Why did you do that?
Providing transparent explanations of learned robot behavior

Authors omitted for anonymous review
Association for the Advancement of Artificial Intelligence
2275 East Bayshore Road, Suite 160
Palo Alto, California 94303

Abstract

Effective collaboration requires transparent communication between teammates. Teammates should be capable of both explaining their actions and their goals, and reciprocally capable of utilizing this same information when provided by a collaborator. This poses a challenge for human-robot collaboration (HRC), as it requires the robot to be capable of extracting, remembering and redeploying meaning from interactions with collaborators. To this end, we present a proof-of-concept, task-independent, robotic system capable of generating feature-rich task representations from natural language interactions. The proposed system uses the OntoSem cognitive architecture to construct hierarchical task networks (HTNs) of human-robot collaborative tasks in one example from a natural speech demonstration from a human partner. Critically, our system not only learns the correct sequence of primitive actions for completing a task, but generates valid and comprehensible names for subtasks. This allows our system to answer *why* queries, corresponding to movement up the hierarchy, *how* queries, corresponding to movement down the hierarchy, and *what* queries, corresponding to stationary and horizontal movement through the hierarchy. We demonstrate the feasibility of the system on a collaborative furniture construction task with a Baxter research robot and a human participant.

Introduction

Transparent communication between teammates is a key component of effective collaboration. In the medical field, for example, it has been demonstrated that standardizing communication protocols between healthcare providers can have a drastic effect on patient care (Pfrimmer, 2009; Leonard, Graham, and Bonacum, 2004). Likewise, research in human–robot interaction (HRI) and human–computer interaction (HCI) has shown that systems with transparent decision making processes assists users in calibrating their levels of trust in the system (Lee and Moray, 1992; Wang, Jamieson, and Hollands, 2009), as well as how they attribute blame to it (Kim and Hinds, 2006).

Within the broader field of human–robot interaction, the goal of human–robot collaboration (HRC) is to design systems capable of working collaboratively with humans; to this end, enabling robots to communicate their goals and intentions is critical. However, unlike the medical domain, communication protocols for HRC cannot generally be standardized or enforced, making this a difficult proposition. Indeed, a number of open questions stand between the HRC community and human–compatible HRC systems: i) What sorts of representational frameworks lend themselves best to the collaborative domain? ii) How can these frameworks accommodate naturalistic social interaction? iii) How can we design these frameworks to be robust while remaining transparent to the user?

Further complicating matters is the need to design systems that can accommodate natural language while also

Figure 1: Transparent communication during human–robot collaboration. Because the robot is equipped with an expressive representation of the task, it is able to respond to queries about the nature of the task.

satisfying the aforementioned three questions. Research in human–human interaction has shown that inter-collaborator communication is often highly situated (Shah and Breazeal, 2010) and multimodal (Wahn et al., 2016). This suggests that human-level HRC systems must be capable of contending with these sorts of ambiguities, as well as the multitude of other sources of ambiguity and disfluency common to natural language.

In an attempt to address the issues mentioned above, we present a proof-of-concept, task-independent, robotic system capable of generating expressive task representations from natural language interactions. The proposed system uses the OntoSem cognitive architecture (McShane, Nirenburg, and Beale, 2016; Nirenburg et al., 2018) to construct rich representations in the form of hierarchical task networks (HTNs) of human-robot collaborative tasks in one example from a natural speech demonstration provided by a human partner. As OntoSem's learning capabilities are limited only by the contents of its ontological world model, our system is capable of learning a multitude of tasks. Critically, our system not only learns the correct sequence of primitive actions for completing a given task, but generates valid and comprehensible names for subtasks. This enabled us to implement a simple, but effective dialogue system for querying the structure of the generated HTN. More specifically, our system is capable of answering *why* queries, corresponding to movement up the hierarchy, *how* queries, corresponding to movement down the hierarchy, and *what* queries, corresponding to stationary and horizontal movement through the hierarchy. As a result, our system can both learn, and subsequently describe, a task through natural language interactions with a human(see figure 1), and thus provides a promising first step towards enabling transparent HRC. We demonstrate the feasibility of the system on collaborative a furniture construction task with a Baxter research robot and a human participant.

In the following sections, we introduce transparent communication, HTNs, and cognitive architectures. Then we discuss the OntoSem cognitive architecture and how it constructs HTNs. Subsequently, we discuss how the structure of the HTN can be leveraged to create a transparent dialogue system. Fnally, we discuss the details of the collaborative furniture assembly paradigm used to demonstrate the system and how OntoSem learns HTNs from spoken instruction, and the demonstration itself.

Background and related work

The goal of this work is to provide a framework that enables humans and robots to collaborate transparently and effectively. However, such a framework must have a representational scheme flexible enough to model collaborative tasks, but expressive enough to be intelligible to a teammate. In addition, this framework should be capable of understanding and utilizing natural language. We found that the HTN formalism to be an intuitive and expressive way of representing tasks. Likewise, much work has been done in designing cognitive architectures capable of extracting meaning from natural language.

Transparent Communication

Enabling artificial agents to explain and justify their behaviors and goals is critical for these systems to effectively collaborate with humans. Since the 1980s, the goal of *Explainable AI* research has been to design making systems that are transparent to the user (Chandrasekaran, Tanner, and Josephson, 1989). In recent years, as powerful, yet inscrutable deep learning systems become more prevalent, the call to design these systems to be more transparent has become more urgent (Samek, Wiegand, and Müller, 2017).

As embodied collaborative robots pose a physical risk to human users, transparent communication in this domain is an active area of research. Roncone, Mangin, and Scassellati (2017) and Wang, Jamieson, and Hollands (2009) found that transparency significantly improved the performance of human–robot teams at completing an assembly task, as well as increased users trust in the system. Indeed, transparency plays a key role in how robots are viewed by human collaborators. For example, Kim and Hinds (2006) found that people more accurately attribute credit and blame to the robot for its actions. This suggests that users have more realistic expectations for transparent robots, and thus are less likely to misuse them.

Hierarchical Task Networks

Hierarchical Task Networks have been employed in AI planning since the 1970s (Sacerdoti, 1975; Wilkins, 1984; Nau et al., 1999). HTNs are an attractive framework because they represent tasks in an intuitive and (proveably) powerful way (Erol, Hendler, and Nau, 1994). Informally, HTNs are tree-like representations wherein the leaves of the tree are the action primitives to be executed and parent nodes are subtasks that may be comprised of other subtasks and/or action nodes. While HTNs ultimately decompose into a flat sequence of actions, as Mohseni-Kabir et al. (2015) discuss, the HTNs have an apparent isomorphism with how humans naturally conceive hierarchical tasks, and so provide a shared framework by which robot and human collaborators can effectively discuss and reason about tasks.

Hayes and Scassellati (2016) and Roncone, Mangin, and Scassellati (2017) augment the expressiveness of traditional HTNs by introducing the *parallel*, *sequential*, and *alternative* topological operators (denoted, $||$, $\rightarrow$, and $\vee$, respectively), which identify subtasks that can be completed in parallel, sequentially, or are disjunctive. While the strengths of the HTN formalism are evident, manually constructing HTNs for complex tasks can be a cumbersome and time consuming process. As a result, many are actively exploring techniques for learning HTNs from demonstration (Hayes and Scassellati, 2016; Nejati, Langley, and Konik, 2006).

Cognitive Architectures

Scheutz et al. (2007) assert that natural language understanding (NLU) is part-and-parcel of human-like human-robot interaction, yet, for robots, human-like NLU remains out of reach. This poses a challenge for HRC, as extracting meaning from language is of critical importance, yet these systems must contend with speech that is often highly situated,

ACQUIRE-1	
AGENT:	HUMAN-1
THEME:	SCREWDRIVER-1
TIME:	<FIND-ANCHOR-TIME
from-sense:	GET-V1
sent-word-ind:	0, [1]
token:	``got''
concept:	ACQUIRE
is-in-subtree:	EVENT
HUMAN-1	
AGENT:	ACQUIRE-1
from-sense:	I-N1
sent-word-ind:	0, [0]
token:	``I''
concept:	HUMAN
is-in-subtree:	OBJECT
SCREWDRIVER-1	
THEME-OF:	ACQUIRE-1
from-sense:	SCREWDRIVER-N1
sent-word-ind:	0, [3]
token:	``screwdriver''
concept:	SCREWDRIVER
is-in-subtree:	OBJECT

Table 1: Meaning representation for the utterance *""I got a screwdriver"*.

fragmented, and context-dependent (Wahn et al., 2016; Shah and Breazeal, 2010). In addition, language must not simply be understood, but the contents remembered, represented, and redeployed if the circumstances of the task or environment demands it. As cognitive architectures seek to model human cognition, they are uniquely positioned to tackle the aforementioned issues that arise from inherently social phenomena and that are central to enabling human-like HRC.

Much work has been devoted to using cognitive systems for language understanding in the context of collaborative systems. For example, Allen et al. (2007) equip their architecture PLOW with a deep natural language understanding capabilities that enables a human user to teach the system internet browser-based tasks using speech. In the robotics domain, Scheutz et al. (2017) recently demonstrated a system, running on two different autonomous robots capable of learning a task from spoken instructions. Ultimately, robots provide the ideal platform for sophisticated natural language understanding systems, as physical embodiment entails the ability to ground symbols and concepts to features of the world (Mohan et al., 2012).

System Design

We build upon the work of Nirenburg et al. (2018), who integrated robot learning with the OntoSem cognitive architecture. OntoSem is equipped with an ontology of 9000 concepts, a lexicon of 25,000 lexical senses, as well as a number of linguistic micro-theories for handling sources of semantic ambiguity. This ontology is a set of relations between pre-defined as well as learned concepts such as "a chair has four legs." Although the ontology needs to be pre-specified before a task can be learned, a system equipped with a sufficiently knowledge-rich ontology can learn the family of collaborative tasks under consideration. Given a sentence, the OntoSem system is capable of 1) deriving a *meaning representation* (MR) from it such as the one depicted in Table 1 using the current ontology system, 2) using the derived MR to enrich the ontology and construct HTNs. These faculties mentioned above encompass OntoSem's long-term memory (LTM), which encapsulates, among other things, OntoSem's semantic understanding of the world. In addition, OntoSem employs a short-term memory system (STM) for storing sequences of time-ordered MRs generated from recent interactions.

The first stage of the robot's learning process takes in sequences of natural language utterances from the user, paired with pre-specified action sequences, and outputs a sequence of MRs. These MRs are obtained automatically by drawing upon its knowledge resources, including OntoSem's ontology and an English semantic lexicon. The meanings of entries in the lexicon are interpreted in terms of the ontological world model, which comprise the system's LTM. The MRs obtained as a result of processing input sequences are stored in the robot's STM.

Once OntoSem has generated MRs for every member of the input sequence and stored them in the STM, it can begin to model the task. The objective here is to produce an HTN from the sequence of MRs during task training. Effectively, the process expects the MRs corresponding to commands in the original input to form the set of terminal nodes in the resulting HTN tree. Non-terminal nodes in the resulting HTN will be created, named and placed in the tree on the basis of the MRs corresponding to utterances in the original input.

The results of learning are made manifest through the automatic augmentation of the robot's ontology. Specifically, as a result of the learning process, the robot will:

(i) create new, complex events,

(ii) name them,

(iii) determine their constituent subevents,

(iv) determine the events of which they are subevents (encapsulated in an HTN), and

(v) record this new knowledge in its ontology.

The HTN generated by OntoSem extends the representational schema developed in Hayes and Scassellati (2016) and Roncone, Mangin, and Scassellati (2017) in two important ways:

(a) all nodes in the HTN are provided a natural language labeling.

(b) we introduce *agent-based* action primitives whereby both the action to be executed and the agent to be executed are represented in the leaf nodes.

Critically, (a) and (b) are not provided *a priori* but rather are automatically generated by OntoSem. These additions accommodate richer task representations that are better suited for the collaborative, multi-agent domain. For example, the fact that actions are agent-based permits the system

to answer queries that naturally arise over the course of collaboration like: "What are you doing?" and "What should I do?" Similarly, natural language node labelings allow the user to query the system about the structure of the HTN in an intuitive way, e.g. "How do we build the back of the chair?" or "Why are we building a seat?"

How to introduce transparency

Transparency between human and robot collaborators have been shown to have a significant effect on the perception of the robot from the human point of view (Pfrimmer, 2009; Leonard, Graham, and Bonacum, 2004). However, beyond merely augmenting perception, Roncone, Mangin, and Scassellati (2017) and Wang, Jamieson, and Hollands (2009) demonstrated how transparency can tangibly improve the efficiency of a human–robot team. Indeed, Scheutz et al. (2007) assert that transparency is a crucial aspect of human-like HRI, and that such robots must be able to "act in a purpose-driven manner, allowing humans to predict the robot's behaviors based on ascribed beliefs, intentions, and desires."

Yet, despite the apparent benefits of endowing systems with transparent decision-making processes, it is not clear what the best way is of designing such systems. One method is to follow in the lead of Roncone, Mangin, and Scassellati (2017) and employ HTNs as the primary representational schema. As HTNs are, by design, goal-oriented, they provide an ideal framework with which to build the sorts of intentional systems outlined above by Scheutz et al. (2007). Indeed, as Mohseni-Kabir et al. (2015) argue, the structure of HTNs appear to have a natural correspondence to how human beings seem to conceive of tasks, making the motivations of the robots that employ such representations inherently more transparent. Additionally, and perhaps most importantly, the dependency structure of the HTN contextualizes its constituent actions and subtasks, and thus provides a straightforward way of explaining any node in the network. For example, determining why a particular subtask needs to be completed is tantamount to situating it in relation to an overarching task, which is itself equivalent to merely searching for the parent of the subtask of interest. Thus, a transparent dialogue system can be devised whereby users' task-related queries can be answered by conducting searches through the HTN, assuming nodes are provided with intelligible labels. We claim that such a system is capable of responding to three distinct categories of queries, *what*, *why*, and *how*, which correspond to directional movement through the HTN.

What queries correspond to stationary or horizontal movement relative to the leaf node the robot is currently executing, and enable the user to query the current, previous or future state of the task. If the leaf nodes of the HTN are agent-based (see previous section), then the robot can respond to *what* queries about a particular agent. Movement through the HTN is stationary in the sense that a query like "What are you doing?" should return current the robot leaf node, but can also be horizontal in the case of the query "What should I do next?", which requires a search through the current subtask's leaf nodes until a leaf node corresponding to a human action is found. Similarly, a query like "What did you just do?" (which might be asked by a user unfamiliar with the task) would result in a horizontal search, but in the opposite direction. In addition to the current node being executed, the system would need to keep track of previous node executed. If the previous or subsequent node belong to a different subtask from the current one, a more sophisticated search is required. In this case, the system would need to provide both the correct leaf node, and it's parent node in order to provide context for ambiguously motivated actions. For example, in response to the "What did you just do?" query, the system should respond with something analogous to "I retrieved the dowel in order to construct a chair leg.", where "retrieved the dowel" corresponds to the leaf node and "construct a chair leg" corresponds to the parent. Further complicating matters is the case where the horizontal search encounters subtasks that can be completed in parallel. In this instance, there are potentially many possible responses to the question "What should I do next?" There are a few ways of resolving this: the robot can choose a subtask at random, it can enumerate all possible subtasks have the user choose (with a follow up speech command), or possible subtasks for the user to choose from, or can potentially make use of user preferences to select a particular task.

Why queries correspond to vertical movement through the HTN, and enable the user to ascertain the rationale for a given action or subtask. Parameterized queries such as "Why should I do X", where "X" is a subtask in the HTN result in a search for the parent node of subtask "X". By keeping track of the last node to be queried, users can employ the non-parameterized, but temporally-contingent "Why?" query. This is equivalent to the parameterized *why* query, only here the parameter is assumed to be the previously queried subtask. While seemingly a minor addition, temporally-contingent queries like "Why?"enable a critical aspect of interpersonal communication that would otherwise be missing, i.e. the ability to follow up on previous queries. Indeed users can follow up on any *what* queries with a "Why?" or, like a precocious child, chain multiple "Why?" queries is succession, resulting in successively higher levels of explanation. In practice, such a line of questioning must cease when the top of the hierarchy is reached, but in theory explanations can always be given, though they may require a linkage to the explicit goal representations for a robot and an understanding of why a particular HTN was executed. Further research should examine what place, if any such high level explanations have in the collaboration, and how they affect the human–robot dynamic.

Finally, *how* queries correspond to movement down the HTN, enabling a user to understand how a given subtask is completed. In many ways, *how* queries resemble an inversion of *why* queries, which is reflected in the direction of movement through the hierarchy; whereas *why* queries provide high level explanations for subtasks by situating them in relation to more broad subtasks, *how* queries provide low-level explanations of subtasks by appealing to their constituent subtasks. Similarly, *how* queries come in the parameterized ("How do we X?"), and non-parameterized ("How?") variety. Parameterized queries such as "How do

we X?", where "X" is a subtask in the HTN result in a search for the children nodes of subtask "X". However, in this case, it is unclear how the non-parameterized *how* ought to function. One method would be to do as is done with the non-parameterized *why* query and have it be temporally-contingent on the previous query. While this would enable the user to follow up on previous queries in a more natural manner, "How?" queries could not be successively chained, as subtasks can have multiple children subtasks. Another complication is the ambiguity in desired explanation detail of a "How?' query. If the "How?" query is targeting a high level task, for example, is it preferable to give a high level explanation of the task or dive down into the constituent leaf actions? The former is more succinct, but less practical, while the latter provides a sequence of actionable steps, but is more verbose.

Demonstration

Here we describe the design of the collaborative furniture assembly task used to demonstrate the system's abilities in the real world on an autonomous robot. This task, which has the human–robot team constructing a chair, has two phases: the learning phase and the collaboration phase. During the learning phase, the human user can provide natural language demonstrations to teach the naive system the task. During the collaboration phase, the user and robot, using the HTN generated during the learning phase, work together to complete the task, during which time the user can query the HTN via speech commands.

Design

The furniture assembly paradigm has been successfully employed to evaluate the efficacy of HRC models (Roncone, Mangin, and Scassellati, 2017; Brawer et al., 2018). The experimental set-up used here, originally developed in (Zeylikman et al., 2018), is largely unchanged from those described in the papers above. This task involves an autonomous robot and a human collaboratively constructing a piece of furniture (here a miniature chair) from prefabricated parts. The robot can assist the user in putting the chair together by retrieving the appropriate parts, or by holding a part steady while the user affixes another to it. We implemented the system using the Robot Operating System on a Baxter Research Robot (cf. Figure 1). In addition, robot visual perception and control is implemented using Aruco (Garrido-Jurado et al., 2014), which uses unique fiducial markers to track parts in 3D space. Speech-to-text (STT) is handled by the Google Cloud STT API. In order for the robot to verbally respond to queries we have developed a text-to-speech system based on the SVOX PICO engine.

Setup

The experiment is broken up into learning and collaboration phases. During the learning phase, the robot receives sequences of action primitives paired with linguistic descriptions of their task-relevance as input, which are then used to generate an HTN. An example of a generated HTN is depicted in figure 2. While these action primitives

Utterance:	"We will build a chair."
Utterance:	"I need a screwdriver to assemble a chair."
Action:	`GET(screwdriver)`
Utterance:	"Now we will assemble the seat."
Utterance:	"First, we will build the front leg of the chair."
Action:	`GET(bracket-foot)`
Action:	`GET(bracket-front)`
Action:	`GET(dowel)`
Action:	`HOLD(dowel)`
Utterance:	"I am using the screwdriver to affix the brackets on the dowel with screws."
Action:	`RELEASE(dowel)`
Utterance:	"We have assembled a front leg."
Utterance:	"Now, another front leg."
Action:	`GET(bracket-foot)`
Action:	`GET(bracket-front)`
Action:	`GET(dowel)`
Action:	`HOLD(dowel)`
Utterance:	"I have assembled another front chair leg."
Utterance:	"Now, the back leg on the right side."
Action:	`GET(bracket-foot)`
Action:	`GET()bracket-back)`
Action:	`GET(dowel)`
Action:	`HOLD(dowel)`

Table 2: Example of input into to the OntoSem architecture during the learning phase for generating a corresponding HTN

(such as `GET(bracket-foot)`, `HOLD(dowel)`, etc.) are domain-specific and pre-specified, the system itself is task-independent so long as the task can be completed using these primitives. Input can be given using precompiled text files, or can generated by having a user provide natural language description via STT software accompanied by a teleoperated robot enacting the corresponding actions. Table 2 depicts an example of what a portion of this input might look like for a chair construction task.

As tasks are represented hierarchically, multiple tasks can be encapsulated in the HTN, so a user can commence the collaboration phase by selecting a task with a verbal command such as "Let's build a chair!" Once a task is selected, the robot will perform a sequence of actions as dictated by the contents of the HTN. At any point during the task the user can issue verbal command to query the structure of the HTN (e.g. "How do we build a chair?", "What should I do?", etc.) which will result in a verbal response from the robot. In the following sections, we provide specific examples of these queries being used in this demonstration and discuss how they were formulated.

What

Tables 3 and 4 depict example exchanges from the chair construction demonstration. During Table 3, the human and the robot are building a leg of the chair. The *what* query employed in line 1 triggers the system to search for a human action leaf node under the current subtask for

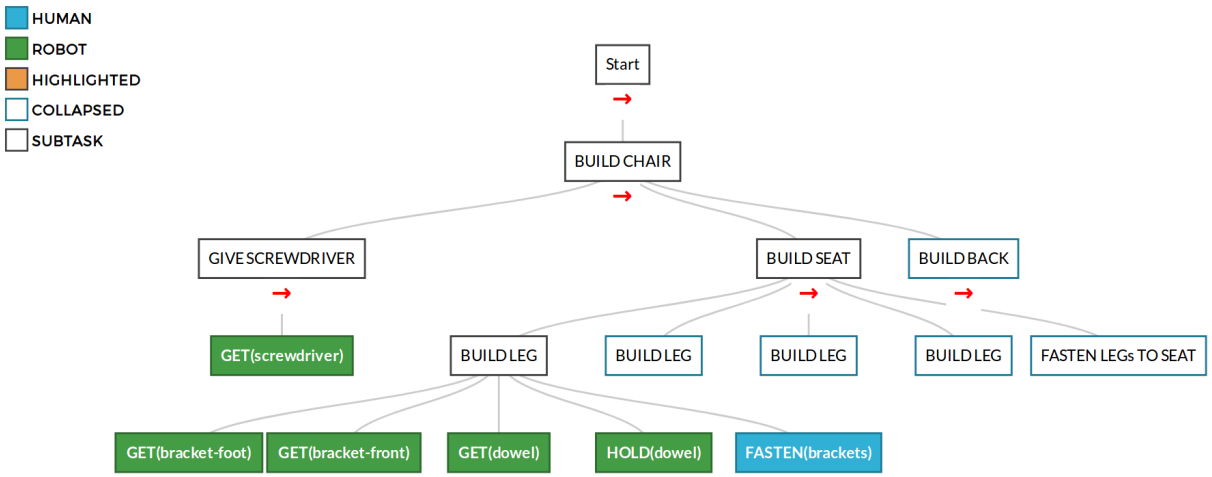


Figure 2: Visualization of complete HTN for chair assembly task generated from natural language demonstration (partially collapsed for increased legibility). The HTN formalism extends the state-of-the-art by including by classifying leaf nodes by the agent who performs them (blue nodes for the human user, green for the robot). Orange nodes (not visible here) highlight the current action being performed. The red arrows present underneath certain subtask nodes is a topological operator that indicates that the children subtasks must be completed in left-to-right order. Additionally, node labels were not provided *a priori* but rather were generated automatically based on the interactions during the learning demonstration.

1.	Human:	**What** should I do next?
2.	Robot:	You should let me hold the dowel so that you can fasten the foot bracket to it.
3.	Human:	**Why**?
4.	Robot:	So that we can build a chair leg.

Table 3: Speach-to-text ranscript of exemplar interactions

1.	Human:	**What** are you doing?
2.	Robot:	Currently, I am getting the top dowel.
3.	Human:	**Why**?
4.	Robot:	So that we can build the chair-back.
5.	Human:	**How** do we build a chair-back?
6.	Robot:	In order to build a chair-back we need to do two things: we need to build the top part of the chair, then we need to fasten the dowels.

Table 4: Speach-to-text transcript of exemplar interactions

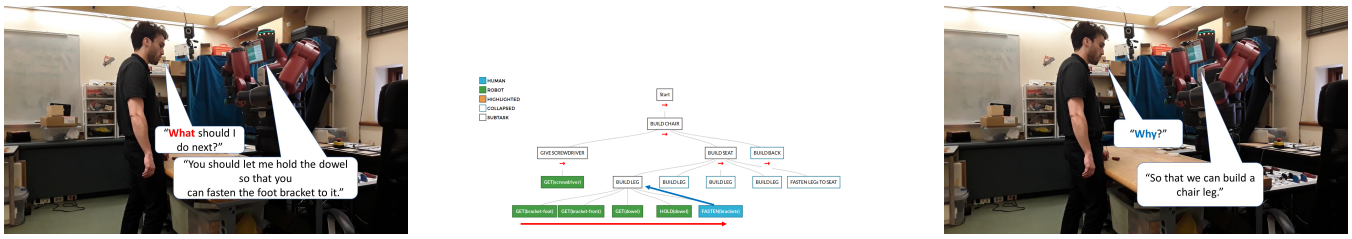
the human to perform. In this instance, there is only one: `FASTEN(brackets)` (refer to the blue node in figure 2), and so the robot responds with the correct action.

This sort of horizontal search through the children of the current subtree produces the best responses to such queries. While a search returning the next human action node across the entire hierarchy is supported by a literal interpretation of this query, such a search runs the risk of returning contextually irrelevant results.

Why

Why queries are employed in line 3 of tables 3 and 4. In both instances the contextual "Why?" is used. This is contextual in the sense that the response is dependent on the previous question asked. In these example exchanges, the preceding queries ("What should I do next?" and "What are you doing?"), both result in searches that return nodes under the current subtask. Because these context dependent *why* queries return the parent subtask of the previously queried node, in these instances they simply return the current subtask. Were a query like "Why do we need to build a seat?" followed up with a "Why?", the response would be something like "Because we are building a chair." by this same algorithm.

It makes sense to respond to *why* queries via upward movement through the hierarchy because the goal of such questions is generally to ascertain context. When a collaborator asks why they should do some action or complete some goal, they are seeking justification for their efforts with regards to some over arching goal (e.g. "You are getting me the screwdriver so that we can finish assembling this chair."). By enabling users to chain "Why?" queries together, we enable



(a) (b) (c)

Figure 3: Using speech queries to traverse the HTN. (a) and (c) depict exchanges taken from the demonstration. The red and blue arrows in (b) correspond to the movement through the HTN triggered by the queries in (a) and (c), respectively.

them to search up the hierarchy until they find a satisfying justification for a particular action or subtask.

How

Line 5 of table 4 depicts an example of a *how* query. *how* queries return the children nodes of a target parent node. In this instance, the target subtask node is `BUILD BACK`, which has two children, `BUILD TOP` and `FASTEN DOWELS`.

Despite the apparent ambiguity in the answer from the robot (e.g., what exactly is the top part of the chair and how is it built?), we believe that our method produces the reasonable amount of detail. For example, in the case where a naive user asks the robot "How do we build a chair?" it would be cumbersome and time consuming for the robot to enumerate every subtask, or all the 20 of the primitive actions. By providing the constituent subtasks of a target subtask in the form of its children nodes, we provide the user with a manageable amount of new information. This way, if there are subtasks that the user is unfamiliar with, they can query them directly without sitting through potentially unnecessary explanations.

Discussion

In this paper we presented a robotic system capable of learning and redeploying a representation of a collaborative assembly task via natural language interactions. Our system uses the OntoSem cognitive architecture to generate HTNs from a natural language demonstration of the task by a human partner. The HTN can be subsequently be deployed by an autonomous robot to complete the task with a human, during which time its structure can be queried by the user. The demonstration with a human participant in the previous section indicates that our system is indeed capable of generating a viable and expressive HTN from only a single demonstration, the structure of which is capable of being transparently communicated via natural language.

As the system in its current incarnation is a proof-of-concept integration of many active areas of research, there are a number of issues that must be addressed before this system can be widely deployed. One of the drawbacks of the current iteration of the system is that the dialogue system used to answer queries about the task is completely separate from the OntoSem architecture. As a result, our system did not leverage OntoSem' sophisticated language parsing techniques, and resorted to a small set of pre-defined (but parameterized) queries. While this limited the flexibility and naturalness of the interaction, it is not a theoretical limitation of our system, but rather a pragmatic one.

In a similar vein, task demonstrations in their current form must be performed in a relatively unnatural way. Currently, in order to teach the system a new task, the user must use spoken instructions in conjunction with the corresponding sequence of actions provided via teleoperation. Ideally, the system should be able to learn a task from spoken instruction alone such as in Scheutz et al. (2017), or by observation, e.g. Nejati, Langley, and Konik (2006). However, the tasks learned in these papers are not collaborative, and so side-step a layer of complexity present in the collaborative domain.

Currently, the field of AI is facing a crisis of opacity. As AI and machine learning algorithms become increasingly more sophisticated and more effective at solving a wider range of problems, the more difficult they become to interpret. While as of yet the consequences of this have been relatively minimal, as these systems proliferate and come to play a larger role in society, the inscrutability of these systems will have profound repercussions.

This is especially relevant to robots, which by their physically embodied nature pose a danger to human beings. Therefore it is imperative that we design these systems with an eye towards transparency, that their capabilities and limitations be readily interpretable to human users. One solution is that AI and robotics researchers take inspiration from "classical" AI frameworks, as we have here. These frameworks, such as HTNs, while expressive and intuitive, were difficult to employ in practice as they are cumbersome to construct for complex tasks. However, by leveraging recent advances in AI and machine learning, many of these difficulties can be sidestepped, enabling a new era of transparent robotics and AI.

References

Allen, J.; Chambers, N.; Ferguson, G.; Galescu, L.; Jung, H.; Swift, M.; and Taysom, W. 2007. Plow: A collaborative task learning agent. In *AAAI*, volume 7, 1514–1519.

Brawer, J.; Mangin, O.; Roncone, A.; Widder, S.; and Scassellati, B. 2018. Situated Human–Robot Collaboration: predicting intent from grounded natural language. In *Intelligent Robots and Systems (IROS), 2018 IEEE/RSJ International Conference on*. IEEE.

Chandrasekaran, B.; Tanner, M. C.; and Josephson, J. R. 1989. Explaining control strategies in problem solving. *IEEE Intelligent Systems* 4:9–15, 19–24.

Erol, K.; Hendler, J. A.; and Nau, D. S. 1994. Umcp: A sound and complete procedure for hierarchical task-network planning. In *AIPS*, volume 94, 249–254.

Garrido-Jurado, S.; Muñoz-Salinas, R.; Madrid-Cuevas, F.; and Marín-Jiménez, M. 2014. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition* 47(6):2280 – 2292.

Hayes, B., and Scassellati, B. 2016. Autonomously constructing hierarchical task networks for planning and human-robot collaboration. In *International Conference on Robotics and Automation (ICRA)*.

Kim, T., and Hinds, P. 2006. Who should i blame? effects of autonomy and transparency on attributions in human-robot interaction. In *ROMAN 2006–The 15th IEEE International Symposium on Robot and Human Interactive Communication*, 80–85. IEEE.

Lee, J., and Moray, N. 1992. Trust, control strategies and allocation of function in human-machine systems. *Ergonomics* 35(10):1243–1270.

Leonard, M.; Graham, S.; and Bonacum, D. 2004. The human factor: the critical importance of effective teamwork and communication in providing safe care. *BMJ Quality & Safety* 13(suppl 1):i85–i90.

McShane, M.; Nirenburg, S.; and Beale, S. 2016. Language understanding with ontological semantics. *Advances in Cognitive Systems* 4:35–55.

Mohan, S.; Mininger, A.; Kirk, J.; and Laird, J. E. 2012. Learning grounded language through situated interactive instruction. In *AAAI Fall Symposium: Robots Learning Interactively from Human Teachers*, 30–37.

Mohseni-Kabir, A.; Rich, C.; Chernova, S.; Sidner, C. L.; and Miller, D. 2015. Interactive hierarchical task learning from a single demonstration. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*, 205–212. ACM.

Nau, D.; Cao, Y.; Lotem, A.; and Munoz-Avila, H. 1999. Shop: Simple hierarchical ordered planner. In *Proceedings of the 16th international joint conference on Artificial intelligence-Volume 2*, 968–973. Morgan Kaufmann Publishers Inc.

Nejati, N.; Langley, P.; and Konik, T. 2006. Learning hierarchical task networks by observation. In *Proceedings of the 23rd international conference on Machine learning*, 665–672. ACM.

Nirenburg, S.; McShane, M.; Beale, S.; Wood, P.; Scassellati, B.; Magnin, O.; and Roncone, A. 2018. Toward human-like robot learning. In Silberztein, M.; Atigui, F.; Kornyshova, E.; Métais, E.; and Meziane, F., eds., *Natural Language Processing and Information Systems*, 73–82. Cham: Springer International Publishing.

Pfrimmer, D. 2009. Teamwork and communication. *The Journal of Continuing Education in Nursing* 40(7):294–295.

Roncone, A.; Mangin, O.; and Scassellati, B. 2017. Transparent role assignment and task allocation in human robot collaboration. *Robotics and Automation (ICRA), IEEE International Conference on*.

Sacerdoti, E. D. 1975. A structure for plans and behavior. Technical report, Artificial Intelligence Center, Sri International, Menlo Park, CA.

Samek, W.; Wiegand, T.; and Müller, K.-R. 2017. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv preprint arXiv:1708.08296*.

Scheutz, M.; Schermerhorn, P.; Kramer, J.; and Anderson, D. 2007. First steps toward natural human-like hri. *Autonomous Robots* 22(4):411–423.

Scheutz, M.; Krause, E.; Oosterveld, B.; Frasca, T.; and Platt, R. 2017. Spoken instruction-based one-shot object and action learning in a cognitive robotic architecture. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, 1378–1386. International Foundation for Autonomous Agents and Multiagent Systems.

Shah, J., and Breazeal, C. 2010. An empirical analysis of team coordination behaviors and action planning with application to human–robot teaming. *The Journal of the Human Factors and Ergonomics Society* 52(2):234–245.

Wahn, B.; Schwandt, J.; Krüger, M.; Crafa, D.; Nunnendorf, V.; and König, P. 2016. Multisensory teamwork: using a tactile or an auditory display to exchange gaze information improves performance in joint visual search. *Ergonomics* 59(6):781–795.

Wang, L.; Jamieson, G. A.; and Hollands, J. G. 2009. Trust and reliance on an automated combat identification system. *Human factors* 51(3):281–291.

Wilkins, D. E. 1984. Domain-independent planning representation and plan generation. *Artificial Intelligence* 22(3):269–301.

Zeylikman, S.; Widder, S.; Roncone, A.; Mangin, O.; and Scassellati, B. 2018. The HRC model set for human-robot collaboration research. In *Intelligent Robots and Systems (IROS), 2018 IEEE/RSJ International Conference on*. IEEE.