[DGN] Graph Convolutional Reinforcement Learning for Multi-Agent Cooperation

- 摘要

MA 环境是动态的，因此相互作用难以表达。本文使用图卷积强化学习以解决该问题。图卷积方法适用于以图像为底层展示多智能体环境，卷积核可以抓住不同 agent 之间关系的表达。潜在特征通过卷积层一层层学习如何协作，而时间关系正则化的一致性又促进了协作。

- 介绍

MARL with Communication 也需要多智能体互相协作。文中方法是将 MA 环境建成一个图模型，每一个 agent 都是一个结点，将局部观测定义为结点特征。然后我们将卷积应用到 agent 图表中。使用多头注意力方法作为卷积核，图卷积就可以从邻居结点中提取特征，使得可以表达结点间的关系。而潜在特征通过卷积层的可视野一层层学习合作策略，此外，关系表达通过时序正则化进一步促进发展一致性合作策略。

图卷积强化学习又称 DGN，是基于 DQN 和端到端训练方法的。DGN 在智能体之间共享权重，易于拓展。DGN 通过关系核(relation kernel)对相互作用进行抽象，而关系核(relation kernel)通过卷积对潜在特征进行提取，同时通过时序正则化促进一致性合作。

与其他权重共享方法不同的是，DGN 允许考虑 agent 的共有可视野以优化策略，以此促进合作。与特征输入顺序无关的关系核(relation kernel)可以有效抓取智能体之间的协作，并将关系表达进行抽象以改善协作。

时序正则化是在连续的时间序列中最小化关系表达的 KL 散度，以促进在多智能体的动态环境中进行长期，一致的合作。

- 相关工作

MARL

MADDPG 和 COMA 都是使用局部奖励和共享奖励的 AC 模型，一个中心化的 critic 使用所有 agent 的观测和动作，使得难以扩展。PS-TRPO 解决了以前 MARL 认为难以解决的问题，同时共享策略参数促进了合作，但是在 agents 之间不共享信息的时候，agents 之间合作受限。VP 价值传播是网络化 MARL 的方法，它使用了 softmax 时序一致对值和策略进行更新。

图卷积

图卷积网络(GCN)以能总结每个结点的属性并输出结点级特征的特征矩阵作为输入。使用 GCN，交互网络可以解释复杂系统中的对象、关系和物理。一些交互框架已经被用于预测未来状态和潜在属性。RRL(relation reinforcement learning)嵌入 **multi-head dot-product attention** 作为关系块嵌入到 NN 中以学习在 agent 状态下一组实体的成对交互表示。RFM(relation forward model)使用监督学习在全局状态下以预测所有其他 agents 的动作，然而在局部观测环境下，RFM 不能很好地精确预测。

- 方法

我们使用 Bi 表示对于智能体结点 i 的所有邻居集合，所谓 i 的邻居依据距离或者其他规则，视环境而定，同时随着时间变化而变化。此外，邻居结点可以互相通信。在该种环境下的直觉是邻居 agent(即关系更加紧密的 agent)更可能互相交互和影响。同时，从邻居以外的所有其他智能体得到信息花费成本高，智能体也无法从全局共享信息中提取有价值的信息。

因为卷积可以逐步增加 agent 的可视野，使得合作范围不受限制(一层层卷据改善合作范围)，因此，只考虑邻居结点很高效。因此，DGN 可以适应动态环境，随着环境的发展而学习。因为 DGN 和 GCN 不同的是，DGN 使用邻居机制动态改变和适应环境。

1. 图卷积

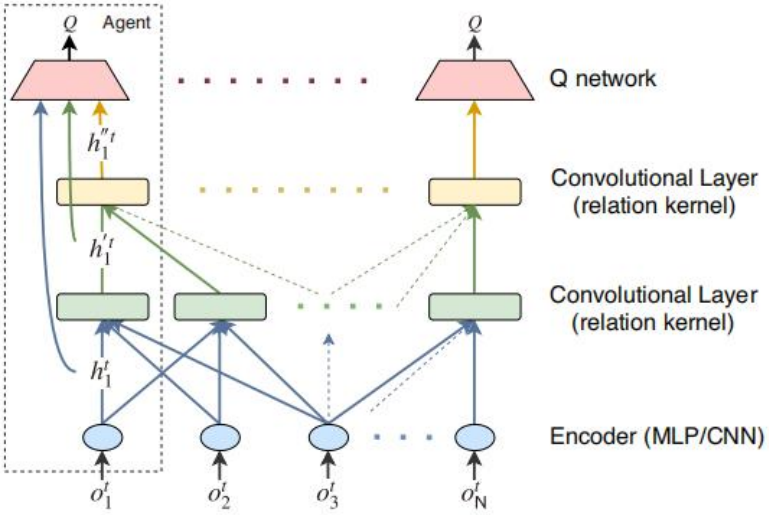

Figure 1: DGN consists of three modules: encoder, convolutional layer, and Q network. All agents share weights and gradients are accumulated to update the weights.

首先我们考虑一下部分可观察环境，o 表示每个 agent 单独的局部观测，是结点的属性，采取动作 ai 以后获得即时回报 ri。在该环境下，定义 DGN 如上图所示，分为三个模块：观测编码器、卷积层和 Q-Network，所有 agents 共享权重，并且将梯度累加以更新权重。

在局部可观测环境下，DGN 将每个 agent 的观测 o 通过观测编码层抽象为特征向量 h(低维度数据通过 MLP 编码，而图像输入通过 CNN 编码)，卷积层在局部领域将不同的特征向量 h 合成(根据距离或什么规则将 i 及其邻居分类合成)并产生新的天赋特征向量 h’，向量通过卷积层，agent 的可视野不断生长，同时更多的信息被收集，因此协作的范围可以增加。但是一旦 agent 及其邻居结点被合成，再多的卷积层也不能合成新的，因此文中通过两层关系卷积层将 agent 关系表达过滤合成。

由于 agents 的数量和位置随时间和环境而变化，明显关系表达图也因此而变化。为解决该问题，作者将所有的 agent 的特征向量组合为 N*L 大小的矩阵 F，N 为 agents 个数，L 为一个 agent 的特征向量长度；然后作者构造了大小为|Bi|+1*N 的第 i 个 agent 的邻接矩阵 Ci，第一行为结点 i 即 agent i 的 one-hot 编码，其他行为结点 i 邻居的 one-hot 编码，因此可以通过 Ci*F 表达局部域中的特征向量具体信息。

由 DenseNet 得到启发，先将特征逐层向前处理，之后喂入 Q-Network，将会组合和重用观测表达和对策略有不同贡献具体到在协作上有不同考虑角度的特征可视野。

$$\mathcal{L}(\theta)=\frac{1}{\mathrm{S}}\sum_{\mathrm{S}}\frac{1}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}}\left(y_i-Q\left(O_{i,\mathcal{C}},a_i;\theta\right)\right)^2, \tag{1}$$

where $y_i=r_i+\gamma\max_{a'}Q\left(O'_{i,\mathcal{C}},a'_i;\theta'\right)$, $O_{i,\mathcal{C}}\subseteq\mathcal{O}$ denotes the set of observations of the agents

其中 C 为上述定义的邻接矩阵，公式为 loss function，O(i, c)是 O 的子集，表示由 C 确定的智能体 i 的可视野集合，O(i, c)是输入而输出智能体 i 的 Q 值，由于环境易变难以学习 Q

function，因此作者在两个 timestep 下固定邻接矩阵 C 不变，Q-loss 累加以更新参数，同时更新参数使用软更新，公式如下：

$$\theta' = \beta\theta + (1-\beta)\theta'.$$

像 CommNet 一样，DGN 也可以看作是一个在所有 agents 优化平均期望回报的目标下以输出动作的中心化 policy 的因式分解。因式分解表示所有智能体共享权重参数，同时模型将智能体及其邻居 agents 连接起来，在每个 timestep 被动态确认。更多的卷积层(就是更大的特征可视野)就会有更高度集中以减少不稳定性。DGN 是基于 agent 关系导入经验的，因此更加考虑 agents 之间的互动。然而不适用于复杂策略，注意到执行期间，每个 agent 只需要从邻居中得到天赋特征而与 agents 数量无关，因此使得容易扩展。

2. 关系核

卷积核将特征在可视野中合成并抽象为天赋特征。核其中一个重要属性为与特征输入顺序无关。CommNet 具有如上属性，但是效益很小不可学习，而卷积核需要可学习。BiCNet 使用可学习核 RNN，但是该核与特征输入顺序有关，此外，卷积核应该可以学习如何去提取智能体之间的关系以合成输入特征。

由 RRL 得到灵感，作者使用多头点乘注意力作为卷积核以计算智能体之间的互动关系。对于每一个智能体 i，定义一个关系集合将邻居集合和智能体 i 的序号结合在一起，每个只能提的输入特征就是一种查询，将 key 和 value 通过每个独立的注意力头进行表达。对于一个注意力头 m 就是一个关系集合 B+i，关系集合中 i 与 j 的关系可以通过以下公式进行计算：

$$\alpha_{ij}^{m} = \frac{\exp\left(\tau \cdot \mathbf{W}_Q^m h_i \cdot (\mathbf{W}_K^m h_j)^{\mathsf{T}}\right)}{\sum_{k\in\mathbb{B}_{+i}} \exp\left(\tau \cdot \mathbf{W}_Q^m h_i \cdot (\mathbf{W}_K^m h_k)^{\mathsf{T}}\right)},$$

t 是尺度因子，对于每一个注意力头，所有智能体的输入特征值表达通过关系进行称量确定，并且组合在一起。然后 M 个注意力头的输出被并排在一起，喂入函数 σ 中，该函数 σ 可以是一层 MLP 或者非线性 ReLU，以产生卷积层的输出：

$$h_i' = \sigma(\text{concate}[\sum_{j\in\mathbb{B}_{+i}} \alpha_{ij}^{m} \mathbf{W}_V^m h_j, \forall m \in \mathsf{M}]).$$

图 2 展示了卷积层带有多头注意力核的计算过程。多头注意力核使得核与输入顺序无关，并且**允许核可以联合处理不同表示的子空间**。更多的注意力头给予更多的关系表达同时也使训练稳定。同时更多的卷积层使得更高级的关系能被提取以帮助协作。

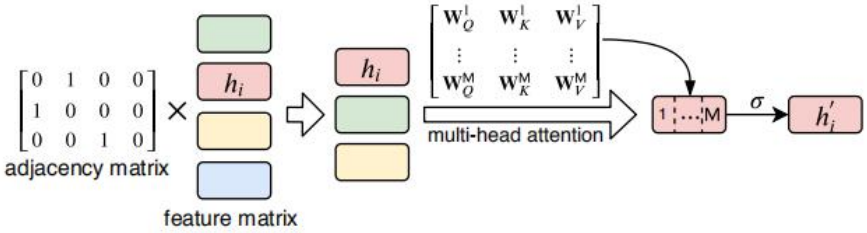

Figure 2: Illustration of computation of the convolutional layer with relation kernel of multi-head attention.

3. 时序关系正则化

使用 Q-learning 训练是将未来估计作为目标对现在进行估计。作者内置该方法并将其应用于模型的关系核中，以下就是具体将该方法应用的解释。直观上讲，智能体之间的关系特征被确定，在即使很短的时间段内，该关系都是稳定不变的，尽管环境一直在变。根据该性质和因为作者使用注意力权重表达关系的紧密，所以作者使用下一个状态的注意力权重贡献作为目标去计算当前的注意力权重贡献。这就可以鼓励 agents 形成一致的关系表达以表现一致的协作，但与一致动作无关，即一致的协作并不一定表现一致的动作，而 RNN/LSTM 则强迫动作的一致性，却无关协作。由于不同状态之间的关系不一定相同但相似，作者使用 KL 散度去计算两个状态的注意力权重贡献的距离。

应该注意的是，我们没有像在正常的 DQN 中那样使用目标网络来产生目标关系表示。因为如果有一个目标网络，那么因为参数不同而导致关系特征是不同的，不能保证学习协作是正确的，因为很容易得到关系特征由注意力权重表示，它是对参数敏感的。

定义 $\mathcal{G}^{\kappa}(O_{i,\mathcal{C}};\theta)$ 作为第 k 层卷积层的第 i 个智能体的注意力权重共享矩阵，那么带有时序关系正则化的 Q loss 表达为如下公式：

$$\mathcal{L}(\theta)=\frac{1}{\mathrm{S}}\sum_{\mathrm{S}}\frac{1}{\mathrm{N}}\sum_{i=1}^{\mathrm{N}}\left(\left(y_i-Q\left(O_{i,\mathcal{C}},a_i;\theta\right)\right)^2+\lambda D_{\mathrm{KL}}(\mathcal{G}^{\kappa}(O_{i,\mathcal{C}};\theta)||z_i),\right. \tag{4}$$

$z_i=\mathcal{G}^{\kappa}(O'_{i,\mathcal{C}};\theta)$ ， λ 是正则化项的系数。时序正则化项在高层为智能体长期时间建立 y 一致策略。

- 实验

- 总结

先使用带有多头注意力机制的卷积核将智能体归类，提取关系特征，然后学习；同时加上时序正则化项保证一致性策略。在各种合作多智能体场景中，DGN 显著优于现有方法。