# diversity is all your need

陈前辛小雨

同济大学

2025 年 6 月 8 日

## 传统强化学习的困境

- **强依赖外部奖励**
- 奖励设计成本高昂
- 限制自主性与通用性

→ 更像是被动的任务执行者

## 核心问题

我们能否让智能体在**没有任务指令**的情况下，像人类一样**自主学习**

## 解决方案：无监督强化学习

**核心思想：** 先学习，后做事 (Learn First, Act Later)

- 在无奖励环境中，自主发现一系列**通用、可复用的技能** (Skills)。

**关键方法：技能发现**

- **目标：** 多样性 (Diversity is All You Need)
- **工具：** 信息论 (Information Theory)
  - 最大化 **互信息** $I(S;Z) \rightarrow$ 技能可区分
  - 最大化 **熵** $H(A|S,Z) \rightarrow$ 探索最大化

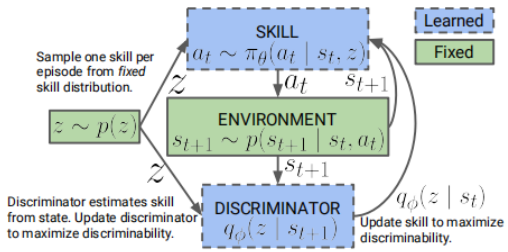
**研究聚焦** 提出一种新的无监督技能发现框架，旨在提升技能的**多样性、实用性与组合性**

# 应用价值

- **解决稀疏奖励环境下的探索问题**
- **作为分层强化学习的“积木块” (Primitives)**
  - **上层策略**：选择“技能”（如“向前走”、“开门”）。
  - **下层技能**：执行具体的“动作”序列。
  - 效果： 显著缩短长时序任务的有效决策步长。
- **降低对人类监督的依赖**
  - 适用于**交互成本低，但奖励评估成本高**（如需人类反馈）的场景。
  - 智能体先自由探索学会技能，再由人类少量标注哪个技能有用。
- **在未知环境中自主发现潜在任务**
  - 无监督地涌现出多样化技能，揭示了智能体在该环境中“能做什么”。

# 本文主要贡献

- **提出了一种无监督技能发现方法 DIAYN**
  - 无需任何外部奖励函数，即可学习到有意义的技能。
- **设计了一个简洁且有效的目标函数**
  - 基于信息论，能够驱动智能体涌现出如行走、跳跃等多样的技能。
  - 仅通过无监督预训练，即可直接解决一些基准测试任务。
- **探索了技能的下游应用方法**
  - 详细阐述了如何将学习到的技能应用于分层强化学习和模仿学习。
  - 展示了如何快速利用已发现的技能来解决新任务。




**Algorithm 1:** DIAYN

**while** *not converged* **do**
  Sample skill $z \sim p(z)$ and initial state $s_0 \sim p_0(s)$
  **for** $t \leftarrow 1$ **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 ($\theta$) to maximize $r_t$ with SAC.
    Update discriminator ($\phi$) with SGD.

# 核心思想与实现方法

## 核心思想 (Core Ideas)

1. **技能决定状态，使其可区分**
   不同的技能 $z$ 应该引导智能体访问不同的状态 $s$。

2. **鼓励探索，同时保持多样性**
   每个技能的策略应尽可能随机（高熵），同时为保持可区分性，必须探索远离其他技能的状态空间。

3. **通过状态而非动作来区分技能**
   我们只关心技能产生的**结果**（状态变化），而不关心其**过程**（具体动作），因为某些动作可能不改变环境。

## 实现方法 (Approach)

**目标：** 最大化一个信息论驱动的目标函数

- **最大化** 状态与技能的互信息 $I(S; Z)$
  技能对状态有控制力，可区分
- **最大化** 条件动作熵 $H(A|S)$ 策略行为尽可能随机，鼓励探索
- **最小化** 条件互信息 $I(A; Z|S)$ 确保技能由状态而非动作区分

## 最终目标函数 (Objective Function)

$$\mathcal{F}(\theta) \triangleq \underbrace{I(S; Z)}_{\text{可区分性}} + \underbrace{H[A \mid S]}_{\text{探索性}} - \underbrace{I(A; Z \mid S)}_{\text{解耦动作}}$$

$$= \underbrace{H[Z]}_{\text{技能多样}} - \underbrace{H[Z \mid S]}_{\text{状态推断技能}} + \underbrace{H[A \mid S, Z]}_{\text{技能内部探索}}$$

# 挑战与解决方案：变分推断 (Variational Inference)

## 挑战：后验概率难以计算

目标函数中的互信息项 $I(S;Z)$ 依赖于条件熵 $H(Z|S)$，而计算它需要知道后验概率 $p(z|s)$。

- $p(z|s) = \frac{p(s|z)p(z)}{p(s)}$
- 其中分母 $p(s) = \int p(s|z)p(z)dz$ 需要对所有技能和轨迹积分，计算上是**不可行 (intractable)**的。

## 解决方案：引入鉴别器

我们引入一个可学习的分布 $q_\phi(z|s)$ 来**近似**真实的后验 $p(z|s)$。

- $q_\phi(z|s)$ 是一个神经网络，通常称为**鉴别器 (Discriminator)** 或分类器。
- **任务：** 输入状态 $s$，输出它由各个技能 $z$ 产生的概率。

## 推导：变分下界 (Evidence Lower Bound, ELBO)

通过詹森不等式，我们可以推导出互信息 $I(S;Z)$ 的一个可计算的下界：

$$
\begin{aligned}
I(S;Z) &= \mathbb{E}_{p(s,z)}\left[\log \frac{p(z|s)}{p(z)}\right] \\
&= \mathbb{E}_{p(s,z)}\left[\log \frac{p(z|s)q_\phi(z|s)}{p(z)q_\phi(z|s)}\right] \\
&= \mathbb{E}_{p(s,z)}\left[\log \frac{q_\phi(z|s)}{p(z)}\right] + \mathbb{E}_{p(s)}\left[D_{KL}(p(z|s)||q_\phi(z|s))\right] \\
&\geq \mathbb{E}_{p(s,z)}\left[\log q_\phi(z|s)\right] + H(Z) \quad (\text{因为 } D_{KL} \geq 0)
\end{aligned}
$$

最终，我们将原始目标 $F(\theta)$ 替换为其下界 $G(\theta, \phi)$ 进行最大化。

# 实现框架：Soft Actor-Critic (SAC)

## 选择 Soft Actor-Critic (SAC) 作为基础

- **为何选择 SAC？** SAC 是一个基于**最大熵框架**的强大算法，其目标函数天生就包含策略熵 $H[a|s]$。
- **完美契合：** 这与 DIAYN 目标函数中的 $H[A|S,Z]$ 项（鼓励技能内部探索）完美对应。SAC 可以**自然地处理**这部分优化目标。
- **策略网络形式：** 学习到的策略以技能 $z$ 为条件，形式为：

$$\pi_\theta(a|s,z)$$

智能体根据当前状态 $s$ 和选定的技能 $z$ 来决定动作 $a$。

# 核心机制：内在伪奖励 (Pseudo-Reward)

创造一个“虚拟”的奖励信号来引导学习

## 问题：强化学习的“燃料”从何而来？

标准 RL 依赖于奖励。在无监督设定下，我们必须从 DIAYN 的目标函数中**“提取”**出一个内在奖励。

### 伪奖励函数定义

我们将目标函数的一部分定义为每一步的奖励 $r_z(s,a)$：

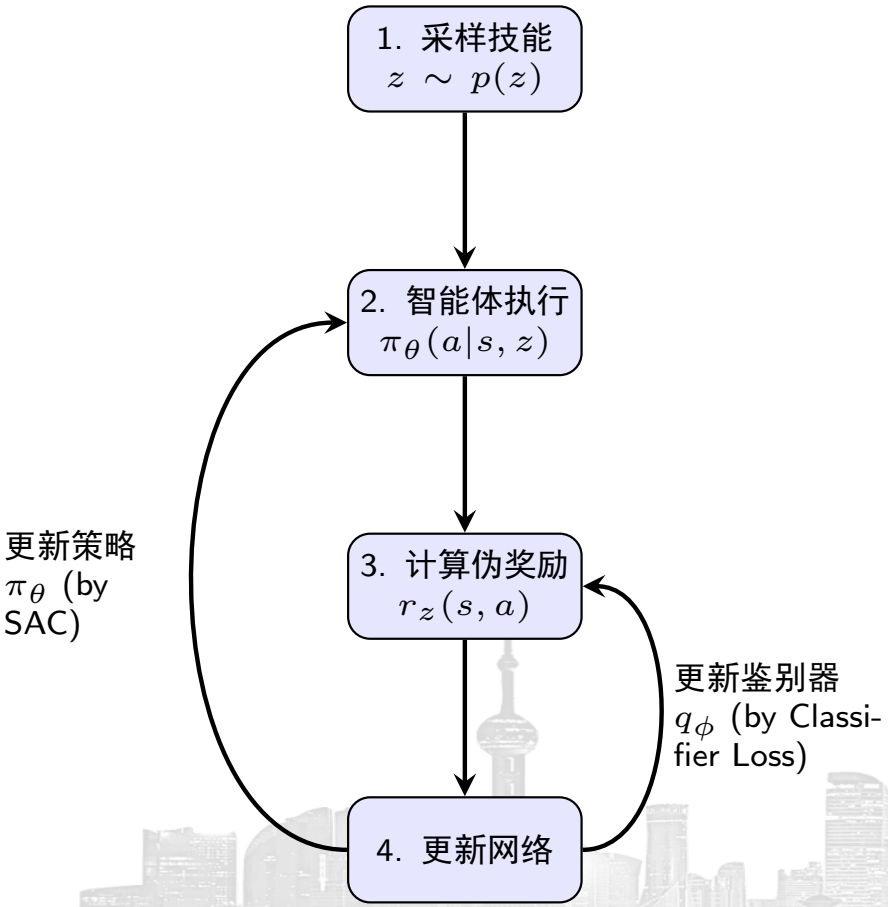
$$r_z(s,a) \triangleq \underbrace{\log q_\phi(z|s)}_{\text{主要驱动力}} - \underbrace{\log p(z)}_{\text{先验修正项}}$$

**直观解释：**

- $\log q_\phi(z|s)$: 鉴别器 $q_\phi$ 认为当前状态 $s$ 多大程度上是由技能 $z$ 产生的。如果置信度高，说明状态 $s$ **很有“辨识度”**，智能体就获得高奖励。

# 总结

## 整体学习流程




- 技能 ‘z‘ 从一个固定的分类分布 $p(z)$ 中采样（通常是均匀的，比如有10个技能，每个被选中的概率是1/10）。
- 在每一轮（episode）游戏开始时，智能体先“想好”一个技能 ‘z‘。
- 在整个这一轮中，智能体都使用这个固定的技能 ‘z‘ 来执行策略 $\pi_\theta(a|s,z)$。
- 智能体（Actor-Critic）的目标是最大化累积的伪奖励 $r_z(s,a)$。也就是说，它会努力去访问那些能让鉴别器 $q_\phi$ 轻松认出当前技能 ‘z‘ 的状态。
- 与此同时，鉴别器 $q_\phi$ 也在学习。它的目标是进行标准的监督学习分类任务：给定一大堆 ‘(状态, 技能)‘ 对，它要学会正确地从状态 ‘s‘ 预测出技能 ‘z‘。