From Rule Extraction to Active Learning Symbol Grounding

Henrik Jacobsson, Geert-Jan Kruijff & Maria Staudte
{henrik.jacobsson,gj,maria.staudte}@dfki.de

***Abstract*— The paper focuses on a fundamental learning problem in adaptive, embodied cognitive systems: Namely, how to learn discrete models of situated, embodied experience which can act as a mediation between sensori-motoric experience and high-level cognitive processes. The paper suggests to address the problem using a combination of bottom up active learning of embodied concepts solely on the basis of the actions and perceptions of the robot, and top-down information obtained through interaction with other agents. The embodied concepts are constructed to be informative for the robot in terms of its sensorimotor prediction capability. From that point the effort of constructing humanlike concepts is shifted towards producing a translation between the sensorimotor based bottom-up ontology and more conventional top-down constructed ontologies. The suggested framework is based on a parameter free rule extraction algorithm that successfully has been applied to the problem of creating finite state descriptions of large, complex and even chaotic simulated dynamic systems. We will briefly describe how this algorithm can be ported to an autonomous robot domain.**

I. Introduction

A fundamental problem in the development of cognitive systems for embodied agents is that of "symbol grounding", i.e. the question how to connect high-level cognitive processes and the representations they create to an embodied, situated understanding of the environment. Empirical studies in developmental psychology and psycholinguistics suggest that this connection is at least in part *mediated* by a distributed collection of category systems that model concepts [1], [2], [3]. In this position paper, we outline an approach to learning such category systems in an active fashion, on the basis of a combination between embodied exploration of an environment and interaction with other agents. The resulting category models do not only categorize available sensory input, but also predict how this input may change under the influence of actions, or environmental dynamics; (cf. the notion of simulation in [2]). This approach is still under development, but promises to provide means for acquiring a genuinely *embodied* understanding of a situation into which we can base higher-level, deliberative situated cognitive processing.

The idea behind the approach is to start from conceiving of sensorimotoric input ("experience") about the environment as a *continuous dynamic system*. This system we need to abstract to a more *discrete* description. This abstracted description takes the form of a discrete computational model that, when given the same input, approximates behaviour of the dynamic system it describes[1].

To implement such an approach, we employ techniques developed for rule extraction from neural networks. The goal there is to translate large and complex trained models into something more comprehensible [7], [8]. The Crystallizing Substochastic Sequential Machine Extractor, or `CrySSMEx` [9], presents one algorithmic approach that has successfully beeen applied to various types of dynamic systems, including Recurrent Neural Networks (RNNs). We suggest that, with some modifications, it could in also be successfully applied to the environment of a robot. Moreover, `CrySSMEx` is suitable for the purpose of concept creation because, at its core, it relies on the generation of a hierachical description of physical characteristics of the state space of the RNN. In robotics terminology, `CrySSMEx` builds up something that resembles a sensorimotor ontology. `CrySSMEx` also constructs a sequence of gradually refined stochastic finite state machines, which are essentially capable of predicting the ontological class.

Our conjecture is that these, bottom-up constructed, representations lead to a grounded set of truly embodied concepts. Embodied, since the only way the concepts can be inferred and validated is to actively collect data from the environment through actions.

The assumption underlying the extracting finite state descriptions from RNNs is that a finite state description will suffice to describe the possibly nonlinear and complex holistic mappings of the RNN. Naturally, this is a crude assumption and even proven to potentially give rise to conflicting results [10]. The advantages of this assumption could, however, be transferred to the sensorimotor interaction domain of a robot:

- Having discrete states makes it possible to use fairly simple probability theory and estimation of probabilities using frequency estimates.
- The model can split the sensor space into more and

[1]There is also a related approach which also investigate the development of language and sensorimotor skills [4], [5], [6]. This came to our attention during the last stages of preparing the camera ready version of this paper. Our initial analys reveals that there are strong similarities in that work with what we are proposing here. There are some important subtle differences, however. The central "meta learner" we suggest here is more explicitly reductionistic and the training mechanisms we suggest are distributed and localized closer to the sensory modalities (and we will never have to explicitly deal with the actual perceptions in the central active learner module). The construction of a finite state machine also make it possible to reason about the consequence of sequences of actions (and planning for disambiguation of sensors). We do also assume a preexisting natural language module and therefore communication skills is not something that needs to be learned from scratch [4].

more special situations, which are increasingly managable for learning algorithms.
- Erroneous predictions can be attributed to identifiable and labelled situations (which is the basis for collecting new data sets).
- Planning and prediction in a discrete model is comparably straightforward to implement.
- A discrete finite state model of the world is more likely to be possible to translate into conceptual representations that underly e.g. human language (which, after all also consist of a discrete set of words and categories).

II. Background

We will now briefly sketch some of the features of `CrySSMEx`. For details about `CrySSMEx`, we refer to the original paper [9]. `CrySSMEx` is parameter free, deterministic, can handle missing data, and produces any-time results. It is also considerably more efficient than earlier algorithms because of the active learning characteristics [11]. `CrySSMEx` is a semi-active learner in that it selects subsets of the data set thate are maximally informative for situations/states that are identified as problematic in terms of prediction capability.

The systems susceptible to `CrySSMEx` analysis belong to a broad set of dynamic systems (further defined in [9]) that have three vector spaces; input, state and output. The dynamics of the systems is such that future states and outputs are determined given the current state and input to the system. How these vectors are determined is, however, concealed from `CrySSMEx`. Some other assumptions are identified in Table I.

The models generated by `CrySSMEx` are sequences of Crystalline Vector Quantizers (CVQs) and Substochastic Sequential Machines (SSMs) [9]. The CVQ describes a hierarchical division of the state space of the RNN. The SSM describes how these discretized states relate to each other temporally by describing how different inputs cause transtions among the states. The goal of `CrySSMEx` is to find states that are as informative as possible for predicting future outputs (and states) of the RNN (cf. Crutchfield's "causal states" [12]).

The CVQ is generated by training if on data selected on basis of what the SSM cannot predict. And the SSM is generated on basis of how the CVQ divides the state space.

`CrySSMEx` can be seen as a meta learner that administrates data sets, learning algorithms and trained models. The underlying training algorithm has been deliberately selected to be poor[2] to give room for improvements. Despite this, `CrySSMEx` has surpassed its predecessors by being successfully applied to RNNs with up to 1000-dimensional state spaces and has also managed to generated finite state models that approximately predict chaotic systems.

III. Porting `CrySSMEx` to a robot

The idea is to let the input, state and output spaces of the RNN be replaced by actions and perceptions of the robot.

[2]Vector quantization with only initiation and no optimization of model vectors.

RNN	Robot
Full observability of state	Partial and noisy obervability through sensors
Homogenous representation of state as vectors	Representation dependent on underlying sensor modality (vision, laser range finder, touch etc.)
Deterministic and mathematically defined behaviour	Indeterministic environment where external event may occur cause changes
Complex and possibly even chaotic dynamics	Fairly "simple" dynamics
N-dimensional	3-dimensonal (or 2.5D)

TABLE I

SOME OF THE DIFFERENT ASSUMPTIONS BETWEEN RNN AND A ROBOT SENSORIMOTOR ANALYSIS.

The SSM transitions between states are triggered by a finite set of actions (i.e. "input" to the environment). The CVQ is "equipped" with models that are trained on collected sensory data. The data should be collected for situations (i.e. SSM states) in which the SSM predictive capability is low. This was suggested in [9] as a curiosity driven approach for analysis of simulated systems ("Empirical Machine" [9]). The output of an RNN is discrete and it is essentially a function of the RNN state. It can simply be conceived as a discrete projection of the state. In the robotic scenario, we suggest that the output is initially represented as a simple discrete sensor, e.g. a touch sensor. The goal for `CrySSMEx` is to predict how the perceptions of this *seed sensor* are affected by the actions of the robot. It should do so by utilizing its other sensors by introducing trained models into the CVQ which analyses the sensor "space". Since the other sensors are used to predict the seed sensor, the correct prediction of these sensors will emerge as a requirement too.

There are of course a number of obstacles to be surmounted before this will be possible. The environment of a robot has several obvious differences from a neat simulated entity such as an RNN. Some of these differences are summarized in Table I. The strength and weakness of `CrySSMEx` is that it exploits the advantageous properties of RNNs. For example, since the RNN is deterministic it means that the successful extraction of a deterministic finite state machine is used as a termination criterion. Obviously, this cannot be strived for when modelling an indeterministic environment perceived through noisy sensors.

Many of these differences mean essentially that in the robot domain one has to be content with less accurate models being constructed. It also means that the main focus must lie on the validation of any attempt to improve the CVQ. The increase of predictive capability of the SSM as a result of a CVQ modification must be assessed prior to the execution of that modification.

In other words, we want `CrySSMEx` to perform what is typically done by the machine learning *researcher*: generation and labelling of training, verification and validation data sets, evaluation of trained models, an employment of successful models to do predict future data.

Initially, the SSM does not predict anything, so all rep-

resentation stems from the interaction through the robot's actions. This does, however, not mean that we insist on a purely bottom up learning approach; since the suggested architecture is essentially a meta learner which focuses on evaluating sensory data analysers, there is nothing that stops the actual learning phase to be overridden by providing some hardcoded sensory analysers as well. For example, the seed sensor may have good predictive power for predicting itself.

The corner stone for constructing a trained model is to maximize its ability to *generalize*. The expected generalization capability increases if more data is used for training. But in our approach, we suggest instead to autonomously specialize the data sets by focusing on specific situation (i.e. subproblems) so that the likelihood of a good generalization increases as a consequence of a simpler problem. `CrySSMEx` when applied to RNNs clearly shows the fruitfullness of this approach because it was successful even with very small subsets of data and with a mediocre training algorithm.

IV. Practical implementation

The approach is being implemented under the EU FP6 IST Cognitive Systems Integrated project: "Cognitive Systems for Cognitive Assistants - CoSy" (`www.cognitivesystems.org`). Within the CoSy project a cognitive architecture is being constructed that facilitates integration of various subarchitectures (in C++, Java, Python). The architecture is already integrated with subarchitectures such as speech recognition and synthesis, natural language processing, a handcrafted category association system, planning, computer vision, vision learning of features, kinematic control etc.

There is, in other words, no lack of sensors and actuators for an active learner to be integrated with in this framework. Of course, initially very simplified settings must be selected to test the feasibility of the approach.

The sensor modalities available are for example vision, laser ranger, touch sensors and the kinematic model etc. The range of sensors that can be used is not limited to typical sensors, but it can also be possible to use "sensors" that are the result of *ad hoc* processing of a sensor to filter out useful information, e.g. rather than using vision data directly, it can be preprocessed by a feature detector. Depending on the underlying modality, different learning algorithms must be empoyed on the collected data sets.

V. Goal

The goal scenario is to let `CrySSMEx` control the robot in a simple setting in which the robot can start from "scratch" to build up a grounded embodied world model. After a while it can hopefully sufficiently predict the consequences of its actions and has established a hierarchical model of world states. At this point, it should be possible to tag states and sets of states with labels through user interaction, e.g. by the user explaining to the robot that "your current action is called *bumping*" the robot could infer that the current state is part of a family of states that are involved in something which can be labelled 'bumping'. The ontology can then be used to assume that also closely related states may be labelled 'bumping' (something which may be verified through an interactive clarification dialogue [13], [14]). At this point there will essentially be a connection between a concept of the robot's world view, which consists of an enumeration of the space of all possible sensory inputs, and a human concept. To learn this concept is a trivial translation once the robot has the capability of predicting 'bumping'. And once the association is made, the user should be able to give high-level commands that can be translated to planning goals, e.g. "avoid bumping". The user may also actively help the robot to explore aspects of its sensorimotor abilities by commands such as "explore grasping" in which case the robot should generate the appropriate behaviour in order to assess and eliminate weaknesses in the predictive ability of states that are labelled 'grasping'.

References

[1] A. Glenberg, "What memory is for," *Behavioral & Brain Sciences*, vol. 20, pp. 1–55, 1997.

[2] L. Barsalou, "Perceptual symbol systems," *Behavioral & Brain Sciences*, vol. 22, pp. 577–660, 1999.

[3] G. Altmann and Y. Kamide, "Now you see it, now you don't: Mediating the mapping between language and the visual world," in *The Interface of Language, Vision, and Action: Eye Movements and The Visual World*, J. Henderson and F. Ferreira, Eds. New York NY: Psychology Press, 2004, pp. 347–386.

[4] P.-Y. Oudeyer and F. Kaplan, "Discovering communication," *Connection Science*, vol. 18, no. 2, pp. 189–206, 2006.

[5] F. Kaplan and V. V. Hafner, "Information-theoretic framework for unsupervised activity classification," *Advanced Robotics*, vol. 20, no. 10, pp. 1087–1103, 2006.

[6] P.-Y. Oudeyer, F. Kaplan, and V. V. Hafner, "Intrinsic motivation systems for autonomous mental development," *IEEE Transactions on Evolutionary Computation*, to appear 2007.

[7] R. Andrews, J. Diederich, and A. B. Tickle, "Survey and critique of techniques for extracting rules from trained artificial neural networks," *Knowledge Based Systems*, vol. 8, no. 6, pp. 373–389, 1995.

[8] H. Jacobsson, "Rule extraction from recurrent neural networks: A taxonomy and review," *Neural Computation*, vol. 17, no. 6, pp. 1223–1263, 2005.

[9] ——, "The crystallizing substochastic sequential machine extractor - `CrySSMEx`," *Neural Computation*, vol. 18, no. 9, pp. 2211–2255, 2006.

[10] J. F. Kolen, "Exploring the computational capabilities of recurrent neural networks," Ph.D. dissertation, The Ohio State University, Department of Computer and Information Sciences, 1994.

[11] D. Angluin, "Queries revisited," *Theoretical Computer Science*, vol. 313, no. 2, pp. 175–194, 2004.

[12] J. P. Crutchfield, "The calculi of emergence: Computation, dynamics, and induction," *Physica D*, vol. 75, pp. 11–54, 1994.

[13] G.-J. M. Kruijff, H. Zender, P. Jensfelt, and H. I. Christensen, "Clarification dialogues in human-augmented mapping," in *Proc. of the 1st Annual Conference on Human-Robot Interaction (HRI'06)*, Salt Lake City, UT, 2006.

[14] G.-J. M. Kruijff, J. Kelleher, G. Berginc, and A. Leonardis, "Structural descriptions in human-assisted robot visual learning," in *Proc. 1st Annual Conference on Human-Robot Interaction (HRI'06)*, 2006.