Self-assessment of Proficiency of Intelligent Systems: Challenges and Opportunities

Alvika Gautam(✉), Jacob W. Crandall, and Michael A. Goodrich

Computer Science Department, Brigham Young University, Provo, UT 84604, USA
alvikag@byu.edu, {crandall,mike}@cs.byu.edu

Abstract. Autonomous systems, although capable of performing complicated tasks much faster than humans, are brittle due to uncertainties encountered in most real-time applications. People supervising these systems often rely on information relayed by the system to make any decisions, which places a burden on the system to self-assess its proficiency and communicate the relevant information.

Proficiency self-assessment benefits from an understanding of how well the models and decision mechanisms used by robot align with the world and a problem holder's goals. This paper makes three contributions: (1) Identifying the importance of goal, system, and environment for proficiency assessment; (2) Completing the phrase "proficient ‹preposition›" using an understanding of proficiency span; and (3) Proposing the proficiency dependency graph to represent causal relationships that contribute to failures, which highlights how one can reason about their own proficiency given alterations in goal, system, and environment.

Keywords: Proficiency · Self-assessment · Goal(s) · System · Environment · Intelligent agents

1 What is Proficiency Assessment?

Proficiency assessment can be operationally defined as *the ability to detect or predict success (or failure) towards a goal in a particular environment given an agent's sensors, computational reasoning resources, and effectors.* Ideally, proficiency assessment approaches need to work a priori, in situ, and a posteriori. Different levels of self-assessment include (a) detecting proficiency (success or failure), (b) assigning a proficiency score (quantification of likelihood of success or degree of failure), (c) providing explanations (reasoning behind the outcome i.e., success or failure), and (d) predicting proficiency, which will allow intelligent systems to make informed decisions about their ability to accomplish tasks based on previous outcomes and their explanations. Communities in both computer science and robotics have addressed questions related to proficiency self-assessment. These include introspection [1–3], monitoring system performance [4, 5], robustness to uncertainties [6, 7], to name a few. However, much work is needed to address more in-depth levels of proficiency self-assessment adequately.

This paper presents initial ideas and frameworks for reasoning about proficiency self-assessment. Proficiency must be defined relative to the following: (i) Goals: desired

M. Zallio (Ed.): AHFE 2020, AISC 1210, pp. 108–113, 2021.
https://doi.org/10.1007/978-3-030-51758-8_15

outcomes of the task that the intelligent agent should reach in a finite amount of time or during long-duration missions; (ii) Environments: world settings in which the intelligent agent operates; and (iii) System Configurations: sensors, actuators, and computational resources that are available to the agent. This paper identifies a causal relationship between these three categories and the mechanisms and models used by AI algorithms to make decisions. The paper is limited to *in situ* aspects (during runtime) of simple detection of proficiency, i.e., whether an agent's actions and the resulting states take it closer towards the desired outcome of the task.

2 Span of Proficiency Self-assessment

Because proficiency depends on the environment, system, and goal, we propose that proficiency assessments should explicitly assert the span for which proficiency applies. We propose that a way to think about span is to consider the following statement:

An agent is proficient ‹*preposition*›, where the ‹*preposition*›, is used to indicate the span or scope of the assessment.

Table 1 proposes a relationship between proficiency span and a representative preposition. The entries of the table represent the instances of variation of these properties (Goal, System, and Environment) over an enumerated set. A "1" in the table indicates that proficiency is defined with respect to a specific goal, specific system, or specific environment, whereas ">1" indicates multiple goals, systems, or environments.

Table 1. Span of proficiency self-assessment

	Goal	System	Environment
At	1	1	1
Within	1	>1	>1
Across	>1	1	1
Over	>1	>1	>1

We recognize that the selection of the prepositions is somewhat arbitrary, but we believe the prepositions provide a common vocabulary for proficiency researchers – we have been part of too many conversations where people were talking past each other because of confusion about the way they were using the term *proficient.*

Proficient At: An agent is said to be "proficient at" a goal if it competently satisfies the goal for a given system configuration and environment condition. Proficiency "at" this level is the minimum requirement for an agent. The "1"s in each cell of this row in Table 1 highlights that being proficient "at" something appertains only to a single environment, system, and environment.

Proficient Within: Most systems are subject to uncertain and dynamic environmental conditions or disturbances during the task. This is aggravated by uncertainties associated

with the system itself, e.g., noisy sensors, failing effectors, time bounds on computation processes. For a given goal, an agent is said to be "proficient within" a range of system configurations and environment conditions. This is represented by the ">1" entries in the "Within" row of the table.

For example, consider a physical robot system that can knowingly fail in three different ways. Hence, for the agent to adequately assess its proficiency towards a goal in the presence of system anomalies, 3! system configurations should be tested. Similarly, an enumerated set of expected variations or changes to the environment is represented by the ">1" entries in the Environment column.

Proficient Across: A system might be comprised of several subsystems working in tandem, but each for their own individual goals. Or a single system might be capable of pursuing multiple goals. "Proficient across" indicates that proficiency needs to be considered with respect to multiple goals, which is represented by the ">1" entry in the "Across" row of the table.

Proficient Over: The "Over" row of Table 1 summarizes the span across which assessment of proficiency should ideally hold. That is, an intelligent agent should be able to assess its ability to competently satisfy multiple goals over a range system configurations, worlds and environment settings.

3 Proficiency Dependency Graph

In our approach towards self-assessment of proficiency, we adopt a proficiency dependency graph (PDG). The graph has six vertices: V = (Outcome, Mechanism, Model, World/Environment, System/Physical robot, Goal) as shown in Fig. 1. A directed edge connects vertex *A* with vertex *B* if " B depends on A."

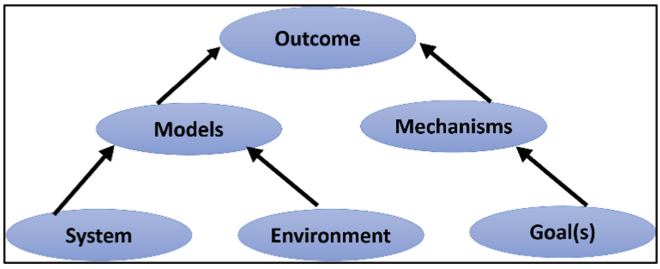

Fig. 1. Proficiency dependency graph for self-assessment

Outcome: The *outcome* vertex represents the evaluation of proficiency, including (a) a binary assertion about whether the agent is proficient or not, (b) an indicator score about the probability that agent can competently solve a problem, or (c) a degree or quality level at which the task can be performed.

Goals, Systems, Environments: The three vertices in the bottom layer represent conditions about the world, system/physical robot, and goal(s) to be solved.

Mechanisms and Models: These two vertices refer to aspects of the algorithms used by the agent to try to achieve its goals given its system and environment. For an intelligent agent, we identify two aspects of algorithms critical to proficiency assessment: *Mechanisms* and *Models*. These are represented as the two vertices in the second level of the graph.

Both models and mechanisms represent assertions about how the agent solves a particular problem. We operationally define a mechanism as a goal specification or set of incentives that explicitly or implicitly encode a goal. Based on our definition, several problem-solving techniques from literature can be thought of as "mechanisms". For example, classifiers often use classification accuracy or precision/recall, MDPs (Markov Decision Process) use rewards, optimal control systems often use objective functions, while planners often use temporal logics to specify a goal.

Models are assumptions made by the agent about (i) how the environment works, (ii) the effect of agent's actions on the environment, and (iii) the relationship between sensors and the environment. It should be noted that the model assumptions are not always explicit, as in reflex agents, and may implicitly be a part of the corresponding mechanism. Examples of explicit mechanisms include: (1) a classifier's use of trees or networks to represent a process by which a decision is made for a given set of training inputs and outputs; (2) an MDP's use of state transition matrices that map present-state-action to next states in a way that represents environment and system; (3) an optimal control system's use of physics-based models for a physical plant, actuators, and sensors; and (4) a planner's use of state transition systems to represent how agent actions affect the world.

Alignment: The mechanism for solving a problem should *align* with the goal, and assumptions about the corresponding model should *align* with the realities of the environment and the system. The connections between the bottom vertices (world, system, goal) and the middle vertices (model and mechanism) represent the two alignment problems, which we refer to as "goal alignment" and "model alignment," respectively.

Consider an MDP problem that can be solved using value iteration, given a set of rewards and a transition probability matrix. Assuming that model alignment holds, an optimal policy may not accomplish a problem holder's goal if the rewards used by the solver do not align with the goal; there is goal-mechanism misalignment. On the other hand, assuming goal alignment, the given transition probability matrix may not correctly model the uncertainties in the system and/or environment, leading to a mismatch between observed distribution of (state, action) pairs and model assumptions. Referring back to Table 1, an agent may not be proficient at a goal (given a system and environment) if either goal or model misalignment exists.

Using Proficiency Alignment: An agent's proficiency at a goal or goals within a number of system configurations and environment is contingent on the conditions of goal and model alignment being met. Misalignments can be useful in generating explanations for proficiency failures because they provide cause and effect reasoning.

For example, suppose that an agent is using an optimal policy derived from an MDP in a long-duration mission. The agent is tracking the empirical history of present-state, action, next state triples. The agent compares the empirical distribution to the transition probability and finds a significant discrepancy. The agent concludes that there is a model misalignment, and reports that its models of the world are likely not sufficiently accurate to perform the mission.

Continuing the example, suppose that the empirical distribution matches the transition probability. Suppose further that the agent has a logic-based recognizer that it uses to determine when tasks are completed. If the recognizer persistently reports failures, the agent can conclude that there is a misalignment between its goals and the rewards used to create the optimal policy. The agent reports that it needs to observe a human performing the task for a while, and then uses inverse reinforcement learning to derive a new set of rewards.

4 Summary and Future Work

This paper presents preliminary ideas for self-assessment of proficiency for an intelligent system. We identified mechanisms and models as the mid-level representative entities, to self-evaluate (i) an intelligent agent's ability of competently satisfying a goal(s) with variations in system configuration and environment settings, and (ii) evaluate cause and effect on (i).

As a part of the future work, we are working on an exhaustive literature review organized according to the definitions of span and the characterization of alignment. This review should provide insight and a narrative into where the state-of-the-art fits within the proficiency dependency graph, what the intended span of the assessment is, and whether the assessments are intended for use a priori, in situ, or a posteriori. Leveraging the findings of the literature review, we plan to formalize our ideas further to develop a generalized framework for proficiency self-assessment of intelligent systems.

Acknowledgments. This work was supported in part by the U.S. Office of Naval Research under Grants N00014-18-1-2503 and N00014-16-1-302. All opinions, findings, conclusions, and recommendations expressed in this paper are those of the author and do not necessarily reflect the views of the Office of Naval Research.

References

1. Daftry, S., Zeng, S., Bagnell, J.A., Hebert, M.: Introspective perception: learning to predict failures in vision systems. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1743–1750. IEEE (2016)
2. Zhang, P., Wang, J., Farhadi, A., Hebert, M., Parikh, D.: Predicting failures of vision systems. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3566–3573 (2014)
3. Wu, H., Lin, H., Guan, Y., Harada, K., Rojas, J.: Robot introspection with bayesian nonparametric vector autoregressive hidden Markov models. In: 2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids), pp. 882–888. IEEE (2017)

4. Kaipa, K.N., Kankanhalli-Nagendra, A.S., Gupta, S.K.: Toward estimating task execution confidence for robotic bin-picking applications. In: 2015 AAAI Fall Symposium Series (2015)
5. Israelsen, B., Ahmed, N., Frew, E., Lawrence, D., Argrow, B.: Machine self-confidence in autonomous systems via meta-analysis of decision processes. In: International Conference on Applied Human Factors and Ergonomics, pp. 213–223. Springer, Cham (2019)
6. Lakhal, N.M.B., Adouane, L., Nasri, O., Slama, J.B.H.: Interval-based solutions for reliable and safe navigation of intelligent autonomous vehicles. In: 2019 12th International Workshop on Robot Motion and Control (RoMoCo), pp. 124–130. IEEE (2019)
7. Havens, A., Jiang, Z., Sarkar, S.: Online robust policy learning in the presence of unknown adversaries. In: Advances in Neural Information Processing Systems, pp. 9916–9926 (2018)