Interactive Learning from Activity Description

@ICML 2021

Khanh Nguyen
kxnguyen@umd.edu
(University of Maryland, College Park)

Dipendra Misra, Robert Schapire, Miro Dudík
(Microsoft Research, New York)

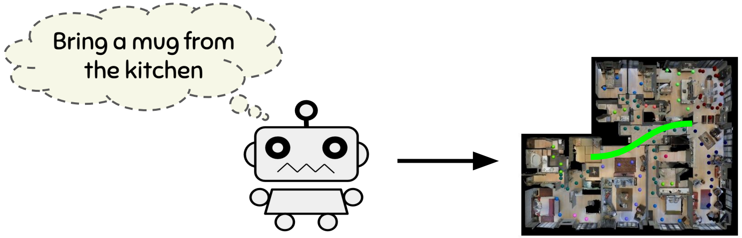
Patrick Shafto
(Rutgers University, Newark)

https://github.com/khanhptnk/iliad

Microsoft Research

Overall

- *Request fulfilling*: executing tasks specified in language in situated environments

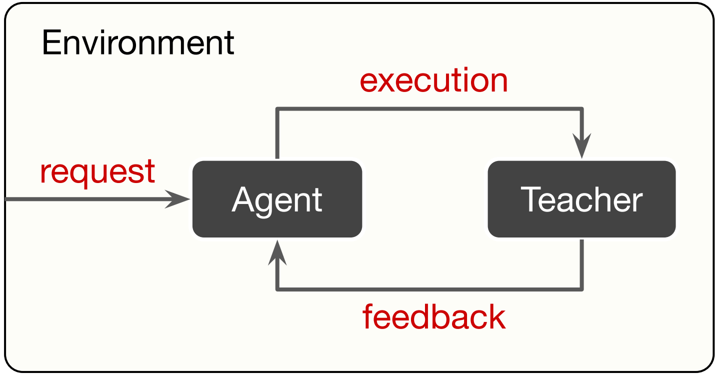


- Agents are typically trained using non-verbal learning frameworks
  - Imitation Learning (IL): learning from *demonstrations*
  - Reinforcement Learning (RL): learning from *rewards*

⇒ Highlight the drawbacks of these frameworks

⇒ Propose a verbal learning framework that offers complementary advantages

Interactive Learning



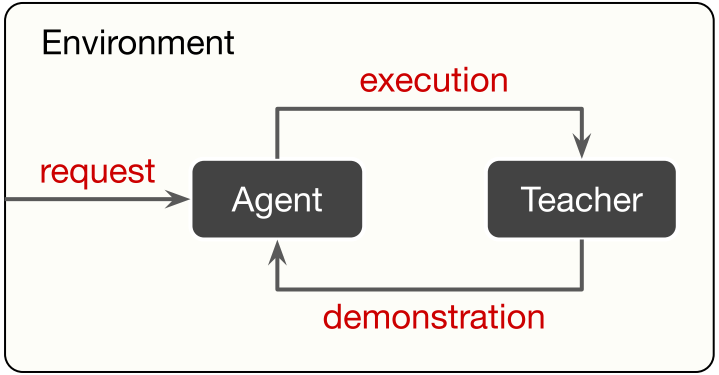
An **agent** interacts with a **teacher** in environments to learn to fulfill requests

Learning proceeds in **episodes**, where

1) Agent receives a request
2) Agent generates an execution
3) Teacher examines the execution and sends a feedback to the agent
4) Agent uses the feedback to update its model

Different types of feedback → Different families of learning algorithms!

Imitation Learning



Feedback = **Demonstration** that illustrates how the request should be executed

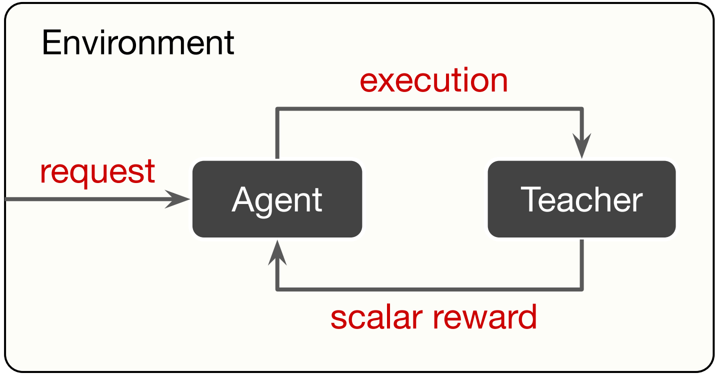
Demonstration = {(state$_i$, action$_i$)}

where action$_i$ ∈ Agent action space

⇒ Teacher must be familiar with the agent's action space and knows how to control the agent to fulfill the request

⇒ Non-experts may need to spend substantial effort in order to acquire such knowledge

Reinforcement Learning

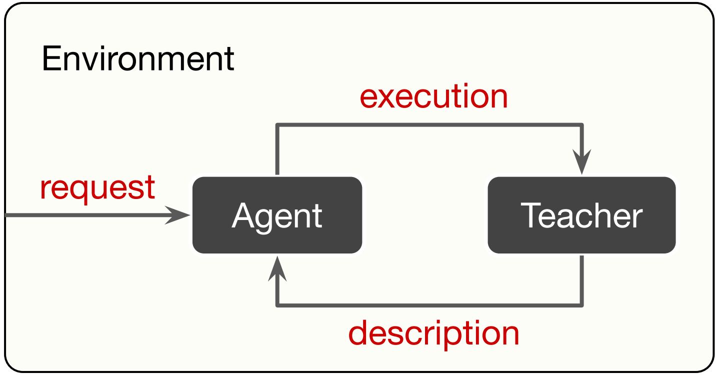


Feedback = **Scalar reward** that evaluates performance of the execution

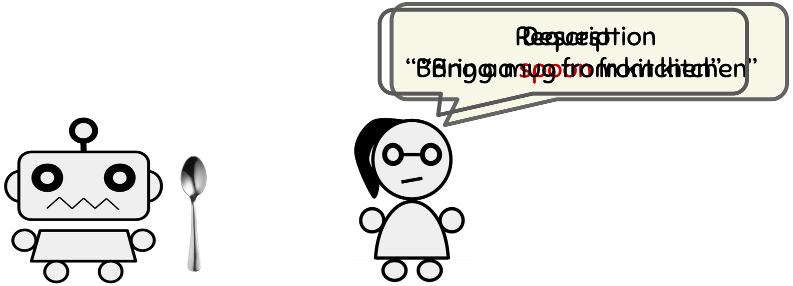
⇒ Scalar reward (a floating-point number) conveys weak learning signals

⇒ Sample inefficiency! (agent may take a lot of interactions with the teacher to achieve high performance)

ILIAD: Interactive Learning from Activity Description



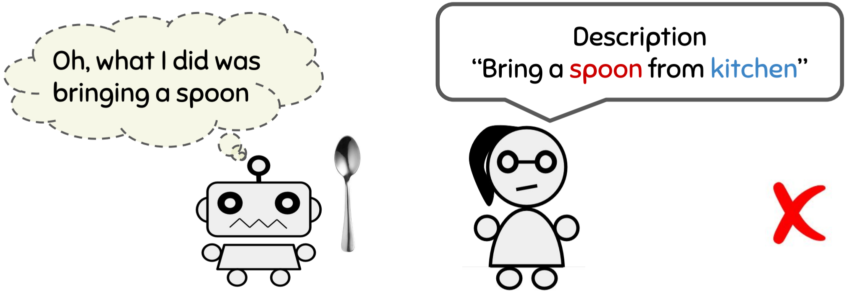
Feedback = **Description** that verbally describes the agent's execution



Language as feedback offers complementary advantages:

- Compared with *demonstration*: less direct, but allows teacher to teach with their natural language;
- Compared with *reward*: requires interpretation, but provides richer learning signals (sample efficiency↑).

How does description feedback enable request fulfilling?



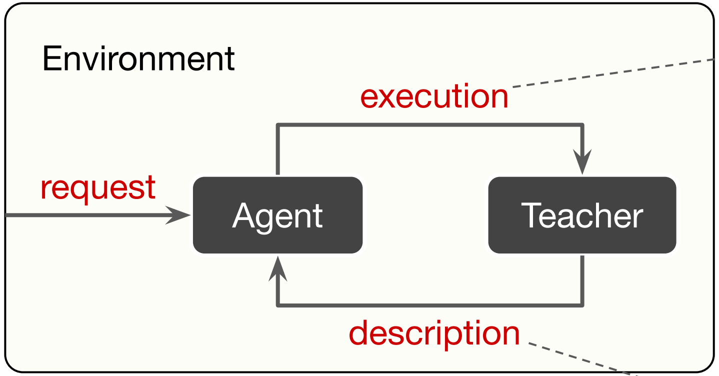
Agent better fulfills requests that are the same as descriptions

- By grounding descriptions to executions, the agent learns to execute descriptions
- Assume that descriptions are given in the same language as that of requests(*)

If agent can leverage compositionality of language, it can also better fulfill similar requests

(*) i.e. they are drawn from the same distribution, formally defined in the paper.

ADEL: a practical implementation of ILIAD

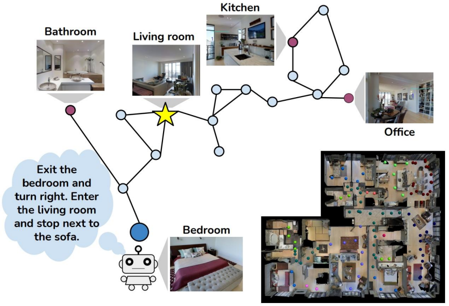


Execution generation via semi-supervised exploration scheme:

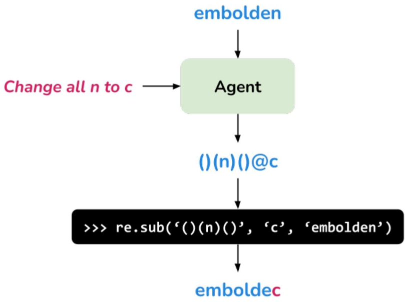
$$e \sim \underbrace{\lambda \cdot P(e)}_{\text{(request-agnostic)}} + \underbrace{(1 - \lambda) \cdot P(e \mid d^*)}_{\text{(request-guided)}}$$

Language grounding by learning to generate execution conditioned on description, i.e. estimating $P(e \mid d)$

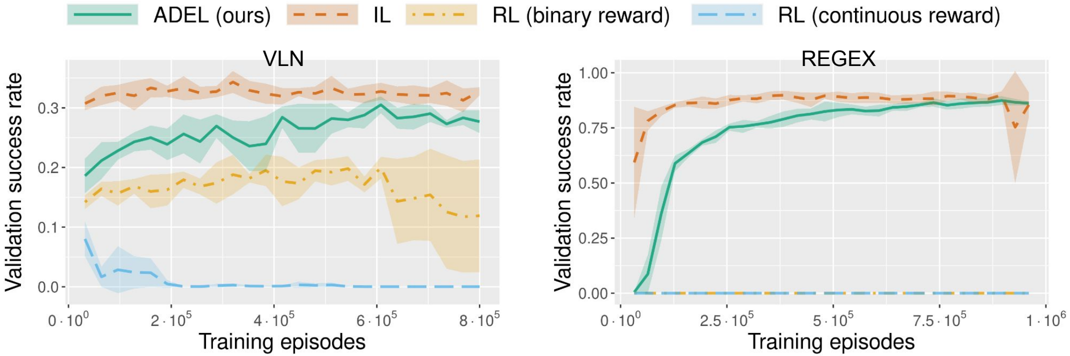
Experimental Results



Vision-language Navigation (VLN)



Word modification via regular expressions (REGEX)



- Achieves competitive success rates with IL (but needs more interactions to achieve);
- Significantly outperforms RL (in terms of success rate and sample efficiency).

Summary

Framework	Teacher has to be familiar with agent's control interface	Agent sample efficiency
Imitation Learning	Yes	Highest
Reinforcement Learning	No	Lowest
ILIAD	No	Medium

- Enhancing the communication protocol is fundamental in improving interactive learning algorithms
- ILIAD/ADEL allows teaching with a natural language and can be more sample-efficient than RL!
  ⇒ suitable for human-agent interaction

Interactive Learning from Activity Description