Dialog with Robots: Papers from the AAAI Fall Symposium (FS-10-05)

Instruction Taking in the TeamTalk System

Alexander I. Rudnicky and **Aasish Pappu** and **Peng Li** and **Matthew Marge**
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213 USA

Abstract

TeamTalk is a dialog framework that supports multi-participant spoken interaction between humans and robots in a task-oriented setting that requires cooperation and coordination among team members. We describe two new features to the system, the ability for robots to accept and remember location labels and the ability to learn action sequences. These capabilities were made possible by incorporating an ontology and an instruction understanding component into the system.

Introduction

Natural language is commonly thought to be an intuitive interface between humans and complex computer systems. Language is said to allow humans to communicate on a more abstract level and express goals and desired outcomes rather than specify explicit parameters or supply low-level instructions. But in practice, dialog applications tend to finesse these issues by applying strong constraints at the task level and by relying on a common understanding of the domain shared by the system and user. Such an approach works well for applications such as information access or transaction processing, but not necessarily for less structured domains. Applications designed for controlling robots should be able to handle unpredictable changes in surroundings and evolving requirements. We believe they can do so through high-level language-based interaction. Supporting such interaction, in turn, requires additional mechanisms to be introduced into a dialog system. We describe two such mechanisms: an ontology that keeps track of acquired knowledge and a learning process that allows the robot to acquire new action combinations.

The TeamTalk System

TeamTalk is based on the Olympus/Ravenclaw (Bohus et al. 2007) architecture developed at Carnegie Mellon. Ravenclaw (Bohus and Rudnicky 2009) is a flexible mixed-initiative dialog manager (DM) that uses plan-like representations to describe possible interactions with humans. In the case of TeamTalk, Olympus has been augmented with a multi-modal interface that allows the system to accept and to generate information from a tablet computer that displays a map and allows the user to input information using a stylus. Tablet inputs are converted to concepts (equivalent to those that are generated by the understanding component) and routed to the DM. Similarly, outputs are generated in an abstract form then realized appropriately by the tablet application. The DM mediates between the user and the TeamTalk Domain Reasoner, which holds the state of the system and communicates with the other components of the overall system (which in turn interface with robots). The components diagram of the system are shown in Figure 1. TeamTalk can spawn and interact with multiple robots, and can simultaneously support multiple human agents.

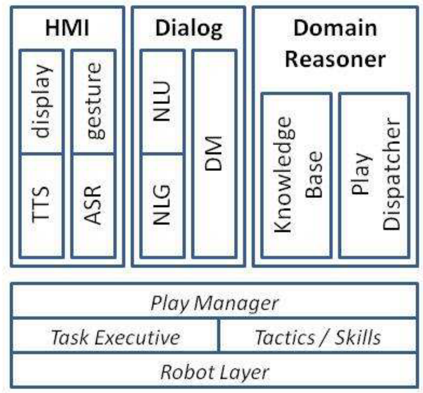

Figure 1: The TeamTalk architecture.

Search-and-Rescue Domain

The search-and-rescue domain supports a rich variety of interactions between humans and robots. Search takes place in an unknown environment that needs to be explored by the team in a coordinated fashion. Features in the environment need to be described so that they can be referenced later. Search is also naturally conducive to delegating tasks to one or more robots (e.g., the human might ask a robot to go down a corridor and check each room for a target). But the process for doing so may need to be explained to the robot. As such, the domain accommodates many possible scenarios, but its complexity can be managed by controlling the amount of knowledge each robot maintains about the environment. Search and rescue is an established focus in the robotics community (Kitano et al. 1999). We have exercised TeamTalk in the search domain, both with robots in the environment (Dias et al. 2006) and with robots in a virtual world (Harris and Rudnicky 2007; Marge et al. 2009) based on the USARSim platform (Wang et al. 2005).

Ontology

The TeamTalk ontology keeps track of relevant entities in the search domain. The ontology is based on the OWL representation, using the Protegé environment (Knublauch et al. 2004). The ontology component responds to queries from the Domain Reasoner and accepts dynamic updates (e.g., the identity of newly labeled locations). The ontology also maintains a complete history of events that have occurred and stores this information for use in later sessions. Events are organized into *episodes* representing a coherent, temporally-related set of events (e.g., a spoken interaction followed by the execution of an action series would constitute an episode). TeamTalk is able to process simple factual queries about the past (e.g., when a location was defined or who was involved in a given episode). The system currently maintains a single knowledge base; once a new instance is created it is immediately transparently available to all robots. While we understand that private knowledge may have a role in team activities it does not appear critical for our current scenarios.

Plays

TeamTalk is able to learn and represent action sequences consisting of individual commands and aggregates. For example, the current system can learn a zig-zag behavior by listening to a series of movement instructions. We refer to these as *plays*, and the implementation makes use of the Play Manager (PM) mechanism described in (Dias et al. 2006). In TeamTalk, we add a Play Dispatcher to allow multiple plays to execute simultaneously. Plays are implemented as dynamic Ruby programs. Each play consists of steps corresponding to individual commands as well as standard programming constructs such as blocks and iteration. The human specifies a new play interactively while the robot manages the recording process and performs consistency checks. This ensures reasonable behaviors, although these may not necessarily abstract to reasonable interaction with the environment. Plays are stored in the ontology and subsequently become available to all robots. Plays provide functional descriptions of new robot behavior; we anticipate introducing more flexible plan-like representations in the future.

Multi-Participant Dialog

In our initial experiments with TeamTalk we found that a conventional dialog architecture could be used to support one-human / many-robot scenarios through the introduction of strict floor-management protocols. Extending this scenario to multiple human participants proved to not be possible, due to the need to simultaneously manage multiple physical (i.e., audio) and virtual (conversational) channels. We consequently began to develop a new architecture that would allow a robot to manage multiple dialogs, corresponding to multiple dynamically created channels, with overall supervision provided by a new Conversation Manager. Our current work focuses on the development of this new level of control and its integration into the existing robot system. The task of the Conversation Manager is to generate appropriate behavior under different conversation states, which we currently categorize as follows: (1) one human-to-one robot, (2) many humans-to-one robot, and (3) many-to-many. Each case can be represented as a state diagram, which keeps track of the conversation and which determines the robot's current responsibilities (e.g., listening, talking, yielding, etc). Our ultimate goal is to develop a control mechanism similar in behavior to that of humans.

Acknowledgments

This has been supported by The Boeing Corporation and by the National Science Foundation. We would like to thank Benjamin Frisch for his invaluable work in supporting the Olympus environment and for designing the TeamTalk virtual environment.

References

Bohus, D., and Rudnicky, A. 2009. The RavenClaw dialog management framework: Architecture and systems. *Computer Speech and Language* 23(3):332–361.

Bohus, D.; Raux, A.; Harris, T. K.; Eskenazi, M.; and Rudnicky, A. I. 2007. Olympus: an open-source framework for conversational spoken language interface research. In *Bridging the Gap: Academic and Industrial Research in Dialog Technology Workshop*. HLT/NAACL.

Dias, M. B.; Harris, T. K.; Browning, B.; Jones, E. G.; Argall, B.; and Veloso, M. 2006. Dynamically formed human-robot teams performing coordinated tasks. In *AAAI Spring Symposium: To Boldly Go Where No Human-Robot Team Has Gone.*

Harris, T. K., and Rudnicky, A. I. 2007. TeamTalk: A platform for multi-human-robot dialog research in coherent real and virtual spaces. In *Proceedings of AAAI*.

Kitano, H.; Tadokoro, S.; Noda, I.; Matsubara, H.; Takahashi, T.; and Shinjou, A. 1999. RoboCup Rescue: search and rescue in large-scale disasters as adomain for autonomous agents research. In *IEEE SMC '99 Conference Proceedings*, 739–743. Tokyo, Japan: IEEE.

Knublauch, H.; Fergerson, R. W.; Noy, N. F.; and Musen, M. A. 2004. The protege owl plugin: An open development environment for semantic web applications. 229–243. Springer.

Marge, M.; Pappu, A.; Frisch, B.; Harris, T. K.; and Rudnicky, A. I. 2009. Exploring spoken dialog interaction in human-robot teams. In *Proceedings of Robots, Games, and Research: Success Stories in USARSim*. St. Louis, MO: IEEE.

Wang, J.; Lewis, M.; Hughes, S.; Koes, M.; and Carpin, S. 2005. Validating USARsim for use in HRI research. In *Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting*, 457–461.