Title: Qualitative representation to enable Companion to interact in an environment

Group: Irina Rabkina and Willie Wilson

Abstract

The Companion cognitive architecture is capable of simulating human cognition through a suite of cognitive components to reason using analogy, deduction, qualitative spatial relations, qualitative processes, plans, and more. Typically, Companion can “interact” with external entities via language and/or sketch. Recent developments have extended the interaction capabilities, enabling companion to be embodied and interact with its environment. We consider three such cases, an agent acting in a Minecraft world, a small Cozmo robot moving blocks on a table, and an intelligent kiosk (PsiKi) that provides information in the CS building. For each of these cases, we plan to develop a common representation that allows Companion to have consistent reasoning capabilities.

Introduction

Many cognitive architectures have been demonstrated to control a body (physical or virtual) and interact in an environment. Some of these cognitive architectures, such as DIARC [1], are designed specifically for this purpose, and others have been adapted to control robots or virtual agents (e.g., SOAR [2], ACT/R-E [3]). Until recently, Companion was limited in its ability to interact, being able to read text and understand sketches, but recently Companion has taken on a couple of embodiments, including a virtual agent for an information kiosk and an autonomous agent in a Minecraft environment. Thus far, these efforts share little commonality, simply taking advantage of Companion’s distributed architecture that allows remote agents to communicate via KQML [4]. The intent of this project is to unify the approaches by proving a common representation to act as an interlingua in communications between the high-level reasoning capabilities of Companion and the low-level signal processing and action execution mechanisms in environments for virtual agents and robots.

Interlingua

The Companion cognitive architecture has many facilities to simulate human-like reasoning, including deductive reasoning, analogical reasoning, natural language understanding, and planning. It is not designed for processing streams of signal data, but these capabilities are readily available in other tools. For example, Microsoft's Platform for Situated Intelligence (\psi) is specifically designed for processing multimodal signal data [5]. Similarly, the Robot Operating System (ROS) is designed to handle low level signal processing for robots [6]. For the small consumer robot Cozmo, their libraries provide basic signal processing capabilities. As for virtual environments, Microsoft's Malmo provides an interface to Minecraft which exposes similar data about the environment of an agent in a Minecraft world.

To enable each of these environments to use an interlingua in communicating with Companion, we propose developing a common representation that expresses three aspects of the communication:

1. Reporting events: processed signal data is converted to a qualitative representation that is provided to Companion.
2. Requests for data: Companion asks the lower-level system for specific information.
3. Executing actions: Companion instructs the lower-level system to have some action executed by the agent.

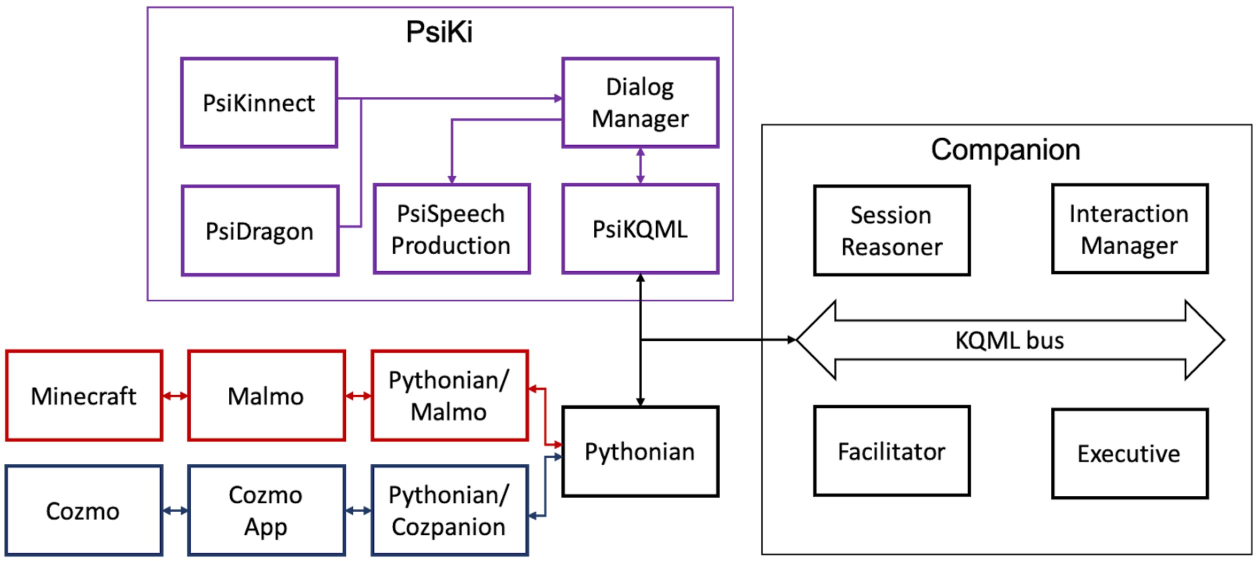
In developing the interlingua, we will review existing interfaces between Companion and Minecraft and Companion and the information kiosk (PsiKi). We will also gain inspiration from other cognitive architectures, such as DIARC, that have a clean separation of robot control and high-level reasoning. Additionally, since KQML is used as a transport mechanism, we will relate the interlingua with the performatives defined in KQML.

Use Cases

To demonstrate the capabilities afforded by the interlingua, we will show that the Companion reasoning capabilities can be accessed in a use cases for three different environments. For the Companion and PsiKi connection, we will demonstrate how the interlingua is used in question answering. Specifically, the PsiKi environment, which uses \psi, will inform Companion of the question being asked, and it will respond with a sequence of actions, which may be a response for the agent to say/display or to show a map. For the Companion and Minecraft connection, we will demonstrate how the interlingua is used to communicate the Minecraft agent's environment (including its physical surroundings, other entities, and any observed actions) to the Companion. The Companion will then respond with an action for the Minecraft agent to accomplish. For the Companion and Cozmo connection, we will demonstrate how the interlingua is used to recognize a person and respond with a pre-programmed behavior specific to that person. For example, when Cozmo sees Willie, the robot may spin and say a short greeting.

Architecture

To build each of these use cases, we will use a combination of existing tools and infrastructure as well as implement a couple of new components. The figure below depicts most of the components involved in this project. In Companion, we will primarily be using the Interaction Manager, which uses FIRE to query the KB, do analogical reasoning, and find plans. For agents to communicate with Companion, we will use the KQML messaging bus that Companion already uses. PsiKi already has a KQML component to communicate with Companion, and this component will need to be revised to support the new interlingua. There is also a generic Pythonian component that allows Companion to connect with Python tools via KQML. There is an existing extension of this component to connect to Malmo, which in turn connects to Minecraft. A new extension of the Pythonian component will need to be developed to connect to the Cozmo App, which then connects to the Cozmo robot.



Plan

The plan for accomplishing the goals of this project consists of developing the interlingua, updating the KB in Companion, revising existing KQML components (PsiKQML, Pythonian), update Companion plans, and building a new Pythonian component for Cozmo. In the first step, developing the interlingua, we will define a representation for each of the three types of communication described above. The end product of this step is knowledge representation stored in krf files. For the second step, we will update the KB in Companion by loading the krf files defined in step 1. Next, components communicating with Companion via KQML need to be updated to use the new interlingua. Some of these changes may be trivial, as they are simply extensions to current KQML messages (i.e. ask and achieve). Most of the effort will be in updating the components to construct event messages to be sent to Companion. New messages being sent to Companion will require updates in Companion. Since many messages are handled in plans, many of these revisions are expected to primarily consist of modifying existing plans. The final step, creates Cozpanion, which is a new Pythonian agent to connect with the Cozmo robot. This component will utilize the Cozmo SDK to connect with the robot, receive data from the robot, and send instructions to it. This step is a risk since it involves writing a new component, and we only have a few days experience working with the SDK. If necessary, this final step will be dropped.

References

1. Scheutz, M., Williams, T., Krause, E., Oosterveld, B., Sarathy, V., & Frasca, T. (2019). An overview of the distributed integrated cognition affect and reflection DIARC architecture. In *Cognitive Architectures* (pp. 165-193). Springer, Cham.

2. Laird, J. E., Kinkade, K. R., Mohan, S., & Xu, J. Z. (2012, July). Cognitive robotics using the soar cognitive architecture. In *Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence, Cognitive Robotics*.

3. Trafton, J. G., Hiatt, L. M., Harrison, A. M., Tamborello II, F. P., Khemlani, S. S., & Schultz, A. C. (2013). Act-r/e: An embodied cognitive architecture for human-robot interaction. *Journal of Human-Robot Interaction*, *2*(1), 30-55.

4. Finin, T., Fritzson, R., McKay, D., & McEntire, R. (1994, November). KQML as an agent communication language. In *Proceedings of the third international conference on Information and knowledge management* (pp. 456-463). ACM.

5. Bohus, D., Andrist, S., & Jalobeanu, M. (2017, November). Rapid development of multimodal interactive systems: a demonstration of platform for situated intelligence. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction* (pp. 493-494). ACM.

6. Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., ... & Ng, A. Y. (2009, May). ROS: an open-source Robot Operating System. In *ICRA workshop on open source software* (Vol. 3, No. 3.2, p. 5).