* can move around and manipulate objects. We adapted WW so that humans can control the wubbles and communicate with one another directly using a peer-to-peer chat facility. Figure 1 shows the the WW interface view.

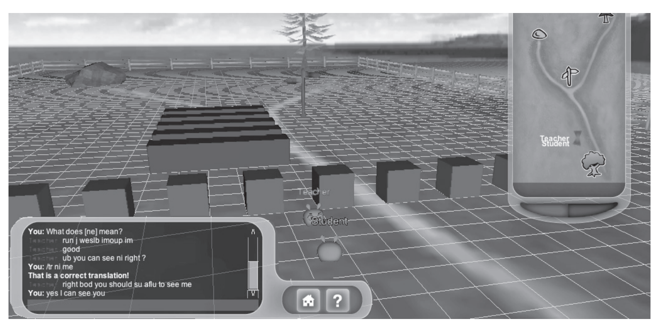

Figure 1: Student interface for Wubble World.

In each instruction session, one human participant was given the role of the *Teacher*, the other was the *Student*. Both Teacher and Student did not know ahead of time what the teaching/learning task was, and the Teacher was led to believe that they were interacting with a computer agent rather than a human Student. The two participants were placed in separate rooms. Both the Teacher and Student were trained to use the WW interface and given a short amount of practice. The interface included controls for moving the wubbles, highlighting objects, and picking up, carrying and putting down objects. The Teacher and Student were not constrained in what they could write to one another.

The teaching task that was presented to each participant required multiple concepts to be taught, some depending on first mastering others. Specifically, the WW environment has two kinds of blocks: cube-shaped red blocks and elongated blue blocks (see Fig. 1). The task was to have 5 smaller red boxes placed in a row at the base of a wall, and an elongated blue block placed on top. This required the Teacher to teach the difference between the blocks, how they were to be positioned, as well as the concept of line of sight; in this way, the teaching task involved concept definitions (line of sight, distinguishing the different block types), rules and conditions (the red blocks must form the base, what line of sight means), and a procedure (how to construct the wall). The Teacher was also asked to verify that the Student had learned each of the concepts taught.

We collected transcripts from six participant pairs.[1] The following enumerates a set of observations based on the transcripts:

1. Modes of instruction are tightly interleaved: while stepping through the procedure for building the wall, telling the Student what to do at each step, the Teacher also defined new concepts ("this is a red block") and rules surrounding their use ("put red boxes first"), all the while providing feedback ("that was good"). One reason for the interleaving is that the teaching environment allows for both teacher and student to engage in interaction together within the environment. In general, Teacher's did not explicitly denote the beginning or end of a procedure, instead relying on other contextual cues, such as asking questions, or asking the Student to try again.

2. Teachers sometimes demonstrated the action themselves, instructing the Student to watch. Other times they told the student what to do, step by step, with the assumption that the student understood this.

3. Teacher feedback ranged from "good" or "no" to more elaborated explanation: "No, the blocks need to be on the ground, not the top"

4. It was common for Teachers using free text to use multiple terms for the same object without explicitly noting the different terms. E.g., "block" versus "box" versus "cube".

5. Students asked questions of clarification, and Teachers gave definitions and reasons for actions or for labeling conditions. This helped both Teacher and Student establish a shared frame of reference.
   - T: "the blocks should be placed close to each other"
   - S: "all the red blocks?"
   - T: "yes"
   - T: "The line of sight is blocked because the blocks are between you and the tree."

6. In some cases, Teachers provided background summaries of what they were going to teach before providing demonstrations or examples, e.g., "Let's imagine that the sign post is an observer to hide from."

Pilot Study 2 - Charlie the Pirate: Constrained Text Interaction

The lessons from WW achieved the goal of providing insight into how to get the Teacher to be expressive, but free text entry from both Student and Teacher sometimes led to quite complex linguistic constructions. Also, we anticipate that in most cases interaction with an electronic student will either be with a physical robot, or with a system in which the Teacher will not be represented in a simulated world as an avatar along with the Student. For this reason, we changed the Teaching domain and interaction protocol. In this pilot, we had Teachers interact with a small physical robot, and limited what the Student could say, in the hopes of finding a balance between expressivity for the Teacher but closer to interactions that might be handled by current artificial agents.

The robot, named Charlie, was built from a Bioloids robotics kit[2] and consists of a 4-wheeled, independent drive chassis, an arm with a simple two-fingered (pincer) gripper, and an arm with an RFID sensor attached at the end (see Fig. 2, right). While each arm has multiple degrees of freedom, the arm controllers were simplified to discrete actions to deploy and retract, open and close, rotate the gripper, and scan with the RFID sensor (this would sweep the sensor back and forth after it was deployed).

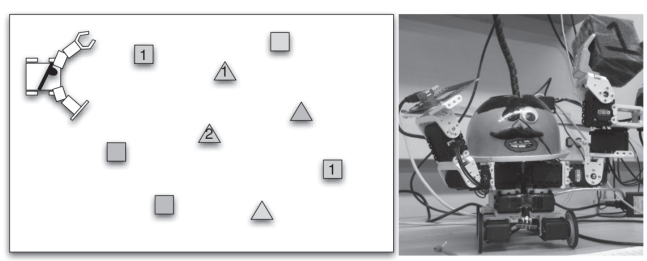

Figure 2: Left: A sample arrangement of blocks of different shapes and colors used in the robot experiment; numbered blocks contain RFID tags, where the number indicates the value of a scan. Right: A picture of Charlie the Robot.

[1] Additional details and transcripts are available here (Morrison, Fasel, and Cohen 2010). Our volunteer participants did not have background knowledge about the state of the art in machine learning or artificial agents, so they did not have advanced knowledge about whether an electronic student is currently possible—this made it possible to maintain the illusion that the Teacher was interacting with an intelligent computer agent.

[2] http://www.trossenrobotics.com/bioloid-robot-kits.aspx

The robot was placed on a table along with small foam blocks of various shapes and colors (see Fig. 2, left). The Teacher stood at one end of the table and used a terminal to enter text to communicate directly with the Student. The Student was located in a partitioned region of the lab; the Teacher could not see the Student and did not know there was an additional person in the room, however the Student could view the table workspace and the Teacher via two web cams. The Student's workstation included a keyboard, and a monitor displaying two windows with the web cam video feed, a peer-to-peer chat terminal, and another terminal indicating the robot's RFID sensor status. Rather than allowing the Student to type free text, we employed two conditions: one in which the Student was not able to respond, and a second condition in which the student can respond by selecting from a set of canned responses: e.g., "Hello!", "I didn't understand", "Yes", "No", and "I'm finished".

The teaching task asked the Teacher to instruct Charlie to "find treasure" amongst the blocks on the table. The rules were that treasure could only be in one of the purple objects on the table. Charlie could see the color and shape of the blocks. For the Teacher's benefit, all of the purple blocks had numbers indicating whether they had treasure or not: 2 indicated a block with treasure. Charlie could not see the numbers on the blocks and instead had to use the RFID scanner to scan the blocks. Charlie's RFID scanner would display a "1" on the Student's sensor if the block was labeled 1, or "2" for blocks with treasure, otherwise it would display nothing. Once Charlie found the block with the treasure, the Student had to direct him to turn the block upside down in order to "bury" it. This task required that Charlie scan all purple blocks until finding the treasure.

We collected transcripts of 5 Teacher/Student pairs, 2 in the "no Student response" condition, and 3 where the Student could use the canned phrases. The participants were again selected from graduate students and staff of the UA Computer Science Department, but all participants had little experience with robots or autonomous agents. The main findings from analysis of the transcripts were the following:

1. Similar to the WW findings, all of the different teaching modes were observed and they were tightly interleaved.
2. In the condition where the Student was not able to respond to the Teacher, we saw the same kind of Teacher expressivity observed in the prior study with Mable (Perry 2008): Teachers reverted to a style that was more like programming than instruction, simply walking Charlie through the task and only occasionally indicating conditions. In the canned response condition, however, while the Teacher's expressions tended to be simpler than in the WW free-text condition, the Teacher tended to ask more questions or conduct small tests of the Student's state of knowledge.
3. Under both conditions, there was a pattern in the complexity of instructions provided by the Teacher: When Charlie followed instructions with no mistakes, the complexity of the instructions increased, whereas the complexity decreased when Charlie failed to follow directions correctly.

Main Study - BLUI: Bootstrapped Learning User Interface

From the prior two pilots we have learned that indeed multiple instruction modes are used when expressable, that they are interleaved during instruction, and that allowing the Student to respond to the Teacher with at least a limited set of response types has a positive affect on Teacher expressiveness. Given this experience, we constructed a prototype interface, the *Bootstrapped Learning User Interface* (BLUI), to make these instruction types available through a GUI interface. We then conducted a WOZ experiment to see how human users with little prior experience might use BLUI.

The UAV ISR Domain

BLUI has been designed to work in an Intelligence, Surveillance and Reconnaissance (ISR) domain in which the Student is the control system of a simulated UAV that will be taught to carry out ISR missions. We use the X-Plane customizable flight simulator environment[3] to simulate a realistic operating environment.

In BLUI, the UAV that the wizard/Student controls is a small airplane with a flight control system that can keep the plane flying in the direction and at the altitude specified by the teacher. (Note that the human teacher is not required to understand flight dynamics; this is the Student's responsibility). A set of flight waypoints can also be specified to indicate the flight trajectory the UAV should follow.

A scenario file can be loaded into the X-Plane environment at any point during the instruction phase. Each scenario generates a version of the world with a given set of objects such as people, cars, and boats. World objects have properties that can be sensed using the UAV's sensors. The Student knows about general geographical features of the scenario, such as bodies of water and mountains, but must be taught to distinguish between different world objects. World objects, in turn, are used to define teaching tasks.

Three UAV sensors (listed below) were made accessible to the Teacher and Student to provide information about world objects. These sensors are the only percepts available for the Student to observe the environment, but are generally under the control of the Teacher (unless the Student has been asked by the Teacher to perform a procedure that uses these sensors). Therefore, teaching the Student how and when to use these sensors is an important part of teaching a task.

- **Wide-area camera** - provides a 360-degree view of the ground around the UAV. The higher the UAV flies, the wider the range of view. This camera can see objects on the ground once they are in range, but can not acquire detailed information about them.
- **High resolution camera** - provides detailed information about objects when it is aimed at and set to track an object within range. This includes many finer-grained properties of objects (such as whether an object has a cargo hold).
- **Radiation sensor** - can detect the level of radiation of objects in range. The range of this sensor is more limited and requires the plane to fly down to the area it is scanning.

[3] Laminar Research: `http://www.x-plane.com/`

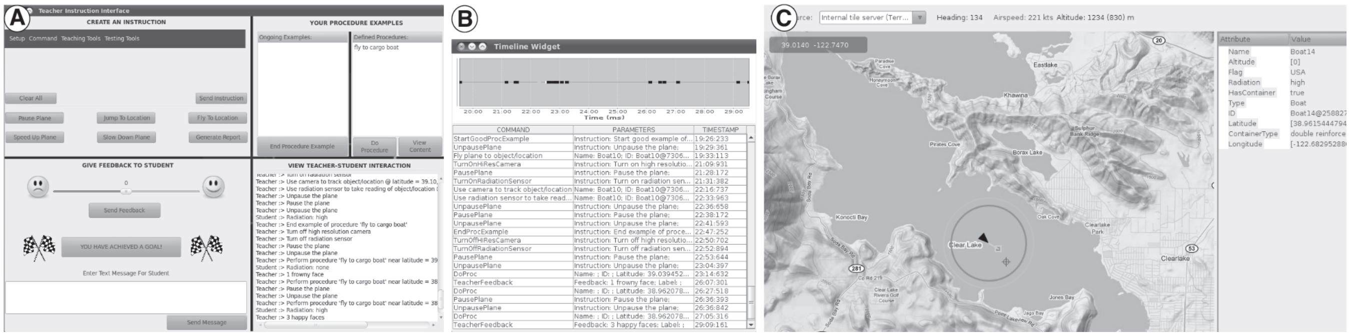

Figure 3: The BLUI Teaching Interface: (A) Teacher Instruction Interface; (B) Timeline; (C) Map Interface

Teacher Interface

The Teacher is provided with three tools to help teach the Student (Fig. 3): (A) the Teacher Instruction Interface used to communicate with Student; (B) a Timeline Display that shows a history of all Teacher instructions; (C) a Map Display that provides information about world objects, UAV sensors and range, and shows the UAV flight path.

The Teacher Instruction Interface (Fig 3-A) was specifically designed to enable the three types of instruction methods discussed earlier. Below we discuss the interface features that may be classified under each Teacher-Student instruction mode:

Teaching by demonstration: The Teacher can group a set of instruction commands into a procedure and demonstrate *good* (positive) and *bad* (negative) examples of a procedure. This can be done in one of two ways. The Teacher can either explicitly state at the beginning of a sequence of commands that they are about to teach a procedure and later explicitly end it, or the Teacher can highlight a set of previous commands in the timeline (Fig. 3-B), and label them as a procedure. Note that this method for teaching procedures has none of the formal semantics or advanced programming interfaces of other teaching systems. Instead it simply allows the Teacher to naturally demonstrate grounded commands in a natural manner and puts the onus on the Student to determine a general policy for enacting these procedures.

Teaching concepts by examples: The Teacher can define concepts (such as "cargo boat") by pointing to and labeling objects (that are visible in the current sensor range) on the map (Fig. 3-C). Names can be re-used and again, the electronic student would need to build a general representation of the concept being taught based on the sensed features of the object (e.g. "Cargo boats are boats with cargo holds").

Teaching by reinforcement: The Teacher can give feedback to the Student, in the form of 1-3 "happy" faces or 1-3 "frowny" faces, to indicate how satisfied he/she is with the Student's performance. The Teacher could also create new or use existing labels to indicate when "goals" were achieved.

Teaching Task

In each of our trials, the human Teacher is presented with the following task:

Your task is to instruct the Student to identify all cargo boats in a specified body of water. There are two main kinds of boats: cargo boats and fishing boats, and you will need to teach the Student how to tell the difference. Once a cargo boat has been identified, the Student needs to take its radiation sensor reading and generate a report.

In order to ensure that the Teacher has no doubt about the correct answer, we provide the Teacher printouts of all scenario files where each boat has been labeled as *cargo* or *fishing*. Additionally, he/she is informed of the property that distinguishes a cargo boat from a fishing boat.

As before, the purpose of the Teaching task with the BLUI is to require that multiple concepts and procedures are taught, some depending on first mastering others. We also want the concepts taught to involve teaching of definitions, procedures, conditions, and potentially sub-goals. In this case, the Teacher would first need to teach the Student the distinction between cargo and fishing boat using camera tracking and assigning object labels. Then the Teacher may instruct the Student to follow a set of actions/procedures that need to be performed every time a cargo boat has been identified. However, since we did not want to bias how the teacher might teach, no description of these possible instruction patterns was given, and instead it was left up to the Teacher to decide how to teach these concepts, given the general task description.

Empirical Results

Thus far, we have run the BLUI WOZ experiments on 12 people. Each participant went through an introductory demo of the teaching interface before beginning the teaching session. On average, the participants spent 30 minutes teaching the Student the assigned task. After the teaching session, each participant was asked whether he/she was able to successfully communicate with the "electronic student"; 7 participants replied "Yes", 3 replied "No" and "2" replied "Mostly". Most participants made use of interface features to teach by demonstration, although some tried to teach the task to the Student by teaching concepts (object labels) exclusively through examples (see Figure 4). Two of the participants only used procedure definitions through demonstration to teach the Student. While we see that the majority

Mode	Phase1	Phase2	Phase3
Demonstration	32.34	34.73	32.93
Concept by Example	33.90	38.81	37.29
Reinforcement	7.50	30.00	62.50

Table 1: Percentage of all commands of a certain mode in each stage.

of the participants taught with multiple instruction modes, it is worth noting that the interface has accommodated two distinct teaching styles. The feedback feature was the least used of the three instruction modes. We also analyzed each teaching session in a fixed time window to detect whether certain instruction modes became more or less popular as the session continued (see Table 1). Interestingly, we did not find any change in preference for teaching by demonstration or teaching by example; however, we did notice a significant shift in the use of teaching through reinforcement. This observation indicates that reinforcement feedback is most useful in this task for fine tuning behavior that has been bootstrapped with other instruction modes.

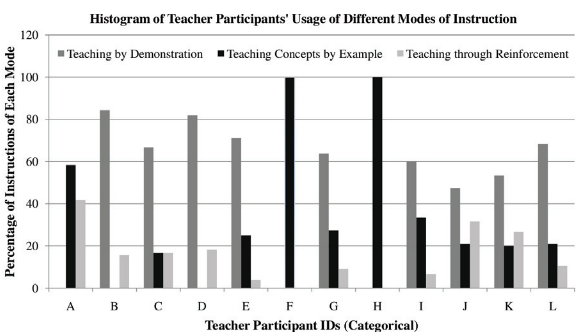

Figure 4: The percentage of each mode of instruction for each participant int he BLUI study.

Conclusion

To the best of our knowledge, our BLUI teacher interface is the first interface to combine all the three common modes of Teacher-Student interaction over the agent's lifetime: teaching by demonstration, teaching concepts by example and teaching through reinforcement. These results are preliminary, and in general there is much more work to be done to better understand human instruction requirements. Also, our interface is still quite primitive in terms of the expressiveness of concepts and procedures. However, so far it does look like the BLUI interface accommodates multiple teaching modes, and we have initial evidence that it supports at least two different teaching styles (mixed demonstration or concepts by example only). Our next step is to use these results to inform the design of the backend of the instruction interface that will package Teacher instructions in an appropriate form for machine learning. This complements the efforts already under way in multi-modal learning systems such as Mable (Mailler et al. 2009).

Acknowledgments The work was supported in part by DARPA contract HR0011-07-C-0060 as part of the Bootstrapped Learning Program and by ONR "Science of Autonomy" contract N00014-09-1-0658. We also thank SRI for support of the X-Plane simulation environment.

References

Argall, B.; Chernova, S.; Veloso, M. M.; and Browning, B. 2009. A survey of robot learning from demonstration. *Robotics and Autonomous Systems* 57(5):469–483.

Chernova, S., and Veloso, M. 2009. Interactive policy learning through confidence-based autonomy. *Journal of Artificial Intelligence Research* 34(1):1–25.

Hewlett, D.; Hoversten, S.; Kerr, W.; Cohen, P. R.; and Chang, Y.-H. 2007. Wubble world. In *AIIDE-07*.

Judah, K.; Roy, S.; Fern, A.; and Dietterich, T. G. 2010. Reinforcement learning via practice and critique advice. In *AAAI*, 481–486.

Knox, W. B., and Stone, P. 2010. Combining manual feedback with subsequent mdp reward signals for reinforcement learning. In *AAMAS*.

Mailler, R.; Bryce, D.; Shen, J.; and Oreilly, C. 2009. MABLE: A modular architecture for bootstrapped learning. In *AAMAS09*.

Morrison, C. T.; Fasel, I. R.; and Cohen, P. R. 2010. Fall 2009 human-instructable computing wizard of oz studies. Technical Report TR10-05, University of Arizona Department of Computer Science, Available at: `http://cs.arizona.edu/~clayton/TR10-05.pdf`.

Natarajan, S.; Kunapuli, G.; Maclin, R.; Page, D.; O'Reilly, C.; Walker, T.; and Shavlik, J. 2010. Learning from human teachers: Issues and challenges for ilp in bootstrap learning. In *AAMAS Workshop on Agents Learning Interactively from Human Teachers*.

Perry, D. E. 2008. Report for the human teacher study. Technical Report TR-2317, BAE Systems.

Smart, W. D., and Kaelbling, L. P. 2002. Effective reinforcement learning for mobile robots. In *ICRA*.

Stumpf, S.; Rajaram, V.; Li, L.; Wong, W.-K.; Burnett, M.; Dietterich, T.; Sullivan, E.; and Herlocker, J. 2009. Interacting meaningfully with machine learning systems: Three experiments. *International Journal of Human-Computer Studies* 67(8):639–662.

Thomaz, A. L., and Breazeal, C. 2008. Teachable robots: Understanding human teaching behavior to build more effective robot learners. *Artificial Intelligence* 172(6-7):716–737.

Thomaz, A. L., and Cakmak, M. 2009. Learning about objects with human teachers. In *HRI*.

Thomaz, A. L.; Hoffman, G.; and Breazeal, C. 2006. Reinforcement learning with human teachers: Understanding how people want to teach robots. In *ROMAN*.

Walsh, T. J.; Subramanian, K.; Littman, M. L.; and Diuk, C. 2010. Generalizing apprenticeship learning across hypothesis classes. In *ICML*.