

**摘要**

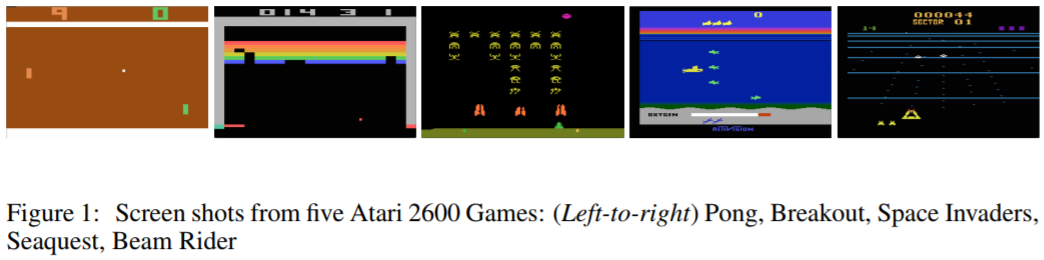
我们（DeepMind）隆重的介绍史上第一个使用强化学习（RL）技术从高维感知信号直接学习控制策略的深度学习（DL）方法。这个计算模型是一个卷积神经网络模型，并由一种Q-learning的变量进行训练。模型的输入是图像像素信号，而输出则是一个用来估计长期奖励的价值函数。我们在Arcade Learning Environment 中选取了7个雅利达2600游戏进行了通用模型实验，即所有7个游戏都采用一样的模型结构。试验结果表明这样的人工智能（AI）模型能在其中的6个游戏中超越之前的研究结果，3个游戏能够达到人类专家水平。

**1 介绍**

如何从高维度输入信号（例如视觉与语言）直接学习控制方法一直以来都是RL的巨大挑战。大多数成功的案例都依赖特征工程来提取信号中的特征，然后使用线性值函数或策略表示技术。讲的再直白一些，这些系统的性能完全取决于特征表示的质量。

如今流行的深度学习（DL）技术能够从源源不断的信号中直接提取高维特征，这样的优势引领了计算机视觉[11, 22, 16]与语音识别[6, 7]。相关方法都使用了大规模神经网络，包括卷积神经网络、多层感知器、受限玻尔兹曼机以及循环神经网络，同时DL技术带来了人们对监督学习与非监督学习的再次探究的机会，是时候思考一下同样的技术是否有助于RL感知数据了。

不过站在DL的角度去考虑RL，我们会发现许多需要解决的问题。第一个问题就是深度学习通常需要大量的带标签数据，而RL则通常从一些频繁、稀疏、充满噪音与延迟的标量奖励信号中进行学习，所以DL的学习效率可能是RL的千倍以上，这使得两种方法在学习过程上较难进行结合。另一个问题就是DL面对的是固定的、独立的学习样本，而RL通常应用于一个学习过程，在这个过程中的学习样本往往是高度相关甚至连续的（例如机械臂抓取物体时的线速度与角速度），且RL学习过程会进行不同的动作尝试，这样会在训练过程中遇到不同的样本集合。



本文证明了卷积神经网络+RL的模型可以解决上述问题，该模型能从复杂环境中的原始视频数据中学习成功的控制策略。我们使用了如下技术来达成这项挑战：

* Q学习[26]算法的变体训练网络；
* 使用随机梯度下降来更新权重；
* 为了缓解相关数据和非平稳分布的问题，我们使用经验重放机制[13]，对先前进行过的场景进行随机抽样，从而让许多历史行为平滑的分布在训练数据中。

我们在Atari 2600游戏环境中实施了上述方法，Atari 2600是一个极富挑战性的RL测试平台，可为AI提供高维视觉输入（210×160 RGB视频，60Hz）以及各种有趣且充满难度的关卡。我们的目标是设计一个能够尽可能玩多个游戏的通用神经网络AI，不对其提供任何特别的游戏信息或人工设计的特征，AI不了解游戏模拟器内部的任何机制，只有视频输入、奖励、游戏控制信号与终端提示，总之，我们要让AI真正像人类一样学会玩游戏。除此之外，网络结构与超参数配置在训练过程中不会发生变化，到目前为止，该AI在已经尝试过的7个游戏中，有6个超越了之前的纯RL算法，有3个游戏甚至超越了人类专业玩家的水平，图1就是其中5个游戏的训练截图。

**2.背景**

如果我们将游戏任务定义为AI在与一种环境 进行离散形式的互动，那么雅利达游戏模拟器的任务就是源源不断的处理动作、观察信息与奖励这样的组序列。游戏进行的每一步我们都称为一个时间步，这种时间步就像玩《俄罗斯方块》时方块从上至下的一层一层落下一样。在每个时间步AI都会从已知动作集合中选取一个动作，这个动作被传入模拟器内部，并触发游戏状态与游戏得分的变化，例如在《超级玛丽》中一次成功的跳跃会导致怪物死亡并增加金币得分。大多数情况下，游戏环境充满了随机性，并且AI并不允许观察游戏内部的变量，只能依靠对游戏画面进行截图，将图像作为像素阵列或向量 来处理，而游戏过程中获得的奖励（或惩罚） 一般来自于游戏得分的变化。良好的游戏得分来源于一系列数据观察与神操作，所以一次神操作的反馈通常要等到上千个时间步后才能体现出来，让AI学会理解这一点非常困难。

由于AI只能观察并处理当前图像，所以它对游戏的全过程实际上是感知混淆状态，很像是一个失意的人。因此很有必要为AI建立记忆机制，起码让它能记住一段时间的序列（），每一个元素都被赋予一个单独的数字用以表示不同的时间步，这样AI就能够依靠这些残破的记忆来进行学习，这种机制比较复杂但能有效利用有限的信息，所以也被称为“有限马尔可夫决策过程（finite MDP）”。MDP将无止尽随机的状态变成了有限序列，使得我们能够应用标准的强化学习方法。

AI在游戏中的目标可以被描述为与环境尽情的互动，根据状态合理选择动作从而最大化未来累计奖励，即从某一时间步开始直至游戏结束的奖励之和。由于AI被赋予了记忆，所以当我们让它玩完一遍游戏后，对于其中的一个时间步t，从t开始直至游戏结束时间步T之间的时间都算是“未来”，那么这种“未来累计奖励”可以写为 ，其中 分别是奖励折扣因子与单个时间步的奖励值。在强化学习中会为每个状态s作出的每个动作a进行打分，用以评价AI所采用策略的优劣程度，如果这个分值是最优的，那么构成整个游戏最优玩法的分值规律可以形成一个函数，称为最优动作值函数 ，也叫Q值函数。而决定这个函数是否最优的元素是AI所采用的策略命名为 ，这种策略是状态序列到动作选择的一种概率分布映射，综上所属就有 。

上面的最优动作值函数遵循着著名的贝尔曼方程（Bellman equation），所以上面的公式可以写成是一个能用于人工智能训练的迭代型公式:

，(公式1)

其中表示在s状态时的下一个状态与一个状态所选择的动作，用即将发生的未来事件来优化当前时间的选择。

通过公式1来优化每个Q值，直至其逼近最佳得分 是强化学习算法的核心思路，但对于像玩游戏这种随随便便过万的状态分布来说，这样的核心思路是远远不够的，因为计算Q值非常消耗资源。随着深度学习的发展，神经网络技术逐渐被人们捧红，因为它能够有效表示不同空间之间的映射并通过优化方法缩小估计值与目标之间的差距，于是人们开始尝试构造神经网络 来逼近 。“AI打开了FIFA，开始控制球员行动了，但它离最强的玩家还是差那么一点”，这种与最优目标的差距通常称为误差，大量的误差可以构成误差函数，也就是下面的公式：

，（公式2）

其中 就是那个目标， 是动作可能性的概率分布（behavior distribution）， 就是神经网络的权重参数，随着误差优化过程发生变化，想要优化误差可以使用下列梯度公式：

。（公式3）

相比上式的全梯度，更多人喜欢计算随机梯度来提升训练效果。如果权重参数是在每个时间步之后就进行更新，那么相当于针对单独采样进行行为估计，这就跟传统的Q-learning算法如出一辙了。

注意两个技术细节，首先上面描述的算法是无模型的（model-free），即AI不需要单独构造游戏环境 ，而是从其中进行采样；其次上述的算法是离线策略的（off-policy），即学习的时候遵循当前策略选择最优动作，玩的时候存在一个贪心算法， ，这会导致有一定概率AI不照着最优策略玩，而是选择探索更多的可能性。

**3. 相关研究**

在强化学习界最知名的成功案例当属TD-gammon，它是一个能玩西洋双陆棋（backgammon）游戏且达到超人类水平的AI程序。这个程序使用类似Q-learning的无模型强化学习算法，并构造了一个单隐含层的多层感知器来估计RL的值函数。

尽管如此，一些人想模仿TD-gammon来设计能以超人类水平玩其他棋类游戏（围棋、国际象棋等）的AI程序，但几乎都失败了。这就导致了像TD-gammon这样的AI仅适用于个别游戏，后来经研究人员分析，可能是西洋双陆棋游戏采用了骰子的缘故，骰子能使AI更容易的探索状态空间，并使值函数保持平滑。

而另一方面，在当时的相关研究显示Q-learning这样的无模型、离线策略、非线性的强化学习算法可能会导致Q网络不稳定，所以一段时间内人们都专注于使用线性函数进行值估计，以确保函数收敛。

直到最近使用深度学习的强化学习算法才得以复兴。深度神经网络被用于估计环境信息；受限玻尔兹曼机被用于估计值函数或策略；Q-learning不稳定的问题也被时序差分梯度技术（gradient temporal-difference）所解决。这些新方法的出现，使得非线性函数能够作为稳定的估值函数，也使得Q-learning搭配线性函数可以适用于有限变化的控制问题中。不过对于使用非线性函数来解决普通控制过程最优化这一问题还未有合适方法。

可能前人与我们的研究最接近的当属神经拟合Q-learning(NFQ)算法了。NFQ使用公式2计算误差，使用RPROP算法更新Q网络的权重参数，但是由于强化学习过程的数据量太大，NFQ采用批量方式进行训练，这样就导致了训练过程的数据分布问题与资源消耗问题，而我们使用随机梯度下降就能大大改善这些问题。NFQ首先使用深度自编码器来学习低阶特征（有外力介入），然后使用NFQ本身来学习所有数据表示，以此方法仅凭视觉数据就能胜任真实世界中的简单控制任务。相比之下，我们的方法则是完全端到端的（无需其他外力）。

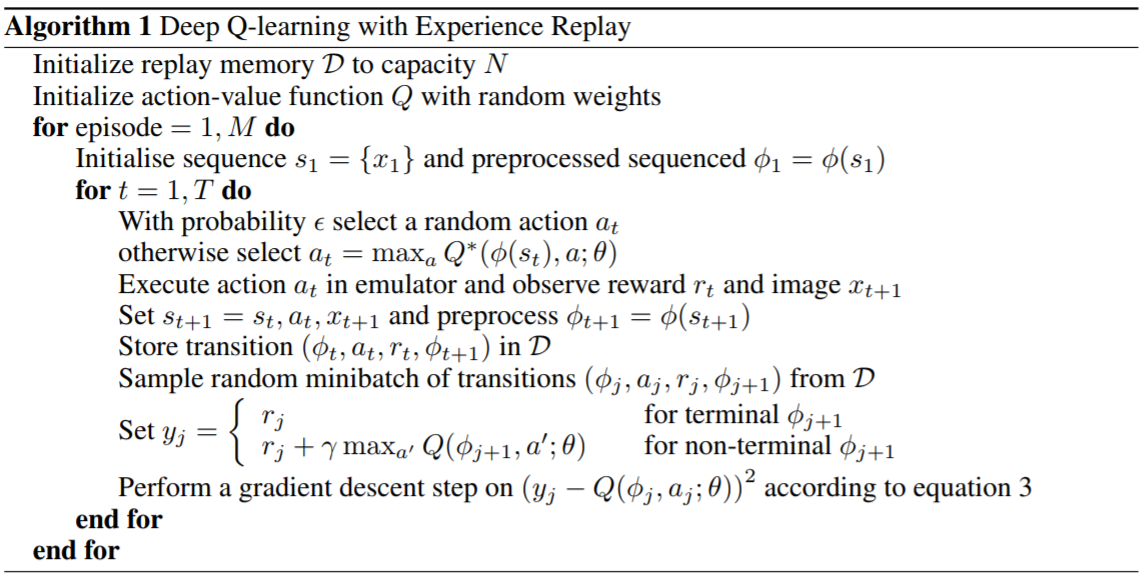
作为最早使用强化学习玩雅利达游戏的论文，引文[3]采用线性估值函数与视觉特征。之后便是引文[2]对其进行了改善，使用大量特征并配以tug-of-war hashing技术让特征能够随机分布在低维空间中。引文[8]描述的HyperNEAT进化结构能够让神经网络为每个雅利达游戏单独改进策略。当利用雅利达模拟器的重置功能反复训练确定的序列时，实验证明这些策略会开始利用几个游戏中的设计缺陷。

**4. 深度强化学习**

深度学习目前取得的成就大多在计算机视觉与语音识别领域，这些成就全部都依赖于海量的数据以供神经网络学习，一旦你拥有足够的高质量数据，那么深度学习所得到的数据表示远优于特征工程的结果。这样振奋人心的成果驱动着我们将深度学习与强化学习相结合，我们的目标就是将强化学习算法与深度神经网络连接起来，让AI在游玩过程中捕获视觉数据并直接交给神经网络进行处理，使用随机梯度来让训练保持高效。

Tesauro的TD-Gammon为我们开了个好头，这个架构从在线策略的采样经验序列中进行学习（不使用贪心算法），在双陆棋游戏中能够很好的更新神经网络参数，达到了20年来这个游戏人类玩家的最高水平，优秀的成绩源自20年计算机硬件的改进与深度神经网络对强化学习的技术扩展。

跟TD-Gammon及其他在线训练案例一样，我们使用了经验回放（experience replay）机制来存放AI的记忆，即针对每个时间步都记录一个 并将若干时间步存放进回放记忆库 （replay memory），这个记忆库里的内容甚至跨越多次游戏经历。当算法开始运行时，AI的经历被不断推入记忆库，接着被更新算法不断随机取出用于学习其中的信息。由于神经网络一旦运行起来就无法改变输入维度与长度，所以我们每次都给予AI取定长的记忆区间，而这整个算法就被称为Deep Q-learning了。



让我们来分析一下这个算法相比标准在线学习的Q-learning有什么优势吧。首先，每一个时间步的游戏经历都能够重复用于训练，这使得数据利用效率比较高；其次，直接对连续发生的场景进行顺序学习是非常低效的，而从记忆中随机采样则打破了这种连续性，能有效降低采样方差；再次，在线策略的数据生成与学习是通过完全一致的策略，这就会导致当前的训练参数会决定下一个数据采样，比如一辆自动驾驶汽车在一个路口左转了，这个信息影响了训练参数的变化，那么接下来汽车永远不会走向右边的路了，本次训练就会缺失右边那些路的信息，AI永远都不会得到一批既包含这个路口左转也包括右边那些路的数据。虽然使用离线策略方法或许能缓解上述问题，但使用经验回放技术则能够彻底解决这些问题。

实践这种算法时，你会发现我们仅仅将最后N个经历序列放入记忆库，而记忆库本身不会区分这些经历的重要性，随机采样过程也不会强调数据的重要性。然而玩游戏时无聊的行走与消灭一个敌人显然后者对提升游戏水平更加重要，所以设计更复杂的抽样策略是很有必要的，例如优先级扫描技术（prioritized sweeping）。

**4.1 预处理与模型结构**

雅利达模拟器给出的图像帧都是210x160像素大小的画面，包含128种颜色，这样的画面不利于快速计算，所以我们进行了一些预处理步骤。这些图像帧在进入系统前会先被转换为灰度值并下采样至110x84像素，最终被裁减为84x84大小的游戏区域以适应卷积网络，上面伪代码中的 就是裁减函数，每次对记忆中选出来的4帧进行裁减并打包为输入数据。

其实有许多利用神经网络的方法都能将Q值参数化，例如使用历史记忆与动作这一一对应的序列作为神经网络的输入，而这样做最大的缺陷就是要为每一个序列元素单独计算Q值，进而导致误差跟动作数量呈现线性关系。我们的方法则使用一种新的结构，其中每个可能的动作都有一个单独的输出单元，只需将状态表示作为神经网络的输入。输出只预测单个动作的Q值。这种结构的主要优点是能在给定状态下计算所有可能操作的Q值，但只需一次正向传播。

接下来我们看看进行过实验的7个游戏所以使用的AI架构：

* 神经网络的输入是84x84x4的图片，之后会经过 的裁减；
* 第一个卷积层是16个4x4的卷积核，步长为4，激活函数为整流非线性函数；
* 第二个卷积层是32个4x4的卷积核，步长为2，激活函数为整流非线性函数；
* 第三层是全连接层，包含256个神经元，激活函数为reLU;
* 输出层是全连接层，Softmax输出。

**5 实验过程**

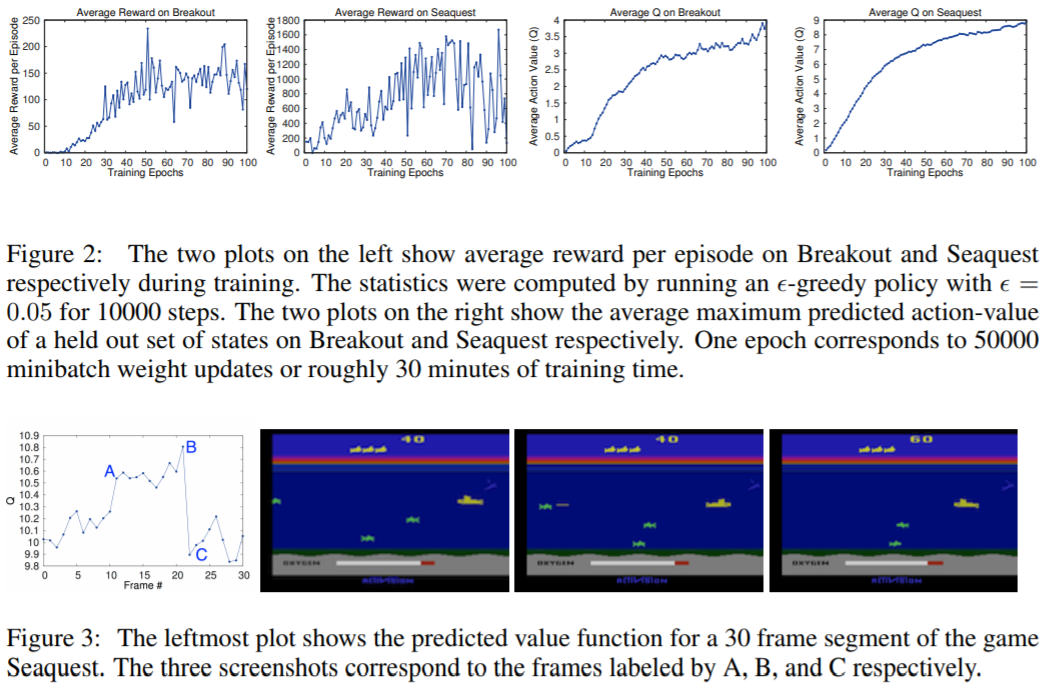
我们在以下7个游戏上完成了实验：Beam Rider、Breakout、Enduro、Pong、Q\*bert, Seaquest、Space Invaders，并且全部都采用统一的网络结构、学习算法与超参设定，结果证明了我们的方法具有泛化性能，不需要为游戏的输入特殊的信息。唯一作出变化的是根据每个游戏会设置不太相同的奖励方式，同时我们缩放了所有游戏的奖励规范，即惩罚性奖励统一采用-1，正面奖励统一采用1，非奖励时刻统一采用0。以这种方式缩放奖励会限制误差导数的规模，并使得AI在多个游戏中更易使用相同的学习方式。与此同时，AI也会因为这种缩放规则而无法区分不同的奖励程度。

训练过程中，我们使用RMSProp作为误差优化算法，数据批量值定义为32。AI行为采用了贪心策略，其主动探索概率在前100万个时间步内从1.0到0.1进行线性衰减，减小到0.1就固定住了。对于每个游戏我们都训练了1000万帧，使用最近期的100万帧作为回放记忆空间。

在以上结构的基础上，我们还采用了帧忽略技术，即AI并不是从每一帧进行观察与动作决策，而是会间隔K个帧。因为让游戏在模拟器中多进行几步要比让AI选择动作计算量少很多，采用这样的技术就可以让AI别那么忙，也能在不显著增加运行时间的条件下多进行约K倍的游戏内容。除了Space Invaders以外其他游戏全部采用k=4的间隔值，Space Invaders中的激光在k=4时会被忽略掉，所以只能采用k=3。

**5.1 训练与稳定**

对于监督训练方式，人们能够轻易的通过区分训练与验证数据集来及时检测训练效果，然而对于强化学习方式，训练期间想要有效评估AI状态可能具有挑战性。我们参考了引文[3]的方式，使用AI进行每一次游戏产生的累计奖励值或多次游戏的平均奖励。平均奖励指标噪声很严重，因为权重的微小变化可能会导致AI访问的状态分布发生重大变化。图2中最左边的两张就是Seaquest与Breakout这两个游戏的平均奖励指标，你会发现噪声非常严重，学习过程不够平稳。后两个使用Q值函数作为价值评估标准的结果就平稳的多，Q值反映出AI在给定状态下按学习好的策略究竟能拥有怎样的折扣奖励获取能力。我们在正式训练开始前预先收集了一些状态信息，追踪AI在这些状态计算出的最大Q值并求得平均值。图2最右边的两张图在其他5个游戏中也是能够呈现出来的，除了在训练期间看到Q值的平稳改善之外，没有遇到任何异常。以上结果表明尽管缺乏理论上的收敛保证依据，我们的方法仍然能够使用强化学习与随机梯度下降来训练大型神经网络。



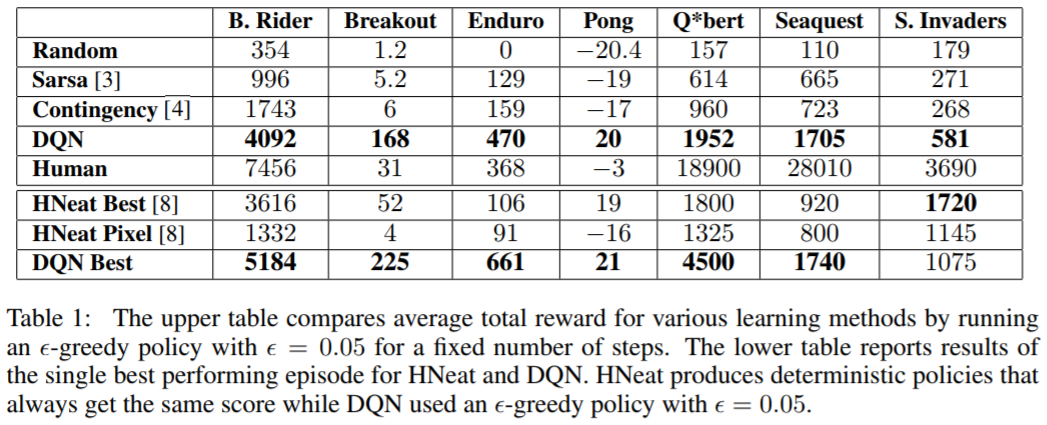
**5.2 可视化值函数**

图3 显示了Seaquest这个游戏的价值函数学习过程，以曲线图的形式进行了可视化。在曲线图中A点是敌船从屏幕左方出现的时候，我们发现曲线发生了跳跃；接着AI向敌船发射鱼雷，B点就是鱼雷击中敌船的时间，可以看到曲线又发生了明显变化；最终，学习曲线在C点开始变为常态，说明了AI能够有效学习游戏过程中的一些事件，并作出合理反应，同时价值函数也会随之波动。

**5.3 主要评估**

我们将相关数据与引文[3]与[4]中最好的实验结果进行了比较，下表中的标签“Sarsa”就是引文[3]采用Sarsa算法，在雅利达游戏环境中采用特征工程所进行的线性策略训练。“Contingency”则是以Sarsa算法为基础但使用部分屏幕数据进行学习的方法[4]。注意这两种方法都在一定程度上使用了先验知识来解决视觉问题，包括背景追踪与色彩处理。由于雅利达游戏属于128色游戏，即很多基本的物体形状都是由128色其中之一的纯色来上色的，所以通过处理色彩可以得到游戏画面中每类物体的二值表示。相较之下我们的AI仅通过图像本身来决定物体及其位置，技术上先进很多。

除了AI玩家以外，下表也包含了人类专家的成绩，人类玩家获得的奖励是取了游玩两小时后所赢得奖励的中值，注意我们所通报出来的人类得分高于引文[3]中Bellemare等人的得分。关于AI成绩我们沿用了引文[3]与[5]的平均方法，即计算含有贪心算法的学习策略（探索概率值为0.05）在一定时间步内所获奖励的平均值。



我们也在下表中通报了引文[8]的相关结果（HNeat Best与HNeat Pixel）。前者使用了单独设计的物体定位算法；后者挑选出特殊的8种颜色并为游戏画面建立了8通道的标签图，这个方法严重依赖某些固定游戏帧（游戏过程）。这两个方法学习出来的AI很难适应随机数据，所以指标是基于单次最高游戏得分评估出来的。相反我们的算法能够适应诸如贪心算法这样随机性的游戏决策过程，结果也证明了，除了Space Invaders这个游戏，我们的平均得分（第4行）与最大得分（最后一行）都获得了不错的成绩。

最终，我们的AI在Breakout，Enduro和Pong这3个游戏上获得了专业人类玩家的水平，在Beam Rider上实现了接近人类的表现，其他游戏则更具挑战性，因为它们需要神经网络找到拥有更长有效游玩时间的策略。

**6 结论**

综上所述，本文介绍了一种新的深度强化学习模型，展示了其仅使用原始像素作为输入来掌握Atari 2600游戏控制的策略构建与学习能力。我们的模型也是一种在线Q学习的变体，它结合了随机小批量更新和记忆回访技术，以此简化深度强化学习的训练过程。我们的方法在没有调整架构或超参的前提下，针对测试的7个游戏中有6个给出了当前最先进的结果。