Deep Recurrent Q-Learning for Partially Observable MDPs（部分可观测马尔可夫决策过程的深度循环Q学习）

作者：Matthew Hausknecht，Peter Stone

单位：Department of Computer Science The University of Texas at Austin

论文发表会议：National conference on artificial intelligence

论文发表时间：Submitted on 23 Jul 2015, last revised 11 Jan 2017

论文查看网址：https://arxiv.org/abs/1507.06527

论文贡献：提出一种基于DQN的神经网络模型（DRQN），将包含卷积神经网络（CNN）的DQN模型和LSTM结合，使强化学习智能体拥有记忆力的特性。

一. 写作动机

Why:

在 *Playing Atari with Deep Reinforcement Learning(Mnih et al., 2013)* 中，DQN是使用智能体（Agent）遇到的包含当前状态的最后4个状态的组成（最后4个画面）作为输入。目的是获得画面中物体/角色的方向、速度等信息。但换句话说，倘若遇到需要记忆特征超过四个画面的时间跨度任务时，对于DQN来说，则会由马尔可夫决策过程（MDP）变成部分可观测的马尔可夫决策过程（POMDP）。

What:

部分可观测的马尔可夫决策过程（Partially-Observable Markov Decision Process, POMDP）是指：当前观测（Observation，obs）的不完整且带有噪音，不包含环境运作的所有状态。导致无法作为环境（Environment，env）的完整描述信息（智能体得到观测跟环境的状态不等价）。

How:

论文作者提出，为避免因部分可观测的马尔可夫决策过程（POMDP）导致DQN在任务环境学习的过程中出现性能下降，引入**Deep Recurrent Q-Network (DRQN)**，是基于LSTM（Long Short-Term Memory，LSTM）和DQN的组合。并证明使用**DRQN**能有效处理部分可观测的马尔可夫决策过程（**POMDP**），当评估智能体时，输入智能体的观测（obs）发生变化（遮盖、画面闪烁）时，因参数化价值函数（Value function）包含循环神经网络层（LSTM）能够使学习到的 策略π_θ 具有鲁棒性，不会发生策略崩塌。

二. 背景介绍

1. Deep Q-Learning（深度Q学习）

使用深度Q学习方法，是通过参数为θ的深度神经网络来近似价值函数（Value Function）$\mathrm{V}(\mathrm{s})$或动作价值函数（Action-Value Function）$\mathrm{Q}(\mathrm{s}, \mathrm{a})$来隐式的学习最优策略$\pi^*$，输入环境的观测（obs），输出对观测

（obs）估计的V值或Q值。

深度Q学习适用场景：连续状态空间（State space）离散动作空间（Action Space）任务。

价值函数的作用为：评估在当前状态-动作下，未来回报（Return）的期望。

使用深度神经网络作为强化学习的参数化值函数近似器的优点：

（1）具有深度学习自动提取特征的能力。

（2）参数化模型将现有可见的观测（obs）泛化到没有见过的观测（obs）：$|\theta| \ll |\mathrm{S} \times \mathrm{A}|$

（3）参数化模型可通过求导数的形式来更新神经网络模型参数。

参数化价值函数为：

$$\begin{aligned} \mathrm{V}_\theta(\mathrm{s}) &\cong \mathrm{V}^\pi(\mathrm{s}) \\ \mathrm{Q}_\theta(\mathrm{s},\mathrm{a}) &\cong \mathrm{Q}^\pi(\mathrm{s},\mathrm{a}) \end{aligned}$$

深度Q学习保持学习稳定的技巧（Trick）：

（1）经验回放（Experience Replay）：针对数据层面的相关性和数据分布变化做改进，使得数据尽可能满足独立同分布（i.d.d）属性。

（2）目标网络（Target Network）：解决在时序差分（Timing Difference，TD）学习时，TD target和当前Q网络高度相关的问题。

深度Q学习的损失函数（Loss Function）为：

$$\begin{aligned} \mathcal{L}_\mathrm{i}\left(\theta_\mathrm{i}\right) &= \mathbb{E}_{(\mathrm{s},\mathrm{a},\mathrm{r},\mathrm{s}')\sim\mathcal{D}}\left[\left(\mathrm{y}_\mathrm{i} - \mathrm{Q}\left(\mathrm{s},\mathrm{a};\theta_\mathrm{i}\right)\right)^2\right] \\ \mathrm{y}_\mathrm{i} &= \mathrm{r} + \gamma \max_{\mathrm{a}'} \hat{\mathrm{Q}}\left(\mathrm{s}',\mathrm{a}';\theta^-\right) \end{aligned}$$

深度Q学习的最优动作a^*为当前状态（state）下价值函数输出Q值最大对应的动作：

$$\mathrm{a} = \arg\max_{\mathrm{a}} \mathrm{Q}(\mathrm{s},\mathrm{a})$$

2. Partial Observability（部分可观测）

马尔可夫决策过程（MDP）五元组（S，A，P，γ，R）：

（1）S是状态的集合

（2）A是动作的集合

（3）$\mathrm{P}\left(\mathrm{s}'|\mathrm{s},\mathrm{a}\right)$是环境的状态转移概率

（4）$\gamma \in [0,1]$是对未来累积奖励（回报 Return）的折扣因子

（5）R是奖励函数，R:S × A

部分可观测的马尔可夫决策过程（POMDP）七元组（S，A，P，γ，R，Ω，O）：

（1）Ω是观测$(\mathrm{obs},\mathrm{o})$的集合，$\mathrm{o} \in \Omega$

（2）O是观测函数，$\mathrm{O}(\mathrm{s}',\mathrm{a},\mathrm{o}) = \mathrm{P}\left(\mathrm{o}|\mathrm{s}',\mathrm{a}\right)$

因为观测（obs）是对状态的部分描述，所以可能会遗漏一些信息。

$$Q(o, a \mid \theta) \neq Q(s, a \mid \theta)$$

论文作者通过实验证明，使用DRQN可以缩小价值函数对于观测（obs）的Q值与对状态（State）的Q值之间的差距。即智能体学习到的策略能在具有POMDP性质的任务环境中具有鲁棒性。

三. 模型架构

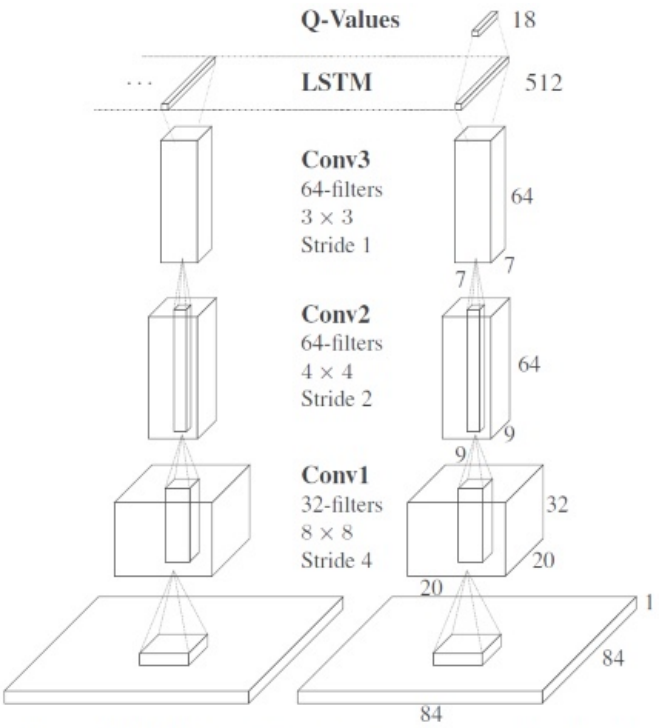


Figure 2: DRQN convolves three times over a single-channel image of the game screen. The resulting activations are processed through time by an LSTM layer. The last two timesteps are shown here. LSTM outputs become Q-Values after passing through a fully-connected layer. Convolutional filters are depicted by rectangular sub-boxes with pointed tops.

模型输入：Atari 游戏的84*84像素的单通道图像

模型输出：游戏对应18个离散动作的Q值

模型架构解释：

①首先使用3层卷积神经网络（Convolutional Neural Networks，CNN）提取图像特征。

②其次传入LSTM，获得游戏画面之间的序列特征。

③最后使用全连接层（Fully Connected Layers，FC）变成Q值。

四. 智能体更新方法

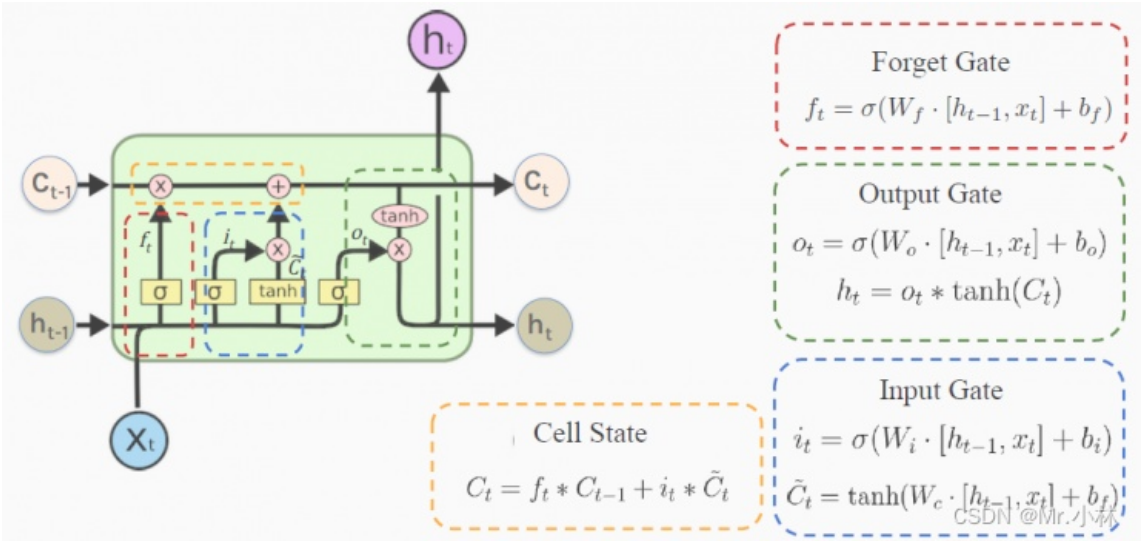
Bootstrapped Random Updates：从经验池（Replay Memory）中随机选择一个回合的轨迹τ ，并从该回合的经验片段中随机选择开始点沿时间步骤顺序更新模型。（以时序差分Timing Difference的方式更新）

$$\tau = \{s_0, a_0, s_1, r_1, a_1, \cdots, s_{t-1}, r_{t-1}, a_{t-1}, s_t, r_t\}$$

例如：选择从轨迹τ的s_2进行顺序更新，直到轨迹τ的终止状态s_t停止。

设计缺陷：使用随机更新方式符合DQN中随机采样经验的更新策略。但在本论文的价值函数设计中包含LSTM层。导致在每次训练更新开始时，**LSTM**会因为 隐含状态h_{t-1} 的输入为零，导致LSTM难以学习比时间序列反向传播到达的时间步骤更长的时间范围。

LSTM结构图：



遗忘门（Forget Gate）：控制哪些数据保留，哪些数据要遗忘。

输入门（Input Gate）：控制网络输入数据流入记忆单元的多少。

输出门（Output Gate）：扛着记忆但与对当前输出数据的影响，即记忆单元中的哪一部分会在时间步t输出。

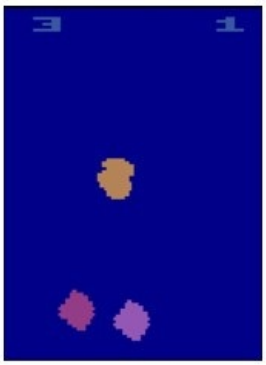
细胞状态（Cell State）：将旧的细胞状态更新为当前细胞状态，由遗忘门和输入门共同控制。

本文使用的智能体的更新方式均为随机更新策略。

五. 任务环境

本文选择9个Atari 2600游戏任务环境进行评估基于DRQN神经网络模型的智能体(Agent)性能。

游戏环境示意图为：



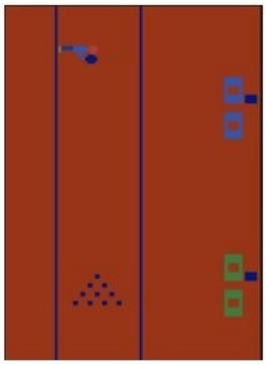
Asteroids

Ms Pacman

Beam Rider

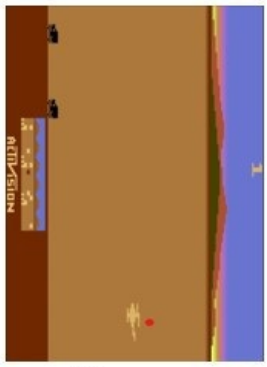
Ice Hockey

Double Dunk

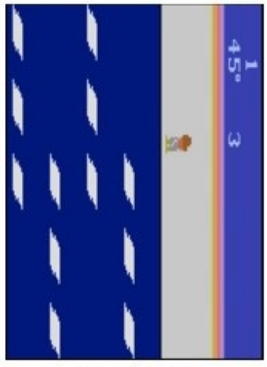


Bowling

Centipede



Chopper Command



Frostbite

游戏环境	介绍
Asteroids	具有自然闪烁的NPC，使其成为潜在的循环学习候选
Ms Pacman	通关类游戏，有闪烁的幽灵和强力药丸
Frostbite	平台类游戏
Beam Rider	射击类游戏
Centipede	射击类游戏
Chopper Command	射击类游戏
Ice Hockey	体育比赛类
Double Dunk	体育比赛类
Bowling	体育比赛类

- Atari 2600在线游戏网页

六. 损失函数与奖励函数

损失函数（Loss Function）：论文作者主要提出基于DRQN的神经网络模型，没有对DQN的强化学习算法进行更改，仍然采用DQN的损失函数进行神经网络模型拟合（动作价值函数，Action-Value Function）。

$$\mathcal{L}_{\mathrm{i}}\left(\theta_{\mathrm{i}}\right)=\mathbb{E}_{\left(\mathrm{s,a,r,s}^{\prime}\right)\sim\mathcal{D}}\left[\left(\mathrm{y_{i}}-\mathrm{Q}\left(\mathrm{s,a};\theta_{\mathrm{i}}\right)\right)^{2}\right]$$
$$\mathrm{y_{i}}=\mathrm{r}+\gamma\max_{\mathrm{a}^{\prime}}\hat{\mathrm{Q}}\left(\mathrm{s}^{\prime},\mathrm{a}^{\prime};\theta^{-}\right)$$

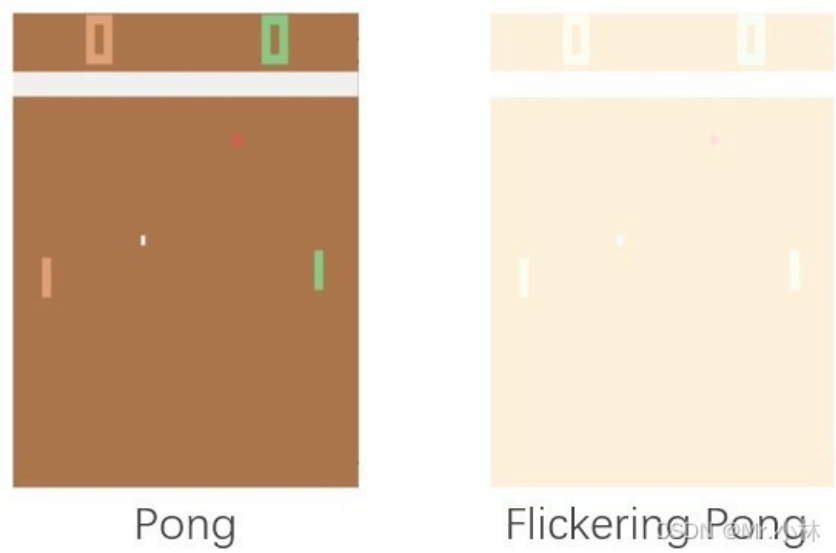
奖励函数（Reward Function）：论文使用的任务环境为Atari 2600游戏环境，根据不同的游戏任务，环境都自带奖励函数，不需要额外定义。

七. 实验证明

1. Flickering Pong POMDP（闪烁的Pong，部分可观测马尔可夫决策过程）

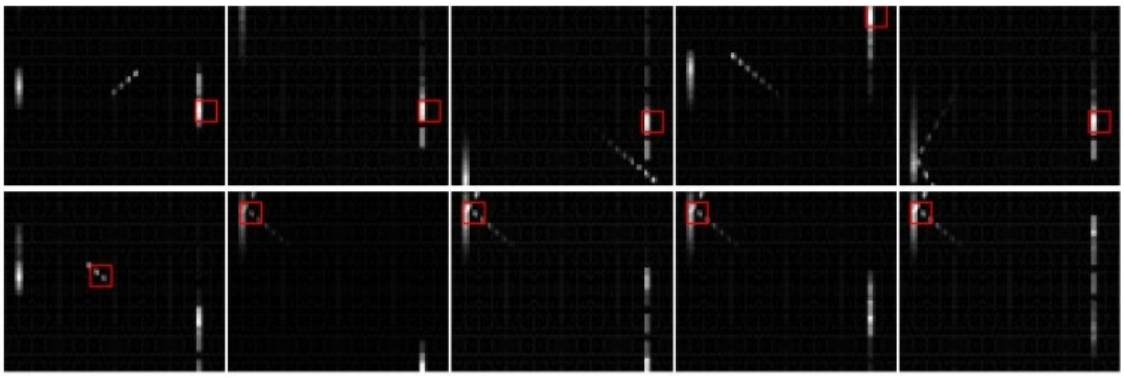
DQN通过输入包含当前观测（obs）的最后四个画面来实现将部分可观测的马尔可夫决策过程（POMDP）转换成为马尔可夫决策过程（MDP）。

实验目的：而为了验证DRQN在具有POMDP性质的游戏环境中对连续的模糊输入具有鲁棒性。引入 **Flickering Pong POMDP** 对Pong游戏的修改。



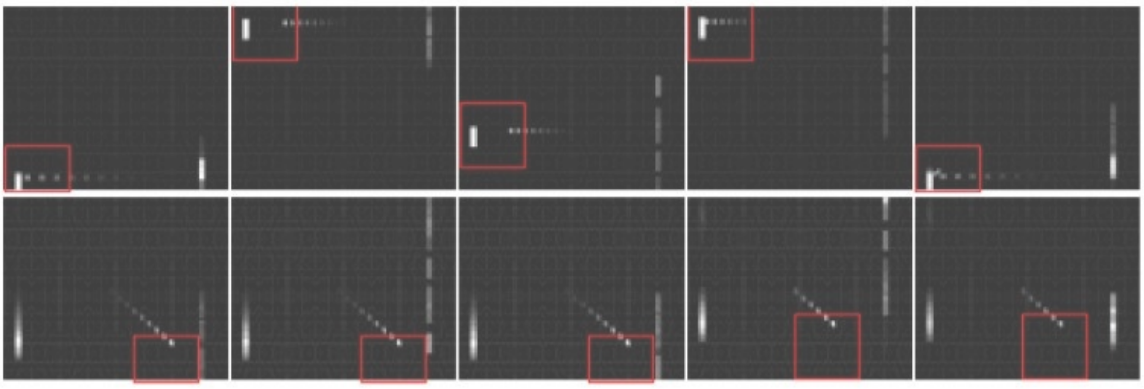
实验设计：在每个时间步长，游戏画面完全显示或完全模糊的概率 $\mathrm{p}=0.5$。使用这种方式使 Flickering Pong 游戏环境 有一半的概率，画面被模糊化，使智能体得到观测（obs）具有POMDP性质。
为了验证 ：
①DQN神经网络模型中卷积层（CNN）能够具有在模糊观测画面中检测物体移动 速度、方向 的能力，在Flickering Pong游戏环境中输入包含当前观测画面的最后10个画面到模型中，并以可视化方式确认。
②DRQN神经网络模型中LSTM层具有在模糊观测画面中检测物体历史特征，在Flickering Pong游戏环境中输入当前观测画面到模型中，并以可视化方式确认。

实验**baseline**：DQN神经网络模型（输入包含当前观测画面的最后**10**个画面）。



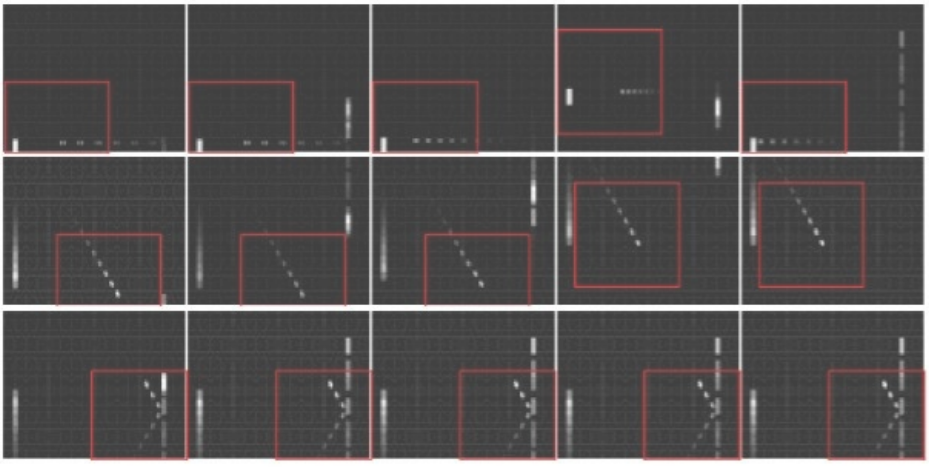
(a) Conv1 Filters

现象：在第一层卷积层中，滤波器主要检测到 Flickering Pong游戏环境 中的球拍。



(b) Conv2 Filters

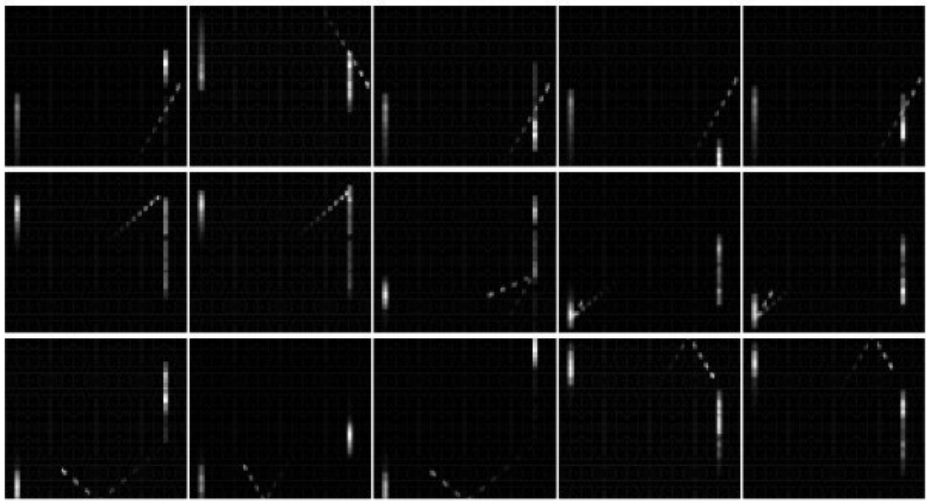
现象：在第二层卷积层中，滤波器开始检测到 Flickering Pong游戏环境 中的球拍以及球的运动方向，有的状态下滤波器也能同时检测到球拍、球。



(c) Conv3 Filters

现象：在第三层卷积层中，滤波器都能检测到球拍和球的相互作用，有偏转、球的速度和移动方向。

实验：为了验证DRQN神经网络模型中LSTM层能够检测到画面与画面间的序列特征，在每个时间步内仅输入当前观测（obs）到的**1**个画面到DRQN模型中。



(d) Image sequences maximizing three sample LSTM units

现象：LSTM层各单元能够透过具有闪烁的单个观测画面，检测到Flickering Pong游戏环境中的高级事件，例如球从墙上反弹等。

实验结论：通过 输入**10**个连续观测画面到含有卷积层的**DQN** 以及 输入**1**个观测画面到含有**LSTM**层的**DRQN** 进行对比，结果显示都能检测到游戏画面的历史特征。基于此，当任务环境具有POMDP性质时，上述两种方法都可供选择。证明使用**LSTM**神经网络层，能够在具有时间序列特征的单个观测画面中整合序列特征。

2. Evaluation on Standard Atari Games （标准的**Atari**游戏评估）

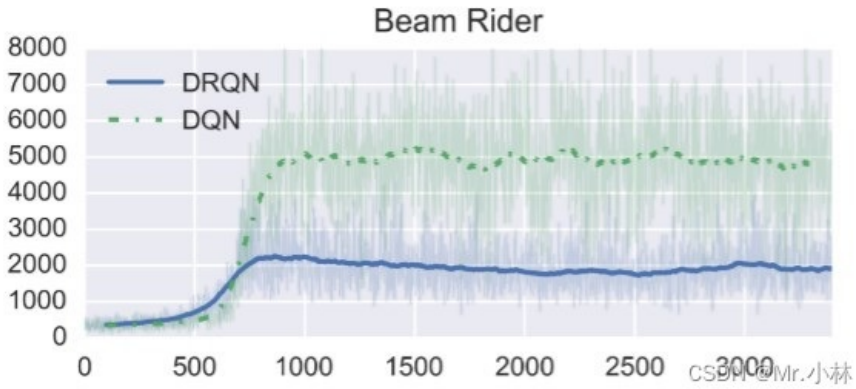
实验目的：为了对比论文作者提出的DRQN神经网络模型和基于卷积神经网络（CNN）的DQN神经网络模型在标准的Atari 2600游戏任务环境中的表现，使用9个不同的 Atari 2600 游戏环境进行综合评估。

实验设计：为了让任务符合MDP性质，设置输入为包含当前观测画面的最后4个画面。并使用 独立 **t** 检验计算得分的统计显著性，显著性水平$\mathrm{P} = 0.05$

实验**baseline**：基于卷积神经网络（CNN）的DQN神经网络模型在9个标准的Atari 2600游戏任务环境中的表现。

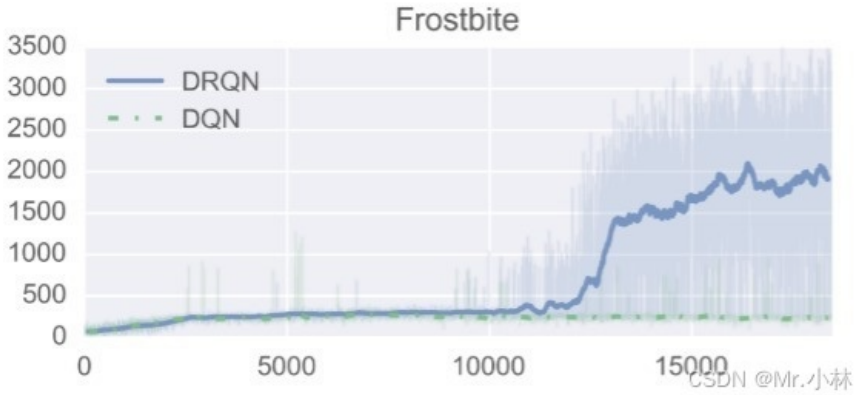
	DRQN $\pm std$	DQN $\pm std$	
Game		Ours	Mnih et al.
Asteroids	1020 ($\pm$312)	1070 ($\pm$345)	1629 ($\pm$542)
Beam Rider	3269 ($\pm$1167)	**6923** ($\pm$1027)	6846 ($\pm$1619)
Bowling	62 ($\pm$5.9)	72 ($\pm$11)	42 ($\pm$88)
Centipede	3534 ($\pm$1601)	3653 ($\pm$1903)	8309 ($\pm$5237)
Chopper Cmd	2070 ($\pm$875)	1460 ($\pm$976)	6687 ($\pm$2916)
Double Dunk	**-2** ($\pm$7.8)	-10 ($\pm$3.5)	-18.1 ($\pm$2.6)
Frostbite	**2875** ($\pm$535)	519 ($\pm$363)	328.3 ($\pm$250.5)
Ice Hockey	-4.4 ($\pm$1.6)	-3.5 ($\pm$3.5)	-1.6 ($\pm$2.5)
Ms. Pacman	2048 ($\pm$653)	2363 ($\pm$735)	2311 ($\pm$525)

实验结论**1**：在9个不同的游戏任务环境中，基于DRQN神经网络模型的智能体得分在5个环境上的表现优于基于DQN神经网络模型的智能体。



实验结论**2**：DRQN在Beam Rider游戏中表现相比DQN明显更差。

个人分析：可能是Beam Rider游戏中的决策依据并不需要时间跨度太久远的观测画面特征，主要依于当前即时观测画面的特征。加入的LSTM层反而导入过多不必要的特征造成智能体决策上的干扰。



实验结论**3**：DRQN在 Frostbite游戏 中表现最好。游戏的任务是要求玩家跳过所有四排移动的冰山并返回屏幕顶部。多次穿越冰山后，已经收集到足够的冰块，可以在屏幕右上角建造一个冰屋。随后玩家可以进入冰屋进入下一关。

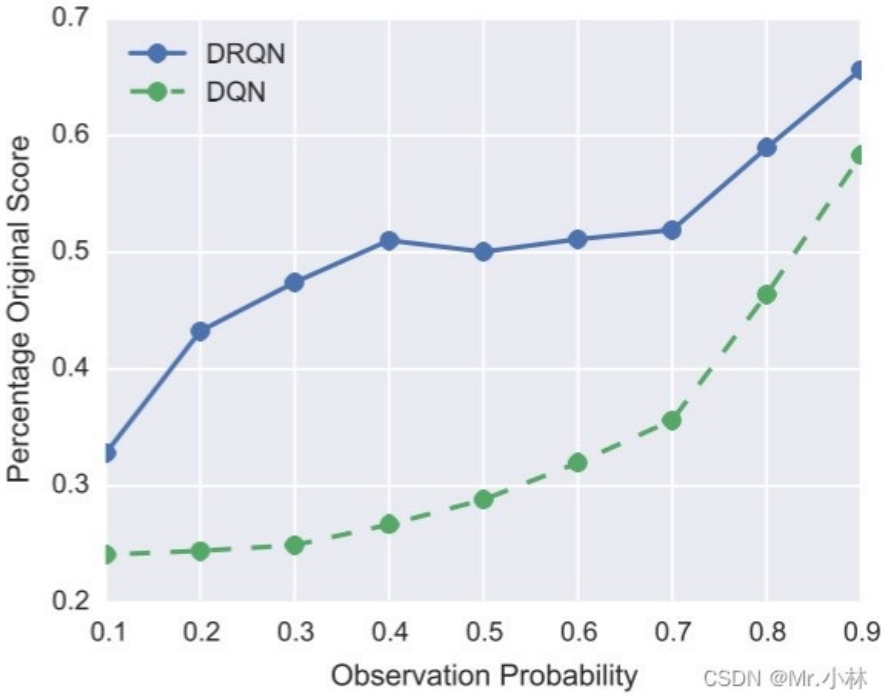
个人分析： Frostbite游戏任务 具有长时间跨度决策性质，因此DRQN的LSTM层能够学习到具有长时间跨度的特征，智能体决策上的依据能够超越输入模型的4个观测画面所具有的特征。

3. MDP to POMDP Generalization（MDP到POMDP的泛化性）

实验目的：为了评估使用基于DRQN模型的智能体在标准的MDP（输入为包含当前观测画面的最后4个画面）上训练得到的策略π，在推广到POMDP性质的相同游戏任务环境时，智能体的策略π能否保持有效性？

实验设计：选择用实验2的9个Atari 2600游戏任务环境进行闪烁（Flickering）设置，在每个时间步长，游戏画面完全显示的概率p按照0.1~0.9的概率依次使用训练好的智能体进行实验。

实验**baseline**：基于卷积神经网络（CNN）的DQN神经网络模型在POMDP性质的Atari 2600 游戏任务环境中的百分比原始分数（Percentage Original Score）。



实验结论：在标准的MDP训练的基于DRQN模型的智能体和基于DQN模型的智能体，分别在闪烁（Flickering）设置的游戏环境中进行评估，通过折线图观察到，基于DRQN模型的智能体性能下降的幅度相比基于DQN模型的智能体更小。

$$\mathrm{Percentage\ Original\ Score} = \frac{\sum_{i=1}^{9} \mathrm{POMDP\ Score}}{\sum_{i=1}^{9} \mathrm{MDP\ Score}}$$
$$\mathrm{i\ is\ the\ number\ of\ game\ environments}$$

八. 相关工作

1. LSTM在解决具有POMDP性质的任务上相比于RNN的优越性

在 *Reinforcement learning with long shortterm memory(Bakker et al., 2001)* 这篇论文中，在具有POMDP性质的Corridor和Cartpole环境上，使用LSTM神经网络作为强化学习中的 优势函数（Advantage Function）近似，相比于使用RNN神经网络能够更好的完成任务。虽然Corridor和Cartpole环境的状态空间特征数量以及动作空间，相比于Atari 2600 游戏环境都很少！

2. LSTM在解决具有POMDP性质的任务框架

在 *Solving deep memory POMDPs with recurrent policy gradients(Wierstra et al., 2007)* 这篇论文首先使用策略梯度（Policy Gradient）结合LSTM神经网络解决具有POMDP性质的问题。但模型在特征提取上属于手工设计的特征，并没有结合深度神经网络模型做到自动提取特征。

九. 结论

1. 基于DRQN模型的智能体在 POMDP性质的 Atari 2600 游戏环境下处理观测（obs）时，能够利用单个观测画面的输入，达到与基于DQN模型的智能体在输入连续10个观测画面得到的历史特征相同。即使用基于DRQN模型能够降低计算复杂度。
2. 根据 标准的 Atari 2600 游戏环境的任务属性不同，在需要接收长时间跨度特征的任务中（例如 Frostbite 游戏），基于DRQN模型的智能体的表现会远超于基于DQN模型的智能体。
3. 基于DRQN模型的智能体在相同的 Atari 2600 游戏环境下遇到出现POMDP的情况下，其泛化性会比基于DRQN模型的智能体好。

十. 贡献

本文通过实验证明，加入 **LSTM**神经网络 的DRQN模型的智能体，面对具有POMDP性质的问题上，其性能表现优于DQN。但在实践过程中，仍需以任务的实际情况而定。通用性不高，并无法带来系统性的提升。

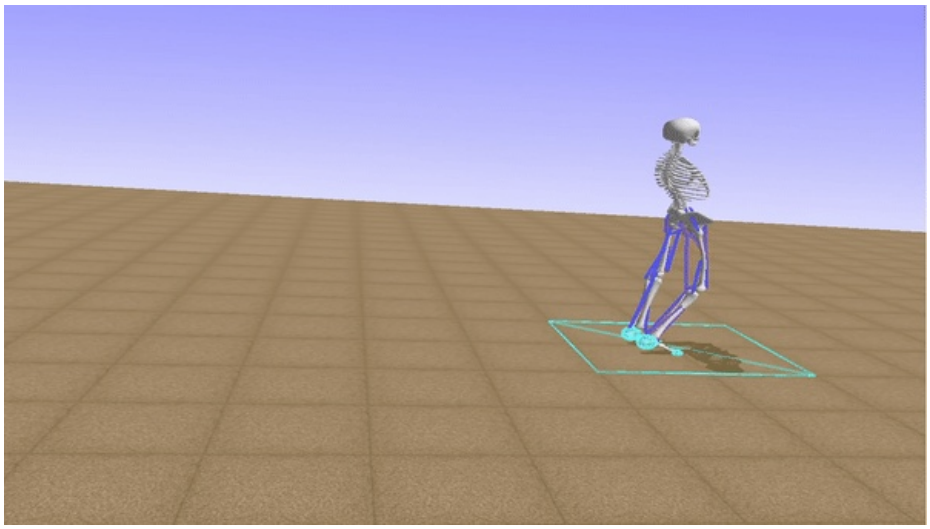
十一. 下一步的工作

1. 强化学习算法层面

在连续状态空间、离散动作空间的任务中，有结合**Double DQN**和**Dueling DQN**的**D3QN**算法。可以在本论文提出的DRQN神经网络模型架构上，使用D3QN强化学习算法来进行实验，观察智能体在本次论文的实验中表现不好的任务环境（例如 Beam Rider 游戏）是否能够得到性能上的提高，并达到系统性的提升。

2. 任务环境层面

在强化学习机器控制领域上的任务情境中，智能体不仅需要对环境输出的观测（obs）学习时间序列特征，还需要将智能体本身输出的动作（Action）纳入成为观测的一部分进行学习。期待之后智能体训练好的 策略π 既能够对环境输出的观测进行考量，也能将历史动作纳入考虑的范围，以输出连贯合理的决策动作。



3. 深度神经网络模型层面

随着自注意力机制为核心的Transformer神经网络模型的出现，Transformer抛弃传统的CNN和RNN，只使用注意力机制（Attention）。同时Vision Transformer的出现打破了NLP领域和CV领域的边界，以Transformer为代表的决策模型成为当前RL领域的新范式。
那么是否在同样是以图像作为观测的 Atari 2600 游戏任务环境上，以DQN算法为基础，使用 基于**Vision Transformer**模型 拟合价值函数，来隐式的学习策略π，以期望在智能体能够达到更好的性能！

参考文献

1. Volodymyr Mnih, J. et al. 2013. Playing Atari with Deep Reinforcement Learning.
2. Wierstra, D.; Foerster, A.; Peters, J.; and Schmidthuber, J. 2007. Solving deep memory POMDPs with recurrent policy gradients.
3. Bakker, B. 2001. Reinforcement learning with long shortterm memory. In NIPS, 1475–1482. MIT Press.
4. 推荐中的序列化建模：Session-based neural recommendation

5. 论文十问-快速理解论文主旨的框架
6. RLChina 2022 强化学习暑期课
7. 蘑菇书EasyRL

==========

作者：林俊杰

研究单位：台湾成功大学制造资讯与系统研究所

研究方向：强化学习、深度学习

联系邮箱：554317150@qq.com