WDS

DataScience@Web

课程学习

(CURRICULUM LEARNING)

汇报人：袁书伟

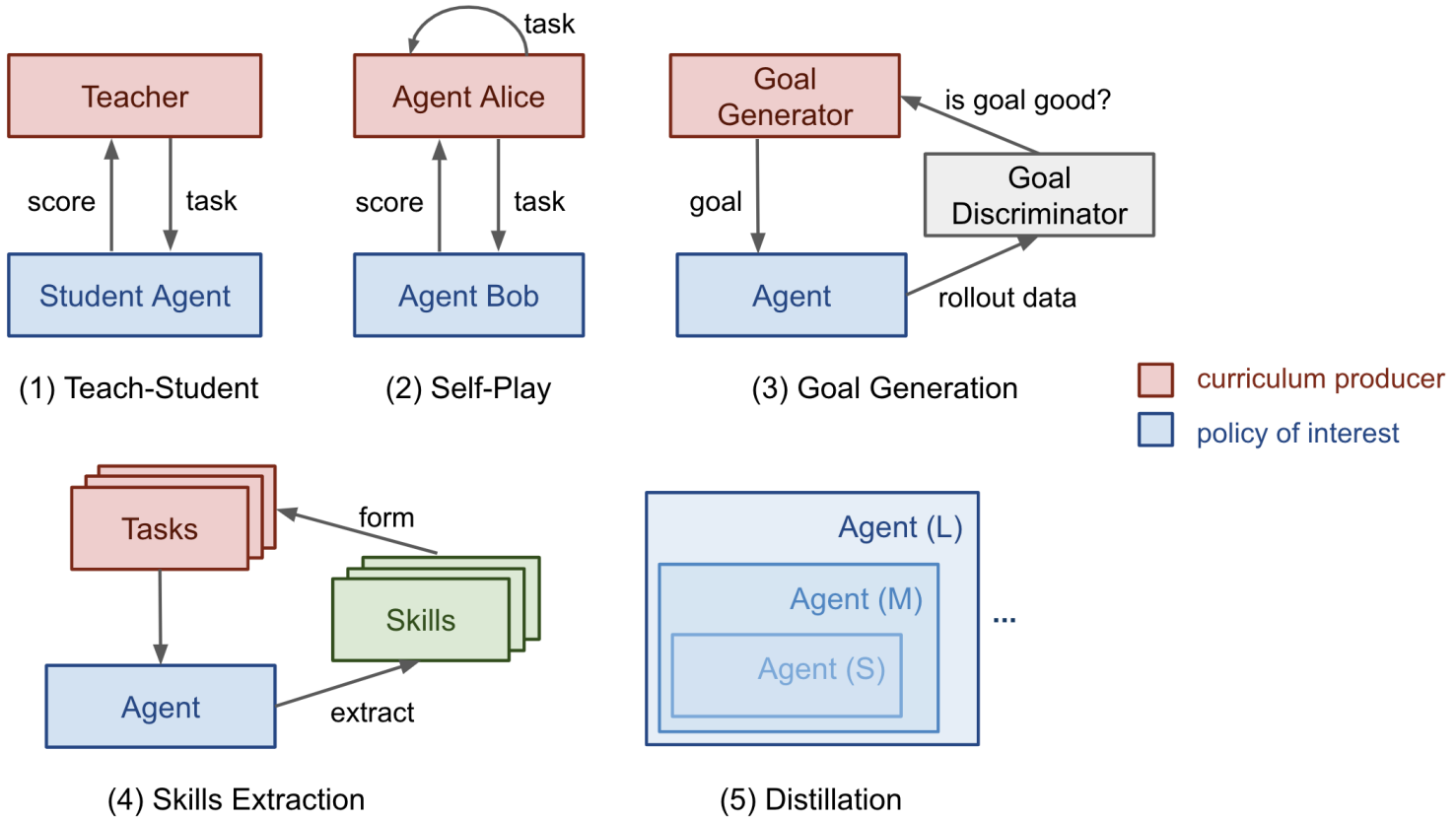
2021.07.29

2021年暑期讨论班

目录

- 针对特定任务的课程学习
- 教师指导的课程学习
- 自动目标生成
- 基于技巧的课程学习
- 蒸馏课程学习

课程学习



在 1993 年，Jeffery Elman 就提出了采用课程学习的方式来训练神经网络的想法。具体来说，Jeffery Elman 使用的课程学习是指：从有限的简单数据集开始，逐渐增加训练样本的难度。

针对特定任务的课程学习

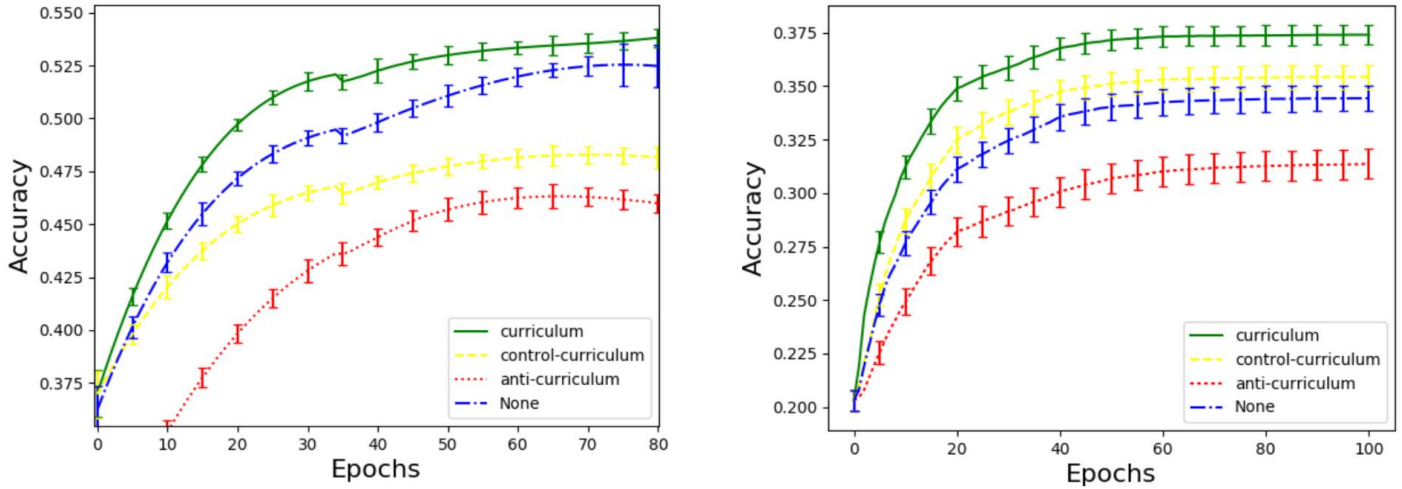
1. 更简洁的样本可能会更快地产生更好的泛化性能。

2. 逐渐加入更困难的样本能够加速在线学习。

Yoshua Bengio, et al. **“Curriculum learning.”** ICML 2009.

针对特定任务的课程学习

通过一个在其他任务上经过预训练的模型在样本上的最小损失来衡量样本难度



a) Large network　　b) Small network

测试图片集（CIFAR100 中的五类 “小型哺乳动物”）上的图片分类准确率。
图中共包含四个实验组，
a）curriculum：根据另一预训练模型（例如，支持向量机）的置信度来对标签排序；
b）control-curriculum：对标签进行随机排序；
c）anti-curriculum：对标签按照与 curriculum 组相反的顺序排序；
d）None：无课程学习。

Daphna Weinshall, Gad Cohen, and Dan Amir. “Curriculum learning by transfer learning: Theory and experiments with deep networks.” ICML 2018.

针对特定任务的课程学习

这篇文章训练了一个 LSTM 去预测一个进行数学运算的简短 Python 程序的输出。

$\texttt{length} \in [1, a]$ $\texttt{nesting} \in [1, b]$.

1.简单课程学习：首先不断增加length到a，接着将length重置到1并且不断增加nesting到b
2.混合课程学习：在length和nesting的区间中随机采样获得样本
3.联合课程学习：简单课程学习+混合课程学习

联合策略总是优于简单课程学习，并且一般（但不总是）都优于混合课程学习。这表明在简单任务的学习过程中混合课程学习对于**避免遗忘**十分重要。

Wojciech Zaremba and Ilya Sutskever. **“Learning to execute.”** arXiv preprint arXiv:1410.4615 (2014).

教师指导的课程学习

使用强化学习来训练教师的策略，学生在这个任务前后得分的变化便是回报

双层优化

教师智能体是一个选择任务的策略，学生智能体则是执行实际任务的强化学习智能体。学生的目标是完美解决一个很难直接学习的复杂任务。为了使得这个复杂任务能够更容易地去学习，我们让教师智能体不断选择合适的子任务来指导学生的训练过程。

在这个过程中，学生应该学习满足以下条件的任务：
1.能帮助学生最快地学习
2.有要被遗忘的风险

Tambet Matiisen, et al. **“Teacher-student curriculum learning.”** IEEE Trans. on neural networks and learning systems (2017).

教师指导的课程学习

训练教师模型等价于部分可观察的马尔可夫决策过程

1. 无法观察到的S_t表示学生的全部状态
2. 观测$o = \left(x_t^{(1)} \ldots\ldots, x_t^N\right)$表示N个任务的相应得分
3. 动作a表示选取一个任务的子序号
4. 每一步得分的定义为得分的差值，$r_t = \sum_{i=1}^{N} x_t^i - x_{t-1}^i$

可以从非平稳多臂赌博机问题中借鉴解决从带噪声的任务分数中估计学习过程，并在探索和挖掘进行平衡的方法——使用 ε-greedy, 或者 Thompson sampling。

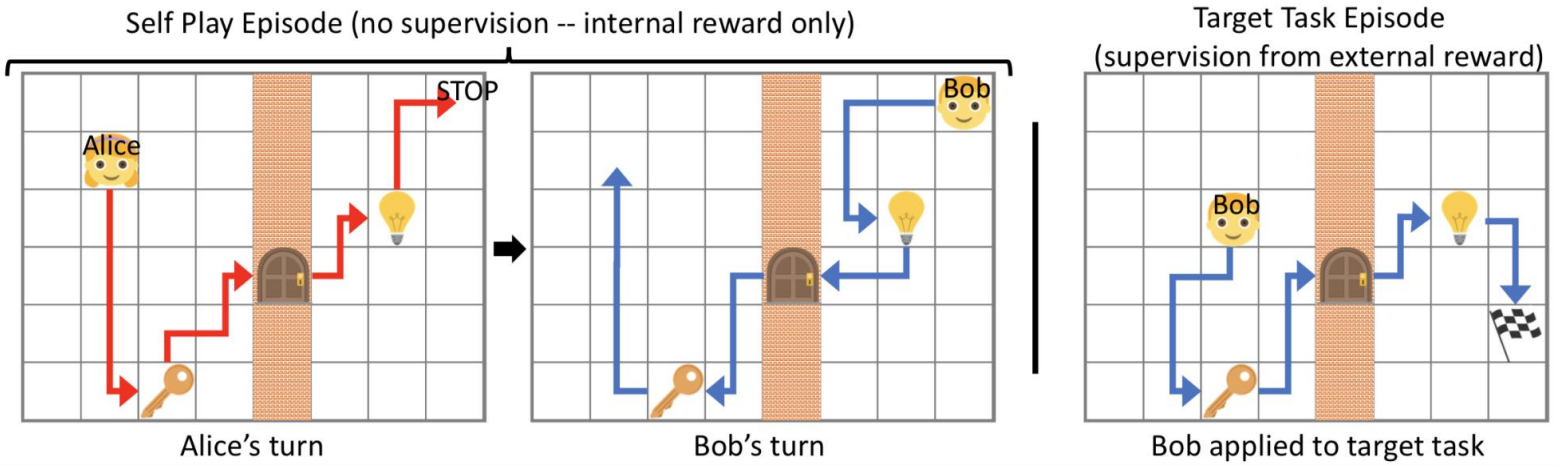
总结来说，核心的思想是使用一个策略去产生使得另一个策略学得更好的子任务。值得思考的是，上述工作均发现从所有的任务随机均匀采样是一个很强的基准算法。

Tambet Matiisen, et al. **“Teacher-student curriculum learning.”** IEEE Trans. on neural networks and learning systems (2017).

通过自我博弈进行课程学习

教师-学生框架（教师指导的课程学习）中，两个智能体做着完全不同的事情。教师学习在不知道任何关于实际任务内容的知识的前提下，为学生挑选任务。

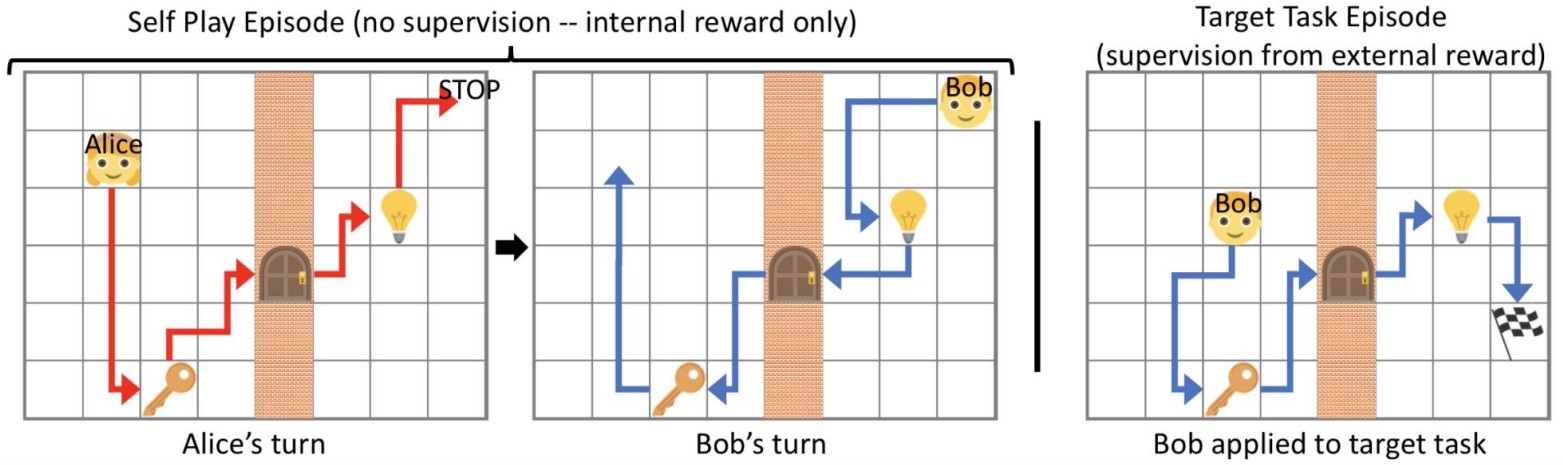
提出了一个通过**不对称自我博弈**的方式来自动化课程学习。两个智能体，Alice 和 Bob，尝试完成具有不同目标的相同任务：Alice 向 Bob 挑战，让 Bob 达到与自己相同的状态；Bob 则试图尽快完成 Alice 的挑战。



在一个包含一个电灯开关、一把钥匙以及一面带有一扇门的墙壁的环境中，智能体需要到达一个目标地点。拿到钥匙可以开门或者关门，关掉电灯让智能体必须使用电灯开关打开电灯。

Sainbayar Sukhbaatar, et al. **"Intrinsic Motivation and Automatic Curricula via Asymmetric Self-Play."** ICLR 2018.

通过自我博弈进行课程学习



可以把 Alice 和 Bob 看作在一个相同的环境里训练的一个强化学习智能体两个独立的、拥有不同大脑的副本。他们中的每一个都拥有各自的参数以及损失函数。自我博弈驱动的训练过程包含两种不同的阶段（episodes）：

- 在自我博弈阶段中，首先Alice将状态从S_0到S_t,接着Bob需要将环境重置到原始状态So从而得到一个内部回报，或者重复Alice的动作。
- 在目标任务阶段中，Bob如果达到目标位置会得到一个外部奖励

因为 Bob 必须重复 Alice状态变化的动作（或逆序，或正序），这个框架只适用于可逆环境或者可重置环境。

Sainbayar Sukhbaatar, et al. **"Intrinsic Motivation and Automatic Curricula via Asymmetric Self-Play."** ICLR 2018.

通过自我博弈进行课程学习

Alice 需要学会将 Bob 拉出他的舒适区，但又不能给他不能完成的任务。Bob 的回报设置为 $R_B = -\gamma t_b$，Alice 的回报设置为 $R_A = \gamma$max(o,tb-ta)。其中 tb 表示 Bob 完成任务的时间，ta表示从开始到 Alice 执行 STOP 动作的时间， γ 是一个标量常数用以放大内部回报到与外部回报同样的数量级。如果 Bob 没能够完成任务，则 tb=tmax-ta。所有的策略都是基于目标的。上述回报函数使得：

1. Bob 会想要尽快完成任务。
2. Alice 会产生让 Bob 花费更多时间才能完成的任务。
3. 当 Bob 无法完成任务时，Alice 下次就不会产生需要过多时间步才能完成的动作。

采用这种方式，Alice 与 Bob 之间的交互就会自动构建一个包含难度不断增加的任务的课程学习。同时，由于 Alice 将任务交给 Bob 之间自己已经完成过了，因而这个任务能够保证一定能被完成。

Sainbayar Sukhbaatar, et al. **“Intrinsic Motivation and Automatic Curricula via Asymmetric Self-Play.”** ICLR 2018.

自动目标生成

强化学习策略通常都需要能够完成一系列的任务。任务的目标一定要小心设置，这样在每个训练阶段，任务对于当前的策略才不会过于简单或者过于困难。一个目标 $g \in G$ 可以被定义为一个状态集合 S^g，无论何时如果一个智能体到达这个状态集合中的任意状态，这个目标都被认为实现了。

生成式目标学习（**Florensa, et al. 2018**）方法依靠一个 **目标 GAN** 来自动生成需要的目标。在这篇文章的实验中，回报十分稀疏，仅有一个目标是否完成的指示器。策略是基于条件的，

$$\pi^* \left(a_t | s_t, g\right) = \arg\max_{\pi} \mathbb{E}_{g \sim p_g(.)} R^g(\pi)$$
$$\text{where } R^g(\pi) = \mathbb{E}_{\pi}\left(\cdot | s_t, g\right) \mathbf{1}\left[\exists t \in [1, \ldots, T] : s_t \in S^g\right],$$

其中$R^g(\pi)$表示期望收益，等价于成功率。给定当前策略采样出的轨迹数据，只要属于某个轨迹中的任何状态属于目标集合，收益就会是正值。

Carlos Florensa, et al. **“Automatic Goal Generation for Reinforcement Learning Agents”** ICML 2019.

自动目标生成

这篇文章迭代执行以下三步直到策略收敛：

1. 根据是否目标对于当前策略来说处于一个合适的等级，或者说难度，来对一个目标集合打标签。
   处于合适等级或者说难度的目标集合被命名为 GOID（Goals of Intermediate Difficulty 的简称）。
   $\mathrm{GOID}_i := \{g : R_{\min} \leq R^g(\pi_i) \leq R_{\max}\} \subseteq G$
   R_{min}以及 R_{max}可以被解释为在T时间步内，到达目标的最小以及最大概率。
2. 使用上一步得到的带标签的目标来训练一个 Goal GAN 模型，用以产生新目标。
3. 使用这些新目标来训练策略。

Goal GAN 能够自动产生一个课程学习策略：

生成器 $G(z)$：产生一个新的目标（希望是一个从 GOID 集合中均匀采样出的一个目标）。
判别器 $D(g)$：判别一个目标是否能够被实现（希望能够告诉我们一个目标是否来自 GOID 集合）。

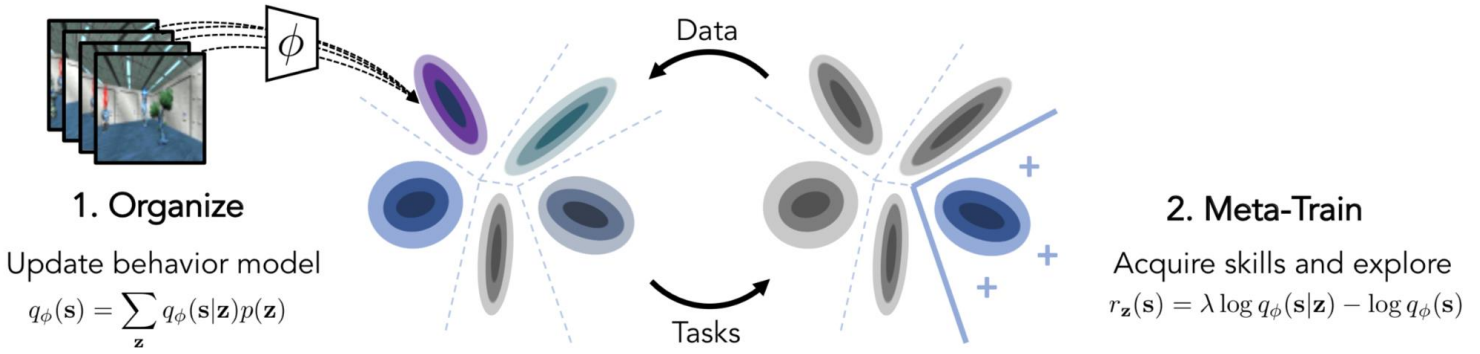
Carlos Florensa, et al. **“Automatic Goal Generation for Reinforcement Learning Agents”** ICML 2019.

基于技巧的课程学习

另一种观点是将智能体能够完成的工作分解为各种技巧，并将每个技巧集映射为一项任务。 当一个智能体在无监督的情况下与环境交互时，是否有办法从这种交互中发现有用的技巧，并通过课程学习将其进一步扩展到解决更复杂任务的解决方案中？

提出了一种自动化课程学习的算法，**CARML**（“Curricula for Unsupervised Meta-Reinforcement Learning”，无监督元强化学习的课程学习算法）。CARML 将无监督的轨迹数据映射到一个隐含技巧空间，从而有利于 **meta-RL** 策略的训练（即可以迁移到训练阶段未见过的任务中）。

CARML 是使用像素级的观测来训练但 DIAYN 是使用真实的状态的来训练。θ参数化的强化学习算法 π_θ通过无监督交互来训练，该交互被形式化为结合了一个习得的奖励函数的 CMP。此设置可以应用到元学习上，因为自定义的奖励函数只能在测试时给出。

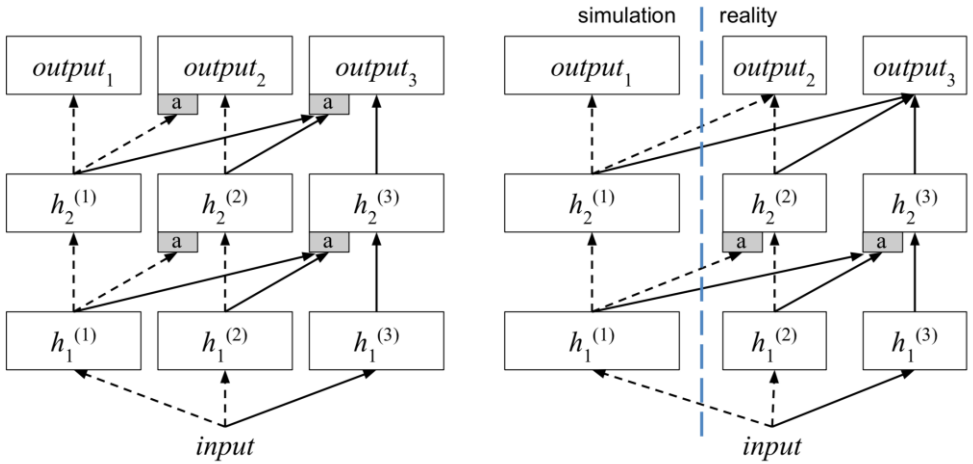


CARML 图示，共包含两个步骤：1）将实验数据构建到一个隐含的技巧空间中；2）使用从习得的技巧构建的回报函数对策略进行元训练。

Allan Jabri, et al. **“Unsupervised Curricula for Visual Meta-Reinforcement Learning”** NeuriPS 2019.

通过蒸馏进行课程学习

渐进神经网络（**progressive neural network**）架构的动机是在不同任务之间有效地迁移习得的技能，同时避免灾难性遗忘。 课程学习是通过一组逐步堆叠的神经网络塔（如本文中的“列”）实现的。



(a) progressive network　　(b) progressive network for sim2real

1. 该网络从一列开始，包含 L 层神经元，对应的激活层记为 $h_i^{(1)}, i = 1, \ldots, L$。我们首先将这个网络在一个任务上训练至收敛，得到参数集合 $\theta^{(1)}$。
2. 一旦转移到下一个任务，我们在原先的网络结构上增加一列以适应新内容的同时，冻结 $\theta^{(1)}$ 从而保留从上一个任务习得的技巧。新的一列的激活曾记为 $h_i^{(2)}, i = 1, \ldots, L$，参数记为 $\theta^{(2)}$。
3. 步骤 2 每遇到一个新任务时均重复执行一次。第 k 列的第 i 层激活层接受所有存在的列的之前的激活层的输入：

$$h_i^{(k)} = f\left(W_i^{(k)} h_{i-1}^{(k)} + \sum_{j<k} U_i^{(k:j)} h_{i-1}^{(j)}\right),$$

其中 $W_i^{(k)}$ 表示第 k 列第 i 层的权重矩阵；$U_i^{(k:j)}, j < k$ 表示用以将第 j 列第 $i-1$ 层投影到第 k 列第 i 层的权重矩阵（$j < k$）。上述所有的权重矩阵都是要学习的。$f(\cdot)$ 表示非线性激活函数。

Andrei A. Rusu et al. “Progressive Neural Networks” arXiv preprint arXiv:1606.04671 (2016).

通过蒸馏进行课程学习

该论文通过在多个 Atari 游戏上训练渐进网络，来检验在一个游戏中学习的特征是否可以迁移到另一个游戏中。 结果表明确实是这样。 尽管有趣的是，在之前的列中学到的具有高度相关性的特征并不总是在新任务上带来良好的迁移性能。 对此，一个猜想是，从旧任务中学到的特征可能会在新任务中引入偏差，从而导致策略陷入次优解。 总体而言，渐进网络比仅微调顶层网络更好，并且可以实现与微调整个网络相似的迁移性能。

Andrei A. Rusu et al. “Progressive Neural Networks” arXiv preprint arXiv:1606.04671 (2016).

谢 谢