
Diversity is All You Need
Learning Skills without a Reward Function
...
Krister Jaanhold, Lisa Yankovskaya

Contents

1. Introduction – no reward function, skills, challenges, contributions
2. Related work – reward function definitions, maximum entropy maximisation, maximising diversity
3. Diversity is all you need – how it works, main ideas, implementation details
4. Learned skills and their application – a demonstration
5. Conclusion

Introduction

Why do agents need to learn skills without reward?

- Environments with sparse rewards
- An unfamiliar environment
- Reduce the amount of supervision.

Introduction

Unsupervised learning of skills is a challenging task.

- Environment: reward functions are unknown
- Each skill individually is distinct
- The skills collectly explore large parts of the state space

Introduction

Skills

- Skills might be useless!
- Skills are not only distinguishable, but also are as diverse as possible
- Diverse skills are robust to perturbations and better exploring the environment.

Introduction

Contribution: the first part

- Propose a method for learning useful skills without any rewards
- Show good results for robotic tasks

Introduction

Contribution: the second part

- Show how these skills can quickly be adapted to solve a new task
- Show how these skills can be used for imitation learning.

Related work ...

... relies on a reward function to jointly learn a set of skills and a meta-policy for the purpose of solving a specific task. However:

- Designing the reward function is difficult
- Bad options aren't selected, thus they can't be improved

Maximum entropy has been achieved by:

- A probabilistic framework
- Soft Q learning
- Various other algorithms, e.g. **soft actor critic**

Related work

There have been various approaches on reward-functions

- Many tasks and reward functions (Hausman *et al.,* 2018)
- Single reward function (Florensa *et al.,* 2017)
- No reward – maximises mutual information (Gregor *et al.,* 2016)

Many studies have stated that complex behaviours can be learned by directly **maximising diversity** (e.g. learning many diverse policies)

Diversity is All You Need (DIAYN)

In order to acquire skills that are useful, we must train the skills so that they maximize coverage over the set of possible behaviors

Diversity is All You Need:

- Paradigm: unsupervised stage and supervised stage
- Unsupervised stage: the agent explores the environment, but does not receive any task reward
- Supervised stage: the agent receives the task reward, and its goal is to learn the task by maximizing the task reward.

Diversity is All You Need: how it works?

Three ideas behind this method:

- The skill dictates the states that the agent visits.
- To distinguish skills we use states not actions.
- The skills should be as diverse as possible.

Diversity is All You Need: how it works?

Three ideas behind this method:

- The skill dictates the states that the agent visits.
  - maximize the mutual information between skills and states, MI(s, z)

Diversity is All You Need: how it works?

Three ideas behind this method:

- The skill dictates the states that the agent visits.
  - maximize the mutual information between skills and states, MI(s, z)
- To distinguish skills we use states not actions.
  - minimize the mutual information between skills and actions given the state, MI(a, z | s)

Diversity is All You Need: how it works?

Three ideas behind this method:

- The skill dictates the states that the agent visits.
  - maximize the mutual information between skills and states, MI(s, z)
- To distinguish skills we use states not actions.
  - minimize the mutual information between skills and actions given the state, MI(a, z | s)
- The skills should be as diverse as possible.
  - maximize a mixture of policies (the collection of skills together with p(z))

Diversity is All You Need: how it works?

$$
\begin{aligned}
\mathcal{F}(\theta) &= MI(s, z) + \mathcal{H}[a \mid s] - MI(a, z \mid s) \\
&= (\mathcal{H}[z] - \mathcal{H}[z \mid s]) + \mathcal{H}[a \mid s] \\
&\quad - (\mathcal{H}[a \mid s] - \mathcal{H}[a \mid s, z]) \\
&= \mathcal{H}[z] - \mathcal{H}[z \mid s] + \mathcal{H}[a \mid s, z]
\end{aligned}
$$

Diversity is All You Need: how it works?

$$
\begin{aligned}
\mathcal{F}(\theta) &= \mathcal{H}[a \mid s, z] - \mathcal{H}[z \mid s] + \mathcal{H}[z] \\
&= \mathcal{H}[a \mid s, z] + \mathbb{E}[\log p(z \mid s)] - \mathbb{E}[\log p(z)] \\
&\geq \mathcal{H}[a \mid s, z] + \mathbb{E}[\log q_\phi(z \mid s) - \log p(z)] \triangleq \mathcal{G}(\theta, \phi)
\end{aligned}
$$

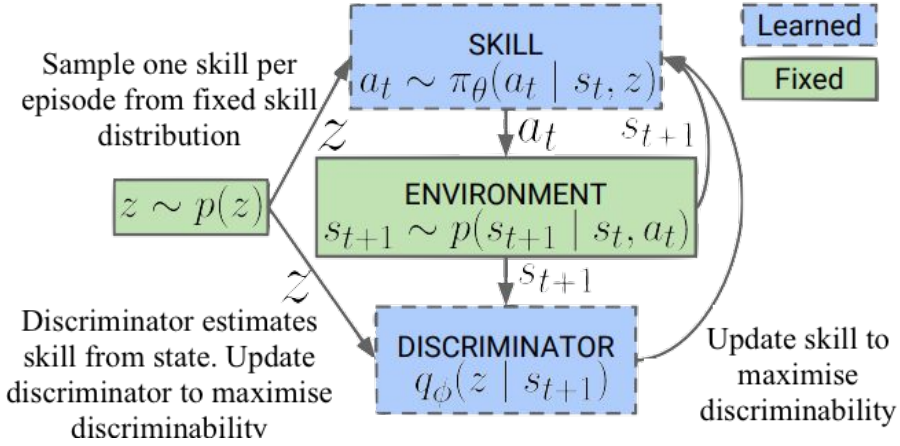
Diversity is All You Need

Implementation

- Uses **soft actor critic** to learn a policy
- Entropy regulariser is scaled by α
- Uses a **pseudo-reward** r_z to maximise the entropy

Diversity is All You Need

Conclusion

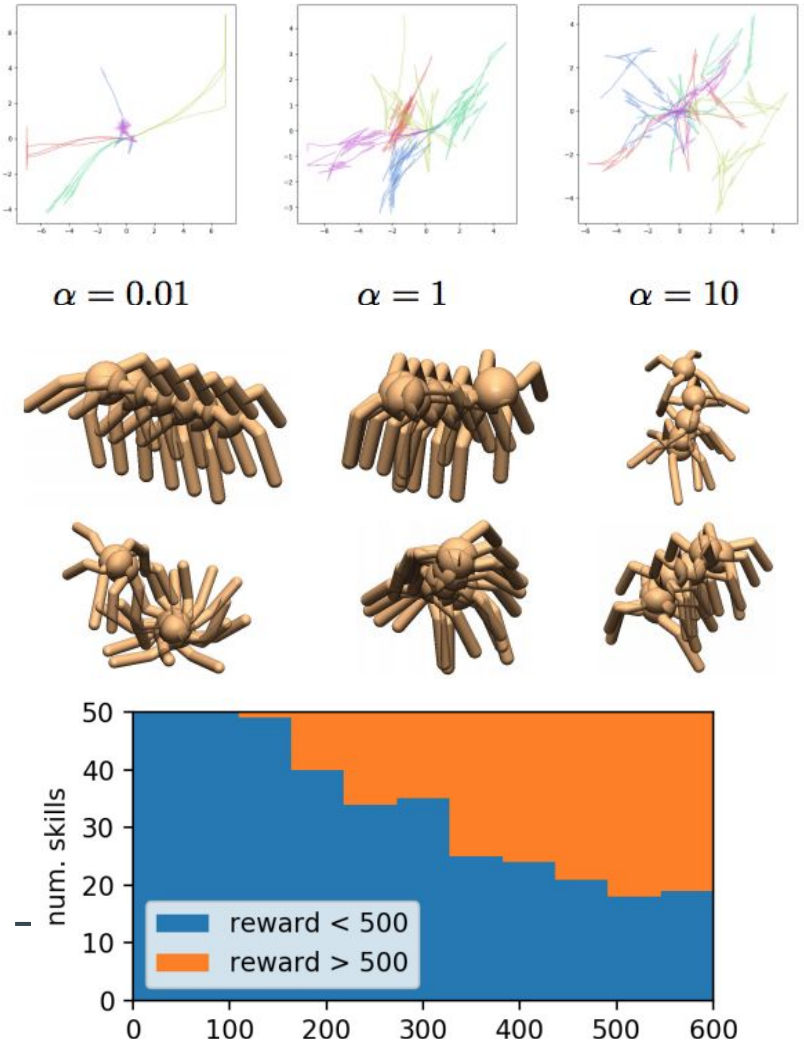


Algorithm 1 DIAYN

Input: skill distribution $p(z)$
repeat
  Sample skill $z \sim p(z)$ and initial state $s_0 \sim p_0(s)$.
  for $t = 1$ **to** *steps_per_episode* **do**
    Sample action $a_t \sim \pi_\theta(a_t \mid s_t, z)$ from skill.
    Step environment: $s_{t+1} \sim p(s_{t+1} \mid s_t, a_t)$.
    Compute $q_\phi(z \mid s_{t+1})$ with discriminator.
    Set skill reward $r_t = \log q_\phi(z \mid s_{t+1}) - \log p(z)$
    Update policy (θ) to maximize r_t with SAC.
    Update discriminator (ϕ) with SGD.
  end for
until

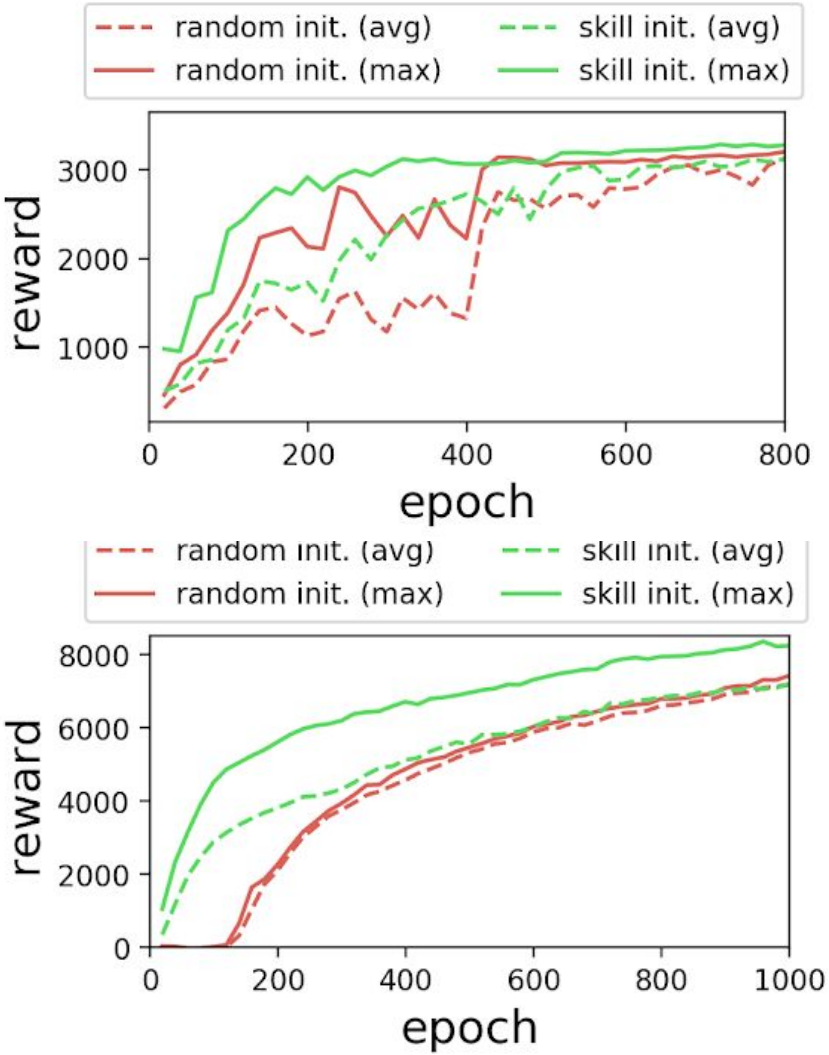
Learning skills

- Does entropy regularization lead to more diverse skills?
- Does the method learn diverse skills for simulated robots?
- Does the method learn useful skills?



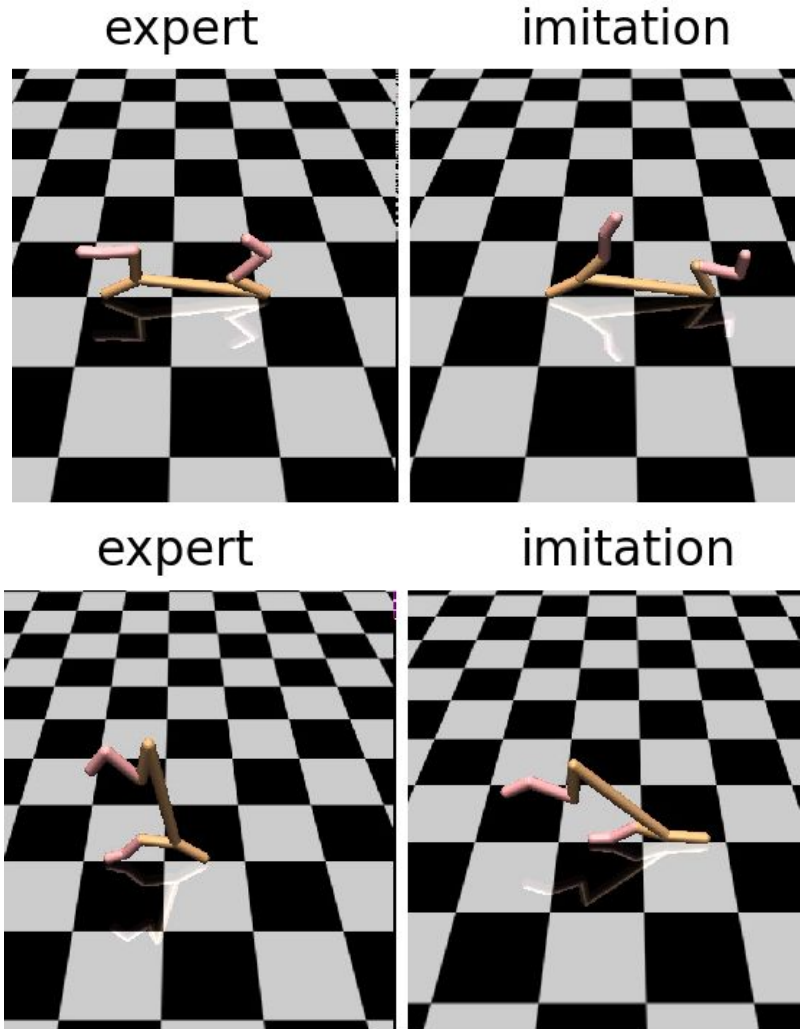
Harnessing skills:

Can we use learned skills to directly maximize the task reward?



Harnessing skills:

Can we apply learned skills to imitate an expert?



Learning diverse skills without a reward can achieve benchmark results on complex tasks!

Thank you for your attention!