Concept Grounding in Robotic Capacity

Daniel Nikovski and Tom Mitchell

Problem: Concepts and categories are the basic building blocks of human and robotic knowledge. They are discrete in nature and provide a concise symbolic description of the situation that a robot is facing and the processes taking places in the world around it. In most intelligent systems, a human designer decides what concepts and categories the system will know about. We address the problem of replacing the human by an autonomous learning algorithm that would extract optimal discrete conceptual descriptions from sensory readings. Such an algorithm would create new concepts and ground them in perceptual data.

Impact: Concept grounding is a principal step towards truly autonomous robots. The end goal is a robot that would be able to construct conceptual descriptions of the world completely autonomously, based only on criteria for success and failure in an ensemble of related tasks such as navigation, recognition, manipulation, etc. A realistic intermediate goal is to develop algorithms for extraction of conceptual information that can be reused between similar tasks and decrease the sample complexity in life-long learning.

State of the Art: Concept grounding is a form of concept formation which deals with the lowest level in building concept hierarchies. One other instance of the concept formation problem, that of building abstract concepts on top of primitive ones in supervised learning mode, has been thoroughly researched and can be considered solved for most practical purposes. Research in symbolic learning methods, however, does not deal with the lowest conceptual level that maps sensory readings, which are continuous variables, to primitive concepts, which are discrete in nature. Such methods assume that the lowest level is provided by the designer of the system. Harnad has argued that primitive concepts cannot be constructed in the abstract and should be grounded in sensorimotor capacity. A recent attempt to implement Harnad's ideas on a real robot has been reported by MacDorman. Similarly, Drescher has applied the early ideas of Jean Piaget on child learning to a real robot.

On a purely sub-symbolic level, research in connectionist learning methods has tried to explain how internal representations develop in a self-organizing manner. However, there have been strong objections to the claim that truly new representations can develop under the supervised learning paradigm, on the ground that the training set already defines and contains discrete classes. On the other hand, unsupervised learning algorithms develop representations that are very little beyond trivial clustering. Clearly, neither paradigm is appropriate to the problem of concept grounding.

Approach: In our opinion, the most relevant learning paradigm for autonomous concept grounding is reinforcement learning. In this paradigm, no training examples are given to the learner, so the internal representations that develop are entirely new. On the other hand, in contrast to unsupervised learning, there is some performance feedback that could be used to construct non-trivial representations. An alternative method for developing representations useful in a particular learning task is to start out with a set of candidate concepts, and assess by means of some algorithm their individual usefulness as state descriptors.

Several initial experiments have been performed to study the measures that can be used to identify a good representation [1]. The task was to learn a successful policy in a simulated sequential decision process whose state is not completely observable. The experimental environment consisted of a single track along which a robot can move back and forth, jumping from one cell to the next (Fig.1(a)). One of the cells is a source of objects, which should be carried to a destination cell. However, the robot cannot distinguish these two cells as special — the only sensory input it has is the state it is in, and it also receives intermittent

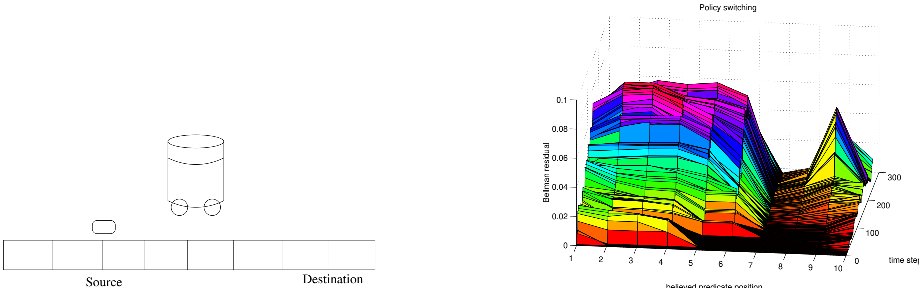

Figure 1: (a) A simulated robot moves on a track and receives positive reinforcement when an object is brought from the source cell to the destination one. (b) Moving averages of the Bellman residuals for ten predicates, plotted at 20-trial steps. The correct predicate quickly achieves the lowest value among all policies.

---

reinforcement if it has collected an object from the source cell and has carried it to the destination cell. Since the environment is not Markovian, a direct application of a reinforcement learning algorithm such as Q-learning fails.

Clearly, the policy of the robot depends on whether it is carrying an object or not, and this policy cannot be based on sensory input only. If the robot is carrying an object, it should head for the destination, otherwise, it should head for the source cell. One appropriate state representation is the number of the cell the robot is in, augmented with a predicate which is true if the robot is carrying an object. Since the robot does not know where the source is, it cannot define the predicate based on that location. Instead, it can define a set of predicates that are based on sensory input only, and find which of these predicates is useful in the learning task at hand.

The set of candidate predicates that were given to the robot in one experiment (Fig.1(b)) corresponded closely to the cells in the track — there was a single predicate for each cell. That predicate was true when the robot had visited this state. Clearly, the one predicate that is most useful is the one corresponding to the source cell, and the objective of the robot is to discover it. To this end, it must have a measure for estimating the usefulness of all predicates in turn.

One successful criterion for selection of predicates is the moving average of the Bellman residual for the policy defined on a particular state representation. If the state representation is good, a learning algorithm such as Q-learning quickly converges to the correct value function. However, if the state representation is poor, the learned value function does not predict well the value of the next state, and the learner is constantly "surprised" by new reinforcement. This reflects in large Bellman residuals (Fig.1(b)). In addition, learning multiple policies at the same time is a solution to the exploitation/exploration dilemma - what is exploitation for the most successful policy so far is exploration for all the rest.

Future Work: Future work includes using temporal probabilistic networks as a learning paradigm and/or designing a language for candidate predicates.

Selected References:

[1] Daniel Nikovski. Temporal predicate induction for model selection in reinforcement learning. In *Stochastic Model Building Workshop*, Duke University, October 1997.