Reinforcement Learning & Imitation Learning: an overview

Deep Learning Seminar
17.05.2022

Eugene Bykovets, D-INFK
Problem class

The problem is to train an intelligent agent to achieve a particular goal in a simulated environment or the real world. How can we do that?
Outline

- **Reinforcement learning**
  - Concept
  - Application domains
  - Notable problems
  - Reward misspecification

- **Imitation learning**
  - Behavioural cloning
  - Direct policy learning
  - Adversarial imitation learning

- **Inverse reinforcement learning**
  - Preference reward learning
  - Adversarial inverse reinforcement learning
Outline

- **Reinforcement learning**
  - Concept
  - Application domains
  - Notable problems
  - Reward misspecification

- **Imitation learning**
  - Behavioural cloning
  - Direct policy learning
  - Adversarial imitation learning

- **Inverse reinforcement learning**
  - Preference reward learning
  - Adversarial inverse reinforcement learning
Reinforcement Learning: Concept

1. We model environment as a Markov Decision Process
   \[ MDP = (S, A, P, R, \gamma, P_0), \]
   where:
   - \( S \) is state space
   - \( A \) is action space
   - \( P(s'|s, a) \) is transition probability
   - \( R(s', a, s) \) is reward signal
   - \( \gamma \) is discount factor
   - \( P_0 \) is initial state distribution

2. Agent uses reward signal \( R(s', a, s) \) from the environment as a guidance to achieve goal

3. Goal is to train the agent behavior (aka policy) \( \pi(a|s) \) which maximizes expected discounted reward:
   \[
   E\left[ \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t, s_{t+1}) \right]
   \]
Reinforcement Learning: Simulated domains
Researchers have long sought a source of clean, limitless energy.

One contender is nuclear fusion, which is the process of smashing and fusing hydrogen, releasing huge amounts of energy.

One way scientists have recreated these extreme conditions is by using a tokamak, a doughnut-shaped vacuum surrounded by magnetic coils, that is used to contain a plasma of hydrogen in extremely high temperature.
Reinforcement Learning: Real-world domains (plasma control)

- Plasma in these machines are inherently unstable, as a result **sustaining the process is a complex challenge**

- **Control system** needs to coordinate the tokamak's many magnetic coils and **adjust the voltage on them thousands of times per second** to ensure the plasma never touches the walls of the vessel, which would result in heat loss and possibly damage

- Swiss Plasma Center at EPFL and DeepMind managed to **train controller with reinforcement learning** in simulated environment and apply in real tokamak
Reinforcement Learning: Problems

• Markovian nature of data is a challenge for optimizers
  o Most of the results are obtained given i.i.d assumption (SGD gradient are biased [1]) all the SGD-like optimizers still affected

• Training domain shift
  o RL can be approximated supervised as i.i.d problem with continuous training domain shift by using experience replay buffer

• Sample efficiency
  o In the worst case, we see all states only once. There is a remedy: the usage large experience reply buffer, but it is not a panacea

• Safety problem
  o Training in real-world can be damaging for agents and the environment

• Reward specification
  o Could we really design a reward that corresponds to our intentions?
Reinforcement Learning: reward misspecification

- CoastRunners game
  - The goal is to gain as much score as possible by collecting items and win the race (as human understands)

- Reward is:
  - Sparse: not every step towards the finish is encouraged
  - Misspecified: Item gathering has superior encouragement

- The result is misbehavior!
Reinforcement Learning: reward misspecification

Maybe we should try to train an agent without a pre-designed reward signal?
Outline

• Reinforcement learning
  o Concept
  o Algorithms taxonomy
  o Application domains
  o Notable problems
  o Reward misspecification

• Imitation learning
  o Behavioral cloning
  o Direct policy learning
  o Adversarial imitation learning

• Inverse reinforcement learning
  o Preference reward learning
  o Adversarial inverse reinforcement learning

• Offline reinforcement learning
Behavioral cloning: Concept

1. Collect demonstrations $\tau^*$ trajectories from expert
2. Treat the expert demonstrations as i.i.d. state-action pairs: $(s_0^*, a_0^*), (s_1^*, a_1^*), ...$
3. Learn $\pi(a|s)$ policy using supervised learning by minimizing the loss function $L(a^*, \pi_\theta(s))$
Behavioral cloning: Problems

- **Markovian** data means the next state depends on the current one
- **Misbehavior** in the current state leads to the accumulation of the error in all the next steps
- **Misbehavior** is very likely in states which different from expert
- **We need an oracle!**
Direct policy learning: Concept

- **Improved concept** of behavioral cloning
- Assume the **presence** of an interactive expert-level demonstrator (**oracle**)
- The main idea is to get **more suboptimal trajectories** to **improve behavioral robustness** in states that are far from that contained in expert data
Direct policy learning: Data/Policy Aggregation

1.

- Initial predictor $\pi_0$
- For $m = 1$:
  - Collect trajectories $\tau^*$ by rolling out $\pi_{m-1}$
  - Estimate state distribution $P_m$ using $s \in \tau^*$
  - Collect interactive feedback $\{\pi^*(s) \mid s \in \tau^*\}$
  - Data Aggregation (e.g. Dagger)
    - Train $\pi_m$ on $P_1 \cup \cdots \cup P_m$
  - (Alternative is policy aggregation, e.g. SEARN)
    - Train $\pi'_m$ on $P_m$
    - $\pi_m = \beta \pi'_m + (1 - \beta) \pi_{m-1}$
Direct policy learning: Problem

- Improved concept of behavioral cloning
- **Assume the presence of an interactive expert-level demonstrator (oracle) expensive!**
- The main idea is to get more suboptimal trajectories to improve behavioral robustness in states that are far from that contained in expert data
Adversarial Imitation Learning: GAN
Adversarial Imitation Learning: GAIL

Policy NN → predicted action

Generated transition: \((s_t, a_t, s_{t+1})\) → Discriminator

Real transition: \((s_t, a_t, s_{t+1})\) → Discriminator

real or fake?
Adversarial Imitation Learning: Concept

- Collect demonstrations $\tau^*$ trajectories from experts;
- Build GAN like architecture where:
  - The generator is now off-the-shelf reinforcement learning (e.g., PPO) algorithm that tries to generate meaningful trajectories
  - Discriminator tries to differentiate real expert trajectories (collected at the beginning) from generated ones
  - In the adversarial two-player game, we try to achieve expert-level policy
Adversarial Imitation Learning: Problems

• Inherited GAN problems:
  • Difficult to optimize
  • Difficult to fine-tune
  • Mode collapse
Inverse Reinforcement Learning

What if we still want to **make underlying intentions** a little bit more **understandable**?
Outline

• Reinforcement learning
  o Concept
  o Algorithms taxonomy
  o Application domains
  o Notable problems
  o Reward misspecification

• Imitation learning
  o Behavioral cloning
  o Direct policy learning
  o Adversarial imitation learning

• Inverse reinforcement learning
  o Preference reward learning
  o Adversarial inverse reinforcement learning
Preference Reward Learning: concept

- Collect trajectories (expert-level included):
  - $\tau_1, \tau_2, \ldots, \tau_n$
- Ask expert to give global score (or rank the trajectories with full order):
  - $\tau_4 < \tau_{104} < \ldots < \tau_2$
- Train you reward function $r_\theta(s)$ in supervised manner with ranking loss-function:
  - $L(\theta) = -\sum_{\tau_l < \tau_j} \log \frac{\exp \sum_{s \in \tau_j} r_\theta(s)}{\exp \sum_{s \in \tau_l} r_\theta(s) + \exp \sum_{s \in \tau_j} r_\theta(s)}$
  - Do not care about reward for individual states,
  - Want that the predicated trajectories rewards align with ground truth rank
Preference Reward Learning: problems

• Ill-posedness nature of the problem (of IRL)
  o Infinitely many “optimal” reward functions w.r.t to a finite amount of expert trajectories
• Imprecise, works well only for “survival” environments
  o We care about staying alive in the environment longer and do not care about achieving the precise goals, as a result, we are fine with imprecise function
Adversarial Inverse Reinforcement Learning: GAIL (recap)

Policy NN → predicted action → Generated transition → Discriminator

Real transition → real action → Discriminator

Predicted transition → predicted action → Discriminator

real or fake?
Adversarial Inverse Reinforcement Learning

Policy NN

Replay buffer

Predicted action

Generated transition

Real transition

Discriminator

Reward NN

Shaping NN

predicted action

(s′, s′+1)

real action

(a′, s′+1)

(s′)

(a)

Real transition

Reward NN

Shaping NN

Discriminator

real or fake?
Adversarial Inverse Reinforcement Learning: problems

- Inherited GAN problems:
  - Difficult to optimize
  - Difficult to fine-tune
  - Mode collapse
  - Reward is not the last version of the network, it is curriculum set
<table>
<thead>
<tr>
<th>Field</th>
<th>Goal</th>
<th>Example</th>
<th>Advantages</th>
<th>Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reinforcement Learning</td>
<td>Optimize expected discounted reward</td>
<td>PPO, SAC, TRPO etc.</td>
<td>Can be straightforward for simple problems</td>
<td>• Sampe efficency</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Training domain shift</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Sim2Real</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Reward misspecification</td>
</tr>
<tr>
<td>Imitation Learning</td>
<td>Imitatie expert behavior</td>
<td>Beahvior cloning</td>
<td>Simple to use</td>
<td>Is not robust</td>
</tr>
<tr>
<td>Direct Policy Learning</td>
<td></td>
<td></td>
<td>Good performance</td>
<td>Expensive expert-level oracle needed</td>
</tr>
<tr>
<td>GAIL</td>
<td></td>
<td></td>
<td>Good performance</td>
<td>GAN-inhereted problems</td>
</tr>
<tr>
<td>Inverse Reinforcement Learning</td>
<td>Learn reward function</td>
<td>Preference Learning</td>
<td>Simple to use</td>
<td>Work only for “survival: problems</td>
</tr>
<tr>
<td>Airl</td>
<td></td>
<td></td>
<td></td>
<td>GAN-inhereted problems, need to build curriculum set</td>
</tr>
</tbody>
</table>
Thank you for your attention!

Questions?
Resources

• Fig. 2. Gridworld problem
  https://towardsdatascience.com/training-an-agent-to-beat-grid-world-fac8a48109a8
• Fig. 3. DeepMind AlphaGo.
  https://ichef.bbci.co.uk/news/976/cpsprodpb/11B23/production/_88738427_pic1go.jpg
• Fig.4. Reinforcement learning: Distributional Soft Actor-Critic (DSAC) in Gym Mujoco
• Fig. 7. Tokomak machine https://www.rts.ch/rts-
  online/medias/images/2021/thumbnail/fk3wxv-25152632.image?w=640&h=640
• Fig. 9. OpenAI. Reward misspecification https://openai.com/blog/faulty-reward-functions/
• Fig. 11.
  https://medium.com/startup-grind/even-smart-vcs-invest-in-cross-industry-clones-6a3c63f830e4
• Fig 12. https://smartlabai.medium.com/a-brief-overview-of-imitation-learning-8a8a75c44a9c
• Fig 14. Direct Policy Learning
  https://smartlabai.medium.com/a-brief-overview-of-imitation-learning-8a8a75c44a9c