Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21)
Doctoral Consortium Track

Data Efficient Algorithms and Interpretability Requirements for Personalized Assessment of Taskable AI Systems

Pulkit Verma*

School of Computing, Informatics, and Decision Systems Engineering
Arizona State University, Tempe, AZ 85281, USA
verma.pulkit@asu.edu

Abstract

The vast diversity of internal designs of taskable black-box AI systems and their nuanced zones of safe functionality make it difficult for a layperson to use them without unintended side effects. The focus of my dissertation is to develop algorithms and requirements of interpretability that would enable a user to assess and understand the limits of an AI system's safe operability. We develop an assessment module that lets an AI system execute high-level instruction sequences in simulators and answer the user queries about its execution of sequences of actions. Our results show that such a primitive query-response capability is sufficient to efficiently derive a user-interpretable model of the system in stationary, fully observable, and deterministic settings.

1 Introduction

The growing deployment of AI systems presents a pervasive problem of ensuring the safety and reliability of these systems. The problem is exacerbated because most of these AI systems are neither designed by their users nor are their users skilled enough to understand their internal working, i.e., the AI system is a black-box for them. Additionally, we now have systems that can adapt to user preferences, thereby invalidating any design stage knowledge of their internal model.

My dissertation work aims to create general algorithms and methods for interpretability which when used with a black-box AI system, can help in estimating its internal model by interrogating it. Consider a situation where a logistics company buys new delivery robots. The person managing these robots is unsure whether the robots correctly understand a task, or if they can even execute it safely. If the manager was dealing with a delivery person, it might ask them questions such as "do you think it would be alright to bring refrigerated items in a regular bag?" If the answer is "yes", it might be a cause for concern. Answers to such questions can help the manager develop an understanding of the robot's frame of knowledge, or "model" while placing a minimal introspective requirement on the robot.

*Advisor: **Siddharth Srivastava**, Arizona State University

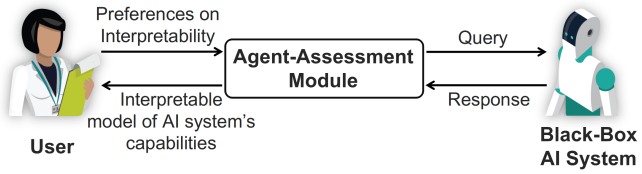

Figure 1: The agent-assessment module uses its user's preferred vocabulary, queries the AI system, and delivers a user-interpretable causal model of the AI system's capabilities.

2 Focus of My Dissertation

Most simulator-based and analytical-model-based AI systems can easily answer the kind of questions discussed above. However, generating the right set of questions to ask the AI system to efficiently estimate the system's model is a challenging problem. The focus of this new direction of research is on solving this problem. This proposed method, when used with any AI system, would also help make them compliant with Level II assistive AI – systems that make it easy for users to learn how to use them safely [Srivastava, 2021].

In my dissertation, I plan to develop an *agent-assessment module* (AAM), shown in Fig. 1, which can derive the model of a black-box AI system in terms of user-interpretable vocabulary. AAM takes as input the user's preferred vocabulary and the list of the AI system's actions. It then queries the AI system and receives its responses. At the end of the querying process, AAM returns a user-interpretable model of the AI system. An advantage of this approach is that the AI system need not know the user vocabulary or the modeling language.

2.1 Generating Interrogation Policies

I aim to create an interrogation policy that will generate the queries for the AI system, and use the AI system's answers to estimate its model in the user-interpretable vocabulary. I plan to generate these queries by reducing the query generation to a planning problem and then use an interrogation algorithm to iteratively generate new queries actively, based on responses to previous queries.

2.2 Inferring the Model

Given the predicates and actions, there is an exponential number of PDDL models possible. To avoid this combinatorial explosion, I plan to use a top-down process that eliminates

large classes of models, inconsistent with the AI system, by computing queries that discriminate between pairs of *abstract models*. When an abstract model's answer to a query differs from that of the AI system, we can eliminate the entire set of possible models that are refinements of this abstract model.

We plan to start research on this front with simplistic queries in deterministic fully observable environments and expand the scope to more general settings. In the future, this mechanism can be extended to more general forms of queries. Similar to active learning, information theoretic metrics can also be utilized to ascertain which queries will be better at any given time in the querying process.

2.3 Related Work

Several action model learning approaches [Yang *et al.*, 2007; Arora *et al.*, 2018] have focused on learning the AI system's model using passively observed data. These approaches do not feature any interventions, hence are susceptible to learning buggy models. Unlike these approaches, our approach queries the AI system and is guaranteed to converge to the true model while presenting a running estimate of the accuracy of the derived model; hence, it can be used in settings where the AI system's model changes due to learning or a software update. In such a scenario, our algorithm can restart to query the system.

2.4 Preliminary Results

We developed a preliminary version of AAM [Verma *et al.*, 2021] that efficiently derives a user-interpretable model of the system in stationary, fully observable, and deterministic settings. In the context of this initial work, user-interpretable means STRIPS-like [Fikes and Nilsson, 1971] models because such models can be easily translated into interpretable descriptions, and they also allow interventions and assessment of causality. In the future, I plan to learn more general and more expressive models of the AI system.

Also, in this version, we used *plan outcome queries* which are parameterized by an initial state and a plan; and ask the AI system, the length of the longest prefix of the plan that it can execute successfully when starting in the given initial state, as well as the final state that this execution leads to. E.g., "Given that the truck $t1$ and package $p1$ are at location $l1$, what would happen if you executed the plan $\langle load\_truck(p1, t1, l1), drive(t1, l1, l2), unload\_truck(p1, t1, l2)\rangle$?".

We compared AAM with the closest related work FAMA [Aineto *et al.*, 2019] in terms of; the accuracy of the learned model, the number of queries asked, and the time taken to generate those queries. Fig. 2 summarizes our findings for systems initialized with IPC domains. AAM takes lesser time per query and shows better convergence to the correct model. FAMA sometimes reaches nearly accurate models faster, but its accuracy continues to oscillate, making it difficult to ascertain when the learning process should be stopped. This is because the solution to FAMA's internal planning problem introduces spurious palm tuples in its model if the input traces do not capture the complete domain dynamics. Also, in domains with negative preconditions like Termes, FAMA was unable to learn the correct model.

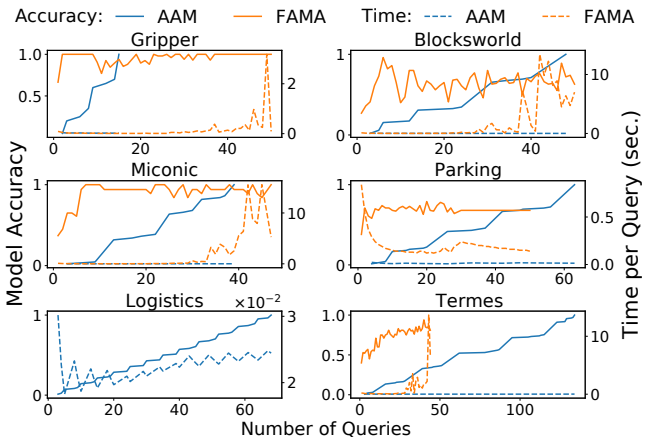

Figure 2: Performance comparison of AAM and FAMA in terms of model accuracy and time taken per query.

We also showed that AAM can be used with simulator-based systems that do not know about predicates and report states as images. To test this, we wrote classifiers to detect predicates from images of simulator-states in the PDDLGym [Silver and Chitnis, 2020] framework. This framework provides ground-truth PDDL models, thereby simplifying the estimation of accuracy. We initialized the AI system with one of the two PDDLGym environments, Sokoban and Doors. AAM inferred the correct model in both cases, and the average number of queries (over 5 runs) used to predict the correct model for Sokoban and Doors were 201 and 252, respectively. Finally, we showed that the models learned by AAM are causal models [Verma and Srivastava, 2021], unlike the ones learned by approaches that only use observational data.

References

[Aineto *et al.*, 2019] Diego Aineto, Sergio Jiménez Celorrio, and Eva Onaindia. Learning Action Models With Minimal Observability. *Artif. Intell.*, 275:104–137, 2019.

[Arora *et al.*, 2018] Ankuj Arora, Humbert Fiorino, Damien Pellier, Marc Métivier, and Sylvie Pesty. A Review of Learning Planning Action Models. *Knowl. Eng. Rev.*, 33:E20, 2018.

[Fikes and Nilsson, 1971] Richard E Fikes and Nils J Nilsson. STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving. *Artif. Intell.*, 2(3-4):189–208, 1971.

[Silver and Chitnis, 2020] Tom Silver and Rohan Chitnis. PDDLGym: Gym Environments from PDDL Problems. In *ICAPS 2020 PRL Workshop*, 2020.

[Srivastava, 2021] Siddharth Srivastava. Unifying Principles and Metrics for Safe and Assistive AI. In *Proc. AAAI*, 2021.

[Verma and Srivastava, 2021] Pulkit Verma and Siddharth Srivastava. Learning Causal Models of Autonomous Agents using Interventions. In *IJCAI 2021 GenPlan Workshop*, 2021.

[Verma *et al.*, 2021] Pulkit Verma, Shashank Rao Marpally, and Siddharth Srivastava. Asking the Right Questions: Learning Interpretable Action Models Through Query Answering. In *Proc. AAAI*, 2021.

[Yang *et al.*, 2007] Qiang Yang, Kangheng Wu, and Yunfei Jiang. Learning Action Models from Plan Examples Using Weighted MAX-SAT. *Artif. Intell.*, 171(2-3):107–143, 2007.