Learnable Controllers for Adaptive Dialogue Processing Management

Geert-Jan M. Kruijff and **Hans-Ulrich Krieger***
German Research Center for Artificial Intelligence
DFKI GmbH, Saarbrücken Germany
{gj,krieger}@dfki.de

Abstract

The paper focuses on how a model could be learnt for determining at runtime how much of spoken input needs to be understood, and what configuration of processes can be expected to yield that result. Typically, a dialogue system applies a fixed configuration of shallow and deep forms of processing to its input. The configuration tries to balance robustness with depth of understanding, creating a system that always tries to understand as well as it can. The paper adopts a different view, assuming that what needs to be understood can vary per context. To facilitate this any-depth processing, the paper proposes an approach based on learnable controllers. The paper illustrates the main ideas of the approach on examples from a robot acquiring situated dialogue competence, and a robot working with users on a task.

Introduction

When talking with other humans, we sometimes listen more carefully, sometimes less. We figure out how much we need to understand, and adjust on the fly how much we process while still keeping on interacting in a contextually appropriate way. We have ways of balancing our resources given noise, uncertainty, cost – as language users we are, in many sense of the word, rather pragmatic.

We take inspiration from this observation to outline here an approach to "any-depth" processing of situated dialogue in human-robot interaction. The aim of the approach is to enable a system to decide online, what processes need to be applied to the input given an (expected) state to be reached, while trying to achieve that as fast as possible. The system dynamically establishes an optimal process configuration to be executed, for part or all of the input.

Why could that be potentially beneficial for situated dialogue processing? True, there is a certain appeal to processing input to as deep a level of understanding as possible. The general idea is that the more the system understands, the more it can figure out what is being talked about, and the better it can act upon that (perceived) understanding. But this is difficult to achieve for spoken dialogue. Input is incomplete, noisy, the system is typically uncertain how to understand it, and how to act upon it. And at the same time we need to maintain a natural flow of conversation: Asking for clarification after each utterance is not helpful.

Technically, the field has tried to address these problems by applying more robust forms of processing, to deal with interpretation and decision making under uncertainty. We are able to deal with missing or wrong input, and still provide an analysis to act upon. A wide variety of approaches exists for how processes and their outputs can be combined, e.g. using fixed configurations (cf. (McTear 2002)) or more flexible divide-and-conquer strategies (Wahlster 2000), in which the same input is provided to different kinds of processes in the expectation that at least one succeeds. Up to a "fixed" depth, the system tries to understand as much as it can. We propose to take these ideas a step further, to decide at runtime what configuration of processes to apply to (parts of) the input to construct analyses that provide that as much as depth as necessary, where necessary.

Intuitively, the approach works as follows. We start from an input stream, say a sequence of words. We assume that the system is able to assign "signs" to some or all of these words. These available signs build up a state space, factored by the possible levels of interpretation. We use this state space as a common control state space between different processes that act on signs. The goal is now for a controller to select a configuration of one or more processes that run on some or all of these signs. Process selection is guided by two factors. The first factor is the expected value of running a process on specific signs, hopefully yielding partial analyses. The expected value is based on how useful these analyses are in relation to a goal state we need to achieve. In the examples below we give two illustrations of such goals, and the reward feedback they yield. One example comes from language acquisition, and concerns the issue of learning to establish enough structure in linguistic analyses to help determine how to interpret a visual observation. The other example comes from task-driven dialogue, and concerns the detection of enough content to determine whether a user indeed made the dialogue move the system expected her to make. The other factor in process selection

*The research presented here was supported by the EU FP7 IP project "Natural Human-Robot Cooperation in Dynamic Environments," (ICT-247870; http://www.nifti.eu) and by the EUF P7 IP project "Cognitive Systems that Self-Understand and Self-Extend" (ICT-215181, http://cogx.eu).

is the potential cost of running a process (typically, time).

Formally, the approach employs ideas from reinforcement learning (Sutton and Barto 1998), including factored models for Markov Decision Processes (Boutilier, Dean, and Hanks 1999; Strehl, Diuk, and Littman 2007), planning with finite receding horizons (Barto, Bradtke, and Singh 1995; Morisset and Ghallab 2008), and self-aware learning (Li, Littman, and Walsh 2008). The novelty of the approach is in use of learnable controllers for managing what processes are invoked to interpret an utterance, and the flexibility in selecting processes to run over subsets of signs.

Examples

Acquiring language competence Picture we have a robot which develops its visual and linguistic competence by interacting with the environment, and a human tutor. Initially, the robots visual resolution is low. It can only see colored blobs, which it recognizes in terms of assigning a blob a distribution over possible color values. Linguistically, we assume that the robot can process audio to words, but that's all.

We start off by showing the robot a blue object, saying "this is blue." The robot lacks a model of how to link these words to what it sees. All it can do is build grounded linguistic sign structures for each of them. A sign structure links a word on the one hand to a structural context (i.e. a part of a factored state), and on the other hand to a meaning that relates it to a situated use. The links in the resulting triadic structure are all weighted. For example, in the presence of "this is" the word "blue" might predict the distribution over possible values the robot has just seen. The expected "strenght" of the prediction is reflected by the weight it has in the context of this sign. We continue this for a while, illustrating different colors to the robot, telling it what it sees. During this time, the robot builds up a space of possible signs. Each time it matches linguistic expressions to their ability to predict a color distribution. A prediction that provides a good match (statistically) to an observation results in an improved weighting of prediction in the sign, while different expressions help establish an extensional representation of the structural contexts the word appears in. (This sets this idea of sign apart from accounts like (Steels 2006).)

As the space of signs grows, so do the number of possible signs that the robot can assign to an input sequence. Processing at this point is, in its simplest form, a matter of sign selection. The task for the controller is thus to select that combination of signs and processes that can be expected to yield the best match to an observation. The feedback resulting from matching the selected sign(s) with the observations is then used both to drive learning with the controller, and the adaptation of the signs itself.

There is little or no need for genuine structure if all the robot hears is color terms. This changes once we start varying the ways we use words, and what we talk about. A nice example of this is a type of experiment discussed for child language acquisition in (Waxman and Gelman 2009): Do children distinguish between "this is a dax" and "this is dax"? Here we see the impact of taking structural aspects of state into account in a sign. The signs for "dax" are different, and accordingly their predictions will be different. But to bring this difference about, we need to do more than select a (single) sign. Differentiating requires the possibility to form more complex signs – i.e. building up a simple structure over "a dax" should those two words be present. Using a mechanism for self-aware learning (Li, Littman, and Walsh 2008) the controller can first of all raise the explicit need for such additional structure (as an aspect of the factored state), to subsequently learn what processes to apply to meaningfully distinguish uses like "this is a dax" from "this is dax."

Varying depth in task-driven interaction Human-robot interaction often deals with a setting in which a human and the robot communicate, to work on some shared task. Confirmations are frequent in such dialogues. "Yes," "okay," – their occurrence can be predicted in a dialogue, forming a natural part of a (predictive) dialogue model. Given this expectation of a confirmation, the controller can opt to select a shallow process that provides just as enough information that the human confirmed, as a deep analysis would. At the same time, the determination of the actual sign space for an input sequence enables the controller to trade this possibility off against the need to process additional signs – e.g. if the human replies with "yes, but ..." which necessitates a deeper interpretation of the "but..." condition (alike "a dax").

References

Barto, A.; Bradtke, S.; and Singh, S. 1995. Learning to act using real-time dynamic programming. *Artificial Intelligence* 72(1-2):81–138.

Boutilier, C.; Dean, T.; and Hanks, S. 1999. Decision-theoretic planning: Structural assumptions and computational leverage. *Journal of AI Research* 11:1–94.

Li, L.; Littman, M.; and Walsh, T. 2008. Knows what it knows: A framework for self-aware learning. In *Proceedings of the Twenty-Fifth International Conference on Machine Learning (ICML-08)*.

McTear, M. 2002. Spoken dialogue technology: enabling the conversational interface. *ACM Computing Surveys* 34(1):90 – 169.

Morisset, B., and Ghallab, M. 2008. Learning how to combine sensory-motor functions into a robust behavior. *Artificial Intelligence* 172(4-5):392–412.

Steels, L. 2006. Semiotic dynamics for embodied agents. *IEEE Intelligent Systems* 21(3):32–38.

Strehl, A.; Diuk, C.; and Littman, M. 2007. Efficient structure learning in factored-state MDPs. In *Proceedings of the 22nd national conference on Artificial intelligence (AAAI'07)*, 645–650.

Sutton, R., and Barto, A. 1998. *Reinforcement Learning: An Introduction*. Adaptive Computation and Machine Learning. The MIT Press.

Wahlster, W., ed. 2000. *Verbmobil: Foundations to Speech-to-Speech Translation*. Artificial Intelligence. Springer.

Waxman, S., and Gelman, S. 2009. Early word-learning entails reference, not merely associations. *Trends in Cognitive Sciences* 13(6):258–263.