Journal of System Simulation

Volume 36 | Issue 5 Article 8

5-15-2024

Gradient-based Deep Reinforcement Learning Interpretation Methods

Yuan Wang
School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China; Science and Technology on Information Systems Engineering Laboratory, Nanjing 210014, China, 2534764965@qq.com

Lin Xu
Science and Technology on Information Systems Engineering Laboratory, Nanjing 210014, China

Xiaoze Gong
PLA 63850 Troops, Baicheng 137001, China, 305118154@qq.com

Yongliang Zhang
Command and Control Engineering College, Army Engineering University of PLA, Nanjing 210007, China

See next page for additional authors

Gradient-based Deep Reinforcement Learning Interpretation Methods

Abstract

Abstract: The learning process and working mechanism of deep reinforcement learning methods such as DQN are not transparent, and their decision basis and reliability cannot be perceived, which makes the decisions made by the model highly questionable and greatly limits the application scenarios of deep reinforcement learning. To explain the decision-making mechanism of intelligent agents, this paper proposes a gradient based saliency map generation algorithm SMGG. It uses the gradient information of feature maps generated by high-level convolutional layers to calculate the importance of different feature maps. With the known structure and internal parameters of the model, starting from the last layer of the model, the weight of different feature maps relative to the saliency map is generated by calculating the gradient of feature maps; it classifies the importance of features in both positive and negative directions, and uses weights with positive influence to weight the features captured in the feature map, forming a positive interpretation of the current decision; it uses weights that have a negative impact on other categories to weight the features captured in the feature map, forming a reverse interpretation of the current decision. The saliency map of the decision is generated by the two together, and the basis for the intelligent agent's decision-making behavior is obtained. The effectiveness of this method has been demonstrated through experiments.

Keywords

DRL, saliency map, interpretability, agent, gradient

Authors

Yuan Wang, Lin Xu, Xiaoze Gong, Yongliang Zhang, and Yongli Wang

Recommended Citation

Wang Yuan, Xu Lin, Gong Xiaoze, et al. Gradient-based Deep Reinforcement Learning Interpretation Methods[J]. Journal of System Simulation, 2024, 36(5): 1130-1140.

第 36 卷第 5 期 系统仿真学报© Vol. 36 No. 5
2024 年 5 月 Journal of System Simulation May 2024

基于梯度的深度强化学习解释方法

王远[1,2]，徐琳[2]，宫小泽[3*]，张永亮[4]，王永利[1]
(1. 南京理工大学 计算机科学与工程学院，江苏 南京 210094；2. 信息系统工程重点实验室，江苏 南京 210014；
3. 63850部队，吉林 白城 137001；4. 陆军工程大学 指挥控制工程学院，江苏 南京 210007)

摘要：DQN等深度强化学习方法的学习过程与工作机制不透明，无法感知其决策依据与决策可靠性，使模型做出的决策饱受质疑，极大限制了深度强化学习的应用场景。为了解释智能体的决策机理，提出一种基于梯度的显著性图生成算法(saliency map generation algorithm based on gradient, SMGG)。*使用高层卷积层生成的特征图梯度信息计算不同特征图的重要性，在模型的结构和内部参数已知的情况下，从模型最后一层入手，通过对特征图梯度的计算，生成不同特征图相对于显著性图的权重；对特征重要性进行正向和负向分类，利用有正向影响的权值将特征图中捕获的特征进行加权，构成当前决策的正向解释；利用对其他类别有负向影响的权值将特征图中捕获的特征进行加权，构成当前决策的反向解释。*二者共同生成决策的显著性图，得出智能体决策行为的依据，实验证明了该方法的有效性。

关键词：深度强化学习；显著性图；可解释性；智能体；梯度
中图分类号：TP391.9 文献标志码：A 文章编号：1004-731X(2024)05-1130-11
DOI: 10.16182/j.issn1004731x.joss.22-1480
引用格式: 王远, 徐琳, 宫小泽, 等. 基于梯度的深度强化学习解释方法[J]. 系统仿真学报, 2024, 36(5): 1130-1140.
Reference format: Wang Yuan, Xu Lin, Gong Xiaoze, et al. Gradient-based Deep Reinforcement Learning Interpretation Methods[J]. Journal of System Simulation, 2024, 36(5): 1130-1140.

Gradient-based Deep Reinforcement Learning Interpretation Methods

Wang Yuan[1,2], *Xu Lin*[2], *Gong Xiaoze*[3*], *Zhang Yongliang*[4], *Wang Yongli*[1]
(1. School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China;
2. Science and Technology on Information Systems Engineering Laboratory, Nanjing 210014, China; 3. PLA 63850 Troops, Baicheng 137001, China;
4. Command and Control Engineering College, Army Engineering University of PLA, Nanjing 210007, China)

Abstract: The learning process and working mechanism of deep reinforcement learning methods such as DQN are not transparent, and their decision basis and reliability cannot be perceived, which makes the decisions made by the model highly questionable and greatly limits the application scenarios of deep reinforcement learning. To explain the decision-making mechanism of intelligent agents, this paper proposes a gradient based saliency map generation algorithm SMGG. *It uses the gradient information of feature maps generated by high-level convolutional layers to calculate the importance of different feature maps. With the known structure and internal parameters of the model, starting from the last layer of the model, the weight of different feature maps relative to the saliency map is generated by calculating the gradient of feature maps; it classifies the importance of features in both positive and negative directions, and uses weights with positive influence to weight the features captured in the feature map, forming a*

收稿日期：2022-12-11 修回日期：2023-03-21
基金项目：国家自然科学基金(61941113)；信息系统工程重点实验室开放基金(05202104)
第一作者：王远(1997-)，男，硕士生，研究方向为规则推理、强化学习。E-mail：2534764965@qq.com
通讯作者：宫小泽(1971-)，男，高工，硕士，研究方向为数据治理、弹药毁伤评估等。E-mail：305118154@qq.com

positive interpretation of the current decision; it uses weights that have a negative impact on other categories to weight the features captured in the feature map, forming a reverse interpretation of the current decision. The saliency map of the decision is generated by the two together, and the basis for the intelligent agent's decision-making behavior is obtained. The effectiveness of this method has been demonstrated through experiments.

Keywords: DRL; saliency map; interpretability; agent; gradient

0 引言

经典强化学习利用神经网络拟合价值函数或者策略函数，其可解释性低下严重限制了强化学习方法在复杂敏感场景的应用。将神经网络如何计算价值函数或者策略函数的依据进行可视化展示，给出了深度强化学习智能体决策最直观的解释。

针对神经网络的可视化解释，学者们引入了许多不同种类的方法，如显著性图[1]。显著性图凸显出了图像中重要的视觉特征，其凸显程度(即显著性分数)体现了图像不同区域对当前任务的重要程度。显著性图生成方法大概可以分为两类：基于扰动的方法和基于梯度的方法。现有的对深度强化学习的可视化研究也大都立足于基于扰动的方法进行展开[2-7]。

本文提出了一种基于梯度的显著性图生成算法 (saliency map generation algorithm based on gradient, SMGG)，生成显著性图，标识出重要特征，作为智能体决策的解释。

主要工作包括：①使用高层卷积层生成的特征图的梯度信息计算不同特征图的重要性。②对特征重要性进行正向和负向分类，利用有正向影响的权值将特征图中捕获的特征进行加权，构成当前决策的正向解释；利用对其他类别有负向影响的权值将特征图中捕获的特征进行加权，构成当前决策的反向解释。

1 相关工作

基于梯度的解释方法的核心思想是利用深度神经网络的反向传播机制，将模型的决策重要性信号从输出层神经元向前传播，利用捕获特征的中间层可视化输入样本的特征重要性。

文献[8]提出反向传播的方法，利用反向传播算法计算输出相对于输入图像像素点的梯度，识别输入图像中的重要部分，以此生成输入所对应的显著性图。文献[9]提出了反卷积的方法。文献[10]将反向传播方法与反卷积方法相结合，提出了导向反向传播方法。这3种方法基本思想是相同的，只是在处理ReLU过程中，各有不同的方法：反向传播中，输入大于0的位置信息得以保留；在反卷积中，输出大于0的位置的信息得以保留；导向反向传播方法则是反向传播和反卷积的结合，保留输入和输出都大于0的位置的信息，如图1所示。但这3种方法都存在一些问题：在神经元饱和时梯度为0，无法有效地表征特征重要性，并且这三种方法通过反向传播得到的特征重要性对类别不敏感，即没有类别区分性。此外，这些方法生成的显著性图存在很多肉眼可见的噪音，如图2所示。

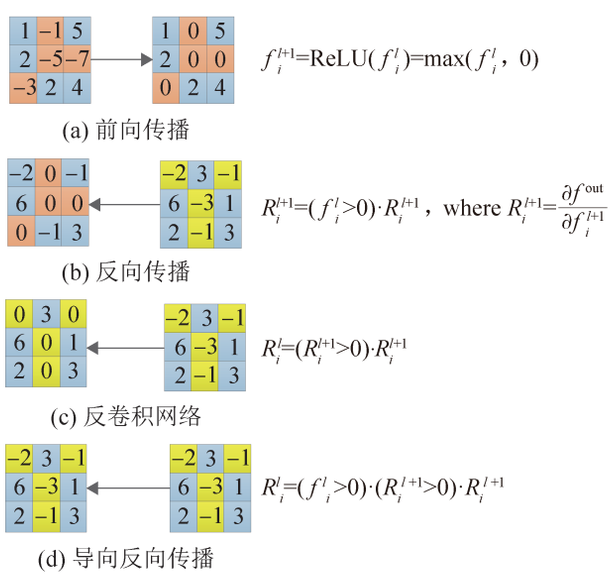
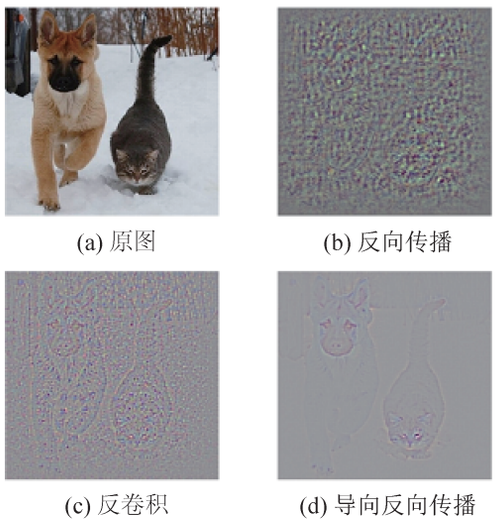


图1 不同传播方法对比

Fig. 1 Comparison of different propagation methods

DOI: 10.16182/j.issn1004731x.joss.22-1480



(a) 原图 (b) 反向传播

(c) 反卷积 (d) 导向反向传播

图2 不同传播方法的噪音问题

Fig. 2 Noise problem of different propagation methods

为了减少噪音的干扰，文献[11]通过引入噪音的手段来消除噪音，提出了一种基于平滑梯度的反向传播方法。向输入图像添加噪声，并对相似的样本进行采样，然后利用反向传播方法求解每个采样样本的决策显著性图，最后求得所有显著性图的平均值，以此消除反向传播等方法中存在的噪音。但是该方法并没有充足的理论依据，只能算一种消除噪音的技巧。

为解决朴素梯度中神经元饱和的问题，文献[12]提出了一种基于积分梯度的方法。该方法主张不只计算当前输入的梯度，而是通过计算某个在非饱和区的基准输入到当前输入的梯度的积分来代替当前输入的梯度，大大减少了梯度为0的情况。然而，这种方法在积分的时候可选的插值路径并不唯一，选择不同的插值路径时，求出的积分梯度结果也可能不同，导致此方法的有效性出现争议。

为了生成带有类判别性的显著性图，文献[13]提出了类别激活映射方法(class activation mapping, CAM)，该方法利用全局平均池化(global average pooling, GAP)操作获取特征向量，再和输出层进行全连接。GAP直接将特征图维度从$K\times W\times H$转化成$K\times 1\times 1$，即对每一层特征图里面的所有像素值求平均。CAM通过计算最后一个卷积层形成的特征图的加权平均，得到某一个特定类别所对应的类别激活图，该图可以突显出卷积神经网络用来识别该类别的重要特征区域。最后，通过显著性图的形式可视化类别激活图，得到最终的解释结果。

然而，CAM方法只适用于输出前自带GAP操作的网络结构，否则，就需要用户修改网络并重新训练，其应用范围十分受限。文献[14]对CAM方法做出改进，提出了一种梯度加权的类别激活映射方法(Grad CAM)。对于一个输入图像，首先，Grad CAM会计算目标类别相对于最后一组特征图中每一个像素的梯度，并使用特征图中像素梯度的平均值，表征该特征图的权重，然后，使用这些特征图权重进行加权求和，以此生成梯度加权的类别激活图，用于定位输入图像中具有类别区分性的重要区域。与CAM相比，Grad CAM可适用非GAP连接的网络结构，无需修改网络结构或者重新训练模型，并且Grad CAM可以利用任意一层特征图来生成显著性图。但是该方法只利用了所预测的类别来生成显著性图，忽略了其他类别的相关信息对决策的反向解释。

为了使解释结果更直观，增强鲁棒性，文献[15]提出了一种新的类别激活映射方法(Eigen-CAM)。通过可视化卷积层学习表示的主成分来增强对CNN预测的解释。Eigen-CAM直观，使用方便，计算效率高，不需要模型进行正确分类，可以与所有CNN模型协同工作，无需修改图层或重新训练模型。但是该方法同样没有考虑反向解释对于决策的影响。

2 基于梯度的深度强化学习解释方法

2.1 设计思路

经过多层卷积和池化的不断编码和抽象，卷

积神经网络的深层蕴含了丰富的语义信息[16-17]。高层到模型的输出所经过的神经元比低层更少，所经历的变化也更少，高层网络的特征较低层而言，对输出有着更直接和紧密的联系，充分利用高层的语义信息对模型输出做出解释，将更加便利和直观。

假设某个深度强化学习网络模型的输出为动作评分(或经过Softmax层处理变为采取这种动作的概率)，且每个动作的评分线性依赖输入图片中的每个像素或者特征($y=\omega x+b$)，则输出的动作评分对输入x的梯度$\boldsymbol{\omega}=\partial y/\partial x$能够直接用来量化每个像素对每个动作评分的重要程度。

本文方法聚焦最后一层的卷积层生成的特征图进行智能体的解释工作。求得每个特征图中的像素点对不同动作评分的梯度，若梯度较大，那么该像素点的微小变化都将引起动作评分的较大变化，表明该像素点对动作的选择十分重要。SMGG将重要性以显著性图的形式可视化，再映射到原图大小的尺寸，来凸显出输入对动作选择的影响，解释智能体行为。

以AC算法[18]中的Actor网络为例，描述了基于梯度的显著性图生成算法的整体流程，主要由以下几个部分构成。

(1) 计算特征图权重：计算每一个特征图每一个像素对每一动作评分的梯度，利用全局平均池化的思想，生成不同特征图对每一动作评分的贡献权重。

(2) 计算显著性：从贡献的“积极−消极”方向出发，将对当前最优动作有积极意义的特征图加权作为智能体当前决策的正向解释；将对其他动作有消极影响的特征图加权取交集，作为智能体当前决策的反向解释；二者共同参与显著性的计算。

(3) 绘制显著性图：将特征图大小的显著性图以伪色彩图的形式表达，并上采样到输入图像尺寸，绘制到输入图像上。

2.2 基于梯度的显著性图生成算法

卷积神经网络高层的特征图捕获了富含高级语义的空间特征[19]，本文利用这些层中的神经元在图像中寻找特定语义的信息来生成显著性图。

给定一个智能体N，其动作空间为A_N，状态空间为S_N，动作价值函数定义为$Q(s,a)$，表示处在s状态下执行a动作的价值，经过AC算法的训练，智能体可以在Atari游戏中有出色的表现。

本文主要关注网络最后一层卷积形成的特征图以及Softmax之前的动作评分$S(a|s)$来计算特征图权重和显著性，利用有正向影响的权值将特征图中捕获的特征进行加权，构成当前决策的正向解释；利用其中对其他类别有负向影响的权值将特征图中捕获的特征进行加权，构成当前决策的反向解释，并借助智能体N进行基于梯度的显著性图生成算法的说明与实验。

算法1　基于梯度的显著性图生成算法

输入：原始图像I_{raw}

输出：带显著性图的图像I_{sm}

1) 初始化M_a^p, M_a^n

2) $w\leftarrow I_{\text{raw}}$的宽度, $h\leftarrow I_{\text{raw}}$的高度

3) $F,S\leftarrow Net(I_{\text{raw}})$ /* 获取特征图与活动打分*/

4) for $a'\neq a$ do

5) 初始化$M_{-a'}^p$

6) for k=0 to dimension(F) do

7) $\omega_k^{-a'}\leftarrow -\text{grad}(S(a'|s), Fk)$ /*计算负权重*/

8) $M_{-a'}^p\leftarrow M_{-a'}^p+\text{ReLU}(\omega_k^{-a'}\cdot F^k)$ /*更新负显著性值*/

9) $M_a^n\leftarrow M_a^n\odot M_{-a'}^p$

10) for k=0 to dimension(F) do

11) $\omega_k^a\leftarrow \text{grad}(S(a|s), Fk)$ /*计算权重*/

12) $M_a^p\leftarrow M_a^p+\text{ReLU}(\omega_k^a\cdot F^k)$ /*更新显著性值*/

13) $M_a\leftarrow$Normalized($M_a^p+M_a^n$)

14) $M_a\leftarrow$UpSample(M_a,h,w)

15) $I_{\text{sm}}=I_{\text{raw}}+M_a$

16) return I_{sm}

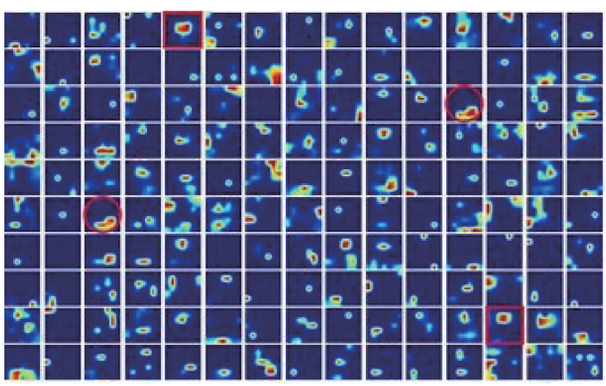
2.3 计算特征图权重

卷积神经网络最后一层卷积层形成的特征图

第 36 卷第 5 期
2024 年 5 月
系统仿真学报
Journal of System Simulation
Vol. 36 No. 5
May 2024

蕴含了足够多的信息，以VGG16模型为例，当输入图3(a)后，模型的最后一层卷积层形成的部分特征图如图3(b)所示。可以看到，红色方形框出的特征图高亮了原始图像中“狗”的区域，红色圆形框出的特征图高亮了原始图像中“猫”的区域。

(a) 猫和狗



(b) VGG16模型部分特征图可视化

图3　卷积神经网络特征图蕴示例
Fig. 3　Example of convolutional neural network feature map

如图4所示，AC算法的Actor网络输入Atari的游戏帧作为状态，经过几层卷积、池化，最后全连接到动作空间大小的向量，得到不同动作的评分，再经过Softmax得到采取不同动作的概率，智能体选择最大概率的动作，予以执行。

假定最后一层特征图F，其通道数为K，特征图F^k表示序号为k的单张特征图。最终动作评分输出的向量维度为A_N，即动作空间为A_N。每个特征图对动作评分的输出各有不同的贡献，结合全局平均池化，令ω_k^a表示特征图F^k对动作a的评分$S(a|s)$的贡献参数。若不做特殊说明，后文提到的动作评分皆为网络Softmax之前的数值。

将不同的特征图加权，即可得到对于某一特定动作的显著性图[14]，并且采用不同特征图对$S(a|s)$的贡献线性加权。

$$M_a = \sum_k \omega_k^a F^k \tag{1}$$

为了易于理解虚拟的参数ω_k^a，假设在输出动作评分之前加一层全局平均池化层，修改后的模型的网络结构如图5所示。

此时，倒数第二层神经元数目为K，全连接到有A_N神经元的输出层，ω_k^a为网络模型最后一层全连接的参数。令G^k表示特征图经过平均池化之后的数值，则有

$$G^k = \frac{1}{\sum_i \sum_j l} \sum_i \sum_j F_{i,j}^k \tag{2}$$

式中：l为原始图像的所有像素；$F_{i,j}^k$为坐标为(i,j)的像素数值，即对特征图F^k遍历高度和宽度，取全部像素点的平均值。

图4　DRL模型
Fig. 4　DRL model

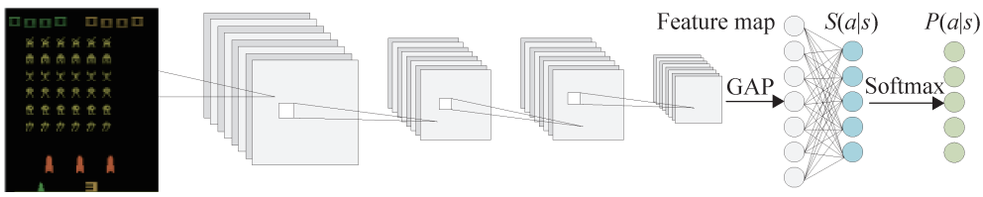


图5　嵌入全局平均池化的设想图
Fig. 5　Imaginary diagram of embedding global average pooling

此时，网络输出动作评分为

$$S(a|s)=\sum_k \omega_k^a G^k \tag{3}$$

参数ω_k^a并不存在于原网络结构，因此需要求解该参数，方可用于显著性的计算。

将式(2)带入式(3)可得

$$S(a|s)=\sum_k \omega_k^a \frac{1}{\sum_i\sum_j l}\sum_i\sum_j F_{i,j}^k \tag{4}$$

假定$Z=\sum_i\sum_j l$，此时参数ω_k^a可以进一步表示为

$$\omega_k^a=\frac{\partial S(a|s)}{\partial G^k}=Z\frac{\partial S(a|s)}{\partial F_{i,j}^k} \tag{5}$$

采用原网络结构，特征图F^k的权重可以由其内任意像素对动作评分$S(a|s)$的梯度表示，为了不失一般性，对每一个像素点求和：

$$\sum_i\sum_j \omega_k^a=\sum_i\sum_j Z\frac{\partial S(a|s)}{\partial F_{i,j}^k} \tag{6}$$

参数ω_k^a与特征图高度和宽度无关，故$\sum_i\sum_j \omega_k^a=Z\omega_k^a$，式(6)可进一步转化为

$$\omega_k^a=\sum_i\sum_j Z\frac{\partial S(a|s)}{\partial F_{i,j}^k} \tag{7}$$

在计算权重参数ω_k^a的时候使用的动作评分为网络 Softmax 之前的数值$S(a|s)$，而非 Softmax 之后的评分(经过 Softmax 后各评分被归一化，归一化后的评分也可以理解为采取不同动作的概率)，二者的区别如下。

假设经过 Softmax 之后，模型输出为$P(a|s)$，与$S(a|s)$的关系为

$$P(a|s)=\frac{e^{S(a|s)}}{\sum_{a'\in A_N} e^{S(a'|s)}} \tag{8}$$

则动作概率$P(a|s)$对动作评分$S(a|s)$的偏导为：

$$\frac{\partial P(a|s)}{\partial S(a|s)}=\frac{e^{s(a|s)}\sum_{a'\in A_N} e^{s(a'|s)}-\left(e^{s(a|s)}\right)^2}{\left(\sum_{a'\in A_N} e^{s(a'|s)}\right)^2}=$$

$$\frac{e^{s(a|s)}}{\sum_{a'\in A_N} e^{s(a'|s)}}\left(1-\frac{e^{s(a|s)}}{\sum_{a'\in A_N} e^{s(a'|s)}}\right)=$$

$$P\left(a|s\right)\left(1-P\left(a|s\right)\right) \tag{9}$$

此时，经过 Softmax 之后的融合权重可由链式法则求得：

$$\alpha_k^a=\sum_i\sum_j \frac{\partial S(a|s)}{\partial F_{i,j}^k}\frac{\partial P(a|s)}{\partial S(a|s)}=$$

$$\sum_i\sum_j \frac{\partial S(a|s)}{\partial F_{i,j}^k}P(a|s)(1-P(a|s)) \tag{10}$$

可以看出，二者的梯度差异是动作概率$P(a|s)$对动作评分$S(a|s)$的偏导$P(a|s)(1-P(a|s))$，而对于已经训练好的网络，该项为定值。因为训练的充分好的网络，预测输出$P(a|s)$的值是趋向于 1 的，所以$P(a|s)(1-P(a|s))$的值趋向于 0，因此存在丢失精度的风险，所以建议使用不经过 Softmax 的动作评分$S(a|s)$来计算权重。

2.4 计算显著性

特征图经过线性加权可以得出当前动作的显著性图，如式(1)所示。

相同的网络模型与网络输入，所形成的特征图也是相同的，但经过不同的权重，线性组合出

第 36 卷第 5 期 系统仿真学报 Vol. 36 No. 5
2024 年 5 月 Journal of System Simulation May 2024

的显著性图却是千差万别的，甚至可以凸显出完全不同的区域。例如，在图3中高层的特征图不止包含了“猫”的相关特征，也包含了“狗”的相关特征，甚至，有的特征图还高亮了“柜子”和“窗户”的位置。若加以不同的权重线性组合，可以呈现出不同的结果。这表明显著性图不同的线性组合结果是由权重参数的不同导致的，数值大的参数表征该特征图对结果有较大的贡献，反之，则贡献较小。此外，特征图的权重参数ω_k^a的正负性表征了特征图对动作评分的正向作用和反向作用。因此，本文从特征图的线性组合中只过滤出对所关注的动作评分有积极作用的特征。

定义M_a^p是正向解释“为什么会采取动作a?”的相应显著性图：

$$M_a^p=\mathrm{ReLU}\left(\sum_k \omega_k^a F^k\right) \tag{11}$$

权重参数ω_k^a若为负值，表征特征图F^k对动作a的评分起反向作用。为了捕获这种反向作用，定义负面权重ω_k^{-a}为特征图F^k对动作a的评分$S(a|s)$的负面贡献参数：

$$\omega_k^{-a}=-\sum_i\sum_j \frac{\partial S(a|s)}{\partial F^k}=-\omega_k^a \tag{12}$$

负面影响贡献参数中也会有负值，这表示对负面影响有负面影响，即起正向作用。

定义M_{-a}^p为解释“为什么不会采取动作a?”的相应显著性图

$$M_{-a}^p=\mathrm{ReLU}\left(\sum_k \omega_k^{-a} F^k\right) \tag{13}$$

特征图对动作评分的负面影响对于解释智能体采取动作a来说并无帮助，但是，若采用逆向思维来利用反向作用，却可以作为其他动作的解释。

假设当前任务是解释为什么模型认为图片中存在“猫”，即高亮出“猫”所在的区域。图3(a)中主要含有“猫”“狗”“窗户”“柜子”等物品，则对于“狗”来说，负面显著性图中不会高亮“狗”所在的区域，而是突出显示了“非狗”所在的区域(可能含有“猫”的区域)。同理对于其他“非猫”来说，负面显著性图中都或多或少包含了“猫”附近的区域，这些区域取交集就可以得到“猫”附近的区域了。

因此，定义M_a^n是“为什么会采取动作a?”的反向解释的相应显著性图，即筛选出“为什么不采取非a动作”的交集：

$$M_a^n=\bigcap\nolimits_{a\neq a'}\mathrm{ReLU}\left(\sum_k \omega_k^{-a'} F^k\right) \tag{14}$$

最终，综合考量两种方向的解释，改写显著性图的计算公式：

$$M_a=M_a^p+M_a^n=\mathrm{ReLU}\left(\sum_k \omega_k^a F^k\right)+\bigcap\nolimits_{a\neq a'}\mathrm{ReLU}\left(\sum_k \omega_k^{-a'} F^k\right) \tag{15}$$

将加权特征图形成的显著性图进行归一化得到灰度图像，采用色度图的Colormap_Jet模式产生伪彩色图像。上采样到输入图像大小，然后与输入图像进行叠加。

3 实验验证

3.1 实验环境与配置

Atari 2600是Atari公司于1977年推出的一款视频游戏机。这款游戏机包括Breakout、Ms. Pacman和Space Invaders等热门游戏。

街机学习环境(arcade learning fnvironment，ALE)是一个建立在Atari 2600仿真器之上的简单框架，它允许用户通过接收操纵杆动作、发送屏幕或者RAM信息、模拟平台的方式来与Atari 2600交互。ALE提供了一个游戏处理层，它通过标记累积得分、游戏是否已经结束，可以将每个游戏转化成一个标准的强化学习问题。每个观察为单个游戏屏幕帧(一个宽160像素，高210像素的二维数组)。总的动作空间包含了18个离散动作，通过操纵杆控制器来定义。一个具体的游戏动作空间则由游戏处理层指定。

当环境运行时，仿真器会每秒生成60帧，最高速度的仿真可以达到6 000帧/s。在每个时间步长上的奖励通过帧与帧之间的得分来指定。一个回合会在“重置”命令后的第一帧处开始，在游戏结束时终止。

OpenAI Gym是一个强化学习领域常用的工具包，提供了一套多样化的环境，从简单到困难，涉及许多不同类型的数据，包括经典的控制环境、2D和3D的机器人以及ALE环境。对其中的Atari游戏环境在ALE的基础上进行了一些改动。

在OpenAI Gym 中，每个游戏都有一些变体，通过它们的后缀来区分。通过这些变体，用户可以自由地配置跳帧和粘滞动作。其中，跳帧是一种使用第k帧的技术，智能体只在每k帧做一次动作，其他帧会默认执行与上一次相同的操作。粘滞动作是一种在没有智能体控制的情况下设置重复动作的技术，设置重复动作的概率遵循其中的概率参数p。跳帧和粘滞动作的配合使用，使确定性的Atari 2600环境增加了随机性。

Pong游戏环境有6种变体，如表1所示。其中，跳帧2~4表示k从2，3，4中随机选择。此外，还有RAM环境，例如，Pong-ram-v0，此时，智能体观察的是机器的RAM信息，而不是视觉输入。

表1　游戏Pong的6种变体
Table 1　Six variants of game Pong

名称	跳帧k	重复动作概率p
Pong-v0	2~4	0.25
Pong-v4	2~4	0
PongDeterministic-v0	4	0.25
PongDeterministic-v4	4	0
PongNoFrameskip-v0	1	0.25
PongNoFrameskip-v4	1	0

由于v0版本的环境中，智能体会有25%的概率执行上一个动作，并非完全执行显式设置的动作；在v4版本的环境中，智能体完全按照显式设置的动作执行。为了保证所见动作即智能体所得动作，Atari 2600实验环境全部采用NoFrameskip-v4版本，以便从连续帧中采样，结合显著性图进行智能体决策的解释。

本文使用了AC算法训练好的智能体和3款Atari游戏进行对比实验。

(1) Breakout是一款弹射游戏，玩家左右控制底部的木板(智能体)反弹小球。小球在击中砖块后，砖块消失，小球反弹且获得积分，如果木板没接住小球则玩家失去一条生命，失去5条生命则游戏结束。

(2) Pong是一款模拟乒乓球的运动游戏，玩家通过在屏幕左侧垂直移动木板(智能体)击打小球，与计算机控制的对手比赛。当一方未能将球传回时，另一方即可获得积分，先达到21分则获胜，游戏结束。

(3) Space Invaders是一款固定射击游戏，玩家通过在底部水平移动激光炮(智能体)并开火来消灭外星人。当外星人从屏幕顶部向底部前进，它们会水平来回移动并向下开炮，玩家用激光炮射杀外星人来获得积分，如果被外星人击中则失去1条生命。失去3条生命则游戏结束。同时，激光炮会受到几个固定防御掩体的保护，但这些掩体可以被外星人和玩家的炮弹摧毁。

软件实验环境为Python3.6.11，Gym0.9.3，PyTorch0.4.1，NumPy1.19.5，SciPy1.2.0，Matplotlib2.2.2。硬件实验环境为处理器Intel (r) Xeon (R) GOLD 6130 CPU，内核128，显卡Tesla V100 15，内存125 GB，显存31 GB。

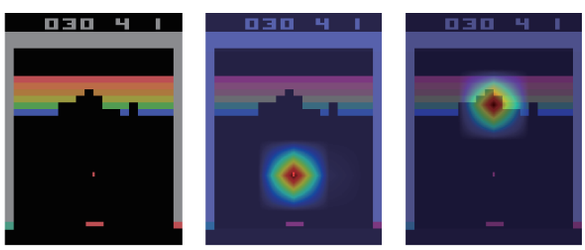
3.2 实验设计与结果分析

目前没有基于梯度的显著性图像生成算法来解释智能体的决策，因此，本文实验主要使用AC算法训练好的智能体与基于扰动的方法GPV (Gupta P, Puri N, Verna S, et al) [4] 和PBSM (perturbation-based saliency methods)[2]进行对比，验证本文方法的可用性。

DOI: 10.16182/j.issn1004731x.joss.22-1480

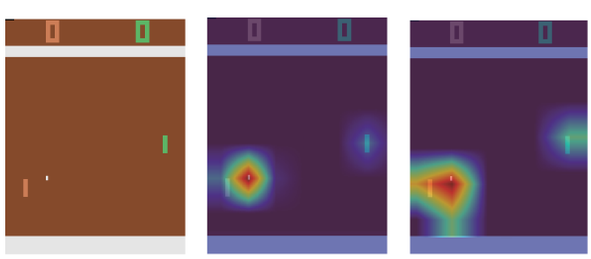
第 36 卷第 5 期
2024 年 5 月
系统仿真学报
Journal of System Simulation
Vol. 36 No. 5
May 2024

在智能体的学习过程中，其策略会逐渐优化，一些早期的策略最终会被摒弃，取而代之的是更好的策略。为了展示智能体变“智能”的过程，本文通过在训练过程中保存2个模型，并用显著性方法将它们可视化来探讨这个问题：第1个模型使用4 000万帧训练，称为Better版本；第2个智能体使用2 000万帧进行训练，称为Normal版本。因为Pong游戏相对简单，2 000万帧已经可以取得非常好的策略，所以，它的Normal版本只使用了100万帧。对于这3种游戏，用SMGG对随机采样的状态进行显著性绘制，如图6~8所示。



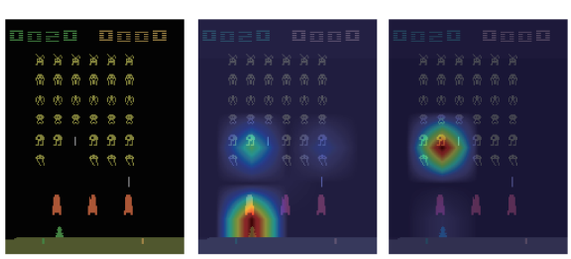
(a) Original Image (b) Normal (c) Better

图6 游戏Breakout智能体的策略学习
Fig. 6 Policy learning of game Breakout agent



(a) Original Image (b) Normal (c) Better

图7 游戏Pong智能体的策略学习
Fig. 7 Policy learning of game Pong agent



(a) Original Image (b) Normal (c) Better

图8 游戏Space Invaders智能体的策略学习
Fig. 8 Policy learning of game Space Invaders agent

图6显示，当小球被击飞的时候，Breakout中的Normal智能体的注意力是集中在小球上的，而Better版本的智能体特别关注了小球瞄准的隧道位置。

图7中，Pong的2个智能体的表现差别并不大，可能是因为这个游戏过于简单，最优策略很快就可以习得。小球被敌人击回的时候，Normal智能体并不会太多关注自身位置，它需要小球距离自己足够近的时候，才能判断出向哪个方向移动，而Better版本则在很远距离的时候，就可以判断出小球的落点位置，因此结合自身位置，很早就可以判断出移动的方向。

图8中，早期智能体策略相对保守，专注于躲避敌方攻击，因此注意力主要集中在宇宙飞船前面的掩体。随着训练的进行，智能体的策略倾向于攻击敌人，这样可以获得更多的得分。

本文利用SMGG、GPV、PBSM对同一智能体进行显著性绘制，采用人工标注的方法以衡量所提方法对智能体行为的解释能力。

本文从2个角度对3种方法生成的500帧显著图进行统计。智能体应该注意的位置高亮显示，不应该注意的位置不高亮显示。计算每种方法的准确率，召回率和F1值(F-Measure)，其结果如表2所示。

表2 指标结果比较
Table 2 Comparison of indicator results

方法	准确率	召回率	F1值
SMGG	0.864	0.958	0.908
GPV	0.749	0.979	0.849
PBSM	0.671	0.650	0.661

PBSM算法是较早提出的可解释方法，性能较差，因此，本文主要与GPV算法进行对比：SMGG方法的准确率是高于GPV算法的，但召回率却是略低于PGV算法的，这说明SMGG标识显著性的粒度大，定位性差，会遗漏非核心特征，但效率高，不敏感，噪音少，适用于了解模型结构的白盒情景。GPV标识的显著性粒度小、定位

性好、不会遗落每一个有用特征，但效率低、较敏感、有些许噪音，更适用于黑盒场景。二者并非存在绝对的优劣，各有所长。

4 结论

本文研究面向智能体决策的深度强化学习可解释性方法，试图让人们了解智能体决策背后的依据，对智能体行为提供解释。提出了一种基于梯度的显著性图生成算法来解释智能体的决策。当面对一个训练好的模型，模型结构已知的情况下，从模型最后一层入手，通过对特征图梯度的计算，生成不同特征图对显著性图的权重，利用有正向影响的权值将特征图中捕获的特征进行加权，构成当前决策的正向解释；利用对其他类别有负向影响的权值将特征图中捕获的特征进行加权，构成当前决策的反向解释。二者共同生成决策的可视化图像，得出智能体决策行为的依据。实验证明了方法的有效性，并与基于扰动的方法做出对比，总结了二者的优劣。未来工作拟采用积分梯度的解决神经元饱和问题。

参考文献：

[1] Itti L, Koch C, Niebur E. A Model of Saliency-based Visual Attention for Rapid Scene Analysis[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998, 20(11): 1254-1259.

[2] Greydanus S, Koul A, Dodge J, et al. Visualizing and Understanding Atari Agents[C]//Proceedings of the 35th International Conference on Machine Learning. Chia Laguna Resort, Sardinia, Italy: PMLR, 2018: 1792-1801.

[3] Iyer R, Li Yuezhang, Li Huao, et al. Transparency and Explanation in Deep Reinforcement Learning Neural Networks[C]//Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society. New York, NY, USA: Association for Computing Machinery, 2018: 144-150.

[4] Nikaash Puri, Sukriti Verma, Piyush Gupta, et al. Explain Your Move: Understanding Agent Actions Using Specific and Relevant Feature Attribution[EB/OL]. (2020-04-03) [2023-12-03]. https://arxiv.org/abs/1912.12191.

[5] Karim M M, Li Yu, Qin Ruwen. Toward Explainable Artificial Intelligence for Early Anticipation of Traffic Accidents[J]. Transportation Research Record, 2022, 2676(6): 743-755.

[6] Hyun Yoo, Soyoung Han, Kyungyong Chung. Diagnosis Support Model of Cardiomegaly Based on CNN Using ResNet and Explainable Feature Map[J]. IEEE Access, 2021, 9: 55802-55813.

[7] Sun K H, Huh H, Tama B A, et al. Vision-based Fault Diagnostics Using Explainable Deep Learning with Class Activation Maps[J]. IEEE Access, 2020, 8: 129169-129179.

[8] Simonyan K, Vedaldi A, Zisserman A. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps[EB/OL]. (2014-04-19) [2023-12-09]. https://arxiv. org/abs/1312.6034.

[9] Zeiler M D, Fergus R. Visualizing and Understanding Convolutional Networks[C]//Computer Vision-ECCV 2014. Cham: Springer International Publishing, 2014: 818-833.

[10] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, et al. Striving for Simplicity: The All Convolutional Net[J]. (2015-04-13) [2023-12-11]. https://arxiv.org/abs/1412.6806.

[11] Smilkov D, Thorat N, Kim B, et al. SmoothGrad: Removing Noise by Adding Noise[EB/OL]. (2017-06-12) [2023-12-13]. https://arxiv.org/abs/1706.03825.

[12] Sundararajan M, Taly A, Yan Qiqi. Gradients of Counterfactuals[J]. (2016-12-15) [2023-12-18]. https://arxiv.org/abs/1611.02639.

[13] Zhou Bolei, Khosla A, Lapedriza A, et al. Learning Deep Features for Discriminative Localization[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA: IEEE, 2016: 2921-2929.

[14] Selvaraju R R, Cogswell M, Das A, et al. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization[C]//2017 IEEE International Conference on Computer Vision (ICCV). Piscataway, NJ, USA: IEEE, 2017: 618-626.

[15] Bany Muhammad M, Yeasin M. Eigen-CAM: Visual Explanations for Deep Convolutional Neural Networks [J]. SN Computer Science, 2021, 2(1): 47.

[16] Yoshua Bengio, Aaron Courville, Pascal Vincent. Representation Learning: A Review and New Perspectives[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(8): 1798-1828.

[17] Mahendran A, Vedaldi A. Visualizing Deep Convolutional Neural Networks Using Natural Pre-images[J]. International Journal of Computer Vision, 2016, 120(3): 233-255.

DOI: 10.16182/j.issn1004731x.joss.22-1480

第 36 卷第 5 期 系统仿真学报 Vol. 36 No. 5
2024 年 5 月 Journal of System Simulation May 2024

[18] Mnih V, Badia A P, Mirza M, et al. Asynchronous Methods for Deep Reinforcement Learning[C]// Proceedings of the 33rd International Conference on Machine Learning. Chia Laguna Resort, Sardinia, Italy: PMLR, 2016: 1928-1937.

[19] 赵佳琦, 张迪, 周勇, 等. 基于深度强化学习的遥感图像可解释目标检测方法[J]. 模式识别与人工智能, 2021, 34(9): 777-786.
Zhao Jiaqi, Zhang Di, Zhou Yong, et al. Interpretable Object Detection Method for Remote Sensing Image Based on Deep Reinforcement Learning[J]. Pattern Recognition and Artificial Intelligence, 2021, 34(9): 777-786.

