Cognitive Robots Learning Failure Contexts Through Experimentation

(Extended Abstract)

Sertac Karapinar
Artificial Intelligence and Robotics Laboratory
Istanbul Technical University
Istanbul, Turkey
karapinarse@itu.edu.tr

Sanem Sariel
Artificial Intelligence and Robotics Laboratory
Istanbul Technical University
Istanbul, Turkey
sariel@itu.edu.tr

ABSTRACT

We propose a lifelong experimental learning method for cognitive robots to build and transfer knowledge among appropriate contexts. Experience gained through learning is used as a guide to future decisions of robots for both efficiency and robustness.

Categories and Subject Descriptors

I.2.9 [**Artificial Intelligence**]: Robotics—*Autonomous vehicles*; I.2.6 [**Learning**]: Concept learning

Keywords

cognitive robots; planning and plan execution; execution monitoring; real-world experimentation; online learning

1. INTRODUCTION

Learning is an important key to robustness and efficiency in future decisions of a robot. Our main objective in this work is to investigate methods that ensure robustness in task execution by a cognitive robot through its real-world experimentation and learning. Our work builds on our previous work and extends it with a lifelong experience-based learning approach [2]. We use Inductive Logic Programming (ILP) as the learning method to frame hypotheses mapping from execution contexts to action outcomes. Execution contexts include symbolic predicates on actions, objects in interest and their relations which are expressed in first-order logic sentences for derivation of hypotheses. Derived hypotheses are then used to devise heuristics for guidance in planning.

Our contribution lies in the way we use learning and incorporate contextual information into a knowledge-based learning approach for hypothesis derivation. Hypotheses can be expressed in first-order logic sentences, and the learning process can use background knowledge to generalize. We show that without a model-based failure isolation, robustness can be ensured by experience-based learning and learning-guided planning to present alternative solutions to failed cases.

Appears in: *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015), Bordini, Elkind, Weiss, Yolum (eds.), May 4–8, 2015, Istanbul, Turkey.*

2. LEARNING FAILURE CONTEXTS

The lifelong experience-based learning problem that we investigate asks for cognitive robots to learn hypotheses ($\mathcal{H}$) that map action execution contexts ($\mathcal{C}$) to failure cases through observations. This is needed to model such cases and prevent from potential damages to the environment or the objects in interest. An incremental and continual approach is needed to model the observations on the outcomes of execution. Furthermore, the learning algorithm should be able to represent hypotheses by logic sentences since the knowledge base of a cognitive robot is represented symbolically to reason and plan for achieving its goals. Since a robot has partial observability, the facts that it can extract from the world do not always cover all the predicates that describe the world but only the observable ones.

The main problem that we address differs from that of existing systems [1, 3, 4, 6] in the way experience is built and learning is accomplished by a cognitive robot. The robot is assumed to have an initial domain knowledge, and we let the robot learn failure contexts from experimentation in the real world. The robot learns incrementally and builds its experience runtime. Then, it updates its planning domain based on its experience on failure contexts to reduce further failures in execution. The robot comes up with correct conclusions by using abstraction and reasoning. These facts necessitate knowledge-based learning methods to be integrated with planning systems.

We use Inductive Logic Programming as the continual *learning* process. Inductive Logic Programming framework for real-world experimentation incorporates the following inputs for each operator o_i corresponding to a real-world action a_i:

1. Real-world observation history Obs is built by each observation at time step t, $obs_t \in Obs$ $(0 \le t \le T)$ maps an observed execution context $c_t \in \mathcal{C}$ that includes symbolic predicates to the observed outcome $(success, failure)$ of the execution of a_i. Eventually, Obs represents a set of positive examples (p) corresponding to successful outcomes and negative examples (n) to failed outcomes of execution of a_i.

2. Background theory $\mathcal{B}$ representing prior knowledge.

Either the known or the observed features of objects to be manipulated, their relations and the observable features of the world state are considered in representations of contexts.

Given these inputs, the learner finds a hypothesis space $\mathcal{H}$, that explains the positive examples and rejects the negative examples about a_i and generalizes $\mathcal{C}$ using $\mathcal{B}$. Whenever needed, new hypotheses may be framed based on new observations or existing ones are abandoned, and the knowledge base (KB) is updated according to this inference result.

We use and enhance the Progol algorithm [5], and investigate its strengths and weaknesses for framing hypotheses on failure cases. Progol is based on inverse resolution applying an inverted deductive proof process. Since the set of classifications are fixed ($success, failure$) in our case, the process can trace back to frame a general hypothesis. We enhanced the algorithm to interpret ambiguous observations by computing probabilities of hypotheses (PROGOL$-P$).

2.1 Hypothesis Framing for Failure Cases

To illustrate how hypotheses are framed for failure cases, a ground robot scenario can be given as follows. Assume that the following observations are taken by the robot:

$obs_1 : category(box) \wedge shape(prism) \wedge color(green) \wedge material(paper) \wedge size(small) \wedge success(pickUp)$

$obs_2 : category(box) \wedge shape(prism) \wedge color(black) \wedge material(paper) \wedge size(large) \wedge success(pickUp)$

$obs_3 : category(box) \wedge shape(cylinder) \wedge color(red) \wedge material(paper) \wedge size(large) \wedge failure(pickUp)$

Each observation obs_t corresponds to an instance of an execution of action $pickUp$. A context c_t, in this example represented by the attributes of the object to be picked up is mapped to the outcome of the observed action ($success$/ $failure$). When the ILP learning is applied on this set, the following hypotheses are derived:

$shape(prism) \rightarrow success(pickUp)$

$shape(cylinder) \rightarrow failure(pickUp)$

After framing these hypotheses, suppose that the robot gets a new observation:

$obs_4 : category(box) \wedge shape(prism) \wedge color(green) \wedge material(plastic) \wedge size(large) \wedge failure(pickUp)$

After getting this observation, hypotheses in the KB are generalized as follows:

$shape(prism) \wedge material(paper) \rightarrow success(pickUp)$

$material(plastic) \vee shape(cylinder) \rightarrow failure(pickUp)$

Note that the probabilities for these hypotheses are computed as 1 since there are no conflicting observations. In this particular example, only the attributes of a single object are taken into account. However, in a more complex scenario involving many objects, the attributes of all objects and their relations are in consideration. As with other supervised learning methods, a sufficient number of observations are needed to come up with correct conclusions. The main superiority of the ILP learner to the conventional learning methods is its knowledge-based representation.

2.2 Learning-based Guidance

Lifelong learning procedure continually frames new hypotheses during execution. Hypotheses derived from observations are then used to provide feedback to improve a robot's performance on its future tasks which enables real-world experimentation. In this work, we particularly use precondition update guidance method for failed operators. A learned failure context is added as a constraint in the corresponding precondition. The other proposed guidance methods are out of the scope of this paper [7].

3. RESULTS AND CONCLUSIONS

We investigate how our autonomous robot autonomously builds experience in a task of cleaning the environment by manipulation of objects randomly scattered around. Whenever a failure is detected, the corresponding observation along with its related context is encoded in the KB. All processes run online on the robot without any supervision.

The overall success rate of the robot in action $pickUp$ is 97.08% in 100 $pickUp$ trials. In our scenarios, we analyze hypothesis generation performance of our system if external failures are introduced by human intervention. We have seen that when the robot is enforced to fail executing $pickUp$ action on a specific object type, the system can generate hypotheses to explain failure cases. In this particular experiment, the other learning algorithms (Naive Bayes Classifier, Bayes Networks, Support Vector Machines and ID3 Decision Tree Learning) are also successful in framing correct hypotheses. However, these algorithms perform poorly where the use of background knowledge is benefical. In our second experiment, we have analyzed this issue. In this case, the robot is enforced to fail when the objects are located in a bounded spatial region. Without background knowledge on locations, the learning algorithms cannot correctly explain the underlying failure situation. However, PROGOL$-P$ algorithm, outperforming the others, can incorporate background knowledge and can abstract the failure context with 97.06% accuracy.

Acknowledgments

This research is funded by a grant from the Scientific and Technological Research Council of Turkey (TUBITAK), Grant No. 111E-286.

REFERENCES

[1] K. Z. Haigh and M. M. Veloso. Learning situation-dependent costs: Improving planning from probabilistic robot execution. *Robotics and Autonomous Systems*, 29:145–174, 1999.

[2] S. Karapinar, S. Sariel-Talay, P. Yildiz, and M. Ersen. Learning guided planning for robust task execution in cognitive robotics. In *The AAAI-13 Workshop on Intelligent Robotic Systems*, pages 26–31, Bellevue, USA, July 2013.

[3] T. Lang and M. Toussaint. Planning with noisy probabilistic relational rules. *J. Artif. Intell. Res. (JAIR)*, 39:1–49, 2010.

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

[5] S. Muggleton. Inverse Entailment and Progol. *New Generation Computing, Special issue on Inductive Logic Programming*, 13(3-4):245–286, 1995.

[6] H. Pasula, L. S. Zettlemoyer, and L. P. Kaelbling. Learning symbolic models of stochastic domains. *Journal of Artificial Intelligence Research (JAIR)*, 29, 2007.

[7] P. Yildiz, S. Karapinar, and S. Sariel-Talay. Learning guided symbolic planning for cognitive robots. In *IEEE International Conference on Robotics and Automation (ICRA), Autonomous Learning Workshop*, Karlsruhe, Germany, May 2013.