Paper Critique

Anirud N, CE21B014

Course: DA7400, Fall 2024, IITM
Paper: Hierarchical Imitation and Reinforcement Learning
Date: 28 August 2024

Make sure your critique addresses the following points:

1. The problem the paper is trying to address
2. Key contributions of the paper
3. Proposed algorithm/framework
4. How the proposed algorithm addressed the described problem

Note: Be concise with your explanations. Unnecessary verbosity will be penalized. Please don't exceed 2 pages.

---

1 Problem Statement

This paper aims to address the issues faced by traditional RL approaches - learning good agents from reward signal alone is difficult during long planning horizons. One way to deal with this is Imitation Learning. But IL requires good amount of expert demonstrations. This paper tries to effectively use the EXPERT feedback, to train the agents. For long horizons, using a HRL - options framework seems to be very efficient.

2 Key Contributions

The paper explores the use of Hierarchies in IL. It introduces hierarchical guidance - high-level experts are used to guide low-level learners. This enables us to focus only on the relevant regions- not concentrating much on the already mastered state-action space. The paper introduces HIL (hierarchical Imitation Learning), where only high-level expert feedback can be used to train the low-level actions. It also introduces HIDAgger - lazy feedback from the expert just to identify if it is a failure or success - in both High level and low level.

3 Proposed Algorithm/Framework

3.1 Types of Supervision

- **HierDemo**(s): *hierarchical demonstration*
- **Label**$_{\text{HI}}(\tau_{\text{HI}})$: *HI-level labeling*
- **Label**$_{\text{LO}}(\tau; g)$: *LO-level labeling*
- **Inspect**$_{\text{LO}}(\tau; g)$: *LO-level inspection*
- **Label**$_{\text{FULL}}(\tau_{\text{FULL}})$: *full labeling*
- **Inspect**$_{\text{FULL}}(\tau_{\text{FULL}})$: *full inspection*

The main idea is to give different costs to different queries and cost is modelled based on complexity

3.2 Hierarchial Behaviour cloning

Algorithm 1 Hierarchical Behavioral Cloning (h-BC)

Initialize data buffers $\mathcal{D}_{\text{HI}} \leftarrow \emptyset$ and $\mathcal{D}_g \leftarrow \emptyset$, $g \in \mathcal{G}$
for $t = 1, \ldots, T$ **do**
  Get a new environment instance with start state s
  $\sigma^\star \leftarrow \text{HierDemo}(s)$
  for all $(s_h^\star, g_h^\star, \tau_h^\star) \in \sigma^\star$ **do**
    Append $\mathcal{D}_{g_h^\star} \leftarrow \mathcal{D}_{g_h^\star} \cup \tau_h^\star$
    Append $\mathcal{D}_{\text{HI}} \leftarrow \mathcal{D}_{\text{HI}} \cup \{(s_h^\star, g_h^\star)\}$
  end for
end for
Train subpolicies $\pi_g \leftarrow \text{Train}(\pi_g, \mathcal{D}_g)$ for all g
Train meta-controller $\mu \leftarrow \text{Train}(\mu, \mathcal{D}_{\text{HI}})$

3.3 Hierarchial Guided DAgger

(Since the algorithm is big , I could not attach it here) The basic idea is to decrease the cost of feedback. So, the paper uses the $INSPECT_{LO}$ - whether the subtasks are completed and $LABEL_{HI}$ - to check whether we are in the relevant part of the state space. Using the high-level policy, we first find the subgoal to be implemented, then the policy for the subgoal is implemented till the subgoal reaches the terminal state, after which another sub-goal is started. The expert feedback is used only if the agent fails. If the entire task fails, $LABEL_{HI}$ labels the correct subgoals. In this way we can narrow down LO learning to relevant part of the state space.

3.4 Hierarchially Guided IL/RL

The higher level space assigns sub-tasks and associated pseudo rewards.For each selected sub-goal, a sub-policy at the LO level executes primitive actions until a termination condition is met, based on pseudo-rewards. AFter this next subgoal is chosen and so on. the expert inspects the overall execution of the learner, and if it is not successful, the expert provides HI-level labels, which are accumulated for training the meta-controller. Pseudo rewards:

$$\begin{cases} 1 & \text{if } (s; g) \\ -1 & \text{if } \neg(s; g) \text{ and } (s; g) \\ - & \text{otherwise,} \end{cases}$$

where > 0 is a small penalty to encourage short trajectories.

4 Advantages and Conclusions of the Algorithm

The hg-DAgger/Q outperforms flat and traditional hierarchical approaches in tasks like maze navigation and Montezuma's Revenge. It achieves higher success rates with lower expert labeling costs, particularly at the low level. It is more efficient in long planning horizons and sparse rewards, showing the advantages of hierarchical guidance over h-DQN. This paper increased the speed and reduced the cost of learning with the help of expert feedback. This paper did not address "learning the termination predicate, while learning to act from reinforcement feedback".