# Multi-Modal Imitation Learning from Unstructured Demonstrations using GANs

Disentangling Imitation Learning from the requirement of structured and isolated demonstrations

IN SUPREMÆ DIGNITATIS

1343

Federico Matteoni

# Introduction to the Problem

**Imitation Learning**: a system is presented with demonstrations of skills that it should imitate.

Usually single skill isolated demonstrations, not applicable to real-world scenarios.

Need of a method capable of:

- Map a video to state-action pairs
- Segment data into simple skills
- Imitate each of the segmented skills

This work focuses on the last two points: **learning a multi-modal stochastic policy** able to imitate different automatically segmented tasks from unlabeled unstructured demonstrations

# We have a MDP:

$$M = (\underset{\text{State space}}{S}, \underset{\text{Action space}}{A}, \underset{\substack{\text{State-Transition probability} \\ P: S\times A\times S\to\mathbb{R}^+}}{P}, \underset{\substack{\text{Reward function} \\ R: S\times A\to\mathbb{R}}}{R}, \underset{\substack{\text{Initial state distribution} \\ p_0: S\to\mathbb{R}^+}}{p_0}, \underset{\text{Reward discount factor}}{\gamma}, \underset{\text{Horizon}}{T})$$

And given a trajectory of states and actions $\tau = (s_0, a_0, \ldots, s_T, a_T)$ we have the trajectory reward given by the sum of the discounted rewards of each state-action pair.

The goal is to find a policy $\pi_\theta(a \mid s)$ that maximizes the expected discounted reward over trajectories induced by the policy. In imitation learning **the reward function is unknown** but we can infer it from the set of demonstrations.
Those demonstrations, in our case of more than one skill to learn, have trajectories originating from a possible mixture of expert policies $\pi_E = \pi_{E1}, \ldots, \pi_{Ek}$. We can estimate $R_E$ and optimize $\pi_\theta$ w.r.t. it recovering the expert policy (**Inverse Imitation Learning**)

We want to model different behaviors, so we aim for a policy with high entropy $H(\pi_\theta)$

The policy input and the trajectory are augmented with another parameter: *i* is a **latent intention variable**, sampled from a uniform or categorical distribution *p(i)*, with the goal of selecting a specific mode of the policy corresponding to one of the different skills.
E.g., a trajectory $\tau = (s_0, a_0, i_0, \ldots, s_T, a_T, i_T)$ from a policy $\pi_\theta(a \mid s, i)$

$$\pi(a \mid s) = \sum_i \pi(a \mid s, i) p(i) \qquad \pi(a \mid s, i) = p(i \mid s, a)\frac{\pi(a \mid s)}{p(i)} \qquad R(\tau_i) = \sum_{t=0}^{T} R(s_t, a_t, i_t)$$

Finding a high entropy policy $\pi_\theta$ able to optimize $R_E$ from an expert policy $\pi_E$ can be defined as sampling multiple demonstrations from the expert policy an optimizing a GAN:

- The policy $\pi_\theta(a \mid s, i)$ is the **generator**
- $D_w(s, a)$ is the **discriminator**, with the goal of distinguishing between samples from $\pi_\theta$ (labeled 0) and samples from $\pi_E$ (labeled 1)

The optimization goal is defined as

$$\max_{\theta} \min_{w} \mathbb{E}_{i\sim p(i),(s,a)\sim\pi_\theta}[\log(D_w(s,a))] + \mathbb{E}_{(s,a)\sim\pi_E}[1-\log(D_w(s,a))] + \lambda_H H(\pi_\theta(a\mid s)) - \lambda_I H(\pi_\theta(a\mid s,i))$$

This aims to find a saddle point (θ, w) where the discriminator $D_w$ has difficulty in distinguishing between samples from $\pi_\theta$ and from $\pi_E$: it maxes on θ in order to **find a policy that confuses the discriminator**, while also minimizing on w in order to **find a discriminator able to successfully distinguish between the two policies**. At the same time, we aim for a **policy with high entropy when presented without latent intention** (so averaged over all possible latent intentions), but able to collapse to a **low entropy policy when presented with a latent intention**.

So

- $\log(D_w(s, a))$ is minimized when the discriminator outputs 0, correctly identifying (s,a) from $\pi_\theta$
- $1-\log(D_w(s, a))$ is minimized when it outputs 1, correctly identifying (s,a) from $\pi_E$
- $H(\pi_\theta(s, a))$ is maximized when $\pi_\theta$ has high entropy averaged over all the latent intentions
- $-H(\pi_\theta(s, a, i))$ is maximized when $\pi_\theta$ has low entropy with intention *i*

The formula of entropy *H(P(x)) = E[-log(P(x))]* can be used to rework the objective function

$$H(\pi_\theta(a\mid s,i)) = -\mathbb{E}_{i\sim p(i),(s,a)\sim\pi_\theta}[\log(p(i\mid s,a))] + H(\pi_\theta(a\mid s)) + H(i)$$

$$\max_{\theta}\min_{w}\mathbb{E}_{i\sim p(i),(s,a)\sim\pi_\theta}[\log(D_w(s,a))]+\mathbb{E}_{(s,a)\sim\pi_E}[1-\log(D_w(s,a))]+(\lambda_H-\lambda_I)H(\pi_\theta(a\mid s))+\lambda_I\mathbb{E}_{i\sim p(i),(s,a)\sim\pi_\theta}[\log(p(i\mid s,a))]+\lambda_I H(i)$$

The key variation from the previous formulation is the second to last term

$$+\lambda_I\mathbb{E}_{i\sim p(i),(s,a)\sim\pi_\theta}[\log(p(i\mid s,a))]$$

With this term we are **rewarding the state-action pairs** generated by the policy that are **able to make the inference of the latent intention *i* easier**.

The last term doesn’t influence the optimization problem because *H(i)* is constant, due to *i* being sampled from a uniform or categorical distribution, so each carrying the same amount of information.

The reward function of the generator

$$\mathbb{E}_{i\sim p(i),(s,a)\sim\pi_\theta}[\log(D_w(s,a))]+\lambda_I\mathbb{E}_{i\sim p(i),(s,a)\sim\pi_\theta}[\log(p(i\mid s,a))]+\lambda_{H'}H(\pi_\theta(a\mid s))$$

Is maximized when

- $D_w$ classifies the sample from $\pi_\theta$ as if it was from $\pi_E$ (meaning that $D_w(s, a) = 1$)
- (s, a) is useful in inferring the latent intention *i*
- $\pi_\theta$ has high entropy averaged over all the latent intentions

# Key results

The main result of this model is its **ability to correctly segment the training demonstrations** and, when presented with categorical latent intentions, **correctly learning a multi-modal policy able to distinguish all the different skills**.

In one of the experimental setup, a robotic arm with two degrees of freedom is able to reach up to four different targets from a random initial position autonomously. When the latent intention is omitted, the network isn't able to discover different skills and struggles in distinguishing correctly all the tasks. With the latent intention, the network is able to reach the targets without problems. The **latent intention also has the effect of encouraging different behaviors for different intentions**, resulting in the arm reaching the target with different trajectories for each intention.

Another experimental setup includes an arm with a grappler at the end, with the double goal of reaching a target and then pushing it to another target. When presented with demonstrations of reaching the target and demonstrations of pushing a grappled target, the model is able to correctly segment the mix of expert policies into separate skills. The two subtasks start from different initial conditions, and the categorical intention is manually changed when the target is grasped. **Changing the intention** results in a pushing policy that brings the object to the designed target, **highlighting the capacity of the model in distinguishing two different subtasks**.

**So this model learns the concept of intention** and is able to perform different tasks based on a intention input, learning a multi-modal policy that is able to imitate all of the automatically segmented skills.

# Conclusions

The main novelty of this model is its **ability to include in a single policy knowledge about different skills and tasks**, which is surely a requirement for future general purpose systems and robots. It's also **able to distinguish different skills automatically**, based on unstructured and unlabeled examples, making the training of such systems much easier and with access to a lot more data possibly resulting in better trained models.

A weakness is highlighted in the paper: in a experimental setup, a humanoid robot is presented with three tasks. The humanoid robot is high-dimensional, with 16 degrees of freedom, so the tasks are much more difficult than the previous ones, and the policy is only able to mimic two of the three tasks with good result, only achieving suboptimal results in the third task. This is still better than a GAN without the latent intention information, which collapses to a unimodal policy that maps all the tasks to a single one out of the three.