Challenges to Decoding the Intention Behind Natural Instruction

Raquel Torres Peralta, Tasneem Kaochar, Ian R. Fasel, Clayton T. Morrison, Thomas J. Walsh, Paul R. Cohen
University of Arizona, Department of Computer Science
Tucson, Arizona
{rtorres,tkaochar,clayton,ianfasel,twalsh,cohen}@cs.arizona.edu

1 Introduction

Human-instructable computing seeks to develop intelligent devices that can be *taught* by natural human instruction. By "natural," we mean patterns of communication that humans use in every-day human-to-human teaching, such as (1) giving explicit *definitions* and *examples* of concepts, rules and conditions, (2) *describing* and providing *demonstrations* of procedures, and (3) providing various kinds of *feedback* in response to the student's behavior. We refer to these patterns of instruction as *natural instruction methods*.

Translating these forms of instruction into inputs for machine learning algorithms (e.g., example-label pairs, execution traces, rewards for reinforcement learning) is non-trivial, because humans opportunistically (and sometimes quite rapidly) interleave instruction methods without explicitly stating what method of teaching they are using. While the problem is intuitive, the actual extent to which teachers change between methods, and the cues that might be available to automatically detect these changes, has not to our knowledge been explicitly studied.

To gain a better understanding of how humans naturally deliver instructions, and what will be needed to eventually map natural instruction into machine learning methods (such as concept [Natarajan *et al.*, 2010], procedure [Winner and Veloso, 2003], or reinforcement [Knox and Stone, 2010] learning algorithms) in an end-to-end human-instructable agent, we perform here a behavioral study similar to that of [Kim *et al.*, 2009], in which human teachers provide instructions to a simulated robot student that is secretly controlled by a human. Through careful analysis of the transcripts, we find that while many teaching patterns are straightforward to map into machine learning targets, humans do indeed use a number of teaching patterns that are difficult to automatically interpret. In particular we found that humans often use what we term *implicit* teaching methods, and we detail here several forms of these implicit methods. We also found that we could group teachers according to three different *organizational styles* based on the frequency and manner of interleaving teaching methods. To highlight the challenges involved in interpreting these and other teacher instructions and teaching styles, we conclude by describing an initial automatic teacher instruction recognition system and describe its results.

2 Experiments

In our experimental protocol, novice human participants used a multi-modal teaching interface to teach a series of inter-dependent concepts and tasks to a simulated electronic student in a Wizard of Oz paradigm.

The teaching interface (shown partly in Fig. 1) was designed so that we did not depend on parsing and interpreting natural language input but instead used standard GUI elements to allow users to provide instructions in the form of text entry, menu and button clicks, and mouse manipulation of icons that could be easily parsed by a machine, even if the true "meaning" of those instructions might be difficult to interpret by an algorithm. While any interface naturally imposes some structure on the user, we attempted to make it as easy as possible for teachers to switch between procedure demonstration, object labeling, testing, and providing feedback to the student (both in the form of positive or negative rewards and indications that a specific goal has been achieved).

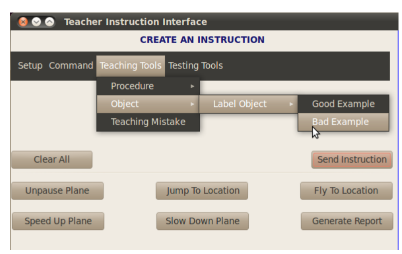

Figure 1: *Teaching Interface: Instruction Command Interface*

We asked 44 participants to teach a task to an "electronic student" in an Intelligence, Surveillance and Reconnaissance (ISR) domain in which the Student controls a simulated unmanned aerial vehicle (UAV) and is taught to carry out missions. The UAV has limited ability to sense the world – camera and radiation sensors – and there are two kinds of objects in the world: cargo boats and fishing boats. The Teacher's task was to teach the Student how to distinguish the types of boat, to use the radiation sensor only on cargo boats, and to generate a report of any high radiation readings. Teaching sessions were recorded and a transcript of the Teacher-Student interaction was generated for each teaching session.

3 Analysis of Teaching Patterns

Our analysis of the transcripts and post-teaching questionnaires is exploratory in nature. Our goal is to discover if human instructions with a flexible interface such as the Instruction Command Interface (ICI) could be easily translated into inputs for machine learning systems, and if not, identify the kinds of teaching patterns a system would need to accommodate. We found that there are indeed several teaching patterns that may be challenging for automated systems, but which are quite common and natural for untrained human teachers.

3.1 Multiple uses of the procedure demonstration facility

According to our transcripts, an estimated 41% used the procedure demonstration facility in ways that were not originally foreseen in our design. In particular:

A. Implicit Object Labeling. Of the 41% of teachers who used the procedure demonstration facility in unexpected ways, 25% appear to have used it as a means to label objects. In these cases, teachers maneuvered the UAV up to an object and then appeared to use the name of the object as the name of a procedure. We called this general pattern *implicit labeling*.

To illustrate the challenge of identifying implicit labeling, we first show an example of *explicit labeling*. In the following transcript excerpt[1], the Teacher selects a boat from the map interface (pointing and clicking with the mouse) and then enters the object label; this produces the following command in the transcript:

```
12: 12:32 Object @lat. = 39.04, long. = -122.89
    is a good example of object label 'Cargo Boat'
    (Object name = Boat11)
```

This command is unambiguous and directly interpretable as labeling the object.

Contrastingly, in implicit labeling the teacher first provides the label as the name of a new procedure, the procedure is started, the UAV is positioned near an object, and then the student may be instructed to track the object. This is demonstrated in the following transcript excerpt; here the teacher introduces the concepts 'Cargo Boat' and 'Fishing Boat':

```
39: 43:29 T: Start good example of procedure 'Cargo Boat'
40: 43:38 T: Fly plane to object/location @ lat= 39.10,
             lon= -122.82 (Object name = Boat10)
41: 44:13 T: Use camera to track object/location @ lat
             39.10, lon -122.82 (Object name = Boat10)
43: 44:52 T: End example of procedure 'Cargo Boat'
   (...change in location...)
53: 48:23 T: Start good example of procedure 'Fishing Boat'
54: 48:30 T: Fly plane to object/location @ lat= 39.09,
             lon= -122.86 (Object name = Boat12)
55: 49:08 T: Use camera to track object/location @ lat.
             39.04, lon. -122.80 (Object name = Boat12)
58: 51:58 T: End example of procedure 'Fishing Boat'
```

A direct interpretation of these actions as defining a procedure appears to miss that the teacher is labeling the boats, however, the name conventions provide the main evidence. In each of these cases, the teacher instructed the student to move the UAV very close to the object (e.g., directly above the boat) and track it (Figure 2). Provided we are correct in interpreting this as implicit labeling, the challenge would be for an automated interface to infer that the teacher is indeed labeling.

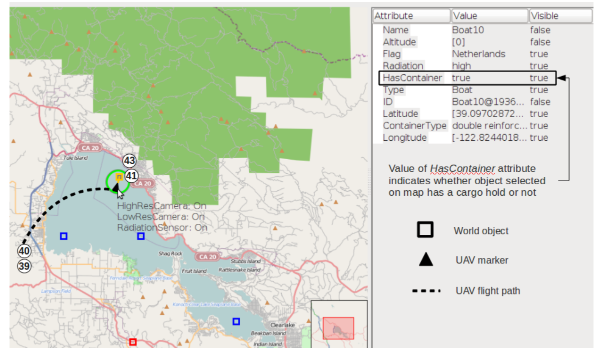

Figure 2: Map Display: A display of the progression of teacher commands (31-43) defining a good example of procedure *Cargo Boat* from the earlier transcript excerpt.

B. Implicit Procedure Definition. In addition to the unexpected use of procedure demonstration for implicit object labeling, we also observed many cases (for all but 2 teachers) in which procedures appeared to be taught but the procedure facility was not used. In these cases, we observed teachers directing the Student to perform sequences of steps repeatedly, such as using the sensors to gather information and then generating a report. It appears that the teachers assumed the Student was "paying attention" to these sequences and would be able to infer the boundaries between the implicit procedure definitions.

C. Ill-defined Procedure Boundaries. We also found that human teachers frequently give procedure definitions with imprecise limits: the teacher provides action commands well outside the explicitly defined boundaries, although those appear to be key parts of the intended procedure. These behavior illustrates that we cannot rely on the human to provide correctly the begins and ends for every lesson.

Overall, these varied uses of the procedure demonstration facility reveal that the mapping from natural human teaching to machine learning inputs requires some sophistication.

3.2 Frequent use of testing followed by feedback

Another class of patterns we observed pertains to the frequency of testing paired with teaching. Half of our teachers preferred to test the Student right after each lesson while the rest tested less frequently. Only five participants did not test the Student at all.

Testing and Feedback. One of the most frequent patterns we observed was the 'teach-test-feedback' loop, where the Teacher immediately follows the introduction of a new concept by testing the Student and then complementing the lesson with some feedback on Student's performance.

3.3 Different "types" of teachers

Finally, our transcript analysis also revealed three distinct styles of teaching based on the *organization* of lessons: *(1) Structured* teachers (16%) provided instructions to the Student in a very organized way, always labeling objects explic-

[1]Transcript excerpts are formatted as follows: Line number, timestamp, source (T indicates Teacher's command and SR indicates Student's response) and content.

itly and using the procedure demonstration facility to define procedures. They tested the Student only on previously taught concepts. *(2) Semi-structured* teachers (50%) began with a less structured teaching style but became progressively more structured as the teaching session continued. They made use of the GUI features almost as intended, sometimes with early exploration of usage. Some of these teachers never made use of the interface's procedure construct tool. *(3) Free style* teachers (34%) were the most difficult to follow, mainly because these teachers made use of GUI features in novel ways, such as using procedures for implicit labeling and sometimes using testing tools before teaching, perhaps because of high expectations of the Student's initial capabilities.

4 Automatically Decoding Teacher Intentions

The goal of this study was to identify teaching patterns that would allow us to automatically convert teacher actions into machine learning patterns. In this section, we describe a baseline system that attempts to convert as much of a transcript as possible into segmented teacher intentions. This includes identifying which instructions appear to be about "setting up a lesson" (e.g., maneuvering the UAV to a particular location) as opposed to actual object or procedure lessons, and when a procedure is really an *implicit definition*. This initial approach is intentionally simple, in order to draw attention to areas where simple techniques are likely to fail.

4.1 Human Transcript Annotation

In the first phase of our approach we identified a set of labels that can be applied to each line of a transcript to indicate how the teacher intended each instruction to be used by a learning system. Two human annotators labeled each of the 44 transcripts, according to the set shown in left column of Figure 3.

4.2 Automatic Transcript Annotation

Once the transcripts were annotated, we constructed a simple script that attempted to use the heuristics the human annotators used while performing the annotation. The center column of Figure 3 provides a quick description of these heuristics.

All 44 transcripts were annotated using the script and the resulting labels were compared with the human labels. We found that for the most part, transcripts could be automatically annotated based on the small set of rules and heuristics. The right column of Figure 3 shows which commands could be automatically translated to agree with the human labels: instructions labeled with a checkmark were annotated with 100% accuracy. Instructions labeled with an 'X' or the check with a star were not labeled accurately: in these cases, the script could not handle implicit object and procedure definitions, and "Setting Up" commands were identified correctly only when the script correctly identify the teaching that followed.

Label	Labeling Heuristic	Script Successful
Explicit Object Label	Teaching command defining good or bad object example using the *object labeling construct*	√
Implicit Object Label	Teaching command defining good or bad object example using the *procedure construct*	x
Explicit Procedure Definition	Teaching command defining a procedure using the *procedure construct*	√
Implicit Procedure Definition	Repetition of similar set of action commands (camera track, take radiation reading or picture, generate report) in different scenario locations	x
Test	Testing command (ask Student to perform a previously taught procedure or give the label of a world object)	√
Feedback	Command evaluating Student's performance (1-3 *happy* or *frowny* faces)	√
Goal Specification	Command labeling a goal that Student has achieved	√
Setting Up for Teaching	All setup commands (change in location, speed, altitude or sensor settings) preceding a teaching command (object labeling or procedure definition)	√*
Setting Up for Testing	All setup commands preceding a testing command	√

Figure 3: Automatic Transcript Annotation (* means only when teaching command has been correctly identified)

5 Discussion and Conclusions

Our analysis has shown that human teachers often give instructions that require considerable interpretation and are not amenable to straightforward, direct identification. Humans make extensive use of implicit instructions, often fail to clearly indicate the beginnings and ends of lessons, appear to rely on meaningful linguistic names for objects and procedure labels,

We found that while a relatively simple set of rules can accommodate important aspects of human instruction, a successful instructable robot will need to deal with the more subtle forms of teaching we found in our transcripts.

A critical next step is to improve recognition of implicit instructions used in semi-structured and freestyle teaching. This in turn will allow us to better detect the different organizational styles of humans. More specifically, the automatic identification of implicit object labeling and implicit procedures as well as teacher-specified procedure boundaries needs to be improved and tested. Not only will this help robots learn from natural instructions, it will also be useful for helping the robot actively guide the interaction through robot-human feedback.

Acknowledgments: Special thanks to Sumin Byeon for his help in conducting the WoZ experiments. This work was supported by DARPA under contract 27001328 and ONR Science of Autonomy contract N00014-09-1-0658.

References

[Kim *et al.*, 2009] E.S. Kim, D. Leyzberg, K.M. Tsui, and B. Scassellati. How people talk when teaching a robot. In *Human Robot Interaction*, pages 23–30, 2009.

[Knox and Stone, 2010] W. Bradley Knox and Peter Stone. Combining manual feedback with subsequent MDP reward signals for reinforcement learning. In *AAMAS*, 2010.

[Natarajan *et al.*, 2010] Sriraam Natarajan, Gautam Kunapuli, Richard Maclin, David Page, Ciaran O'Reilly, Trevor Walker, and Jude Shavlik. Learning from human teachers: Issues and challenges for ILP in bootstrap learning. In *AAMAS Workshop on Agents Learning Interactively from Human Teachers*, 2010.

[Winner and Veloso, 2003] Elly Winner and Manuela M. Veloso. DISTILL: Towards learning domain-specific planners by example. In *Proceedings of ICML'03*, 2003.