Analysis of Situated Interactive Non-Expert Instruction of A Hierarchical Task to a Learning Robot

by

Preeti Ramaraj

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Computer Science and Engineering)
in the University of Michigan
2023

Doctoral Committee:

Emeritus Professor John E. Laird, Chair
Assistant Professor Nikola Banovic
Professor Joyce Y. Chai
Dr. Shiwali Mohan, PARC, SRI International
Associate Professor Chris Quintana

Preeti Ramaraj
preetir@umich.edu
ORCID iD: 0000-0002-3847-9462

ACKNOWLEDGEMENTS

Deciding to do a PhD is knowing that you are embarking upon a long journey but it is not always clear how long the journey is really going to be. While my PhD feels like an eternity, there are more than a few people without whom I could never have gotten to this side.

I want to first thank my committee as a whole - Prof. John Laird, Prof. Nikola Banovic, Prof. Joyce Chai, Prof. Chris Quintana, and Dr. Shiwali Mohan for being a very supportive group of people who were understanding when my thesis changed shape from one year to the next.

I was very lucky to have met my advisor John 8 years ago with dreams in my eyes of what research could be. It did not matter that I had not even engaged in an ounce of research when I first spoke to him - he treated my ideas as valuable then as he does now after working together for all these years. He encouraged me to pursue the research direction that I was most interested in, even if it was not in his wheelhouse, encouraging me to seek guidance from wherever it made sense. It is a testament to our journey that my thesis is a qualitative analysis of a user study. Outside of research among all the tumultousness in the department, I knew I was supported even if my views were different from his. I am coming out of this with a continued love of research and the hope to stay in it, which I credit to John.

I asked Nikola to be my project mentor during a particularly difficult phase when it was not clear what my PhD journey ahead was going to look like. However, since day 1, he has been nothing but supportive, and believed in my project ideas even when I could not get myself to believe them. He guided me from an HCI perspective and was with me every step of the way through my dissertation project. I wish I had met him earlier, but I am grateful for the time and mentorship I have received.

The first time I spoke to Shiwali was a 3 hour conversation about research that functioned as an informal interview for my internship under her guidance. But that internship led to a mentorship for which I cannot be thankful enough. We worked together remotely for a year, and yet, it was one of my most productive years in the PhD. For this, I have to describe my gratitude for Shiwali serving not only as my research guide, reminding me of my worth as a scientist, but also as an overall mentor, who saw me fully for who I was and lifted me up more times than I can recount. I count myself truly fortunate to have met her.

I also want to thank many current and former members of the Soar lab namely Mazin Assanie, Nate Derbinsky, James Boggs, Lizzie Goeddel, Peter Lindes, Steven Jones, James Kirk, Aaron Mininger, Jule Schatz, and Bryan Stearns who welcomed me when I was a new member of the lab and continued to provide a warm presence through various parts of my PhD. I want to shout out to Steven Jones whose company and music recommendations helped me through several dark days. I also want to especially thank Aaron who was not only kind in always helping me while sitting at the desk beside me, but also for doing such incredible software engineering work. Despite him being in Congo, I could make all the possible changes in Rosie without too much pain. This was a huge technical effort made easy because of his good clean code with very useful comments.

Using NVivo software for doing my qualitative analysis and writing about it felt like a herculean task. It was made easier due to help and input from Anindya Das Antar, Sumit Asthana and Snehal Prabhudesai. I also appreciate the many coffee chats with Prof. Chad Jenkins which were most times interesting, but oftentimes provided much-needed encouragement and reminders that I was capable of becoming a successful researcher at the end of my PhD.

A PhD is not just about becoming an accomplished yet humble researcher. Given the many many unknowns, it is also about personal growth and discovering many facets of yourself. For that, I have way too many people to thank for witnessing my journey from near and far.

I want to thank Mohamed El Banani for being the warm and jovial presence that I looked forward to every time I entered my lab. I would not be here without his support through my ups and downs and very valuable research feedback which were sometimes harsh but always came from a place of love (though he may not readily admit to this).

Harman Kaur is my research sister and friend mentor. She introduced me to one of the loves of my life - qualitative research and believed and respected my research ability when I thought I had none. I am so grateful for our long walks discussing home away from home, her presence and reassurances no matter when I needed it, and of course, her cat babies. I am most thankful to Megan Shearer for being the friend who kept me grounded and cared for me in explicit ways even when I ran away with any and every unnecessary expectation of my own self. I discovered my love for all outdoor activities - biking, tubing, cross-country skiing (even if it was that one time) because of her encouraging and much-needed company. She was my co-chair who was meant to be. Our text chain continues to support me even after my PhD and I can never stop being grateful for them.

Jeeheh Oh - my forever roomie. Her fierce protectiveness of me, her ability to spot when I was down and make me calm, and the effect of her presence can hardly be defined in these pages. I was lucky to find her that first day at school, and I am glad that she is a part of my journey. I am also thankful to my quarantine family which includes Meera Krishnamoorthy, Jeeheh Oh, Jiaxuan Wang, and Varshini Ramaraj who kept me sane with their presence through the pandemic.

I met Laura Burdick as well on my first day of school and would never have imagined that this was how our friendship would flourish. I want to thank her for sharing her time, her love, and her kids with me. They brought me joy during low times and she brought me consistent company through our walks in the midst of nature.

Josie Granner - Where do I even begin? From our gift of focus spreadsheet to our weekly check-ins, she has been a crucial part of this phase of my life besides our friendship otherwise. I am grateful to her for giving me the opportunity to be wholly me through everything and without it, I would not be here, still whole.

I want to thank Jaylin Herskovitz, especially for her company during a difficult time, but also for our many encouraging conversations and fun times as well. I want to thank Elizabeth Harlow for some much-needed co-working company and friendship, especially during this last year. It was wonderful having monthly coffee chats with Wilka Carvalho discussing all things about the world, but also running into his friendly presence at school. I am also thankful to Jovin Dsa for being a familiar loving presence when I moved to this country but also for showing me all the wonderful things in Ann Arbor and his absolutely brilliant company.

I am fortunate that friends from my past life continued to stay in my life and witness my journey. I am most grateful to Anjana Iyer who listened to all my trials and tribulations through the years and provided some vital support even if it was fully virtual, and provided her home and beautiful country to me when I truly needed a break. Cinjal Shah continued to stay in touch with me on a weekly basis ever since I began this journey of moving to this country. His consistent presence through text mattered more than I can describe. I am also really grateful to Ashwini Ashokkumar who not only provided so much validation from an academic perspective but was such a fantastic co-working buddy that I travelled across the country to co-work with her. I am also grateful to Pooja Pai, Sanaya Pinto, Kirtana Puthran, Siddharth Raikar, Namrata Rao, and Eshita Shah for their timely calls and texts where I was reminded that I am loved for who I am, irrespective of my PhD.

None of this could have been achieved without the support of my family. I consider it the most serendipitous thing that my sister Varshini Ramaraj lived just a street away from me. Her couch has functioned as a familiar space for many a working afternoon with the promise of some very welcome snacks. Knowing she is in my corner has sustained me more than I can describe. Appa and Amma always told me that I could study as much as I wanted, for as long as I wanted, and without such encouragement, I could not have even imagined doing this PhD. I love them all for supporting me through the most difficult times and the distance.

Last, I am grateful for my mind, body, and soul that were resilient through these grueling 8 years, and truly thankful that I learned to move from surviving to thriving.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE

LIST OF TABLES

TABLE

LIST OF APPENDICES

ABSTRACT

Interactive Task Learning (ITL) is an approach to designing robots that can learn tasks on the fly from human instruction and demonstration in a shared environment. ITL approaches until now have focused on extending a robot's capabilities so that it can efficiently learn a wide variety of task components as a part of hierarchical tasks from an expert human instructor. However, the typical instructor is likely to be a non-expert who does not have a good mental model of the robot's physical and mental capabilities and is therefore likely to face challenges during the teaching process. Our hypothesis is that we need to focus on building better robot learners that actively interact to make the teaching process more accessible and efficient for the non-expert teacher.

Towards this goal, we conducted a human participant think-aloud study (N=14) where we asked participants to teach a multi-step hierarchical task of baking a pizza to an ITL robot in a simulated environment. We built a templated instruction interface, which participants could use to provide two types of action and two types of goal instructions to the robot. In this dissertation, we present a qualitative analysis of the data collected from this study. We identify the different types of knowledge that non-expert teachers leveraged during instruction, namely knowledge about the task, environment, interface, and robot. Participants had access to this knowledge as a result of their prior experiences as well as through interaction with the robot using the interface and the environment. We also present our characterization of different aspects of the teaching processes used by non-expert teachers. We observed that participants developed complex strategies to teach the task to the robot. They actively evaluated the robot's task progress and became effective at teaching the task. However, we found that participants encountered challenges during the teaching process, which included difficulty in providing desired instructions and encountering failure situations. Participants also found the knowledge about the task, environment, interface, and robot to be insufficient at

times, which led them to have incomplete and incorrect mental models. However, participants were motivated to continue teaching even after encountering these challenges and 13 out of 14 participants successfully finished teaching the task to the robot.

To address challenges faced by non-expert teachers, we propose extensions to existing robot interaction approaches that would allow the robot to use its knowledge of the various aspects of the task and itself to help the instructor teach better. To enable the non-expert instructor to provide their desired instructions, we propose extensions to the instruction interface to allow for more complex instructions and accommodate free-form instructions when necessary. To improve the instructor's mental models, we propose that the robot can leverage its access to various sources of knowledge to provide relevant updates and knowledge to the instructor. To determine how the robot can help the non-expert instructor recover from these failures, we recommend the need for more research to understand more about the different failures that can occur during teaching and to conduct studies to evaluate how non-experts can effectively resolve them. In the end, we emphasize the value of iterative development in this dissertation towards creating robots that can successfully interact with non-experts in the real world.

CHAPTER 1

Introduction

We envision a future where robots will operate alongside humans in dynamic environments such as homes, offices, hospitals, and warehouses. These can take the form of performing high-effort tasks in warehouses or assisting older populations at their homes doing everyday chores. To truly contribute effectively across these domains, the robots need to be able to perform tasks more diverse than can be planned for at design time. An attractive solution is to enable people to teach robots new tasks and relevant information about their environments on the fly.

Interactive Task Learning (ITL) [31] is a research approach that contributes to this solution through the design of agents and robots that can learn complete task definitions and associated knowledge through natural interaction with a human instructor. ITL approaches until now have focused on extending a robot's capabilities so that it can efficiently learn a wide variety of task components from a human instructor and also leverage its knowledge to improve its task performance. This focus on learning and performance relies on the assumption that the instructor is an expert who has a good mental model of the learner robot, i.e., the instructor has a good understanding of what the learner robot's physical capabilities, what it knows and does not know, how it processes information, how new instructions will interact with previous instructions, and whether it can apply its knowledge to do tasks in the environment and how it proceeds to do so.

This knowledge is vital because, without reasonable estimates of a learner robot's capabilities, it is challenging for a human teacher to teach efficiently and effectively. ITL relies on the fact that people interactively teach each other, and they can leverage that skill to teach robots as well. People are successful at teaching other people because they leverage their common ground [13, 27] to determine the learner's knowledge and capabilities and teach accordingly. However, in the case of human-robot teaching interactions, people do not share similar common ground with the robot to leverage during instruction unless the instructor is an expert with specialized knowledge about the robot. In fact, if the instructor (irrespective of being an expert or not) has never interacted with this robot before, they are likely to possess incomplete or incorrect assumptions about what the robot knows, can do, or cannot do. This will likely result in challenges that an instructor will face while teaching the robot, such as encountering failures due to imperfect and inefficient instructions.

Prior work has examined methods such as robot transparency (*What can the robot see right now?*), providing explanations (*I don't know what a block is*), asking for help (*Can you tell me what a block is?*) to improve individual task performance (*how can the robot successfully build a tower?*). While that work has demonstrated that such mechanisms improve a person's mental model of the robot, it has been limited to the scope of the robot learning individual actions or performing known tasks. There is little work on what, when, and how the robot should communicate to improve human-robot teaching interaction, particularly for multi-step, hierarchical tasks.

My hypothesis is that we need to focus on building better robot learners that actively interact in a way to make the teaching process more accessible and efficient for the non-expert teacher. A learner must understand what the teacher intends with each instruction and respond appropriately while absorbing new information as it is presented. As a first step to understanding the properties of such interaction, we need to identify the current gaps that make the teaching process difficult for a non-expert. Toward this goal, this dissertation attempts to answer the following questions:

- *How do non-experts teach multi-step hierarchical tasks to an ITL robot?*: We need to understand the various aspects of a teaching process as it relates to a multi-step hierarchical task. This includes how non-experts intend to teach the task, how they decompose it into individual instructions, and their resultant success or failure at teaching the task.

- *Do instruction templates contribute to efficient and effective teaching?*: While the robot has a language-comprehension system that can understand complex language instructions, it is stilted. Through prior pilot studies, we learned that non-experts find it difficult to create robot-understandable instructions from scratch. This is because the robot cannot engage in human-like flexible interaction. It is imperative to understand if instruction templates can serve as an intermediate solution to overcome the problem of mismatched language skills while still enabling successful teaching interactions.

- *What are some salient problems people face that can be solved by extensions to the robot's interaction capabilities?*: Given the complex nature of the robot's learning system and the non-expert's lack of a mental model of the robot, people are likely to face problems while instructing the robot. If we can determine what these problems are, we can build corresponding interaction mechanisms in the robot to alleviate these issues, making it a better learner.

Therefore, in this dissertation, we conducted a think-aloud study (N=14) where non-expert teachers directly interacted with an ITL robot to teach it a hierarchical, multi-step baking task through situated interaction using predefined instruction templates. We studied this problem using an existing robotic agent Rosie, an ITL robot implemented in the Soar cognitive architecture and set up in a simulated kitchen environment. Rosie can learn over 60 games and puzzles [25], mobile delivery, navigation, and procedural tasks [39] and various procedural kitchen-specific tasks [42] using a combination of situated natural language commands and demonstration.

This situated interaction study setup provides rich sources of information to study the teaching process since it includes the environment where the teaching interaction is situated, the robot responses within the context of the teaching process, and the interface using which the teacher interacts with the robot. We propose methods using the analysis from this study to build robots that help humans build mental models of robots and are easier and more natural to teach.

1.1 Research Contributions

This dissertation makes the following research contributions:

- We describe a qualitative analysis of a think-aloud study where we identified and characterized different aspects of non-experts directly teaching a hierarchical task to a robot.
  - We provide a detailed breakdown of how participants use the instruction interface and the simulated environment as a part of the interaction.
  - We describe the participants' interpretations of a complex learning robot regarding its knowledge, capabilities, and learning process through its verbal responses and actions in the environment.
- We identify relevant failure situations that are specific to a multi-step hierarchical teaching process by non-experts.
- We provide computational insights, potential extensions to existing approaches, and scope for research to address these failure situations.

One of the key contributions of this dissertation is that this research was executed using direct interaction with an existing ITL system. Wizard-of-Oz techniques enable researchers to determine ideal robot implementations that satisfy non-expert needs, but we cannot develop robots that can function in the actual world until we investigate the features of direct interaction. While this was an initial exploration, this study sets this research up to pursue iterative design. By proposing potential implementations for the robot, future implementations can be effectively compared with the results of our current study. We propose that more robotic systems must engage in iterative processes to contribute to the goal of building robotic systems that can operate in the real world.

1.2 Road map

Chapter 2 provides a background on the research approach of Interactive Task Learning, specifically highlighting the desiderata that are relevant to the goals of this dissertation.

Chapter 3 reviews literature related to our research and prior work that has been done to improve human-robot interaction.

Chapter 4 gives an overview of Rosie, the ITL robot used in this thesis. Additionally, it defines the problem of teaching a multi-step hierarchical task to an ITL robot and describes the knowledge and expertise required to teach such a task.

Chapter 5 provides a description of the simulated environment where Rosie was embodied as a mobile robot. It also provides a description of the templated instruction interface that was developed for this dissertation.

Chapter 6 introduces the research questions and then describes the design and method used to study how non-experts teach a baking task to a robot. We then describe the participant recruitment process and present participant demographic and task background information.

In Chapter 7, we first provide the findings from the qualitative analysis performed using the multidimensional data collected from the study with 14 participants. We organize the findings using the different types of knowledge used or required by the participants. We then provide a quantitative analysis by analyzing the command breakdowns in the task, as well as the failures encountered during the task.

In Chapter 8, we situate the findings from Chapter 7 using our proposed research questions. We then provide an interaction design proposal in Chapter 9, where we propose extensions to robot interaction mechanisms that can contribute to an efficient and effective non-expert teaching process.

Chapter 10 restates the contributions of this dissertation and provides a summary of the proposed extensions from Chapter 9.

CHAPTER 2

Interactive Task Learning

Interactive Task Learning [31] is a research approach that focuses on designing agents that can learn complete task definitions and associated knowledge through natural interaction with a human instructor. Interactive Task Learning describes three axes of desiderata, namely, (a) Dynamic Scalability to new tasks, (b) Task Performance, and (c) Ease of Interaction. Many efforts toward building ITL agents aim to satisfy these desiderata using different approaches. These approaches can be described in terms of dimensions of natural interaction (language, non-verbal modalities), types of task knowledge learned (concepts, actions, procedures), as well as the operative domains (virtual environments, mobile phones, robot simulators).

Dynamic Scalability to new tasks refers to building agents that show diversity in task learning. This diversity can take the form of learning different aspects of a task, learning different types of knowledge, learning from different types of instructions, or learning from and applying its learned knowledge to different situations and environments. Towards this goal, we have seen progress in general task learning in agents that can learn different types of tasks and concepts and generalize across multiple examples (e.g., agents built on DIARC [50], Rosie [25, 39, 40] and AILEEN [41]), as well as agents whose learning is domain-general (e.g., DIARC [50], Pumice [33] and Rosie).

The axis of Task Performance focuses on how the agent uses its learned knowledge to perform tasks efficiently and effectively. Rosie leverages the Soar cognitive architecture to reason about its situation on the fly. It also compiles deliberate processing to save time and processing effort for similar tasks it may encounter in the future. Other ITL agents have focused only on learning to successfully ground language in the environment through explicit or implicit learning [10, 34, 57], to eventually support task learning.

Ease of Interaction, as the name suggests, refers to the ease with which the human can interact with the agent. Interactive Task Learning, by definition, requires that there is a teacher present to teach the learner. We can define teaching as a specialized form of interaction where the teacher interacts with the learner with the specific intention of communicating knowledge to the learner and the goal that the student learn that specific knowledge. Therefore, this axis supports the overarching goal of ITL to enable non-expert human teachers to interact with these intelligent agents naturally. Despite this goal, most prior work in ITL has focused on enabling dynamic scalability and efficient task performance in ITL agents by building and extending their learning abilities. There is only limited work that focuses on enabling Ease of Interaction. In this thesis, our emphasis is not on Performance and Dynamic Scalability; instead, we focus on Ease of Interaction.

To enable Ease of Interaction, the following desiderata need to be achieved - Efficient Task Learning, Effective Interaction, Accessible Interaction, and Efficient Interaction.

- *Efficient Task Learning* corresponds to an agent being efficient in terms of time to process instructions while minimizing its interactions (such as asking questions) by maximizing the knowledge it can retrieve from these instructions. This requires the agent to be reactive and speedy in processing new information along with its existing knowledge and not take an inordinate amount of time to respond to the instructor.

- *Effective Interaction* corresponds to an agent's ability to extract useful meaning from instruction irrespective of the modality through which it is provided and be robust to instruction errors. In addition, the agent should interact so that the instructor can identify the intention and content of the interaction based on the current context. However, research in natural language has not progressed to the point where ITL agents can adeptly engage in flexible interactions with humans. Additionally, agents do not share adequate common ground with humans to recover quickly from failures, unlike those observed in typical human interactions.

- An agent capable of *Accessible Interaction* should enable an untrained instructor to teach in a natural and unconstrained manner. This means that communication with the agent should support multiple natural modalities and not require extensive knowledge of the internal working and knowledge of the system.

- Lastly, *Efficient Interaction* is focused on minimizing the information that the instructor needs to communicate, both in terms of concepts related to a task as well as quantitative measures such as the number of words and turns required to communicate this information.

Both Accessible Interaction and Efficient Interaction require that an agent can provide information about its task knowledge and be capable of explaining its reasoning and performance if necessary. This would enable the instructor to leverage this information to skip instructions for the aspects of the task that the agent already knows and focus on aspects it does not. However, given the limited flexibility and specialized knowledge required to interact with typical learning agents, it is currently a steep learning curve for untrained instructors to teach ITL agents.

The agent we use in this thesis, Rosie, is currently set up for Efficient Task Learning from instruction and demonstration, given the prior work's focus on learning. However, the instructor is expected to have extensive knowledge about the robot or, at the very least, have been trained to interact with Rosie. As a result, interactions are less efficient and more difficult for instructors who are not robot experts. This exhibits the previous lack of focus on Efficient and Effective Interaction. Therefore, in this thesis, our goal is to study the teaching processes employed by non-experts to determine the barriers to Ease of Interaction in Rosie and ITL in general and propose interaction mechanisms that will contribute to *efficient* and *effective* teaching interactions in the future.

CHAPTER 3

Related Work

In this chapter, we first describe the closest prior work that has studied human teaching interactions, where humans provide inputs of various granularities to different kinds of learning systems. While this dissertation focuses on human-robot teaching interactions, there has been considerable work in related fields to understand how ease of interaction can be achieved in human-agent interactions. Therefore, we describe the related work comprising human interaction with tutorial-based systems, AI agents, and other robotic systems. Lastly, we describe what we can learn from research on explainability to enable efficient and effective human-AI interactions and describe work in this space that has leveraged explainability towards this goal.

3.1 Instruction of Intelligent Systems

Interactive Machine Learning (IML) or Interactive Machine Teaching (IMT) are interchangeably used to describe human-in-the-loop systems that allow the human to provide relevant information to influence the building of a machine learning model to do future tasks. The teacher's role can vary depending on the granularity or type of information provided (labels, reinforcement signals, user interaction), as well as the relative expertise of the teacher (non-expert, data scientist, subject-matter expert, ML practitioner). Prior work has studied how people teach in IML/IMT settings and confirms that people are motivated to provide explicit instructions to teach agents. Krening and Feigh [28] studied people's interaction experiences using critique or action advice to train the same reinforcement learning agent. Participants preferred giving action instructions because they could immediately observe the results of their advice, thus getting confirmation that the agent was incorporating their input. Similar to our study, Sultanum et al. [53] conducted a formative study to understand how people would teach a hypothetical computational learner. Even within a simple task of detecting a visual concept, people engaged in an iterative teaching workflow that combined providing and refining knowledge over time. While their study focused on the processes people undertook to teach a single concept, our study focuses on the teaching workflow process for a multi-step hierarchical task.

Prior work has also specifically studied how non-expert teachers teach tasks to a robot. Kaochar et al. [22] study natural instruction methods through examples, demonstrations, and reinforcement signals. Their task included teaching a procedure to the robot, and they observed that participants often taught procedures as a set of consecutive actions repeatedly in different locations but did not explicitly define the set as an individual procedure in advance of providing the actions. In human-robot interactions in a collaborative task setting, comparisons between human-human and human-robot interactions show that people provide more physical action instructions and less belief update instructions while interacting with a robot [36]. Consistent with these findings, in our study, participants often referred to the higher-level task they were teaching but either forgot to define it first or used consecutive primitive actions to instruct the robot to do the task rather than organize it in advance as a part of the higher-level task. Participants also provided action instructions more often than goal descriptions.

In our prior work [47], we analyzed situated teaching interactions to create an explicit teaching action taxonomy and proposed a theory-based dialogue system to support the taxonomy and teaching by non-experts. While this work allowed participants to use language flexibly to analyze their teaching interactions, our current work extends this by studying participants' teaching in a more structured setting while focusing on their process of planning and updating their model of the robot and the task itself through the think-aloud analysis. While all of these studies yield important insights that we observe in our study as well, these prior interactions occurred in Wizard-of-Oz settings whereas, in our study, non-expert participants interacted and provided instructions to the robot directly in a simulated environment.

Researchers have also studied strategies typically employed by teachers in a human-robot setting. These include studying how people teach robots singular concepts (e.g., graspability [23]), providing feedback [26], or providing reinforcement signals [58]. Thomaz and Breazeal [58] find that the person is able to provide appropriate guidance toward the next steps when the robot communicates specific aspects of its internal state. Our work differs from these efforts by focusing on end-to-end teaching of procedures using actions and goals. Dalvi et al. [14] present a teachable reasoning system whose erroneous model beliefs can be corrected by teachers through a question-answering type of interaction. However, it requires the teacher to both already understand the various reasoning mistakes the model can make, as well as learn the specific actions required to correct its knowledge which cannot be expected of a non-expert.

3.2 Ease of Interaction in Tutorial-based Systems

The teaching process in the Rosie agent is comparable to tutorial instruction, where the teacher provides a natural language description of procedures using general situations and abstract objects [17]. The teaching process in Rosie can be defined as a combination of a tutorial instruction process and situated instruction. Maclellan et al. support this proposition when describing tutorial instruction work in learning cognitive systems [35]. They describe Rosie as a direct instruction agent, where the direct instruction is slightly inverted since a lot of learning occurs through Rosie's leading questions and clarification mechanisms.

There are challenges that must be overcome when considering tutorial instruction in human-robot settings. Gil describes that tutorial instruction can be complex because procedures involve an abundance of interrelated information in terms of situating relevant objects in the environment, steps, and sub-steps of the procedures, as well as conveying control structures [17]. This contrasts with one-shot learning from demonstration or individual situated instructions, where a particular state or a contiguous set of actions is used to illustrate the procedure. Tutorial procedures often involve natural language descriptions of general situation information and abstract concepts and objects while using situated instruction and demonstration to supplement the teaching process.

TellMe [18] is one such system that learns through tutorial instruction and enables efficient and effective instruction through agent communication. The agent aids the instructor in teaching by reasoning over its knowledge and instructions and asking follow-up questions when necessary. However, it does not have any situated learning. Despite the obvious challenge in this space, no work systematically studies mechanisms a robot can employ to aid the teacher in a complex task teaching process in a situated setting. In this thesis, we aim to address this gap by analyzing situated teaching processes with non-experts and propose implementations that can aid the teacher in a human-robot teaching interaction.

3.3 Enabling Accessible and Efficient Robot Interaction

One of the ways through which teaching can be made more accessible is by enabling humans to interact with the robot through multi-modal interaction. Prior work in HRI has looked at enabling accessible interaction by allowing people to instruct robots using multi-modal interaction strategies such as combining language instructions with gestures [33, 58] or demonstrations [1, 11, 33, 43]. Some prior work in ITL has demonstrated efficient task learning and accessible interaction by enabling learning from single demonstrations and supporting language input rather than having people provide multiple demonstrations [1]. However, these methods are restricted to individual domains and are not generalizable across different types of tasks. Pumice [33] is an exception where the system does learn app-agnostic complex tasks and allows for multi-modal input from the human but is restricted to a smartphone setup. In this dissertation, the instructor interacts with the robot using language alone.

A person cannot effectively interact with a robot to successfully perform a task without a good mental model of the robot. A person's mental model includes task-specific expectations of the robot, including its knowledge and understanding of the given situation, which influences their interaction with the robot. Using and interacting with an intelligent agent alone may not improve the person's mental model of the agent [29]. Therefore, the robot must be able to use knowledge cues to correct the person's mental model if it is inaccurate [24]. Towards this, there has been a significant amount of research on transparency mechanisms that allow the person access to the knowledge and internal working of the robot to correct and update their mental model. This includes the robot providing information about itself in terms of its situational understanding and knowledge through language [9, 45, 55, 59], gaze [6, 45, 58], gestures [6, 11, 45] and visualization [32, 45], to elicit better input from a human collaborator. This process contributes to efficient interaction since the robot provides information about its reasoning and performance. However, this work focuses on the robot learning individual actions from humans or performing known tasks with a human rather than learning new generalizable complex task configurations.

Given the common occurrence of failures in interaction with robots, being able to debug the failure and recover from it is a crucial problem that needs to be solved. Implementing mechanisms that allow humans to debug the robot through direct access to its underlying mechanisms [20] or through robot explanation about its cause of failure [15] are effective in resolving and recovering from such failures. The robot can also ask questions to recover from failures or gain better instruction to proceed further in its task process. This can take the form of robot-initiated clarification questions that can be used to recover from interaction failures caused by ambiguity, misunderstanding, or non-understanding [37]. Including context-specific information in such clarification questions has also been shown to elicit more accurate responses from a human [49]. In previous research [48], we implemented visualization mechanisms in the Rosie agent to supplement the existing verbal modality. We conducted a user study to determine how people use these transparency mechanisms to identify the causes of interaction failures. We observed that access to visualization mechanisms did not result in a significant difference in the person's accuracy of identification of failures compared to verbal modalities alone. Our hypothesis is that when faced with ambiguous failures, the verbal modality provided a clear explanation that participants could use as opposed to having to interpret the visualization. Therefore, we restrict the robot's responses in this dissertation to the verbal modality alone.

3.4 Explainability in Intelligent Systems

Given the recent explosion of deep learning systems that are usually black box in nature, explainable AI has also gained popularity since researchers are motivated to make their systems understandable to humans. Tim Miller's literature analysis of the nature of explanations in human conversations provides the following findings that are relevant for explainability in AI systems [38]:

1. Explanations are contrastive: people are not as curious about why event P happened and instead want to learn why event P happened instead of event Q.

2. People do not expect a complete description of the cause and effect of an event. Humans select one or two explanations among many and this selection is influenced by cognitive biases.

3. Providing probabilities and statistical relationships is not as effective as providing causes as a part of explanations.

4. Explanations are social: they are typically presented relative to the explainer's beliefs about the explainee's beliefs.

Prior work in human-robot interaction has demonstrated the usefulness of these findings in enabling robots to provide proactive explanations. Thielstrom et al. [56] compare people's responses to various explanations that can be generated during a robot failure. They find that people find contrastive explanations more helpful to determine the solution to failures. Zhang et al. [61] find that when human interpretations of the robot's task model are incorporated during plan creation, plans are more explicable and predictable to humans. Prior work has also shown that people better understand robot behavior when provided explanations describing the causal chain of reasoning that contributed to the executed actions [7, 52].

In this dissertation, we study the robot teaching process to identify situations where the robot can provide relevant information about its knowledge and capabilities as well as task-related information that the instructor can use to interact more efficiently with the robot.

CHAPTER 4

Problem Definition

Throughout this dissertation, the terms **robot** and **agent** are used interchangeably to represent the ITL agent, and the terms **teacher** and **instructor** are used interchangeably to represent the human teacher.

4.1 Rosie and Cognitive Architectures

Here, we describe the ITL robot Rosie that is the basis for this dissertation in terms of its knowledge, abilities, and internal learning mechanisms that an instructor can leverage to teach the agent various tasks. Rosie [25, 39, 40] is an ITL robot that is implemented in the Soar cognitive architecture [30]. Rosie has been embodied in four robotic forms: a tabletop robot arm with a Kinect sensor, a mobile robot, the Fetch robot, and Cozmo. Rosie can learn over 60 games and puzzles [25], mobile delivery, navigation, and procedural tasks [39] and various procedural kitchen-specific tasks [42] using a combination of situated natural language commands and demonstration. Learning a task involves learning different aspects of the task, including goal and failure definitions, actions, concepts, state descriptions, and task-related if-then conditions. Soar includes long-term memories encompassing procedural, semantic, and episodic memory and short-term memory, which is working memory. The working memory contains information relevant to the task, including the agent's current state, goals, and beliefs about the world. Procedural memory stores rules - knowledge required to perform internal and external actions and respond to situations in the working memory to execute a task. Semantic memory stores long-term declarative knowledge, including facts about the world and representation of tasks, including goals, failure conditions, and possible actions. When the robot learns a task, it first builds a declarative representation of the task instructions and stores it in its semantic memory. The robot then retrieves, interprets, and executes these instructions to perform the requested aspect of the task. During interpretation, declarative knowledge is automatically compiled into procedural knowledge, which allows the robot to perform the same task more efficiently in the future without the need for interpretation.

In this thesis, we will only focus on the space of tasks that Rosie can learn in the simulated environment embodied as a mobile robot. Currently, an instructor can interact with Rosie to:

1. Teach a new task
   (a) Teach a subtask that belongs to the overall task
   (b) Teach a new action
2. Extend an existing task
   (a) Teach new sub-tasks within the same context or add general sub-tasks (adding more sub-tasks to the task)
   (b) Add context-specific sub-tasks
3. Give a command to execute a previously learned task

In this thesis, we assume that the robot is a *perfect learner*. This means that we do not focus on improving the robot's learning mechanisms and instead study and propose additional robot interaction capabilities towards improving the *interaction* between the robot and the teacher. Therefore, our research focuses only on the first type of interaction, namely, teaching a new task.

4.2 Teaching an ITL robot

We first outline the multiple aspects and steps that constitute a teaching task with the Rosie robot as follows:

1. This teaching interaction is situated in the environment with a mobile robot.
2. The teacher and the robot always interact one-to-one in the shared environment.
3. Teaching a task involves providing and teaching different aspects of the task, such as the task name, sub-tasks and steps, new actions, primitive actions, relevant concepts, objects, locations, state descriptions, and if-then conditions.
4. The human determines the relevant aspects of the task they will provide as a part of the teaching process.
5. The human also determines the order in which they will provide each aspect of the task.
6. The human leverages the robot's existing knowledge about the world and its innate actions (and eventually its learned knowledge from their interactions) to teach new tasks. They will not provide new innate knowledge to the robot.
7. The human teaches this through back-and-forth interaction in a step-by-step fashion.
   (a) The human provides one instruction at a time that contains limited information.
   (b) The human waits for the robot's response to the instruction or confirmation of learning before moving to the next command.
   (c) If the robot responds with a directed question, the human answers that in order to finish that turn and successfully finish teaching that command before moving ahead.
8. The human determines that the robot has successfully learned the task.

This list contributes a framework using which we can analyze the teaching processes of non-expert teachers and identify the specific points at which the robot can respond. For example, the different aspects of a task provided in point 3 help determine what information is relevant. As described in points 4-6, the human teacher is in charge of the content and mode of teaching. However, the robot's responses must help facilitate the teacher's instructions and make them more efficient. The delineated teaching process defined in point 7 enables equally spaced intervals and provides predetermined opportunities for the robot to provide relevant information.

4.3 Example of Expert Instruction

We now provide an example of a current teaching interaction with Rosie in a kitchen in the mobile-world environment. This is an example of an efficient interaction that an expert teacher can pursue because they have access to critical knowledge. The expert instructor leverages this knowledge and their prior teaching experience to construct efficient instructions. This interaction occurs in the mobile world domain in a simulated kitchen environment containing appliances and objects relevant to the teaching interaction. We will use this example to later describe a) the types of critical knowledge necessary to teach this task, and b) how the non-expert teacher's lack of access to this knowledge can lead to failures and inefficiencies.

4.3.1 Teaching Interaction Example

1. Teacher: Bake the pizza.
2. *Rosie: What is the next goal or subtask of bake?*
3. Teacher: Turn on the oven.
4. *Rosie: What is the next goal or subtask of bake?*
5. Teacher: Grab the pizza.
6. *Rosie: What is the next goal or subtask of grab?*
7. Teacher: Open the fridge.
8. *Rosie: What is the next goal or subtask of grab?*
9. Teacher: Pick up the pizza.

10. *Rosie: What is the next goal or subtask of grab?*

11. Teacher: Put the pizza on the table.

12. *Rosie: What is the next goal or subtask of grab?*

13. Teacher: Close the fridge.

14. *Rosie: What is the next goal or subtask of grab?*

15. Teacher: You are done.

16. *Rosie: What is the next goal or subtask of bake?*

17. Teacher: Open the oven.

18. *Rosie: What is the next goal or subtask of bake?*

19. Teacher: Put the pizza in the oven.

20. *Rosie: What is the next goal or subtask of bake?*

21. Teacher: Close the oven.

22. *Rosie: What is the next goal or subtask of bake?*

23. Teacher: Wait for thirty seconds.

24. *Rosie: What is the next goal or subtask of bake?*

25. Teacher: The goal is that the pizza is on the counter.

26. *Rosie: What is the next goal or subtask of bake?*

27. Teacher: Close the oven.

28. *Rosie: What is the next goal or subtask of bake?*

29. Teacher: Turn off the oven.

30. *Rosie: What is the next goal or subtask of bake?*

31. Teacher: You are done.

32. *Rosie: I am ready for a new task.*

4.3.2 Teaching Process

We now present the context for the above interaction while describing the different decisions that an expert instructor makes towards achieving the goal of this interaction, namely, teaching Rosie to *bake the pizza*. We also describe how the expert instructor leverages their knowledge of the robot to teach the task efficiently and effectively.

The interaction begins with the teacher instructing the robot to *Bake the pizza*. The agent does not know this task; therefore, it drives the interaction by requesting the instructor to either provide a goal or describe a process for the *Bake* process. In this instance, the teacher describes a procedure and provides consecutive steps for the agent to execute. For example, when the instructor asks the robot to *Turn on the oven* at step 3, the agent follows the instruction and first goes to the *oven* and then turns it on. The agent also learns this action as a part of the *Bake* task.

The agent can acquire knowledge about and learn new actions as sub-tasks of other tasks. At step 5, the teacher instructs the agent to *Grab the pizza* as a part of the *Bake* task. Since the agent does not know this action, it requests the goal or sub-task of the action *grab*. Once the teacher has provided the agent with steps to do the *grab* task at step 15, the teacher says *"You are done"*, which marks the end of the current teaching process and execution of the "grab" action.

The teacher can also provide goals as a part of the teaching procedure. At step 25, the teacher provides a goal instruction as a part of the *Bake* task. The robot internally performs a means-end analysis to determine the steps needed to achieve this goal. Since the robot knows that the pizza is in the oven at this time, it will first open the oven before picking it up. It will then go to the counter and then put the pizza on the counter, thus achieving the goal state.

At step 29, the instructor says *"You are done"* and is finished teaching the *Bake the pizza* task.

4.3.3 Types of Robot Knowledge Relevant for Teaching

In the task teaching example, we highlighted some of the learning capabilities of the robot that the expert instructor leverages during the instruction. We categorize this critical knowledge relevant to the teaching process into three types described below. We provide all three types for the sake of completeness. In this thesis, our research will focus only on methods related to the **robot's task knowledge** and not on the robot's learning strategy or instruction capabilities.

1. **Robot's Task Knowledge**: The robot's task knowledge comprises the following:

    - Knowledge about the world (objects, properties and locations of these objects, rooms, and spatial knowledge to navigate local and extended situations)
    - Primitive actions that can be used to teach more complex tasks.
    - Previously learned knowledge about tasks in terms of sub-tasks, primitive actions, and goals.
    - Robot's progress on executing a task or sub-task.

2. **Robot's Inbuilt Learning Strategy**: The robot has an inbuilt strategy for processing instructions, focusing on learning efficiently through instruction (how does the robot learn?). This learning strategy influences how the robot applies this knowledge (transfers this knowledge) when provided with similar instructions in a different context. These strategies include:

    - how it processes the knowledge present in the instruction in terms of generalization or specificity of the applicable knowledge, and
    - how it individually learns each sub-task and links each sub-task when they contribute to an overarching task.

3. **Instruction Taking Capabilities**: Rosie has the following learning capabilities that it leverages to drive the interaction and further its learning process.

    - Rosie will ask for the goal/first step of a verb and/or task it has not encountered before.

- For an entity (person, object, etc.) referenced in an instruction, Rosie will first try to find it in the same location. If it can't find the entity, it will first perform an episodic memory retrieval cued on the last time it was in that location to retrieve the last known position of the entity. If Rosie has never encountered the entity before, Rosie will ask for the location of the person/object.
- Rosie can ask if it does not know a grounding for a referring expression, such as, "Who is my relieving officer?"
- It can ask "What do I do next?", "What is the goal?" - it can further the interaction.
- It can answer questions about its knowledge and environment and knowledge application in the environment.

4.4 Challenges for a Non-Expert Instructor

Given the overarching goal of Interactive Task Learning is to leverage people's natural teaching abilities, successful robot teaching should not rely on people to have expert knowledge. A typical instructor will likely be a non-expert who does not have a good mental model [44] of the robot i.e., they will not possess detailed knowledge of the robot's knowledge, capabilities and learning abilities, unlike an expert instructor.

This means that the instructor may not have the requisite knowledge to efficiently and effectively teach a robot successfully since they cannot make the same assumptions that an expert instructor would make. We hypothesize that not having access to the relevant information during a teaching interaction can result in instruction-level issues that cause local failures and inefficiencies in the teaching process at best, and inability to finish the teaching process at worst.

Therefore, our goal in this dissertation is to understand ways in which non-expert instruction differs from expert instruction by studying how non-experts teach a task to a robot that has only been taught by experts before. We conducted a study where non-experts teach a new task to the ITL robot Rosie using the setup described in Section 4.2. We analyze the teaching processes and identify key cases where people lack knowledge through the task analyses, so that we may develop interaction mechanisms in the robot that provide relevant information to facilitate non-expert instruction. Also, we analyze situations where this lack of knowledge results in inefficient and failure scenarios to propose mechanisms that will allow non-experts to effectively handle such situations.

CHAPTER 5

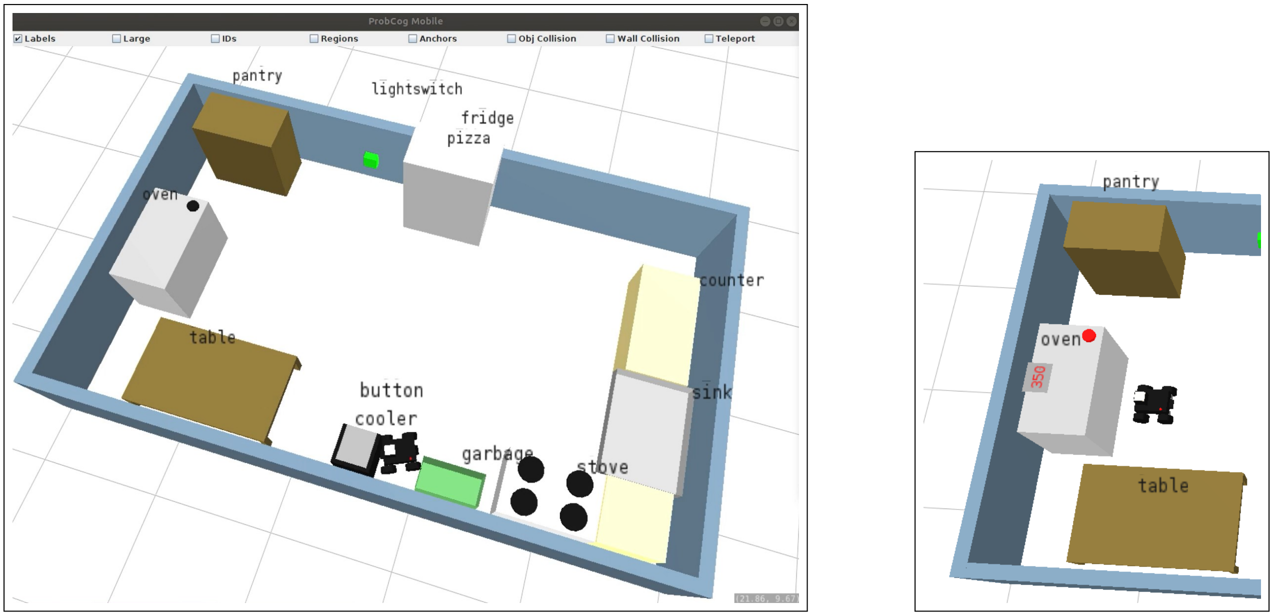
Task Environment and Interface Design

To support situated teaching interaction, the robot needs to operate in a physical world so that the instructor can use references to objects and their relationships in the world to provide instructions. In this dissertation, we leverage an existing mobile simulated task environment that was developed for the Rosie Robot to learn hierarchical tasks. We also developed a templated instruction interface that an instructor can use to provide instructions that follow a predefined format. We introduce template instructions in this dissertation specifically to make the instruction process easier for the instructor and to avoid language comprehension errors while interacting with the robot.

5.1 Task Environment

The task environment used in this thesis is a 2.5-dimensional simulated environment based on the April Robotics Toolkit[1] and was developed for Aaron Mininger's thesis [39] to demonstrate Rosie's expansive task learning abilities. This simulated environment based in the real world enables natural teaching since it is more familiar to a non-expert participant while avoiding physical and mechanical issues that occur with a mechanical robot. Using a simulated environment also contributes to replicability that we can leverage when conducting iterative studies to study human teaching behavior. The robot in this environment is capable of moving objects and going from one location to another which contributes to easier teaching as a result of observable feedback.

[1] https://april.eecs.umich.edu/software/java.html



(a)

(b)

Figure 5.1: (a) Simulated environment for Rosie embodied in mobile robot (b) Partial environment with a turned-on oven

To avoid significant development work, I extended this existing environment to create an indoor kitchen space (Figure 5.1a) for a baking task (described in section 6.2) that is used in the study. The black figure in the center is the simulated robot, Rosie. There are labels attached to all the objects that Rosie could operate or handle. The list of objects includes a counter, a sink, a stove, a water cooler with a button on it, a garbage bin, a table, an oven, a pantry, a fridge with a pizza inside it, and a light switch. Each object has individual affordances and categories that we provide in appendix B. However, when the oven was turned on (Figure 5.1b), the display was modified to indicate that it was on (with a red circle denoting the turn-on light) and already at 350 degrees, as required for the study task.

5.2 Template Interaction Interface

We introduce a template interaction interface (Figure 5.3) that an instructor can use to provide instructions to the robot. This replaces the free-form chat interface depicted in Figure 5.2. In prior work [47], we observed that people use many different sentence forms to provide similar instructions to the robot. Ideally, the robot should be able to extract the relevant information from different types of instructions towards its learning process. While Rosie can process many types of fairly complex instructions, it is not flexible enough to comprehend free-form instructions that do not follow a predefined format.

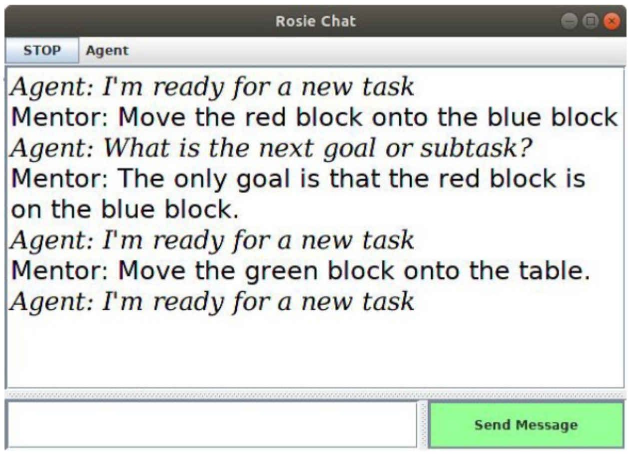


Figure 5.2: Original Rosie Chat Interface. (Adapted from Mininger, 2021)

Since the focus of our research is on understanding the transfer of knowledge between the instructor and the robot, we decided to use template instructions that are sufficient for all commands but also restrict the instructions to be processable by the robot. Studies in linguistics have shown that humans engage in effective dialogue through automatic alignment of their linguistic representations [46]. In addition to Rosie directing its learning process by responding with appropriate dialogues and questions, the templates improve robustness by contributing to the predictability of robot linguistic interaction. The templates were designed to allow for four instruction formats - two for actions and two for goal descriptions. The templates are as follows.

- Action: Verb + *Object* (e.g. Empty the *garbage*)
- Action: Verb + *Object1* + Preposition + *Object2* (e.g. Fetch the *coffee* from the *kitchen*)
- Goal: The goal is that *Object* + Linking-Verb + Object State (e.g. The goal is that the *garbage* is empty)

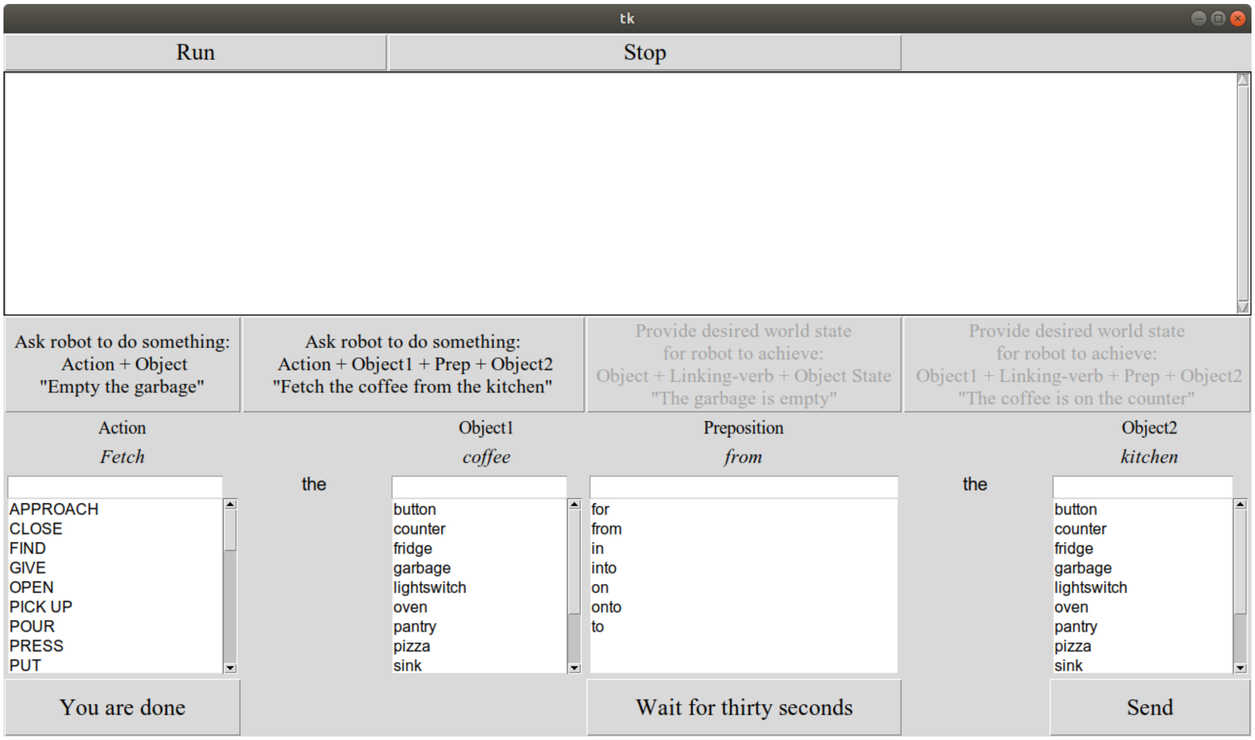


Figure 5.3: Second Action Template Interface

- Goal: The goal is that *Object1* + Linking-Verb + Preposition + *Object2* (e.g. The goal is that the *coffee* is on the *counter*)

We chose these four templates since they provide adequate flexibility for the instructor to provide the instructions required for object interactions. For each part of speech, we provided a drop-down list of available valid words. For the list of objects and object states, we restricted the list to only the objects in the experiment environment and their corresponding possible states.

The template interface is divided into four rows. On the first row, there are two buttons *Run* and *Stop* to start and stop the robot. The second row comprises the text area where the instructor can see the instruction they provided as well as the corresponding robot responses. The third row contains the buttons to enable each of the four templates. On pressing a specific template button, the fourth row gets updated with the corresponding entries and drop-down lists. The goal template buttons are disabled by default. This is because the robot can process a goal instruction only after it requests one when learning a new action using the prompt "*What is the next goal or subtask of*

Action1?" at which time the goal template buttons are enabled. An example of the second type of action template can be seen in Figure 5.3. Each part of speech has a corresponding text entry and a drop-down list of valid values. The text entry can be used to enter values and they will be auto-completed using only the values in the list. The remaining template information along with the complete list of values in the drop-down list can be found in appendix B.

Two additional instructions were provided in the form of buttons, namely, "*Wait for thirty seconds*" and "*You are done*." We included these instructions as buttons since they did not fit the existing templates. However, they were necessary instructions required to teach the experimental task. These can be seen in the fifth row of the interface along with the *Send* button that is used to send the selected instruction to the robot.

5.2.1 Robot Action Knowledge

In the verb drop-down list, we included primitive actions that the robot already knows (which were displayed in capital letters) as well as teachable actions it does not know but can learn (which were displayed in lowercase letters). The teachable actions consisted of a limited set of verbs (e.g., bake, heat, cook, etc.) that were related to the study task described in chapter 6. If the instructor used a teachable action as a part of the instruction such as, "`Grab the pizza`," the robot would respond with "What is the next goal or subtask of `grab`?" The instructor can then provide corresponding actions and goals using all four templates. Once the instructor determines that they are done teaching the `grab` task, they can press the "`You are done`" button and continue the overall teaching process if this was a sub-task or proceed to teach other tasks.

5.2.2 Robot Part of Speech Knowledge

For the remaining parts of speech, we only included words that were related to and could be used in the simulated environment presented to the participants. For the object drop-down list, we only included the objects that were present in the environment. For the object state drop-down list in the first goal template, we included only the object states that applied to the objects present in the environment. For example, the oven and the stove could be turned on and off, and both the fridge and the oven could be open or closed. Therefore, the object state drop-down included these four states. Similarly, the preposition and the linking-verb drop-down list were populated only with the values that contribute to instructions processable by the robot.

5.2.3 Robot Environment and Interaction Knowledge

To successfully interact with the robot, the instructor needs to know the following about the robot as well as its interaction capabilities. For the study described in chapter 6, we directly provided the participants with the following information.

1. The robot has complete knowledge of the environment.
2. The instructor cannot directly interact with any object using the mouse. Instead, the instructor must ask the robot to perform actions to interact with objects.
3. The robot can handle only one object at a time.
4. The instructor must provide step-by-step instructions and wait for the robot to respond after each instruction before providing the next instruction.
5. The difference between primitive actions and teachable actions as described in section 5.2.1.

5.3 Summary

In this chapter, we described the setup used for the study to enable an instructor to interact with the ITL robot. The environment enables the instructor to observe the robot's manipulation of the environment and its actions and movements in response to their instructions. The templates serve as an initial guide about the types of instructions that the robot can comprehend. In addition, the values in the drop-down lists serve as a way for the instructor to determine the robot's task knowledge. The instructor can thus, leverage the values already available in these lists to form instructions relevant to the task, rather than creating instructions from scratch.

CHAPTER 6

Study Design

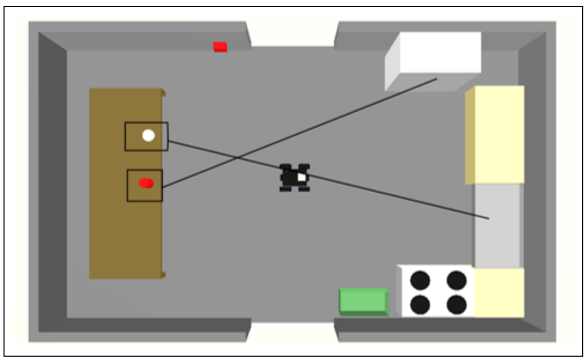
We conducted a think-aloud study [12, 19] to understand *how non-experts teach new tasks to a robot*. We then conducted a qualitative analysis of the multidimensional data available of the participants' direct interaction with the robot with the following research questions in focus.

1. How do non-experts teach hierarchical tasks to an ITL robot?
2. Do instruction templates contribute to efficient and effective teaching?
3. What are some salient problems that non-experts face that can be solved by extensions to the robot's interaction capabilities?

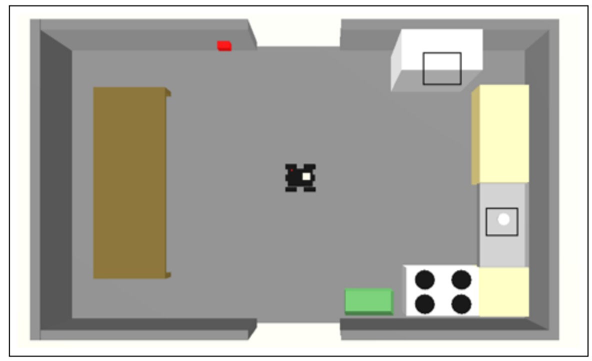
Using this analysis in Chapter 7, we propose robot mechanisms that could potentially help improve robot task instruction by humans in chapter 8.

6.1 Pilot Study

We conducted a pilot study to test out the environment setup using an initial template implementation and two simple pick-and-place tasks: `Move the plate into the sink` and `Move the ketchup into the fridge`. We did not provide these commands to the participants. Instead, we provided them an initial state and final state as depicted in Figure 6.1 so that they had to generate appropriate commands for the robot. For the pilot study, we provided a wider array of options in the template commands to provide more flexibility during teaching. However, participants ran into language-related errors that the robot could not resolve in real-time. Since our research questions do not include language mismatch issues, we modified the template by restricting the provided options to those that were relevant to the tasks and objects in the environment. This also led us to provide labels for objects in the environment so that the participant's focus could be on the teaching process alone. The pilot study also allowed us to determine and add relevant descriptions to the template buttons to make it easier for the participant to determine how to use each template.



(a) Initial environment state of pilot training task



(b) Final environment state of pilot training task

Figure 6.1: For the pilot study, participant provided commands to the robot to get from the initial environment state to the final environment state

With two participants, we provided a training session where they were given step-by-step instructions on how to teach Rosie a teachable action using a goal command as described below.

1. Instructor: Pick up the plate (Primitive action)

2. Robot: I am ready for a new task.

3. Instructor: Move the plate into the sink.

4. Robot: What is the goal or next subtask of Move?

5. Instructor: The goal is that the plate is in the sink. (Robot puts the plate into the sink)

For the main study, we decided not to include this specific training session, as it actively familiarized the participant with the robot's learning mechanisms in advance of the actual teaching task. Instead, we provided text instructions in the study document to ensure that the participant was aware of the distinction between primitive and teachable actions without the knowledge actively disrupting the study setup. Since all participants were able to successfully provide instructions for these two simple tasks, we decided to pursue a more complex task for the main study that we describe in the next section.

6.2 Final Study Method

The university's internal review board approved the study materials and protocol as *Exempt*. Before each study, I, as the researcher, executed a set of steps to set up the environment to be uniform across all participants. This included generating a random number for each participant (to anonymize their data), recording their instruction interaction, setting up the desktop to show the simulated environment, instruction interface, and instruction document in a preconfigured manner, and recording the screen during the teaching process.

When the participant arrived, the researcher went over the consent form. Once verbal consent was obtained, the participant filled out a pre-study questionnaire that asked for their demographic information (age and gender), relevant background knowledge about teaching interactions with people and intelligent agents, and oven use. We used the guidelines in [51] to ask about gender. We provided non-overlapping ranges as well as the option to not disclose age information. We provided participants with specific options to collect information about education, interaction experience with intelligent agents, frequency of and reasons for oven use, and if they had taught oven tasks to another person or a child. To understand the effect of prior experience with intelligent agents or programming, we asked participants to describe their interaction with intelligent agents, if any, as well as their background and if it involved programming, computer science, or robotics.

Once this was completed, the researcher reviewed a study instruction document with the participants. This document provided information about the robot, the instruction templates, the difference between primitive and teachable actions, and the additional command buttons on the instruction interface. While the participant did not undergo a training session to provide a teachable action instruction, the document provided information on the interaction that would ensue as a result of providing a teachable instruction. The researcher emphasized to the participant that the study was designed to test the robot and not the person, and therefore, any failures that they might encounter would be a result of robot deficiency. This document was available to the participants at all times if they wanted to reference it during the study. This study document is provided in appendix A.

The researcher then provided the participants with details about the think-aloud protocol. After verbally describing the think-aloud protocol to the participant, the researcher asked the participant to watch a short one-minute-long pre-recorded video[1] demonstrating the think-aloud method. This video displays a person exploring a website while they think out loud throughout the exploration process. After this, the interviewer reiterated the salient parts of the video relevant to the experiment and clarified any questions that the participant had about the think-aloud process.

Next, the participant began interacting with the robot through training tasks. The training tasks were focused on familiarizing the participants with the instruction templates alone, without emphasis on the robot's learning mechanisms. This is because our goal was to understand if and how non-experts can teach the robot without extensive training using the robot's existing interaction alone. The researcher asked participants to perform three training tasks described in the document. To minimize "cold-start" effects, participants were asked to begin verbalizing their thoughts starting from the training tasks. If the participant was silent for a long time, the researcher prompted them with the question "Can you please share what you are thinking right now?" For the first two training tasks, we provided them valid instructions for the two action templates that they could select from the drop-down lists for each part of speech and then provided instructions to send this command to the robot. For the third training task, we asked them to press the button with the command "Wait for thirty seconds." We included the third training task because it relates to the main task. However, we did not provide any context during the training task to avoid influencing the participant during the main task. After the training tasks were done, the interviewer encouraged participants to continue

[1]https://www.nngroup.com/articles/thinking-aloud-demo-video/

the think-aloud process for the main task while sometimes pointing out successful examples from their think-aloud process during the training task.

For the main task, we asked the participants to teach the robot to bake a pizza. Towards this, we provided them with the following task description - *The robot must cook (350 degrees for 30 seconds) a pizza using the oven, and the cooked pizza must end up on the counter.* This task was chosen because it does not require any specialized knowledge and is easy to understand, but it also has adequate hierarchical structure and complexity to explore natural teaching processes.

The researcher did not intervene to provide more information during the training task, except to tell the person they could move on to the next task when the robot said, "I am ready for a new task." With a few participants, the researcher reminded them about information that was already available in the study document, including the difference between primitive and teachable actions, the relation of "you are done" to teachable actions, and that the participant was not being tested in this experiment.

Once the task was done, the researcher asked any clarifying questions they had based on their observation of the study to the participants. After this, the participants were asked to fill out a Usability Metric for User Experience (UMUX) [16] Likert scale (range 1-5) to provide their subjective assessment of the robot's perceived usability, or in this case, instructability. After participants provided information for payment, the researcher then answered any questions that the participants had about the study.

6.3 Participant Recruitment

We recruited participants via our university's participant pool portal, where potential participants can express their interest in participating in a given research study. We recruited adult participants over the age of 18 and located within walking or driving distance of the university location since the study took place in person. Participants were paid $15, and the study was capped at one hour though all participants finished the task in that time frame.

We recruited participants for our study until we reached data saturation resulting in a total of 14 participants (8 female, 6 male, 1 transgender (self-described)). Their age range distribution was: 18-24: 2, 25-34: 8, 35-44: 3, 45-54: 1, 55-64: 1. Table 6.1 lists participant demographics and task-related information from our pre-study survey.

We performed a qualitative and quantitative analysis of the multidimensional data collected from this study. We describe these analyses and our findings in the next chapter.

ID	Education	Interaction with AI agents	Description	Background	Oven Use Frequency	Oven Use Tasks	Instructed person to do oven tasks?	Description
P56	Doctoral Degree	None	N/A	Computer Vision	Never	N/A	No	N/A
P92	Master's degree	Siri, Alexa	Ask about weather, play music	CS, AI	Once a week	Roasting, Heating, Baking, Broiling	No	N/A
P26	College degree	Other	Google Maps for navigation	No CS or robotics	Daily	Roasting, Heating, Baking, Broiling	No	N/A
P67	Some college	None	N/A	Music, performing arts No CS or robotics	Once a week	Heating, Baking, Broiling	Yes	Brief verbal checklist
P23	College degree	Siri, Alexa, Google Now	Limited voice based commands	Electrical Engineering Limited programming	Less Often	Baking	No	N/A
P49	College degree	Siri, Alexa, Google Now	Asking for directions, cooking, weather	Anthropology and Bio No CS or robotics	Daily	Baking, Grilling	No	N/A
P31	Master's degree	Siri	Providing demonstrations on UI	CS, HCI	Once a week	Roasting	No	N/A
P47	Master's degree	Siri	Check weather, set timer, play music	No CS or robotics	Most days	Roasting, Heating, Baking	No	N/A
P78	Master's degree	None	N/A	Teaching, pet sitting No CS or robotics	Once a week	Roasting, Baking	No	N/A
P7	College degree	Siri, Alexa, Google Now	Check weather, set timer, play music	Philosophy, Spanish No CS or robotics	Once a week	Roasing, Heating, Baking, Broiling, Grilling	Yes	Walking through steps in chronological order
P59	College degree	None	N/A	SQL Programming No robotics	Daily	Roasting, Heating, Baking, Broiling	Yes	Teach kids how to cook, kitchen safety
P74	Doctoral degree	None	N/A	Engineering, some background in scripting languages	Once a week	Roasting, Baking, Broiling	Yes	Teach kids to bake
P11	College degree	Siri	Ask basic questions to be googled	Accounting No CS or robotics	Once a week	Baking	Yes	Set temperature, check food when timer goes off
P99	Master's degree	Siri, Alexa	Ask for recommendations, play music	Computer Scientist	Once a week	Roasting, Baking	Yes	Teach turn on oven, oven safety

Table 6.1: Participant Demographic and Task Background Information

CHAPTER 7

Analysis

In this chapter, we describe the analyses performed using the data collected from the study. We first provide an overarching summary of the results from the qualitative and quantitative analyses in Section 7.1. We then describe the inductive thematic analysis process in Section 7.2, followed by our findings about the participant teaching process and their interpretations of the robot from Section 7.3 to Section 7.8 We first identify six types of knowledge that were available to learn and use during the study in Section 7.3. We situate our remaining findings using these types of knowledge. We describe how participants used this knowledge, identify instances where participants did not have sufficient knowledge to proceed confidently through their teaching process, and describe how they found alternative ways to continue teaching the task. We also make note of the participants' interpretations of the robot through analysis of their teaching interaction.

We then present our quantitative findings in Section 7.9. We provide a task command analysis in Section 7.9.1 where we characterize participant teaching processes using the various commands provided by the participants. We also describe the various types of failures encountered by participants in Section 7.9.2 and provide summary statistics in Table 7.2. In the end, we provide the results from a usability survey filled out by participants at the end of the study.

7.1 Summary of Analysis

13 out of 14 participants successfully taught the task to the robot. Participants could use both teachable and primitive actions along with goal statements to teach the robot. We observed that

9 participants used teachable actions, and six participants used goal statements as a part of their teaching process. The number of turns taken to teach the complete task ranged from a minimum of 29 turns to a maximum of 123 turns.

12 out of 14 participants faced some type of failure during the teaching process. Out of these 12 participants, 8 faced failures where the robot reached an irrecoverable state that required the robot to be restarted. After the restart, the researcher reset the instruction and environment state to the last successful state before the failure so that the participant could continue the teaching process from where they left off. These failures included low-level task failures, where the robot could not perform an action as well as language-related failures, such as when the robot could not understand a command. However, all participants continued teaching the robot even after encountering these failures.

We now summarize our findings from our qualitative data analysis. As a part of the study setup, multiple sources of knowledge were available to the participants to use during the study. Participants began the interaction by developing complex strategies to teach the baking task to the robot. In order to teach the task to the robot, the participants actively gathered knowledge about different types of knowledge through exploration and interaction and leveraged it to aid their instruction process.

However, at times, participants did not find this knowledge sufficient for their teaching process, which led to incorrect and incomplete mental models. Participants could not extract all the available information from the interface and found the updates in the environment in response to their commands insufficient. This led to participants not having adequate information to confirm the success of the robot's action in the environment. Overall, as a result of the lack of adequate knowledge, participants did not have consistently reliable mental models of the robot.

Naturally, these incomplete mental models resulted in failures and inefficient scenarios that led participants to experience learned uncertainty about the task-teaching process in terms of their ability to teach as well as the robot's ability to learn the task. Participants were, therefore unable to efficiently teach the robot.

Despite facing these issues, participants were motivated to continue teaching the robot. When the robot ran into inefficient or failure scenarios, participants suggested ways to recover from these situations. Additionally, when they could not find a way to provide their desired instruction, they found alternative ways to provide similar instruction.

7.2 Qualitative Analysis Process

We transcribed audio recordings of the collected data using the *rev*[1] transcription service in preparation for the inductive thematic analysis [4]. We then open-coded the think-aloud data along with the video recordings of the interaction and the memo written by the researcher during the study for all 14 participants and generated 340 open codes. These open codes pertained to the participant's thinking process about the task itself, along with their interaction with the interface (e.g., button clicks, text entries) and environment, and the interactions between the participant and the researcher. Through iterative discussions with some of the thesis committee members, we created inclusion/exclusion criteria for the open codes and merged individual codes. After this process, we were left with 279 codes that we used to perform axial coding, a process where different codes were clustered together to identify themes. We used affinity diagramming on a *Miro*[2] board to cluster open codes to identify categories and clustered these categories to identify themes.

7.3 Types of Knowledge Available During Teaching Process

Participants had access to different types of knowledge to teach the robot. There were some types of knowledge that they already possessed, whereas others that they learned through exploration and as a result of their teaching interaction. We identify and describe these types of knowledge below.

1. **Task Knowledge**: Participants came in with prior task knowledge that they brought to bear during the teaching task (presented in Table 6.1). This comprised knowledge about the task,

[1]https://www.rev.com
[2]https://miro.com

including baking or heating food items using the oven.

2. **Instruction Knowledge**: 6 participants had prior experience teaching baking-related tasks to others that they could leverage to teach the task in our study. Participants with experience interacting with virtual AI agents could leverage their experience to provide simple instructions to the robot.

3. **Interface Knowledge**: Participants could update their knowledge of the interface by using different parts of the interface to provide instructions to the robot. The interface was comprised of information about the instruction templates available to use, their enable/disable conditions, the autocomplete text input, and the options available to be selected in these templates, as well as the extraneous commands available in the form of buttons. The interface also contained a text area that provided a history of the interaction.

4. **Environment Knowledge**: Participants could update their knowledge about the task-specific kitchen space in terms of the objects, their visible affordances, and their locations to incorporate into their teaching process. Participants also used the options available in the interface to update their knowledge about the environment.

5. **Robot Knowledge**: Participants could update their mental model of the robot in terms of its environment and task knowledge, interaction expectations, learning process, and task progress through various sources that we describe below.

    (a) **Interface**: Participants could use the options available to determine the different types of instructions and input the robot could or could not process in this setup. This included the types of actions (primitive and teachable actions) as well as the types of values (among object and object states) available to incorporate in a command.

    (b) **Interface Text Area**: The participant could use the robot's varying responses in the interface text area to determine the success or failure of their last instruction. They could also use these responses to determine the robot's learning process while teaching a larger task.

(c) **Changes in The Environment**: Participants could determine the robot's ability to perform a particular action or achieve a goal state through visual changes in the environment (e.g., the robot picks up an object and it appears on top of it). The robot's movement in response to the instruction also provided relevant information to the participant.

6. **Current State Knowledge**: Participants could use a combination of the previously described knowledge sources to assess task progress and determine the current situation in the middle of their teaching process. Participants could use a combination of the history of robot responses in the text area along with the history of its actions and movements in the environment.

7.4 Complex strategies were developed to teach the robot

On being introduced to the actual task, the participant usually began by combining their recent experience of using the templates in the training task and the information in the environment to develop strategies to teach the defined task to the robot. Based on their prior real-world experiences, they broke down the task into sub-tasks or individual commands and used the various feedback mechanisms to contribute to their developing strategies. Some participants checked whether there was a time limit for the task in order to incorporate that into their planning process. The researcher informed them that the task process was not evaluated on the time taken.

7.4.1 Participants decomposed the task teaching process as a part of developing their plan

Once the main task was introduced to the participants, they compared it to the individual training instructions and noted that the actual task is ".. *a more complex task, has multiple steps, and I'm guessing I'll have to do it in multiple steps*" (P31). Then, most of them (12 participants) organically decomposed the task into smaller tasks. P56 planned out loud by deciding that they "*can split this* *into two subtasks. So first one is cooking a pizza using oven at a specific degree, at a specific time period. So, and the second one is the cooked pizza must end up on the counter.*" Seven participants also listed the steps that needed to be executed to complete teaching the task.

> "*So I need to turn the oven on probably at some point. Need to grab the pizza from the fridge. Put it in the oven. Wait 30 seconds and then take it out of the oven.*" (P99)

> "*I'm just saying what I'm thinking about doing- I'm gonna need Rosie to preheat the oven. So, I'm gonna have her go over to the oven first and turn on the oven and it says for 30 seconds, okay, so I know how to do that. Once that's happened, then I'm gonna have Rosie retrieve the pizza, it looks like from the fridge. And then we'll need to place it in the oven. We'll have to determine how long to wait for the pizza in the oven, maybe another 30 seconds. And then I need Rosie to somehow get the pizza to the counter.*" (P59)

Three participants also referred to using goals in addition to actions in order to teach the task. P31 said, "*I guess I'll start with this option here, but maybe I'll later on change it to a goal if I can't get this to work.*"

7.5 Different types of knowledge were used to aid instruction

Participants explored the various knowledge sources defined in Section 7.3 to gain information that they then used to plan the next steps in their teaching process. They then continued to leverage this knowledge to adapt their own teaching behavior over time.

7.5.1 Participants leveraged different types of knowledge to plan next steps

In addition to using their prior knowledge about the task, participants gathered knowledge through the interface, environment, and the robot's behavior to plan their next teaching steps.

7.5.1.1 Participant assigned real-world task expectations to the task at hand

Participants (3 participants) who frequently used the oven tended to assume that the stove and the oven were interchangeable. P23 began the task with "*It just says cook. So I'm just then gonna take it to the stove.*" This is particularly telling since the stove and the oven were in two different locations. P74 first planned to start by "*...turning on the stove, I guess...*" but on observing the environment closely, realized that "*... the stove and the oven are not in the same place. I didn't notice that.*"

Six participants planned to "*preheat the oven so the pizza can be baked*" (P7) and "*figure out how to get it to the right temperature*" (P47) since the task required the oven to be set at a predefined temperature. P92 also expected to "*wait for it if it is preheating.*" Even though in our task, the pizza was in the fridge, P78 expected to "*go get the pizza out of the freezer.*"

7.5.1.2 Participants used knowledge from the interface to plan next steps

Participants had to use the template interface provided to them to instruct the robot. Naturally, they (5 participants) first decided to scroll through the drop-down lists to see the options available to them. P92 said, "*I'm just gonna look through the possible commands just to familiarize myself with the different options,*" and scrolled through the action drop-down list. Eleven participants also read through the action list with the explicit intention of finding an existing relevant option to teach the task. P49 said, "*Okay, so the pizza is in the oven. I closed it and now I got to...I'm just looking at the choices now to see if anything helps.*"

Additionally, two participants took advantage of the template structure and the drop-down list options to determine their teaching plan. For example, as a part of their planning process, P49 attempted to create a command by entering values in different templates but not sending them immediately.

"*And then, I'm gonna put pizza.*" (P49 selects options on the first action template)

"*Does that make sense? Put pizza?*"

"*No, I have to use this one maybe,*" (P49 switches from the first to the second action template)

"*Wait, let's see. Put pizza in oven.*" (P49 selects options on the second action template)

Participants (7 participants) sometimes already knew what action they wanted to do and scrolled through the drop-down list or used the auto-complete function on the text area to look for their desired option. While doing the training task `Approach the lightswitch,` P23 searched for both the verb and the object using auto-complete ("*let me try searching here*") and on finding it successfully, said, "*Okay, this seems good enough.*"

Three participants also used the auto-complete feature to look for their desired option when they unintentionally skipped over the desired option on the list, or the desired option did not exist on the list. P92 said, "*I guess the first thing, do I need to preheat the oven?*" and scrolled through the action list. When they could not find it, they switched to entering values in the text area to auto-complete - "*Let's search.*" On not finding it, "*No, it does not look like that (exists).*"

7.5.1.3 Participants used robot behavior and responses to plan next steps

Participants actively observed and incorporated the robot's actions, movements, and verbal responses into their planning process.

The study instructions asked participants to provide one instruction at a time and wait for the robot's response before providing the next instruction. P67 began selecting the instruction "*So I will have it put down...*" and then realized that "*Oh, it's not ready for me yet*" since the robot was still doing the previous action and had not responded indicating it was ready for the next instruction.

On being instructed on a new unknown teachable action, the robot responded with "What is the next goal or subtask?" to learn the different aspects of the teachable action. 8 participants planned in response to this request from the robot. P11 decided to use the instruction `Bring the pizza to the oven` to which the robot responded "What is the next goal or subtask of bring?" P11 said, "*So I think this is where I want to tell the robot I want the pizza in the oven and that's the goal of bringing it to the oven.*"

8 participants waited to visually confirm that the previous instruction was successfully executed

before planning the next instruction. P92 instructed the robot to `Put the pizza in the oven`. Once it performed it successfully, P92 observed that - "*Oh, it's in there. Okay.*" and then proceeded to plan the next steps - "*So I should probably close the open door.*"

Four participants used the previous instruction log to decide their next steps. P49 read through their previous instructions ("*Okay, fetch the pizza, cook the pizza. Approach the ... pizza*") and determined that they ".. *did turn on the oven*" and decided their next instruction based on that. Participant P7 also kept track of prior robot movements to decide what specific instruction to provide next. P7 said, "*Assuming I don't need to move just because I haven't moved, moved between the oven and the table at this point. So we'll just go straight to picking up the pizza.*"

7.5.1.4 Participant used current state information to plan next steps

Participants often used the current state of the robot and the environment, as well as the stage of the task and last interaction, to inform their next steps.

As a part of this, they would note the current stage - "*Okay, let's see, we already cooked the pizza*" (P23) - and then interpret the current situation to decide the next steps. For example, P74 decided to "*Put the pizza on, or onto, the counter.*" as a part of the required task, and having done it, determined that "*we probably ought to go back and close the oven and turn it off.*"

Seven participants used information available in the environment alone to determine their next steps. P67 said, "*And I'm just now noticing the fridge door is still open, so we better head over responsibly*" and proceeded to provide the commands `Approach the fridge` and `Close the fridge`. Two participants accounted for both the locations of objects and the proximity of the robot to varying objects to plan their steps. P47 noted the location of the pizza and used it to determine their next instruction - "*Okay, so I see that the pizza is in the fridge, so I think first I will tell the robot to approach the fridge.*" While P11 was teaching the task `Heat the pizza`, they decided that they could "*go back to turn on the oven since the robot's there.*"

At the end, six participants assessed the current state after teaching the task successfully to determine that they were done teaching the task.

"*So it seems that it is done. It is reheated for 30 seconds and on the counter.*" (P92)

7.5.2 Participants updated their knowledge as a part of the teaching process

As the participants continued to provide consecutive instructions, we observed that they actively interacted with the robot to update their knowledge through exploration of the interface and observation of the robot and the environment.

7.5.2.1 Participants explored the interface to update their mental model of the interface

While the training session provided the participants with an initial experience of using the interface, participants used the teaching process to explore it further to use it better over time.

Among the four templates available, the action templates were always activated, whereas the goal templates were only activated when a new task was being taught. P56 noted that - "*So I'm not allowed to use the third fourth template here. Just one, two.*"

When four participants clicked on the second action template, they noted its difference while comparing it to the first action template. P11 noted that "*this is different. It's more complicated.*"

As a part of their exploration, two participants also noted different properties of the options available across the different drop-down lists. P23 noted "*it's alphabetically sorted*" while referring to the order of options, and P74 noted that there is no ambiguity between the available prepositions. Four participants also made observations about the values present in the list, such as P26 noting that "*I guess there's no preheat and there's no, like, timer for it.*" P92 discovered that they could type in values to be auto-completed and said, "*.. can type here. I got that.*"

7.5.2.2 Participants used the interface to update their mental model of the task process

Once participants got familiar with the interface through natural exploration, participants leveraged the various aspects of the interface to update their understanding of the task teaching process.

P47 noted that there was a similarity across the two training tasks, referring to the similarity in the process of selecting options for the two commands. Two participants also sought confirmation from the researcher about potential interface capabilities. P31 asked "*but these are the only available options. I can't add something new, right? Or can I type something new?*"

Two participants explicitly reasoned about which template would be better suited to the command that they wished to provide. P47 wanted to "*tell the robot to approach the fridge*" and said the following while deciding on using the first template for their desired command - "*I need to go back to the first part because it is just* `Approach the fridge`*.*"

Participant P47 also used the provided template options to determine the robot's knowledge. P47 decided their next step would be to `turn on the oven` because they determined that "*it does know how to turn on the oven.*"

7.5.2.3 Participants reasoned about differences in available actions

Participants made observations about the various actions available to instruct the robot as a part of their planning process. P99 noted the interchangeability of some of the actions ("*Bring carry, clean, clear, cook, deliver, empty, fetch. Oh, some of these are kind of the same thing.*"). 2 participants reasoned about the differences in the roles of different actions to determine the appropriate action to teach.

> "*Am I heating it or baking it? I don't know if it has previously been cooked or has the cheese been melted at one point or is it loosely on top of the pizza? I guess I will say bake.*" (P92)

As a part of the teaching process, one participant also compared the visual feedback from different actions. P31 said the following while comparing the results of the two training tasks.

> "*things like approach the light switch is pretty easy for me to tell if it was right. But for the button, I can't tell yet.*" (P31)

While providing instructions, six participants referred to and reasoned about the information provided about the differences between primitive and teachable actions. P47 reasons about this difference while scrolling through the available actions - "*..there is* `bake`*,* `cook`*, but those are things* *that we need to teach the robot to do*" referring to the fact that teachable actions were in lowercase letters to distinguish them from the capitalized primitive actions.

P92 also actively compared the teaching process between primitive and teachable actions.

> "*So I'm probably quicker at, like, going through the tasks or the verbs that it already knows as opposed to figuring out, okay, which verb should I, or like, kind of action should I teach it how to do.*" (P92)

7.5.3 Participants became effective at teaching using learned knowledge

The process of updating their knowledge as a part of the teaching process enabled the participants to leverage this knowledge to engage in more effective teaching.

7.5.3.1 Participants effectively translated their teaching plan into action

Through the various exploration steps, participants were able to use the gained knowledge to translate their teaching plan into corresponding instructions for the robot.

P11 could effectively translate their training session experience to the actual teaching task. P11 referred to the third training task to use the `Wait for thirty seconds` command to set the time to bake during the actual task: "*ah okay this is probably when the wait 30 seconds, that's how I should do the time.*" We also observed this ease when participant P67 used the task prompt to identify the first sub-task. P67 wished to approach the refrigerator first since the pizza was stored there.

> "*so the goal is to get the pizza on the counter, after cooking it in the oven at 350 degrees for 30 seconds. So with that in mind, I think it would make the most sense to start with approaching the refrigerator first.*" (P67)

Three participants also intended to use the time when the robot was executing their previous instruction to prepare the next instruction to send on the interface. For example, after P7 had sent the instruction `Wait for thirty seconds`, P7 planned to "*prep the next task of "remove pizza* *from oven.”*” As the robot successfully executed the participant instructions, they began to expect the robot to successfully do specific tasks and nine participants directly moved on to the next task as planned before. For example, once P49 had given the instruction `Approach the oven`, they said, “*And now I think I’m gonna turn on the oven next*” and directly proceeded to provide the instruction `Turn on the oven`.

At times, however, five participants also quickly acknowledged the success of the robot before moving on to the next task:

> “*Okay, yep. He’s got the pizza. And we then close the fridge.*” (P74)

Since some of the participants naturally broke down the task into two halves, three participants acknowledged the end of the first half of the task, which is that they cooked the pizza, and then moved on to the second half of the task. Once the pizza had cooked in the oven for thirty seconds, P56 referred back to the document to read aloud part of the task description (“and the cooked pizza must end up on the counter”) and then decided the next step was to “*pick up again the pizza from the oven.*”

7.5.3.2 Participants changed their teaching plan on the fly

The effectiveness of the teaching process came through, especially when participants changed their plan on the fly. Participants were able to use the current state and robot behavior to influence their plan change and then execute the new plan through updated instructions.

Six participants easily switched from one template to another depending on whether they wanted to provide a specific command or a simple command as a part of their plan. After turning on the oven using the `Turn on the oven` command, P23 switched to the second action template and said, “*okay now I’m doing a complex state again*” about it and then proceeded to provide the command `Cook pizza in the oven.`

At times, eleven participants started entering values and on planning out loud and determining a different plan, switched to a different template to provide their new desired instruction. P74 said, “*try turning on the stove, I guess.*” when the second action template was active. They began selecting the values on there but then switched to the first template since it was better suited to that specific command: "*And this would be the first kind.*"

Participants also displayed familiarity with the different types of action instructions and intention to teach the robot when two participants changed from teaching a primitive action to a teachable action. P99 provided primitive actions to retrieve the pizza from the fridge and temporarily put it on the counter when they had the following realization about primitive actions: "*Well hold on. So I'm doing all this stuff but I could also teach it to do this stuff. These are all subactions.*" P99 then proceeded to teach it a new action with the instruction `Cook the pizza in the oven.`

Participants typically had a plan going into the task. However, at times, due to the current state of the task teaching process, including the environment state, five participants changed their order of actions in anticipation of the next part of the plan. For example, P47 instructed the robot to pick up the pizza. Once the robot had the pizza, P47 realized that the robot might not be able to operate the oven while holding the pizza. So they described their thoughts related to their change of plan:

> "*Okay, so I will put the pizza on the counter, even though it is not cooked yet. Because I may need to go open the oven or heat up the oven or something.*"

7.5.4 Participants actively evaluated the robot's task progress

Evaluating a student's progress is a common part of the teaching process. It is through this evaluation that the teacher can decide whether the student is learning successfully, thus providing feedback about their own teaching process. In our study, we observed participants providing instructions and then evaluating the robot's actions and responses to the instruction as a part of the overarching teaching process.

7.5.4.1 Participants used robot actions and responses to evaluate progress

All the participants evaluated the robot's progress with respect to the provided instruction using visual confirmation of the robot's movement or action or the combination of the two. In response to P59's instruction `Approach the lightswitch`, the robot proceeded to move towards the lightswitch. P59 responded to this movement: "*And I can see the robot, yeah, doing the task.*" After P78 provided the instruction `Pick up the pizza from the fridge`, the robot moved towards the fridge and opened it, and the pizza that was inside the fridge appeared on top of the robot. When this happened, P78 acknowledged this success by saying "*Alright. Good job.*"

The robot has built-in responses to various instructions as described in chapter 5. Six participants used these responses to evaluate the robot's progress. P67 provided the instruction `Approach the fridge` and once the robot did this action, it responded, "I am ready for a new task." P67 used this response to judge the robot's success and to move on to the next task: "*All right, with the robot being ready for the next task, I think it would be helpful to open the fridge.*"

Three participants sometimes had uncertainty about portions of the task, which was resolved in response to the robot's action. For example, P31 wanted to set the oven temperature to 350 degrees but was unable to find options to be able to do that: "*I have to figure out where I can, how I can get this to a 350 degrees.*" However, once they sent the command `Turn on the oven`, the oven displayed 350 on top as shown in Figure 5.1b. Therefore, their doubt was resolved as a result of this action.

> "*oh great. Okay, so I guess by default the, so it shows 350 there, so I guess that's the default temperature.*" (P31)

7.5.4.2 Participants judged robot actions and responses

As a part of their evaluation process, eight participants paid attention to various aspects of the robot's actions and verbal responses and acknowledged them as they occurred. P49 planned to provide the command `Open the oven` and acknowledged it with "*Okay.*" once the oven was opened successfully. Two participants also acknowledged when the robot moved in response to a command. When the robot started moving in response to P7's command `Approach the lightswitch`, P7 commented: "*Robot is going in the right direction of the light switch.*" P7 also sounded positive in response to the robot's action success. P7 provided the command `Pick up the pizza` and once the pizza appeared on top of the robot, said: "*Great, picked it up.*" One participant also noted when the robot responded with "I am ready for a new task" in order to move on to the next section.

7.5.4.3 Participants noted robot failure and non-responses

As a part of their evaluation process, participants also noted when the robot failed at performing a task or if the action or movement did not visibly occur.

Sometimes as a result of an unexplainable failure in the robot, the robot did not perform the action or respond accordingly, and three participants noted that. P67 planned to send the instruction `Turn off the oven` ("*So with that in mind, I'm going to turn off the oven.*"), and when the robot did not do anything, P67 commented "*Well, at least attempt to.*"

For specific failures, the robot would provide some explanation of what failure occurred. When P56's attempt to `Put down the pizza in the oven,` failed, the robot responded with "The put-down task failed." P56 noted this response: "*Failed. Okay.*"

On being provided the command `Press the button on the watercooler`, one can see the robot moving towards the watercooler; however, there is no explicit display of the button being pressed. P7 noted this discrepancy: "*You see, the button doesn't do anything.*"

Participant P7 also noted when the robot did not move in response to a command. P7 provided the command `Approach the table` after providing the command `Approach the oven.` Since the table and the oven were right next to each other, the robot did not move in response to the second command, and P7 acknowledged this ("*Did not move*").

7.6 Insufficient knowledge resulted in incorrect and incomplete mental models

Participants actively explored the various knowledge sources to gain information and then leveraged that knowledge to aid their teaching process. However, the information available was not always sufficient leading to incomplete mental models, which led to inefficiencies and failures when interacting with the robot. Insufficient explanation when the robot failed also led to participants incorrectly updating their mental model about the task and the robot.

7.6.1 Participants did not have sufficient knowledge of how the interface works

We designed this template interface and updated it based on participant feedback in a pilot study. However, conducting this study gave us some more feedback on ways how a) we could have provided better instructions to participants and, b) this interface could have been designed better to be more intuitive and understandable to participants.

7.6.1.1 Participants did not have a sufficient model of the available instructions

As described before in Section 5.2, the actions in the drop-down list had been divided into primitive and teachable actions, where the primitive actions were capitalized to distinguish them from the teachable actions that were provided in lowercase letters. However, the values in the lists corresponding to other parts of speech did not have any such distinction and were all presented in lowercase. P31 had to confirm if the capitalization difference was restricted to actions alone.

> "*I see the objects and prepositions and things and they're all lowercase, so I guess I'm wondering, does it know how to deal with those, because they're not in caps.*"

P92 did not realize that the text area where the commands and the robot responses appeared could be scrolled up to see the previous interaction. P92 was also concerned that the two button commands (`You are done` and `Wait for thirty seconds`) were indistinguishable and there might be an effect on the other button command on sending the first one.

> "*So I'm thinking, do I go ahead and press you are done? Is that gonna mess with the 30 seconds? Is that like a separate instruction?*"

Multiple participants (P11, P31, P47, P74) were unsure as to why they would use the `Wait for thirty seconds` when having to use it during the training task ("*I guess I am curious why there is* *a waiting for 30 seconds option?* - P47). Unfortunately, we could not provide explicit information about this command since it would have interfered with the study conditions.

7.6.1.2 The interface did not provide sufficient indication of selections

While the value that the participants clicked on the drop-down list was populated in the attached text box, the option was not clearly selected or highlighted in the drop-down list leading to a lack of certainty about its selection in the first place. Five participants had to self-correct or confirm multiple times that their desired option was, in fact, selected.

This was true of the template selection as well. While the selected template was updated on the interface depending on the button pressed, there was no additional confirmation as to what template the participant was actually using. Leaving the button to be displayed as pressed or highlighted would have given additional feedback to confirm this selection.

7.6.1.3 Participants faced difficulties due to idiosyncratic interface features

The lack of sufficient feedback from the interface about the option selected led to six participants unintentionally sending the wrong option. Participant P7 unintentionally sent `Open the pizza` when they intended to send `Pick up the pizza` because the two verbs were situated next to each other on the list.

Due to limitations in the Rosie system, the `Wait for thirty seconds` did not always appear on the interface after it was sent. Also, due to the way in which the `Wait for thirty seconds` command was set up, it was not possible to interact with the interface (for example, to set up the next command) while the command was running. This is because the system was literally paused for that duration. Participants (P7, P56, P74, P78) encountered this issue.

The text inputs were set up to be cleared whenever the template was changed. Participants P31 and P99 encountered this problem while partially entering a command on one template and switching to another to deliberate about the command. However, when they returned to the original template, their partially selected options were no longer present.

7.6.1.4 Participants were unclear about the goal activation conditions

Participants were not actively provided information about the goal templates even though they were told that they could provide a goal in response to a robot's request. This was decided because we wanted the participants to figure out the learning process of the robot through interaction. The goal templates would only activate when a teachable action was provided since the robot could only process a goal instruction as a part of that learning process.

Multiple participants (P31, P47, P56, P92) tried clicking the goal template buttons when they had not yet provided a teachable action instruction. They encountered its inactive state; therefore, the goal templates did not update on the interface.

> "*And I can't use either of these goal options right now, so that's interesting. Okay.*" (P31)

This also meant that some participants (P23, P26) discovered the goal templates by chance when they were exploring the interface after providing a teachable action instruction.

> "*Oh, I see. Now there's like a third template. I didn't see the third or four templates.*" (P26)

However, this lack of information also led to P26 assigning the wrong reason for the goal template activation. P26 wrongly assumed that the goal templates were activated in response to the primitive command `Approach` when it was in response to the teachable action `Heat` that they had provided before.

> "*I know there's a clue because every time, like I approach you know, in the third and fourth template, I'm like, it doesn't come up with when it's somewhere else. So I know that. I know that the robot has given me a hint that it's here. That it's in one of these like two, the last two templates.*"

7.6.2 Participants did not get sufficient updates from the environment

While participants interacted with the robot, they had to keep track of the environment to observe the effects of the instruction provided to the robot. Participants were sometimes unsure about the changes in the environment, leading to uncertainty about the success of the instruction.

7.6.2.1 Participants made incorrect assumptions about the environment

Participants (P11, P23, P26, P74) were unsure about and sometimes even possessed wrong assumptions about the environment. P74 noted their confusion about the indicator on the oven that turned red when it was on: "*And there's a red dot on it. I'm not sure what that means.*"

Both participants P74 and P78 made the assumption that the oven will automatically preheat to 350 degrees on seeing the number on top of the oven.

> "*Okay. It's on 350. Now, I don't know if it has the preheat or not, but I'm assuming it does.*" (P78)

Because the researcher had mislabeled the water cooler just as a "cooler" in the environment, P26 spent a little time looking for the water cooler as a part of the training task. The researcher modified the label after this encounter for the remaining 12 participants since it did not modify the conditions of the actual task.

7.6.2.2 Participants were uncertain about object states and locations because of insufficient evidence

Participants (P11, P47, P92) displayed uncertainty about object states and their corresponding locations. P92 wondered about the state of the pizza ("*Okay, I'm not entirely certain. Is that soft?*") P47 was unsure if the fridge was really open even after providing the command `Open the fridge.`

> "*Um, okay. What is this? Oh, so does this mean the fridge is open? Okay, I think the red circle is the pizza*"

P92 was unsure about the location of the pizza: ("*I'm assuming that means pizza's in the fridge but I guess I will find out eventually when I have to open it.*").

Participants (P7, P59, P99) felt the need to rely on their assumption that the oven will stay at 350 or preheat to 350 ("*I'm gonna guess it's just at 350 to begin*" - P7).

Once participants provided the instructions to finish baking the pizza, the pizza object in the environment did not display any changes corresponding to its cooked state. Therefore, participants (P26, P47, P49, P56) were unsure how else to check if the pizza was adequately cooked or heated.

> "*One thing that I couldn't check the pizza is actually cooked. I didn't check it. But I don't know how can I check that one?*" (P56)

Eventually, participants (P7, P23, P26) just decided to move on to the next subtask of moving the pizza to the counter without being able to explicitly check its state.

> "*And then we'll just put the presumably cooked pizza on the counter.*" (P7)

7.6.2.3 Participants wanted the ability to confirm environment changes

When the robot performed the command `Press the button on the watercooler,` P31 was unsure about the action's success without observing the changes in the environment.

> "*I'm not sure how to know if it really pressed the button. I'm not sure. I don't like see a way, some feedback for that.*"

After placing the pizza in the oven, but before turning on the oven, P11 tried to get from the researcher ways to confirm the status of the oven: "*And then I'm not sure if the oven actually turned on. Is there a way I can tell if the oven is on?*"

Observing the red circle in the fridge with the label pizza was not sufficient for P92 to know the state of the pizza. They decided to scroll through the object drop-down list, and on finding "pizza" on the list were able to confirm that pizza was already made.

> "*So the pizza's already made. I don't have to make the pizza. I just have to put it in the oven to reheat it. Okay, that makes sense.*"

Participants P26 and P99 wanted to explicitly confirm the locations of objects. P99 provided the command `Open the fridge` to confirm that the pizza is, in fact, inside the fridge.

> "*I want it to open the fridge assuming the pizza's in the fridge.*"
>
> (After the fridge is opened)
>
> "*Ha, ha ha. I see it.*"

Participants P59 and P74 wanted to explicitly confirm that the robot knew the updates that occurred in the environment after specific commands were provided.

> "*still trying to figure out whether the oven's preheated or not. I don't see any actions in the list that give me- that seem to give me an option to ask Rosie to check for me.*" (P59)
>
> "*Okay, and it looks like it's already set for 350, that I can see, but I don't know how the robot knows that it is.*" (P74)

7.6.3 Participants did not have sufficient information to confirm robot action success

Participants did not get sufficient information from the environment, the robot, or a combination of both sources to adequately confirm the success of the robot's action.

7.6.3.1 Participants were unsure about object states after robot actions

As described before, in response to the command `Press the button on the watercooler`, it was not clear from the environment that the robot actually pressed the button. Both participants P7 and P92 were not sure of the robot's success at doing the `Press` action. Participant P7 waited for a long time in anticipation of visual feedback from the robot ("*You see, the button doesn't do anything.*). Participant P92 decided to rely on the robot's response "I am ready for a new task" alone to assume action success and continue teaching.

P26 turned on the oven and saw the 350 degrees, but were still unsure as to whether the oven was heating.

P47 was not sure that the oven would stay at 350 degrees and said, "*I will keep an eye on if the oven temperature changes because it is at the temperature I need right now*."

Two participants were also worried after placing the pizza in the oven that the pizza may get burned in the process.

> "*Okay, well I'm just gonna say wait for 30 seconds. And as I'm waiting, I'm gonna hope that the pizza doesn't become charred and inedible.* (P67)

7.6.3.2 Participants find robot's response insufficient to determine task success

While the robot's response "I am ready for a new task" usually indicated the success of the previous action, it was not always sufficient to determine task success. This was partially because the training tasks were set up for success so "I am ready for a new task" was an indicator of success, but this was not true of all possible tasks.

Participant P67 once reasoned incorrectly about the meaning of the response. While reading the training tasks, P67 assumed that the robot's response "I am ready for a new task" meant that the robot waited for 30 seconds even when they did not provide the command.

Participants (P7, P11, P31, P59, P92) also decided to trust the robot's response as a marker of task completion without sufficient proof from other sources.

> "*I will wait for it to say that. Okay. It says, "I'm ready for a new task," so I think I can now open.*" (P31)

7.6.3.3 Participants required additional signals of robot action completion

Participants (P11, P23, P31, P56) were unsure that the robot actually succeeded in its action. After providing the command `Approach the lightswitch`, P23 did not register that the action was already completed and said, "*Okay, so I'm now simply, I'm now looking at the visual and waiting* *for the robot to perform the action.*" P56 was unsure that the robot had in fact completed this action because they thought "*there is a little bit gap here*" referring to the environment.

Participants P7 and P92 resigned to not getting external verification of the fact that the robot did in fact wait for 30 seconds in response to the command.

> "*I guess I'm trusting that it knows how to count for 30 seconds. Okay, yeah, I didn't see a counter.*" (P92)

7.6.3.4 Participants wanted the ability to confirm robot actions

Participants P7 and P59 expressed the need for a way to additionally verify the state of objects or the task once the robot had performed the action.

> "*still trying to figure out whether the oven's preheated or not. I don't see any actions in the list that give me that seem to give me an option to ask Rosie to check for me.*" (P59)

Participants P7 and P31 were additionally motivated to confirm the correctness and completion of the action through visual feedback.

> "*Really, is there any way to confirm that the button was pressed?*" (P7)

Since P56 required additional signals of robot action completion for the training action `Approach the lightswitch`, they assumed the need to confirm this through other actions and began to find options for that before the researcher intervened and said that they could move on since it was still the training session.

During the actual task, P56 was motivated to conduct unit-test-like actions to test the success of the robot's action and explained why they are choosing to perform these tests.

> "Find the, `find the pizza from fridge.`
>
> *This, how can I check whether the robot have or not pizza actually, so, hmm. Let's test that if the robot actually can put and put down and retrieve that one again. So, I don't know, let me put down this one.*

`Put down the pizza.`" (P56)

7.6.4 Participants did not have consistently reliable mental models of the robot

As a result of different media or their experiences with other intelligent agents, participants came in with prior assumptions about the robot's capabilities. As they interacted with the robot, they evaluated the robot using their pre-existing mental models and based on its actions updated their mental model. At times, this update was additive but some updates also surprised the participants. When the robot failed, participants seemed to require more information than was available to them.

7.6.4.1 Participants had prior or task-dependent expectations of robot

Nine participants had expectations about the `Wait for thirty seconds` command.

> "*I'm thinking that,by telling the robot to wait for 30 seconds, most people would expect to find that the robot would just remain stationary. And, in doing so, the next course of action would be seeing if the robot is ready for a new task.*" (P67)

Two participants also anticipated robot movement and action in response to specific commands. After providing the `Approach the lightswitch` command, P26 described that "*Okay, I thought it was gonna physically push it or something.*" P74 expected the robot to do the action successfully even before it actually did the action ("*We want to tell it to approach the light switch... And then the robot does it*").

Three participants assigned ideal expectations to responses from the robot when it runs into issues.

> "*In my head, at least I'm assuming it's processing in an ideal world, what to get back to me on, where I would like a robot like this to help me out would be used to [sic] because the instructions are fairly clear, right?*" (P23)

Participants (P11, P78, P92) also expected the robot to already know how to do several actions that would be considered typical in everyday life. For example, P78 began to provide a teachable action command `Carry the pizza to the oven` only to then realize the robot does not know how to do it: "*Oh, it doesn't know carry yet. Okay. I forgot the lowercase ones they don't know.*" P11 also has expectations about typical oven tasks that the robot does differently.

> "*the oven is still open so I think I'll have to tell it to close the oven when that's something I might expect...I include opening and closing the oven as part of that task.*" (P11)

Ten participants mistakenly assumed that the robot has to be explicitly instructed to approach an object before interacting with it and added that as an action before performing a few of the tasks. Our conjecture is that this was because the first training action referred to the `Approach` action leading participants to believe this, even though in the second training action, the robot does the action with the water-cooler without needing to be instructed to approach it.

> "*I noticed it's set up that it has to approach the light switch first. It's not just like turn on the light switch or turn it off.*"

7.6.4.2 Participants reasoned about robot actions and movements

Participants (P31, P59) used instructions to explore the robot's capabilities.

> "*so I'm gonna say approach. Okay. Actually I'm gonna try something just for my learning, but I wanna, I'm gonna say something like put the pizza into the oven.. I'm not sure if that will work yet. I'm guessing that it won't be able to do that if the robot's not already over at the oven, but let me try.*" (P31)

Participants (P11, P26, P31, P49, P59) also used the robot's actions and responses to specific commands to reason about its behavior in and knowledge of the environment to update their model of the robot. P59 notes that the robot automatically knows to approach the referred object when having to interact with it, rather than being first asked to approach it first.

> "*I'm aware from this task that I don't have to tell it to approach the water cooler. And just telling it to press the button it knew to go over there already, so I'm gonna keep that in mind. So, earlier I saw Rosie just moved straight from the light switch to the water cooler with the task. So, I'm going to go to the first choice here and have Rosie turn on the oven.*" (P59)

P78, in addition, reasoned about its knowledge to change from a general instruction (`Fetch the pizza`) to provide a more specific instruction (`Pick up the pizza from the fridge`).

> "*Fetch pizza. Now, is it gonna know where the pizza is? Let me try that and see...Pick up pizza from the ..We'll go with fridge.*"

P7 was able to derive confidence in the robot's ability based on its action. When the robot moved in response to the command `Approach the lightswitch,` P7 said: "*Robot is going in the right direction of the light switch. That's promising.*"

The teaching process also included participants (P11, P26, P31, P49, P67) reasoning about the robot's learning process throughout the task.

> "*..if I have to just say turn the oven on or something,.... Then it, I don't know, will it then ask me for how, what temperature to set it to? I wonder if that's something the agent is gonna ask me.*" (P31)

7.6.4.3 Robot actions did not meet participant expectations

The robot's actions and responses did not always match the pre-existing expectations of participants.

Having interacted with intelligent agents like Siri and Alexa in the past, P49 was surprised by the text-only response of the robot: "*But I'm not hear...It didn't say I'm ready for a new task. Oh, it doesn't talk back....Oh, I'm like waiting for it.*"

Given this mismatch in expectations, four participants sometimes felt the need to confirm the success of the robot's action or movement with the researcher. Once the robot did the movement in response to `Approach the lightswitch`, P56 was unsure and asked the following to the researcher.

> "*Shall we say that it's succeeded in approach? I mean, there is a little bit gap here*"

Since participants could not see the button being actively pressed as a part of the task `Press the button on the watercooler`, three participants wished to confirm the success with the researcher.

> "*So for something like a button, how do I know if it pressed it or not?*" (P31)

Twelve participants noted that the oven was directly set to 350 degrees when they turned on the oven and six participants were surprised to see it.

> "*Oh, I see now. It's just magically 350 degrees. I wish that were the case in real life.*" (P92)

The robot was set up such that it could automatically open a receptacle to pick up something inside it, however, it expected an explicit open instruction when it had to put something inside the receptacle. P56 was naturally intrigued by this discrepancy.

> "*Oh, now I see I should open the fridge first but somehow we pick, picked up the pizza, right? [laughs] I guess it combines like opening the fridge and picking up the fridge [sic] at once with the instruction.*" (P56)

7.6.4.4 Participants did not trust the robot's ability to keep time

For the actual task, baking the pizza for 30 seconds was one of the requirements. In anticipation, we provided the command `Wait for thirty seconds`. However, four participants expressed uncertainty about the robot's ability to wait for precisely thirty seconds.

P56 comments: "*So wait for 30 seconds. I hope that time doesn't exceed in the robot space anyway.*"

P7 on being prompted to think out loud after providing the `Wait for thirty seconds` command, wished to have confirmed that the robot actually waits for 30 seconds: "*Yeah, I'm honestly right now just thinking that I was waiting for 30 seconds, but I did not look at the timer at the top of the screen to confirm if that's 30 seconds.*"

P92 was unsure but hoped that the robot "*knows how to count for 30 seconds.*"

7.6.4.5 Participants were unsure about the physical properties of the robot

Participants expressed uncertainty when it came to asking the robot to do actions that would require the robot to possess specific physical properties.

This included P47's uncertainty about the height of the robot to be able to `Put the pizza on the counter`.

> "*Hopefully the robot is tall enough*" (P47)

Participants (P7, P59, P67) were unsure if the robot would be able to do other tasks while holding the pizza.

> "*I so don't know how many like arms the robot has. So I'm trying to open the oven with the pizza being held.*" (P7)

> "*What if I just said, "Approach the oven." And let's see if it still, if it carries it. All right, it's taking it. Don't drop that pizza.*" (P78)

Four participants were unsure that the robot had access to the updates or knowledge in the environment. After P74 gave the command `Turn on the oven`, the oven updated to 350 degrees on top. However, P74 said: "*Okay, and it looks like it's already set for 350, that I can see, but I don't know how the robot knows that it is.*"

Participants (P26, P92) also assigned capabilities to the robot when it was inactive. When the robot failed and did not provide a response to a command, P92 assumed the following: "*Okay, I suppose maybe it's internally waiting for 30 seconds and it will tell me.*"

7.6.4.6 Participants were mistaken or unsure about the reason for the robot's failure

The discrepancy between the participant's mental model of the robot and the robot's actual model was especially evident when the robot failed.

When the robot ran into a failure state when it either did not understand the command or was unable to do a task because of unsatisfied preconditions, three participants were not sure why the robot failed.

The robot was unable to understand the goal statement `The goal is that the oven is on` and responded with "I don't understand" when being given the command.

> "*You know I provide desired state, the goal is that the oven is on, which is a fairly all right desired state. It keeps saying, I don't understand.*" (P23)

For the primitive action put-down, when an object needed to be placed in a closed receptacle, the robot could not perform the put-down action unless the receptacle was opened first. So when participants (P31, P56, P59, P92) gave the command "Put/Put down the pizza in the oven," the robot failed and responded with "The put-down task failed." The robot responded with this message irrespective of whether the verb phrase `Put` or `Put down` was used since internally it processes them similarly. However, this led to obvious confusion from the participants' perspective.

> "*Okay, so robot says, "The put-down task failed." Okay, so I guess this put word actually doesn't mean, wait, what? That's weird. I said put, not put-down.... I'm not sure why it failed.*" (P31)

Without sufficient information, participants at times attributed the failure to incorrect reasons. P67 provided the goal `The goal is that the oven is on` as a part of the larger task `Bake the pizza in the oven.` The robot failed because it didn't understand the goal but P67 assumed failure at the task level rather than the goal statement: "*Doesn't understand, okay. Okay, well, let's try this again. Um, so bake wasn't it.*"

When the robot failed to do the put-down task because the closed oven was not opened first, P78 assumed wrongly that the robot failed because they missed the incorrect intermediate step.

> "*Oh, it failed. Oh, take the pizza to the oven. Maybe I should do that first.*"

7.7 Incomplete and incorrect mental models led to uncertainty, mistrust, and failures

When participants did not have adequate information to confirm their knowledge of the robot or the progress of the task, it led to uncertainty about the task teaching process and the robot's ability to do the task. This lack of information also made it harder for participants to quickly recover from failures. This led to participants facing issues that made it difficult for them to efficiently teach the robot.

7.7.1 Participants displayed learned uncertainty about the task

Some participants encountered multiple issues while trying to teach the task to the robot. This led to a lack of confidence in their own teaching process where they began to anticipate failures and were uncertain about their process and also did not trust in the robot's ability to learn the task successfully.

7.7.1.1 Participants displayed uncertainty about task aspects

Despite the task description provided, two participants displayed uncertainty about task aspects not explicitly defined in the task description.

> "*Realize I did not close the oven door. I don't know if that's important*" (P7)
>
> "*I left the fridge open, but I don't think I like I'm not sure if it's important that Rosie learns to close the fridge. I'm gonna assume not.*" (P99)

Participants (P26, P59, P92) continued to express uncertainty about preheating the oven even after encountering an alternative to preheat, or observing that the oven was at 350 degrees.

"*I am looking at the actions just to make sure- still trying to figure out whether the oven's preheated or not. I don't see any actions in the list that give me-that seem to give me an option to ask Rosie to check for me.*" (P59)

Participants P11 and P56 were unsure about being done with teaching the task without a clear marker of the pizza being cooked.

"*I'm not sure, the pizza didn't change. So I don't know if that means it didn't actually cook while it was in the oven or if I can assume since the oven was on at that temperature and I waited 30 seconds that it cooked - it would be my remaining question.*" (P11)

7.7.1.2 Participants were not confident in their teaching

There were different aspects of the task where participants did not sound confident in their ability to teach successfully.

P92 sounded intimidated by the task description in comparison to the training task: "*So I need to figure out what is, where everything is in the room and go?*" P49 expressed uncertainty at being able to find the desired option on the drop-down list: "*Okay, I think it's asking me to click approach, if I find it.*"

Three participants apologized or laughed awkwardly in response to smaller issues they ran into while teaching the task.

However, when they did run into failures more often, some participants (P26, P49) felt like it was their fault that they could not teach the task properly.

"*Man, I feel like I'm failing. I feel like there might be no answer to this. Oh my gosh, tell me what I did wrong.*" (P49)

7.7.1.3 Participants wait a long time before providing the next instruction or thinking out loud

Participant uncertainty also came through in the amount of time they waited before providing the next instruction or thinking out loud. Usually, this is when the researcher prompted them to describe what they are thinking out loud.

This was sometimes a result of not seeing expected updates in the environment ("*You see, the button doesn't do anything*" - P7) or quietly planning tasks ("*So I'm trying to open the oven with the pizza being held*" - P7).

Two participants also waited quietly while the robot performed the tasks. P74 indicated their hesitance about the think-aloud process: "*the thinking out loud is a little bit unnatural for me, but I will try.*"

7.7.1.4 Participants required additional reference to teach task

Two participants were unsure or did not remember the last commands that were sent.

> "*And now I think I need to start the oven? Or did I already start it? Wait, did I already? Did I turn on the oven? I'm forgetting.*" - P49

Participants (P23, P47, P49, P78, P99) also had to refer back to the document either to verify how to use the interface or how to teach a teachable action:

> "*Let me... I'm scrolling back because I don't remember what needs to be done after I have the statement in place*" (P23)

7.7.1.5 Participants sought clarification from researcher

Even though the researcher had informed the participants that they are unable to answer most questions from the participant except for the information that was already provided, participants still needed to seek clarification during the teaching process.

Participants (P26, P49, P74, P78, P99) asked clarification questions regarding the robot teaching process. P74 wanted to confirm that the teachable actions could be used and that "*they are not pre-programmed.*"

Having done the individual commands in the training tasks, P26 and P74 inquired whether the actual task needed to be done using one command as well: "*do I have to do everything in just one swipe? Or can I give multiple commands?*" (P26).

Three participants were also unsure and tried to confirm with the researcher whether they had provided all the commands required to finish teaching the task.

> "*Oh no, what happens if I'm stuck? I really wanna figure this out though... I'm close, right?*" (P49)

7.7.2 Participants were unable to efficiently teach the robot

At the beginning of the study, participants were provided only a minimal description of the robot's learning process. This was because we wanted to explicitly study whether participants are able to learn about the robot's learning process through interaction. While participants displayed some success in adapting their interaction to teach better, the combination of inadequate training and inadequate knowledge about the robot's learning process resulted in participants being unable to teach the task in an efficient manner.

7.7.2.1 Participants did not display adequate knowledge of robot instruction

Participants P49 and P78 were unsure how to provide instructions to successfully teach a teachable task. In response to a request for a goal or sub-task for the task `Carry`, P78 is unsure how to proceed and whether they ought to repeat their instruction.

> "*Maybe? Is that okay to do the one I tried before? Maybe not. Maybe. Okay, got it. Let's see*" (P78)

The robot engaged in a strict back-and-forth interaction where the teacher is expected to wait for the robot's response before providing the next command. However, in the flow of teaching the task, participants (P26, P78) sent two consecutive commands without waiting for a robot response before the first one.

While teaching a new task, it was not clear to P99 how to indicate the end of teaching that task. They had begun teaching the task `heat` and had finished providing the actions that they considered a part of heat:

> "*I guess, in general, I'm thinking at what point do I tell it the sub-task is done? Like I'm done teaching heat.*"

This usually led the participant to start providing instructions for a new task without ending the previous one. P11 intended to teach the task `bake` and then the sub-tasks `heat, empty` under the task `bake`. This would have looked like the following:

1. `Bake the pizza`
2. `Heat the pizza`
3. Teach subtasks of `Heat`
4. `You are done`
5. `Empty the oven`
6. Teach subtasks of `Empty`
7. `You are done`

However, they missed providing the command `You are done` which led to the robot learning `Empty` as a sub-task of `Heat` rather than `Bake`.

In general, participants did not have a good grasp on when and how to provide the command `You are done` despite being provided information about it. This led to participants (P7, P23, P59, P67) incorrectly providing it to end a primitive action ("*I see the robot move to the light switch there. Okay, I'm gonna say that's done.* " - P67) or to end the complete task ("*Click `You are done`?*" - P7).

7.7.2.2 Participants misinterpreted the robot learning process

Because of their lack of familiarity with teaching this robot, participants were often unaware and thus ascribed incorrect learning mechanisms to the robot or taught it in a way that could not be processed by the robot.

Participant P49 wanted to teach the overarching task of cooking the pizza and provided the instruction `Cook the pizza in the oven` which referred to the preparation involved as well. However, they reused the verb `Cook` to refer to the act of the pizza cooking inside a hot oven. As much as another human may understand the multiple uses of the word "cook," the robot did not know how to process this command.

The robot is set up to request a goal or sub-task as a part of a new task learning process until the teacher gives the command `You are done` to end the teaching of that particular task. However, participants ascribed various incorrect reasons for this request from the robot. Participants (P23, P49, P67) mistakenly assumed it was a bug on the part of the robot.

> "*I think it's stuck in some sort of weird loop with me wherein anything I do it just keeps asking for the next goal or sub task, but I feel my request, it's fairly clear.*" (P23)

P49 also interpreted this request as a reason to provide a more specific command `Turn on the oven for the pizza` despite the oven already being on.

Seven participants repeated or rephrased instructions even though the action was already executed in hopes of a different response from the robot. When the task `Put the pizza in the oven` failed due to the oven being closed, P59 decided to repeat the command: "*I'm not sure if I had this is what I had clicked the first time or not, so I'm gonna try it a second time*".

Participant P67 assumed that the robot's continuous requests for goals meant that it did not know them yet and decided to post-hoc teach them after the actions were already done. For example, P67 had already provided individual instructions to bake the pizza in the oven but since the robot continued to request a goal or sub-task, they provided the action but in a goal format: "*Oh, so it's still interested in the goal of bake. Okay. I guess it's, wants to learn. The goal is that the pizza, huh, the goal is that the pizza is in the oven.*"

7.7.2.3 Participants were unsure that the robot would execute tasks successfully

Participants (P7, P47, P49) were unsure if individual actions would work and said that they would "try" teaching the next action.

> “ *I will see-I am going to try “pick up the pizza,” and see if that works.*” (P47)

Participants (P7, P26, P49, P92) were also unsure if the robot would be able to do a new task.

> “*Cook the pizza and see if that works. Okay, “cook the pizza in the oven.” Maybe it’ll cook with start. But maybe start oven? I don’t know if it’s gonna work.*” (P49)

P92 had already turned on the oven and put the pizza inside and was concerned that the process of cooking may exceed the prescribed time description: “*So it says for 30 seconds, so hopefully it’s not already cooking.*”

P92 was also unsure about what the result would be once they provided an action instruction: “*Put the pizza in the oven. Is that going to work?*”

P56 was unsure about the phrasing of their instruction to put the pizza on the counter until they saw the robot do it successfully: “*So I’m not sure. Put down using template two and put down the pizza on the counter, I guess this one.*”

Participants (P7, P26, P49, P92) were unsure if the task instructions would work properly and yet decided to use the robot’s response as a way to move forward. P49 uses the robot’s request for a goal or sub-task to try to provide instruction.

> “*Maybe it’ll cook with start. But maybe start oven? I don’t know if it’s gonna work cause it’s not a [primitive action]*”

7.7.2.4 Participants experience difficulty with task

Some participants were stuck being unable to teach the task as desired and expressed uncertainty and frustration about the process.

P49 planned as they went along and oftentimes were unsure what to do next: “*Okay, turn on. Now, I don’t know what to do. Okay, let’s see. I don’t know what I’m looking [for].*”

Some participants (P11, P23, P26, P49) had a difficult time translating their desired plan into robot instructions even after trying multiple times. P26 assumed they had to explicitly increase the heat of the oven to 350 degrees and attempted to provide such an instruction multiple times in vain despite seeing the update on the oven in the environment.

> "*I'm just trying to determine, like how to actually handle the robot. Like turn up the, like the amount of like, what's it called, the degrees to actually 350. But it just can't.*"

P23 was frustrated with the teaching process because they were unable to set the time command using the drop-down options, and somehow did not choose to use the command `Wait for thirty seconds.`

> "*So at this point, either I would give up in an real world or..But now I'm stuck in a waiting loop after trying to just set it time. Because I've spent longer on giving instructions to set for 30 than the actual time it would've taken to cook the entire thing, right?*

After encountering multiple failures, P26 and P49 felt like this was a teaching process with no answer. P26 asked the researcher "*Is there a trick question?*" with regards to the task.

Even though some participants managed to teach it in a shorter way, due to their multiple failures, P26 found this teaching more effortful than doing the task at home: "*When I do this at home, I never really notice how much work it actually takes.*"

7.7.2.5 Participants were unable to teach the task as desired

Some participants were unable to teach the task as they originally desired or planned.

Participants often discovered that they could not provide any other command while the robot waited in response to `Wait for thirty seconds`. P47 provided the `Wait for thirty seconds` after putting the pizza in the oven but realized that the oven was not closed. However, since there was no way to provide a new command while the robot was waiting, they had to resign to waiting without closing the oven door even though it was not ideal: "*Oops, I didn't close the oven. (laughs) I will just let it...I'll keep waiting.*" P7 also resigned to waiting without providing the next action.

P23 spent a long time during the task trying to look for or generate the 30-second option as a part of the drop-down list. P23 did not consider using the `Wait for thirty seconds` command. Having spent a lot more than 30 seconds time trying to figure out the options, P23 resigned to turning off the oven without providing it their desired time-related command.

> "*I just turned off the oven. I'm assuming it's cooked by now.*"

P92 and P31 encountered the put-down task failure multiple times when they tried to put the pizza into the closed oven and tried different actions in response before eventually figuring out the need to open the oven first. P92 commented on the long sidetrack before they could finally move on to the next part of the task: "*Okay, now I can finally fetch pizza from table.*"

One participant also faced difficulty in clearly separating different parts of the task. As a part of teaching the higher-level task `Bake`, P47 had already asked the robot to pick up the pizza when they realized that they had to open the oven. P47 determined that the robot could not open the oven while holding the pizza so they needed to place the pizza somewhere else. They noted that this action is not ideally part of the bigger task.

> "*So I will have it put down...Oh, it keeps asking me sub-task of bake. This technically isn't part of baking, but I will put down the pizza on the table.*"

7.7.2.6 Participants felt limited by instruction options in the UI

One of the interesting things that emerged as a response to having an inadequate mental model of the robot's learning process is that participants expressed the need for capabilities beyond the current possible robot instruction.

P26 described their desire to provide compound statements.

> "*Cause there are no compounds, so there's not pick up pizza, and then...Which isn't like, it's not bad. I don't want it to be confused if it gets confusing.*"

Given that the task specification included both the temperature and time component, seven participants often looked for 350 degrees and 30 seconds in the drop-down list but these options were not available.

> "*Okay, the more complicated ones don't seem to have numbers preset for the temperature. So I want to tell it how long and how hot to bake it for as guess I'm getting stuck in that these other options I have to be able to command don't have a temperature or time* (P11)

Three participants explored both goal templates to determine if there was a way to provide their desired goal even though it was not supported. P23 wanted to provide the instruction `The goal is that the pizza is cooked`, but "cooked" was not available as an object state since this assumes that the robot already knows how to cook a pizza. On not finding it, P23 was visibly frustrated: "*I believe my goal is to, pizza is cooked, right? doesn't make sense, right?*"

Participants (P26, P31) also wanted to provide action instructions that had a sentence structure that was not supported by the template. The interface was set up to give a popup error if any of the fields were empty when the command was sent. P31 wanted to provide the first type of action using the second template but encountered the error and specified the need for this ability.

> "*This is a small piece of feedback. I wanted to say approach the fridge and not have the rest here.*"

Participant P26 wanted to provide a command that would allow the heat to slowly increase to 350 degrees in the oven. Even though they observed the 350 degrees on the oven, they believed that the oven was not at that temperature yet. However, this command was not supported.

> "*Right now, looking for, some way to raise the heat. I can't find anything for now. There's no command for it. There isn't even like, there's command for turning up the heat or setting the heat to like certain temperature.*" (P26)

7.8 Failure situations led to attempts at recovery

When faced with inefficient and failure scenarios as a result of incorrect and incomplete mental models, participants were motivated to recover from these scenarios. At times, they proposed suggestions for capabilities that would help them recover or found alternative means to continue the teaching process. This shows that participants desired to continue teaching the task despite encountering these scenarios.

7.8.1 Participants experienced inefficient or failure scenarios and intended to fix them

As has been detailed in the previous sections, participants encountered various inefficient or failure scenarios while interacting with the robot. They noted these failures, considered alternative plans in hindsight, but also suggested ways in which this task could have been set up to be taught easier than the current situation.

7.8.1.1 Participant actions led to inefficient or failure scenarios

Twelve participants executed actions that resulted in them encountering various inefficient and failure scenarios with respect to the robot. All the failure and inefficient scenarios have been detailed in Section 7.9.2. The issues that participants faced while interacting with the interface have been detailed in Section 7.6.1.

7.8.1.2 Participants realized potential issues after they had provided instructions

Participants P7 and P47 realized that they did not close the oven door only after providing the instruction to the robot to wait for 30 seconds ("*Oops, I didn't close the oven.*" - P47). P47 decided to close the oven after the `wait` command and then asked the robot to wait for another 30 seconds.

P26 sent the `Wait for thirty seconds` command but immediately determined that "*That wasn't right*" because they were not sure if the oven was already heating at the right temperature.

P99 was not sure the robot could open the oven while holding the pizza, so they decided to put the pizza on the counter in the meantime. However, when they began to teach a new action `Heat` within the task of `Cook`, P99 noted that "*these are not really in order.*"

P49 provided the command `Cook the pizza in the oven` in response to the robot's request for a goal or subtask, but immediately noted that "*Maybe that's too specific.*"

While waiting for the robot to restart because of an unexpected failure, P56 realized that '*Oh, now I see I should open the fridge first, but somehow we picked up the pizza, right?*" since they had provided the `Pick up` command without asking the robot to open the fridge first.

7.8.1.3 Participants considered post-hoc alternative plan

At different points in time, participants (P26, P74, P92, P99) described their desire and their consequent alternative plan on how they could have taught different parts of the task more efficiently.

> "*Oh my gosh, I put the pizza all the way on the counter on the other side and I could have put it on the table right next to the oven.*" (P99)

While considering an alternative efficient plan, P92 expected the robot to already know how to do a teachable action `fetch`, when, in fact, it needed to be taught. P92 provided consecutive instructions `Open the fridge` and `Pick up the pizza` and then realized that they could have instead used the command `Fetch the pizza from the fridge` so that it was more efficient. However, given that the robot did not know the action `Fetch`, they would have had to teach it irrespective.

> "*So, I see now how I could have done that with fewer things like fetched the pizza from the fridge.*"

P31 analyzed their teaching process and noted that they did not teach any new actions: "*I didn't teach it anything new. I just used the existing actions, so I'm not sure if I did what you wanted.*"

For the eight participants who faced a robot failure that required a restart, the researcher asked the participants to describe if the interruption had any effect on their task-teaching process at the end. Participants P67 and P92 said that the interruption itself did not have any effect but decided to describe their alternative plan in retrospect on being asked that question.

> "*I realize in retrospect that the instructions I gave it, when I said it to bake the oven, I should have gone from turn on the oven, instead of to bake the pizza, to just turn on oven, wait for 30 seconds, just keep it simple.*" (P67)

7.8.1.4 Participants provided ideas about how to make the task easier to teach

When participants encountered issues while teaching the robot, they suggested ways in which the teaching process could have been made easier for them.

Participants (P23, P31, P49) suggested potential updates in the interface and the environment that would have helped them in their teaching process. P31 said the following in response to the "The put-down task failed" message on the interface.

> "*I think I should be checking up here more to make sure that it worked or not, 'cause I think an alert that would've said failed might have helped me see it quicker then, 'cause I'm not looking up here as much.*"

When P23 and P26 encountered robot failures, they suggested unavailable alternative ways using which they could have recovered from the failure.

> "*I would try finding somewhere to override the statements and be able to do it with a more complex structure. Or try finding a help in terms of the menus of how do I give it full instructions.*" - P23

When P56 could not be certain that the pizza had been cooked, they suggested tests that they could do to determine that: "*Hmmmm maybe controlling the waiting time like I waited there 30 seconds ...but maybe we can test like applying various kinds of time periods during the oven and checking whether the color might be changed.*"

7.8.2 Participants found alternative ways to instruct when they could not execute desired instruction

Most participants created a teaching plan once the task was introduced to them. However, some of the parts of the plan could not be realized because the desired options were not present or the instruction form was not supported. Participants switched strategies and found alternative ways to still execute their plan.

7.8.2.1 Participants changed their plan on encountering limitations or unexpected responses

Participants typically switched templates to access a specific instruction type, but two participants also switched templates when they could not find their desired option in the currently selected one. When P23 could not find a time-related option in the drop-down list, they decided to provide a command from a different template.

> "*but again I can't figure out here in this list the time variable. So I'm still like prepare the pizza for, it keeps saying invalid input. So I'll do, prepare the pizza maybe let's see if it gives me something.*" (P23)

Two participants also changed their teaching plan in response to the robot not moving in response to an instruction. P7 wanted to repeat the command `Approach the table` in anticipation of picking up the pizza that's on it. However since the robot did not move before in response to the command, they decide to skip to the next action. P7 said, "*Assuming I don't need to move just because I haven't moved, moved between the oven and the table at this point. So we'll just go straight to picking up the pizza.*"

Nine participants also changed their planned teaching process in response to the robot failing to do the previous instruction. They looked for and sent an alternative instruction to replace the instruction that failed. P78 encountered the error "The put-down task failed" because the oven was closed when they tried to put the pizza in the oven.

"*Oh, it failed. Lemme try another one. Oh, take the pizza to the oven. Maybe I should do that first.*"

7.8.2.2 Participants used alternative actions or goals to teach the desired task

As a part of their changed plan, when six participants could not teach their desired task, they found alternative action or goal options to teach the same task.

Participant P11 wanted to set the temperature so decided to change the task from `Bake` to `Heat`.

> "*So I want to tell it how long and how hot to bake it for....Okay I wanna go back to teaching it to heat, I guess instead of bake.So heat the pizza. So that's essentially what baking is heating it.*"

P92 wanted to preheat the oven and on scrolling through the options, they encountered the action `Turn on` and decided to use that instead.

> "*Okay wait, I see now it says turn on but does it turn on and preheat instantly? I don't know. Okay, now I'm going to turn the oven on.*"

On encountering a request for goal or sub-task from the robot about the task `Bake`, P23 provided a teachable action `Cook` that is interchangeable with the original action and provided the instruction `Cook the pizza in the oven`.

> "*What is the next goal of bake? What is the goal is to cook the pizza for me.*"

7.8.2.3 Participants used alternative options when they could not find their desired option

Participants planned the commands that they wanted to provide in advance but the options were not always available. In response, they looked for alternative options in the drop-down list to provide a different command for the same intention.

Multiple participants (P11, P23, P26, P31, P78, P92) changed their plan when the desired option was not found among the different drop-down lists across the templates. They would typically scroll through the different lists available to find a suitable option.

> "*as guess I'm getting stuck in that these other options I have to be able to command don't have a temperature or time, well I guess, okay, I can teach it heat.* (P11)

While looking for an alternative, four participants scrolled through the list with the intention of finding one that is the closest to their desired command. When P49 could not find the options to provide the command `Cook the pizza for 350 degrees`, they provided an alternative command `Cook the pizza in the oven` instead.

When participants P11 and P92 could not find a 30 second option in the drop-down lists, they deferred to using the button command `Wait for thirty seconds`.

7.8.2.4 Participants engaged in creative actions in response to unavailable options

On not finding their desired option in the drop-down lists, participants P11 and P23 expected that teaching specific actions would update the drop-down list with new action-relevant options.

> "*Okay, the more complicated ones don't seem to have numbers preset for the temperature. So I wonder if I need to teach it bake and that will allow me to set the temperature.*" (P11)

On not finding the option to enter 350 degrees, P23 decided to send an incomplete command and encountered empty input errors as a result of that: "*So I'm still like prepare the pizza for, it keeps saying invalid input.*"

Both P26 and P31 asked if they could enter their own commands and tried to type them in response to not finding their desired options ("*I can't add something new, right? Or can I type something new?*" - P31). The interface however did not allow entry of new values not in the list.

7.8.2.5 Participants were immersed in the teaching process

The previous set of examples shows that the participants were immersed enough in the teaching process that they were willing to find alternative ways through which they could successfully teach the task.

Nine participants identified with or personified the robot while teaching it the task and waiting for it to perform in response to the instructions. They assigned gendered pronouns to it and sometimes even human emotions when it failed or did not do the task successfully.

> "*Well, let's try turning off the oven and hope it doesn't hold that grudge against me.*" (P67) "*It's probably confused 'cause my instructions (laughs) are really...* (P49)

Even when faced with a situation where the robot had to be restarted, all the participants were willing to continue teaching the robot after the restart. This was also evident when they were stuck in the teaching process and unsure how to proceed. P99 expressed regret that they missed out an opportunity to teach it a new teachable task rather than just providing a primitive action.

> "*Oh man, I could have taught it Heat the oven, which would've been turn on the oven.*"

P26 intended to teach it task aspects outside of those explicitly specified in the task description: "*I know this isn't like, well, realistic, but I should probably close the fridge.*" Participant P7 also took time to verify the correct input before sending instructions.

We also observed P67 matching the language used across the interface and task instructions while thinking out loud.

7.9 Quantitative Analysis

We characterized the actual teaching interactions that took place between the participants and the robot. We identify the various types of commands provided, as well as provide a quantitative description of the failures encountered during the task.

7.9.1 Task Command Analysis

13 out of 14 participants successfully taught the task to the robot. P49 almost finished teaching the task, but after the robot performed a set of unexplainable actions in response to the command `Approach the counter`, P49 was no longer certain about how to proceed with the task.

Participant	Completed Task?	Teachable Actions	Teachable Action List	Primitive Actions	Goal Commands	Wait for 30 seconds	Why was it used?	You are done	Was it used correctly?	Total Commands
P7	Yes	0	N/A	16	0	1	cook for 30	1	No (end full task)	18
P11	Yes	4	Bake, Bring, Empty, Heat	6	3	1	cook for 30	0	N/A	14
P23	Yes	9	Bake, Cook(6), Deliver, Prepare	25	2	0	N/A	1	No (end full task)	37
P26	Yes	7	Bake, Heat(5), Start	48	2	3	cook for 30	2	No (misclick, end full task)	62
P31	Yes	0	N/A	15	0	1	cook for 30	0	N/A	16
P47	Yes	1	Bake	20	0	2	cook for 30 in open oven then closed oven	1	Yes	24
P49	No	8	Bake, Cook(4) Fetch(2), Start	16	0	0	N/A	1	No (end full task)	25
P56	Yes	0	N/A	17	0	1	cook for 30	1	No (end full task)	19
P59	Yes	0	N/A	12	0	2	turn on oven for 30 cook for 30	1	No (end full task)	15
P67	Yes	2	Bake(2)	21	10	1	cook for 30	0	N/A	34
P74	Yes	0	N/A	14	0	1	cook for 30	0	N/A	15
P78	Yes	4	Carry(4)	17	1	1	cook for 30	0	N/A	23
P92	Yes	4	Bake, Fetch, Heat(2)	27	0	1	cook for 30	2	Yes	34
P99	Yes	2	Cook, Heat	17	0	1	cook for 30	0	N/A	20

Table 7.1: Breakdown of commands provided by participants for the task during the study

We provide a breakdown of the type of commands that participants provided as a part of the task in Table 7.1. 9 out of 14 participants used teachable actions as a part of their teaching process. The number of commands provided by these participants is on average higher than participants who did not provide teachable action instructions. However, the advantage of teaching teachable actions is likely to show through when multiple higher-level tasks are taught that leverage previously taught teachable actions, as has been shown by evaluation of communication in James Kirk's thesis [25]. In this study, since only one task was required, participants who used only primitive actions engaged in shorter interactions. However, P11 was an exception since they used 4 teachable actions, and yet used the least number of commands to successfully teach the task. Only 5 participants used goal commands as a part of their teaching interaction.

All participants that used the `Wait for 30 seconds` command used it to cook the pizza for the prescribed amount of time. P23 had taken a significant amount of time trying to find the option 30 seconds in the template and eventually determined that since more than thirty seconds had passed as they figured the commands out, the pizza ought to have been cooked by then. Out of the 8 participants who used the `You are done` command, only 2 participants (P47 and P92) used the command appropriately to end the process of teaching a teachable action. The other participants used `You are done` to indicate the end of the complete task rather than a specific teachable action. This clearly shows that this command needs to be modified to be more natural to use appropriately by the human teacher.

Participant	Total Number of Turns	Unknown Word	Sub-task Failed	Rosie didn't understand command	Unknown Failure	Instructor Invalid Command	Researcher Fault	Number of Task Failures	Number of Failures that led to Restart
P7	36	0	0	0	0	1	0	1	1
P11	29	0	0	0	0	1	0	1	0
P23	73	1	0	2	1	0	0	4	2
P26	123	1	0	2	0	2	1	6	4
P31	37	0	3	0	0	0	0	3	0
P47	48	0	0	0	0	0	0	0	0
P49	51	0	0	1	5	0	0	6	2
P56	39	0	1	0	0	1	0	2	1
P59	32	0	2	0	0	0	0	2	0
P67	69	0	0	5	1	0	0	6	1
P74	31	0	0	1	0	0	0	1	0
P78	47	0	1	0	2	0	0	3	2
P92	71	1	2	0	0	0	1	4	2
P99	41	0	0	0	0	0	0	0	0

Table 7.2: Breakdown of task-level failures encountered by participants during the study

7.9.2 List of Robot Failures and Unexpected Behaviors

In this subsection, we provide a list of robot failures and unexpected robot behaviors that occurred across participant interactions that confused participants and indicates a need for a better explanation for these behaviors.

A portion of the unexpected behaviors occurred while providing primitive actions. This was a result of how the low-level preconditions were defined for the `pick-up` and `put-down` actions.

The participants were informed that the robot could handle only one object at a time. Therefore, participants expected that the robot could not do any other task while holding an object, such as a pizza. However, when provided an `Open` command, the robot was able to open the receptacle while holding the pizza.

When participants (P11, P23, P56) provided the command `Pick up the pizza` while the pizza was still in the fridge and the fridge was closed, the robot automatically opened the fridge to pick up the pizza without needing to be instructed to do so. When the pizza was on the floor, P23 provided the command `Put down the pizza into the oven`. The robot automatically picked up the pizza to put it inside the oven.

Participant P49 encountered unexpected robot actions in response to valid action commands. P49 was teaching the task `Cook the pizza` and had provided a set of commands that contributed to the task. However, the robot seemed to have reached an unresponsive state and displayed incorrect behavior when P49 provided the command `Approach the counter`. Instead of moving towards the counter, it performed other unrelated actions including going to the oven and picking up the pizza. One hypothesis for this behavior is that it was performing the partially taught task as a result of being stuck in its learning process.

We now describe a list of failures that participants encountered as a part of the teaching process.

1. **Unknown Word**: Some of the verbs in the list were not part of the robot's vocabulary. When participants (P23, P26, P92) encountered this error, the robot would respond with "unknown-word" and reach an unresponsive state. Therefore, the robot had to be restarted in response to this failure. Participants P26 and P92 encountered this failure as a result of using the command `bake`. The researcher added this to the vocabulary after this error was encountered by these two participants since it was relevant to the task.

2. **Sub-task failed**: When participants tried to put the pizza in a closed oven, the task failed since the robot needs to be explicitly instructed to open the oven first. The robot responded with "The put-down task failed" since it could not do this action successfully.

3. **Rosie did not understand the command**: In response to specific commands such as `The goal is that the oven is on` or `Start the oven`, Rosie responded with "I don't understand" since it could not process the sentence successfully.

4. **Invalid Command**: When the participants provided an invalid command such as `Open the pizza`, the robot simulator provided the reason for failure on the command line; however,

it resulted in the robot going into an unresponsive state. This required a restart of the robot agent.

5. **Unknown failure**: There were a few times that the robot reached an unresponsive state or shut down the interface without any reasonable explanation.

6. **Researcher fault**: After restarting the robot, the researcher used a combination of commands and direct manipulation of objects to restore the agent to its previous successful state before the failure that required the restart. However, while doing this for participants P26 and P92, the researcher entered two commands consecutively without waiting for a robot response. This resulted in the robot going into an unresponsive state, requiring another restart of the robot.

7.9.3 Task Failure Analysis

We provide a breakdown of the various task-level failures encountered by the participants in Table 7.2. 12 out of 14 participants encountered at least one task failure during the teaching process. We described the types of failures presented in the table in the previous Section 7.9.2. Among the multiple types of failures, unknown-word failures and invalid command failures always resulted in the robot reaching an irreversible state and thus required a restart of the robot. P26 faced the largest number of task failures that required a robot restart, and this is also reflected in the fact that they had the maximum number of turns to finish teaching the task successfully. P67 tried repeating the same goal command multiple times even when the robot responded that it did not understand the command, leading to a larger number of task failures. P31 encountered the same "put-down task failed" error three times before determining the actual reason for the failure and providing the "`Open the oven`" instruction before putting the pizza in the oven. The various failures encountered by the participants led the robot developers to discover edge cases that we had not encountered before, leading to a better understanding of the robot's capabilities and limitations.

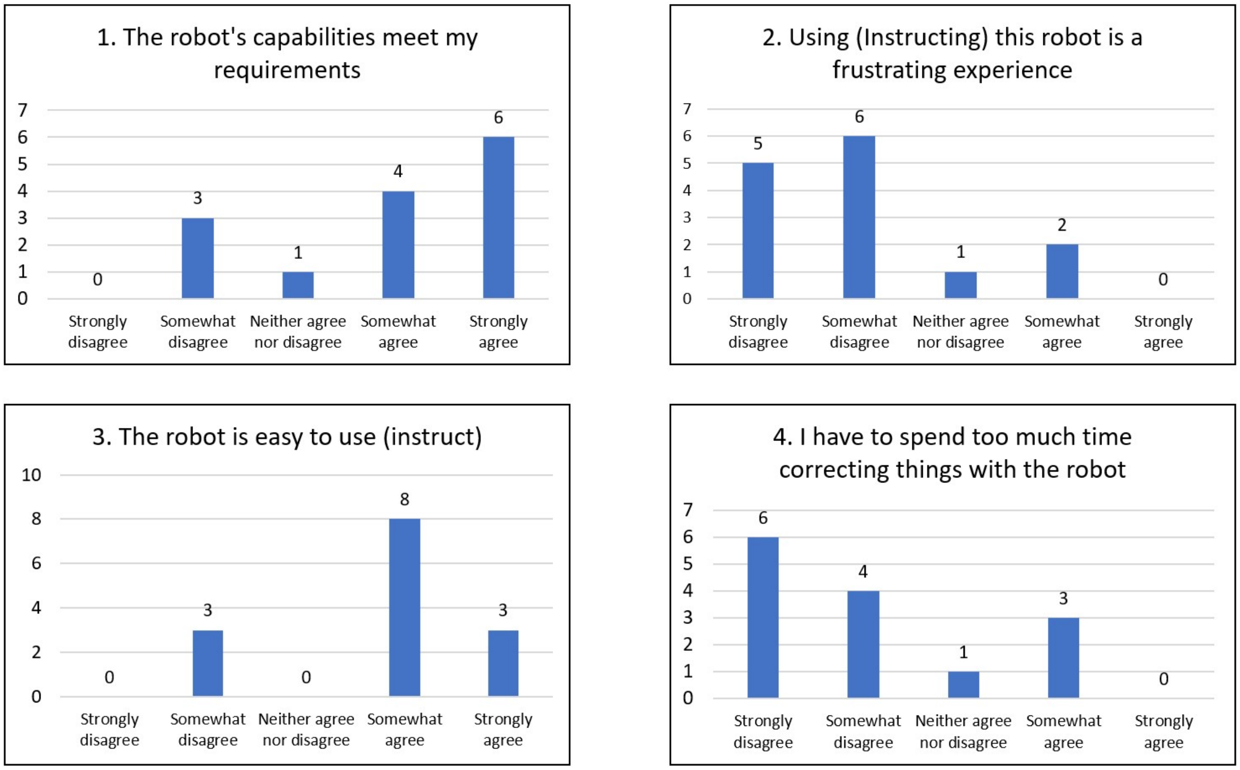


Figure 7.1: Participant responses to the UMUX Likert scale questionnaire

7.10 Usability Metric for User Experience Results

We asked participants to fill out a 4-question UMUX Likert scale questionnaire whose results are provided in Figure 7.1. We observe that at least two-thirds of the participants agreed with the positive statements (1 and 3), and disagreed with the negative statements (2 and 4). The participants who did not fall under this category were participants such as P23, P26, and P49 who encountered multiple task-level failures with the robot, which indicates the need for improvement in the system which we identify and propose in Section 9.3.

7.11 Limitations

While the analyses in this dissertation provide us an extensive understanding of a non-expert situated teaching process, it is only an initial exploration into the process.

While Rosie is capable of understanding many more complex instructions, the templates we provided were limited to 4 types of instructions for initial exploration. The instruction in this study was restricted to verbal modality alone due to the restrictions in the robot. Allowing the participants to engage in other multimodal inputs such as gestures would have enabled more natural human-robot interaction.

Given that this study was set in a simulated environment, while our results can be extended to the real world, we are likely to get different insights when conducting this study in a real-world kitchen environment.

There were 8 participants where the robot failed, and we had to restart the robot teaching process. However, the researcher reset the robot and the corresponding environment to the last successful state before the failure using a combination of previously used commands and manually modifying the environment. For participants who encountered this restart, at the end of the task, the researcher asked them to describe if and how the interruption affected their task teaching process.

For the first two participants, the researcher had unintentionally missed adding "bake" to the robot's vocabulary. Given the necessity of "bake" to the task, the researcher added it to the robot's vocabulary for the remaining 12 participants. The researcher had also unintentionally mislabeled the water cooler just as a "cooler" in the environment. The researcher modified the label after this encounter for the remaining 12 participants since it did not modify the conditions of the actual task.

CHAPTER 8

Discussion

In this chapter, we situate and organize our findings using our previously identified research questions which are as follows;

1. How do non-experts teach hierarchical tasks to an ITL robot?

2. Do instruction templates contribute to efficient and effective teaching?

3. What are some salient problems that non-experts face that can be solved by extensions to the robot's interaction capabilities?

In the following sections, we summarize our findings from Chapter 7 to answer each of the above research questions. We then use the findings that answer the question about salient problems faced by participants to propose suggestions for robot interaction mechanism design that can help improve the human-robot teaching process in Chapter 9.

8.1 How Do Non-Experts Teach Tasks?

Complex Strategies: Participants naturally developed complex strategies to teach the robot a defined multi-step hierarchical task. They decomposed the task into individual parts as a part of their teaching process (Section 7.4).

Used Multiple Sources of Knowledge: Participants leveraged their real-world knowledge as well as used the different types of knowledge (interface, environment, robot) available for the task to aid their teaching process (Section 7.5). Participants also actively updated their knowledge from different sources as a part of their teaching process (Section 7.5.2). Participants reasoned about the robot's actions, movements, and responses that contributed to their mental model of the robot (Section 7.6.4.2).

Engaged in Effective Teaching over Time: As they explored the teaching setup throughout the task, participants were able to effectively translate their teaching plan into corresponding instructions for the robot (Section 7.5.3.1). Participants were able to modify their teaching steps on the fly using the knowledge available to them either by changing the overall plan, or by easily switching between templates to provide a different set of instructions (Section 7.5.3.2).

Evaluated Robot Behavior: Participants actively evaluated the robot's progress through a combination of its actions and responses (Section 7.5.4). Participants also noted when the robot failed or did not respond as expected.

Task Completion: No incentives were provided to participants to complete the task or to complete the task efficiently (in terms of time or number of interactions). Yet, we observed that participants were motivated to continue teaching the task, even when faced with failures and robot restarts (Section 7.8.2).

Instruction Efficiency: Participants displayed an inclination to teach efficiently. While a few participants explicitly mentioned this inclination (Section 7.8.1.3), it was also evident in how participants ordered their instructions (e.g., providing an instruction based on the proximity of the robot to the referred object).

8.2 Are Instruction Templates Effective?

Instruction templates did contribute to efficient and effective teaching. 13 out of 14 participants were able to teach the complete task successfully, directly providing instructions to the robot.

Effective Template Use: While we do not have direct evidence from this study, our prior study [47] has shown that people use multiple types of sentence forms while teaching a task. The robot cannot successfully parse these sentence forms and they would lead to comprehension failures. The templates enabled participants to effectively translate their desired plan into valid instructions, without the necessity to additionally determine the robot's language comprehension ability. This is because the template instructions already followed a valid format, and there were predefined options available for the participants to choose from, rather than having to determine these from scratch. The predefined options for each part of speech also avoided typographical errors in addition to serving as an additional source of information about the robot. Participants confirmed the object names and states of the objects (e.g., pizza) using the lists available. Participants actively leveraged this template format and the available predefined options to determine their teaching plan (Section 7.5.1.2).

Desire to Provide Complex Instruction: Due to the limited nature of templates, some participants could not find the options to provide their desired instruction. For example, multiple participants looked for a preheat action among actions and "30 seconds" under the noun list that did not exist. Participants also wanted to provide instructions with compound sentences or sentence structures not supported by the current templates (Section 7.7.2.6). However, they found alternative ways (section 7.8.2) to provide the same instruction, but only after some significant effort.

8.3 Salient Problems Faced during the Teaching Process

In the process of teaching, participants encountered difficulties through their process of interaction with the robot. Some of these were a result of the participant's incomplete mental model of the robot that was not updated properly as a result of insufficient updates from the environment and the robot. This led to participants displaying a lack of confidence in their ability to teach and the robot's ability to learn. This lack of confidence was exacerbated when the robot ran into failure or inefficient situations, resulting in incorrect updates to the participant's mental model of the robot in terms of its knowledge and capabilities.

8.3.1 Issues with the Interface

We summarize the difficulties people faced while using the interface described in section 7.6.1.

Interface Shortcomings: Some of these issues were due to the inherent design of the interface. When participants selected an option in the drop-down box, it did not have sufficient visual indicators to mark its selection requiring the participant to confirm the selection multiple times. Since the options were close together, this lack of visual indicator also led to two participants sending an unintended command due to wrong selection. When participants selected partial options on a template and switched to another template briefly before coming back to the original one, the selections were no longer available. Even though participants were told to wait for the robot's response before providing the next instruction, two participants provided consecutive commands without waiting for the robot's response. Multiple participants were curious about why they would use the `Wait for thirty seconds` command when doing it as a part of the training task.

Lack of Interface Knowledge: While the templates changed when the corresponding template buttons were pressed, there was no way to tell which template was currently active since the buttons did not stay pressed, providing no indicator to the participant. It was not clear to participants under what conditions the goal templates would be activated, since by default they were inactive. This lack of knowledge led participants to make the wrong assumptions about what triggered the goal template activation (Section 7.6.1.4). Some participants were also unaware or discovered later that the text area in the interface could be used to track the interaction that had taken place.

8.3.2 Insufficient Environment Update

Insufficient Visual Indicator: Participants did not get sufficient visual updates from the environment to confirm specific events (Section 7.6.2). When the robot had to press the button on the water cooler as a part of the training task, there was no visual indicator of the button being pressed. Similarly, once the participants had finished providing the instructions to cook the pizza, there was no visual change to the pizza object to indicate that it was successfully cooked.

Desire to Confirm Environment Changes: In addition to visual indicators, participants wanted the ability to explicitly confirm that these changes took place in the environment (Section 7.6.2.3) after the robot had performed the action. When the oven displayed 350 on top when turned on, the 350 display alone was not sufficient for participants to know that the oven was already at 350 degrees. Participants were unsure if the oven would stay at 350 or if it had a preheat function that would automatically preheat it to 350 degrees. Participants also wanted a way to confirm the states and locations of objects with the robot (e.g., a ready-made pizza is in the fridge).

8.3.3 Insufficient Task Update

Participants explicitly wanted ways to confirm the success and completion of an action by the robot, but these ways did not exist.

Insufficient Robot Response: Participants found the robot's response "I am ready for a new task" after it had completed the previous action insufficient. After the robot performed the `Wait for 30 seconds` command, it said, "I am ready for a new task" but some participants were unsure it actually waited for 30 seconds. Given the lack of any other information, participants resigned to using this response as a way to move forward (Section 7.7.2.3).

The ambiguity inherent in the response also led to confusion and mistaken assumptions for the participant (Sections 7.6.3.2, 7.6.3.4).

Uncertainty about Task Aspects: Participants at times were unsure about which commands they provided last (Section 7.7.1.4). Participants were also unsure about what aspects of the task were crucial for Rosie to learn (e.g., is closing the fridge door part of the bake task?). Even after providing instructions and observing visual feedback or receiving a response from the robot, participants were still unsure whether specific aspects of the task were executed properly (Section 7.7.1.1).

Need for Robot Action Confirmation: Sometimes, despite having visual indicators for the action in the environment (Section 7.6.3.3), participants wanted additional ways to confirm that an action had been done. Participants wanted to confirm the robot's action when it acted in a way that differed from their expectations (Section 7.6.4.3). Participants also wanted a way to ensure that the robot could do the action before instructing it to do that action (Section 7.6.3.4).

8.3.4 Incomplete and Incorrect Mental Models of the Robot

Participants displayed incomplete and incorrect mental models of various aspects of the robot and their interaction with it. We describe them in detail below.

Lack of Knowledge about Robot's Physical Capabilities: We found participants were unsure about the physical properties of the robot (e.g., the number of arms of the robot) and the corresponding capabilities of the robot (e.g., can the robot open the oven while holding the pizza?). While some of this can be attributed to the robot in the simulated environment, even in real life, participants are not likely to be familiar with the physical properties and abilities of the robot and would require explicit information about the robot.

Unmet Robot Task Knowledge Expectations: Participants expected the robot to already know how to do several actions that would be considered typical in everyday life (e.g., Carry the pizza to the oven). However, these actions were teachable actions that the robot was yet to learn. Participants also expected the robot to have common-sense knowledge about task expectations, such as opening the oven to do something that should always be followed by closing the oven. Rosie needed to be explicitly instructed to close the oven.

Insufficient Proof of Internal Actions: While participants got explicit proof of their command's success when the robot manipulated objects in the environment, this was not true when there were visual changes in the environment that the robot needed to be aware of (e.g., oven displayed 350 when it was turned on) or when the robot performed internal actions such as "waiting for thirty seconds." These situations often left participants unsure about the robot's action success, and they wanted ways to explicitly confirm the success.

Incomplete/Inaccurate model of Robot Learning Process: Multiple participants did not understand the robot's repeated requests for a goal or sub-task as a part of the teaching process. Instead, they assumed it was because the robot did not know the goal yet (even though it had already learned one in a previous instruction) or assumed it was a failure response. Participants also repeated the same instruction even after the instruction was completed in hopes of a different response from the robot. (Section 7.7.2.2).

Incomplete/Inaccurate Model of Robot Instruction: While the robot can learn complex task representations, not all participants leveraged this capability. While people using primitive actions can be partially explained by the need for efficiency to do this one task, almost a third of the participants did not teach any new actions.

Participants required additional reference to the study document to figure out how to teach a teachable task (Section 7.7.1.4) or sought clarification about it from the researcher (Section 7.7.1.5). It was not clear to participants when and how to end a teachable task teaching process. This often led participants to provide instructions for a new task without explicitly ending the previous one. Even though the command `You are done` was meant to be used to end the teaching of a teachable task, all participants but one who used it, used it to end the complete baking task (Section 7.7.2.1).

8.3.5 Failures Encountered during the Teaching Process

As a result of insufficient information about the task and incomplete and incorrect mental models of the robot, participants encountered failure scenarios during the teaching process. The lack of clarity about the reason for failures also contributed to participants incorrectly updating their mental models of the robot. This contributed to some participants finding the teaching task difficult and also led to mistrust in the robot's ability to do the task.

Encountered Failures: As a part of their teaching process due to either unexpected robot behavior or as a result of their incomplete or incorrect mental models, participants encountered inefficient or failure scenarios as detailed in section 7.9.2. Sometimes, participants realized how their instructions were imperfect only after providing the instruction (Section 7.8.1.2).

Unclear Reason for Failure: When participants ran into failure situations with the robot, the robot often provided some response, but the reason for the failure was still not clear to the participant (Section 7.6.4.6). This often led participants to make mistaken assumptions about the robot's knowledge or learning mechanisms. Additionally, when some of these failures occurred in the middle of a multi-step task, the source of the failure was not clear to the participants. These mistaken assumptions unless corrected immediately will only hinder the interaction process from there on.

Difficulty in Teaching Task: The incomplete and inaccurate models of the robot's instruction and learning process led participants to have a difficult time teaching the robot. Some of the participants faced this because they could not find the desired options in the drop-down list or the templates to translate their plan into appropriate instructions (Section 7.7.2.5). They attempted to provide commands multiple times but would encounter the robot behaving in an unexplainable manner, or running into a failure situation. This led them to express frustration and uncertainty about the process (Section 7.7.2.4).

Mistrust in Robot's Ability: The combination of the robot encountering failures and not getting sufficient updates on performing an action led participants to mistrust the robot's ability to learn or perform a new task (Section 7.7.2.3).

We address these problems in the next chapter and propose interaction implementation that can help humans be more competent and feel more confident in a hierarchical task-teaching process.

CHAPTER 9

Interaction Design Proposal

In Section 8.3, we saw that participants faced multiple problems while instructing the robot to perform the baking task. Participants were sometimes unable to provide their desired instruction. Participants did not always get sufficient information from various sources of knowledge, leading to incomplete and incorrect mental models of the robot. Participants also encountered failure scenarios where the robot did not perform the expected task.

In this chapter, we address those problems and propose interaction mechanisms that can help resolve these issues during a teaching interaction. This proposal aims to leverage the fact that the robot has access to its own instruction, knowledge, capabilities, and task progress, including success and failure. Therefore, using these interaction mechanisms, the robot can help improve the teaching process for the non-expert instructor.

In Section 9.1, we describe interface changes that can be made to improve the instruction experience for the instructor. Additionally, we describe how the robot interaction can be extended to handle more complex instructions. In Section 9.2, we describe how the robot can provide information about its environment, task, instruction, and learning process, as well as its knowledge and capabilities, to the instructor to help them build better mental models of the robot. In Section 9.3, we discuss primitive-level and hierarchical-task-level failures in detail and then describe the implications of such failures and their recovery on the overarching teaching process. In these three sections, for problems that cannot be sufficiently addressed by interaction mechanisms, we recommend research directions that can be pursued to understand these problems at a greater depth.

9.1 Helping the Instructor Interact Better

While templates allow for a successful teaching process, we identified **interface shortcomings** that ought to be fixed for the future. For option selections in the drop-down list and the template button selection on the interface, the selection must be highlighted or actively appear pressed to provide clear evidence of selection to the instructor. To prevent the instructor from sending the wrong command due to incorrect selection, there needs to be a text area that displays the instruction to the instructor as they select the options so that they can confirm their instruction before sending it. To ensure that participants wait for the robot's response before sending the next command, the "send" button could be disabled until the robot's response has been received.

We also observed that participants had the **desire to provide more complex instructions**. The language that Rosie can comprehend is structured and follows a predefined format but it is also a lot more extensive than the four instruction templates provided in our interface. Future work can include iterative studies to find the best way to extend the templates to support more complex instructions.

While templates provide a useful structure that allows participants to provide **effective instructions**, in cases where people find templates insufficient and find it **difficult to teach the task**, we could build in an option to provide free-form instructions. Given Rosie's current language comprehension abilities, we can leverage current large language models (LLMs) to convert natural human instruction into a Rosie-understandable structured format. If the provided instruction cannot be directly converted into a corresponding valid instruction, Rosie can use the content in the original instruction to suggest a similar yet Rosie-understandable instruction to the human teacher. For example, let us assume that a participant wants to provide the command `Heat the oven up to 350 degrees`. Rosie is not set up to understand this specific command. However, Rosie can understand the command `Heat the oven`, so one could use an LLM to extract this structure from the provided command. To help the participant continue the teaching process, Rosie can use its semantic memory to find commands related to the "oven" and "heat" to then provide a suitable valid command such as `Turn on the oven`. While there has been some promising work looking into how LLMs can be used to translate natural language instructions to structured planning language for AI planners [60], initial results indicate the need for expert involvement to account for situations where the LLMs are not successful. Therefore, more research is required to understand how one can leverage LLMs for non-expert instruction of robotic systems.

9.2 Helping the Instructor Build Better Mental Models

While participants were successful at teaching the task, they were still a) uncertain about environment changes and the task progress and, b) demonstrated a lack of knowledge about the robot in terms of its task knowledge, physical capabilities, instruction, and learning process as described in section 8.3. However, the robot has access to its own knowledge while also possessing knowledge about the environment and the task that it is currently learning. Therefore, in this section, we describe how the robot can help the instructor build a better mental model of the task, environment, and robot by providing relevant information.

9.2.1 Provide Environment Information

We observed the need for more **visual indicators** for two events in our study, namely, the button being pressed and the pizza being cooked. If we use the same tasks in future studies, we could extend the simulated environment to provide an explicit change in the display of the button using both form and color. Similarly, we can make changes to the pizza object so that its color is changed appropriately after being in a hot oven. While these examples are particular to our study, these observations clearly indicate that participants look for visual indicators in response to actions.

In addition to visual indicators, the robot could also include this information as a part of its task-completion response. In our study, when the oven visual is updated to show the temperature to be 350 after being turned on, the robot could say: "The oven is now at 350 degrees. I am ready for a new task." However, providing such information for every environment update can become excessive, especially in a complex environment with multiple people and entities. Therefore, it would be pertinent to study what changes are relevant based on task context and the interaction up until that point, as well as the instructor's knowledge.

Participants also wanted to explicitly **confirm the changes that took place in the environment**. Since the robot has access to knowledge about the environment and its own processes, it should be able to answer questions about updates to the environment (e.g., what is the temperature of the oven?). Multiple participants looked for a "preheat" function or assumed the oven was preheating when it was turned on, even though the oven in our study did not have a preheat capability. If there are appliance-like objects in the environment that have more complex functions, the robot could have access to this information and be capable of answering questions about these functions.

9.2.2 Provide Task-Related Information

When participants were **uncertain about different task aspects** and found the **robot's response insufficient**, they felt the **need to confirm the robot's actions**. Therefore, Rosie could use its current state knowledge and episodic memory to answer questions about its previous actions. Rosie can also proactively provide this information to the instructor. For example, if the instructor has taught a few actions as a part of a larger task and is **uncertain about the task aspects**, Rosie can summarize what it has learned up until now to help the instructor visualize the task taught until that point.

Our study also shows that it would be useful for the robot to provide explicit information when it performs any **internal action** or computation. To enable this, the robot must possess an internal taxonomy differentiating between internal and external actions and proactively provide information about internal actions since the instructor does not have access to that information otherwise. Besides those observed in our study, these can extend to scenarios where the robot creates an internal plan to determine how to do a particular multi-step task. While the instructor will have the opportunity to observe the actions in real time, knowing the robot's plan in advance can help bolster the instructor's confidence in the robot's ability to do the task.

9.2.3 Provide Instruction and Learning Process-Related Information

We intentionally provided a simple training session that did not involve teaching the robot a new task since we wanted to study how much participants could grasp the robot's learning process through interaction alone. Through our findings, we discovered that this was insufficient since participants still displayed an **incomplete and incorrect mental model of the robot instruction and learning process**. Participants clearly require more help to truly leverage the complex learning capabilities of the robot.

One way to achieve this would be to provide a more intensive training session that allows the participant to explore the various learning abilities of the robot. We could then study the effectiveness of such a training session and the participants' willingness to leverage the robot's learning process by conducting a study. This study would ask participants to instruct the robot to do similar tasks consecutively. For example, the first task would be to heat a pizza in the oven for 30 seconds, whereas the second task would be to heat bread in the oven for a minute. Given the importance of efficiency to human teachers, doing this study will also allow us to evaluate the interaction using the number of turns taken to teach the two different tasks.

However, initial training alone will not be sufficient since we observed that participants needed additional reference (Section 7.7.1.4) during the task to remind themselves of the robot's learning process. This is where we can take advantage of Rosie's knowledge of its learning mechanisms. Mohseni et al. [43] study the role of task-grouping suggestions provided by a robot on the teaching process of non-experts when teaching a hierarchical tire rotation task. They find that people expend lesser teaching effort and achieve higher teaching efficiency when the robot provides task-specific suggestions in comparison to when it provides no suggestions. In a similar vein, we need to build scaffolding in Rosie's interaction model that allows it to provide relevant information and teaching-specific suggestions during the teaching process. For example, if the instructor is providing instructions to teach a new task `Empty` within the higher-level task `Heat`, Rosie could explicitly tell the instructor, "I am currently learning the task `Empty` under the task `Heat`." Rosie could also remind the instructor that they can use the command `You are done` to end the teaching of the current task.

9.2.4 Provide Robot Knowledge and Capability Information

As described in chapter 3, there has been significant prior work studying how to make robots more transparent to humans. Through our study, we identified specific situations where people displayed **lack of knowledge about the robot's physical capabilities** and had **unmet task expectations** of the robot. Given people's lack of familiarity with robots, it would benefit the instructor if Rosie could use its interaction capabilities to describe its knowledge of the environment and the affordances available to it.

This could take the form of Rosie describing the objects it can see when it enters a new room and occasionally giving examples of how it can manipulate different objects at different times (e.g., I can open and close the fridge).

Rosie is also capable of answering questions about its task and environment knowledge. This capability could be extended to encourage the participants to ask questions when they want to compare their expectations of Rosie's knowledge of a task to its actual task knowledge.

To do this effectively, we would have to conduct studies where participants interact with the robot long-term to understand what information is relevant at what times and how often it must be provided until the instructor can be confident about the robot's knowledge and **trust in its ability to do a task**.

The robot can provide even more relevant knowledge to contribute to the instructor's mental model if it possesses a mental model of the instructor. This can include modeling the instructor's mental states [3, 5], uncertainty [55], and even the instructor's mental model of the robot [8]. We could leverage existing mental modeling techniques [54], especially from those that have modeled human cognition and behavior [21].

9.3 Helping the Instructor Recover from Robot Failure and Inefficient Scenarios

As described in Table 7.2, participants faced multiple types of failures in their teaching process. Participants had a few suggestions on what would have helped them when faced with a failure (Section 7.8.1.4). This included an alert popup letting the instructor know that the robot had failed, and also having access to a help menu to get help in providing instructions.

We observed both primitive command-level failures and task-level failures. Failures at different levels of the task structure naturally require different types of responses at different granularities depending on the expertise of the human interacting with the robot.

9.3.1 Primitive Command-Level Failures

Given our focus on non-experts, the low-level failures present an interesting problem. Five participants encountered the "The put-down task failed" message as a result of the robot being unable to execute the `put` or `put down` task. The put-down task is defined as follows:

$$\text{put-down}(a, \text{in}(b)) \text{ if grabbed}(a) \hat{}\ \text{receptacle}(b) \hat{}\ \text{openable}(b) \hat{}\ \text{is-open}(b)$$

So, Rosie can put down an object a in object b only if object a has been grabbed by the robot and object b is a receptacle that can be opened and is currently open. In our study, all participants encountered this failure because the oven was not open when they tried to put the pizza inside the oven. But the robot's message was insufficient as the participants were still **unclear about the reason for failure**. For this particular command failure in Rosie, since this command has a precondition that was not satisfied, one solution would be to describe the unsatisfied precondition to the participant as a way of explanation that the participant can use to instruct accordingly. Das et al. [15] show that providing context-based-history explanations for internal failures (failures not observable in the environment) significantly improves the human's ability to identify a solution to resolve that failure compared to no explanation.

But the solution is not as simple if the robot encounters lower-level primitive command failures that do not have such preconditions. For example, if Rosie is asked to approach an object, it has to `go to` the location of the object and then `face` the object. If Rosie encounters an internal error that interferes with its ability to `face` the object, this is low-level information that is only available by explicitly debugging the robot. We cannot expect the non-expert to have debugging skills. Therefore, one problem to be studied here is if we can train non-experts to provide alternative commands to recover from a low-level failure without having to directly debug the system.

9.3.2 Higher Level Task Failures and Inefficiencies

To understand the types of failures that can occur with a hierarchical task teaching process, let us use the following example.

1. `Bake the pizza`
2. `Heat the pizza`
3. Teach subtasks of `Heat`
4. `You are done` (End `Heat` Task)
5. `Empty the oven`
6. Teach subtasks of `Empty`
7. `You are done` (End `Empty` Task)
8. `You are done` (End `Bake` Task)

In this case, an overarching `Bake` Task is being taught to the robot. `Heat` and `Empty` are being taught as sub-tasks of the larger task `Bake`. There are two kinds of scenarios that can occur in this teaching scenario that can derail the teaching process.

One situation is if the instructor provides an instruction that they did not intend to add to the task. For example, in the study, participants would pick up the pizza only to realize the oven is not open. They would then have to place the pizza on the table to open the oven, which is ideally not a part of the `Heat` task. One participant opened the pantry by mistake when they intended to open the oven. Participants also noticed that they taught the task in the wrong order. While Rosie performs a retrospective analysis to eliminate actions that are not relevant to the overarching goals of the task (here, eliminating the action of placing the pizza on the table or opening the pantry) from its final task representation, this is the kind of internal processing information that may not be necessary or even useful to a non-expert early in their interaction.

In a simple blocks world, Appelgren and Lascarides [2] implement mechanisms in a robot to incorporate an instructor's corrective feedback (e.g., "No, put red blocks on blue blocks") when it performs an action (e.g., putting a blue block on a red block) that violates a predefined set of rules and to reverse its action. In our situation, while this type of approach can be useful if the instructor recognizes their mistake immediately, relying on it alone assumes that the instructor is aware of all aspects of the task beforehand. This is not realistic since, in our study, we observed some of the participants planning as they went along during their instruction of the robot.

Another solution would be to extend the interface to allow the instructor to access the learned actions as a part of the task and manually edit it for the future. While this would require participants to learn a new interface, it would also provide them confidence that these mistakes are reversible and can be fixed once the task is learned.

But the above solutions may not work as effectively if the instructor runs into a failure that puts the robot in an irrecoverable state or if the action is not reversible (e.g., the pizza is already heated). In Rosie, this can occur if the instructor mistakenly provides an invalid command (e.g., Open the pizza) or if some action randomly results in the robot reaching an irrecoverable state. If such a failure is encountered in the middle of the instructor providing a set of instructions to teach a task, Currently, Rosie would need to be restarted, and the instructor would have to repeat all the instructions. An intermediate solution can be for the robot to store the instructions for the partially taught task as it goes along so that it can be recovered after the restart. But given the scope of failures that one can encounter with a real robot, these types of solutions are not sufficient.

While we look towards interacting with real-time learning systems, these failures are inevitable. Given that people interact with each other using situated dialogue while teaching real-time tasks, it would be useful to explore what we can learn from humans about recovering from failures. This would begin with providing explanations that are more human-like in nature as described in section 3.4. More studies need to be conducted with real robots to identify the different types of failures that can occur and evaluate how to resolve and recover from these failures. This can only be done if we adopt an iterative research process where we develop and extend robots and study their interaction with non-expert teachers as a part of the development process.

CHAPTER 10

Conclusion

In this dissertation, we conducted a think-aloud study with 14 non-expert participants to analyze how non-experts teach multi-step hierarchical tasks to an Interactive Task Learning robot. Through qualitative and quantitative analysis of this study, we highlight the inherent complexity of teaching such a task. We also describe in detail participant interpretations of a complex learning robot in terms of its knowledge and capabilities and its corresponding effect on participant teaching processes. While 13 out of 14 participants successfully taught the task, they faced many challenges during the teaching process. These challenges are to be expected since interacting with a sophisticated robotic system such as Rosie will always involve a learning curve. To address and resolve these challenges, this chapter summarizes the opportunities and potential implementations for future robot interaction development discussed in Chapter 9 that will enable non-experts to interact efficiently and effectively with such a robot.

While the simple templates provided for initial exploration were effective for teaching this task, we observed the need for extensions to these templates to support more complex instructions. To consider instructions that may not be supported by templates, we propose using technologies like large language models to convert free-form instructions that instructors provide to robot-understandable language, thus enabling effective interaction.

In addition, we must build robots that can leverage their knowledge about their environment, task knowledge, learning process, task execution, and capabilities to provide relevant knowledge to the instructor. The robot must proactively provide information about its internal and external task execution and corresponding environmental changes. To aid the instructor in their teaching process, the robot must actively provide knowledge of its learning process, including providing potential instruction suggestions. This would help the instructor build a better mental model of the robot's learning and, as a result, leverage its learning capabilities to teach more efficiently and effectively over time. The robot must also be capable of answering clarification questions regarding these types of knowledge. This process can be improved by personalizing the interaction with each instructor if the robot has a mental model of the instructor that it updates based on its interaction with them over time.

Lastly, we must develop robots that can help instructors recover from inefficiencies and failures that occur during teaching interactions. More research is required to determine the best way to help non-experts recover from low-level system failures that typically require debugging skills, and are therefore not accessible to non-experts. We also find that there are challenges involved when a robot fails and reaches an irrecoverable state during the process of being taught a multi-step hierarchical task. We find that existing research is not sufficient and more studies need to be conducted to understand the scope of failures that can occur in a hierarchical task teaching process and to understand how non-expert instructors can recover from such failures in real time.

We found that even when faced with failure scenarios from which they could not always recover, people were motivated to go on training the robot. This gives us hope that improving robot interaction to address these failures can lead to more promising outcomes that can be compared to the findings of our study. We emphasize the value of iterative development in this dissertation in order to create robots that can interact with non-experts in the real world. We must continue to study and understand how humans interact with robots to realize our imagined future in which robots work alongside humans.

APPENDIX A

Study Instructions

You will provide instructions to the robot agent Rosie operating in a simulated environment (shown below) so that it can learn and perform the task successfully. You will be given a task description before the study.

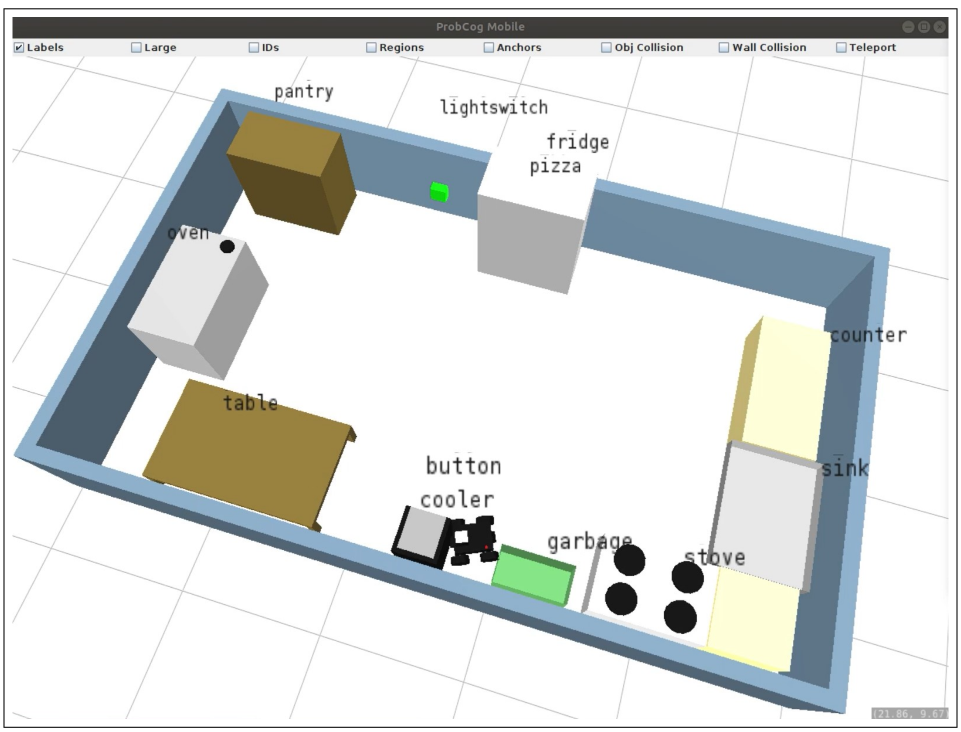


Figure A.1: Simulated Environment

A.1 Instructions about The Robot

1. You will interact directly with the robot agent. However, the interviewer may have to
   (a) intervene from time to time to correct typographical errors in the instruction,
   (b) modify instructions that cannot be processed by the robot or,
   (c) restart the agent when it fails to proceed with a task.
2. **We are testing the robot, not you!**
3. You can zoom in and out of the environment using the scroll button.
4. Here is what you need to know about the robot agent and the interaction with it.
   (a) The robot has complete knowledge about the environment.
   (b) You cannot directly interact with any object using the mouse. Instead, you must ask the robot to perform actions to interact with objects.
   (c) The robot can handle only one object at a time.
   (d) Please wait for the robot to respond after each instruction.
   (e) Please provide step-by-step instructions
5. The instructions you provide will follow the providing templates (See application)
   (a) Action: Verb + Object (Empty the garbage)
   (b) Action: Verb + Object1 + Preposition + Object2 (Fetch the coffee from the kitchen)
   (c) Goal: The goal is that Object + linking-verb + Object State (The goal is that the garbage is empty)
   (d) Goal: The goal is that Object1 + linking-verb + Preposition + Object2 (The goal is that the coffee is on the counter)
6. Robot's existing knowledge (See application)

   (a) The robot has actions it already knows (in capital letters) and actions/tasks that you can teach it (in lowercase letters)
   (b) When you teach a new action/task “action-x,” the robot responds “What is the next goal or subtask of action-x?”
   (c) Once you are done teaching an action/task, press the “You are done” button (on the bottom left) to continue with the overall task.

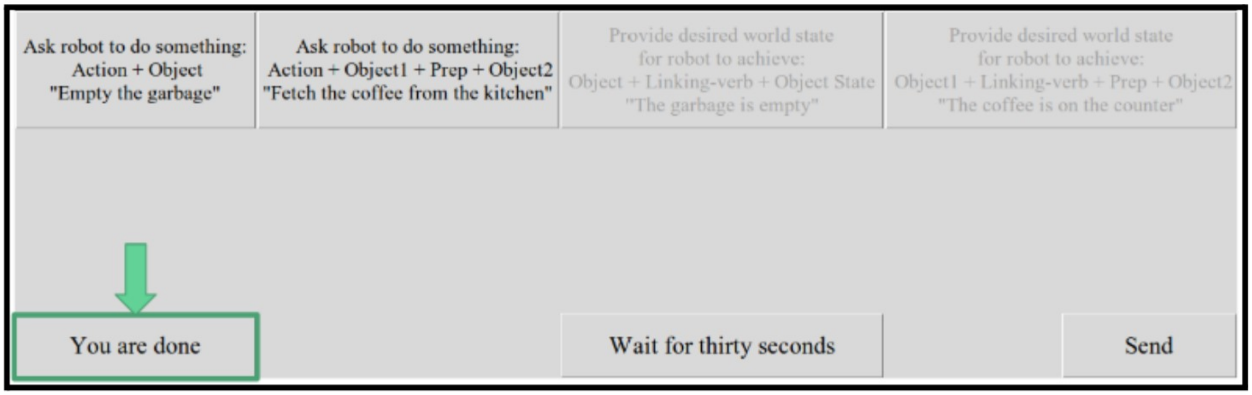


Figure A.2: `You are done` command

7. Think-aloud study -
   (a) You will now watch a demo video of what think-aloud looks like https://www.nngroup.com/articles/thinking-aloud-demo-video/
   (b) Please narrate your thought process as you perform the teaching task.
   (c) During the study, the interviewer will refrain from commenting so as to not interrupt the interaction.
   (d) **Remember, we are testing the robot, not you! So the customer (or the participant) is always right!**
8. You can come back and refer to this document during the study.
9. If you have any questions or you are unsure what to do next, you can ask the interviewer.
10. Interviewer: Run Rosie and start screen recording and audio recording

A.2 Training Tasks

You will first undergo a training session in order to familiarize yourself with the robot instruction interface. Please start thinking out aloud from this point on.

A.2.1 Training Task 1

1. Press the action template 1 button to access the first action template (Figure A.3

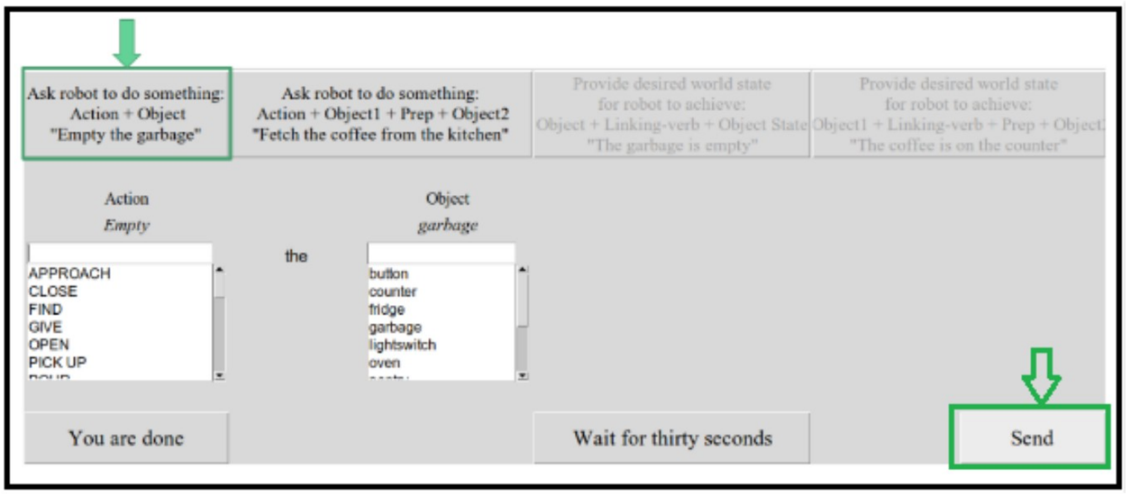


Figure A.3: Action Template 1

2. Provide the following instruction: Approach the lightswitch
3. Press "Send" Button.
4. Once the robot completes this action successfully, it will respond "I am ready for a new task."

A.2.2 Training Task 2

1. Press the action template 2 button to access the second action template (Figure A.4
2. Provide the following instruction: Press the button on the watercooler.
3. Press "Send" Button.

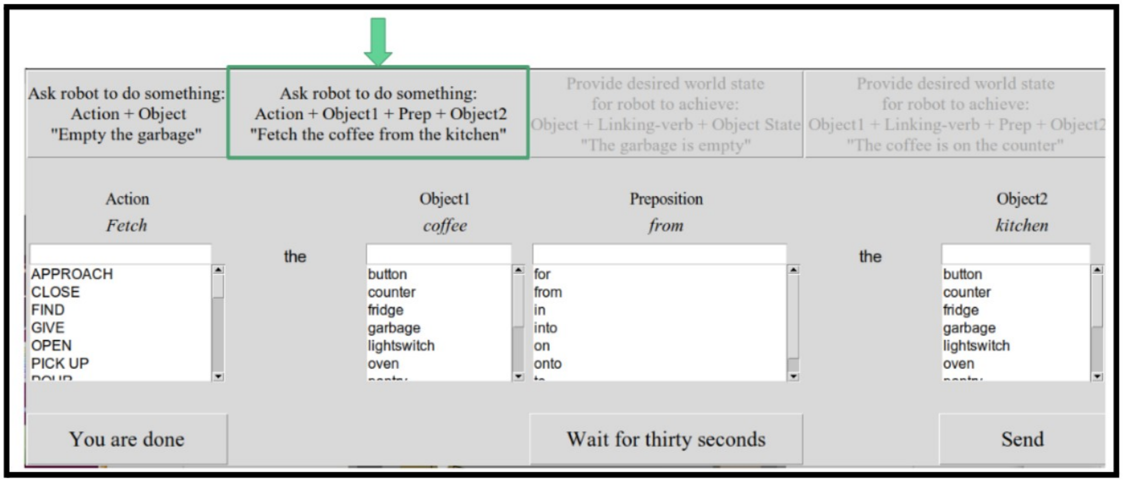


Figure A.4: Action Template 2

4. Once the robot completes this action successfully, it will respond "I am ready for a new task."

A.2.3 Training Task 3

1. Press the button "Wait for thirty seconds"

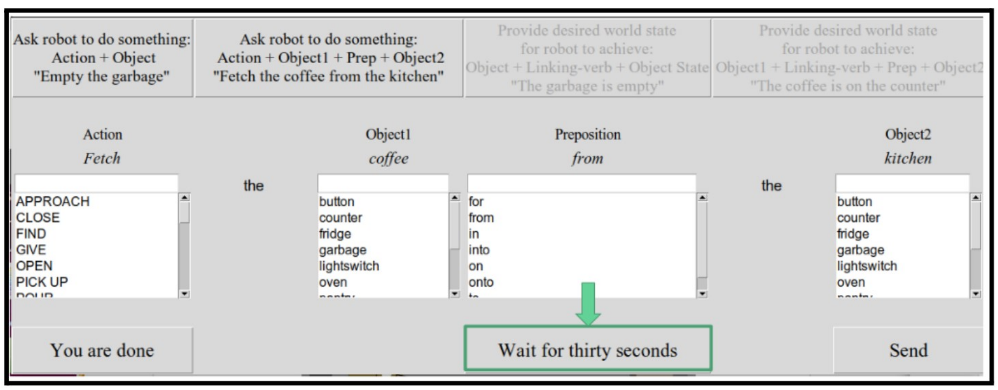


Figure A.5: `Wait for thirty seconds` command

2. Once the robot completes this action successfully, it will respond "I am ready for a new task."

You will now move on to the actual study.

A.3 Main Experiment

1. For the main experiment, you will provide instructions to the robot for the following task:

   The robot must cook (350 degrees for 30 seconds) a pizza using the oven and the cooked pizza must end up on the counter.

2. Provide instructions using the template interface provided to interact with the robot agent.

3. Once you are done teaching the task, please tell the interviewer that you have completed.

4. You can come back and refer to this document during the study.

5. Any questions?

APPENDIX B

Additional Environment and Template Information

In this appendix, we present supplementary information about the objects in the environment in terms of their categories and affordances. We then present the views of the first action template and thet goal templates described in chapter 5. Additionally, we present the lists of different parts of speech used to populate the drop-down lists in individual templates.

B.1 Categories and Affordances of Objects

The following is a list of the categories and affordances of the objects in the study environment.

- Button - Category: Object; Affordances: Pressable
- Counter - Category: Furniture, Object, Surface
- Fridge - Category: Appliance, Object, Receptacle; Affordances: Openable
- Garbage - Category: Furniture, Object, Receptacle; Affordances: Always-Open
- Lightswitch - Category: Object; Affordances: Activatable
- Oven - Category: Appliance, Object, Receptacle, Surface; Affordances: Activatable, Openable
- Pantry - Category: Furniture, Object, Receptacle; Affordances: Openable

- Pizza - Category: Food, Object; Affordances: Grabbable
- Sink - Category: Drain, Furniture, Object, Receptacle; Affordances: Always-Open
- Stove - Category: Appliance, Object, Receptacle, Surface; Affordances: Activatable, Openable
- Table - Category: Furniture, Object, Surface
- Watercooler - Category: Appliance, Dispenser, Object, Receptacle; Affordances: Always-Open

B.2 Action and Goal Templates

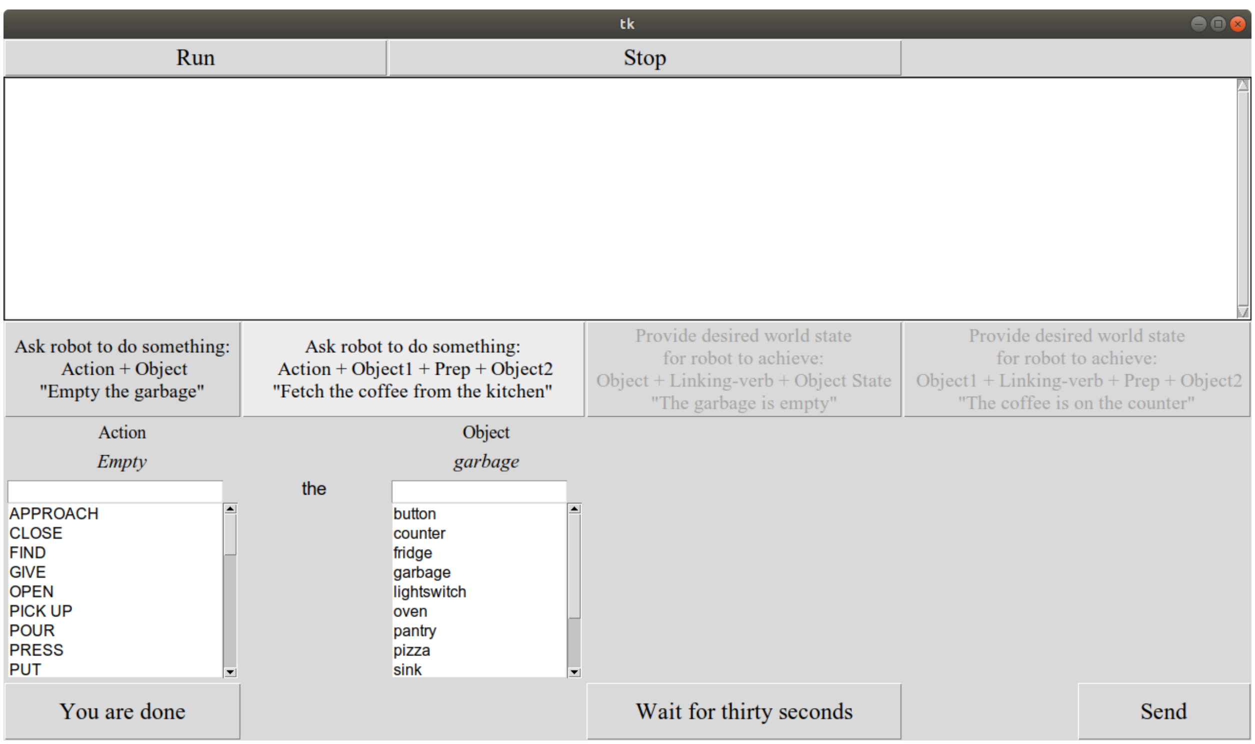


Figure B.1: First Action Template Interface

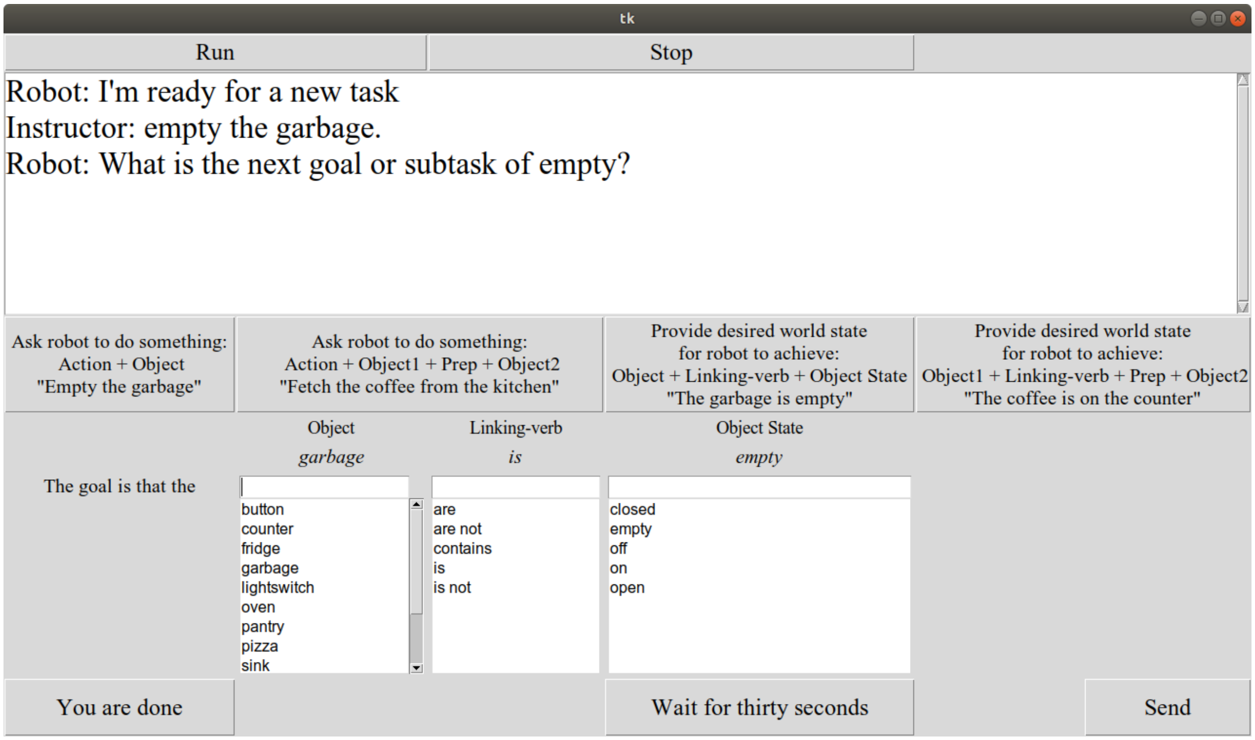


Figure B.2: First Goal Template Interface

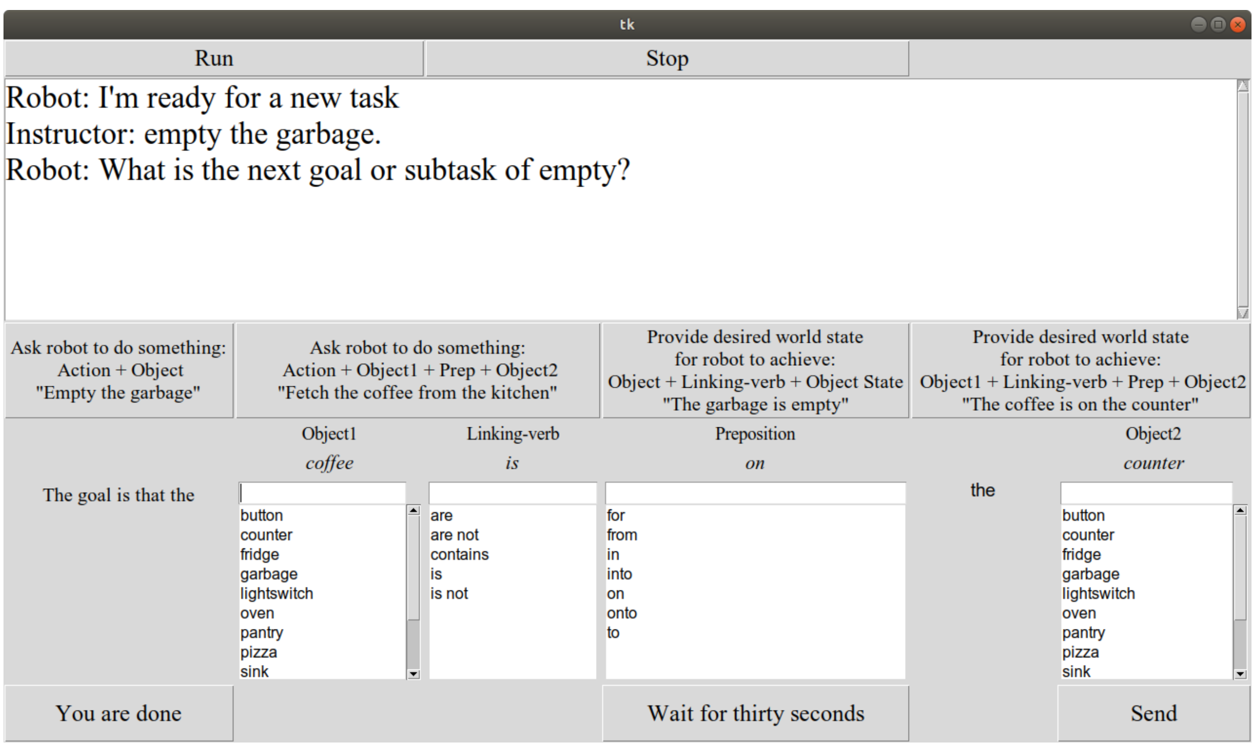


Figure B.3: Second Goal Template Interface

B.3 List of Parts of Speech

We also present the lists of different parts of speech including actions, linking verbs, nouns, object states, and prepositions that were provided as a part of the action and goal templates.

Prepositions	Object States	Linking Verbs	Nouns
from	on	are	button
for	off	are not	counter
in	open	contains	fridge
into	closed	is	garbage
on	empty	is not	lightswitch
onto			oven
to			pantry
			pizza
			sink
			stove
			table
			watercooler

Table B.1: Lists of parts of speech (besides actions) provided in instruction templates

Primitive Actions	Teachable Actions
Approach	Bake
Close	Bring
Find	Carry
Give	Clean
Open	Clear
Pick Up	Cook
Pour	Deliver
Press	Empty
Put	Fetch
Put Down	Get
Scan	Heat
Turn Off	Grab
Turn On	Keep
	Make
	Move
	Prepare
	Refrigerate
	Shut
	Start
	Store
	Take

Table B.2: List of primitive and teachable actions provided as a part of action templates

BIBLIOGRAPHY

[1] Sonya Alexandrova, Maya Cakmak, Kaijen Hsiao, and Leila Takayama. Robot programming by demonstration with interactive action visualizations. In *Robotics: science and systems*. Citeseer, 2014.

[2] Mattias Appelgren and Alex Lascarides. Interactive task learning via embodied corrective feedback. *Autonomous Agents and Multi-Agent Systems*, 34:1–45, 2020.

[3] Chris Baker, Rebecca Saxe, and Joshua Tenenbaum. Bayesian theory of mind: Modeling joint belief-desire attribution. In *Proceedings of the annual meeting of the cognitive science society*, volume 33, 2011.

[4] Virginia Braun and Victoria Clarke. Using thematic analysis in psychology. *Qualitative research in psychology*, 3(2):77–101, 2006.

[5] Cynthia Breazeal, Jesse Gray, and Matt Berlin. An embodied cognition approach to mindreading skills for socially intelligent robots. *The International Journal of Robotics Research*, 28(5):656–680, 2009.

[6] Cynthia Breazeal, Cory D Kidd, Andrea Lockerd Thomaz, Guy Hoffman, and Matt Berlin. Effects of nonverbal communication on efficiency and robustness in human-robot teamwork. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 708–713. IEEE, 2005.

[7] Joost Broekens, Maaike Harbers, Koen Hindriks, Karel Van Den Bosch, Catholijn Jonker, and John-Jules Meyer. Do you get it? user-evaluated explainable bdi agents. In *Multiagent System Technologies: 8th German Conference, MATES 2010, Leipzig, Germany, September 27-29, 2010. Proceedings 8*, pages 28–39. Springer, 2010.

[8] Connor Brooks and Daniel Szafir. Building second-order mental models for human-robot interaction. *arXiv preprint arXiv:1909.06508*, 2019.

[9] Joyce Y Chai, Rui Fang, Changsong Liu, and Lanbo She. Collaborative Language Grounding Toward Situated Human-Robot Dialogue. *AI Magazine*, 37(4), 2016.

[10] Joyce Y. Chai, Lanbo She, Rui Fang, Spencer Ottarson, Cody Littley, Changsong Liu, and Kenneth Hanson. Collaborative effort towards common ground in situated human-robot dialogue. In *Proceedings of the 2014 ACM/IEEE International Conference on Human-Robot Interaction*, HRI ’14, page 33–40, New York, NY, USA, 2014. Association for Computing Machinery.

[11] Crystal Chao, Maya Cakmak, and Andrea L Thomaz. Transparent active learning for robots. In *Human-Robot Interaction (HRI), 2010 5th ACM/IEEE International Conference on*, pages 317–324. IEEE, 2010.

[12] Elizabeth Charters. The use of think-aloud methods in qualitative research an introduction to think-aloud methods. *Brock Education Journal*, 12(2), 2003.

[13] Herbert H Clark. Common ground. In *Using language*, chapter 10, pages 92–121. Cambridge University Press, 1996.

[14] Bhavana Dalvi, Oyvind Tafjord, and Peter Clark. Towards teachable reasoning systems: Using a dynamic memory of user feedback for continual system improvement. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9465–9480, 2022.

[15] Devleena Das, Siddhartha Banerjee, and Sonia Chernova. Explainable ai for robot failures: Generating explanations that improve user assistance in fault recovery. In *Proceedings of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*, pages 351–360, 2021.

[16] Kraig Finstad. The usability metric for user experience. *Interacting with computers*, 22(5):323–327, 2010.

[17] Yolanda Gil. Human tutorial instruction in the raw. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 5(1):1–29, 2015.

[18] Yolanda Gil, Varun Ratnakar, and Christian Frtiz. Tellme: Learning procedures from tutorial instruction. In *Proceedings of the 16th International Conference on Intelligent User Interfaces*, IUI '11, page 227–236, New York, NY, USA, 2011. Association for Computing Machinery.

[19] C Dominik Güss. What is going through your mind? thinking aloud as a method in cross-cultural psychology. *Frontiers in psychology*, 9:1292, 2018.

[20] Bradley Hayes and Julie A Shah. Improving robot controller transparency through autonomous policy explanation. In *Proceedings of the 2017 acm/ieee international conference on human-robot interaction*, pages 303–312. ACM, 2017.

[21] Laura M Hiatt, Cody Narber, Esube Bekele, Sangeet S Khemlani, and J Gregory Trafton. Human modeling for human–robot collaboration. *The International Journal of Robotics Research*, 36(5-7):580–596, 2017.

[22] Tasneem Kaochar, Raquel Torres Peralta, Clayton T Morrison, Ian R Fasel, Thomas J Walsh, and Paul R Cohen. Towards understanding how humans teach robots. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 347–352. Springer, 2011.

[23] Faisal Khan, Xiaojin Zhu, and Bilge Mutlu. How do humans teach: On curriculum learning and teaching dimension. In *NeurIPS*, 2011.

[24] Sara Kiesler. Fostering common ground in human-robot interaction. In *ROMAN 2005. IEEE International Workshop on Robot and Human Interactive Communication, 2005.*, pages 729–734. IEEE, 2005.

[25] James Kirk. *Learning Hierarchical Compositional Task Definitions through Online Situated Interactive Language Instruction*. PhD thesis, Department of Computer Science and Engineering,University of Michigan, Ann Arbor, 2019.

[26] W Bradley Knox, Brian D Glass, Bradley C Love, W Todd Maddox, and Peter Stone. How humans teach agents. *IJSR*, 4(4):409–421, 2012.

[27] Nicole C Krämer, Astrid von der Pütten, and Sabrina Eimler. Human-agent and human-robot interaction theory: Similarities to and differences from human-human interaction. In *Human-computer interaction: The agency perspective*, pages 215–240. Springer, 2012.

[28] Samantha Krening and Karen M Feigh. Interaction algorithm effect on human experience with reinforcement learning. *ACM Transactions on Human-Robot Interaction (THRI)*, 7(2):1–22, 2018.

[29] Todd Kulesza, Simone Stumpf, Margaret Burnett, and Irwin Kwan. Tell me more? the effects of mental model soundness on personalizing an intelligent agent. In *Proceedings of the sigchi conference on human factors in computing systems*, pages 1–10, 2012.

[30] John E Laird. *The Soar cognitive architecture*. MIT press, 2012.

[31] John E Laird, Kevin Gluck, John Anderson, Kenneth D Forbus, Odest Chadwicke Jenkins, Christian Lebiere, Dario Salvucci, Matthias Scheutz, Andrea Thomaz, Greg Trafton, et al. Interactive task learning. *IEEE Intelligent Systems*, 32(4):6–21, 2017.

[32] Daniel A Lazewatsky and William D Smart. Context-sensitive in-the-world interfaces for mobile manipulation robots. In *RO-MAN, 2012 IEEE*, pages 989–994. IEEE, 2012.

[33] Toby Jia-Jun Li, Marissa Radensky, Justin Jia, Kirielle Singarajah, Tom M. Mitchell, and Brad A. Myers. Pumice: A multi-modal agent that learns concepts and conditionals from natural language and demonstrations. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, UIST '19, page 577–589, New York, NY, USA, 2019. Association for Computing Machinery.

[34] Peter Lindes, Aaron Mininger, James R Kirk, and John E Laird. Grounding language for interactive task learning. In *Proceedings of the First Workshop on Language Grounding for Robotics*, pages 1–9, 2017.

[35] Christopher J MacLellan, Erik Harpstead, Robert P Marinier III, and Kenneth R Koedinger. A framework for natural cognitive system training interactions. *Advances in Cognitive Systems*, 6:1–16, 2018.

[36] Matthew Marge, Felix Gervits, Gordon Briggs, Matthias Scheutz, and Antonio Roque. Let's do that first! a comparative analysis of instruction-giving in human-human and human-robot situated dialogue. In *Proc. of SemDial*, 2020.

[37] Matthew Marge and Alexander I Rudnicky. Towards Overcoming Miscommunication in Situated Dialogue by Asking Questions. In *AAAI Fall Symposium: Building Representations of Common Ground with Intelligent Agents*, 2011.

[38] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, 267:1–38, 2019.

[39] Aaron Mininger. *Expanding Task Diversity in Explanation-Based Interactive Task Learning*. PhD thesis, Department of Computer Science and Engineering,University of Michigan, Ann Arbor, 2021.

[40] Shiwali Mohan. *From Verbs to Tasks: An Integrated Account of Learning Tasks from Situated Interactive Instruction*. PhD thesis, Department of Computer Science and Engineering,University of Michigan, Ann Arbor, 2015.

[41] Shiwali Mohan, Matt Klenk, Matthew Shreve, Kent Evans, Aaron Ang, and John Maxwell. Characterizing an analogical concept memory for newellian cognitive architectures. In *Advances in Cognitive Systems*, 2020.

[42] Shiwali Mohan, Aaron H Mininger, James R Kirk, and John E Laird. Acquiring grounded representations of words with situated interactive instruction. In *Advances in Cognitive Systems*, volume 2, pages 113–130. Citeseer, 2012.

[43] Anahita Mohseni-Kabir, Charles Rich, Sonia Chernova, Candace L. Sidner, and Daniel Miller. Interactive hierarchical task learning from a single demonstration. In *2015 10th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 205–212, 2015.

[44] Donald A Norman. Some observations on mental models. *Mental models*, 7(112):7–14, 1983.

[45] Leah Perlmutter, Eric Kernfeld, and Maya Cakmak. Situated language understanding with human-like and visualization-based transparency. In *Robotics: Science and Systems*, volume 12, pages 40–50, 2016.

[46] Martin J Pickering and Simon Garrod. Toward a mechanistic psychology of dialogue. *Behavioral and brain sciences*, 27(2):169–190, 2004.

[47] Preeti Ramaraj, Charles L. Ortiz, and Shiwali Mohan. Unpacking human teachers' intentions for natural interactive task learning. In *2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN)*, pages 1173–1180, 2021.

[48] Preeti Ramaraj, Saurav Sahay, Shachi H. Kumar, Walter S. Lasecki, and John E. Laird. Towards using transparency mechanisms to build better mental models. In *Advances in Cognitive Systems: 7th Goal Reasoning Workshop*, volume 7, pages 1–6, 2019.

[49] Stephanie Rosenthal, Manuela Veloso, and Anind K Dey. Acquiring accurate human responses to robots' questions. *International journal of social robotics*, 4(2):117–129, 2012.

[50] Matthias Scheutz, Thomas Williams, Evan Krause, Brley Oosterveld, Vasanth Sarathy, and Tyler Frasca. An overview of the distributed integrated affect and reflection cognitive diarc architecture. In *Cognitive Architectures*, Intelligent Systems, Control and Automation: Science and Engineering. Springer International Publishing, 2019.

[51] Katta Spiel, Oliver L. Haimson, and Danielle Lottridge. How to do better with gender on surveys: A guide for hci researchers. *Interactions*, 26(4):62–65, jun 2019.

[52] Sonja Stange and Stefan Kopp. Effects of a social robot's self-explanations on how humans understand and evaluate its behavior. In *Proceedings of the 2020 ACM/IEEE international conference on human-robot interaction*, pages 619–627, 2020.

[53] Nicole Sultanum, Soroush Ghorashi, Christopher Meek, and Gonzalo Ramos. A teaching language for building object detection models. In *Proceedings of the 2020 ACM Designing Interactive Systems Conference*, pages 1223–1234, 2020.

[54] Aaquib Tabrez, Matthew B Luebbers, and Bradley Hayes. A survey of mental modeling techniques in human–robot teaming. *Current Robotics Reports*, 1:259–267, 2020.

[55] Stefanie Tellex, Ross A Knepper, Adrian Li, Daniela Rus, and Nicholas Roy. Asking for Help Using Inverse Semantics. Robotics: Science and Systems Foundation, 2014.

[56] Ravenna Thielstrom, Antonio Roque, Meia Chita-Tegmark, and Matthias Scheutz. Generating explanations of action failures in a cognitive robotic architecture. In *2nd Workshop on Interactive Natural Language Technology for Explainable Artificial Intelligence*, pages 67–72, 2020.

[57] Jesse Thomason, Aishwarya Padmakumar, Jivko Sinapov, Nick Walker, Yuqian Jiang, Harel Yedidsion, Justin Hart, Peter Stone, and Raymond Mooney. Jointly improving parsing and perception for natural language commands through human-robot dialog. *Journal of Artificial Intelligence Research*, 67:327–374, 2020.

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

[59] Emily Wu, Nakul Gopalan, James MacGlashan, Stefanie Tellex, and Lawson LS Wong. Social Feedback For Robotic Collaboration. 2016.

[60] Yaqi Xie, Chen Yu, Tongyao Zhu, Jinbin Bai, Ze Gong, and Harold Soh. Translating natural language to planning goals with large-language models. *arXiv preprint arXiv:2302.05128*, 2023.

[61] Yu Zhang, Sarath Sreedharan, Anagha Kulkarni, Tathagata Chakraborti, Hankz Hankui Zhuo, and Subbarao Kambhampati. Plan explicability and predictability for robot task planning. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 1313–1320. IEEE, 2017.