计算机科学
COMPUTER SCIENCE

基于序列建模的生成式强化学习研究综述

姚天磊, 陈希亮, 余沛毅

引用本文

姚天磊, 陈希亮, 余沛毅. 基于序列建模的生成式强化学习研究综述[J]. 计算机科学, 2024, 51(11): 213-228.

YAO Tianlei, CHEN Xiliang, YU Peiyi. Review of Generative Reinforcement Learning Based on Sequence Modeling [J]. Computer Science, 2024, 51(11): 213-228.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.231000037

基于序列建模的生成式强化学习研究综述

姚天磊　陈希亮　余沛毅
陆军工程大学指挥控制工程学院　南京 210007
(ytl0730@qq.com)

摘　要　强化学习是机器学习中关于如何学习决策的分支,是一个序列决策问题,通过与环境反复交互试错找到最优策略。强化学习可以与生成模型结合使用来优化其性能,通常用于微调生成模型,提高其创建高质量内容的能力。强化学习过程也可以视为一个通用的序列建模问题,对任务轨迹上的分布进行建模,通过预训练生成模型产生一系列动作来获取一系列的高回报。在对输入信息进行建模的基础上,生成式强化学习能够更好地处理不确定性和未知的环境,更高效地将序列数据转换成用于决策的策略。首先针对强化学习算法和序列建模方法进行了介绍,对数据序列的建模过程进行了分析,然后按神经网络模型的类型进行分类探讨了强化学习的发展现状,在此基础上梳理了与生成模型结合的相关方法,并分析了强化学习方法在生成式预训练模型中的应用,最后总结了相关技术在理论和应用上的发展状况。

关键词:人工智能;强化学习;神经网络;生成模型;序列建模
中图分类号　TP181

Review of Generative Reinforcement Learning Based on Sequence Modeling

YAO Tianlei,CHEN Xiliang and YU Peiyi
College of Command and Control Engineering,Army Engineering University of PLA,Nanjing 210007,China

Abstract　Reinforcement learning is a branch of machine learning on how to learn decisions,which is a sequential decision-making problem that involves repeatedly interacting with the environment to find the optimal strategy through trial and error. Reinforcement learning can be combined with generative models to optimize their performance,and is typically used to fine-tune generative models and improve their ability to create high-quality content. The reinforcement learning process can also be seen as a general sequence modeling problem,modeling the distribution on task trajectories,and generating a series of actions through pre-training to obtain a series of high returns. Based on modeling input information,generative reinforcement learning can better handle uncertain and unknown environments,and more efficiently transform sequence data into strategies for decision-making. Firstly,an introduction is given to reinforcement learning algorithms and sequence modeling methods,and the modeling process of data sequences is analyzed. The development status of reinforcement learning is discussed according to different neural network models used. Based on this,relevant methods combined with generative models are summarized,and the application of reinforcement learning methods in generative pre-training models is analyzed. Finally,the development status of relevant technologies in theory and application is summarized.

Keywords　Artificial intelligence,Reinforcement learning,Neural network,Generative model,Sequence modeling

1　引言

BERT 和 GPT 等大规模预训练模型[1]的兴起与发展,极大地推进了人工智能技术的落地。模型通过大量的参数和复杂的预训练目标可以从有标签或无标签的数据中学习到知识,其不仅为自然语言处理任务提供了强有力的技术支持,同时在包括强化学习在内的诸多领域中也展现了出巨大潜力。作为求解序贯决策问题的有效方法之一,强化学习通常将问题建模为马尔可夫决策过程。智能体在训练时首先会通过动作与环境交互,在此过程中智能体得到新的状态,再由环境给出一个回报奖励。通过多次的迭代,智能体最终学到完成任务的最优动作。强化学习过程如图 1 所示。

然而传统的强化学习一般是在离散的环境下进行,动作空间和样本空间的大小都有一定的局限性。例如当输入图像或音频等数据时会产生很高的维度,此时靠传统的强化学习技术来处理就很困难。深度强化学习将深度学习与强化学习相结合,运用深度神经网络强大的表征能力去拟合 Q 表或策略来解决状态动作空间过大或连续状态动作空间的问题。

到稿日期:2023-10-07　返修日期:2024-03-15
基金项目:国家自然科学基金(62273356)
This work was supported by the National Natural Science Foundation of China(62273356).
通信作者:陈希亮(383618393@qq.com)

深度神经网络的引入极大地提升了强化学习算法的通用性，以卷积神经网络(CNN)为代表的网络结构的引入，使得强化学习可以处理图像类输入，并能够实现高维特征的有效提取[2]。在处理部分可观察或者含噪的信息特征时，强化学习往往可以通过结合循环神经网络来解决问题。但是这些方法并没有显式解决序列建模问题。以 Transformer[3] 为代表的生成式预训练方法，在自然语言处理领域展现出了极强的长时间依赖关系序列建模能力。生成式强化学习可以通过学习序列数据来生成策略，以解决强化学习中的问题。这种方法通常使用 Transformer 等深度学习架构来对序列数据进行建模，然后将这些序列数据转换成能够用于决策的策略。与传统的强化学习算法相比，生成式强化学习具有更好的稳定性和表现能力，可以处理更复杂的环境和任务。

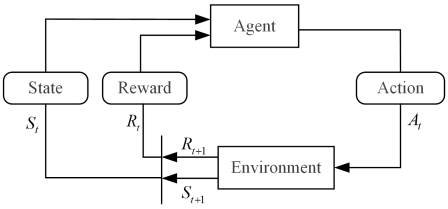

图 1 强化学习体系结构

Fig. 1 Reinforcement learning architecture

但 Transformer 作为强化学习函数近似器，还存在诸多挑战。1)Transformer 的巨量样本需求与强化学习低效的采样效率之间的矛盾；2)强化学习基于自然时间顺序接收单帧状态信息和 Transformer 基于位置编码的顺序信息处理方式之间的矛盾；3)当前时序差分学习(Temporal Difference，TD)方法可能会导致对短期奖励的过度短视，减少对未来奖励的需求等。针对这些问题，Chen 等[4]将强化学习的状态转移单元(包括状态、动作、奖励以及未来奖励总和)当作一串序列数据，并通过 Transformer 架构将建模这些序列作为学习的核心任务，在离线强化学习任务中取得了显著的效果。然而离线强化学习是从固定、有限的经验中产生最有效的行为的方法，可能会出现错误传播和价值高估，在此情况下的探索利用尤为困难。事实上，通过在状态、动作和奖励序列上训练自回归模型，可以实现策略抽样到自回归生成建模的转变。因此，Janner 等[5]从强化学习的基础要素状态、动作、奖励和值函数构成的状态转移单元出发，试图建模出产生这个序列的概率分布，这种方法在 Atari，OpenAI Gym 等任务中取得了巨大的成功。

基于以上内容与研究，本文首先介绍了序列建模的相关方法，按神经网络的类型进行分类，梳理了近年来国内外强化学习的前沿技术，深入分析了序列建模的过程与思想；然后总结了结合强化学习与以基于 Transformer 为主的预训练生成模型的前沿方法，并探讨了其在不同任务中的表现与局限性；最后对未来的发展与应用进行了综述。

2 相关概念

2.1 MDP 模型

目前强化学习的主流方法首先是将要解决的问题建模为马尔可夫决策(Markov Decision Process，MDP)[6]。MDP 过程是强化学习问题在数学上的理想化形式，描述了完全可观测的环境，由一个五元组 (S,A,P,r,γ) 来定义，其中 S 为状态合集，A 为动作合集，P 为状态转移概率，r 为奖励函数，γ 为折扣因子。在此模型中，智能体采取动作 a 与环境交互后，由环境反馈一个奖励信号 r 和一个新的状态 S'，智能体再根据奖励信号和新的状态更新策略。该过程最核心的目标就是计算出一组选择 A 的最优策略，从而最大化累积回报 $R=\sum_{k=0}^{\infty}\gamma^k r^{k+t}$。

然而在大多数情况下，智能体无法获取到环境的整体状态，只能观测到部分信息，在这种情况下通常将问题建模为 POMDP(Partially Observable Markov Decision Processes)。POMDP 可以被看作 MDP 的扩展，是环境状态部分可知的情况下序贯决策的理想模型，通常由七元组 (S,A,T,r,O,Z,γ) 表示，其相比 MDP 增加了一组观察结果集 O 以及一个观察函数 Z，表明了状态和观察之间的关系。由于在 POMDP 中智能体无法确定自己处于哪种状态，因此智能体必须通过不断收集环境信息来获取当前状态，以此来更新自身状态的可信度。

2.2 序列建模

序列是一种带有前后顺序关系的数据，或者是带有位置顺序的固定序列信息。通常假设数据序列是由某个潜在过程生成的，而此过程可以被建模为一个概率分布，序列建模的核心目标就是学习这个概率分布。目前序列建模的方法有很多种，如 n-gram 模型、隐马尔可夫模型(Hidden Morkov Model，HMM)、循环神经网络(Recurrent Neural Network，RNN)、长短时记忆网络(Long Short-Term Memory，LSTM)、Transformer 架构等。这些方法在处理不同类型的序列数据时各有优势，需要根据具体问题选择合适的模型。合理地对数据序列进行建模可以有效地提高模型的预测准确率，能够及时地对模型参数进行调整以适应不同的数据环境。

传统的深度强化学习将数据序列建模为马尔可夫模型。在 DQN 算法中，将 Atari 游戏的连续 4 帧画面作为一个状态输入，采用卷积神经对画面状态进行高维信息提取。循环神经网络等技术的出现为神经网络提供了记忆功能，通常使用记忆能力来提升智能体获取环境信息的效率[7]，进一步提高了处理序列信息的效率。Transformer 架构使用了注意力机制，将序列中任意两个位置之间的距离缩小为一个常量[8]，具有更好的并行性，为处理序列问题提供了更高效的方法。

序列建模在很多领域都有广泛的应用，如自然语言处理、语音识别、图像生成、推荐系统等[9]。在这些领域中，序列建模可以从数据中提取有用的信息，对原始序列进行预测和分析，或者生成新的数据序列。

2.3 深度强化学习

深度强化学习就是将深度学习运用到强化学习任务中。与传统强化学习相同，都是根据输入的状态计算出最优动作，不同的是深度强化学习是通过深度神经网络来完成这一过程，但解决的仍然是决策问题。

深度强化学习算法通常可以分为基于值(Value-based)的方法和基于策略(Policy-based)的方法。在基于值的学习算法中，通过 Q 函数产生的回报去评估当前状态 s_t 下每个

动作的好坏，根据 $\arg\max Q_\pi(s_t,a_t)$ 操作选取最佳动作，典型的算法有 DQN 和 DDQN 等。当处理连续动作空间时，要想采用基于值的强化学习，必须先对动作空间进行离散化，但连续空间到离散空间的映射往往是指数级的，求解变得很困难。此外，随机策略的选择往往具有不确定性。基于策略的方法可以很好地解决连续空间的问题，通过对策略 π 直接进行建模，学习 $\pi_\theta(s,a)=p(a|s,\theta)$，给定状态 s，可以直接得到动作 a[10]。基于策略的方法同样也适用于离散动作空间，可以从当前状态输出一个 $|A|$ 维的离散分布，根据这个概率分布来选取动作[11]，典型的算法有 PPO 和 DPG 等。将基于策略与基于值的两种算法相结合，可以得到 Actor-Critic 算法。Actor 与 Critic 是两种不同的神经网络，Actor 基于概率来预测行为，Critic 估计每一个状态的价值。TD-error 为动作价值函数估计值与实际值的差异，如果 TD-error 是正的，下个动作就要加大更新幅度；反之就减小 Actor 的更新幅度。Critic 基于 Actor 的行为评判行为的得分，Actor 根据 Critic 的评分修改选择行为的概率。

深度强化学习结合深度神经网络强大的表征感知能力和强化学习的决策能力，在处理高维的复杂任务中取得了优异的表现。相对于传统的强化学习，其具有更强的学习能力、泛化能力和适应性，能够处理更复杂的任务，具有更高的学习效率和可解释性，使强化学习技术真正从理论走向了现实。由于神经网络是建立在样本独立同分布的基础上的，通过概率统计的方法来拟合输入与输出之间的关系，因此当数据在分布之外时准确率将大幅下降。此外，深度强化学习算法应用中的奖励函数设计难、样本利用率低、模型过拟合等问题也需要进一步解决。

2.4 生成式强化学习

生成模型(Generative Model)是对概率分布进行建模的一种机器学习模型，它可以从给定的数据中学习出概率分布，并且可以利用学习到的概率分布生成新的数据样本[12]。简单来说，生成模型是通过学习输入和输出的关系，对输入数据的概率分布进行建模，从而能够生成新的数据。与判别模型不同，生成模型不仅能够预测输出的类别或标签，还能够推断输入数据的概率分布。常见的生成模型有自回归模型、生成对抗网络、扩散模型以及各类基于 Transformer 的混合模型等。

生成式强化学习(Generative Reinforcement Learning，GRL)是一种结合了生成模型和强化学习的技术，其主要目标是让智能体学习环境中的概率分布并生成最优策略。这种方法不需要显式地定义状态空间和动作空间，而是通过生成模型来表示环境。生成模型可以在给定输入的情况下，生成下一个状态或动作的建议[13]。相比传统的强化学习，生成式强化学习可以通过生成新的策略来探索未知的环境并与环境交互，能从少量的经验中学习知识，在更短的时间内学习到更好的策略，并且适用于不确定性和未知的环境。同时，生成式强化学习对先验知识和推理复杂度的要求更高，需要设计合适的奖励函数和策略网络来处理复杂的推断和规划问题[14]。

3 序列建模相关方法

3.1 循环神经网络

RNN 在自然语言处理等领域已经取得了广泛的应用。传统神经网络虽然层与层之间是全连接的，但是每层的节点之间是不存在连接的。而在循环神经网络中，隐藏层的输入包括上一时刻隐藏层的输出和输入层的输出，隐藏层内部节点之间存在连接，每一步都能够实现参数共享[15]。

卷积神经网络可以有效地处理空间信息，循环神经网络则能够更好地处理序列信息。与传统的神经网络相比，循环神经网络可以处理变长的输入序列，每个输出的状态都是由前一时刻的状态和当前时刻的输入来决定，对数据有一定的记忆功能，其结构如图 2 所示。图 2 左侧部分是未展开的形式，包含输入、隐藏层和输出；右侧部分是展开的形式，其中横轴的长度可以被看作是输入序列的长度。

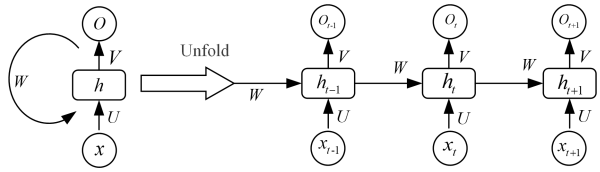

图 2 RNN 结构图

Fig. 2 RNN structure diagram

从循环神经网络的结构中可以看到，t 时刻隐藏层状态 h_t 由前一时刻的隐藏层状态和当前时刻的输入所决定，展开的参数 U，V 和 W 在每个时间步都是共享的，这样的结构使得循环神经网络可以很好地适应变长输入[16]。由于参数共享，RNN 能够更有效地处理变长输入，模型本身能够获取序列层级的规律，相比 MLP 更适合对序列数据建模[17]。在序列层级关系的学习中，RNN 的参数共享机制更符合奥卡姆剃刀原则，能够有效地减少参数量，降低过拟合的风险。RNN 理论上可以处理任意长度的序列数据，但在实际应用中会存在梯度消失或梯度爆炸等问题，不能训练具有长期依赖的序列[18]。

LSTM 是一种特殊的 RNN[19]，在 Simple RNN 的基础上优化了隐藏层的结构，加入了门结构，使其具有长期记忆的能力。LSTM 模型的关键就在于 LSTM 单元，其结构如图 3 所示。

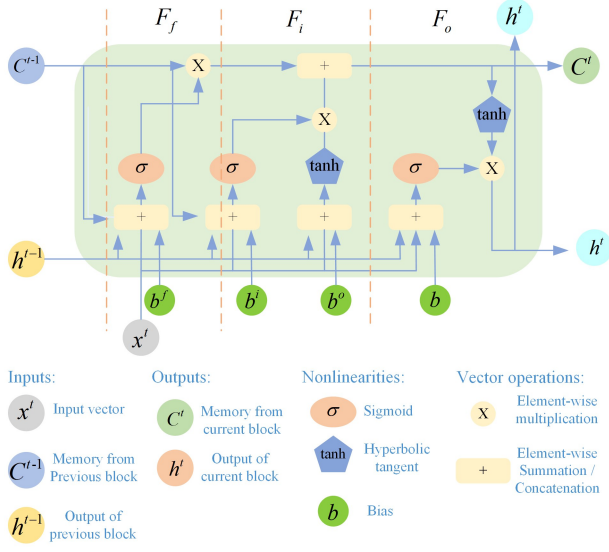

图 3 LSTM 结构图

Fig. 3 LSTM structure diagram

避免长期的依赖性问题是 LSTM 中最核心的问题，为此 LSTM 引入了门控机制和记忆单元，使模型能够在处理信息的同时更好地捕捉长期依赖关系。其中，输入门可以接收新的数据来对当前状态进行更新，将前一层的隐藏状态信息与当前的输入信息传入到 sigmoid 层中，再使用一个 tanh 层为保留的信息生成一个向量来更新细胞状态 $\tilde{C}_t$。输出门决定 LSTM 的输出内容，输出的信息基于细胞状态进行一定的过滤[20]。利用 tanh 层将细胞状态值映射到区间[−1,1]，将其与 sigmoid 层的输出相乘，得到应输出的信息，再将隐藏状态作为当前细胞的输出，把新的细胞状态和新的隐藏层状态传递到下一个时间步长中去。遗忘门可以决定细胞中应该丢弃的信息，输出为 1 时，则全部保留；输出为 0 时，则全部丢弃。相较于传统的 RNN，门结构能够有效地缓解梯度消失或梯度爆炸问题，从而处理更长的序列数据。

一般情况下，相对于隐马尔可夫模型和时间递归神经网络，LSTM 有着更优异的表现，可以更好地捕捉序列内部的复杂规律，对数据的性质没有过多的要求，在实际应用中更加灵活和方便。很多场景下可以将 LSTM 作为复杂的非线性单元来构造更大型的深度神经网络[21]，但由于其结构相对复杂，计算量相较 RNN 更大，训练数据时容易出现过拟合[22]。当序列长度变化较大时，可能需要重新调整模型的结构和参数。

3.2 注意力机制

注意力机制(Attention Mechanism)借鉴了生理学上的知识，例如人们观察一幅图片时，总会首先把注意力集中在图片中比较突出的位置，聚焦于输入信息中更为重要的信息，从而降低对重要程度低的信息的关注度或者直接无视不相关的信息[23]。在神经网络的训练中，由于输入的参数权重不同，可以将计算资源优先分配给指定的任务，从而在众多的数据信息中快速获取到特定的数据，在硬件资源有限的情况下改善资源分配的问题[24]，以此来提高模型效率。

3.2.1 自注意力/多头注意力机制

在神经网络的实际训练中，输入的向量之间存在着一定的关系，如果无法准确预测这些信息之间的关系，则会导致训练的模型效果不佳。为了解决普通神经网络无法建立多个输入之间的相关性的问题，研究者引入了自注意力机制(Self Attention)[25]。自注意力机制在序列的每个单元都能获取完整的数据，可以更多地聚焦于输入中的关键信息，并且能够实现并行计算。

注意力机制中包含 3 个序列，分别为 $\boldsymbol{Q}$(Query)，$\boldsymbol{K}$(Key)以及 $\boldsymbol{V}$(Value)，其中 $\boldsymbol{Q}$ 为词查询向量，$\boldsymbol{K}$ 为键向量，$\boldsymbol{V}$ 为内容向量。自注意力机制中 $\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}$ 这 3 个序列都来自同一输入，在物理意义上基本相似，都代表一组序列中不同 token 组成的矩阵。其中 $\boldsymbol{K}$ 与 $\boldsymbol{V}$ 两者长度相等，$\boldsymbol{Q}$ 长度可以不同。先使用 Embedding 方法将输入序列中的文本信息转化为向量 $\boldsymbol{X}=\{x_1,x_2,x_3,\cdots,x_n\}$，再将转化后的向量分别与 3 个矩阵 $\boldsymbol{W}^Q$，$\boldsymbol{W}^K$，$\boldsymbol{W}^V$ 相乘求得 $\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}$，计算 $\boldsymbol{Q}$ 与 $\boldsymbol{K}$ 的相似度评分 *score* 并对其归一化处理，对其结果进行 softmax 后再乘 $\boldsymbol{V}$ 即得到最终结果[26]。其核心公式如式(1)所示：

$$Attention(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=\mathrm{Softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{1}$$

为了提高 Self Attention 的拟合性，研究者提出了多头注意力机制(Multi-Head Attention)[27]。多头注意力机制可以使用不同版本的查询向量 $\boldsymbol{Q}$ 来实现多个注意力模块的运行，可以将一个线性投影建立在不同的投影空间，得到输入向量之间不同的关系，对计算结果进行拼接。多头注意力机制与自注意力机制最大的不同在于其复制了多个单头，广义上单头数为 1 时也属于多头注意力，其中每个头部都有各自的查询、键和值矩阵，可以形成各自的得分向量及对应的注意力权重向量，从而使得模型能够学习更多的信息。

多头注意力机制在自然语言处理中应用非常广泛。比如在语言翻译和文本分类任务中，需要有效地理解文本的含义，不仅要关注输入的语言序列，还需要关注输出的目标语言，因此就必须要关注信息中的各个部分。多头注意力机制可以发挥自己的优势，帮助模型高效地完成这些任务。

3.2.2 全局/局部注意力机制

在 LSTM 中，每次输入一串文本中的词向量，在每个时刻即获得一个隐藏状态 $\boldsymbol{h}_s$，最终的目标隐藏状态需要等到最后一个时间步才能求得。在此过程中，模型只会关注于最终的隐藏状态 $\boldsymbol{h}_t$，并不会利用到之前时间步内获得的 $\boldsymbol{h}_s$，这会导致模型在面对一组输入文本的时候没有侧重点，对文本中每一个词向量都给予相同的关注度，使得计算效率不佳[28]。此时可以使用全局注意力机制(Global Attention)[29]来解决此问题。全局注意力机制可以学习到全部的 source 信息，在上述过程中会使用每一个时间步内产生的隐藏状态来计算上下文向量 $\boldsymbol{c}_t$。在得到 s 时刻的隐藏状态 $\boldsymbol{h}_s$ 和最终的隐藏状态 $\boldsymbol{h}_t$ 后，s 时刻单词的权重参数 $a_t(s)$ 计算过程如式(2)所示。将每个时刻的 $a_t(s)$ 与 $\boldsymbol{h}_s$ 数乘计算后则得到 $\boldsymbol{c}_t(s)$，再对其进行数学计算得到上述的 $\boldsymbol{c}_t$[30]。式(3)给出了 3 种 *score* 的计算。

$$a_t(s)=align(\boldsymbol{h}_t,\bar{\boldsymbol{h}}_s)=\frac{\exp(score(\boldsymbol{h}_t,\bar{\boldsymbol{h}}_s))}{\sum_{\bar{s}}\exp(score(\boldsymbol{h}_t,\bar{\boldsymbol{h}}_s))} \tag{2}$$

$$score(\boldsymbol{h}_t,\bar{\boldsymbol{h}}_s)=\begin{cases}\boldsymbol{h}_t^{\mathrm{T}}\bar{\boldsymbol{h}}_s, & \text{dot}\\ \boldsymbol{h}_t^{\mathrm{T}}\boldsymbol{W}_a\bar{\boldsymbol{h}}_s, & \text{general}\\ \boldsymbol{v}_a^{\mathrm{T}}\tanh(\boldsymbol{W}_a[\boldsymbol{h}_t;\bar{\boldsymbol{h}}_s]), & \text{concat}\end{cases} \tag{3}$$

与全局注意力机制不同，局部注意力机制只提取了部分隐藏状态信息，有选择性地选取上下文窗口中的信息，因此缓解了计算量太大的问题。总的来说，全局注意力机制与局部注意力机制在实际应用中都很重要。当输入的序列信息长度合适时，两种机制在效果上并没有明显的差异，但局部注意力机制还需要计算一个位置向量[31]，而位置向量的预测通常有一定的偏差，这会影响对齐向量的准确率。

3.3 扩散模型

目前主要的生成模型如图 4 所示。其中生成对抗网络(GAN)通过判别器和生成器相互博弈对抗，不断调整参数来优化模型的性能，最终生成器能够输出与真实数据高度相似的新样本。变分自编码器(VAE)的核心思想是通过解码器和编码器的训练，学习数据的潜在分布，从而实现对数据的重构与生成。流模型(Flow-based Models)主要通过一系列的可逆变换建立先验分布和实际数据分布之间的映射关系，从而实现数据的生成。扩散模型(Diffusion Model)源自非均衡

热动力学，是一类基于概率似然的生成模型，通过对一个正态分布变量进行去噪来学习数据分布 $p(x)$，将先验数据分布转换为随机噪声，再对变换进行逐步修正，重新建模一个与先验分布相同的新样本[32]。

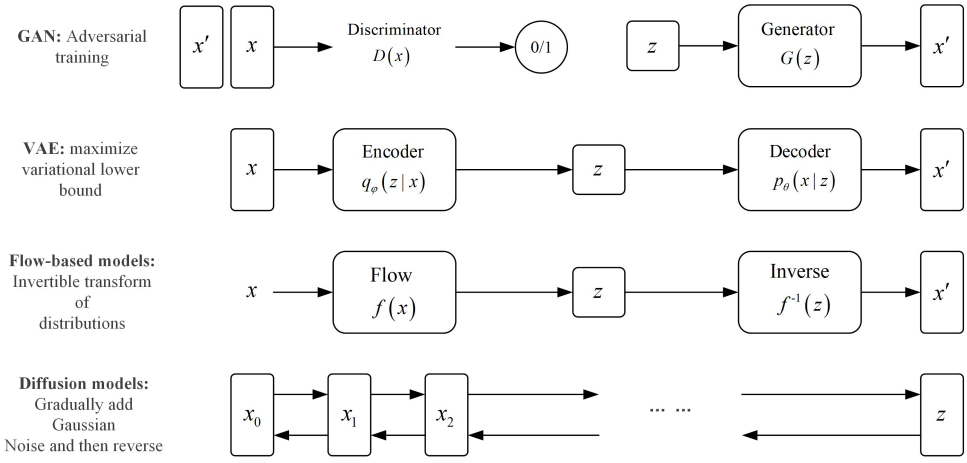

图 4 各类生成模型的对比

Fig. 4 Comparison of various generative models

平衡模型的灵活性和可计算性是生成式建模的核心问题之一。扩散模型先通过正向扩散系统地扰动样本的分布，再使用反向扩散来恢复数据的分布，从而构建一个灵活且具有可计算性的生成模型[33]。假设原始样本的分布为 $q(x_0)$，$x\sim q(x_0)$ 表示从样本中随机采样；正向扩散 $q(x_t|x_{t-1})$ 指向样本 x_{t-1} 逐步添加高斯噪声得到 x_t，直至数据完全变为噪声。向样本逐步添加噪声的过程如式(4)所示，可通过高斯分布 $\varepsilon\sim\mathbb{N}(0,\mathrm{I})$ 来表示，其中 β_t 为一个随时间变化的变量。

$$q(x_t|x_{t-1})=\mathbb{N}(x_t|\sqrt{1-\beta_t}x_{t-1},\beta_t\mathrm{I}) \tag{4}$$

反向扩散过程 $p(x_{t-1}|x_t)$ 通过神经网络，根据 x_t 去学习 x_{t-1} 的概率分布，对其逐步进行去噪，最终得到和原始样本高度相似的新样本。其过程如式(5)所示，其中 θ 为神经网络的参数。

$$p_\theta(x_{t-1}|x_t)=\mathbb{N}(x_{t-1};\mu_\theta(x_t,t),\sum_\theta(x_t,t)) \tag{5}$$

3.4 Transformer 架构

Transformer 使用注意力机制代替语言处理任务中常用的循环神经网络，并结合前馈神经网络完成整个过程的计算。其最大的改进就是实现了并行计算的能力[34]，大大提高了计算效率，在处理长序列数据方面表现出了优越的性能。Transformer 最初是作为机器翻译的序列到序列模型被提出，但后来的工作表明，基于 Transformer 的预训练模型能够在广泛场景中实现最优性能，例如自然语言处理、图像分类等。现有工作将基于 Transformer 的预训练生成模型与强化学习相结合，其在各类任务中均取得了优异的表现。Transformer 与其他序列建模方法的对比如表 1 所列。

表 1 序列建模方法的对比

Table 1 Comparison of methods based on sequence modeling

模型方法	原理	优势	局限性
RNN	在隐藏层中有一个循环连接，每个时间步都可以根据之前的记忆输出新的结果，能更有效地解决时序结构的任务和问题	可处理任意长度的输入并且不增大模型规模，每个时间步权重矩阵共享，学习效率高	不能处理长序列数据，容易导致梯度消失和梯度爆炸
LSTM	通过加入输入门、输出门、遗忘门来控制信息的流动，可以选择性地记住选择的信息而舍弃无用的信息	在序列建模方面具有一定的优势，具有长时记忆功能，改善了长期依赖问题	不能并行计算，序列长度超过一定限度后仍然存在梯度消失现象
GRU	通过加入重置门和更新门来控制前一状态信息的传递	模型简单，参数量少，相对于 LSTM 计算复杂度更小，训练速度更快	仍然不能完全解决梯度消失问题
Diffusion Models	对正态分布变量进行去噪来学习数据分布，将先验数据分布转换为随机噪声，再对变换进行逐步修正，重新建模一个与先验分布相同的新样本	训练过程平稳，容易训练；拥有更高的精度，生成质量高	训练计算成本高，采样速度慢；没有编码能力，无法编辑隐空间
Transformer	由编码器与解码器构成，使用位置嵌入来构建文本的顺序，通过自注意力机制和全连接层来计算	可以并行计算，计算两个位置之间关系的成本与距离无关，具有良好的可扩展性	局部信息获取能力不强，位置信息编码存在问题，计算效率有待提高

3.4.1 基本结构

Transformer 的整体模型架构图如图 5 所示，可以将其视为一个 Encoder-Decoder 结构的模型。Transformer 结构可分为 4 个部分。第一个部分需要对输入序列中的每个词向量添加位置编码 Position Encoding，以便在后续过程中获取单词的位置信息；第二个部分 Encoder Block 实际上是由 6 个编码器组成，每个编码器结构相同，其包含多头注意力和全连接神经网络，但是编码器之间的权重参数可能不同；第三个部分 Decoder Block 同样是由 6 个解码器组成，每个解码器包含多头注意力与全连接神经网络[35]。由于在注意力机制中，模型可以观测到序列中的完整信息，因此每个解码器比编码器多增加了 Masked Multi-Head Attention，其中 Mask 代表掩码，

可以隐藏部分序列信息，防止模型观察到当前时刻之后的信息，使得训练与预测的一致性得到保证。最后一层为模型的输出，在经过线性变换与 softmax 操作后得到一个概率分布，输出的值为概率分布中概率最大的单词。

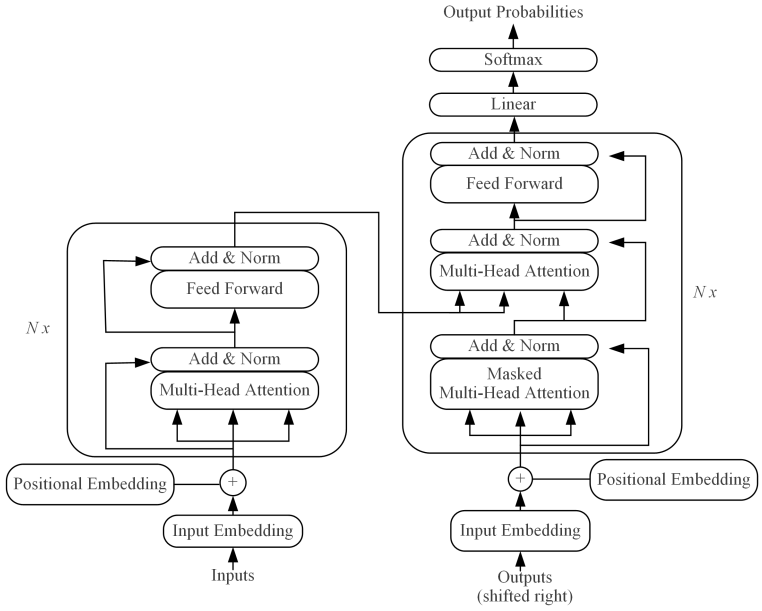

图 5 Transformer 结构图

Fig. 5 Transformer structure diagram

Transformer 实现了并行运算，并且可以解决长距离依赖的问题，提升了训练效率，在很大程度上解决了预测质量的瓶颈问题。但其依赖于自注意力机制，因此可能存在获取局部信息不准确的问题，并且仍然会受到嵌入程序的质量影响。

3.4.2 预训练模型

Transformer 强大的序列建模能力已经在自然语言处理等领域展现出极大的优势，基于 Transformer 架构的预训练模型也相继涌现，其中训练效果相对突出的有 OpenAI 开发的 GPT 模型以及 Google 公司研发的 BERT[36] 模型。GPT 模型只包含解码器，与以往的模型相比，其可以作为一个拥有强大泛化能力的模型对目标任务进行微调，而不需要针对不同的任务去使用不同的模型[37]。GPT 系列模型主要使用自监督学习进行训练，但可以与强化学习方法相结合以解决更复杂的任务[38]。基于 Transformer 的预训练信息如表 2 所列。

表 2 基于 Transformer 的预训练模型的对比

Table 2 Comparison of pre-training models based on Transformer

模型	架构	参数量	主要数据集
BERT	Encoder	1.1×10^8	Wikipedia, BookCorpus
ALBERT	Encoder	1.2×10^7	Wikipedia, BookCorpus
GPT-1	Decoder	1.17×10^8	BookCorpus
GPT-2	Decoder	1.542×10^9	WebText, BookCorpus, Wikipedia
GPT-3	Decoder	1.75×10^{12}	Common Crawl, WebText2, Books1, Books2, Wikipedia
GPT-4	Decoder	1.8×10^{13}	Common Crawl, WebText2, Books1, Books2, BookCorpus, Wikipedia
BART	Encoder-Decoder	4×10^8	BookCorpus, English Wikipedia
T5	Encoder-Decoder	1.1×10^{11}	C4

OpenAI 于 2018 年首次发布第一代 GPT 模型后，又陆续在其基础上进行了改进，每一代 GPT 模型在前一代的基础上融合了更大规模的参数量，在性能上取得了显著提升。GPT 模型使用了 Transformer 架构来代替传统的 LSTM 作为特征提取器，主要结构由 12 层的 Transformer block 组成，每层只包含一个 Masked Multi Self-Attention 以及一个 Feed Forward。其训练过程分为预训练和微调两部分。在预训练过程中，模型通过非监督学习方式在大量语料数据中学习，不同于使用双向语言模型的 ELMo(Embeddings from Language Models)根据上下文来预测目标词语，GPT 模型中的 Masked Multi Self-Attention 只能通过上文来预测接下来的词，使用的是单向语言模型。在微调阶段，通过有监督的方式对前面得到的预训练模型根据不同的任务来进行调整。GPT 模型的提出在一定程度上改善了实际语料数据中标签稀疏的问题，提高了对无标签原始数据的利用效率，减轻了 NLP 中有关任务对监督学习的依赖。但 GPT 在长文本的处理上需要不断计算，生成的错误信息会在数据末端聚集，使得生成结果质量不佳。

ChatGPT 最近受到了广泛的关注，被各界所追捧，在实际应用中表现出了非凡的知识推理以及内容生成能力。其本质是基于 GPT3.5 的对话生成模型，通过与用户之间的对话来进行自然语言交互。ChatGPT 的实现过程包含 3 个阶段，分别是监督微调、训练回报模型以及采用强化学习方法来提高模型的能力。ChatGPT 可被视为改进版的 InstrcuctGPT，向通用性人工智能迈出了坚实的一步[39]。

GPT 和 BERT 等系列生成式预训练大语言模型的飞速

发展已经在现实生活的各个方面产生了重大影响。模型强大的文本理解及编辑能力、图像分析能力以及数据挖掘等能力在诸多场景发挥着重要作用。在医疗诊断上，生成式人工智能可以根据患者的病理信息和潜在数据创建一个合成的对照患者，通过模拟不同的治疗手段对其产生的效果，来提高临床试验的效率[40]。英矽智能在多模态生成强化学习平台Chemistry24上基于结构生成化学分子的药物设计方法，研制出了一种用于癌症治疗的抑制剂。在自然科学上，生成式人工智能已经开始在量子领域探索，以更有效地模拟电子的强相关态，增进对材料科学和量子科学的理解。在军事上，生成式人工智能不仅能够在广泛的文本、图像等数据中提取情报，在作战筹划、战场态势把控上也有着出色的能力。未来，人工智能技术间接辅助战争甚至主导战场的趋势已逐步显现。此外，在金融量化、艺术文化以及人文教育等方面，生成式人工智能都发挥着出色的应用效果[41]。

随着AI生成模型的不断发展和广泛应用，各领域对如何使用AI生成模型都有着深刻见解。合理地利用AI生成模型对解决关键的社会问题具有广泛的影响。但如何正确使用该技术去解决问题并产生积极意义，在安全与道德规范方面还需进一步把控。

4 基于序列建模的生成式强化学习

不同于监督学习、无监督学习，强化学习主要解决的是序列决策问题，通过不断与环境进行交互，不断更新行为策略，从而获得最大累计奖励。上文介绍了序列建模的常用方法，接下来将概述在序列建模的基础上如何将强化学习和各类神经网络以及生成式模型相结合的方法。

4.1 基于循环神经网络的强化学习

深度强化学习可以很好地解决连续动作状态问题，但是需要输入整个环境状态来进行决策。其在处理相似的问题时往往需要重新训练，对不同的问题泛化能力较弱，针对复杂的任务需要花费更多的时间和计算资源[42]。合理地利用循环神经网络，可以有弥补解传统深度强化学习的不足。

在Atari游戏中，通常将游戏的4帧画面作为一个状态来获取物体的运动信息，然而DQN算法无法获取4帧前的信息，超过4帧信息的任务都无法满足MDP条件，因为游戏未来的状态和奖励不仅仅取决于DQN当前的输入[43]。此时可以将问题建模为部分可观测的马尔可夫问题(①POMDP)。Hausknecht等[44]将循环网络加入了DQN，使用LSTM替换DQN中的卷积全连接层，把改进后的算法称为Deep Recurrent Q-Network(DRQN)，其只需要1帧画面数据就可以获取物体的运动信息，达到DQN的性能水平。算法使用部分可观察的信息来学习，通过全局可观察的信息来进行评估，能够更好地处理信息丢失的问题，其效率与所观察信息的质量有关。在训练时，循环网络和卷积网络同时迭代更新。更新可分为两种方式，分别是顺序更新和随机更新。两者当前时刻的输入状态都是由前一个时间步所决定，不同的是顺序更新会在随机选取的时间点上一直训练到结束，而随机更新会提前设置好训练的步长。

深度强化学习通过大量的数据训练，可以让智能体学会复杂的行为策略。Duan等[45]试图让智能体通过少量的任务就能学习更多的知识，提出了RL^2方法。RL^2利用RNN的权重进行编码，接收强化学习算法中的全部信息，将强化学习本身作为一个强化学习任务，智能体通过过去时刻的信息学习MDP来调整策略，从而学习到强化学习算法。Li等[46]使用循环深度学习模型结合强化学习与监督学习，利用数据中的监督信号来学习隐藏状态表示，并使用RNN和LSTM模型缓解了RL中的长期依赖型问题。Stamatelis等[47]不对任何先验概率、动作以及观察集作出任何假设，将无模型的强化学习模型与RNN相结合，仅通过与训练环境交互或访问大型训练数据集来解决未知环境中的主动序贯假设检验问题。Querido等[48]将视觉注意和主动感知应用到无模型RL智能体上，并将循环注意力模型(Recurrent Attention Model，RAM)与近端策略优化算法相结合，观察模型是否具有与包含完整观察信息的Model-Free RL算法相似的性能。D'Alonzo等[49]通过原始轨迹数据检测RL的对称性来创建低级状态控制和语义表示，使用RNN来区分候选对称性的原始轨迹和变换轨迹，创建对数据集级别的所有对称性不变的高级表示，再将RL行为的属性传递给用户。

循环神经网络可以有效地处理强化学习中的序列决策问题，相比传统的深度强化学习具有更好的泛化能力，能够更好地捕捉序列数据中的时间信息和长期依赖关系，在状态信息不完整的条件下依旧可以保持良好的性能。但是在训练的过程中其仍然需要耗费大量资源，不能并行计算；在面对较长序列和较深的网络结构时，其计算时间成本会大大提高。

4.2 基于扩散模型的强化学习

机器学习中，扩散模型或扩散概率模型是一类潜变量模型，是用变分估计训练的马尔可夫链。扩散模型的目标是通过对数据点在潜在空间中的扩散方式进行建模，学习样本的潜在结构。

扩散模型在CV和NLP等领域已经取得了令人振奋的成果，最近一些工作研究将扩散模型与强化学习相结合来解决序列决策问题，利用扩散模型来建模分布复杂的轨迹或提高策略的表达性。现有研究表明，扩散模型能够高效地输出action来进行实时决策，并且能够建模完整(S,A,r,S')的轨迹段来生成数据，提升强化学习策略的性能。He等[50]基于扩散模型提出了具有通用性的多任务强化学习算法(MT-Diff)。为了更高效地建模多任务数据，算法使用Transformer架构代替了传统的U-Net网络，并结合提示学习，利用多任务数据中的大量信息，在每个任务之间执行隐式信息共享，通过单个模型高效完成多任务决策，并对原始数据集进行增强，从而提升各种离线算法的性能。实验表明其具有出色的生成能力和数据建模能力。Ada等[51]为缓解离线强化学习训练期间缺乏在线交互导致的分布偏移问题，引入了一种基于扩散策略的状态重建方法(SRDP)，将状态重建特征学习纳入扩散策略中，有效促进了状态的可泛化学习。此外设计了一种多模态上下文Bandit环境来证明SRDP具有更强的泛化能力和更快的收敛速度。

强化学习的一个普遍的趋势是性能随着参数量的增加而提升，随着对大模型需求的增加，高质量的数据集成为不可

或缺的部分。与其通过昂贵的人工演示或建造仿真模拟器来收集新数据，利用大量已有的低质量数据也是一种可行途径。决策扩散模型可以在较低质量的数据上进行训练，然后使用奖励函数进行引导，以生成接近最优的轨迹。Bogdan 等[52]引入了一种简单的无模型算法，用于学习奖励最大化策略。扩散值函数(Difusion Value Function，DVF)通过扩散模型从状态序列预训练无限范围转换模型，避免了时间差异学习和基于自回归模型的方法的缺陷。该模型不需要任何动作或奖励信息，可以用来构造状态-动作价值函数，从中解码最佳动作。Felipe 等[53]通过比较模拟低奖励行为的决策扩散模型和模拟高奖励行为的决策扩散模型来考虑提取奖励函数的问题与逆强化学习相关的设置。其定义了两个扩散模型的相对奖励函数的概念，设计了一种实用的学习算法，通过将奖励函数的梯度与两个扩散模型的输出差异对齐来提取信息。该方法在给定的环境中匹配到了正确的奖励函数，并且证明了使用学习到的奖励函数指导基本模型可以显著提升标准运动基准测试的性能。

扩散模型可以很好地输出动作来进行决策，并且能够建模完整轨迹段来生成数据，促使强化学习算法更好地利用数据，进而提升策略的性能。扩散模型通过其逆向过程中每一步的噪声进行预测和建模，在一定程度上能够避免强化学习中不稳定的梯度信号的问题，使强化学习算法可以获得更加稳定的学习效果[54]。但对于需要强化学习实时控制和决策的场景而言，扩散模型相比其他生成模型的生成速度更慢，处理数据的类型也不够广泛，仍存在着一些局限性。

4.3 基于 Transformer 的强化学习

相比传统的循环神经网络模型，Transformer 在处理序列数据方面具有更好的并行性和更短的训练时间，可以在强化学习任务中处理各种序列，例如 Multi-Agent 序列、Multi-Entity 序列以及轨迹序列等，通过编码器模块来提取形态特征和学习特定任务的表示。现有工作将强化学习任务看作条件序列建模问题，可以将 Transformer 本身直接作为序列决策模型，通过自回归来生成一系列具有高回报的动作。由于 Transformer 架构在各种任务中被证明具有高效性，一些工作开始研究是否可以利用 Transformer 使通用智能体能够解决多任务或者多问题。本节基于以上工作，将现有研究分为表征学习、序列决策和通用智能体 3 个方面。

4.3.1 表征学习任务

表征学习(Representation Learning)在机器学习领域可以将输入的原始数据进行特征工程去执行特定的任务，是一种学习特征技术的方法集合。在表征学习中，引入线性组合和非线性模块，把输入数据映射到更高的层次，再使用一定次数的类别转换，可以使模型从输入数据中学习复杂的转换函数[55]。

由于强化学习任务中的时间序列、多智能体序列等数据具有顺序性质，而 Transformer 可以高效地处理序列问题，因此可以使用 Transformer 编码器去处理强化学习任务。Wu 等[56]使用基于 Transformer 和对抗生成网络的时间序列预测模型来对时间序列数据进行特征学习，模型用 Sparse Transformer 代替了 Transformer，用 α-entmax 代替了传统的 softmax 操作，将稀疏转换器当作生成器来学习稀疏注意力图，在一定程度上提高了模型的性能和效率。Lim 等[57]基于注意力机制，提出了一种新颖的预测模型 Temporal Fusion Transformer(TFT)，TFT 结合了 High-performance Multi-horizon Forecasting 与 Interpretable Insightsinto Temporal Dynamics，使用静态协变编码器把静态特征映射到网络中，再编码上下文向量来调节时间动态，从收集到的时变输入信息中学习长短期的时间关系，并通过分位数预测来获得目标值的范围。Wu 等[58]将基于 Transformer 的时间序列预测模型应用到医学检测上，使用自注意力机制来学习序列数据中的复杂信息，构建 M 步前向的预测问题，将前 N 个时间步作为输入，模型需要预测的序列为后 M 步的时间步数据。其使用最大最小缩放来处理数据，并采用特定长度的滑动窗口来构建训练信息。模型可用于单变量或多变量的序列数据以及时间序列嵌入。Tang 等[59]利用 Transformer 的注意力机制来处理感官序列并构建一个排列不变的策略，通过训练来整合本地收集的数据，在不兼容的多任务强化学习设置中，使用 Transformer 来提取形态领域知识[60]。ViT 使用 Transformer 代替卷积结构学习图像的特征信息[61]，将输入图像裁剪并进行线性映射等处理，在大规模数据集上性能超过了 CNN。Gated Transformer-XL(GTrXL)[62]是将 Transformer 应用到 RL 的一次尝试，是第一个将 Transformer 用作内存架构来处理轨迹的有效方案。GTrXL 使用 gating layer 代替了多头注意力后接的残差层，通过身份映射重新排序修改 Transformer-XL[63]架构，以提供从时间输入到 Transformer 输出的路径，有助于从一开始就稳定训练过程。CoBERL[64]提出了一种新的表征学习目标，结合了 LSTM 可以高效捕获短时间内的依赖关系的能力以及 Transformer 无最近偏差的优势，使用改进后的掩码输入预测中的自注意力机制来学习更好的特征。UPDeT[65]结合策略解耦与 Transformer，将每个观测作为各个实体的集合，可以将训练好的模型部署到多智能体的任务中，能与任意多智能体强化学习算法相结合，并通过策略解耦提升了模型的表征能力。

随着内存容量和参数规模的扩大，Transformer 的通用性能已经显著优于 LSTM 与 RNN。在强化学习中，特别是部分信息可观测的场景下，可以利用特征构造减轻信息不全带来的影响。具体来说，Transformer 可以被用于结构化状态表示的关系推理，从而提取实体之间的关系，更好地进行策略学习。在实际应用中，研究人员可以根据具体问题的需求和场景，灵活地将 Transformer 架构应用于强化学习任务的表征提取中。

4.3.2 顺序决策任务

强化学习的标准处理依赖于将长期问题分解为更局部的子问题，在无模型算法中通常采用最优性原则的形式，这是一种动态规划方法(如 Q 学习)。在基于模型的算法中，一般采用单步预测模型的形式，将预测高维、依赖于策略的状态轨迹的问题简化为估计相对简单的、与策略无关的转移分布的问题。相关研究表明，可以不使用传统的策略梯度法或时序差分法来解决强化学习问题，而是将其作为一个序列生成问题，将 Transformer 本身直接作为序列决策模型，通过将轨迹

视为状态、动作和奖励的非结构化序列来研究，结合 Transformer 优异的序列建模能力来解决问题。

Transformer 在 NLP 和 CV 等领域已经取得了优异的成绩，不少研究者也将注意力投向强化学习领域。其中，离线强化学习(Offline RL)由于可以在大量的离线数据集上进行训练而备受关注，研究者将状态、动作和奖励作为简单的数据流，再利用大容量序列模型来解决问题。Decision Transformer(DT)对离线数据中的奖励 r(rewards)、状态 s(states)以及动作 a(actions)等轨迹进行联合分布建模，并引入了 $\hat{R}_t$(return-to-go)[66]方法作为训练中的回报函数来代表 t 时刻后的所有 reward 之和，建模后的轨迹可以表示为 $\tau=(\hat{R}_1, s_1, a_1, \hat{R}_2, s_2, a_2, \cdots, \hat{R}_T, s_T, a_T)$，如图 6 所示。

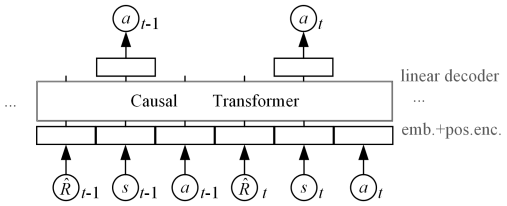

图 6 Decision Transformer 结构图

Fig. 6 Decision Transformer architecture

DT 将编码后的数据信息输入到设定的 GPT 模型中，通过因果自注意掩码(Causal Self-Attention Mask)自回归地预测动作。模型相比传统的强化学习，避免了折扣奖励所带来的短视影响，可以适应多模态数据。Trajectory Transformer(TT)与 Decision Transformers 是同时期的研究成果，都是将强化学习问题作为序列生成任务，再通过序列建模去解决。TT 的核心任务同样是对离线数据上的轨迹分布进行序列建模，在训练期间使用 BeamSearch 进行规划，但将其作为奖励最大化函数有导致短视的贪心策略风险。为解决此问题，模型同样将 return-to-go 增加到训练轨迹中，并将其作为一个附加量包含在内，与其他量一同进行离散化。通过将轨迹 $\tau=(s_1, a_1, r_1, s_2, a_2, r_2, \cdots, s_T, a_T, r_T)$离散化为 $\tau=(s_t^1, s_t^2, \cdots, s_t^N, a_t^1, a_t^2, \cdots, a_t^M, r_t, \cdots)$，原始轨迹的长度由 T 变为 $T(N+M+1)$，如图 7 所示。

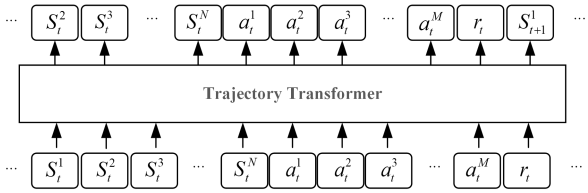

图 7 Trajectory Transformer 结构图

Fig. 7 Trajectory Transformer architecture

实验表明，TT 具有优异的长时间步预测能力，可以在同一框架下执行模仿学习、目标条件强化学习和离线强化学习。Wang 等[67]为了解决离线强化学习中数据集规模不足的问题，研究了 Bootstrap Transformer 的算法。该算法利用自举(Boostrap)的方式，使用自回归(Autogressive)或 Teacher Forcing 方法生成更多的轨迹数据，以此来提升训练性能。Furuta 等[68]提出的 Hindsight Information Matching(HIM)，通过学习策略 $\pi(a|s,z)$使轨迹满足式(6)。

$$\min_{\pi} E_{z\sim p(z),\tau\sim\rho_z^{\pi}(\tau)}\left[D(I^{\phi}(\tau),z)\right] \tag{6}$$

其中，D 代表散度，$I^{\phi}(\tau)$为轨迹所包含的信息。该工作提出了用于任意选择统计数据的广义 DT(GDT)，展示了它在离线多任务状态边际匹配和模仿学习中的应用。Yamagata 等[69]在 DT 的基础上进行进一步改进，提出了 Q-Learning Decision Transformer，使用保守值函数对数据集中的 return-to-go 进行重新标记，从而将 DT 与动态规划相结合并提高其拼接能力，在一定程度上缓解了 DT 泛化能力弱以及 return-to-go 难设置的问题。Xu 等[70]基于轨迹提示的方法，将一段能够提示任务特性的数据序列片段 $\tau_i^*=(\hat{r}_1^*, s_1^*, a_1^*, \hat{r}_2^*, s_2^*, a_2^*, \cdots, \hat{r}_K^*, s_K^*, a_K^*)$插入 DT 原本的输入序列之前，针对离线元强化学习(Offline Meta-RL)无法在一个统一的 Task Domain 中进行泛化等问题，提出了 Prompt-DT 架构，其性能大大优于元强化学习算法。合理的提示学习方法在生成式强化学习中发挥着重要作用，常见的提示学习方法如表 3 所列。

表 3 常见的 Prompt Learning 方法

Table 3 Common Prompt Learning methods

Type	Task	Input([X])	Template	Anwser([Z])
Text CLS	Sentiment	I love this book	[X] The book is [Z]	Great Wanderful …
	Intention	What is the price of this GPU?	[X] The question is about [Z]	quantity Product
	Topics	He prompted the LM	[X] The text is about [Z]	sports Science
Text-pair CLS	NLI	[X1]:An old man with… [X2]:A man walks…	[X1]? [Z],[X2]	Yes No
Text-span-CLS	Aspect Sentiment	Terrible weather but hot business	[X] What about weather? [Z]	Bad Awful …
Tagging	NER	[X1]:Wang went to Shanghai. [X2]:Shanghai	[X1][X2] is a [Z] entity	organization location …
Text Generation	Summarization	Las Vegas police…	[X] TL;DR:[Z]	The victim… A woman… …
	Translation	Je vous aime	French:[X] English:[Z]	I love you. I fancy you …

Laskin 等[71]提出的 Algorithm Distillation 算法，将训练的历史信息通过因果序列模型来进行建模，优化预测损失来学习上下文的策略，将强化学习算法提取到神经网络中。不同于专家序列和蒸馏学习，Algorithm Distillation 通过建模包含模仿损失的离线数据来进行上下文强化学习，可以在基于上下文改进策略的情况下不更新网络中的参数。先前的 Transformer 大多数都在时间范畴上使用，Meng 等[72]将目光转向空间维度上，提出了 Swin Transformer 架构，将非标准固定的图像分割成小块像素，再利用局部自注意力机制在固定大小的位移内操作。通常，在模型训练中会使用 ResNet 的预训练、网络参数的特殊初始化、对比表示学习以及分布式训练等方法。Mao 等[73]从强化学习的网络结构模型上出发，为了在取得同等性能效果的情况下不使用这些复杂的优化技巧，提出了 Transformer in Transformer(TIT)架构，其由 Inner Transformer 与 Outer Transformer 组成，分别用于提取空间信息处理单个观测，以及提取时间信息来处理观测历史。实验结果表明，TIT 只需要跟随强化学习进行训练更新，不使用复杂的优化方法即可实现较好的效果。Hu 等[74]基于 Transformer 架构提出 Graph Decision Transformer(GDT)，其将离线强化学习中的输入序列建模为因果图来对动作进行预测，获取不同概念数据间的依赖关系，在一定程度上缓解了 Transformer 因关注所有令牌信息而无法捕获依赖关系，从而导致限制长期依赖性学习的问题。

强化学习任务在实际中通常是部分可观测的，目前的方法通常是将循环神经网络与强化学习架构相结合。但 RNNs 训练难度大，需要在每个训练时间步内初始化隐藏状态。为此 Esslinger 等[75]提出了 Deep Transformer Q Network，其通过 Transformer 架构和自注意力机制代替以往的循环层对智能体的历史信息进行编码，达到更快收敛且更稳定的效果。目前大多数基于 Transformer 进行序列转化的方法都是基于离线强化学习的，然而强化学习即时化通常涉及在线部分，通过与环境的交互对离线数据集上预训练的策略模型进行微调。Zheng 等[76]提出了 Online Deciosion Transfomrer(ODT)，在一个统一的框架中融合了离线预训练与在线微调，将 DT 中的确定性策略替换为随机性策略，并定义轨迹级策略熵来促进在线微调期间的探索，将序列正则化器与自回归建模目标相结合，实现样本高效的学习探索与微调。Zhu 等[77]研究了从无动作离线数据集中提取知识来训练在线学习的方法，提出了 AF-Guide 模型。该模型由一个 Action Free Decision Transfomrer 组成，将数据集中的轨迹序列建模为 $\tau=(s_1,r_1,s_2,r_2,\cdots,s_T,r_T)$，从离线数据集中学习预测下一时间步的状态。基于未来奖励的 return-to-go 方法给予了序列训练很好的引导，但是其他类型的后验信息能否被用来提升顺序决策能力还需要继续探讨与研究。

当作为一个在传统强化学习框架下训练的表示模块时，Transformer 架构的优化通常是不稳定的。当使用 Transformer 通过序列建模来解决决策问题时，监督学习范式可以有效消除 deadly triad 问题。在监督学习的框架下，策略的性能深受离线数据质量的约束，exploitation 和 exploration 之间的明确权衡不复存在，因此使用 Transformer 结合强化学习和监督学习时，模型通常能够生成更好的策略。

4.3.3 多模态任务

Transformer 在 CV 和 NLP 领域已经展现了强大的优势，Decision Transformer 等模型也在离线强化学习任务中取得了显著的效果，在其基础上不断改进和优化的各类算法也在各类任务中大放异彩。Transformer 架构已经应用到众多领域，迈向多模态、多任务的研究工作也陆续展开。一些工作借鉴了 CV 和 NLP 中对大规模数据集进行预训练的思想，并尝试从大规模多任务数据集中抽象出通用策略。Hu 等[65]将多智能体强化学习算法运用到游戏场景中，并基于 Transformer 架构提出了一种通用的策略解耦模型 UPDeT，其将得到的策略分布与输入数据解耦，再利用自注意力机制获取权重信息。该模型可以结合多种多智能体强化学习任务，拥有强大的泛化能力，在训练速度和性能上展现出了明显的优势。Lee 等[78]研究了是否能使用相同的策略训练出通才型强化学习智能体，并基于 Decision Transformer 架构提出了 Multi-Game Decision Transformer(MGDT)。该模型在训练中使用的数据集由专家数据和非专家数据组成，并内置了一个二元分类器来判断在某一时刻是否需要从 return-to-go 的先验分布计算出专家级的后验分布，并根据贝叶斯公式成比例地预设专家级返回概率。模型除展现了良好的性能，还证明了强化学习多任务同模型的可行性。DeepMind 进一步提出了一种多模态、多任务、多实施例的通才策略 Gato[79]，在包含大量图像、自然语言以及时间决策等多模态数据集上进行序列建模来进行训练。该模型把数据序列转化为 token 序列，将 RL，NLP 以及 CV 等领域结合到一起，能够执行来自不同领域的一系列任务，包括文本生成和决策制定。具体来说，Gato 统一了共享标记化空间中的多模态序列，并在部署中采用基于提示的推理来生成特定于任务的序列。元强化学习可以学习一组通用的策略信息，从而在其他多种目标任务中快速学习较优的策略。Melo[80]提出了一种基于记忆网络的元强化学习算法 TrMRL，其结合历史记忆来动态地表示任务并递归地创建 Episode Memory，在基于 Transformer 的架构上使用 Multi-Head Self Attention 机制来快速学习策略。

通用智能体一个重要能力的体现就是可以在开放的任务环境中高效地学习知识。北京大学和北京志源人工智能研究院的团队使用开放世界游戏《我的世界》(Minecraft)作为研究的测试环境[81]，智能体在该环境中需要通过局部的信息来获取资源，面临着信息探索率低、奖励稀疏等问题。研究者通过强化学习和动态规划将任务拆分为学习基本技能和技能规划两个阶段，结合内在回报函数，使用强化学习来训练，并且利用生成式预训练模型来构造游戏中的技能关系图，再基于其生成结果搜索得到任务规划。同样是以 Minecraft 来模拟真实环境，Ghost in the Minecraft(GITM)[82]在游戏中取得了比以往智能体更优异的表现。传统的 RL 智能体的性能瓶颈在于如何高效地将极为复杂的任务映射到最底层的键盘鼠标操作。GITM 采用生成式预训练模型作为智能体核心的新范式，首先利用外部知识将复杂任务分解为简单的子任务，为每个子任务制定结构化动作；然后根据反馈的信息调整规划，在与环境的交互过程中观察信息，不断通过学习成功的经验来

提升自身；最后再使用底层的键盘鼠标操作来执行结构化动作。GITM 可以进一步应用在 Minecraft 更加复杂的任务中，展现了强大的能力和可扩展性，使得智能体能够在游戏中长时间生存、发展，探索更高级的世界。

离线强化学习使用已有的数据集来训练，随着其不断发展，出现了越来越多的高质量数据集。更多样化和更丰富的模拟环境为强化学习的发展奠定了基础。在更广泛、更实际的场景中结合预训练生成模型来训练更强大的通用智能体已经成为强化学习的发展方向之一。

4.4 大语言模型中的强化学习

人工反馈的强化学习(Reinforcement Learning with Human Feedback，RLHF)可以在智能体的训练中通过人类的反馈信息来帮助模型学习，将人类专家的思想信息和决策经验结合到模型学习中来提高训练效果。RLHF 的训练过程可以分为 3 个阶段，分别为收集人工反馈信息来预训练一个语言模型，训练奖励模型，以及使用强化学习算法来优化策略。在第一个阶段会预训练一个语言模型，比如 OpenAI 发布的 InstructGPT 和 ChatGPT 中都使用了 GPT-3 的网络结构。第二个阶段会基于预训练的语言模型来训练奖励模型，奖励模型也可以是其他经过微调后的语言模型。Anthropic 使用偏好模型预训练(Preference Model Pretraining，PMP)来优化预训练模型，而无需再进行微调。奖励模型可以反映出语言模型的输出信息在人类专家角度上的表现，模型的输入为提示(Prompt)以及生成的文本，模型的输出为反映文本表现的标量奖励数字。损失函数如式(7)所示：

$$loss(r_\theta)=-E_{(x,y_0,y_1,i)\sim D}[\log(\sigma(r_\theta(x,y_i)-r_\theta(x,y_{1-i})))] \tag{7}$$

其中，x，y 代表 post 和 summary，r 代表参数为 θ 的奖励模型的值，σ 表示 sigmoid 函数。第三个阶段则是使用奖励模型的输出，通过强化学习方法来微调语言模型。首先将微调任务建模为强化学习问题，将 token 的输出位置作为动作空间，输入的 token 序列作为观察空间，将奖励模型中的初始 reward 加上特定的约束项作为奖励函数。通过离线 RLHF 进行语言模型的对齐优化框架如图 8 所示。

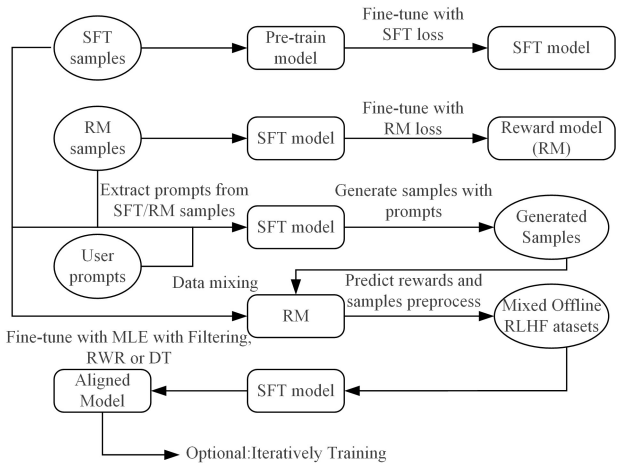

图 8 离线 RLHF 框架

Fig. 8 Offline RLHF framework

Bai 等[83]使用 RLHF 来微调语言模型，并提出一种迭代的在线训练方式，在特定时间间隔内对偏好模型和策略进行更新，并探索了强化学习奖励和策略以及初始化之间 KL 散度的线性关系。Ramamurthyr 等[84]将人类偏好与预训练大语言模型高效地匹配，基于开源模块库 RL4LMs 提出了通用的强化语言理解基准，其能够获取自动评估的人类偏好回报函数，再使用自然语言策略优化(NLPO)算法来学习如何对语言生成过程中的组合动作空间进行优化处理。然而，Gao 等[85]指出，根据古德哈特定律，过度优化奖励模型的值会影响实际表现，并对此提出一个固定的“黄金标准”奖励模型来代表人类专家策略，通过不同的优化方法来观察模型的得分情况。

ChatGPT 和 InstructGPT 会在人工生成的数据上训练模型，通过收集提示数据下的不同回复，将每两条回复信息作为一组训练样本并由人工进行标注，再根据每组数据间的奖励值差来拟合人工标签，以此来训练模型。之后通过强化学习将模型的参数作为策略来训练，奖励模型会根据策略的采样回复生成一个奖励，再将其反馈给模型以对策略进行优化；同时，为避免对策略的过度优化，引进了 KL 惩罚项。与 ChatGPT 和 InstructGPT 不同，Glaes 等提出 Sparrow 架构，将奖励模型进一步分为 Rule Reward 模型和 Preference Reward 模型两种，两者都是基于人类专家的价值反馈。前者可以观察生成的文本是否符合设置的规则，控制智能体的道德准则；后者则可以在生成的多个候选回复中找出最合适的文本选项，从而生成最优答案。对于 GPT-3 和 GPT-4 等大规模预训练语言模型来说，除了是否具备严密的逻辑思维能力外，能否处理异构信息等复杂的问题也需要探讨。UCLA 的研究人员建立了 Tabular Math WordProblems(TabMWP)[87]数据集，需要对数据集中的文本以及表格数据进行逻辑计算得到问题答案；并提出了 PromptPG 方法，该方法可以将 TabMWP 中选项的选择同步为 RL 中的上下文老虎机(Contextual Bandit)问题，再通过强化学习中的策略梯度算法来学习一个高效的策略。

RLHF 在预训练大语言模型中展现了不可替代的作用，但仍然只作为辅助的方法。强化学习领域依然缺少基础模型。DeepMind 的 Adaptive Agents 团队提出了 Adaptive Agent(ADA)[88]——一种自适应智能体，在对其进行大规模训练后可以在一个大规模的开放式任务空间中使其具有上下文学习的能力，只与环境任务进行少量的交互就能高效地优化策略。模型的训练可分为 3 个部分，首先让智能体学习设定的信息，通过自动化的方法选择相关任务对 No-Opfiltering 和 Prioritised Level Replay(PLR)技术进行扩展，使智能体在提高采样效率的同时能够自主选择越来越复杂的任务；再使用基于模型的强化学习算法训练智能体，让 ADA 能够预测未来时间步的价值以及奖励等信息；最后对模型蒸馏，以实现扩展。虽然 RLHF 等技术有着广泛的应用和关注，但收集人类偏好等数据的成本依旧很高，并且人工进行标注的标准很难绝对一致；同时 RLHF 高度依赖于标注的质量，在无事实训练数据上存在着一定的潜在变量。

大模型使用大规模数据来进行训练，在极高的训练成本下赋予了模型强大的通用性。人类反馈强化学习以及

Transformer 模型可以实现目标对齐并提高智能体的推理能力，进而保障模型生成内容的质量，使模型生成的数据更加安全可控。大模型从通用到专用的技术转变，将成为大模型发展的下一个制高点。

4.5 算法分析与总结

深度神经网络给强化学习带来了强大的感知能力，能够有效地处理高维数据。经典的 DQN，PPO 和 Actor-Critic 算法能够在高维度空间下进行策略学习。但其高度依赖 MDP 模型，在整体环境信息不完全可观测的情况下表现不佳。此时引入循环神经网络，其记忆功能可以帮助智能体获取更为准确的环境信息，在处理序列数据方面表现更好。扩散模型可以用于对环境动态建模，以生成虚拟轨迹来指导策略学习。这种方法可以提高强化学习的样本效率，并加速训练过程。此外，扩散模型还可以用于构建基于模型的强化学习方法，通过预测未来的状态和奖励来指导智能体的决策[89]。

Transformer 相比传统的神经网络进行了较大的改进，在与强化学习结合后能够更好地处理序列数据并高效地捕获表征信息，具有更好的建模能力和计算效率，可直接作为决策生成模型应用到多个任务和领域。但随着其规模的不断扩大，模型的计算推理成本和训练代价显著提高。Parisottop 等[90]提出了一种 Actor-Learner Distillation(ALD)方法，其利用单独的 Actor 和 Learner 模型之间连续形式的模型压缩，通过连续蒸馏将 Transformer 模型作为学习器蒸馏到作为 Actor 的 LSTM 中，在具有 Transformer 高效性能的同时能够保持合理的计算成本。Effective Transformer 通过智能批处理策略，将序列进行批量调整，删除单个批次中的 padding。GOBO 将训练后量化(ONNX)结合到 Transformer 架构上，模型通过混合精度对权重参数以及激活函数进行量化，高精度使用浮点型数据，低精度则使用整型数据表示，通过观察权重的均值和标准差来获取异常值，以此提高模型的性能。通过 Transformer 来获取强化学习中决策序列的依赖关系需要大量的离线数据，但在实际中大部分的任务都很难获取到高质量的离线数据，需要从在线学习中获取环境状态等信息。目前的大多数研究都集中在离线学习框架中，如何将在线强化学习与 Transformer 有效结合仍是一个有待探讨的问题。

表 4 基于序列建模的强化学习算法及模型总结

Table 4 Summary of reinforcement learning algorithms and models based on sequence modeling

分类	典型算法/模型	算法/模型简述	局限性
基于神经网络的强化学习	DQN	融合神经网络和 Q 值来选择最优动作	经验数据的存储内存有限，依赖于完整的观测信息
	PPO	一种新型的 Policy Gradient 算法，通过提出新的目标函数可以在多个训练步骤实现小批量的更新，缓解了步长难以确定的问题	可能存在探索性不足的问题，容易陷入局部最优，数据效率和算法鲁棒性不足
	AC	Actor 使用策略函数生成动作与环境交互，Critic 作为价值函数来评估 Actor 的表现	Critic 难收敛，两个网络更新前后都存在相关性，使得模型学习效率低
基于循环神经网络的强化学习	DRQN	将 DQN 卷积层后一层的全连接层替换为了 LSTM 网络，最终输出结果为每个动作 a 对应的 $Q(s,a)$ 值。	不能并行计算，序列长度超过一定限度后仍然存在梯度消失现象
	RL^2	通过 RNN 接收 RL 算法的所有信息来构造 agent，隐藏层在每个 episode 开始后依然保留，再使用标准 RL 算法来训练 agent	对于长时间范围的任务，该策略性能不足，对外层的 RL 算法依赖高
	GBAC	将基于 glimpse 的 hard 硬注意力机制与无模型 RL 算法相结合	模型的决策与人类偏好可能存在差异
基于 Transformer 的强化学习	GTrXL	调整了 layer normalization 在 Transformer Block 中的位置，使用 gating layer 代替了 Multi-Head Attention 和 Multi-Layer Perception 后接的残差层	GTrXL block 不能参数共享，有进一步提高的空间
	UPDeT	通过 transformer 模型将策略分布与观测信息解耦来生成策略，使用自注意力机制确定权重值	将个体的 observation 视为 entity 时，拆分动作空间易丢失部分信息
	DT	将 RL 轨迹进行联合分布建模，建模后的轨迹为 $\tau=(\hat{R}_1,s_1,a_1,\hat{R}_2,s_2,a_2,\cdots,\hat{R}_T,s_T,a_T)$，通过 causal self attention mask 自回归地预测动作	可解释性不足，局限于离线数据集，对于 return-to-go 需要引入先验信息
	TT	对离线数据上的轨迹分布进行序列建模，将其离散化为 $\tau=(s_t^1,s_t^2,\cdots,s_t^N,a_t^1,a_t^2,\cdots,a_t^M,r_t,\cdots)$，在生成结果中寻找最优序列	对数据集的规模大小有一定的要求，在连续的环境下进行离散化比较繁琐
	Gato	具有多模态、多任务、多具身的特点，能够处理多个领域的任务。其中的 RL 部分只采用监督学习方法，不涉及 reward 设计机制	仅通过扩大模型规模无法确保计算结果的正确性，不能从根本上提升模型的性能，计算资源消耗过多
大模型中的强化学习	RLHF	通过人工标注的信息给模型提供正反馈或负反馈来引导模型学习	依赖于标注数据的质量，人工标注的成本较高
	PromptPG	将示例的选择转化为 RL 中的 contextual bandit 问题，通过策略梯度法训练策略网络来学习从少量训练数据中选择的最优 in-context 示例	模型缺乏高级逻辑推理能力，无法理解一些特定领域问题
	Adaptive Agent	通过假设驱动的探索行为使用即时获取的信息来改进策略，从而实现接近最佳的性能	无法突破 ADA 模型任务分配的固有局限性

5 应用与发展

从前馈神经网络到循环神经网络，再从扩散模型、Transformer 架构到 GPT 等大模型，强化学习不断地与前沿技术相结合发展，旨在增强智能体的泛化能力并提高解决问题的效率。

生成式人工智能在自然语言生成以及图像生成等方向已经获得了广泛的应用。通过对数据源进行分析,生成式人工智能能够学习自然语言和图像的规律,从而生成逻辑性强的文本或图像[91]。在应用方面,生成式人工智能可以分析通用语言来生成日常的文字和对话,可以学习图像的特征来生成逼真的人物画像和风景照等[92]。在游戏领域,生成式强化学习可以通过自主学习和智能决策来帮助智能体在游戏中获取更高的得分[93],可以使用生成式强化学习算法来训练智能体,通过对智能体进行模拟和优化实时生成智能策略使它们的实力相当,保持游戏的动态平衡。在机器人技术领域,生成式强化学习可以帮助机器人更好地适应环境和任务,提高机器人的灵活性和适应性。在资源优化领域,生成式强化学习可以通过智能调度和决策来优化资源的分配和利用,提高资源的利用率和效益。在各类生成式预训练模型中,强化学习也发挥着不可替代的作用。例如在生成式对话中,通过使用强化学习方法来优化模型的策略,以生成更符合用户需求的对话[94]。此时 GPT 模型作为策略函数,强化学习算法则用来更新模型的参数。从技术角度分析,强化学习等算法也将更普遍地应用于生成式人工智能中。除目前火热的 GPT 和 BERT 大语言模型外,国内也迅速跟上这股技术潮流,目前已经研发出百度文心大模型、阿里通义大模型、腾讯混元大模型以及华为盘古大模型等大语言模型。

生成式强化学习已逐步应用到各大领域[95],但目前面临着一系列亟需解决的瓶颈问题。算法需要平衡探索和学习之间的关系,进一步提升探索效率和样本利用率,从而更高效地找出最优解,但是如何准确获取平衡点仍然是一个需要深入研究的问题。由于强化学习学习到的策略是侧重于从状态映射到最高回报函数时的动作,包含极端事件的策略也有可能符合期望,因此目标函数的策略安全性和稳定性无法同时保障。常用的缓解措施是使用基于模型的算法以及预训练等方法约束智能体的学习规则来保障安全[96]。对于回报函数的设置问题,在特定的任务中可以使用逆强化学习,根据示范直接学习奖励。此外,智能体的迁移问题也是生成式强化学习中的一个重要问题,良好的迁移能力可以提高模型的采样效率并降低训练成本,利用之前任务中的知识来提高智能体在新环境中的表现。不同于自然语言处理以及计算机视觉等任务,强化学习任务具备低维结构,并且不包含先验信息,因此需要更有效地提高模型的可迁移性和泛化能力。想要把强化学习运用到特定的决策领域中,通常需要建立一个有效的 MDP 过程,但实际应用场景中通常存在的信息部分可观测、非平稳过程等问题,对传统的强化学习算法性能有所影响[97]。此外,生成式强化学习对数据质量有着较高的要求,并且普遍存在缺少特定任务需求的问题。目前的生成式强化学习算法往往只能应用于特定的任务和环境,缺乏通用性。如何设计一种通用的生成式强化学习算法,使其适用于不同的任务和环境,也是当前的研究热点之一。因此,其所代表的决策类问题如何去构建具有广泛性和应用性的大型预训练模型也需要进行深入的研究与探讨。

结束语 近年来,以 Transformer 为代表的生成式预训练方法在自然语言处理等领域取得了重大进展。基于该架构极强的长时间依赖关系能力与良好的预训练效果,强化学习可以通过序列建模更有效地学习序列数据中的有效信息,再将这些序列信息转换成能够用于决策的策略。与传统的强化学习算法相比,生成式强化学习具有更好的稳定性和表现能力,可以适用于更复杂的环境和任务。随着人工智能新兴技术的涌现以及强化学习研究的不断深入发展,未来生成式强化学习将会在各个领域发挥更重要的作用,高效地解决更复杂的问题。

参 考 文 献

[1] RADFORD A,NARASIMHAN K,SALIMANS T,et al. Improving Language Understanding by Generative Pre-Training[J]. Computation and Language,2017,4(6):212-220.

[2] MNIH V,KAVUKCUOGLU K,SILVER D,et al. Playing Atari with Deep Reinforcement Learning[C]// Proceedings of the Deep Learning Workshop at NIPS. San Diego:NIPS,2013:812-826.

[3] VASWANI A,SHAZEER N,PARMAR N,et al. Attention is All You Need[C]// Advances in Neural Information Processing Systems. San Diego:NIPS,2017:5998-6008.

[4] CHEN L,LU K,RAJESWARAN A,et al. Decision Transformer:Reinforcement Learning via Sequence Modeling[C]// International Conference on Learning Representations. Washington DC,2021:3307-3319.

[5] JANNER M,LI Q,LEVINE S. Reinforcement Learning as One Big Sequence Modeling Problem[C]// Proceedings of the Annual Conference on Neural Information Processing Systems. San Diego:NIPS,2021:1213-1225.

[6] LI H,UMAR N,CHEN R,et al. Deep Reinforcement Learning [C] // ICASSP 2018-2019IEEE International Conference on Acoustics,Speech and Signal Processing. NewYork: ICASSP,2018:2432-2449.

[7] HOPFIELD J J. Neural networks and physical systems-with emergent collective computational abilities[J]. Proceedings of the National Academy of Sciences of the United States of America,2018,79:2554-2558.

[8] BAHDANAU D,CHO K,BENGIO Y. Neural Machine Translation by Jointly Learning to Align and Translate[C]// International Conference on Learning Representations. Washington DC,2015:1409-1420.

[9] URIEL S,HAGGAI R,YOTAM E,et al. Sequential Modeling with Multiple Attributes for Watchlist Recommendation in E-Commerce [C] // Proceedings of the 15th ACM International Conference on Web Search and Data Mining(WSDM). 2022:937-946.

[10] SCHULMAN J,WOLSKI F,DHARIWAL P,et al. Proximal Policy Optimization Algorithms[C]// Advances in Neural Information Processing Systems. San Diego:NIPS,2017:2054-2068.

[11] SILVER D,LEVER G,HESS N,et al. Deterministic Policy Gradient Algorithms [C] // International Conferenceon Machine Learning. New York:ICML,2014:1892-1904.

[12] SKORDILIS E, MOGHADDASS R, FARHAT M T, et al. A Generative Reinforcement Learning Framework for Predictive Analytics[C]// 2023 Annual Reliability and Maintainability Symposium(RAMS. 2023: 1-7.

[13] ZHAO S Y, GROVER A. Decision Stacks: Flexible Reinforcement Learning via ModularGenerative Models[C]// Proceedings of the 37th International Conference on Neural Information Processing Systems(NIPS '23). 2023: 80306-80323.

[14] GOODFELLOW I, POUGET J, MIRZA M, et al. Generative Adversarial Nets[C]// Neural Information Processing Systems MIT Press. San Diego: NIPS, 2014: 3844-3852.

[15] ZHANG B, SENNRICH R. A Lightweight Recurrent Network for Sequence Modeling[C]// Proceeding of the 57th Annual Meeting of the Association for Computational Linguistics. 2019: 1538-1548.

[16] BO P, ERIC A, QUENTIN A, et al. RWKV: Reinventing RNNs for the Transformer Era[J]. arXiv. 2305. 13048, 2023.

[17] KHOI M N, QUANG P, BINH T N. Adaptive-saturated RNN: Remember more with less instability[J]. arXiv: 2304. 11790, 2023.

[18] HOCHREITER S. The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions[J]. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 1998, 6(2): 107-116.

[19] CHEN J K, QIU X P, LIU P F, et al. Meta Multi-Task Learning for Sequence Modeling[C]// Proceedings of the AAAI Conference on Artificial Intelligence. Menlo Park: AAAI, 2018.

[20] LIU Y J, MENG F D, ZHANG J C, et al. GCDT: A Global Context Enhanced Deep Transition Architecture for Sequence Labeling[C]// Annual Meeting of the Association for Computational Linguistics. Stroudsburg: ACL, 2019: 426-436.

[21] SUTSKEVERI, VINYALS O, LE Q. Sequence to Sequence Learning with Neural Networks[C]// Advances in Neural Information Processing Systems 34—35th Conference on Neural Information Processing Systems. San Diego: NIPS, 2016: 3844-3852.

[22] CHO K, MERRIENBOER B, GULCEHRE C, et al. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation[C]// Proceedings of the 2014 Conference on Empirical Methodsin Natural Language Processing. Stroudsburg: ACL, 2014: 96-112.

[23] CHAUDHARI S, MITHAL V, POLATKANG, et al. An Attentive Survey of Attention Models[J]. ACM Transactions on Intelligent Systems and Technology(TIST), 2021, 12(5): 1-32.

[24] TOOMARIAN N, BARHEN J. Fast temporal neural learning using teacher forcing[C]// IJCNN-91-Seattle International Joint Conference on Neural Networks. 1991: 817-822.

[25] LIN Z, FFENG M, SANTOS C, et al. A Structured Self-attentive Sentence Embedding[J]. arXiv: 1703. 03130, 2017.

[26] WICKENS C. Attention: Theory, Principles, Models and Applications[J]. International Journal of Human-Computer Interaction, 2021, 37(5): 403-417.

[27] CORDONNIER J, LOUKAS A, JAGGI M. Multi-Head Attention: Collaborate Instead ofConcatenate[J]. arXiv: 2006. 16362, 2020.

[28] SUNDERMEYER M, SCHLUTER R, NEY H. LSTM Neural Networks for Language Modeling[C]// Annual conference of the International Speech Communication Association. Baixas: ISCA, 2012: 106-119.

[29] LIU Y, SHAO Z, HOFFMANN N. Global Attention Mechanism: Retain Information to Enhance Channel-Spatial Interactions[J]. arXiv. 2112. 05561, 2021.

[30] LUONG M, PHAM H, MANNING C. Effective Approaches to Attention-based Neural Machine Translation[C]// Conference on Empirical Methods in Natural Language Processing. Stroudsburg: ACL, 2015: 2067-2081.

[31] ROMBACH R, BLATTMANN A, LORENZ D, et al. High-Resolution Image Synthesis with Latent Diffusion Models[C]// 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR). 2022: 10674-10685.

[32] MING K, JAESEOK J, YOUNG J. Diffusion Models already have a Semantic Latent Space[C]// International Conference on Learning Representations(ICLR). 2023: 312-325.

[33] LIU L, LIU X, GAO J, et al. Understanding the Difficulty of Training Transformers[C]// Conference on Empirical Methods in Natural Language Processing. Stroudsburg: ACL, 2020: 1667-1679.

[34] KALYAN K, RAJASEKHARAN A, SANGEETHA S. AMMUS: A Survey of Transformer-based Pretrained Models in Natural Language Processing[J]. arXiv: 2108. 05542, 2021.

[35] ZHANG C, LI C, ZHANG C, et al. One Small Step for Generative AI, One Giant Leap for AGI: A Complete Survey on ChatGPT in AIGC Era[J]. arXiv: 2304. 06488, 2023.

[36] DEVLIN J, CHANG M, LEE K, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[C]// Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg: ACL, 2019: 4171-4186.

[37] GARG D, HEJNA J, GEIST M, et al. Extreme Q-Learning: MaxEnt RL without Entropy[J]. arXiv: 2301. 02328, 2023.

[38] WANG Y T, PAN Y H, YAN M, et al. A Survey on ChatGPT: AI-Generated Contents, Challenges, and Solutions[C]// IEEE Open Journal of the Computer Society. 2023: 1-20.

[39] WU Y N, ZHOU Q, ZHANG T H, et al. Discovery of Potent, Selective, and Orally Bioavailable Inhibitors against Phosphodiesterase-9, a Novel Target for the Treatment of Vascular Dementia[J]. Journal of Medicinal Chemistry, 2019, 62(8): 4218-4224.

[40] BILGRAM V, LAARMANN F. Accelerating Innovation With Generative AI: AI-Augmented Digital Prototyping and Innovation Methods[J]. IEEE Engineering Management Review, 2023, 51(2): 18-25.

[41] XU H, JIANG L, LI J. et al. Offline RL with No-OOD Actions: In-Sample Learning viaImplicit Value Regularization[J]. arXiv: 2303. 15810, 2023.

[42] WANG H N,LIU N,ZHANG Y Y,et al. A Reviewof Deep Reinforcement Learning[J]. Frontiers of Information Technology & Electronic Engineering,2020,21(12):63-82.

[43] MNIH V,KAVUKCUOGLU K,SILVER D,et al. Human-level control through deep reinforcement learning[J]. Nature,2015,518:529-533.

[44] HAUSKNECHT M,STONE P. Deep Recurrent Q-Learning for Partially Observable MDPs[C]// AAAI Fall Symposium on Sequential Decision Making for Intelligent Agents. Menlo Park:AAAI,2015:1-9.

[45] DUAN Y,SCHULMAN J,CHEN X. et al. RL^2:Fast Reinforcement Learning via SlowReinforcement Learning[J]. arXiv:1611.02779,2016.

[46] LI X,LI L,GAO J. et al. Recurrent Reinforcement Learning:A Hybrid Approach[J]. arXiv:1509.03044,2015.

[47] STAMATELIS G,KALOUPTSIDIS N. Active hypothesis testing in unknown environments using recurrent neural networks and model free reinforcement learning[J]. arXiv:2303.10623,2023.

[48] QUERIDO G,SARDINHA A,MELO F. Learning to Perceive in Deep Model-Free Reinforcement Learning[J]. arXiv:2301.03730,2023.

[49] D'ALONZO M,RUSSELL R. Symmetry Detection in Trajectory Data for More Meaningful Reinforcement Learning Representations[C]// Appears in Proceedings of AAAI FSS-22 Symposium. Menlo Park:AAAI,2022:1452-1468.

[50] HE H,CHEN J B,XU K,et al. Diffusion Model is an Effective Planner and Data Synthesizer for Multi-Task Reinforcement Learning[J]. arXiv:2305.18459,2023.

[51] ADA S. E,OZTOP E,EMRE U. Diffusion Policies for Out-of-Distribution Generalization in Offline Reinforcement Learning[J]. arXiv:2307.04726,2023.

[52] BOGDAN M,WALTER A,BAUTISTAM,et al. Value function estimation using conditiona ldiffusion models for control[J]. arXiv:2306.07290,2023.

[53] FELIPE N,TIM F,JOAO F. Extracting Reward Functions from Diffusion Models[J]. arXiv:2306.01804,2023.

[54] SARTHAK M,ORBINIAN A,STEFAN B,et al. Diffusion Based Representation Learning[C]// Proceedings of the 40 th International Conference on Machine Learning. 2023:24963-24982.

[55] FENG Y S,LI J. A Review of Research on Deep Learning Based on the Development of Representation Learning[J]. Microcontrollers & Embedded Systems,2022,22(11):3-6.

[56] WU S,XIAO X,DING Q,et al. Adversarial Sparse Transformer for Time Series Forecasting[C]// Neural Information Processing Systems. San Diego:NIPS,2020:844-856.

[57] LIM B,ARIK S,LOEFF N,et al. Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting[J]. International Journal of Forecasting,2021,37(4):1748-1764.

[58] WU N,GREEN B,XUE B,et al. Deep Transformer Models for Time Series Forecasting:The Influenza Prevalence Case[J]. arXiv:2001.08317,2020.

[59] TANG Y,HA D. The Sensory Neuron as a Transformer:Permutation-Invariant Neural Networks for Reinforcement Learning[C]// Neural Information Processing Systems. San Diego:NIPS,2021:384-397.

[60] KURIN V,IGL M,ROCKTSCHEL T,et al. My Body is a Cage:the Role of Morphology in Graph Based Incompatible Control[C]// International Conference on Learning Representations. Washington DC:ICLR,2021:471-484.

[61] DOSOVITSKIY A,BEYER L,KOLESNIKOV A,et al. An Image is Worth 16x16 Words:Transformers for Image Recognition at Scale[C]// International Conference on Learning Representations. Washington DC:ICLR,2021:571-583.

[62] PARISOTTO E,SONG H,RAE J,et al. Stabilizing Transformers for Reinforcement Learning[C]// International Conference on Machine Learning. New York:ICML,2019:1423-1436.

[63] DAI Z,YANG Z,YANG Y,et al. Transformer-XL:Attentive Language Models beyonda Fixed-Length Context[C]// Annual Meeting of the Association for Computational Linguistics. Stroudsburg:ACL,2019:932-947.

[64] BANINO A,BADIA A,WALKER J,et al. CoBERL:Contrastive BERT for Reinforcement Learning[C]// International Conference on Learning Representation. Washington DC:ICLR,2021:1074-1083.

[65] HU S,ZHU F,CHANG X,et al. UPDeT:Universal Multi-agent Reinforcement Learning via Policy Decoupling with Transformers[C]// International Conference on Learning Representations. Washington DC:ICLR,2021:720-734.

[66] SCHMIDHUBER J. Reinforcement Learning Upside Down:Don't Predict Rewards-Just Map Them to Actions[J]. arXiv:1912.02875,2019.

[67] WANG K,ZHAO H,LUO X,et al. Bootstrapped Transformer for Offline Reinforcement Learning[C]// Thirty-Sixth Conference on Neural Information Processing Systems. NewOrleans. San Diego:NIPS,2022:1244-1258.

[68] FURUTA H,MATSUO Y,GU S. Generalized Decision Transformer for Offline Hindsight Information Matching[C]// International Conference on Learning Representations. Washington DC:ICLR,2021:784-796.

[69] YAMAGATA T,KHALIL A,SANTOS R. Q-learning Decision Transformer:Leveraging Dynamic Programming for Conditional Sequence Modelling in Offline RL[J]. arXiv:2209.03993,2022.

[70] XU M,SHEN Y,ZHUANG S,et al. Prompting Decision Transformer for Few-Shot Policy Generalization[C]// International Conference on Machine Learning. New York:ICML,2022:206-222.

[71] LASKIN M,WANG L. In-context Reinforcement Learning with Algorithm Distillation[J]. arXiv:2210.14215,2022.

[72] MENG L,GOODWIN M,YAZIDI A. Deep Reinforcement Learning with Swin Transformer[J]. arXiv:2206.15269,2022.

[73] MAO H Y,ZHAO R,CHEN H,et al. Transformer in Transformer as Backbone for Deep Reinforcement Learning[J]. arXiv:2212.14538,2022.

[74] HU S,SHEN L,ZHANG Y,et al. Graph Decision Transformer[J]. arXiv:2303.03747,2023.

[75] ESSLINGER K,PLATT R,AMATO C. Deep Transformer Q-

Networks for Partially Observable Reinforcement Learning[J]. arXiv:20222206.01078,2022.
[76] ZHENG Q,HENAFF M,AMOS B. et al. Semi-Supervised Offline Reinforcement Learning with Action-Free Trajectories[J]. arXiv:2210.06518,2022.
[77] ZHU D,WANG Y,SCHMIDHUBER J. et al. Guiding Online Reinforcement Learning with Action-Free Offline Pre-training [J]. arXiv:2301.12876,2023.
[78] LEE K,NACHUM O,YANG M,et al. Multi-Game Decision Transformers[C]// Neural Information Processing Systems. San Diego:NIPS,2022:1844-1852.
[79] REED S,ZOLNA K,PARISOTTO E,et al. A Generalist Agent [J/OL]. Transactions on Machine Learning Research, 2022: 2835-8856. https://openreview.net/forum?id=1ikK0kHjvj.
[80] MELO L. Transformers are Meta-Reinforcement Learners[J]. arXiv:2206.06614,2022.
[81] YUAN H,ZHANG C,WANG H. et al. Plan4MC:Skill Reinforcement Learning and Planning for Open-World Minecraft Tasks[J]. arXiv:20232303.16563,2023.
[82] ZHU X Z,CHEN Y T,TIAN H,et al. Ghost in the Minecraft: Generally Capable Agents for Open-World Environments via Large Language Models with Text-based Knowledge and Memory[J]. arXiv:2305.17144,2023.
[83] BAI Y,JONES A,NDOUSSE K,et al. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback[J]. arXiv:2204.05862,2022.
[84] RAMAMURTHY R,AMMANABROLU P,BRANTLEY K, et al. Is Reinforcement Learning(Not) for Natural Language Processing:Benchmarks,Baselines,and Building Block for Natural Language Policy Optimization[J]. arXiv:2210.01241,2022.
[85] GAO L,SCHULMAN J,HILTON J. Scaling Laws for Reward Model Over optimization[J]. arXiv:2210.10760,2022.
[86] GLAESE A,MCALEESE N,TRKEBACZ M,et al. Improving alignment of dialogue agents via targeted human judgements [J]. arXiv:2209.14375,2022.
[87] LU P,QIU L,CHANG K,et al. Dynamic Prompt L-earning via Policy Gradient for Semi-structured Mathematical Reasoning [J]. arXiv:2209.14610,2022.
[88] TEAM A,BAUER A,et al. Human-Timescale Adaptation in an Open-Ended Task Space[J]. arXiv:2301.07608,2023.
[89] COLEMAN M,RUSSAKOVSKY O,ALLEN C,et al. Discrete Diffusion Reward Guidance Methods for Offline Reinforcement Learning[C]// ICML 2023 Workshop:Sampling and Optimization in Discrete Space. 2023.
[90] PARISOTTO E,SALAKHUTDINOV R. Efficient Transformers in Reinforcement Learning using Actor-Learner Distillation[C]// International Conference on Learning Representations. Washington DC:ICLR,2021:107-123.
[91] SILVA D D,MILLS N,EI A M,et al. ChatGPT and Generative AI Guidelines for Addressing Academic Integrity and Augmenting Pre-Existing Chatbots[C]// 2023 IEEE International Conference on Industrial Technology(ICIT). 2023:1-6.
[92] RAHMAYANTI S R,FATICHAH C,SUCIATI N,et al. Sketch Generation From Real Object Images Using Generative Adversarial Network and Deep Reinforcement Learning[C]// 2021 13th International Conference on Information & Communication Technology and System(ICTS). 2021:134-139.
[93] AYDIN A,SURER E. Using Generative Adversarial Nets on Atari Games for Feature Extraction in Deep Reinforcement Learning[C]// 2020 28th Signal Processing and Communications Applications Conference(SIU). 2020:1-4.
[94] YU C,WANG F. Generative AI:How It Changes Our Lives ? Take Vision & Language as an Example[C]// 2023 International VLSI Symposium on Technology,Systems and Applications(VLSI-TSA/VLSI-DAT). 2023:1-11.
[95] LIU X,YANG H,GAO J,et al. FinRL:deep reinforcement learning framework to automate trading in quantitative finance [C]// Proceedings of the Second ACM International Conference on AI in Finance. 2022:264-278.
[96] DALAL G,DVIJOTHAM K,VECERÍK M. et al. Safe Exploration in Continuous Action Spaces[J]. arXiv:1801.08757,2018.
[97] ZHOU B S,ZHU Y. A Counting Method based on Deep Reinforcement Learning Combined with Generative Adversarial Network[C]// 2022 International Conference on Machine Learning, Cloud Computing and Intelligent Mining(MLCCIM). 2022:431-434.

YAO Tianlei, born in 2000, postgraduate. His main research interest is deep reinforcement learning.

CHEN Xiliang, born in 1985, Ph.D, associate professor. His main research interests include command information system engineering and deep reinforcement learning, etc.

(责任编辑:柯颖)