Learning a Prior over Intent via Meta-Inverse Reinforcement Learning

Kelvin Xu, Ellis Ratner, Anca Dragan, Sergey Levine, Chelsea Finn
University of California, Berkeley
Motivation: a well specified reward function remains an important assumption for applying RL in practice
Motivation: a well specified reward function remains an important assumption for applying RL in practice
**Motivation:** a well specified reward function remains an important assumption for applying RL in practice
Motivation: a well specified reward function remains an important assumption for applying RL in practice.

- Often easier to provide expert data and learn a reward function using inverse RL.
Motivation: a well specified reward function remains an important assumption for applying RL in practice

- Often easier to provide expert data and learn a reward function using inverse RL
- Inverse RL frequently requires a lot of data to learn a generalizable reward
**Motivation:** a well specified reward function remains an important assumption for applying RL in practice

- Often easier to provide expert data and learn a reward function using **inverse RL**
- Inverse RL frequently **requires a lot of data to learn a generalizable reward**
  - This is due in part with the **fundamental ambiguity of reward learning**
Goal: how can agents infer rewards from one or a few demonstrations?
Goal: how can agents infer rewards from one or a few demonstrations?

- **Intuition:** demonstrations from previous tasks induce a prior over the space of possible future tasks.
Goal: how can agents infer rewards from one or a few demonstrations?

- **Intuition:** demonstrations from previous tasks induce a prior over the space of possible future tasks
Goal: how can agents infer rewards from one or a few demonstrations?

- **Intuition:** demonstrations from previous tasks induce a prior over the space of possible future tasks
Goal: how can agents infer rewards from one or a few demonstrations?

- Intuition: demonstrations from previous tasks induce a prior over the space of possible future tasks

Shared Context → Efficient adaptation
Meta-inverse reinforcement learning: using prior tasks information to accelerate inverse-RL
Meta-inverse reinforcement learning: using prior tasks information to accelerate inverse-RL

Learn a prior over intent through meta-learning over meta-training tasks: $T_{\text{train}}$
Meta-inverse reinforcement learning: using prior tasks information to accelerate inverse-RL

Meta-training time

Learn a prior over intent through meta-learning over meta-training tasks: $\mathcal{T}_{\text{train}}$

Evaluation time

New task $\mathcal{T}$

Rapid adaptation

$\phi = \theta - \alpha \nabla_{\theta} \mathcal{L}(\theta, \mathcal{T})$

Adapted reward $r_{\phi}$

MANDRIL
Meta Reward and Intention Learning
Our instantiation:
(background) Model-agnostic meta-learning
Our instantiation:
(background) Model-agnostic meta-learning

\[
\theta \leftarrow \theta - \alpha \nabla_{\theta} L_{\text{train}}(\theta)
\]

Fine-tuning
[test-time]

pretrained parameters

training data for new task
Our instantiation:
(background) Model-agnostic meta-learning

Fine-tuning
[test-time]

\[ \theta \leftarrow \theta - \alpha \nabla_{\theta} \mathcal{L}_{\text{train}}(\theta) \]

training data for new task

pretrained parameters

Our method

\[
\min_{\theta} \sum_{\text{task } i} \mathcal{L}_{\text{test}}^{i}(\theta - \alpha \nabla_{\theta} \mathcal{L}_{\text{train}}^{i}(\theta))
\]

MANDRIL
Meta Reward and Intention Learning
Our instantiation:
(background) Model-agnostic meta-learning

Fine-tuning
[test-time]
\[ \theta \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}_{\text{train}}(\theta) \]

Our method
\[
\min_\theta \sum_{\text{task } i} \mathcal{L}_{\text{test}}^i(\theta - \alpha \nabla_\theta \mathcal{L}_{\text{train}}^i(\theta))
\]

Intuition: Learning a prior over tasks, and at test time, inferring parameters under prior
(Grant et al. ICLR ’18)
Our approach: Meta reward and intention learning
Our approach: Meta reward and intention learning

Learn a prior over intent through meta-learning over meta-training tasks: $\mathcal{T}_{\text{train}}$

Our approach: embed deep MaxEnt IRL [1,2] into meta-learning
Our approach: Meta reward and intention learning

Our approach: embed deep MaxEnt IRL [1,2] into meta-learning

$$\min_{\theta} \sum_{\text{task } i} \mathcal{L}^i_{\text{test}}(\theta - \alpha \nabla_{\theta} \mathcal{L}^i_{\text{train}}(\theta))$$

Domain 1: SpriteWorld environment

Meta-Training

Evaluation time
Domain 1: SpriteWorld environment

- Each task is a specific landmark navigation task
Domain 1: SpriteWorld environment

Meta-Training

Evaluation time

- Each task is a specific landmark navigation task
- Each task exhibits the same terrain preferences
Domain 1: SpriteWorld environment

- Each task is a specific landmark navigation task
- Each task exhibits the same terrain preferences
- Evaluation time varies the position of landmark and uses unseen sprites
Domain 2: First person navigation (SUNCG)
Domain 2: First person navigation (SUNCG)

- Tasks require both learning navigation (NAV) and picking (PICK)
Domain 2: First person navigation (SUNCG)

- Tasks require both learning navigation (NAV) and picking (PICK)

Task illustration
Domain 2: First person navigation (SUNCG)

- Tasks require both learning navigation (NAV) and picking (PICK)
Domain 2: First person navigation (SUNCG)

- Tasks require both learning navigation (NAV) and picking (PICK)

Task illustration

- Tasks share a common theme but differ in visual layout and specific goal

Agent view
Results: With only a limited number of demonstrations, performance is significantly better.
Results: With only a limited number of demonstrations, performance is significantly better
**Results:** With only a limited number of demonstrations, performance is significantly better.
Results: With only a limited number of demonstrations, performance is significantly better.
Results: Optimizing initial weights consistently improves performance across tasks

- Success rate is significantly improved on both test and unseen house layouts especially on the harder PICK task

<table>
<thead>
<tr>
<th>METHOD</th>
<th>TEST</th>
<th>UNSEEN HOUSES</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PICK</td>
<td>NAV</td>
</tr>
<tr>
<td>Behavioral Cloning</td>
<td>0.4</td>
<td>8.2</td>
</tr>
<tr>
<td>MaxEnt IRL (avg gradient)</td>
<td>37.3</td>
<td>83.7</td>
</tr>
<tr>
<td>MaxEnt IRL (from scratch)</td>
<td>42.4</td>
<td>87.9</td>
</tr>
<tr>
<td>MANDRIL (ours)</td>
<td>52.3</td>
<td>90.7</td>
</tr>
<tr>
<td>MANDRIL (pre-adaptation)</td>
<td>6.0</td>
<td>35.3</td>
</tr>
</tbody>
</table>
Reward function can be adapted with a limited number of demonstrations
Reward function can be adapted with a limited number of demonstrations
Reward function can be adapted with a limited number of demonstrations
Reward function can be adapted with a limited number of demonstrations
Reward function can be adapted with a limited number of demonstrations
Thanks!
Tuesday, Poster #222