第 58 卷　第 3 期
2022 年 5 月

南京大学学报(自然科学)
JOURNAL OF NANJING UNIVERSITY
(NATURAL SCIENCE)

Vol. 58, No. 3
May, 2022

DOI:10. 13232/j. cnki. jnju. 2022. 03. 006

深度强化学习结合图注意力模型求解TSP问题

王　扬,陈智斌*,杨笑笑,吴兆蕊
(昆明理工大学理学院,昆明,650000)

摘　要:旅行商问题(Traveling Salesman Problem,TSP)是组合最优化问题(Combinatorial Optimization Problem,COP)中的经典问题,多年以来一直被反复研究. 近年来深度强化学习(Deep Reinforcement Learning,DRL)在无人驾驶、工业自动化、游戏等领域的广泛应用,显示了强大的决策力和学习能力. 结合DRL和图注意力模型,通过最小化路径长度求解TSP问题. 改进REINFORCE算法,训练行为网络参数,可以有效地减小方差,防止局部最优;在编码结构中采用位置编码(Positional Encoding,PE),使多重的初始节点在嵌入的过程中满足平移不变性,可以增强模型的稳定性;进一步结合图神经网络(Graph Neural Network,GNN)和Transformer架构,首次将GNN聚合操作处理应用到Transformer的解码阶段,有效捕捉图上的拓扑结构及点与点之间的潜在关系. 实验结果显示,模型在100-TSP问题上的优化效果超越了目前基于DRL的方法和部分传统算法.

关键词:深度强化学习,旅行商问题,图注意力模型,图神经网络,组合最优化
中图分类号:O22,TP18　　**文献标志码:**A

Deep reinforcement learning combined with graph attention model to solve TSP

Wang Yang, Chen Zhibin*, Yang Xiaoxiao, Wu Zhaorui
(Faculty of Science, Kunming University of Science and Technology, Kunming, 650000, China)

Abstract: Traveling Salesman Problem (TSP) is a classic problem in Combinatorial Optimization Problem (COP), which has been repeatedly studied for many years. In recent years, Deep Reinforcement Learning (DRL) has been widely applied in driverless, industrial automation, game and other fields, showing strong decision-making and learning ability. In this paper, DRL and graph attention model are combined to solve TSP by minimizing the path length. Specifically, the behavioral network parameters are trained by an improved REINFORCE algorithm to effectively reduce the variance and prevent local optima; Positional Encoding (PE) is used to the encoding structure to make the multiple node satisfy translation invariance during the embedding process and enhance the stability of the model. Further, we combine Graph Neural Network (GNN) and Transformer architecture, and apply GNN aggregate operation processing to transformer decoding stage for the first time, which effectively capture the topological structure of the graph and the potential relationships between points. The experimental results show that the optimization effect of the model on the 100-TSP problem surpasses the current DRL-based methods and some traditional algorithms.

Key words: Deep Reinforcement Learning (DRL), Travel Salesman Problem (TSP), graph attention model, Graph Neural Network (GNN), Combinatorial Optimization (CO)

基金项目:国家自然科学基金(11761042)
收稿日期:2022－01－19
* 通讯联系人,E-mail:chenzhibin311@126.com

组合最优化(Combinatorial Optimization,CO)是运筹学和计算机科学领域的一个交叉学科,主要研究具有离散结构的优化问题,即研究如何从一组有限的对象中找到一个最优对象的一类问题[1]. 其中,旅行商问题(Traveling Salesman Problem,TSP)是CO领域经典的NP难问题[2],广泛应用于交通运输、物流配送等工程应用中,可以描述为:途经需要访问城市的地点当且仅当一次,再回到起点城市,使之路径最短[2]. 有动态规划法、割平面法、最近邻点法、蚁群算法等[1]诸多传统方法求解TSP问题,但这些算法的设计需要特定的专业知识[3]. 随着TSP问题实例规模的不断增大及动态随机因素的增加,传统方法的求解将花费大量时间,计算成本也随之增加.

随着深度强化学习(Deep Reinforcement Learning,DRL)在决策问题中的广泛应用,基于DRL的组合最优化问题(Combinatorial Optimization Problem,COP)求解方法不断被提出[4],该方法主要以端到端的形式输出解,大部分模型都基于编码解码的结构,即通过DRL算法训练建立的模型得到一种向目标解中自动添加点和边集的策略,直到构造出完整的解. 此方法减轻了对特定问题和特定领域的知识依赖程度,为求解COP问题提供一种全新的思路. Kwon et al[5]提出一种多目标最优策略优化(Policy Optimization With Multiple Optima,POMO)框架,以纯数据驱动的方式,利用DRL结合Transformer架构[6]求解TSP等COP问题,实验结果证实了模型的有效性. 本文受DRL中智能体可以与环境不断交互学习策略[7]和POMO框架的启发,提出一种基于DRL训练图注意力模型的框架求解TSP问题.

本文的主要贡献:

(1)编码阶段采用位置编码(Positional Encoding,PE)技术,使多重的初始节点坐标在嵌入的过程中满足平移不变性,进而高层的神经网络(Neural Network,NN)能够提取有效的位置信息,从而增强模型的稳定性.

(2)结合图神经网络(Graph Neural Network,GNN)和Transformer架构,首次将GNN的聚合操作应用于Transformer的解码过程,使向量空间具有更强的灵活性,以便捕捉图的拓扑结构及点与点之间的潜在关系,让更多的信息被表征挖掘.

(3)本文提出的模型与目前基于DRL的方法和传统算法相比,使100-TSP问题的路径优化精度得到显著提高,降低了最优间隙,为求解COP问题提供一种全新的思路.

1　相关工作

1.1　传统方法求解TSP问题　求解TSP问题有三种传统方法:精确算法(Exact Algorithm)、近似算法(Approximation Algorithm)、启发式算法(Heuristic Algorithm)[1]. 精确算法往往通过枚举法、整数规划法(Integer programming,IP)、动态规划法(Dynamic programming,DP)等寻找TSP问题的最优解[1],当节点数大于40时不易计算,其中Gurobi[8]和Concorde[9]是基于精确算法下最先进的TSP求解器,100节点内可计算最优解. 在多项式时间内,近似算法可以得到有质量保证的近似解,在最坏情况下给出的解也不高于(最小化问题)最优解的一定倍数,其中Christofides算法[10]、最邻近算法[11]、最远插入法[1]均可近似求解度量TSP问题. 启发式算法指那些快速有效但缺乏理论支撑的算法,可以根据相关问题快速有效地设计算法,通过不断迭代的方式求得问题的次优解或有一定概率得到最优解,常用遗传算法、蚁群算法、LKH-3[12]、2-opt算法等求解TSP问题. 随着TSP问题实例规模的增大、动态因素的增多,传统算法很难快速、智能地求解复杂的TSP问题.

1.2　深度强化学习求解TSP问题　2015年Vinyals et al[13]针对Seq2Seq序列模型以固定的维度输入和输出,对其进行改进,提出指针网络(Pointer Network,PN)架构. 此架构为基于机器学习(Machine Learning,ML)等新方法求解COP问题的工作奠定了很好的理论基础,之后的很多工作都基于此框架展开研究. 监督学习(Supervised Learning,SL)的训练过程需要大量标签,TSP问题的高质量标签不易获得,2016年Bello et al[14]将NN和DRL算法结合,提出神经组合最优化(Neural Combinatorial Optimization,NCO)模型,采用REINFORCE算法训练PN网络,克服了对COP问题数据标签的依赖,扩大了TSP的

求解范围. 2018 年 Deudon et al[15]改进 NCO 模型,设计新的评判网络基准,推理阶段加入 2-opt 操作提高解的质量. 2018 年 Nazari et al[16]将 PN 结构拓展成能够处理动态信息的 COP 模型,具体采用卷积神经网络(Convolutional Neural Network,CNN)替代编码层的长短期记忆网络(Long Short-Term Memory,LSTM),有效求解带容量的车辆路径问题(Capacitated Vehicle Routing Problem, CVRP). 2020 年 Costa et al[17]提出基于 DRL 学习 2-opt 操作的局部搜索启发式算法,使模型更容易拓展到更一般情形的 k-opt 操作,提升了 TSP 问题的求解质量.

图的特殊结构可以承载更多的节点信息,因此可以通过低维的向量信息来表征图的节点及拓扑结构,这种方法可以很好地处理非欧几里得数据,让网络模型容易学习到有效的特征信息. 鉴于此,2017 年 Dai et al[18]将 DRL 和 GNN 结合,提出 S2V-DQN 模型,求解大规模 TSP 问题. 2019 年 Kool et al[19]基于 Transformer 结构[6],采用多重注意力机制(Multi Head Attention,MHA)的自注意力机制计算方法提取深层节点的特征信息,有效防止了节点信息的丢失,该模型与 PN 网络相比,泛化能力更好,训练速度更快. 2019 年 Chen and Tian[20]提出 NeuRewriter 架构,通过 DRL 训练模型,以迭代的方式不断改进 CVRP 问题的解直到收敛. 2020 年 Li et al[21]提出一种基于 DRL 解决多目标 COP 问题的框架(DRL-MOA),在 PN 网络中融入分解策略和邻居多参数传递策略,有效求解 TSP 问题. 2021 年 Wu et al[22]提出一种直接策略的方法,通过 self-注意力机制参数化策略模型,在模型训练阶段即可得到 TSP 和 CVRP 问题的解. 2021 年 Xin et al[23]提出多重解码注意力模型(MDAM),求解多目标路径问题,并在编码中加入 Embedding Glimpse 层,提升了模型整体优化性能.

2 模型结构

DRL 中的智能体会根据当前的环境状态作出相应的行为决策,并根据行为的反馈不断调整自身学习到的策略,从而达到特定的学习目标,使得 DRL 适用于序贯决策问题. 一般 TSP 问题中任意两城市之间的距离与城市的排列顺序无关,路径的选取中强调环境因素会影响决策,与 DRL 的行为选择有天然的相似性,可以定义环游长度为[14]:

$$L(\pi|s)=\left\| x_{\pi(n)}-x_{\pi(1)} \right\|_2+\sum_{i=1}^{N-1}\left\| x_{\pi(i)}-x_{\pi(i+1)} \right\|_2 \quad (1)$$

首先将 TSP 问题建模成马尔可夫决策过程(Markov Decision Process, MDP),再利用 DRL 算法训练参数 θ,即学习到一个最优策略 $\pi_t=p_\theta(a_t|s)$,以较大概率输出路径长度最短的环游 L. 该策略可以建模为:

$$p_\theta(a_t|s)=\prod_{t=1}^{N}p_\theta(a_t|s,a_{1:t-1}) \quad (2)$$

通过上述方法的建模实现了 DRL 与 TSP 问题的结合,其中智能体状态 S 为每个已访问城市的坐标列表,状态空间为下一步将要访问的城市概率,动作转移函数为第 t 步将要选择的城市 π_t,奖励为已访问城市路径总距离的负数(最短路径),行为策略是状态 S 到动作 A 的映射,最终输出选择城市的概率. 本文提出的图注意力模型由编码和解码组成(详见 2.2 和 2.3),如图 1 所示.

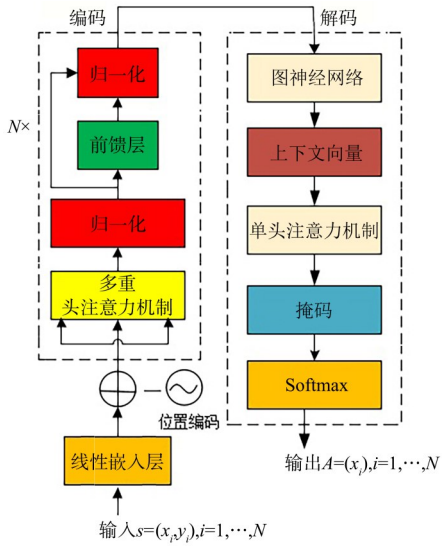

图 1 本文提出的图注意力模型示意图

Fig. 1 Schematic diagram of graph attention model proposed in this paper

2.1 多重初始点 TSP 问题的策略(式(2))以 a_1 为起点与 $(a_2,a_3,\cdots,a_N)$ 中任意起点得到的解是等价的,如果轨迹 $\tau=(a_1,a_2,a_3,a_4)$ 是一个最

优解,那么轨迹$\tau'=(a_2,a_3,a_4,a_1)$也是一个最优解.图2展示了四个节点的多重选择.先前工作[14−20]只考虑单一最优路径,而本文类似Kwon et al[5],采用N种不同的点序列表征最优解,模型在编码、解码、推理阶段均放置N种不同的初始起点$\{a_1^1,a_1^2,\cdots,a_1^N\}$,其中每个节点都能被策略网络选取.实验结果显示多重轨迹的构造可以防止陷入局部最优,能更有效地寻求最短路径,行为网络通过蒙特卡洛方法(Monte Carlo method)采样N种不同初始起点的轨迹$\{\tau^1,\tau^2,\cdots,\tau^N\}$,其中,每个轨迹被定义为($M$为节点个数):

$$\tau^i=(a_1^i,a_2^i,\cdots,a_M^i),i=1,2,\cdots,N \tag{3}$$

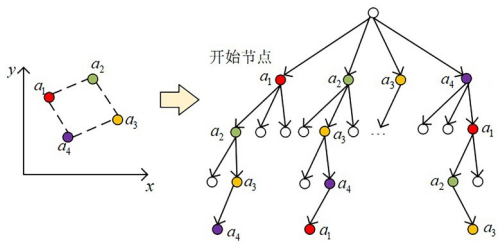

图2　多重初始点示意图

Fig. 2　Schematic diagram of multiple initial points

2.2　编码结构　本文针对TSP问题的编码类似Transformer架构[6]的编码部分,考虑多重起点的嵌入(输入排列是变化的).环境中生成的节点坐标进行线性嵌入操作时不能有效捕获每个节点的位置信息,因此本文模型采用PE操作,使得节点坐标在嵌入的过程中满足平移不变性,以便高层的NN能够提取更多有效的位置信息;再将处理后的向量嵌入到MHA层,提取深层网络的节点信息.其中,PE,MHA被定义为:

$$PE_{t,i}=\begin{cases}\sin\left(t10000^{\frac{d}{2i}}\right),i\text{是奇数}\\ \cos\left(t10000^{\frac{d}{2i}}\right),i\text{是偶数}\end{cases},PE_t\in R^d \tag{4}$$

$$A^i=Attention(Q^i;K^i;V^i)=\mathrm{softmax}\left(\frac{Q^iK^{i\mathrm{T}}}{\sqrt{d}}\right)V^i,i=1,2,\cdots,H \tag{5}$$

$$MHA(Q,K,V)=Concat(A^1,A^2,\cdots,A^H)W_O \tag{6}$$

其中,t表示编码节点的位置,$d=512$为嵌入维度,$H=8$为注意力机制的头部数,Q,K,V是查询、键、值向量.注意力机制的输出A^i被连接并映射到$W_O\in R^{d\times d}$空间,得到MHA层的输出;再传入批次正则化(Batch Normalization,BN)处理层,经过非线性函数ReLU激活后,传入前馈网络层(Feed Forward,FF);再次由BN层处理输出编码向量.其中,BN层和FF层被定义为[6]:

$$\hat{f}_i=BN\left(X_i+MHA_i(Q,K,V)\right),i=1,\cdots,N \tag{7}$$

$$f_i=BN\left(\hat{f}_i+FF\left(\hat{f}_i\right)\right),i=1,\cdots,N \tag{8}$$

节点坐标经过上述MHA模型、BN层、FF层编码成序列向量,传入解码层继续做图嵌入(Graph Embedding,GE)、点嵌入(Node Embedding,NE)、上下文向量(Context)、掩码(Mask)等处理(详见2.3),输出选择下一个节点的概率,直到所有节点都被选择,构成一个环游策略.

2.3　解码结构　TSP问题中,每个节点信息具有一定的相似性且与邻居节点相关,将其抽象为节点和边集的关系可建模成图模型.因此,在GE层中所有被编码的城市坐标X可由GNN中的聚合操作解码,使向量空间具有更强的灵活性和丰富多样的计算形式以便捕捉图的拓扑结构及点与点之间的潜在关系,让更多的信息被表征挖掘,那么解码后的嵌入将会有更好的表现.本文首次将GNN的聚合操作应用到Transformer架构的解码阶段中,其中,GE结构的表达式可刻画为:

$$x_i^l=\gamma x_i^{l-1}\Theta+(1-\gamma)\varphi_\theta\left(\frac{\left\{x_j^{l-1}\right\}_{j\in N(i)\cup\{i\}}}{|N(i)|}\right) \tag{9}$$

其中,$x_i^l\in R^{d_l}$是l层$\left(l\in\{1,\cdots,L\}\right)$的变量,$\gamma$是一个调整权重矩阵特征值的参数,$\Theta\in R^{d_{l-1}\times d_l}$是权重矩阵,$N(i)$是点$i$的邻接集合,$\varphi_\theta:R^{d_{l-1}}\to R^{d_l}$是通过GNN表达的聚合函数[24].

考虑具有对称性质的TSP问题,即城市节点组成的图由一个完全图刻画,因此GE结构的表达式可写为:

$$X^l=\gamma X^{l-1}\Theta+(1-\gamma)\Phi_\theta\left(\frac{X^{l-1}}{|N(i)|}\right) \tag{10}$$

其中,$X^l\in R^{N\times d_l}$,$\Phi_\theta:R^{N\times d_{l-1}}\to R^{N\times d_l}$是通过GNN表达的聚合函数[24].

下一个解码阶段,类似Kool et al[19],引入上下文节点c,经过多重起点随机选择初始节点后,加入遮掩技术(访问过的节点不能再次被访问),

有效计算编码后节点的注意力分配,以较大概率输出下一个访问的城市节点. 图 3 展示了最优路径 $\pi=(3,1,2,4)$ 的构造过程. 具体地,通过水平拼接操作将编码层的图嵌入、初始节点 π_1、先前节点 π_{t-1} 聚合成一个三维向量,记作 h_c^i,描述为:

$$h_c^i=\begin{cases}\left[\bar{h},h_{\pi_{t-1}},h_{\pi_1}\right] & t>1\\ \text{none} & t=1\end{cases} \tag{11}$$

其中,$t=1$ 时,不采用解码控制第一个节点的选择,使用 N 种不同的上下文节点嵌入,得到 $h_c^1,h_c^2,\cdots,h_c^N$,h_c^i 表示上下文节点 c 的嵌入信息. 因此,模型中查询、键、值向量可以被表示为:

$$\begin{aligned}q_c&=W^Q h_c\\ k_i&=W^K h_i\\ v_i&=W^V h_i\end{aligned} \tag{12}$$

为了计算 $p_\theta\left(a_t\middle|s,a_{1:t-1}\right)$ 输出概率,最后一步采用单头的注意力机制处理解码向量,其中掩码技术和输出向量 p_i 可表示为($C=10$):

$$u_{cj}=\begin{cases}C\cdot\tanh\left(q_c^{\mathrm{T}}k_j\right) & \text{if } j\neq\pi_{t'},\forall t'<t\\ -\infty & \text{其他}\end{cases} \tag{13}$$

$$p_i=p_\theta\left(a_t\middle|s,a_{1:t-1}\right)=\frac{e^{u_{ci}}}{\sum_j e^{u_{cj}}},i\text{邻接}j \tag{14}$$

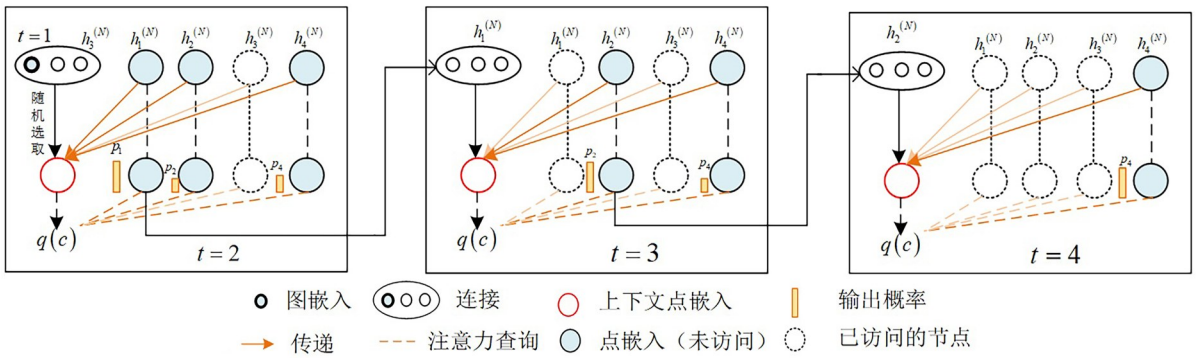

图 3 四个节点的 TSP 问题解码示例图

Fig.3 Schematic diagram of TSP problem decoding for four nodes

2.4 模型训练 鉴于 SL 对标签的大量需求,实际工程应用中 TSP 问题的高质量标签又不易获得,而 DRL 的方法不需要大量的标签数据,因此本文采用 DRL 的方法训练网络模型. TSP 问题的优化目标是路径长度 $L(\pi)$ 最小,总奖励即为路径总长度的负数 $-L(\pi)$. 由于 REINFORCE 算法[25]是以总奖励作为参数更新的,因此该算法天然适用于训练求解 TSP 问题,大多数 COP 问题通常也采用该算法对策略参数 θ 进行优化[4]. 此算法求解 TSP 问题的一个主要缺陷是不同路径之间的方差很大,导致训练不稳定,这是在高维离散空间中常见的问题,为了减小策略梯度(Policy Gradient,PG)的方差,本文引入一个和 $R(\tau^i)$ 相关的基准函数,记为 $\overline{r(\tau^i)}$,表达式如下:

$$\overline{r(\tau^i)}=\frac{1}{N}\sum_{i=1}^{N}r(\tau^i) \tag{15}$$

受到交叉熵损失函数(Cross-Entropy Loss)[26]的启发,在基准线 $\overline{r(\tau^i)}$ 上加入超参数 $\beta=0.1$,调节奖励值的变化频率,防止过早收敛,以便更好地衡量不同城市间的差异分布程度. 后文的实验结果证明该方法的收敛速度优于原始的基准线. $\overline{R(\tau^i)}$ 可表示为:

$$\overline{R(\tau^i)}=\beta\times r(\tau^i)+(1-\beta)\overline{r(\tau^i)} \tag{16}$$

PG 法通过寻找一个参数 θ 使得目标函数 $J(\theta)$ 最大,参数 θ 优化的方向是使得总回报 $R(\tau^i)$ 越大,即轨迹 $\{\tau^1,\tau^2,\cdots,\tau^N\}$ 的概率 $P_\theta(\tau^i)$ 越大. 因此,$J(\theta)$ 的梯度可以被近似为:

$$\nabla_\theta J(\theta)\approx\frac{1}{N}\sum_{i=1}^{N}\left(R(\tau^i)-\overline{R(\tau^i)}\right)\nabla_\theta\lg p_\theta\left(\tau^i\middle|s\right) \tag{17}$$

模型通过一个随机策略学习行为网络的参数 θ,上述公式对 θ 的梯度进行计算并更新,不断迭代训练从而得到最优的策略 $\pi_t=p_\theta(a_t|s)$. 算法 1 描述了模型训练流程,通过这种共享奖励值基准线的构造,代替模型中的评判网络,简化模型的结构,实现 TSP 问题序列到解序列的精准映射.

算法1　改进的REINFORCE算法

输入:训练集S,每个起始点的数字N,训练次数T,批次大小B,可微分的策略函数$\pi_\theta(a_i|s)$

输出:策略π_θ

随机初始化网络参数θ

Repeat

　根据策略$\pi_\theta(a|s)$生成轨迹τ_i

　for 训练次数$=1,\cdots,T$ do

　　$S_i \leftarrow Sampleinput(S),\forall i=\{1,\cdots,B\}$

　　$\{\alpha_i^1,\alpha_i^2,\cdots,\alpha_i^N\} \leftarrow Selectstartnodes(S_i),\ \forall i=\{1,\cdots,B\}$

　　$\tau_i^j \leftarrow Samplerollout(\alpha_i^1,S_i,\pi_\theta),\ \forall i=\{1,\cdots,B\},\forall j=\{1,\cdots,N\}$

　end for

$\bar{r} \leftarrow \frac{1}{N}\sum_{j=1}^{N} r(\tau_i^j),\forall i=\{1,\cdots,B\}$

$\overline{R} \leftarrow \beta \times r(\tau_i^j)+(1-\beta)\overline{r(\tau_i^j)}$

$\theta \leftarrow \theta+\alpha\nabla_\theta J_\theta$

until θ收敛(奖励值稳定)

2.5　模型推理　近年来基于DRL的方法已在COP问题中取得较好的成果,同时也看到,这些方法大多还需结合一些传统的运筹优化方法,如贪婪(Greedy),每次选取输出概率最高的节点(最优的解);波束搜索(Beam Search),宽度受限广度优先搜索的方式;采样(Sampling),采样一定数量的解,取最优的解[4]. TSP问题经过模型训练的整体框架如图4所示,模型训练后得到的序列向量,经上述方法推理改善后,其最优间隙能显著降低,进一步提高解的质量. Kwon et al[5]提出一种八距离扩大的推理方法,由于模拟实验的坐标有对称性,本文采用四距离扩大的方法,即将所有节点坐标(x,y)转换为$(x,1-y)(1-x,y)(1-x,1-y)$三种形式,并在3.2的实验结果对比中放置单路径搜索、全路径搜索、八距离扩大三种推理方式.3.3给出了POMO模型推理方法的消融实验(Ablation Experiment),结果显示推理方法有效.

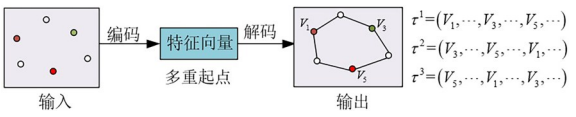

图4　TSP问题编码解码结构示意图

Fig. 4　Schematic diagram of encode-decode structure of TSP problem

3　数值实验

3.1　实验环境和超参数的设置　基于Pytorch-1.9.0深度学习平台,在Windows10操作系统上使用Nvidia RTX 1650 GPU运行本文模型和POMO模型. 分别在20,50,100个节点的TSP问题上进行训练和测试,每个训练批次和测试批次都分别放置在10万和1万的单位平方形中. 最优间隙以目前Gurobi,Concorde等专业求解器(已得到100个节点内的最优解)为基准. LKH3、2-opt、最远插入、最邻近等传统算法在Intel Core i5-9300H CPU上运行. 其他结果来自原始文献. 所有节点实验中,每个城市的坐标由(x_i,y_i)表示,所有城市均放置在$[0,1]\times[0,1]$单位平方形中,训练和测试阶段使用相同的数据分布. PG算法的每个批次放置64个节点,每个城市被嵌入128维的欧几里得空间,MHA中的头部$H=8$,FF输入层和输出层的维度都是512维,使用$L=3$的GNN聚合GE层的坐标嵌入,Adam优化器的学习率为$\eta=10^{-4}$,权重衰减率为$w=10^{-7}$.

3.2　TSP问题实验结果对比　分配每个节点(N)作为一个初始节点,以N种轨迹τ^i高效寻找二维欧几里得空间中TSP20,TSP50,TSP100的最短路径问题. 首先通过目前最先进的专业求解工具Concorde[9]和Gurobi[8]计算获得TSP问题的最优解作为其他模型计算最优间隙的基准;其次,对比近年来基于DRL方法求解TSP问题的模型;最后放置本文模型的优化效果. 表1针对TSP问题对比了本文的模型和其他模型的优化效果,但没有比较Vinyals, Bello, Nazari, Dai, Deudon[13-16,18]的相关模型,因为已经被Kool et al[19]的注意力机制模型超越. 表中的黑体字表示本文模型优于目前基于DRL的方法,$n\times$ augment表示节点坐标变换为原来的n种形式. 由表可见,本文模型在推理阶段的求解时间比部分传统算法更快,与目前基于DRL的方法相当. 图5对比了Christofides算法[10]、2-opt等传统方法和POMO框架、图指针网络(Graph Pointer Net-

表 1 不同模型在 TSP 问题上的优化结果比较

Table 1 Optimization results for TSP problem by different models

模型	20-TSP			50-TSP			100-TSP		
	花费	间隙	时间	花费	间隙	时间	花费	间隙	时间
Concorde[9]	3.83	0.00%	5 min	5.69	0.00%	13 min	7.76	0.00%	1 h
Gurobi[8]	3.83	0.00%	7 s	5.69	0.00%	2 min	7.76	0.00%	17 min
OR-Tools	3.86	0.94%	1 min	5.85	2.87%	5 min	8.06	3.86%	23 min
LKH3[12]	3.83	0.00%	42 s	5.69	0.00%	6 min	7.76	0.00%	25 min
2-opt [1]	3.95	3.13%	1 s	6.11	7.38%	7 s	8.50	9.53%	33 s
Farthest Insertion[11]	3.89	1.56%	1 s	5.97	4.92%	2 s	8.34	7.47%	10 s
Nearest Neighbor[1]	4.48	16.9%	1 s	6.94	21.9%	3 s	9.68	24.7%	7 s
Kool et al(Greedy)[19]	3.85	0.34%	≪1 s	5.80	1.76%	2 s	8.12	4.53%	6 s
Kool et al(Sampling)[19]	3.84	0.08%	5 min	5.73	0.52%	24 min	7.94	2.26%	1 h
Costa et al[17]	3.83	0.00%	15 min	5.71	0.12%	29 min	7.83	0.87%	41 min
Wu et al[22]	3.83	0.00%	1 h	5.70	0.20%	1.5 h	7.87	1.42%	2 h
Kwon et al(single trajec)[5]	3.83	0.12%	≪1 s	5.74	1.03%	3 s	7.84	1.12%	8 s
Kwon et al(no augment)[5]	3.83	0.04%	≪1 s	5.71	0.35%	10 s	7.79	0.50%	54 s
Kwon et al(8×augment)[5]	3.83	0.00%	16 s	5.69	0.05%	1 min	7.77	0.14%	7 min
Ours (single trajec)	3.83	0.13%	≪1 s	5.73	0.70%	5 s	7.84	1.08%	8 s
Ours (no augment)	3.83	0.05%	≪1 s	5.70	0.28%	10 s	7.79	0.47%	55 s
Ours (8×augment)	3.83	0.00%	17 s	5.69	0.03%	1 min	7.77	0.12%	7 min
Ours (4×augment)	**3.83**	**0.00%**	9 s	**5.69**	**0.01%**	42 s	**7.76**	**0.09%**	3 min

work, GPN)[27]、PN 网络、NCO 模型、S2V-DQN 模型等经典模型与本文模型的最优间隙. 四距离扩大的推理方法使 20-TSP 的最优间隙达到 0.00%(越低效果越好),50-TSP 的最优间隙达到 0.01%,100-TSP 的最优间隙达到 0.09%,均优于目前基于 DRL 的方法.

3.3 消融实验 表 2 展示了本文模型在 TSP 问题上的消融实验结果,其中 $n\times$ augment 表示节点坐标变换为原来的 n 种形式,证明了四距离扩大推理方法的有效性. 其中,推理时间缩短了约 50%,TSP50,TSP100 的最优间隙也略有提高,说明此方法是合理的.

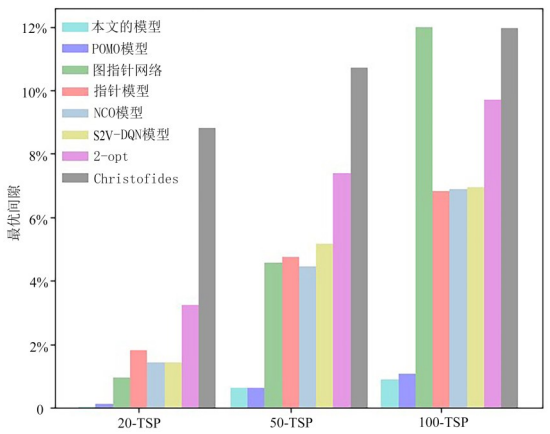

图 5 经典模型最优间隙的对比图

Fig. 5 Comparison diagram of optimal gap in classical models

表 2 POMO 模型在 TSP 问题上的消融实验

Table 2 Results of ablation experiments for TSP problems by POMO model

模型	20-TSP			50-TSP			100-TSP		
	花费	间隙	时间	花费	间隙	时间	花费	间隙	时间
POMO (8×augment)[5]	3.83	0.00%	16 s	5.69	0.05%	1 min	7.77	0.14%	7 min
POMO (4×augment)[5]	3.83	0.00%	7 s	5.69	0.03%	40 s	7.77	0.13%	3 min

3.4　收敛性对比　编码结构中引入PE操作后,对比POMO模型,本文模型在200个批次内可以稳定收敛到较优解,如图6所示.多重起点的初始解和PE层的处理提升了模型的整体优化性能,在训练过程中可得到高质量的解.

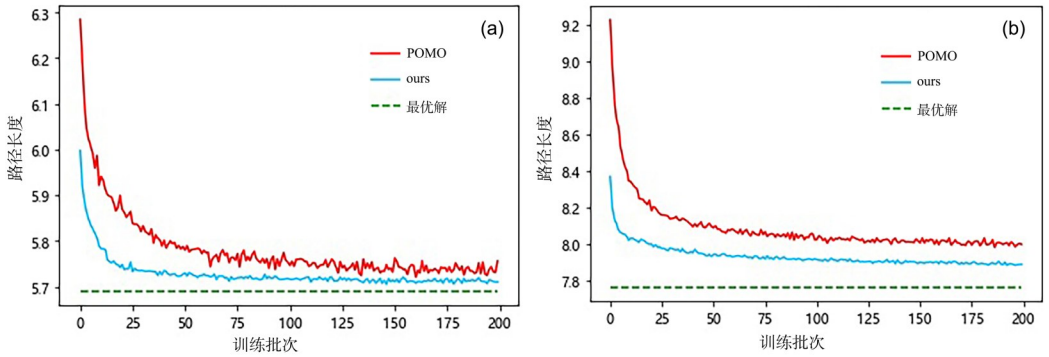

图6　本文模型与POMO模型在TSP50/100 (a,b)上的训练损失对比图

Fig. 6　Training loss of our model and POMO model on TSP50/100 (a, b)

3.5　泛化能力对比　以DRL算法结合图注意力模型的求解方法摆脱了传统算法针对相同结构问题专门设计算法的弊端,模型一旦训练完成(得到求解问题的最优策略),即可对任意类似大小问题进行泛化求解.TSP问题的泛化能力的比较结果如表3所示,可见无论在小范围还是大范围的规模上,本文算法的泛化能力都有较好的表现.

表3　本文模型对TSP问题的泛化能力比较

Table 3　Generalization ability of our model for TSP problems

模型	20-TSP			50-TSP			100-TSP		
	花费	间隙	时间	花费	间隙	时间	花费	间隙	时间
Ours (TSP20)	—	—	—	5.73	0.68%	1 min	8.05	3.73%	5 min
Ours (TSP50)	3.83	0.13%	3 s	—	—	—	7.84	1.03%	4 min
Ours (TSP100)	3.83	0.05%	1 s	5.71	0.33%	30 s	—	—	—

3.6　时间花费对比　本文模型分训练和推理两个阶段,每个阶段的耗时如表4所示.由表可见,TSP20在训练阶段耗时3 h,但TSP20在推理阶段仅耗时9 s就可得到最优解.所以综合来看,与传统算法相比,本文算法具有较大的优势.

表4　本文模型在训练和推理阶段的时间花费

Table 4　Time cost for training and reasoning by our model

阶段	TSP20	TSP50	TSP100
训练模型	3 h	24 h	136 h
推理(single trajec)	≪1 s	5 s	8 s
推理(no augment)	≪1s	10 s	55 s
推理(8×augment)	17 s	1 min	7 min
推理(4×augment)	9 s	42 s	3 min

4　结论和展望

本文提出一种基于DRL训练图注意力模型的框架.鉴于模型中多重起点的放置,编码初始阶段采用PE编码,使多重的初始节点坐标在嵌入的过程中满足平移不变性,进而高层的NN能够提取有效的位置信息,从而增强模型的稳定性,有效防止局部最优.首次将GNN的聚合操作应用于Transformer的解码中,使向量空间具有更强的灵活性和丰富多样的计算形式,以便捕捉图的拓扑结构及节点与节点之间的潜在关系,让更多

的潜在信息被表征挖掘. 模型训练以$\overline{R(\tau^i)}$作为REINFORCE算法的基准函数,可以有效减小方差,优化了模型的整体性能. 该模型求解TSP100问题的效果超越了目前基于DRL的方法和部分传统算法,推理速度超越目前最先进的专业求解器Concorde,且模型具有很好的泛化能力.

未来的工作将考虑求解更大规模的TSP问题,并采用DRL的方法求解更多类型的COP问题,提高模型的泛化能力.

参考文献

[1] Cook W J, Cunningham W H, Pulleyblank W R, et al. Combinatorial optimization. New York, NY, USA: Wiley-Interscience, 2010: 11－22.

[2] Papadimitriou C H. The Euclidean travelling salesman problem is NP - complete. Theoretical Computer Science, 1977, 4(3): 237－244.

[3] 林敏,刘必雄,林晓宇. 带Metropolis准则的混合离散布谷鸟算法求解旅行商问题. 南京大学学报(自然科学), 2017, 53(5): 972－983. (Lin M, Liu B X, Lin X Y. Hybrid discrete cuckoo search algorithm with metropolis criterion for traveling salesman problem. Journal of Nanjing University (Natural Science), 2017, 53(5): 972－983.)

[4] Bengio Y, Lodi A, Prouvost A. Machine learning for combinatorial optimization: A methodological tour d′horizon. European Journal of Operational Research, 2021, 290(2): 405－421.

[5] Kwon Y D, Choo J, Kim B, et al. POMO: Policy optimization with multiple optima for reinforcement learning. 2020, arXiv:2010.16011.

[6] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need//Proceedings of the 31st International Conference on Neural Information Processing Systems. Red Hook, NY, USA: Curran Associates Inc., 2017, 30: 6000－6010.

[7] Li Y X. Deep reinforcement learning: An overview. 2017, arXiv:1701.07274.

[8] Optimization I G. Gurobi optimizer reference manual. https://www.gurobi.com, 2015.

[9] Applegate D L, Bixby D E, Chvatal V, et al. The traveling salesman problem: A computational study. Interfaces, 2008, 38(4): 344－345.

[10] Christofides N. Worst - case analysis of a new heuristic for the travelling salesman problem. Pittsburgh, PA, USA: Carnegie-Mellon University, 1976.

[11] Johnson D S. Local optimization and the traveling salesman problem//The 17th International Colloquium on Automata, Languages and Programming. Springer Berlin Heidelberg, 1990: 446－461.

[12] Helsgaun K. An effective implementation of the Lin-Kernighan traveling salesman heuristic. European Journal of Operational Research, 2000, 126(1): 106－130.

[13] Vinyals M, Fortunato M, Jaitly N. Pointer networks//Proceedings of the 29th International Conference on Neural Information Processing System. Cambridge, MA, USA: MIT Press, 2015 (28): 2692－2700.

[14] Bello I, Pham H, Le Q V, et al. Neural combinatorial optimization with reinforcement learning. 2017, arXiv:1611.09940.

[15] Deudon M, Cournut P, Lacoste A, et al. Learning heuristics for the TSP by policy gradient// Proceedings of the 15th International Conference on the Integration of Constraint Programming, Artificial Intelligence and Operations Research. Springer Berlin Heidelberg, 2018: 170－181.

[16] Nazari M, Oroojlooy A, Takáč M, et al. Reinforcement learning for solving the vehicle routing problem//Proceedings of the 32nd International Conference on Neural Information Processing Systems. Red Hook, NY, USA: Curran Associates Inc., 2018(31): 9861－9871.

[17] Costa P R O D, Rhuggenaath J, Zhang Y Q, et al. Learning 2-opt heuristics for the traveling salesman problem via deep reinforcement learning// Proceedings of the 12th Asian Conference on Machine Learning. Bangkok, Thailand: JMLR, 2020: 465－480.

[18] Dai H J, Khalil E B, Zhang Y Y, et al. Learning combinatorial optimization algorithms over graphs// Proceedings of the 31st International Conference on Neural Information Processing Systems. Red Hook, NY, USA: Curran Associates Inc., 2017, 30: 6351－6361.

[19] Kool W, Van Hoof H, Welling M. Attention , learn to solve routing problems. 2019, arXiv:1803.08475.

[20] Chen X Y, Tian Y D. Learning to perform local rewriting for combinatorial optimization// Proceedings of the 33rd Neural Information Processing Systems. Vancouver, Canada: NIPS, 2019:6278－6289.

[21] Li K W, Zhang T, Wang R. Deep reinforcement learning for multiobjective optimization. IEEE Transactions on Cybernetics, 2020, 51(6): 3103－3114.

[22] Wu Y X, Song W, Cao Z G, et al. Learning improvement heuristics for solving routing problems. IEEE Transactions on Neural Networks and Learning Systems, 2021:1－13.

[23] Xin L, Song W, Cao Z G, et al. Multi-decoder attention model with embedding glimpse for solving vehicle routing problems//Proceedings of the 35th AAAI Conference on Artificial Intelligence. Palo Alto, CA, USA: AAAI Press, 2021:12042－12049.

[24] Scarselli F, Gori M, Tsoi A C, et al. The graph neural network model. IEEE Transactions on Neural Networks, 2008, 20(1):61－80.

[25] Williams R J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine Learning, 1992, 8(3):229－256.

[26] Ho Y, Wookey S. The real-world-weight cross-entropy loss function: Modeling the costs of mislabeling. IEEE Access, 2019(8):4806－4813.

[27] Ma Q, Ge S W, He D Y, et al. Combinatorial optimization by graph pointer networks and hierarchical reinforcement learning. 2019, arXiv: 1911.04936.

（责任编辑　杨可盛）