

基于变压器的场景表示学习增强学习在自动驾驶决策中的应用

刘昊晨, 黄志宇, 莫晓宇, 吕晨*, IEEE 高级会员

摘要: 由于互动交通参与者的随机性和道路结构的复杂性, 城市自动驾驶的决策具有挑战性。基于强化学习(RL)的决策方案虽然有望处理城市驾驶场景, 但存在样本效率低、适应性差的问题。在本文中, 我们提出了场景再现变压器, 通过改进场景表示编码和顺序预测潜在蒸馏来增强 RL 的决策能力。具体而言, 构建了多级 Transformer (MST)编码器, 不仅可以模拟自我车辆与其邻居之间的交互意识, 还可以模拟智能体与其候选路线之间的意图意识。采用具有自监督学习目标的顺序潜在变压器(SLT)将未来预测信息提取到潜在场景表示中, 以减少探索空间并加快训练速度。最后, 基于软演员评论(SAC)的决策模块将场景再现变压器中精炼的潜在场景表示作为输入并生成决策。该框架在五个具有挑战性的密集交通模拟城市场景中进行了验证, 其性能通过在成功率、安全性和效率方面的数据效率和性能的实质性改进来定量地体现出来。定性结果表明, 我们的框架能够提取相邻代理的意图, 从而实现更好的决策和更多样化的驾驶行为。代码和结果可在 <https://georgeliu233.github.io/Scene-Rep-Transformer/>上获得

索引术语——自动驾驶、决策、强化学习、场景表示。

我的介绍。

做出安全、平稳和智能的决策是自动驾驶汽车(av)面临的主要挑战[1], [2], 尤其是在复杂的城市驾驶场景中[3]。各种交通参与者和道路结构之间复杂的交互动态使得这些目标难以实现。基于强化学习(RL)的方法在解决这些挑战方面得到了普及[4]-[6]。强化学习智能体可以从与驾驶环境的交互中学习(主要是在模拟中), 并且能够处理各种交通情况和任务, 将研究人员和工程师从繁琐的基于规则的决策方案中解放出来[7], [8]。然而, 基于强化学习的方法仍然会遇到一些常见的问题

刘辉, 黄振中, 莫新, 吕振, 就职于新加坡南洋理工大学机械与航空航天工程学院, 新加坡 639798。(e-mail: haochen002@e.ntu.edu.sg, zhiyu001@e.ntu.edu.sg, xiaoyu006@e.ntu.edu.sg, lyuchen@ntu.edu.sg)
本工作得到了 suga - nap 基金(No.;新加坡南洋理工大学 M4082268.050)资助。
通讯作者: 吕 c

null 可能阻碍其进一步发展。一个主要的缺点是数据效率, 这意味着训练一个功能性的驾驶策略需要通过与环境的交互获得大量的样本。当面对城市驾驶场景时, 比如无保护的左转弯或交通密集的道路分支, 这个问题就会加剧。另一个问题是如何构建和编码城市驾驶场景的状态表示, 这是将强化学习推广到更复杂场景的瓶颈。与常见的强化学习问题(环境或智能体的状态相对简单)不同, 城市驾驶的状态表示应该涵盖异构信息, 如交通流的动态特征、潜在智能体相互作用和道路结构。此外, 在场景表示中提取预测信息可以帮助智能体理解某些决策的后果, 促进对驾驶策略的学习, 从而进行决策。

为此, 本文提出了一种基于变压器的结构, 称为 scene - rep Transformer, 以更好地捕捉异构元素之间的关系, 并将预测信息注入到场景表示中, 从而增强基于强化学习的决策方案的能力, 加快训练速度。首先, 我们构建了一个多级变压器(multi-stage Transformer, MST)来编码原始的异构场景元素信息, 如 agent 的历史运动状态和候选路径点, 并融合这些信息来表示它们之间的相互作用。MST 在给定的时间步长产生关于驾驶场景的单个潜在特征向量。其次, 采用自监督学习方法, 引入序列潜变(SLT), 从未来时间步长的潜动作对序列中提取共同特征到当前步长的潜动作对中。SLT 的目的是获得对环境序列动态的核心理解, 并减少探索空间, 因为当前潜在表示包含有关未来步骤的一致信息。SLT 仅用于训练过程中, 以方便潜在向量的学习并加快训练速度。最终, 基于 sac 的决策模块将精炼后的潜在特征向量作为输入, 输出质量更好的决策。本文的主要贡献可以概括如下:

- 1)设计了一种新的基于变压器的场景表示学习框架, 以有效增强基于强化学习的决策系统在城市驾驶场景中的训练和测试能力。

- 2)提出多级 Transformer 对异构场景元素信息进行编码，并对智能体和地图路径点之间的交互进行建模，从而实现强化学习模块的潜在场景表示。提出了一种具有增强自监督学习目标的序列潜在变形器，将预测序列信息提取到潜在表征中，以方便强化学习训练。
- 3)通过性能提升对所提出的框架进行了分析证明。在多个具有挑战性的交互式驾驶场景下进行了进一步验证，结果表明，在数据效率、任务成功率和可解释性方面，强化学习驾驶智能体的性能得到了显著提高。我们还研究了框架中不同组件的效果。

2. 相关工作

A. 针对自动驾驶的强化学习

强化学习(RL)最近被广泛应用于自动驾驶的决策任务中，并取得了良好的效果[4]，胜过了基于规则的方法，这些方法通常是繁重的，不能很好地完成复杂的任务[2]。另一种有竞争力的方法是模仿学习[9]，[10]，它依赖于专家数据的监督学习，但在部署中存在分布移位问题。另一方面，RL可以避免这个问题，因为它依赖于与环境的交互，但可能需要大量的交互来训练一个可接受的策略。因此，最近的研究提出了一些提高 RL 数据效率的方法，如用专家动作指导 RL 决策[5]，[11]，[12]，因为这样可以显著减少动作探索空间。在我们的工作中，我们建议利用表征学习技术来获得更好的、可预测的、可推广的环境表征，而不是使用昂贵的专家指导或演示，以提高 RL 的训练数据效率和测试性能。另一个问题是驱动决策的动作空间的恰当选择。一种主流是端到端决策[8]、[13]，其中直接用转向[14]、油门或端点[15]映射连续的动作空间；同时，它被困在不连续的一般性上。[7]。因此，在我们的工作中，我们通过在各种自动驾驶场景中引入战略决策[16]来缓解动作设置，结果很有希望。

RL 的另一个解决方案是顺序学习。状态-动作-奖励对的专家序列使用 Transformer 解码器以自回归的方式传播。TT[17]使用波束搜索推断每一步的离散序列映射，而 DT[18]学习动作映射。与 IL 相比，这些方法迫切需要专家数据，并且在冗长的情况下重建未来。因此，我们转而使用 SLT 提取无重建预测潜函数作为 RL 的辅助任务。

B. 强化学习中的表征学习

良好的场景表示对于帮助智能体理解其环境和内部的复杂性至关重要

null 证明其决策能力。一种直观的方法是直接使用端到端网络绘制带有原始感官数据的策略[19]。然而，不可控的结果导致了分解为感知、决策和控制模块的共识[8]。端到端方法现在通过联合学习所有模块[21]，在超现实模拟器上测试它们的性能[20]。专注于决策任务，我们反过来利用感知过程后的中级观察[7]。具有多模态信息(例如，地图和交通代理)输入的驾驶场景最初以栅格化格式(图像)表示，该格式已用于基于视觉的驾驶系统，并显示出简单性和适应性。然而，栅格化的图像不可避免地失去了智能体和地图之间的明确关系[22]，这可能需要一个庞大的网络和更多的数据来恢复某些关系。一种并行的技术路线是交通 agent 和道路地图的统一矢量化表示[23]，广泛应用于前沿运动预测任务[24]、[25]。在我们的工作中，我们采用了密集的矢量化表示，统一了在线图搜索下智能体的位置和高清地图的路点[26]。它动态地提供了将地图信息与决策任务相关联的每个代理的本地意图。场景表示的另一个重点是对不同元素之间的交互进行建模(例如，agent-agent 和 agent-map)。一些研究已经在 RL 框架中探索了图神经网络(gnn)来制定多个智能体之间的相互作用。例如，DeepScene-Q[27]引入了基于 gnn 的编码器建模交互，用于城市驾驶决策。DQ-GAT 通过引入图注意网络(GAT)进一步改进了 DeepScene-Q 方法[28]。在这项工作中，我们采用了一种更强大的基于 Transformer 的交互建模结构，该结构在运动预测领域被广泛用于对环境进行编码[24]、[25]、[29]。

除了更有效的环境编码外，通过表征学习提高 RL 性能的另一个方面是将环境动态的预测知识提炼成潜在表征。全局值函数(Global value function, GVF)为场景表示提供了预测值[30]，但在复杂的城市场景中可能面临重要采样问题，面对稀疏奖励的改进有限。[31]提出了一种基于模型的模仿方法，从潜在空间中学习全局预测知识。此外，在 DrQ[33]和 CURL[34]中提出了一种学习自一致性[32]潜表示的增强方法，并已应用于基于图像的驾驶系统。自潜伏一致性最初是通过引入重建损失，通过基于自编码器的方法获得的[35]，这给系统带来了大量的计算负担。相比之下，自监督表示学习方法通过 Siamese 编码器直接引入自隐状态配对，消除了重建损失[36]。然而，大多数自监督方法只考虑当前或单步一致性对全局动力学的预测；然而，序列一致性是在部分可观察环境下获得更好表征的另一个关键因素。因此，在这项工作中，我们构建了一个增强

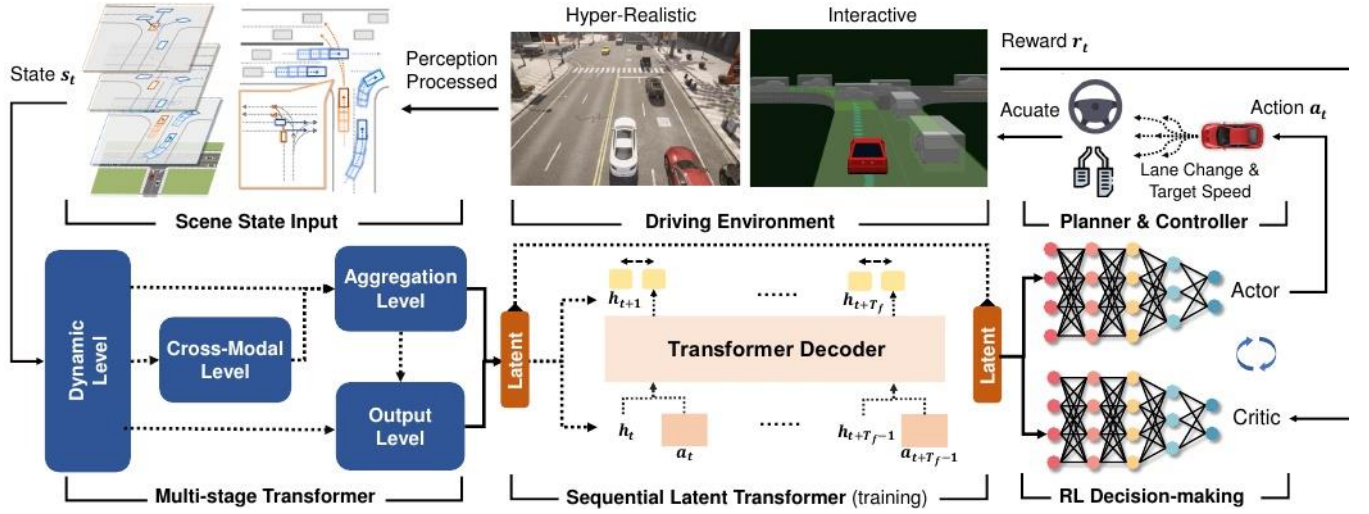


图 1 所示。概述我们的 RL 决策框架与场景再现变压器。给定感知处理的矢量化场景输入，多级变压器(MST)通过交互感知对多模态信息进行编码。在训练过程中，顺序潜伏变压器(SLT)使用连续潜伏-动作对进行表征学习，以确保未来的一致性。软演员-评论家(SAC)模块采用潜在特征向量为下游规划和控制任务做出驱动决策。实验在超现实和交互式驾驶环境中进行。

表示学习的方法，并遵循自预测表示(SPR)的思想[37]，它在雅达利游戏中取得了出色的性能。我们采用 Transformer 来更好地适应自动驾驶的部分可观察环境，而不是用于全局预测的循环结构。

3. 使用场景再现变压器的强化学习

答: 框架

所提出的 RL 框架侧重于在给定表示自动驾驶代理及其邻居的动态和地图信息的复杂状态输入的情况下进行智能决策。本节提供了所提出框架的概述，该框架由三个主要部分组成，如图 1 所示。多级 Transformer 将矢量化的场景表示作为输入，对 agent 之间的信息和交互进行编码，并输出一个潜在的特征向量。然后，顺序 latent Transformer 将跨时间的连续潜在特征向量与相应的动作嵌入一起，并输出一系列预测性潜在向量，这些潜在向量以顺序一致性进行训练。基于软演员临界的决策模块最终从场景编码器中获取潜在向量并生成决策输出。在推理(部署)阶段使用训练好的多级变压器和决策模块，而在训练期间使用顺序潜在在变压器进行表示学习[32]。

问题表述:在学习过程中考虑两个目标。对于 RL 决策，目标遵循用元组表示的马尔可夫决策过程(MDP) $_{t_{null}}$ ，一个 t_{null} ， $r_{t_{null}}$ ，年代 $(t+1_{null}, \gamma)$ 。具体来说，自动驾驶智能体接收状态表示 $s_{t_{null}} \in S$ 来自对环境的感知，并执行动作 $a_{t_{null}} \in A$ 根据其策略 $\pi(A|t_{null})$ 年代 t_{null} 。然后，环境返回一个奖励 $r_{t_{null}}$ (年代 t_{null} ，一个 t_{null}) 并过渡到下一个状态 $s_{t+1_{null}} \in S$ 。RL 的目标是优化策略 π

null 作为最大化 $p_{t_{null}}$ 期望累积折现收益: $\max E_{\pi_{t_{null}}}(\gamma \sum_{t=0}^{\infty} r_{t_{null}})$ 。序列潜伏 Transformer 模块中自监督学习的目标是强化预测潜伏 h 之间的序列相似性 $\text{sim}(h_t, h_{t+1})$ 和未来的 ground-truth $p_{t_{null}}$ $\text{sim}(h_t, h_{t+1})$ 在全球预测视界 TG : $\min TG L(h_t, h_{t+1})$ 。

状态表示:在时间步长 t 处，状态输入 $s_{t_{null}}$ 包含所有代理及其本地候选路由路点的历史运动状态:

$$s_t = [\mathcal{M}_t; \mathcal{K}_t],$$

其中 $M_t = \{M_{tego}, M_{t1}, M_{t2}, \dots, M_{tn}\}$ 包含受控自我车辆 M_{tego} 和附近 n 辆车辆 $M_{t1:n}$ 在一定距离 d_{null} 。每个运动轨迹是某一载具在历史视界 T 上的运动状态序列 h_{null} : $M_{t_{null}} = \{t_{null} - T_{hnull} + 1, t_{null} - T_{hnull} + 2, \dots, t_{null}\}$ ，每个运动状态 mt 包括位置、速度和朝向角度: $mt_{null} X = (t_{null} y_{t_{null}}, v_{x_{t_{null}}}, v_{y_{t_{null}}}, \psi_{t_{null}})$ 。 K_t 表示候选路线(见 A 节)， $K_t = \{K_{tego}, K_{t1}, K_{t2}, \dots, K_{tn}\}$ ，对应于场景中不同的 agent。每个 agent 的候选路由路点 K_{tego} 、 K_{tn} 由其当前位置 K 之前的一组本地候选路由序列组成 $t_{null} N = \{k_{n,1t}, k_{n,2t}, \dots\}$ 。每条局部候选路线由跨越未来视界 T 的一系列路点组成 K_{null} : $k_{t_{null}} = \{k_t, \dots, k_{t+T_k}\}$ ，其中 $k_t = (w_{xt}, w_{yt}, w_{\psi t})$ 为航路点的位置和航向角。它是智能体意图的简洁表示，因为它可以通过注意力机制勾勒出智能体未来的位置，提高决策的可解释性。状态表示 $s_{t_{null}}$ 由所提出的多级变压器编码，然后用于决策的 off-policy RL 算法。

这项工作的重点是决策，而不是设计端到端自动驾驶系统。因此，感知和控制模块将不进行讨论。相反，

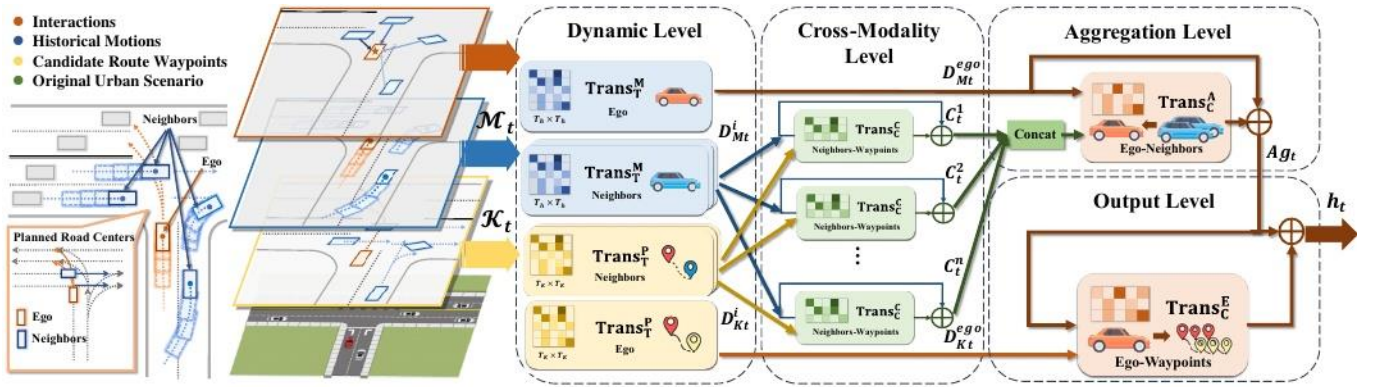


图 2 所示。MST 结构概述。左侧的城市场景可以分层为明确的表示，如历史运动状态 M_t 和候选路线路径点 K_t 。在后续的多阶段结构中对智能体交互的隐式层进行建模：1) 动态层沿时间轴编码 M_t 和 K_t ；2) 跨模态层对相邻智能体的交互进行建模，通过 3) 聚合层聚合到自我智能体；4) 输出层增加了自我-路径点和自我-邻居交互特征，输出潜在表征 h_t 。

从感知模块中获取状态表示 h_t 作为输入，我们的框架输出下游控制。

B. 多级变压器

多级变压器 (MST) 的体系结构作为一个分层编码器来映射场景状态 h_t 到一个潜在的特征 h_t 。结构为 $h_t \leftarrow \Phi_{MST}(h_t)$ 。按照 1) 模型应该有的直觉设计：1) 更好地捕捉驾驶场景的潜在动态；2) 候选路线信息被有效地用于未来意图的指导；3) 由于变压器网络的二次复杂度，计算效率得到保证；4) 注意机制可以很好地处理场景中 agent 数量的变化，具有卓越的可解释性。

我们将首先介绍多头注意，它是提议的 MST 结构中的核心机制，用于在不同级别上表示 Transformer 内部的交互。

$$\text{MHA}(\mathbf{Q}, \mathbf{K}, \mathbf{V}, \mathbf{M}) = \text{concat}(\text{head}_1, \dots, \text{head}_h) W^O, \quad (1)$$

其中，来自 h 个头部的注意结果被连接起来，以捕获信息的不同方面。每个头部捕获给定的 $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ 的注意加权值，表示由矩阵 $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V$ 转换的查询，键和值：

$$\text{head}_i = \text{Att}(\mathbf{Q} \mathbf{W}_i^Q, \mathbf{K} \mathbf{W}_i^K, \mathbf{V} \mathbf{W}_i^V, \mathbf{M}). \quad (2)$$

注意加权值通过与注意掩码 \mathbf{M} 的缩放点积计算得出：

$$\text{Att}(\mathbf{Q}, \mathbf{K}, \mathbf{V}, \mathbf{M}) = \text{softmax} \left(\frac{\mathbf{Q} \mathbf{K}^T}{\sqrt{d_k}} \cdot \mathbf{M} \right) \mathbf{V}, \quad (3)$$

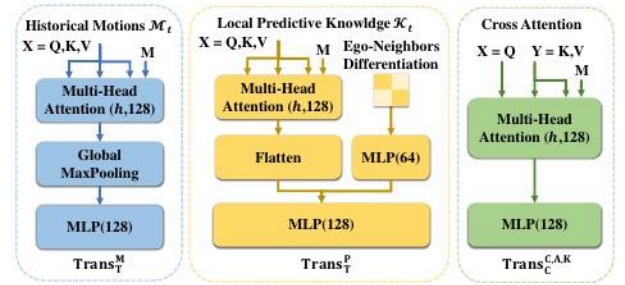


图 3 所示。在我们提出的 MST 中，沿不同轴的网络结构。

在维 k 为键的维数。注意掩码 $\mathbf{M} \in d_q \times d_k$ 允许不同数量的代理，因此城市场景的高度动态性可以很好地适应不同的元素和时间步长。

接下来，我们将提供详细的 MST 多级结构，如图 2 所示。该框架遵循随后的计算流程：1) 动态层，分别编码代理的历史运动和候选路由路点的表示；2) 跨模态层，对智能体及其候选路径点之间的跨模态交互进行建模；3) 聚合层，将相邻智能体的第二级潜伏组合到自我；4) 从第三阶段开始查询自我代理的候选路由路点的输出级别，与选定的信息流连接。每一层的编码都在下面详细介绍。

1) 动态级：给定输入状态 h_t ，动态电平分别对历史运动 M 进行编码 h_t 和对应的局部候选路点 K_t 对于每辆车。在此我们引入一个沿时间轴的单层 Transformer，用于动态电平编码。

$$\text{Trans}_T^M(\mathbf{X}, \mathbf{M}) = \text{MLP}(\text{MaxPool}(\text{MHA}(\mathbf{X}, \mathbf{M}))), \quad (4)$$

其中输入 $\mathbf{X} = \mathbf{Q}, \mathbf{K}, \mathbf{V}$ 表示时间特征的自关注机制，并采用最大池化来防止过拟合。

为了提取候选路线路点的时间特征，将车辆嵌入 Emb 连接起来，以区分自我车辆和邻居之间的分类特征，因为它们的表示在不同的情况下被查询

第2级和第4级的启发式。候选路径路径点的时间转换器表示为:

$$\text{Trans}_T^P(\mathbf{X}, \mathbf{M}) = \text{MLP}(\text{concat}(\text{MHA}(\mathbf{X}, \mathbf{M}), \text{Emb})). \quad (5)$$

我们省略了位置嵌入输入, 因为 $\mathbf{M}_{t, \text{null}}$ 和 $\mathbf{K}_{t, \text{null}}$ 本身携带密集的位置关系。输出动态嵌入可以表示为:

$$D_{Mt}^i = \text{Trans}_T^M(M_t^i, \mathbf{M}_M^i), d_{kt}^i = \text{Trans}_T^P(\mathbf{k}_t^{i,j}, \mathbf{M}_{kj}^i), \quad (6)$$

其中 $I \in \text{ego}, 1, 2, \dots, n$ 代表所有车辆(ego 和从 1 到 n 的邻居), $j \in 1, 2, \dots$ 是每辆车的候选路点个数。反式 $\text{M}_{\text{null}, T}$ 和 Trans_{LT} 表示分离的时间变换器, 编码历史运动和每辆车 \mathbf{M} 的所有候选路线路点 $\text{t}_{\text{null}}^{\text{innull}} \mathbf{K}_{\text{t}_{\text{null}}^{\text{innull}}}^{\text{innull}} = \{\mathbf{k}_{\text{t}_{\text{null}}^{\text{innull}}}^{\text{innull}} \mathbf{k}_{\text{t}_{\text{null}}^{\text{innull}}}^{\text{innull}}\}$ 带时间掩模 MiM , $\text{MiK} = \{\mathbf{M}_{\text{t}_{\text{null}}^{\text{innull}}}^{\text{innull}}, \mathbf{M}_{\text{t}_{\text{null}}^{\text{innull}}}^{\text{innull}}\}$ 。第一阶段输出车辆运动动力学 \mathbf{D} 的潜在表征 $\text{M}_{\text{t}_{\text{null}}^{\text{innull}}}^{\text{innull}}$ 并匹配候选路点的潜在集: $\text{DKt} = \{\text{di}, \text{kt}, \text{di}, \text{kt}, \dots\}$ 。

2)交叉模态水平:第二阶段寻求抽象相邻车辆的运动状态 DMti 与其局部路点 DKti 之间的相互作用。这一层次是在现实世界的启发式下建模的, 即相邻车辆之间没有合作, 因此未来的路点仅由它们自己的历史运动动态来查询。跨模态 Transformer 沿着预测数的轴线设计如下:

$$\text{Trans}_C(\mathbf{X}, \mathbf{Y}, \mathbf{M}) = \text{MLP}(\text{MHA}(\mathbf{X}, \mathbf{Y}, \mathbf{M})), \quad (7)$$

其中跨模态输入 $\mathbf{X} = \mathbf{Q}$, $\mathbf{Y} = \mathbf{K}$, \mathbf{V} 和掩模 \mathbf{m} , 跨模态动态表示为:

$$C_t^i = \text{Trans}_C(D_{Mt}^i, \text{concat}(D_{Kt}^i, \mathbf{M}_k^i) + D_{Mt}^i, \quad (8)$$

其中 $i = 1, 2, \dots, n$ 。加入残差连接以强调动态 $\text{D}_{\text{M}_{\text{t}_{\text{null}}^{\text{innull}}}^{\text{innull}}}$ 的相邻车辆, 同时保留 \mathbf{K} 不可用情况下的信息流 $\text{t}_{\text{null}}^{\text{innull}}$ 。

3)聚合级别:第三阶段关注自我车辆与其他相邻车辆的运动和地图信息流进行决策的代理级交互。仍然假设在场景中, 相邻车辆之间不存在交互。因此在这一层, 场景表示是通过聚合与自我动态本身 DMtego 和相邻车辆的交叉模态输出 $\{\text{Clt}, \dots, \text{Ctn}\}$ 进行编码的, 结构也遵循交叉注意方案:

$$\text{Ag}_t = \text{Trans}_C^A(D_{Mt}^{\text{ego}}, \text{concat}(D_{Mt}^{\text{ego}}, C_t^1, \dots, C_t^n), \mathbf{M}_n). \quad (9)$$

4)输出级别:输出阶段还计算自我车辆的未来启发式, 这应该通过邻居车辆的交互感知来确定。直觉是在给定自我动力学及其与周围邻居的交互的情况下查询候选路线路点, 这些邻居也知道他们的未来意图。另一个交叉注意转换器被设计用来模拟这种意图:

$$h_t = \text{Trans}_C^E(\text{Ag}_t, \text{concat}(D_{Kt}^{\text{ego}}, \mathbf{M}_k^{\text{ego}}) + \text{Ag}_t). \quad (10)$$

最终表现为 $\mathbf{h}_{t, \text{null}}$ 也通过输出层的残留连接传递。它是一个结构良好的表示, 给出了来自输入状态的不同层次的所有交互。

nullC. 软演员评论家

在给定潜在表征 \mathbf{h} 的情况下, 我们实现了软行为-批评(SAC)决策 $\mathbf{a}_{t, \text{null}} \leftarrow \Phi(\mathbf{s}_{t, \text{null}})$ 通过 MST 结构。SAC 是一种无模型的非策略强化学习方法, 在连续决策任务中具有最先进的性能[38]。它的目标是利用温度参数 α 最大化累积奖励和熵, 以规范勘探开发过程:

$$\mathbb{E}_{s_t, a_t \sim \pi} [\Sigma_t \mathcal{R}(s_t, a_t) + \alpha \mathcal{H}(\pi(\cdot | s_t))]. \quad (11)$$

SAC 同时学习随机策略网络 $\pi_{\Phi, \text{null}}$ (行动者)和双 Q 函数网络 $Q_{\theta, \text{null}1, \text{null}2}$, 用于 Q 值的方差减少(批评家)。给定潜在表示 $\mathbf{h}_{t, \text{null}}$, Q 函数网络根据从回放缓冲区 \mathcal{D} 中采样的 MDP 元组 $\tau = (st, at, rt, st+1, \gamma)$ 上的平均贝尔曼平方误差(MBSE)进行更新:

$$\mathcal{L}_c(\theta_i) = \mathbb{E}_{\tau \sim \mathcal{D}} [(Q_{\theta_i}(h_t, a_t) - (r_t + \gamma y_Q))^2], \quad (12)$$

其中 $I = 1, 2$, 潜在场景表示 $\mathbf{h}_{t, \text{null}} \leftarrow \Phi(\mathbf{s}_{t, \text{null}})$ 是通过 MST 获得的。时差(TD)目标 y_Q 由 Q 函数更新由上面的 MDP 元组 τ 给出, 动作由当前策略 π 采样:

$$y_Q = \mathbb{E}_{a' \sim \pi} \left[\min_{i=1,2} Q_{\theta_i}(\bar{h}_{t+1}, a') - \alpha \log \pi_{\Phi}(a' | h_{t+1}) \right], \quad (13)$$

其中 $\mathbf{h}_{t+1} \leftarrow \Phi(\mathbf{s}_{t+1})$, $\bar{h}_{t+1, \text{null}} \leftarrow \Phi^-(\mathbf{s}_{t+1})$ 。在评论中, 目标 q 值及其表示 $\bar{h}_{t+1, \text{null}}$ 通过目标网络 Φ^- 获得, 目标网络在每个梯度步上通过 Polyak 平均动态更新。双 Q 网络的最小 q 值 $\theta_{1,2, \text{null}}$, 用来防止高估状态动作值的问题。

参与者策略网络输出多变量高斯分布 $\mathcal{N}(\mu, \Sigma)$ 的参数(均值和协方差 $\mu, \Sigma \in \mathcal{A}$), 每个参数都具有动作空间的维度。行动者通过软 q 最小化来更新:

$$\mathcal{L}_a(\phi) = - \mathbb{E}_{\tau \sim \mathcal{D}} \left[\min_{i=1,2} Q_{\theta_i}(h_t, a_t) - \alpha \log \pi_{\Phi}(a_t | h_t) \right], \quad (14)$$

其中策略取自 MST 的状态表示, 但停止梯度反向传播: $\mathbf{h}_{t, \text{null}} \leftarrow \text{sg}(\Phi(\mathbf{s}_{t, \text{null}}))$ 。参数 α 控制熵项的权衡。在训练过程中是否自动调优

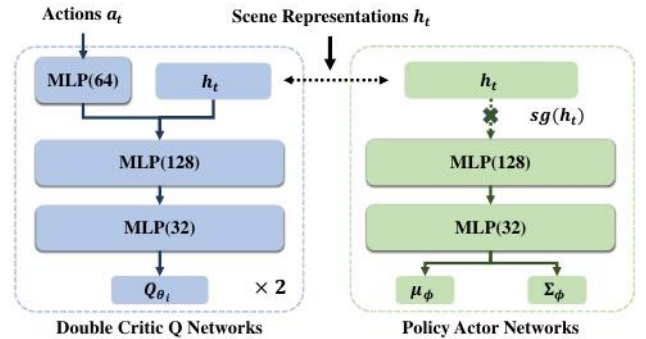


图4所示。演员-评论家网络的结构。对于连续决策任务, 采用均值和协方差对角线的多元高斯模型对策略进行建模。

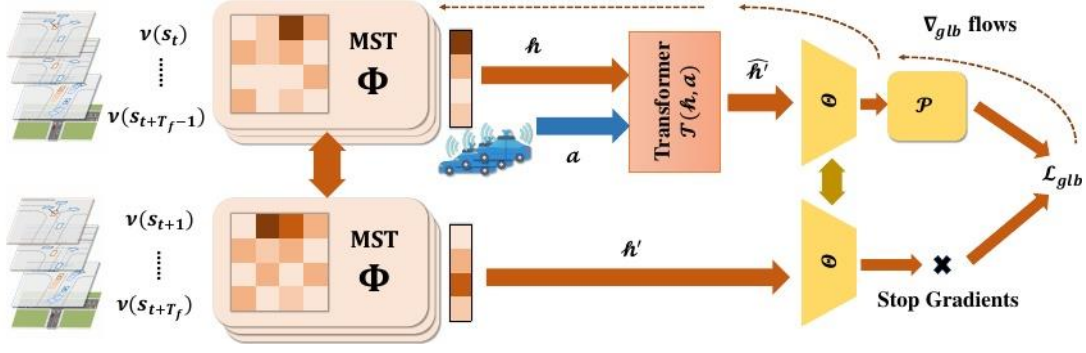


图 5 所示。顺序潜伏变压器的计算流程。通过这种暹罗网络结构捕捉到未来视界之间的连续一致特征。所有增强的未来状态都由 MST Φ 编码。然后，它遵循具有表示转移模型 T 的顺序表示学习，并获得该辅助任务的投影相似度损失 L_{glb} 。SLT 框架的证明可以在定理 1 中找到

[38]。动作从当前策略 a 中采样 $t_{null} \sim \pi_{\phi_{null}}(\cdot | \Phi(s_{t_{null}}))$ ，取策略 a 的均值 $t_{null} = \mu_{\phi_{null}}(\cdot | \Phi(s_{t_{null}}))$ 在测试期间。

D. 顺序潜伏变压器

从全局的角度来看，决策过程可以通过事先了解决策结果而得到改善。这可以改进为：1) 全局预测提取减少探索空间的连续状态-动作对；2) 路线路径点 K 的注意权值 t_{null} 可以被全局预测隐式引导。遵循指导方针，应该设计一个辅助任务来学习潜在在表征 $h_{t_{null}}$ 可预测到下一步状态 $h_{t+1_{null}}$ 以当前动作为条件 $a_{t_{null}}$ 在培训。可以将其建模为前向环境过渡：

$$h'_{t+1} \leftarrow T(h_{t+1} | h_t, a_t). \quad (15)$$

尽管如此，单步前向模型面临一系列问题：1) 表示是顺序依赖的，因此结果可能由一系列动作导致；2) 驾驶场景的高度动态性需要提取序列表征来对抗方差；3) 自编码器方法在复杂的城市状态重构中积累了巨大的误差。此外，当前辅助任务的建模缺乏分析独创性。因此，在训练过程中提出了一种序列潜变(SLT)作为辅助表征学习任务。我们还推导了所提出的管道的理论证明。

为了处理第一个问题，应将 Eq. 15 中的过渡模型修改为自回归结构，以模拟场景表示和未来视界 T 下相应动作之间的顺序关系 f_{null} ：

$$h'_{t+k} \leftarrow T_{k \leq T_f}(h_{t+k} | h_{<t+k}, a_{<t+k}). \quad (16)$$

对于第二个问题，引入了一种表示学习方法。该框架通过暹罗网络计算相同状态输入的不同方面之间的投影潜在在相似性，从输入的不同方面捕获不变特征。第三个问题也得到了解决，因为该方法是无重建的。为了解决方差问题，我们采用了表示学习方法

$h_{t_{null}}$ ，它增加了输入 $v(s_{t_{null}})$ ，通过用增广的不同方面平均相同的表示来减少方差。它也有利于决策，因为 q 值高估问题将在状态输入的不同方面使用相同的 q 值进行正则化 [33]。

SLT 计算框架：图 5 显示了 SLT 的结构，它遵循 **simsiam** 风格的表示学习方法 [39]。给定未来视界 T 上连续的状态-动作对 f_{null} ：

$$(v(s_t), a_t), (v(s_{t+1}), a_{t+1}), \dots, (v(s_{t+T_f}), a_{t+T_f}),$$

那么所有增广状态都将通过 MST: h 进行编码 $t_{null} \leftarrow \Phi(v(s_{t_{null}}))$ 。设 h, a 表示表示和动作序列 $h < T_{k_{null}}, < T_{k_{null}}$ ，未来的步长表示可以集成 $h' = h_{t:t+T_{k_{null}}}$ 。如图 6 所示，通过基于变压器的自回归解码器获得预测的未来潜在表示 h' ， $h' = T(h, a)$ 。 h' ， h' 都用预测器进行预测，用于相似性计算： $z = \text{sg}(\Theta(h'))$ ， $z' = P(\Theta(h'))$ ，其中 Θ and P 分别是 MLP 投影仪和线性预测器。为目标停止梯度流 z 。全局预测的相似度损失 $L_{glb_{null}}$ 与 RL 决策同步更新。在推理过程中，我们严格遵循一般表征学习范式 [32]，它只保留增广 Φ without 更新损失项。在算法 1 中给出了 SAC 与所提出的场景再现变压器的实现。

相似度损失：为了防止表示崩溃，我们提出余弦相似度作为相似度损失函数 [40]：

$$L_{glb}(z, \hat{z}) = -\frac{z \cdot \hat{z}}{\|z\| \cdot \|\hat{z}\|}. \quad (17)$$

损失是单独更新的，因为所有目标不具有相同的大小，并且可能为了优化而相互冲突。涉及相似损失计算的神经网络分别是 Φ 、 T 、 Θ 和 P 。

状态增强(State Augmentation)：给定输入状态时的状态增强 t_{null} 针对驾驶场景，包括两个步骤。1) 将绝对坐标转换为以自我车辆为中心沿其纵轴的笛卡尔坐标，以更多地关注局部变化。2) 所有点在 $[-]$ 之间进行随机旋转处理 $\pi_{null}^{null} / 2_{null}$ ， $\pi_{null}^{null} / 2_{null}$ 由自我中心构建

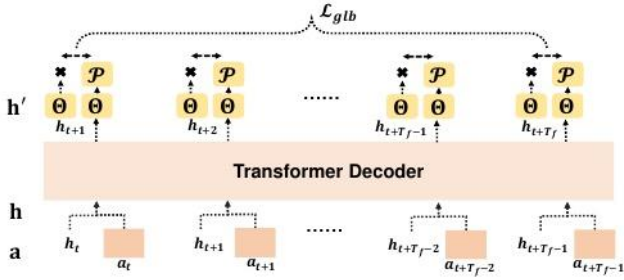


图 6 所示。基于变压器的过渡模型 T 的结构。它是时序转换器解码器给出了每个时间步长的 h_t 的连接输入，输出是下一步的表示预测，作为预测和预测的自回归方式。

对状态的不同看法。

数据收集:在训练期间，我们将转换元组 τ 收集到重播缓冲区中，同时维护一个队列 D^{null} 收集相应的未来状态-行为者对 $(s, a) = (s_t:t+t_{f,\text{null}} : t+t_{f,\text{null}})$ 在时间范围内 T_f ;

SLT 分析验证:SLT 旨在为可预测的表示提取核心一致的未未来特征。理论上，它被客观化为最大化未来 γ_{null} 信息瓶颈 γ_{null} 对于潜在动态模型 $T: \max I(h, s|a) - \beta I(h, s)^{\text{inull}}|a)$ ，其中 I 为采样指数[41]。SLT 的理论证明可以在定理 1 中找到

四、实验

为了验证场景再现变压器在真实场景特征下的决策性能，我们引入了 CARLA[20]和 SMARTS[42]两个真实仿真平台进行全面验证。在 CARLA 中，我们专注于超现实动态和城市元素的验证。在 SMARTS 中，模拟具有超现实的相互作用和随机交通流。

A. 驾驶场景

为了彻底评估现实世界动态和城市真实场景中的决策效率，我们将 CARLA 模拟器纳入验证目的。CARLA 提供了超现实的物理和动态，为每个参与者的交通行为提供了内置的人工智能代理。在我们的实验中，我们在 CARLA Town-10 选择了一个具有代表性的城市场景。如图图 7 所示，自我自动驾驶车辆被分配到一个要求苛刻的无信号城市交叉口进行决策验证。

在本任务中，自我载体被授权与图 7 a 所示的不同类型的参与者进行竞争。每个参与者都由人工智能代理控制，在交互过程中随机分配行为(耐心、攻击性、温和)。对于图 7 b)所示的任务完成，侧道上的自我车辆应首先设法左转。它将面对互动交通，包括:来自两个相对车道的迎面交通，来自左侧车道的左转交通，以及穿过人行横道的行人。此外，ego 车辆的任务是执行至少一次变道，避免右转交通到达目的地区域。这个选定的场景检查了现实动态和多样化城市交通下的回避、转弯和变道的交互式驾驶决策。

交互感知是场景再现变压器的核心贡献。因此，必须在实际交通交互领域确认其决策能力，包括更广泛的随机行为和交通动态。为此，我们致力于通过 SMARTS 模拟器中的仿真平台推进我们的验证，以进行真实世界的交通流量和交互模拟。

为了体现现实交互和行动者行为下的决策绩效，我们在官方 SMARTS 平台中选择了三个具有挑战性的场景。SMARTS 提供了超现实的交互和不同情况下的交通流

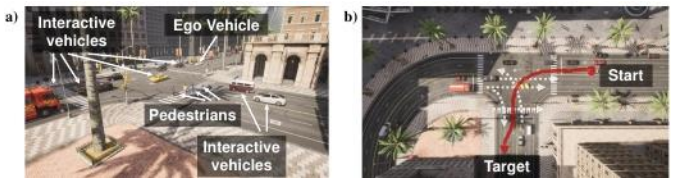


图 7 所示。CARLA 镇 10 的互动场景。a)设计的有车辆和行人的左转场景;b)任务设置的鸟瞰图。红线表示一个示例自我轨迹。白线表示社会行动者(车辆和行人)可能的互动轨迹。

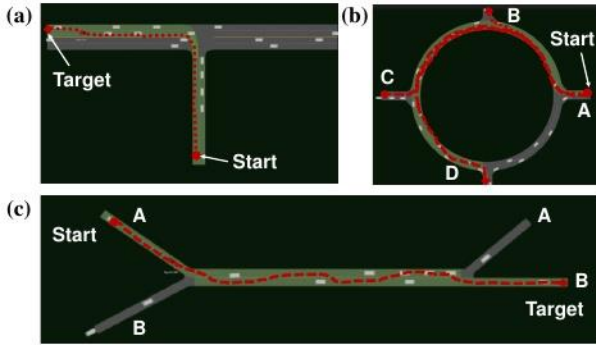


图 8 所示。在 SMARTS 仿真平台上设计城市实验场景。a)无保护左转;b)回旋处;c)双合并。绿色表示自我车到达目标的可行区域。红点提供了从起始路线到任务目标的示例

粒度。它进一步提供了基于更接近现实的物理引擎的随机演员行为和缺陷。如图 8 所示。从现实交互和驾驶行为的不同方面选择场景来验证我们的框架。

a)无保护的左转:该场景建立在交通繁忙的城市无信号 T 型十字路口。自动驾驶车辆被要求在没有交通信号保护的情况下完成左转任务，因此交通流不会停止。从 2 车道单行道的次要道路开始，ego 车辆需要在正在进行的交通流中做出明智的决策，进入 4 车道双向主干道上最右边的车道。这个场景主要关注智能交通流穿越的机动。

b)双合并:所面临的场景是一个双入口双出口的双车道合并路段。请注意，设计的交通流比场景 a)更重，因为对于每个入口，所有出口都分配了交通流。自动驾驶汽车被要求从 A 出口合并到 b 出口，这就对自动驾驶汽车学习变道集中机动以完成任务提出了更高的要求。

c)回旋处:设计为具有 4 个出口的一般城市回旋处，交通流量密集。自动驾驶汽车以穿越 A 为切入点，设计了 3 个任务，难度依次递增，分别为出口 B、C、d，需要更复杂的决策和更远的距离来完成任务。

为了确保客观和实际的验证，我们对齐了 CARLA 和 SMARTS 的情景设置。起点是沿着起始路线随机分配的，而目标是某条路线的终点。模拟的重置发生在以下情况下:1)自我车辆到达目标;2)情景步长超过最大时间步长限制;3)自我车辆与其他物体碰撞或驶离路线。为了建立实际的模拟，CARLA 和 SMARTS 中的两个场景都是根据官方配置设计的。在 CARLA 场景中，参与者是随机生成的，并安装了内置的 AI 代理，以展示任意选择的各种行为。对于 SMARTS 模拟，每个场景都包含一个官方默认设置，如表六所示。在宏观粒度上，整个场景按密集流量级别配置。基于默认值

对于逼真的行为，在 CARLA 上对演员进行更随机和逼真的行为采样。模拟间隔设置为 0.1s。

B. 决策过程

所设计的状态空间、动作空间、奖励函数如下所示。

1)状态 s:状态输入 $s_{t, null} = [M_{t, null}; K_{t, null}]$ 有两种不同的形式。对于历史运动，它的状态空间 $S_{M, null} \in [n + 1, T, h_{null}, 5]$ 表示车辆(自我和邻居)的数量，历史步骤和特征维度。对于局部候选路径路点，其状态空间为 $S_{K, null} = [n + 1, n_{k, null}, T_{K, null}, 3]$ ，其中第二维表示局部候选路线 k 的最大个数，可以根据不同道路结构的最大车道数进行调整。

2)动作 a:此处我们将下游控制器执行的战略级决策结合起来，而不是对加速和转向的端到端控制。决策动作 $a_{t, null} = [V_{t, null}, L_{t, null}]$ 由目标速度 V 的二维矢量组成 t_{null} 和变道 $L_{t, null}$ 。前者为连续输出 $V_{t, null} \in (0, V_{max, null})$ 范围从 0 到最大速度限制 $V_{max, null}$ 。后者是一个离散动作 $L_{t, null} \in \{-1, 0, 1\}$ ，其中 $L_{t, null} = \pm 1$ 表示左/右变道，而 $L_{t, null} = 0$ 表示保持车道。根据决策命令，来自 SMARTS 模拟器的内置运动规划器[42]将提供相应的路线，然后使用控制器在 ego 车辆上执行车道跟随控制(横向和纵向)。请注意，对于 L 来说，一个完整的变道过程需要多个步骤 $t_{null} = \pm 1$ ，否则会被控制器修正回当前车道。在 CARLA 模拟器中也遵循类似的过程。根据动作命令，搜索和处理的运动可行规划路线用于后续控制，而不是 CARLA 中提供的原始点。

为了对连续和离散空间的混合动作建模，我们引入了一个二维归一化多元高斯分布。 V 的 $t_{null} \in (0, V_{max, null})$ ，则归一化为 $[-1, 1]$ 。对 $L_{t, null}$ ，我们将高斯的第二维离散为 3 个大小相等的箱子 $[-1, -1/3], [-1/3, 1/3], [1/3, 1]$ 为 $L_{t, null} = 0$ ， $[-1, -1/3]$ 为 $L_{t, null} = -1$ ， $[1/3, 1]$ 为 $L_{t, null} = 1$ 。我们选择这个设置是因为 1)目标速度应该是精确的，因为它对安全至关重要;2)速度和变道决策行为在某种程度上是相关的。

3)奖励函数 $r(s, a)$:为了强调框架对所提出的场景再现变压器的贡献，我们选择一个简单且稀疏的基于目标的奖励作为奖励函数:

$$r_t = r_{target} + r_{penalty}, \quad (18)$$

其中，如果自我车辆到达目标， $r_{target} = 1$ 是一个指标，如果自我车辆与其他车辆碰撞或驶出可行路线， $r_{penalty} = -1$ 。在其他情况下，简单奖励保持为零。为了彻底验证，我们的实验还结合了两个标注的奖励函数[8], [43]，用于自动驾驶决策。结果可以在 ¹ 找到。我们还期望通过奖励塑造来提高结果。

¹ <https://github.com/georgeliu233/Scene-Rep-Transformer>

C. 评价指标

为了公平地评估所提出框架在不同场景下的性能，我们使用训练期间的平均成功率作为主要评估指标。在测试阶段，我们使用额外的度量标准，即：，碰撞率和停滞率。

- 成功率(suc. %):量化自我车辆成功到达目标的情节百分比。在测试过程中，我们衡量的是总测试集的成功率。对于训练用例，成功率是在过去 20 集中测量的。
- 碰撞率(col. %):它计算自我车辆与其他车辆碰撞的剧集百分比，这是安全性能的关键指标。
- 停滞(stagg. %):它计算自我车辆保持静止并超过最大时间限制的情节比例，这表明了保守的程度。

D. 比较基线

为了对所提出的方法进行全面评估，我们将 SAC 与场景再现变压器与其他现有的场景表示和决策方法进行了比较。基准方法是：

- 1)数据正则化 q 学习(data - regularization Q-learning, DrQ)[33]:一种最先进的基于图像的 RL 方法(Dr-SAC 用于连续动作设置)。堆叠栅格化图像用于在连续时间步长上表示具有道路和车辆的场景。这些图像还将经过随机增强以进行正则化。它采用卷积神经网络(CNN)编码状态和软行为者-评论家(SAC)算法进行决策。
- 2)软演员评论家(Soft Actor Critic, SAC)[38]:一种不需要场景理解框架的 RL 基线决策方法。它对状态输入采用相同的表示，但使用 LSTM 进行编码，并汇总每一步的信息。
- 3)近端策略优化(PPO)[44]:一种最先进的决策策略方法。它改进了 RL 策略，考虑了在情景视界内使用代理演员-评论家目标保持接近最后更新策略的约束。我们采用最常见的基于图像的方案作为输入(与基线 1 相同)。
- 4)基于规则的驾驶员模型(Rule-based Driver Model, RDM)[42]:使用与相邻车辆相同的行为模型实现，但具有确定性设置和无缺陷的行为。
- 5) Decision Transformer (DT) [18]: Transformer 实现 RL 决策的强大基线。它通过变压器解码器将 RL 建模为 MDP 序列的顺序学习。

E. 实现细节

我们提出的方法和其他基线方法(RDM 除外)在每个城市场景中都进行了 100,000 步的训练，每个实验使用不同的随机种子进行 5 次。对于 DT，我们成功地重放了 500 次日志

null 每个场景的 MDP 轨迹，并按照其原始实现进行训练。所有方法的神经网络都使用 Tensorflow 和 Adam 优化器在单个 NVIDIA RTX 2080 Ti GPU 上进行训练，学习率为 1e-4，单个场景下一种方法的训练过程大约需要 2 小时。与实验相关的参数列于表 VII 中。对于每种方法，我们在训练过程中取成功率最高的策略网络，用于后续的测试阶段。

V. 结果和讨论

A. 培训结果

我们首先将 Scene- Rep Transformer 的训练性能与比较基线一起进行评估。训练结果由城市市场景显示，其中每条训练曲线代表平均训练成功率的某些基线的平均值(实线)和标准差误差带。每 200 步记录一次结果，并通过指数移动平均(EMA=0.99)进行平滑处理。图 9 显示了不同方法在所有场景下的训练结果，以及消融方法(MST SAC)。

结果表明，基于场景再现变压器的 SAC 在所有 5 个不同任务的城市市场景中都获得了最佳的训练性能。所有场景下的训练成功率都得到了提升，尤其是在环行- c、无保护左转弯、双合并等要求更高的城市市场景下。所提方法的样本效率也得到了提升，在更少的训练步骤中显示出更快的收敛速度。结果表明，我们提出的场景再现变压器可以快速适应不同的场景，并且能够在仅仅 50% 的总体训练步骤中达到 70% - 80% 的最终成功率。在这里，我们给出了一些详细的解释，说明我们提出的方法与基线方法相比具有优越的性能。

与 on-policy RL 相比，PPO 可以很好地适应更简单的任务，如 roundround - a(图 9c)，具有快速收敛性，平均成功率为 0.7，仅在 40k 训练步骤中收敛。但当遇到更复杂的交通和更长的里程时，它的表现就会变差。在最初的政策改进(20k-30k)之后，训练曲线仍然是平坦的。这部分是由于 on-policy 学习对奖励和参数的敏感性。这可能导致数据不足，并且在处理随机交通流和远程机动时需要进行更密集的微调。

与无策略 RL 相比，SAC 驾驶策略在驾驶场景中表现出令人满意的随机性。与 PPO 相比，具有相同状态输入的 SAC 显示出最终成功率(0.5-0.73)的显著提高。然而，SAC 的收敛和不稳定性留下了很长的时间，这反映在不同试验之间的较大方差上($\sigma > 0.1$)(见图 9a, b)。这主要是由于 q 值估计或状态表示编码的状态动作对采样不足造成的。

与高效 RL 相比，与 SAC 基线相比，DrQ 将成功率从 10.5% 提高到 49%。它利用数据增强技术，使 q 值估计的方差可以快速收敛

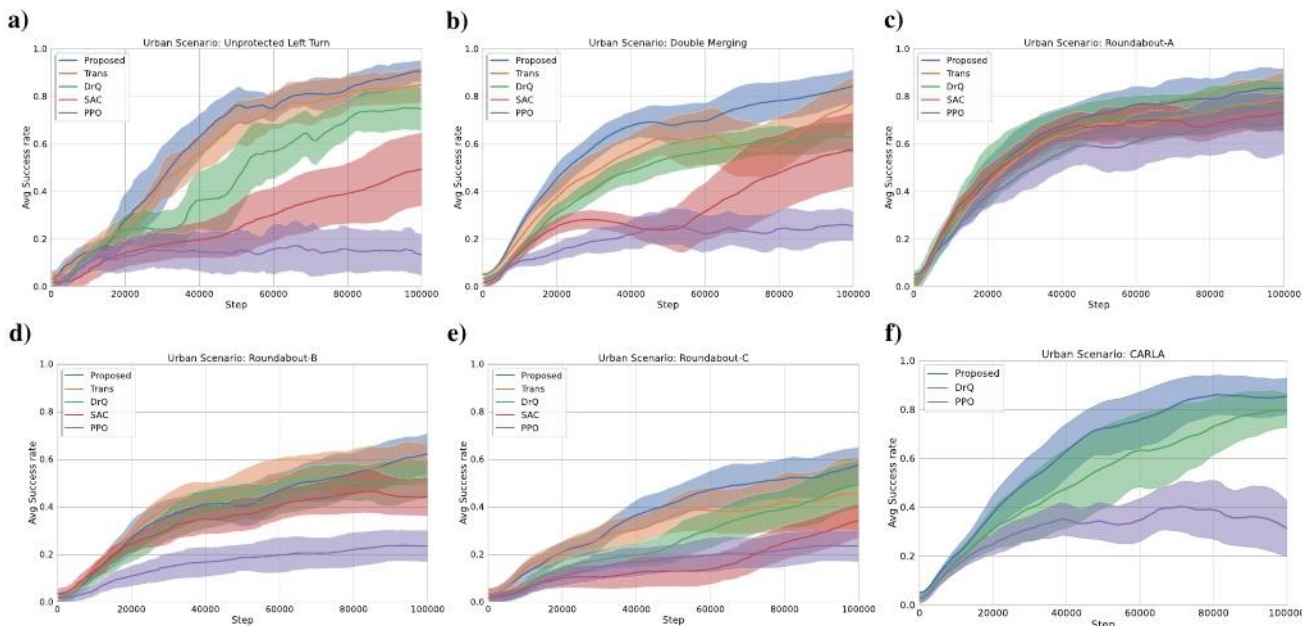


图 9 所示。我们的框架使用基线方法在不同城市场景下的平均训练成功率量化的训练过程:a)无保护的左转弯;b)双合并;c)环形交叉口- a:自我车辆到达 B 出口;d)回旋处- b:自我车辆到达 C 出口;e)回旋处- c:自我车辆到达 D 出口;f) CARLA 中的交互式城市场景。蓝色曲线(Proposed)表示场景再现变压器(Scene-Rep Transformer), 橙色曲线(Trans)表示仅使用 MST 的烧蚀训练性能。其他三条训练基线显示为绿色(DrQ), 红色(SAC)和紫色(PPO)。

相同 q 值的状态的更多方面。它显示了 40k-50k 训练步骤的更快收敛, 以达到与 SAC 基线相同的性能水平。然而, 性能仍然受到栅格化图像输入和 CNN 结构的限制, 因为图像不能显式地模拟车辆之间的相互作用。因此, 在合并等高度交互场景下, 其最终性能接近 SAC 基线的上误差带(图 9b)。

与仅 MST 相比, SAC 与 MST 在 SAC 基线的基础上提高了所有场景下的成功率, 并将收敛效率提高到 50% - 66% 的训练步长, 以达到与 DrQ 基线相同的性能。这也表明框架中的 SLT 可以保持顺序驾驶场景与其决策之间的一致性。因此, 所提出的方法的性能优于仅 MST, 从左转弯场景的 84%(图 9a)到最简单的环形交叉口 a 情况的 23%(图 9c)。大大提高了收敛性也是该框架的一个优势, 只需 40% - 60% 的训练步骤即可达到 DrQ 方法的最佳性能。

在 CARLA 环境中的表现:在 CARLA 环境中也报道了类似的训练表现。进一步验证了在不同交通参与者和真实动态的真实城市场景下的学习能力。由于不同的图像设置, 这里我们直接使用 MST 作为所有基线的场景编码。因此, DrQ 和 PPO 的性能得到了提高, 这也体现了多阶段编码的有效性。

综上所述, Scene-Rep Transformer 在训练过程中的性能提升可以源于三个方面。首先, 采用多级变压器结构 (multi-stage Transformer structure, MST) 来表示自我车辆与其相邻车辆之间的相互作用, 并采用局部候选路径点的自适应融合来表示局部意图

null 自我和邻居的相互作用, 以进行决策。其次, 采用 SAC 算法进行决策, 满足城市场景的学术性。最后, 通过时序潜变量(SLT)进一步提高采样效率。通过全局预测知识的指导, 减少了状态-动作探索空间, 全局预测知识只保留了不同城市场景中顺序未来状态-动作对之间的一致信息, 从而在有限的训练步骤下提高了收敛性。

B. 测试结果

测试场景是用相同的道路网络构建的, 但在随机性和随机种子方面对表 VI 中的参数进行了不同的设置。与其他基于学习的决策方法相比, 在测试过程中还引入了基于规则的方法(RDM)。下面的每个基线和场景再现变压器都在每个城市场景下测试了 50 集。在测试过程中, 三个评估指标(成功率、碰撞率和停滞)的统计数据如表 1 所示。注意, 这三个指标加起来不一定等于 100%, 因为自我车辆代理偶尔可能会到达错误的目的地。

表 1 中的测试结果与训练结果一致。我们提出的场景再现变压器 RL 框架在所有五个城市场景中都实现了最佳性能。所提出的方法带来了最高的测试成功率和更少的碰撞故障案例。除了全局预测知识带来的性能外, 变压器辅助的局部预测能力也在 MST 基线中得到验证, 与 DrQ 基线相比, 总体上有所改善。然而, MST 仍然受到其对状态空间的唯一指导的限制, 而不是通过全局预测顺序引导两个状态动作空间。因此, 对于城市场景来说

表我

	Left Turn			Double Merging			Roundabout-A			Roundabout-B			Roundabout-C		
	Succ.%	Coll.%	Stag.%	Succ.%	Coll.%	Stag.%	Succ.%	Coll.%	Stag.%	Succ.%	Coll.%	Stag.%	Succ.%	Coll.%	Stag.%
RDM	2	54	44	0	100	0	68	30	0	2	98	0	0	100	0
PPO	36	50	10	36	64	0	66	34	0	42	58	0	38	50	12
SAC	68	28	0	62	22	0	76	24	0	48	52	0	46	48	6
DrQ	78	20	0	76	14	0	80	20	0	72	28	0	68	30	2
DT	66	32	0	70	30	0	76	22	0	68	32	0	66	30	0
MST	88	12	0	92	4	0	84	16	0	76	24	0	66	34	0
Proposed	94	4	0	96	2	0	88	12	0	82	18	0	76	24	0

表二世
任务完成时间测试结果(s)

	Left Turn	Double Merging	Roundabout		
			A	B	C
PPO	36.4±6.7	36.3±3.0	23.0±4.9	43.5±1.8	63.5±1.7
SAC	19.2±0.5	34.9±1.9	25.2±1.3	44.0±1.1	60.7±1.4
DrQ	18.2±0.4	34.9±1.7	22.4±0.3	40.3±0.5	57.4±1.8
DT	26.3±4.6	26.7±5.6	25.5±4.8	34.4±4.2	50.1±6.7
MST	17.3±2.9	30.9±2.5	28.3±3.3	35.3±1.9	58.5±2.2
Proposed	12.5±0.4	28.6±0.6	24.5±1.2	33.7±1.2	56.6±2.2

需要长时间的驾驶操作(表 1 中的环形交叉口 b 和 C)，使探索空间复杂化，MST 的性能与 DrQ 相似，因为它们都只强调状态空间。SAC 和 PPO 在训练结果上的表现是相似的，这主要是由于在前一节中说明了它们的局限性。与之前的结果一致，DT 在长期任务(R-C)中表现更好。然而，由于冗长和缺乏交互建模，DT 被交互式 and 短期任务(左转弯)所损害。值得注意的是，基于规则的驾驶员模型在测试中表现得很糟糕，因为它无法应对随机交通流，并与攻击性车辆发生碰撞，以切断自我车辆的行驶路线。

对于关于驾驶效率的度量(停滞率%)，基于学习的基线显示出相当大的性能，很少有自我车辆停滞的情况。超过时间步长限制的案例主要集中在 PPO 和 SAC 的 roundaround - c 场景，因为它需要更长的任务完成时间，并且在这种场景下可能会遇到交通堵塞。效率低下的最高停滞发生在无保护的左转弯中基于规则的模型。由于基于规则的驾驶智能体在面对来自主干道的密集交通时存在过度保守的行为，自我车辆将在 t 型十字路口保持等待，直到达到时间限制。测试结果清楚地表明，当遇到更复杂和随机设置的城市场景时，基于规则的驾驶模型表现不佳。

为了进一步研究使用场景再现变压器的基于学习的基线的驾驶效率，我们计算了每个城市场景中测试时段成功试验完成时间的平均值和标准差(表 2)。结果反映了该方法在大多数城市场景下提高了行驶效率。PPO 和 SAC 代理完成任务所需的时间较长。它们都表现出严格遵循 PPO 常规操作的试探性行为，或者由于环境因素而表现出随机性行为

null 表 3
CARLA 模拟的测试结果

	Succ.%	Coll.%	Stag.%	Completion time (s)
PPO	56	38	6	22.6±1.7
DrQ	84	10	4	21.3±1.4
DT	70	18	8	22.6±1.5
Proposed	88	6	2	22.9±1.6

SAC 局部预测偏差。由于训练过程中的数据正则化机制，DrQ 具有更高的运行效率和更小的方差。通过对多步未来对的数据增强，提议的场景再现变压器进一步扩展了这一点，从而产生了类似的功能。

为了验证动态和演员多样性的超现实表演，我们在 CARLA 中进行了与 SMARTS 中使用的相同设置和指标的测试。结果显示，所有基线都有类似的趋势，与 SMARTS 中的左转弯场景相比，性能略有下降。这种下降部分归因于更复杂的动态和交通参与者(即行人和社会车辆)的多样化范围，这些参与者在我们的框架中没有针对其类型进行专门建模，并直接服务于状态输入。值得注意的是，CARLA 的性能下降幅度大于 SMARTS。这可能是由于 CARLA 中其他参与者的相互作用偶尔产生的死锁效应。

C. 消融研究

为了探索预测表示学习和候选路径点在场景再现变压器中的作用，进行了一项烧蚀研究。在训练好的 MST 模型上，用部分分量建立了两条烧蚀基线:1)Ego: MST 去除 Ego 车辆路径路径点 DKtego 的输出电平 Transformer;2)邻居(N):即 MST 移除所有与 K 相关的分量 tnull;3) Neighbor-SLT (N- SLT):由 SLT 增强的邻居再训练基线。所有烧蚀基线在 Section V-B 中使用相同的测试设置进行测试，结果显示在表 IV 中。

Ego 基线的结果暗示候选路线路径点 Ktnullego null 与有效的 RL 基线(如 DrQ)相比，是 MST 性能提高的关键。这是因为自我智能体的路径路径点为自我车辆提供了一个过滤区域，以呈现给定相邻车辆及其相应意图的信息(Ktnullnnull)，增强了场景理解能力。因此，没有 K，性能下降明显 tnullego null 在这些有多个潜在意图的场景中

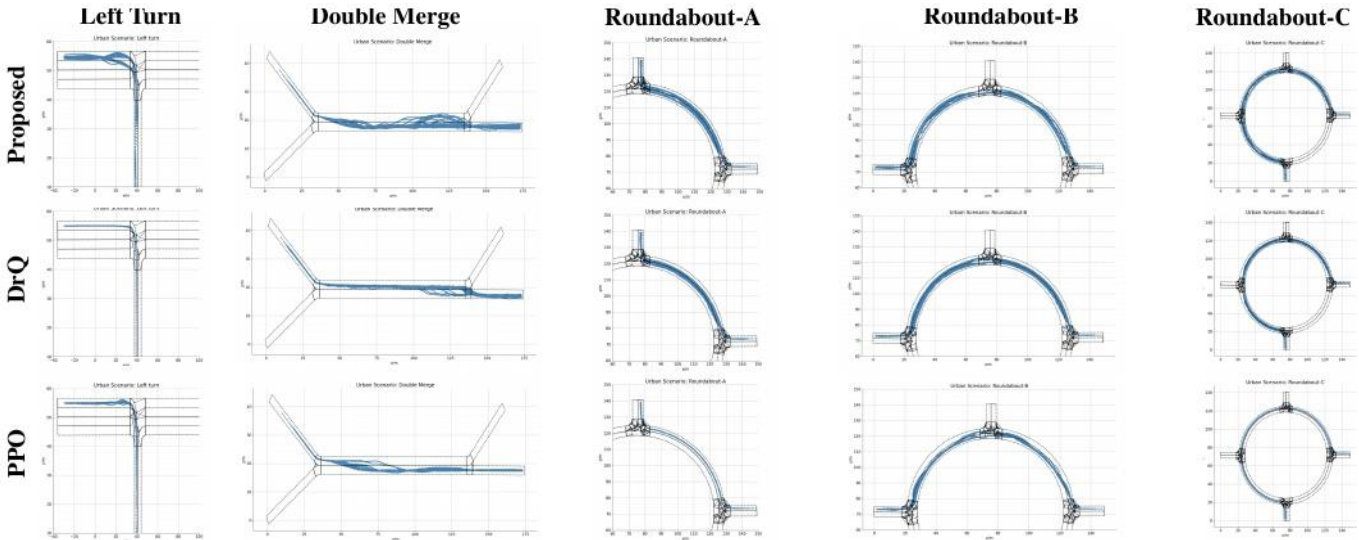


图 10 所示。在测试阶段，成功完成给定任务的自动驾驶车辆的轨迹。列表示五种不同的城市场景，而行表示我们的方法与 DrQ 和 PPO 的比较。轨迹分布表明，场景再现变压器中的注意力预测信息为任务完成提供了更多样化的长期策略(轨迹)。

表 4
烧蚀年代 拟议框架的研究

		Proposed	MST	Ego	N-SLT	N	SAC
Left Turn	Succ. %	94	88	78	68	56	68
	Coll. %	4	12	20	24	36	28
Double Merging	Succ. %	96	92	90	80	72	62
	Coll. %	2	4	4	20	25	22
Roundabout-A	Succ. %	88	84	82	74	68	76
	Coll. %	12	16	14	26	32	24
Roundabout-B	Succ. %	82	76	72	70	44	48
	Coll. %	18	24	28	30	56	52
Roundabout-C	Succ. %	76	66	62	66	34	46
	Coll. %	24	34	38	32	64	48

(左转)。此外，邻居基线的差结果表明场景理解更依赖于 $K_{\text{null}}^{\text{null}}$ 在 MST 情况下，由于邻近车辆的本地意图对自我智能体的决策提供了更多的信息。图 14 中注意权重的结果表明，基于多级变压器的结构是具有通过注意自适应信息流理解城市场景中意图和交互能力的关键。

全局预测表示学习的作用在于它对减小状态-动作探索空间 $S \times A$ 的指导。在辅助表征学习目标的基础上，通过 SLT 对潜在表征进行训练 t_{null} 对于决策，将共享以行动为条件的顺序一致特征。它可以发现 q 值的全貌，减小状态作用空间，同时提高政策 π 的抽样效率。因此，驾驶政策的提升体现在 1)更好的量化绩效(表二、表四);以及 2)到达目的地的驾驶手法更加多样化。前者依赖于全局预测引导自我车辆的策略更有效地发现 q 值的有利区域。后者是通过以下事实推断出来的:在接近的时间步长内，可能存在多个驾驶机动的最优轨迹来到达目的地。

表 5
对 SLT 和状态表示的烧蚀评估

Methods	Scenario	Norm L_2	maxQ	minQ
Proposed	left turn	0.756	1.386	0.857
	merge	0.764	1.212	0.787
MST	left turn	0.682	1.235	0.411
	merge	0.619	1.18	0.748
CNN+SLT	left turn	0.661	1.137	-1.361
	merge	0.47	1.174	-0.975
CNN	left turn	0.522	0.742	-1.31
	merge	0.427	1.135	-1.012
Neighbors+SLT	left turn	0.492	0.971	0.246
	merge	0.612	1.198	-0.109
Neighbors	left turn	0.422	0.668	0.236
	merge	0.175	1.193	0.975

为了进一步证明前一种 q 值效率的说法，在勘探空间中，全局预测学习(SLT)后的 q 值应该与普通 MST 更加可分离。因此，给定代表 $S \times A$ 中所有可能探索的双 q 网络的平均连接输入和匹配 q 值的平均值，进行 PCA。PCA 归一化的前两个分量由状态和动作表示为连接输入。为了量化可分性，我们测量了 $L_2^{\text{null}}Q$ 值组(第 25 - 75 百分位)与极值之间的距离。为了进一步研究 SLT 的机制，这里我们比较了 1)CNN 的额外状态表示:通过 CNN 编码的栅格化 BEV 图像;2)邻域:自我车辆和邻域的矢量化轨迹。表 V 清楚地反映了可分性的整体改善(10.8% - 26.6%)，以及在为每个状态表示添加全局预测学习(SLT)后 q 值的提升。图 11 中的定性结果显示，所提出的方法具有更好的分离能力

为了表明后者对全局预测知识的主张，收集了测试过程中成功试验的轨迹，并在图 10 中显示。每个城市场景中的驾驶轨迹证实了驾驶多样性的说法

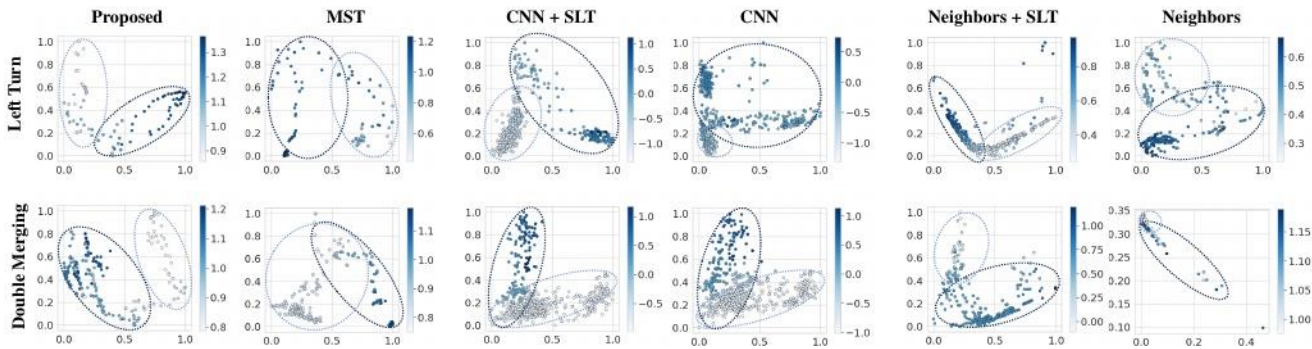


图 11 所示。仅通过场景再现变压器和 MST 对学习到的潜在表征 $S \times A$ 的前两个维度进行主成分分析(PCA)。每个点的颜色条通过双 q 网络与相应的平均 q 值进行映射。

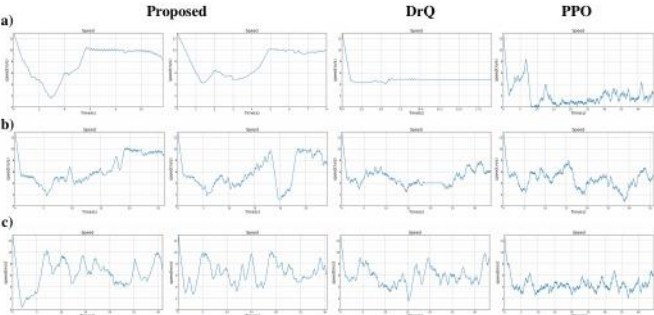


图 12 所示。通过场景再现变压器、DrQ 和 PPO 基线进行成功测试的自我车辆的主要速度模式。测试了三种方法:a)无保护的左转弯;b)双合并;c)Roundabout-B。

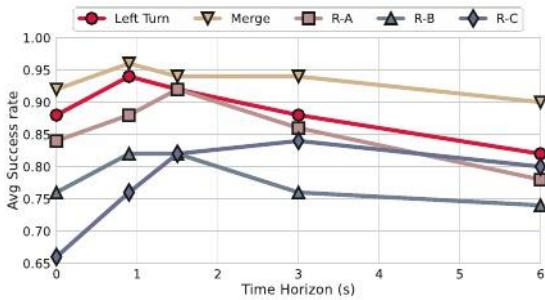


图 13 所示。选择不同 T_G 进行 SLT 训练的比较。峰在较长时间的驾驶任务中，表现会晚一些。

动作。例如，在无保护的左转中，自我车辆不仅学会了左转到内车道并执行变道的“捷径”，而且还学会了直接左转到第二车道。更频繁的变道发生在双合并和环形交叉的任务不同位置。值得注意的是，与 DrQ 相比，驾驶轨迹更接近每条车道的中心，这在某些城市场景中表现出一定的多样性。这种特征可能是由于局部预测的中心路点信息。对于 PPO，先前关于过拟合机动的主张也被 PPO 的单调轨迹分布所验证。

为了研究不同任务长度的合适未来视界设置，在图 13 中，我们比较了不同未来视界 T 的 SLT 训练的每个场景的测试成功率 $G_{null}=(0,3,5,10,20)$ 。结果表明，在较短的未来视界下，改进更快，如果 $T_{G_{null}}$ 太大。表现的巅峰来得较晚，长度较长

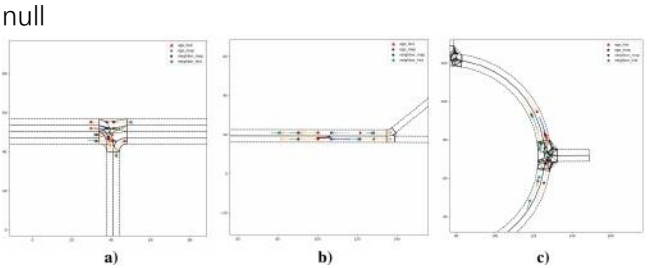


图 14 所示。注意可视化导致 a)无保护的左转, b)双重合并, c)环形交叉。候选路径点和交互的意图信息通过多级变压器的自适应信息流进行调整。

给定任务(R-C)。这是由于额外的冗长和需要频繁更新预测学习过程。对于长期任务，选择更长的 $T_{G_{null}}$ 结果与表二相比，表现更好。

图 12 所示的丰富的车辆动态模式也反映了驾驶机动的多样性。在 Scene-Rep Transformer 中至少可以看到两种重要的模式。例如，在无保护的左转中(图 12a)，ego 车辆已经学习了保守的动态模式，在交叉路口低速轻推，换道，并在换到第二车道后保持恒定的高速。它还学会了在没有变道的情况下采取咄咄逼人的方式。在合并的情况下，自我车辆能够频繁地变道以到达目的地，或者只是进行一次变道并等待前面的车辆，但以相似的时间步长到达目的地。DrQ 和 PPO 基线的表现相似且无能，因为它们无法有效减少探索空间。

六、结论及未来工作

本文提出了 Scene-Rep Transformer，这是一种新颖的基于 Transformer 的表示学习框架，可以提高 RL 决策的样本效率和性能。该框架由两个主要模块组成:用于编码多模态场景状态输入并获取交互式驾驶场景的表示理解的多级 Transformer，以及用于将顺序预测信息提取到当前潜在向量中以指导决策过程的顺序 latent Transformer。软演员-评论家(SAC)决策模块将精炼的潜在表示作为输入并生成决策

输出。所有模块都是端到端的集成和训练，目的是获得更高的奖励和成功率。我们通过五个具有挑战性的模拟城市驾驶场景进行了广泛的验证。定量和定性结果都反映了所提出方法在样本效率、测试性能、可解释性和驾驶机动多样性方面的整体优越性能。我们还研究了框架中不同组件的影响，发现多级 Transformer 能够自适应地感知智能体和地图之间的意图和交互，顺序潜伏 Transformer 可以有效地减少探索空间。

未来的工作将侧重于改进或动态调整全局潜在预测的深度，从而降低长期预测引起的噪声。我们还旨在明确考虑预测的不确定性，并将我们的框架纳入端到端驱动系统。

参考文献。

[1]黄志, 吴建, 吕振, “基于逆向强化学习的驾驶行为建模”, IEEE 智能交通系统学报, 2021。

[2]朱磊, 于福荣, 王勇, 宁必兵, 唐涛, “智能交通系统的大数据分析研究”, 《智能交通系统学报》, 第 20 卷, 第 2 期。1, pp. 383-398, 2018。

[3]刘志强, 刘志强, “基于自动驾驶技术的自动驾驶系统研究”, 《IEEE》, 第 8 卷, 第 1-4 页, 2020。

[4]陈志强, 陈志强, 陈志强, “基于深度强化学习的自动驾驶系统研究”, 智能交通系统学报, 2012。

[5]黄志, 吴建, 吕春, “基于专家先验的深度强化学习方法”, IEEE 神经网络与学习系统学报, 2022。

[6]吴健, 黄志, 黄伟, 吕春, “基于人工引导的经验强化学习方法及其在自动驾驶中的应用”, arXiv 预印本 arXiv:2109.12516, 2021。

[7]陈志强, “基于深度强化学习的自动驾驶汽车运动规划”, 《智能交通系统学报》, 2020。

[8]陈建军, 元 b, M. Tomizuka, “无模型深度强化学习在城市自动驾驶中的应用”, 2019 IEEE 智能交通系统会议(ITSC)。IEEE, 2019, pp. 2765-2771。

[9]黄志, 吕超, 邢昉, 吴健, “基于多模态传感器融合的深度神经网络的端到端自动驾驶”, IEEE 传感器学报, vol. 21, no. 11。10, pp. 11781 - 11790, 2020。

[10]陈建军, 元 b, M. Tomizuka, “基于深度模仿学习的城市自动驾驶”, 2019 IEEE/RSJ 智能机器人与系统国际会议(IROS)。IEEE, 2019, pp. 2884-2890。

[11]吴建军, 黄之, 黄之, 胡志明, 邢宇, 吕超, “人在环深度强化学习在自动驾驶中的应用”, arXiv 预印本 arXiv:2104.07246, 2021。

[12]刘勇, 黄之, 吕超, “基于专家演示的深度强化学习方法”, arXiv 预印本 arXiv:2102.09243, 2021。

[13] O. psamez - gil, R. bararea, E. López-Guillén, L. M. Bergasa, C. Gomez-Huelamo, R. gutisamez 和 A. Diaz-Diaz, “基于深度强化学习的自动驾驶车辆控制”, 多媒体工具与应用, 第 81 卷, 第 81 期。3, pp. 3553-3576, 2022。

[14]王晓明, 王晓明, 王晓明, 王晓明, Zöllner, “基于深度 q 网络的智能驾驶仿真研究”, 《IEEE 智能汽车研讨会》, 2017, 第 4 期, 第 244-250 页。

[15]. feh<s:1>, S. Aradi, F. heged<e:1>, T. b<s:1>, P. Gáspár, “混合 ddpq 方法在车辆运动规划中的应用”, 2019。

null[16]段军, 李, 关勇, 孙强, 程斌, “基于标签化驾驶数据的自动驾驶决策的分层强化学习”, 智能交通系统, vol. 14, no. 16。5, pp. 297-305, 2020。

[17]李晓明, 李, “强化学习是一个大的序列建模问题”, 中国科学技术论坛, 2012。

[18]陈丽娟, 陆凯, 李凯, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, I. Mordatch, “决策转换:基于序列模型的强化学习”, 神经信息处理系统进展, vol. 34, pp. 15 084-15 097, 2021。

[19]张晓明, 张晓明, 李晓明, 张晓明, 张等, “自动驾驶汽车的端到端学习”, arXiv, 第 4 期, 2016。

[20]李晓明, 李晓明, 李晓明, “基于智能驾驶的城市自动驾驶仿真系统”, 北京交通大学学报(自然科学版)。PMLR, 2017, pp. 1-16。

[21]张志强, 张志强, “一种基于图像识别的图像识别方法”, IEEE/CVF 计算机视觉与模式识别学术会议论文集, 2013,pp. 391 - 391。

[22]黄之, 莫晓霞, 吕志明, “基于神经网络的自动驾驶多模态运动预测”, arXiv, 预印本 arXiv: 109.06446, 2021。

[23]高建军, 孙晨, 赵红华, 沈勇, 李, “基于矢量表示的智能体动态图像编码”, IEEE/CVF 计算机视觉与模式识别会议论文集, 2020,pp. 11 525-11 533。

[24]杨建军, 张志强, 张志强, 蒋洪涛, 林俊杰, 刘勇, 张志强等, “场景变换:一种统一的多任务行为预测和规划模型”, arXiv, 第 1 卷第 1 期, 2012。

[25]王晓明, 王晓明, 王晓明, “面向未来运动估计的图形化热图输出”, 国际机器人与自动化会议(ICRA), 2012。IEEE, 2022, pp. 9107-9114。

[26]王晓明, 张晓明, “基于自动驾驶的自动驾驶地图系统”, 《IEEE 智能汽车研讨会》, 2014。IEEE, 2014, pp. 420-425。

[27]张晓明, 张晓明, 张晓明, “基于动态交互感知场景的自动驾驶强化学习”, IEEE 国际机器人与自动化会议(ICRA)。IEEE, 2020, pp. 4329-4335。

[28]蔡鹏, 王红华, 孙艳, 刘勇, “基于深度 q 学习和图关注网络的自动驾驶系统”, arXiv 预印本 arXiv: 108.05030, 2021。

[29]刘勇, 张, 方磊, 姜清, 周斌, “基于堆叠变压器的多模态运动预测”, IEEE/CVF 计算机视觉与模式识别会议论文集, 2021,pp. 7577-7586。

[30]张晓明, 张晓明, 王晓明, “基于深度学习的自动驾驶学习模型研究”, arXiv, 2004,12(4):444 - 444。

[31]肖宇, F. Codevilla, C. Pal, A. M. López, “基于行为的自动驾驶表征学习”, arXiv 预印本 arXiv:2008.09417, 2020。

[32]王志强. Grill, F. Strub, F. altch, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar 等, “引导你自己的潜在——一种自我监督学习的新方法”, 《神经信息处理系统进展》, 第 33 卷, 第 21 271-21 284 页, 2020。

[33]张晓明, 张晓明, 张晓明, “基于深度强化学习的图像增强算法”, 国际会议, 2016。

[34]张晓明, 张晓明, 张晓明, “基于非监督学习的深度学习研究方法”, 《arXiv》, 第 4 期, 2012。

[35]王勇, 罗毅, 刘建军, 陈, 李同, “基于自编码的自动驾驶系统”, IEEE 智能交通系统学报, vol. 23, no. 23。1, pp. 641-650, 2020。

[36]陈, 洪亮, 徐辉, 李同, 邓永元。杨, “Multisiam:基于 au-的自监督多实例连体表示学习”

自动驾驶”，《IEEE/CVF 计算机视觉国际会议会议录》，2021 年，第 7546-7554 页。

[37]张晓明，张晓明，张晓明，“基于自预测的深度学习研究方法”，《arXiv》，2012 年第 4 期。

[38]周明，G. Tucker，陈杰，Kumar，朱，A. Gupta，P. Abbeel 等，“软演员评价算法及其应用”，arXiv 预印本 arXiv:1812.05905, 2018。

[39]陈，何凯，“简单连体表示学习的探索”，IEEE/CVF 计算机视觉与模式识别会议论文集，2021,pp. 15 - 15 758。

[40]张晓明，李晓明，张晓明，“基于深度学习的深度学习模型研究”，计算机科学与技术，2011。PMLR，2019,pp. 2170-2179。

[41]张晓明，李晓明，李晓明，“深度变化信息瓶颈”，《arXiv》，2016 年第 1 期。

[42]周，罗毅，J. Villella，杨勇，D. Rusu，苗军，张，M. Alban，I. Fadakar，陈等，“面向自动驾驶的可扩展多智能体强化学习训练学校”，arXiv 预印 arXiv:2010.09776, 2020。

[43]王勇，王勇，张东，杨勇，熊仁仁，“自动驾驶的学习分层行为和运动规划”，2020 年 IEEE/RSJ 智能机器人与系统国际会议(IROS)。IEEE, 2020, pp. 2235-2242。

[44]张晓明，张晓明，张晓明，等。基于最小二乘算法的优化算法[j]. arXiv，2017,27(4):557 - 557。

[45]朱，李同，何德东，王勇，孟其华。孟，“深度强化学习监督办公环境中的自主探索”，2018 年 IEEE 机器人与自动化国际会议(ICRA)。IEEE, 2018, pp. 7548-7555。

[46]张晓明，张晓明，“信息瓶颈方法”，《arXiv》，2004 年第 4 期。

[47]冈田明，张志刚，“基于自监督学习的不确定性学习方法”，arXiv 预印本 arXiv: 357 - 357, 2012。

[48]张晓明，张晓明，张晓明，“基于深度学习的学习行为研究”，《arXiv》，第 4 期，第 4 - 12 页，2019。

[49]张晓明，张晓明，张晓明，“基于多变量的机器学习方法研究”，计算机科学与技术，2013。PMLR，2019,pp. 5171-5180。

[50]冈田和谷口明，“基于模型的强化学习:基于潜在想象的非重构”，机器人与自动化国际会议(ei)。IEEE, 2021, pp. 4209-4215。

附录

定理 1 (SLT 等价于信息瓶颈)。提出的具有 *Sim-Siam*[45]表示学习目标的 *SLT* 管道等于潜在动态模型的**未来状态信息瓶颈目标**[46]:

$$J : \max [I(h_{1:T}, s_{1:T}|a_{1:T}) - \beta I(h_{1:T}, s_{1:T}^i|a_{1:T})] \quad (19)$$

式中 $T = T_{\text{null}}$ 代表未来的地平线。 $s \in \mathcal{S}$, $a \in \mathcal{A}$, $h \in \mathcal{D}$ 表示驾驶场景的状态、动作和表示。 I 为批量采样指数[41]。

证明。根据 P_{null} 引理 1，我们可以推出目标下界: $J \geq L_{\text{NCE}} - \beta L_{\text{KL}}$ 。

从引理 3，我们证明了 L 的有效性 $L_{\text{NCE}}^{\text{null}}$ 通过 *SLT*。然后，作为 L 的等价 $L_{\text{NCE}}^{\text{null}}$ 在引理 2 中证明了 *Sim-Siam* 管道， $L_{\text{KL}}^{\text{null}}$ null =常数。[47]由于 h 在我们的表示学习管道中是确定性的，因此定理得到了证明。

引理 1(信息瓶颈的下界)。跨未来时间步的信息瓶颈的建议目标可以由[48]来界定:

$$J \geq \sum_t (\mathcal{L}_t^{\text{NCE}} + \beta \mathcal{L}_t^{\text{KL}}) \quad (20)$$

其中 $L_{\text{NCE}t}$ 表示批处理的 *InfoNCE* 目标[49]:

$$\mathcal{L}_t^{\text{NCE}} = \mathbb{E}_{h \sim \Phi} \left[\log p(h_t | s_t) - \log \sum_i p(h_t | s_t^i) \right] \quad (21)$$

Φ 是提出的 MST, $L_{\text{KL}t}$ 为 KL-divergence:

$$\text{KL}(q(h_t | h_{t-1}, a_{t-1}, s_t) \| p(h_t | h_{t-1}, a_{t-1})) \quad (22)$$

证明。考虑信息增益的第一项:

$$\begin{aligned} I(h_{1:T}; s_{1:T} | a_{1:T}) &= \mathbb{E} \left[\sum_t \log p(s_{1:T} | h_{1:T}, a_{1:T}) - \log p(s_{1:T}, | a_{1:T}) \right] \\ &\geq \mathbb{E} \left[\sum_t \log p(s_{1:T} | h_{1:T}, a_{1:T}) \right] - \\ &\quad \text{KL} \left(p(s_{1:T} | h_{1:T}, a_{1:T}) \| \prod_t p(s_t | h_t) \right) \\ &= \mathbb{E}_{h \sim \Phi} \sum_t \log p(s_t | h_t) \end{aligned} \quad (23)$$

由于独立性，最后一项导出为 $\log p(s_{1:T} | a_{1:T}) = 0$ 。通过使用贝叶斯规则和批处理 *InfoNCE*，未来每一步的预期期限变成:

$$\mathbb{E}_{h \sim \Phi} \log p(s_t | h_t) \geq \mathcal{L}_t^{\text{NCE}} \quad (24)$$

考虑信息增益的第二项:

$$\begin{aligned} I(h_{1:T}; s_{1:T}^i | a_{1:T}) &= \mathbb{E} \left[\sum_t \log p(h_t | h_{t-1}, a_{t-1}, s_t) - \log p(h_t | h_{t-1}, a_{t-1}) \right] \\ &\leq \mathbb{E} \left[\sum_t \log q(h_t | h_{t-1}, a_{t-1}, s_t) - \log p(h_t | h_{t-1}, a_{t-1}) \right] \\ &= \sum_t \text{KL}(q(h_t | h_{t-1}, a_{t-1}, s_t) \| p(h_t | h_{t-1}, a_{t-1})) \end{aligned} \quad (25)$$

表示最后一项为 $P_{\text{null}t} L_{\text{KL}t}$ ，提议的下界是证明。

引理 2 (InfoNCE 与 Sim-Siam 的等价)。模拟-模拟表示学习管道在式 *中最大化 L_{NCEt}* 隐式。

证明。表示式 21 中的项:

$$\begin{aligned} \mathbb{E}_{h \sim \Phi} \log p(h_t | s_t) &= \mathcal{L}^1 \\ \mathbb{E}_{h \sim \Phi} \log \sum_i p(h_t | s_t^i) &= \mathcal{L}^2 \end{aligned} \quad (26)$$

由[47]所证明的定理，我们可以得出 $L^1_{\text{null}} \approx L^1_{\text{glb null}}$ 在式 17 中， P 在 *Sim-Siam* 中的存在隐含地使 L 最小化 L^2_{null} 。因此，引理被证明。

引理 3 (SLT 到 InfoNCE 的有效性)。通过 $SLT\ T$ 重新表述式 17 中的 h 是有效的, 因为未来表示 $h_{1:T}$ 不能通过 $\Phi_{without}$ 知道基真未来状态直接采样。

证明。重新表述 $L^{NCE}_{t_{null}}$ 使用 SLT 过渡模型 T , 对于每个时间步 $T \in [1, T]_{f_{null}}$, 我们总能得到多步预测 $h_{t_{null}} \leftarrow p(h_{t_{null}} | h_{<t_{null}}, \text{一个}_{<t_{null}})$ 源自 SLT T 。批处理预测公式为 $h \sim T E p(ht|h < T, a < T)$ 。然后表示为 $h_{t_{null}}$ 每个未来时间步长的表示 h 可以分配给 $L^{NCE}_{t_{null}}$ 用[50]中证明的重要抽样方法进行 SLT T 。因此, 引理被证明。

路点生成: $K_{t_{null}}$ 需要图结构高保真(HD)地图 GHD[26]和当前状态 $Mego0, M1:n0$ 。这些信息在真实情况下由感知模块进行预处理。我们可以通过以下算法在线生成自我和邻居的路点:

表六世
官方智能场景的设置(默认)

Scenario Name	Left Turn	Double Merge	Roundabout
Max Timesteps	400	400	400/600/800
Flows (/route)	20	40	20
Vehicles (/flow)	4	6	4
Imperfection	$\mathcal{N}(U[0.3, 0.7], 0.1)$		
Impatience	$U[0, 1]$		
Cooperative	$U[0, 1]$		

表 vii 实验中使用的参数

Notation	Meaning	Value
n	Max number of neighboring vehicles	5
V_{max}	Speed limit (m/s)	10
T_h	Historical horizon	10
T_K	Route waypoints horizon	10
T_G	Global predictive horizon	3
N_k	Number of candidate route waypoints	2
γ	Discount rate	0.99
λ	Polyak averaging weight	0.005
α	Initial entropy weight	1
N_{init}	Warm-up random searching step	5000
N_{buffer}	Replay buffer capacity	20000
N_B	Batch size	32
N_{train}	Training steps	100000