小 型 微 型 计 算 机 系 统
Journal of Chinese Computer Systems
DOI:10.20009/j.cnki.21-1106/TP.2023-0463
2025 年 1 月 第 1 期
Vol.46 No.1 2025

基于 GAT 和 Transformer 的车辆行为预测

王 昀,蔡 英,范艳芳,柳军杰,张 哲

(北京信息科技大学 计算机学院,北京 100101)
E-mail:ycai@bistu.edu.cn

摘 要: 车辆行为预测可以辅助自动驾驶系统进行决策,提高自动驾驶的安全性和效率.然而在不同道路场景下,周围交通参与者(如汽车、自行车及行人等)之间动态的变化会导致预测车辆位置信息存在较大误差,这可能使自动驾驶车辆无法及时采取避让或紧急制动等措施.本文旨在针对结构化和非结构化道路场景,构建交通参与者之间的动态交互时空图,并运用深度学习技术设计一种基于 GAT 和 Transformer 的 GAN 模型,用于车辆行为预测.GAT 被用于学习不同参与者之间的相关性和交互规律,而 Transformer 被用来提取交通参与者运动状态信息的时序特征.分别在 NGSIM 和 ApolloScape 数据集进行仿真实验.结果表明,本文模型在长时域的预测表现出更高的精度,同时还具备更轻量级的特点.

关 键 词: 自动驾驶;车辆行为预测;深度学习;图注意力网络
中图分类号: TP183 **文献标识码:** A **文 章 编 号:** 1000-1220(2025)01-0023-10

Vehicle Behavior Prediction Based on GAT and Transformers

WANG Yun, CAI Ying, FAN Yanfang, LIU Junjie, ZHANG Zhe
(College of Computer, Beijing Information Science & Technology University, Beijing 100101, China)

Abstract: Vehicle behavior prediction can assist autonomous driving systems in making decisions, thereby enhancing the safety and efficiency of autonomous driving. However, in different road scenarios, dynamic changes among surrounding traffic participants, such as cars, bicycles, and pedestrians, can lead to significant errors in predicting vehicle position information. This may prevent autonomous vehicles from taking timely evasive or emergency braking actions. This paper aims to construct dynamic spatiotemporal graphs of interactions among traffic participants for structured and unstructured road scenarios. It utilizes deep learning techniques to design a GAN model based on GAT(Graph Attention Network) and Transformer for vehicle behavior prediction. GAT is employed to learn the correlations and interaction patterns among different participants, while Transformer is used to extract temporal features of traffic participants' motion states. Simulation experiments are conducted on the NGSIM and ApolloScape datasets. The results demonstrate that our model exhibits higher accuracy in long-term predictions while maintaining a more lightweight profile.

Keywords: autonomous driving; vehicle behavior prediction; deep learning; graph attention network

0 引 言

随着自动驾驶技术的快速发展,道路安全和出行效率迎来了全新的机遇和挑战,而其中一个挑战就是如何精准预测车辆的行为,因为对周围车辆驾驶行为的精准预测能让自动驾驶系统做出更可靠的驾驶策略,从而进一步提高驾驶安全,减少交通事故的产生,提高出行的效率和舒适性.

然而,由于不同道路场景下车辆周围的交通参与者(如汽车、自行车及行人等)的动态变化以及各自对车辆的影响程度不同,导致现有的车辆行为预测方法存在着较大的位置信息误差,这可能会导致自动驾驶汽车在执行制定的行驶路径时无法采取必要的避让或者紧急制动措施.此外,车辆本身有限的存储和计算资源需要采用轻量级的模型去实现较高精度的预测.

在这一背景下,本文旨在解决车辆行为预测中的上述挑战,特别是针对结构化和非结构化道路场景.本文提出了一种新颖的方法,通过构建动态交互时空图,运用深度学习技术,设计了一种基于 GAT(Graph Attention Network)和 Transformer 的 GAN(Generative Adversarial Network)模型,以提高车辆行为预测的精度和轻量级性能.具体而言,利用 GAT 来学习不同交通参与者之间的相关性和交互规律,从而更准确地捕捉周围交通环境的动态变化.与此同时,Transformer 被应用于提取交通参与者运动状态信息的时序特征,进一步提高了预测性能.实验表明,本文提出的模型在长时域的预测中表现出更高的精度,同时还具备更轻量级的特点.

1 相关工作

目前,车辆行为预测按最终的输出结果大致分为两种:输出车辆可能的行为模式(例如左转、右转、直行等);以及输出

收稿日期:2023-09-27 收修改稿日期:2023-11-17 基金项目:北京市自然科学基金-海淀原始创新联合基金项目(L192023)资助. 作者简介:**王 昀**,男,1997 年生,硕士研究生,研究方向为车辆行为预测、深度学习、自动驾驶;**蔡 英**,女,1966 年生,博士,教授,CCF 会员,研究方向为车联网、边缘计算、网络安全及密码学算法;**范艳芳**,女,1979 年生,博士,副教授,研究方向为信息安全、车联网和边缘计算;**柳军杰**,男,2000 年生,硕士研究生,研究方向为行人轨迹预测、深度学习;**张 哲**,男,2000 年生,硕士研究生,研究方向为区块链、数据共享.

车辆在未来一段时间位置信息(即车辆轨迹).这两种输出结果在本质上具有同一性[1].

根据车辆在驾驶环境受限制的因素可将车辆行为预测模型分为三类,基于物理的模型,基于意图的模型和基于交互的模型.

基于物理的模型,将车辆视作一个仅受物理定律约束的实体,考虑了车辆在运动过程中受到的物理限制和力的影响,进而预测车辆未来的驾驶意图或者位置信息.Qian 等[2]提出一种基于自回归移动平均模型,将车辆前一时刻的位置信息和加速度作为模型输入,然后预测下一时刻车辆的位置信息和速度.这类模型只适用于预测时域较短(小于 1 秒)的预测.

基于意图的模型会事先对车辆的运动进行建模,将车辆可能的运动模式划分为一系列的簇,而每一个簇对应着车辆一个典型的运动模式(例如直行、左转、右转等),在预测过程中通过推导出与识别意图相对应的输入从而生成车辆的未来的运动信息.其中 Deo 等[3]提出了一种高斯混合模型使其以概率的方式探索车辆意图的潜在行驶空间,并为车辆每个可能意图生成相应的样本轨迹.这类模型预测时域为 2 ~3 秒的任务中展现了良好的性能,但不能考虑当前车辆状态的不确定性,且将车辆视作独立的个体,没有考虑到车辆之间的相互影响,泛化能力差.

基于交互的模型中,车辆被视为受其他交通参与者(车辆、行人等)影响的机动实体,即假设其他交通参与者对车辆运动产生影响.由于该类模型比物理模型预测的时间窗口更长,比基于意图的模型更加稳定,现已成为车辆行为预测中对车辆建模的主流方法.考虑车辆之间的交互影响需要大量的数据驱动,而基于深度学习的方法能够挖掘出数据的内在规律从而达到不错的预测精度.Shi 等[4]提出了一种基于双向 LSTM(Long Short Term Memory,LSTM)和时间卷积神经网络的预测模型,通过研究车辆不同行为(例如:跟随、变道等)间的依赖性来提升预测的精度,甚至可以预测组合行为,但该模型只针对自身车辆行为的内在依赖关系而忽略了周围车辆对其的外在影响,在长时域的预测中精度较低.高振海等[5]提出了基于单双向长短时记忆的模型(Monodirectional and Bidirectional LSTM,MB-LSTM)用于车辆位置信息的预测.该模型结合了车辆环境的上下文信息和车辆行为意图信息,以提高预测精度.然而,在复杂道路场景中,考虑到交通参与者类型较为单一且缺乏对参与者间动态交互变化的考虑,该模型的预测精度较低,泛化能力也较差.Fang 等[6]提出了一种轨迹建议网络(Trajectory Proposal Network,TPNet)用于预测车辆的轨迹信息,先生成一组候选的未来轨迹作为假设,再通过对满足物理约束(交通规则和可移动区域)的候选轨迹进行筛选来进行最终预测,但只能针对固定道路环境建模,泛化能力差.Choi 等[7]提出了一种基于随机森林(RF)算法和 LSTM 编码器-解码器架构的轨迹预测方法,先为目标车辆周围的区域定义占用网格图,通过空间关系来建立车辆间交互,最后预测出目标车辆在未来时间步将占用的行和列,但是仅靠空间关系难以建立车辆间长期交互影响,在长时域预测中精度较低.An 等[8]通过充分利用多车辆间的时空相关性,提出了一种基于图卷积网络(GCN)和交互感知网络的模型 DGInet 用于预测车辆的轨迹信息,但将周围车辆的交互影响同等对待,没有量化出周围不同车辆的重要程度从而会损失部分精度.Gao 等[9]提出了一种层次图神经网络 VectorNet,将包含交通参与者和道路上下文信息(红绿灯等)的高清图像进行矢量化并提取相互之间的交互信息作为网络的输入来预测车辆的轨迹信息,但考虑过多无用交互信息(例如:车道与红绿灯之间)以及缺乏对历史时间信息重要程度的量化导致在长时域的预测中计算开销大和预测精度低的问题.Li 等[10]将 GCN 和 LSTM 结合,提出了 GRIP 模型用以预测车辆位置信息,使用固定图来表示密切车辆的交互,但是缺乏对车辆高速移动性以及车辆间动态交互变化的考虑,容易导致预测误差增大.Lin 等[11]提出了一种基于时空间注意力机制的 LSTM 模型用于研究周围车辆对目标车辆的影响,但没有考虑不同类型交通参与者对目标车辆的影响,泛化能力较差.Li[12]等设计了一种基于五辆车的聚类结构,通过提取车辆间碰撞减速率和车道间距等特征作为输入采用基于自注意力机制的方案用于预测车辆间的交互行为,但只针对固定对象进行交互,没有考虑车辆交互的动态变化.

综上所述,基于物理和意图的模型预测精度低,泛化能力差.而现有基于交互的模型中:1)对道路结构化程度不同可能带来的交互实体变化欠缺考虑;2)大部分模型仅考虑固定对象间的交互,未能有效理解驾驶场景中动态变化的特性,不适用于交互频繁变化的复杂场景;3)平等对待目标车辆周围的交通参与者,忽视了其在预测中的重要作用,容易导致预测的精度低的问题.为了解决上述问题,本文首先将道路场景划分为结构化道路和非结构化道路,根据道路场景中不同的交互影响来构建动态交互无向图,运用图注意力网络(Graph Attention Networks,GAT)为相邻交通参与者分配不同的权重并提取空间维度的语义信息,之后通过 Transformer 网络在提取时间维度的语义信息的同时建立历史信息与预测信息的长依赖关系.因此本文针对车辆行为预测任务设计了一种基于 GAT 和 Transformer 的 GAN 模型.

2 目标车辆的行为预测框架

本文针对车辆行为预测任务设计相应的模型,首先通过根据道路结构化程度将道路情况划分为结构化道路和非结构化道路,根据不同道路场景构建目标车辆与周围交通参与者的动态交互无向图.随后运用图注意力网络和自注意力网络理论,设计基于 GAT 和 Transformer 的 GAN 模型,模型架构如图 1 所示.该模型主要包含生成器和判别器两个模块.其中生成器模块会先将目标车辆与其他交通参与者间的运动状态表示为时空图的形式,而后运用 GAT 图神经网络理论针对周围交通参与者对目标车辆影响程度以及目标车辆自身状态信息进行推理,从空间维度上捕获影响目标车辆的重要信息并作为 Transformer 编码器的输入,Transformer 编码器能从时间维度上推理出目标车辆及其周围交通参与者运动状态的变化,从而预测出目标车辆的未来的位置信息.判别器模块主要由 Transformer 网络编码器组成,该编码器接收目标车辆的真实位置信息和生成器预测的位置信息作为输入.它解析位置

信息的数据样本,并输出一个概率值,用于表示输入信息是真　　实数据样本的概率.

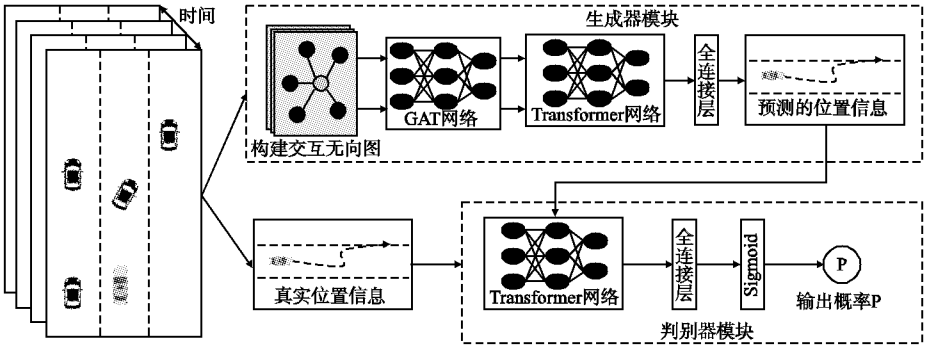


图 1　基于 GAT 和 Transformer 的目标车辆行为预测框架

Fig. 1　Framework for predicting target vehicle behavior based on GAT and Transformers

2.1　目标车辆与周围交通参与者的交互关联关系

结构化道路具有清晰的道路标识线,道路背景环境较为单一,例如城市主干道、高速公路等,如图 2 所示. 这类道路场

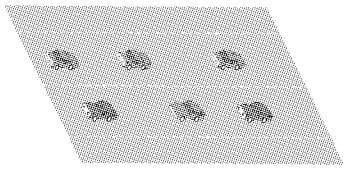


图 2　结构化道路场景图

Fig. 2　Structured road scene map

景下车辆与车辆的交互比较频繁,车辆与其他交通参与者(行人、自行车等)较少,而清晰明显的道路标识能让车辆的驾驶行为相对规范,因此针对结构化道路场景,本文假设目标车辆受到其受其左侧、右侧、前方和后方的车辆的直接交互影响的直接交互影响,如图 3 所示.

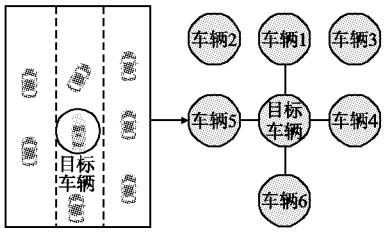


图 3　结构化道路中的交互影响

Fig. 3　Interactive impact in structured roads

非结构化道路通常缺乏车道线和清晰的道路边界,道路环境较为复杂,例如城市的非主干道、乡村街道等,如图 4 所示. 在这种情况下,由于缺乏明确的道路标识作为参考,模型不仅需要考虑车辆之间的交互影响,还需综合考虑其他交通参与者对车辆行为的影响. 因此针对非结构化道路场景,本文提出了以下假设:目标车辆受到一定范围内车辆和其他交通参与者的交互影响. 鉴于车辆的高速移动性和其他交通参与者的低速移动性,本文在考虑交互影响时采取了不同的范围选 择策略如图5所示,对于低速移动的行人、自行车等参与者仅 考虑10米范围内的交互影响,而对于车辆则考虑40米范围内的交互影响.

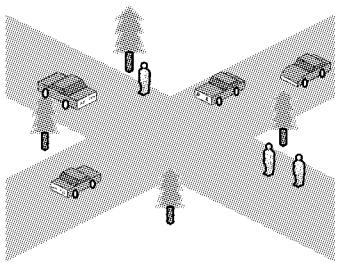


图 4　非结构化道路场景图

Fig. 4　Unstructured road scene map

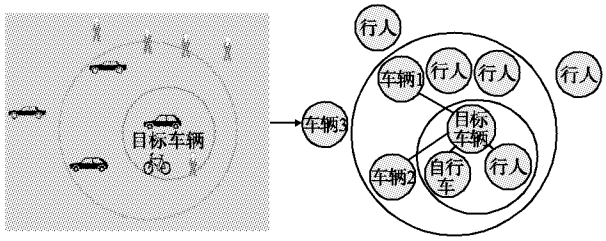


图 5　非结构化道路中的交互影响

Fig. 5　Interactive impact in unstructured roads

2.2　构建道路场景中的动态交互时空图

结合上述的交通场景,建立描述结构化道路和非结构化道路场景下交通参与者间交互关系的时空图. 如图 6 所示,将

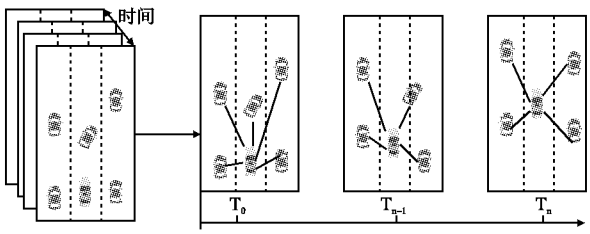


图 6　交通参与者动态交互时空图

Fig. 6　Dynamic interaction spatiotemporal map of traffic participantss

目标车辆以及其他交通参与者作为图的“节点”,然后根据同一时刻各交通参与者的空间位置确定交互影响作为图的“边”.

在考虑交通参与者间的双向交互影响下将其动态交互性描述为无向图网络结构 $G=\{g_0,g_1,\cdots,g_{T-1},g_T\}$. 其中 g_T 表示 T 时刻构建的无向图. 而 $g_T=(V^T,E^T)$,其中 $V^T=\{S_0^T,S_1^T,\cdots,S_n^T\}$ 为交通参与者组成的节点集合,n 为时刻道路场景存在的交通参与者数,即“节点”数;$E^T=\{e_{S_0^TS_0^T},e_{S_0^TS_2^T},\cdots,e_{S_{n-1}^TS_n^T}\}$ 为交通参与者间交互关系组成的“边”集合,$e_{S_0^TS_1^T}$ 表示交通参与者 S_0 和 S_1 在 T 时刻存在交互关系的“边”. 随后构建“节点”的特征矩阵的集合 H 和“边”的邻接矩阵的集合 A 并将其作为图注意力机制网络的输入,而后得到空间维度上交通参与者间的交互规律信息 H_G 作为图注意力机制网络的输出. 其中特征矩阵的集合 $H=\{H_0,H_1,\cdots,H_{T-1},H_T\}$ 是不同时刻无向图的特征矩阵集合,特征矩阵 H_T 包含了 T 时刻中每个交通参与者的位置信息 (x^T,y^T),横向和纵向的瞬时速度 (vx^T,vy^T) 以及类型 c,如公式(1)所示:

$$\boldsymbol{H_T}=\begin{bmatrix}\boldsymbol{h_{S_0}}^T\\ \boldsymbol{h_{S_1}}^T\\ \cdots\\ \boldsymbol{h_{S_{n-1}}}^T\\ \boldsymbol{h_{S_n}}^T\end{bmatrix}=\begin{bmatrix}\boldsymbol{x_{S_0}}^T,\boldsymbol{y_{S_0}}^T,\boldsymbol{vx_{S_0}}^T,\boldsymbol{vy_{S_0}}^T,\boldsymbol{c_{S_0}}\\ \boldsymbol{x_{S_1}}^T,\boldsymbol{y_{S_1}}^T,\boldsymbol{vx_{S_1}}^T,\boldsymbol{vy_{S_1}}^T,\boldsymbol{c_{S_1}}\\ \cdots\\ \boldsymbol{x_{S_{n-1}}}^T,\boldsymbol{y_{S_{n-1}}}^T,\boldsymbol{vx_{S_{n-1}}}^T,\boldsymbol{vy_{S_{n-1}}}^T,\boldsymbol{c_{S_{n-1}}}\\ \boldsymbol{x_{S_n}}^T,\boldsymbol{y_{S_n}}^T,\boldsymbol{vx_{S_n}}^T,\boldsymbol{vy_{S_n}}^T,\boldsymbol{c_{S_n}}\end{bmatrix}\tag{1}$$

邻接矩阵的集合 $A=\{A_0,A_1,\cdots,A_T\}$,邻接矩阵 A_T 包含了 T 时刻交通参与者间交互的关联关系,如公式(2)所示:

$$\boldsymbol{A_T}=\begin{bmatrix}\boldsymbol{a_{S_0S_0}}^T,\boldsymbol{a_{S_0S_1}}^T,\cdots,\boldsymbol{a_{S_0S_{n-1}}}^T,\boldsymbol{a_{S_0S_n}}^T\\ \boldsymbol{a_{S_1S_0}}^T,\boldsymbol{a_{S_1S_1}}^T,\cdots,\boldsymbol{a_{S_1S_{n-1}}}^T,\boldsymbol{a_{S_1S_n}}^T\\ \cdots\\ \boldsymbol{a_{S_{n-1}S_0}}^T,\boldsymbol{a_{S_{n-1}S_1}}^T,\cdots,\boldsymbol{a_{S_{n-1}S_{n-1}}}^T,\boldsymbol{a_{S_{n-1}S_n}}^T\\ \boldsymbol{a_{S_nS_0}}^T,\boldsymbol{a_{S_nS_1}}^T,\cdots,\boldsymbol{a_{S_nS_{n-1}}}^T,\boldsymbol{a_{S_nS_n}}^T\end{bmatrix}\tag{2}$$

而 $a_{S_iS_j}^T$ 如公式(3)所示:

$$\boldsymbol{a_{S_iS_j}}^T=\begin{cases}1,(\boldsymbol{S_i}\text{ 和 }\boldsymbol{S_j}\text{ 满足交互关联关系})\\ 0,\text{其他}\end{cases}\tag{3}$$

2.3 基于 GAT 和 Transformer 网络的生成器

生成器模块如图 7 所示.

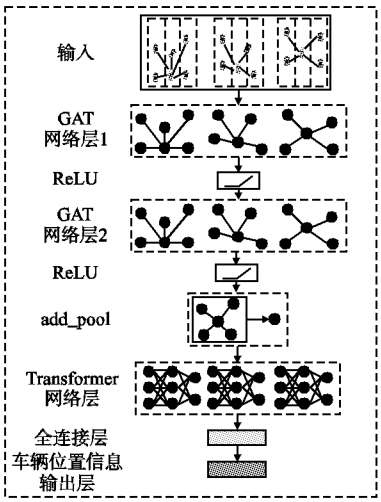


图 7 生成器模块

Fig. 7 Generator module

交通参与者动态交互时空图的特征矩阵集合 H 和邻接矩阵集合 A 作为 GAT 图神经网络的输入,经过 2 层 GAT 图神经网络的深度学习,得到交通参与者间的交互规律信息 H_G,而后利用 add_pool 图池化操作将交互规律信息进行求和操作汇聚成图级别的特征信息 $g_G=\{g_G^0,g_G^1,\cdots,g_G^T\}$,不仅能够获取每张无向图的全局信息,还可以使得具有相同的维度大小,如公式(4)所示:

$$\boldsymbol{g_G^i}=\sum\boldsymbol{h_G^i}\tag{4}$$

$h_G^i\in H_G$ 表示第 i 张图的“节点”特征矩阵. g_G 虽然包含了空间维度上的全局交互信息,但缺乏时间维度上的关联信息,因为不同时刻图之间存在前后关联关系,而此前的 GAT 网络层和 add_pool 操作并没有提取到该类信息. 运用 Transformer 神经网络对 g_G 中构建图信息间的时序关系,同时还能够使得模型能够在不同位置上建立输入信息不同程度的依赖关系,从而更好地在时间维度上捕获预测信息同历史信息的长距离关联关系. Transformer 网络由位置编码、多头注意力机制(Multi-Head Attention)、Add&Norm 和前馈神经网络层(Feed-Forward Networks,FFN)构成.

通过对 g_G 进行基于正、余弦函数的位置编码为其添加时序的前后关系. 假设 $g_G\in\mathbb{R}^{n\times d}$,$n$ 为图的数量,d 为特征维度,采用具有相同形状的矩阵 P 与 g_G 相加之后得到具有位置信息的输出 g_G',而 P 的计算表现形式如公式(5)所示:

$$\begin{cases}P_{i,2j}=sin\left(\dfrac{i}{10000^{\frac{2j}{d}}}\right),i\in(0,n),j\in(0,\lfloor d/2\rfloor)\\ P_{i,2j+1}=cos\left(\dfrac{i}{10000^{\frac{2j}{d}}}\right),i\in(0,n),j\in(0,\lfloor d/2\rfloor)\end{cases}\tag{5}$$

将 g_G' 作为多头注意力机制的输入捕获内部的多种依赖关系并拼接得到输出 $MultiHeadAttention^N(g_G')$,本质上是采用 N 个不同的查询 Q、键 K 和值 V 并行计算出对应的注意力分数后进行拼接组合,如公式(6)所示:

$$MultiHeadAttention^N(g_G')=W^A\cdot CONCAT(Attention(Q_1,K_1,V_1),\cdots,Attention(Q_N,K_N,V_N))\tag{6}$$

$CONCAT(\,)$ 为将 N 个注意力分数 $Attention(Q,K,V)$ 进行拼接的函数;W^A 为多头注意力机制中的权重参数矩阵. 为了保证数据特征分布的稳定性,使得生成器能够推理出真实数据分布的内在规律,采用 Add&Norm 组件. 其中包含残差连接(skip connect)和批量归一化(Batch Norm,BN),残差连接能解决随着网络层次加深带来的梯度消失和网络退化问题. 而批量归一化如公式(7)所示:

$$\begin{cases}BN(x)=\gamma\odot\dfrac{x-\mu}{\sqrt{\sigma^2+\varepsilon}}+\beta(\varepsilon>0)\\ \mu=\dfrac{1}{n}\sum\limits_{i=1}^{n}x_i\\ \sigma^2=\dfrac{1}{n}\sum\limits_{i=1}^{n}(x_i-\mu)\end{cases}\tag{7}$$

x 表示输入信息为 $[x_1,x_2,\cdots,x_n]$,μ 表示 x 的样本均值,σ^2 为样本方差,ε 是一个小的常量,γ 和 β 分别表示拉伸参数和偏移参数,形状与输入 x 相同,是可学习的参数. 利用批量归一化可将输入数据转换成 μ 为 0,σ^2 为 1 的数据分布. 利用 Add&Norm 将 $MultiHeadAttention^N(g_G')$ 更新为 $\bar{g}_G$,最后经过前馈神经网络和全连接层的映射即可生成目标车辆未来一段时间的位置信息:

$O^{T+pred}=\{(x^{T+1},y^{T+1}),\cdots,(x^{T+pred},y^{T+pred})\}$. 其步骤如下:

$$\bar{g}_G=f_{Add\&Norm}(MultiHeadAttention^N(g'_G)) \tag{8}$$

$$h_{FFN}=f_{FFN}(\bar{g}_G) \tag{9}$$

$$O^{T+pred}=f_{FC}(W_{FC}h_{FFN}+b) \tag{10}$$

式中,$f_{Add\&Norm}()$为 Add&Norm 函数;$f_{FFN}()$为前馈神经网络函数;h_{FFN}为前馈神经网络映射的信息;W_{FC}为全连接层的权重参数矩阵;b 为全连接层的偏置参数.

生成器预测目标车辆位置信息的算法伪代码如算法 1 所示.

算法 1. 生成器生成目标车辆的位置信息

输入: g = (V,E):多头注意力机制头数:K;

节点特征矩阵:H = $\{h_1,h_2,\cdots,h_N\}$;

GAT 层数:D;

Transformer 层数:L;

GAT 权重参数矩阵:$W^{d(k)}$, $\forall d\in\{1,\cdots,D\}$, $\forall k\in\{1,\cdots,K\}$

激活函数:σ

1. **begin**
2. **for** $d=1:D$ **do**
3. $e_{ij}^{d(k)}\leftarrow a([W^{d(k)}h_i^d \parallel W^{d(k)}h_j^d])$
4. $\alpha_{ij}^{d(k)}\leftarrow sofimax_j(e_{ij}^{d(k)})$
5. $h_i^{d+1}\leftarrow \parallel_{K=1}^{k}\sigma(\sum_{j\in N_i}\alpha_{ij}^{d(k)}W^{d(k)}h_j^d)$
6. **end for**
7. $z_i\leftarrow h_i^D, \forall i\in N$
8. $H_G\leftarrow z_i$
9. $g_G\leftarrow f_{addpool}(H_G)$
10. $g'_G\leftarrow f_{PosEncoding}(g_G)$
11. **for** $n=1:L$ **do**
12. $\bar{g}_G^n\leftarrow f_{Add\&Norm}(MultiHeadAttention(h_{FFN}^{n-1}))$
13. $h_{FFN}^n\leftarrow f_{FFN}(\bar{g}_G^n)$
14. **end for**
15. $O^{T+pred}\leftarrow f_{FC}(W_{FC}h_{FFN}^{(L)}+b)$
16. end

输出: O^{T+pred}

其中 $e_{ij}^{d(k)}$ 表示的是第 d 层第 k 个头计算的注意力系数,表示节点 i 和节点 j 的相关性;$\parallel$ 表示拼接操作;$a()$是一个映射,将拼接后的高维特征映射成一个实数;$\alpha_{ij}^{d(k)}$ 表示对 $e_{ij}^{d(k)}$ 归一化的注意力权重;z_i 为节点 i 最终的输出特征;$f_{addpool}()$为add_pool 操作的聚合函数;$f_{PosEncoding}()$为位置编码函数;h_{FFN}^{n-1}为 Transformer 网络中第 $n-1$ 层的输出特征,h_{FFN}^0表示添加位置编码信息的 g'_G.

2.4　基于 Transformer 轨迹编码的判别器

判别器模块如图 8 所示.

判别器接收两类信息,一个是目标车辆在预测时域内的真实位置信息,另一个则是生成器在预测时域中预测的位置信息,并将概率作 P 为输出用来判断接收的车辆位置数据信息是否为真实的数据样本. 判别器模块的目标是尽可能准确地区分真实数据和生成器生成的数据. 对于真实的数据,应该尽可能让输出的概率 P 趋近于 1;而对于生成的数据,应该尽可能让 P 趋近于 0,这样做的目的是为了提升生成器学习的能力,让其学习到真实数据的分布特征,从而生成更具真实性的数据. 本文的判别器主要由 2 层 Transformer 网络和一个全连接层构成,将车辆的位置信息作为 Transformer 神经网络的输入得到车辆位置信息特征,再经过全连接层的映射和 *sigmoid* 函数的计算,即可得到该位置信息属于真实数据的概率 P. 主要步骤如下:

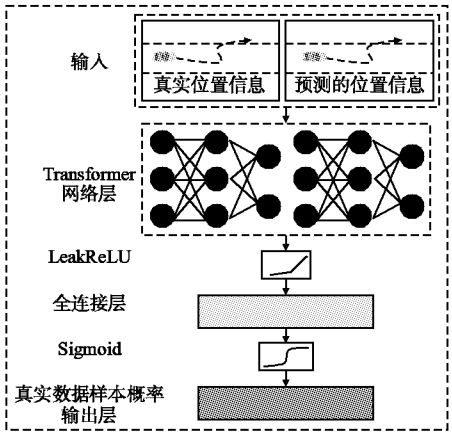


图 8　判别器模块

Fig. 8　Discriminator module

$$x'=f_{PosEncoding}(x) \tag{11}$$

$$x_A=f_{Add\&Norm}(MultiHeadAttention^N(x')) \tag{12}$$

$$x_{FFN}=f_{FFN}(x_A) \tag{13}$$

$$x_{FC}=f_{FC}(W_{FC}x_A+b) \tag{14}$$

$$P=sigmoid(x_{FC}) \tag{15}$$

其中输入信息 x 为车辆的位置信息;x′为经过位置编码后得到更新后的信息;x_A 为多头注意力机制和 *Add&Norm* 的计算得到的输出信息.

2.5　生成器与判别器的博弈过程

综上所述,在本文设计的模型中,生成器负责生成虚假的车辆位置信息预测结果,它的输入包含了目标车辆与周围交通实体间历史的动态交互信息,判别器负责区分真实的位置信息与生成器生成的“假”信息. 在整个训练过程中,判别器提升区分真假数据的能力,而生成器试图生成尽可能逼真的预测结果以便“骗”过判别器. 这两个模块相互博弈,不断迭代优化最终处于纳什均衡的状态. 生成器和判别器训练过程的算法伪代码如算法 2 所示.

算法 2. 生成器与判别器的训练过程

输入: 训练集:K

对抗训练迭代次数:T

小批量样本数量:M

1. **begin**
2. 随机初始化 $\boldsymbol{\theta},\boldsymbol{\phi}$;
3. **for** $t=1:T$ **do**
4. 先训练判别器 D(x;ϕ);
5. 从 K 中采集 M 个样本的真实位置$\{x^{(m)}\}$,$1\leqslant m\leqslant M$;
6. 从 K 中采集 M 个样本$\{g^{(m)}\}$,$1\leqslant m\leqslant M$;
7. 使用随机梯度上各更新 ϕ
8. $\frac{\partial}{\partial\phi}[\frac{1}{M}\sum_{m=1}^{M}(logD(x^{(m)};\phi)+log(1-$

$D(G(g^{(m)};\theta);\phi))]$

9. 后训练生成器 G(g;θ)

10. 使用随机梯度上升更新 θ;

11. $\frac{\partial}{\partial\theta}[\frac{1}{M}\sum_{m=1}^{M}(G(g^{(m)};\theta)-x^{(m)})^2]$

12. **end for**

13. **end**

输出:G(g;θ)

3 仿真实验

3.1 实验条件

仿真实验所采用的操作系统为 Windows 10 64bit,仿真开发环境采用 Pytorch 1.12.0,Pytorch_Geometric 2.2.0 和 PyCharm2022,开发语言 python 3.8.16,主要硬件有 CPU:12th Gen Intel(R) Core(TM) i7-12700H 2.30 GHz,RAM:16GB,GPU:NVIDIA GeForce RTX 3060.

3.2 数据集

本文采用的是 NGSIM(Next Generation Simulation)中的 I-80 公路数据集和 ApolloScape 数据集. NGSIM[13] 是一项由美国联邦公路局发起的数据采集项目,其中的 I-80 为美国 80 号州际高速公路,具有清晰的道路标识,可以用来结构化道路场景的模拟仿真实验. 而 ApolloScape[14] 是由百度公司发布的自动驾驶所使用的大规模数据集,其中包含了大量的城市非主干道、乡村等不同环境下的数据,可以用来非结构化道路场景的模拟仿真实验. 本文从 NGSIM 的 I-80 公路数据中筛选出了 2219 组数据,同时从 ApolloScape 数据中筛选出了 3214 组数据. 在非结构化道路情境下,考虑到车辆被观察的时间较短,本文选择了历史序列 1 秒(共 2 帧)来分别预测未来 1 秒和 2 秒的目标车辆位置信息. 相比之下,在结构化道路场景中,韩皓等人的研究[15] 表明,对于车辆位置信息的最佳预测历史序列长度为 3 秒. 因此,本文在该场景下选择了历史序列 3 秒(共 6 帧)来分别预测未来 1 ~5 秒的目标车辆位置信息. 在每个数据集中,本文按照 8:2 的比例划分了训练集和测试集.

3.3 数据预处理及损失函数

在本文中,将观察到的目标车辆第一个时刻位置作为原点,垂直于道路边界方向为 x 轴,沿道路边界方向为 y 轴,重新为每个时刻的目标车辆和周围交通参与者分配坐标,如图 9 所示.

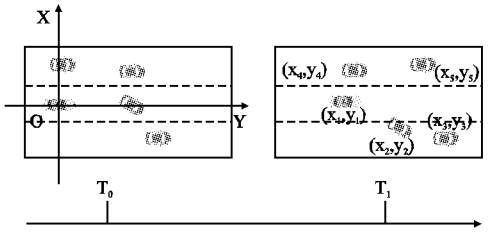


图 9 建立坐标系

Fig. 9 Establish a coordinate system

通过参与者前后时刻的位置变化分别计算出其 x 在方向和 y 方向的速度 vx 和 vy,此外为每个交通参与者分配类型特征 $c\in\{1,2,3,4,5\}$,其中 1 表示小型车辆;2 表示大型车辆;3 表示行人;4 表示自行车;5 表示其他(例如宠物等).

在本文中针对判别器和生成器的训练过程中采用不同的损失函数策略可以加快收敛速度和防止模型崩溃. 对于判别器,本文使用了二进制交叉熵损失(BCELoss)作为损失函数,该损失函数针对二分类问题使用对数变换避免了数值上溢和下溢问题,使训练过程更加稳定;而对于生成器,选择均方误差损失(MSELoss)作为损失函数,该损失函数一方面计算生成器生成的数据与真实数据之间的平均平方差,使得在训练过程中更容易收敛,另一方面 MSELoss 对异常值较为敏感,可以帮助生成器更专注于生成接近真实数据的样本,从而提高生成器的生成质量.

3.4 评估指标

本文中的实验评估指标有两个,第 1 个平均绝对误差(Mean Absolute Error,MAE),它是预测值和真实值之间误差的平均值,如公式所示:

$$MAE=\frac{1}{n}\sum_{i=1}^{n}|Y_P^i-Y_T^i| \tag{16}$$

第 2 个为均方根误差(Root Mean Square Error,RMSE),它是预测值和真实值之间平方差值平均值的平方根,如公式所示:

$$RMSE=\sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_P{}^i-Y_T^i)^2} \tag{17}$$

公式中 Y_P^i 为第 i 个的预测值,Y_T^i 为真实值.

3.5 实验参数设置

实验参数主要分为模型内部主要的超参数设置和模型训练参数. 其中模型内部主要的超参数设置如表 1(生成器)和表 2(判别器)所示,模型的训练参数如表 3 所示.

表 1 生成器主要超参数设置

Table 1 Main hyperparameter settings of generator

参 数	参数大小/设置
GAT 网络层 1	in_channels = 5, out_channels = 5, heads = 8
GAT 网络层 2	in_channels = 40, out_channels = 16, heads = 8
激活函数	ReLU()
Transformer 网络层	Q_K_V_size = 32, heads = 8, 层数 = 2

表 2 判别器主要超参数设置

Table 2 Main hyperparameter settings of discriminator

参 数	参数大小/设置
Transformer 网络层	Q_K_V_size = 32, heads = 8, 层数 = 2
激活函数	LeakReLU(), Sigmoid()

表 3 模型训练的超参数设置

Table 3 Hyperparameter setting for model training

参 数	参数大小/设置
训练轮次(Epoch)	100
批量大小(Batch_size)	64
优化器	Adam
学习率	0.001

3.6 对比实验

为了测试本文提出的基于 GAT 和 Transformer 时空注意力机制的 GAN 模型的预测效果,在 NGSIM 和 ApolloScape 两个数据集中分别与 DGInet[8]、VectorNet[9] 和 GRIP[10] 的预

测模型进行对比. 其中,在 ApolloScape 数据集的测试效果如表 4 所示.

表 4　ApolloScape 数据集中各模型的 MAE 和 RMSE

Table 4　MAE and RMSE of each model in the ApolloScape dataset

预测时域/秒	本文模型 (MAE/RMSE)	DGInet (MAE/RMSE)	VectorNet (MAE/RMSE)	GRIP (MAE/RMSE)
1	0.107/0.119	0.113/0.124	0.143/0.178	0.188/0.216
2	0.122/0.141	0.144/0.162	0.185/0.226	0.248/0.261

从表 4 可以看出在预测时域 1 秒的任务中本文提出的模型预测效果相较于 DGInet、VectorNet 和 GRIP,MAE 分别提升了 5.31 百分点、25.17 百分点、43.08 百分点,RMSE 分别提升了 4.03 百分点、33.14 百分点、44.90 百分点. 而在预测时域 2 秒中本文模型比上述 3 个模型,MAE 分别提升了 15.28百分点、34.05 百分点、50.80 百分点,RMSE 分别提升了 12.96 百分点、37.61 百分点、45.98 百分点.

在 NGSIM 数据集的测试效果如表 5 所示.

表 5　NGSIM 数据集中各模型的 MAE 和 RMSE

Table 5　MAE and RMSE of each model in the NGSIM dataset

预测时域/秒	本文模型 (MAE/RMSE)	DGInet (MAE/RMSE)	VectorNet (MAE/RMSE)	GRIP (MAE/RMSE)
1	0.088/0.104	0.095/0.109	0.102/0.112	0.134/0.145
2	0.137/0.148	0.159/0.196	0.185/0.261	0.229/0.312
3	0.254/0.261	0.293/0.304	0.287/0.296	0.343/0.371
4	0.296/0.307	0.334/0.342	0.374/0.387	0.422/0.431
5	0.325/0.336	0.376/0.398	0.439/0.451	0.489/0.513

从表 5 中可以看出,该数据集中本文所设计的预测模型在长短时域内对目标车辆未来的位置信息预测的精度最高. 在预测时域 1 秒中本文相较于其他 3 个模型 MAE 平均提高了 18.48 百分点,RMSE 平均提高了 13.34 百分点,在预测时域 2 秒中 MAE 平均提高了 26.65 百分点,RMSE 平均提高了 41.52 百分点,在预测时域 3 秒中 MAE 平均提高了 17.6 百分点,RMSE 平均提高了 18.54 百分点,在预测时域 4 秒中 MAE 平均提高了 20.70 百分点,RMSE 平均提高了 19.89 百分点,在预测时域 5 秒中 MAE 平均提高了 25.07 百分点,RMSE 平均提高了 25.19 百分点. 不难看出随着预测时域的增加,对目标车辆未来位置信息的预测误差也随之增加,这是由于长时域会导致车辆运动的不确定性增加. 对模型的性能要求也在增加. 但本文的模型的平均绝对误差和均方根误差增长较为缓慢,说明基于时空注意力机制能够增加预测模型的语义信息,从而提高了目标车辆位置信息的预测精度. 综合来看,本文模型在两个数据集上的测试结果均有较高的精度,泛化能力更强.

此外,本文还从模型计算的时间开销进行对比,结果如图 10 和图 11 所示.

图 10 展现的是各模型在 NGSIM 测试集(445 个样本)中完成时域 1 ~5 秒预测所需的时间开销,图 11 展现的是各模型在 ApolloScape 测试集(643 个样本)中完成时域 1 秒和 2 秒预测所需的时间开销. 可以看出本文模型所需的时间开销均是最小,这是由于一方面本文构建无向图的方式有效地减少了计算中的冗余信息,另一方面本文所采用的GAN模型

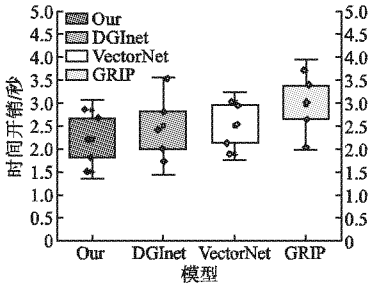


图 10　NGSIM 中完成测试集预测时域 1 ~5 秒的时间开销

Fig. 10　Time cost of completing test set prediction in the time domain of 1 to 5 seconds in NGSIM

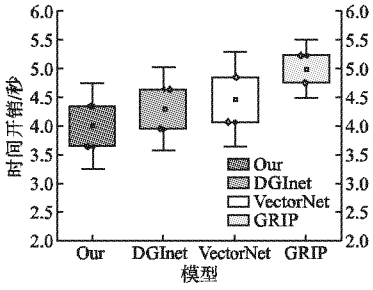


图 11　ApolloScape 中完成测试集预测时域 1 ~2 秒的时间开销

Fig. 11　Time cost of completing test set prediction in ApolloScape in the time domain of 1 to 2 seconds

产生的样本是一次生成,与其他模型需多次运用马尔可夫链采样来生成样本有所不同.

3.7　消融实验

本文设计了消融实验来验证所采用的交互边构建方法的合理性和优越性.

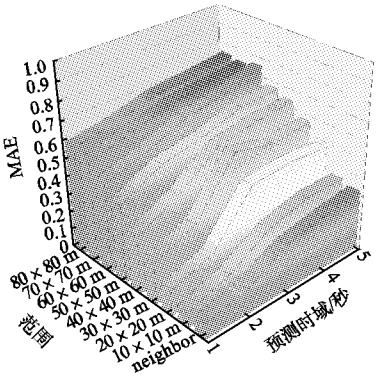


图 12　考虑不同范围内邻近车辆交互影响的 MAE

Fig. 12　MAE considering the interaction effects of adjacent vehicles in different ranges

在 NGSIM 数据集上,分别考虑了目标车辆半径 10 米 ~80 米范围内交通参与者的直接交互影响. 同时,与本文只考虑邻近参与者(neighbor)进行比较,结果如图 12 和图 13

所示.

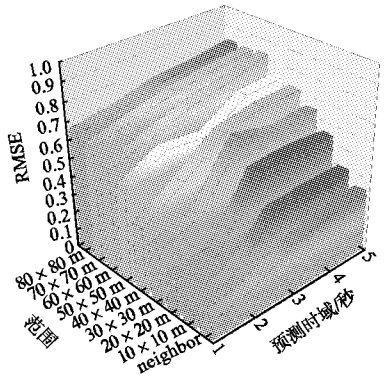


图 13 考虑不同范围内邻近车辆交互影响的 RMSE

Fig. 13 RMSE considering the interaction effects of adjacent vehicles within different ranges

由实验结果观察到,在 1 ~ 5 秒的预测时段内,仅考虑邻近车辆的交互影响所得到的 MAE 和 RMSE 均最小. 表明在结构化道路的情况下,仅考虑邻近车辆的交互影响就足以实现较高的预测精度.

在 ApolloScape 数据集中,由于涉及到不同类型的交通参与者,本文对于行人、自行车等这些移速较慢的参与者,本文考虑了 0 ~ 50 米的交互范围. 实验结果如图 14 所示.

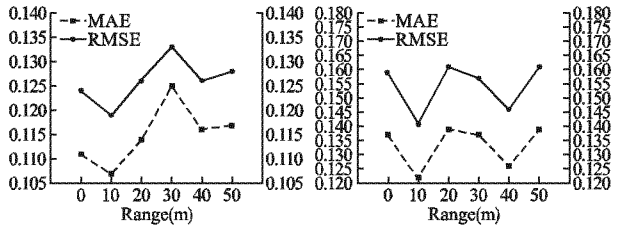


图 14 考虑不同范围内其他类型参与者交互影响的 MAE 和 RMSE

Fig. 14 MAE and RMSE considering the interaction effects of other types of participants within different ranges

其中,其中范围为 0 米表示忽略了这类交通参与者的交互影响. 值得注意的是,在对移速较慢的交通参与者进行预测时,最佳效果是将交互范围限制在 10 米内,随着范围的增大,其对目标车辆的影响逐渐减小. 针对高速移动的车辆这一类交通参与者,本文则分别考虑了 10 ~ 80 米范围内的交互影响. 结果如图 15 所示.

可以观察到,对于具有高速移动性的车辆,仅需考虑 40 米范围内的交互影响即可取得较好的预测效果.

3.8 模型可解释性及实例分析

相较于其他深度学习模型的“黑盒”特性,本文的模型更注重可解释性能更好地展现其内部机制和预测结果的解释. 本文采用了 GAT 图神经网络来建立目标车辆与周围交通参与者之间的空间注意力机制. 其核心思想是每个节点可以根据与其相邻节点之间的关系来计算一组注意力权重. 同时,本文借助 Transformer 神经网络构建了预测信息和历史信息之间的时间注意力机制,以深入探究预测信息与历史信息之间的依赖关系. 通过 GAT 网络,本文展示了目标车辆作为源节点与其相邻节点之间的注意力权重分布,从而研究了源节点对相邻节点特征的关注情况. 此外,通过 Transformer 的应用,本文呈现了历史信息对预测信息的影响程度,揭示了这两者之间的动态关联.

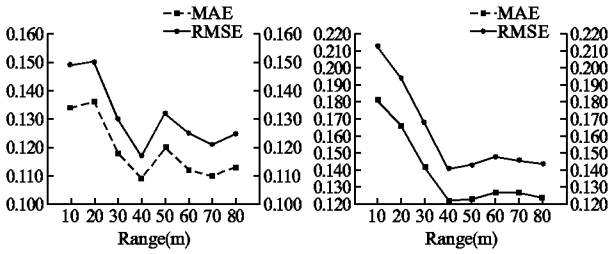


图 15 考虑不同范围内车辆交互影响的 MAE 和 RMSE

Fig. 15 MAE and RMSE considering the impact of vehicle interaction within different ranges

在 NGSIM 数据集中,源节点与其相邻节点之间的注意力权重如图 16 所示.

其中,H1 ~ H8 为 GAT 网络中多头注意力机制的头数,x 和 y 为坐标,vx 和 vy 分别为 x、y 方向的速度,c 为交通参与者的类型. 随着预测时域增加,所学习到的注意力分数没有出现显著的变化,这表明 GAT 网络更加注重空间维度的信息. 此外,图中还反映出目标车辆对周围车辆的位置、速度和类型赋予不同的重要程度. 然而,在结构化道路场景中,车辆之间的互动占主导地位,因此节点类型这一特征得到的权重较低. 相比之下,车辆的高速移动性导致其位置和速度特征获得了较高的权重分配.

在 NGSIM 数据集中,历史信息对预测信息的影响权重如图 17 所示.

其中,T0 ~ T5 代表了 6 个历史时间步. 从图中可以观察到,历史时刻对预测信息的影响程度各不相同. 随着预测时域的增加,较早时刻的权重逐渐减小,而较近时刻的影响逐渐增大. 在 ApolloScape 数据集中,源节点与其相邻节点之间的注意力权重如图 18 所示.

观察结果表明,在非结构化道路场景中,涉及到大量不同类型的交通参与者,这导致模型对节点类型的赋权更加重要. 值得注意的是,本文的模型也会对节点的速度特征赋予较高的权重,因为速度被认为是影响预测结果的一个关键因素.

在 ApolloScape 数据集中,历史信息对预测信息的影响权重如图 19 所示,预测时域越长,较后时刻的影响程度越大,较前时刻的影响会逐渐淡化.

为了更直观体现本文模型的在长时域预测的性能效果,从两个数据集中各选取一个样本进行 2 秒和 5 秒的预测,结果如图 20 所示,其中,黑色实线表示的是真实位置信息. 可以看出,本文模型的预测结果更接近真实的位置信息,预测精度更高.

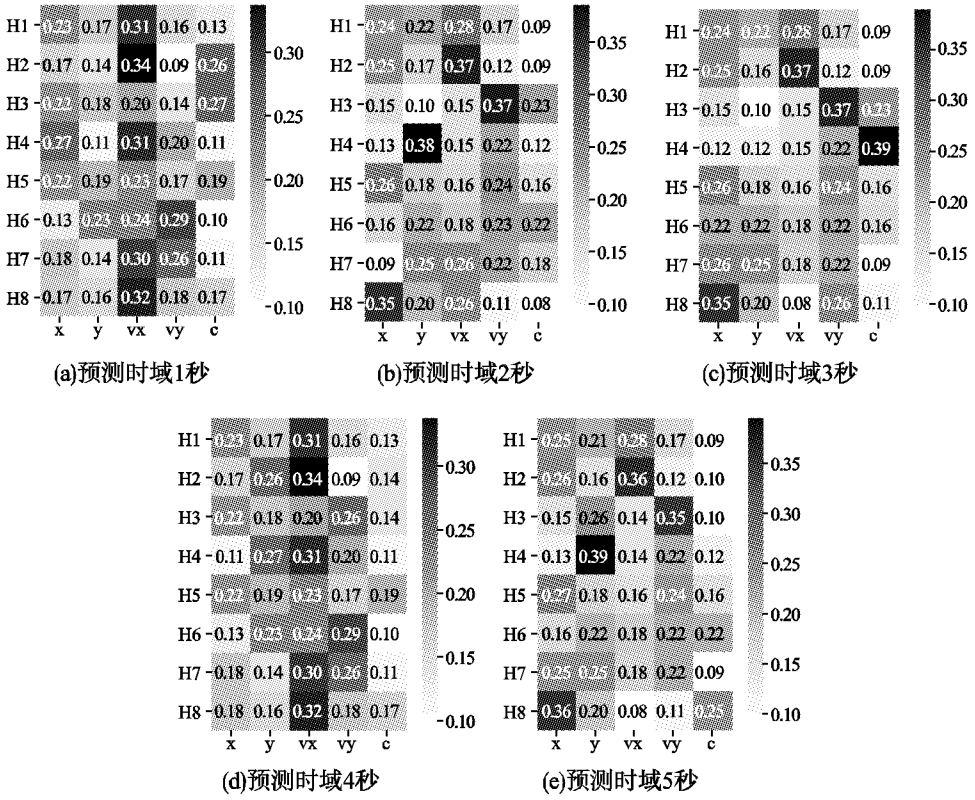


图 16　NGSIM 中源节点与相邻节点注意力权重

Fig. 16　Attention weights of source and adjacent nodes in NGSIM

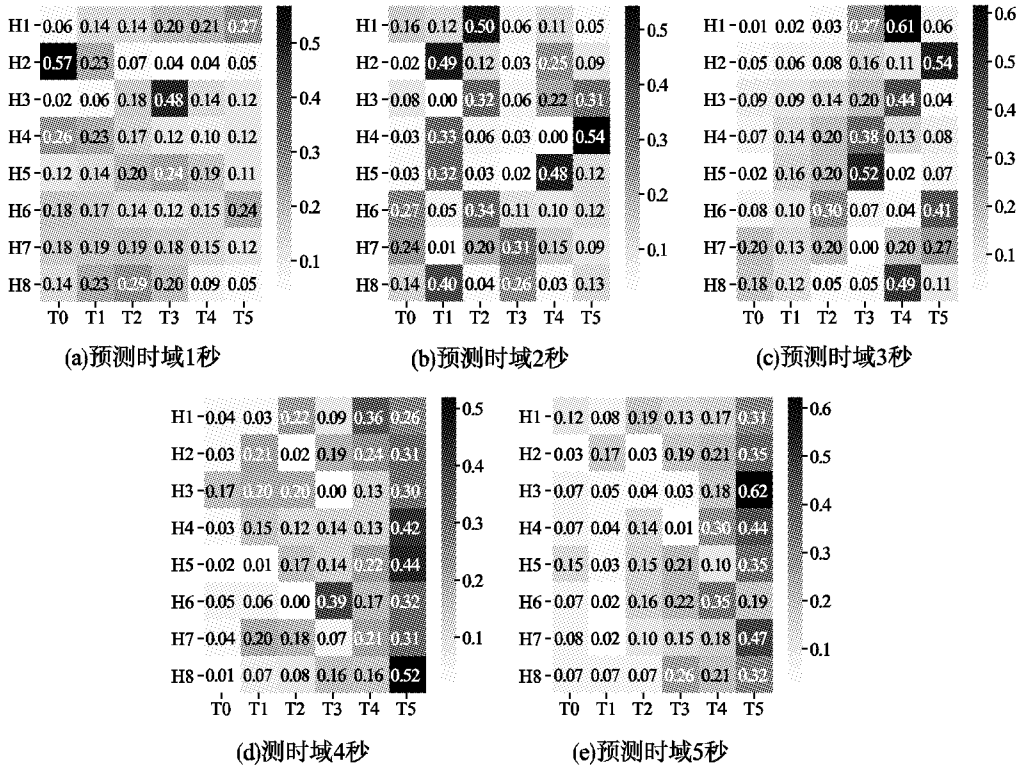
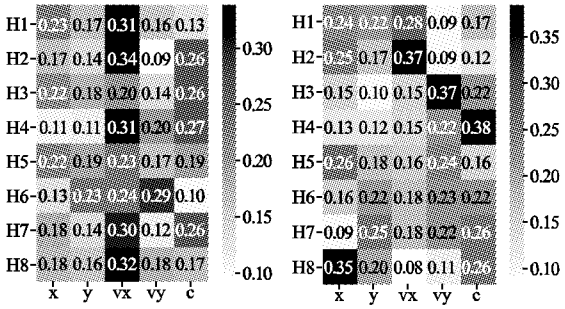


图 17　不同预测时域下历史信息对预测信息的影响权重

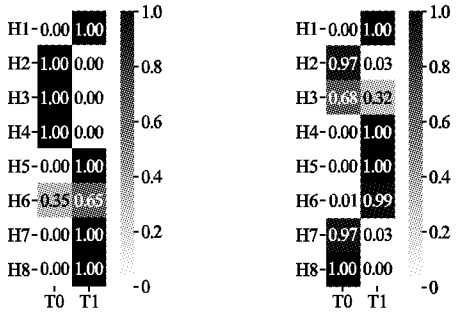
Fig. 17　Influence weights of historical information on predictive information in different prediction time domains



(a)预测时域1秒 (b)预测时域2秒

图 18 ApolloScape 中源节点与相邻节点注意力权重

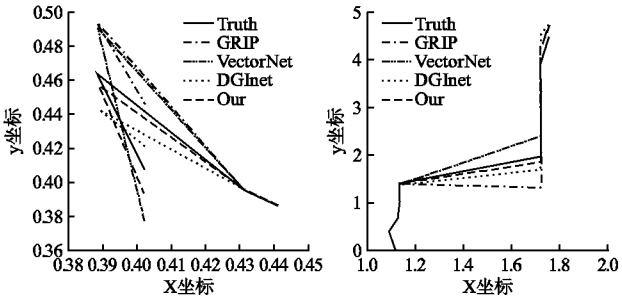
Fig. 18 Attention weights of source and adjacent nodes in ApolloScape



(a)预测时域1秒 (b)预测时域2秒

图 19 ApolloScape 中不同预测时域下历史信息对预测信息的影响权重

Fig. 19 Influence weights of historical information on predictive information under different prediction time domains in ApolloScape



(a)预测时域2秒 (b)预测时域5秒

图 20 预测位置信息和真实位置信息对比

Fig. 20 Comparison between predicted location information and actual location information

4 结束语

本文提出了一种基于 GAT 和 Transformer 的 GAN 模型并用于车辆行为预测，分析了结构化和非结构化道路环境下被预测车辆与其周围交通参与者的相对位置关系，构建了参与者之间的动态交互时空图，采用 GAT 图神经网络对其推理，为被预测车辆的相邻交通参与者分配不同的权重并提取空间维度的语义信息，运用 Transformer 网络提取时间维度语义信息的同时建立历史信息与预测信息的长依赖关系，使其输出更为合理的预测结果，并在 NGSIM 和 ApolloScape 自动驾驶数据集上进行了验证. 结果表明，该模型能更好地推理出交通参与者之间的交互关系，提高自动驾驶车辆对周围交通参与者状态的认知能力. 本文模型是基于车辆与周围交通参与者的动态交互时空图数据进行训练的，未考虑参与者具备一定的先验知识（如驾驶意图等），同时，车辆的运动的位置信息也受到交通环境中交通流等的影响，因此后续研究中将考虑参与者的驾驶意图以及将更多道路信息融入到模型中，得到更加精准的预测结果.

References:

[1] Mozaffari S, Al-Jarrah O Y, Dianati M, et al. Deep learning-based vehicle behavior prediction for autonomous driving applications: a review[J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 23(1): 33-47.

[2] Qian L P, Feng A, Yu N, et al. Vehicular networking-enabled vehicle state prediction via two-level quantized adaptive kalman filtering[J]. IEEE Internet of Things Journal, 2020, 7(8): 7181-7193.

[3] Deo N, Rangesh A, Trivedi M M. How would surround vehicles move? a unified framework for maneuver classification and motion prediction[J]. IEEE Transactions on Intelligent Vehicles, 2018, 3(2): 129-140.

[4] Shi K, Wu Y, Shi H, et al. An integrated car-following and lane changing vehicle trajectory prediction algorithm based on a deep neural network[J]. Physica A: Statistical Mechanics and its Applications, 2022, 599: 127303.

[5] GAO Z H, BAO M X, GAO F, et al. A uni-modal network prediction method for surrounding vehicle expected trajectory in intelligent driving system[J]. Automobile Technology, 2022, (11): 1-9, doi: 10.26914/c.cnkihy.2022.090385.

[6] Fang L, Jiang Q, Shi J, et al. Tpnet: trajectory proposal network for motion prediction[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 6797-6806.

[7] Choi D, Yim J, Baek M, et al. Machine learning-based vehicle trajectory prediction using v2v communications and on-board sensors[J]. Electronics, 2021, 10(4): 420.

[8] An J, Liu W, Liu Q, et al. DGInet: dynamic graph and interaction-aware convolutional network for vehicle trajectory prediction[J]. Neural Networks, 2022, 151: 336-348.

[9] Gao J, Sun C, Zhao H, et al. Vectornet: encoding hd maps and agent dynamics from vectorized representation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 11525-11533.

[10] Li X, Ying X, Chuah M C. Grip: Graph-based interaction-aware trajectory prediction[C]//IEEE Intelligent Transportation Systems Conference (ITSC), 2019: 3960-3966.

[11] Lin L, Li W, Bi H, et al. Vehicle trajectory prediction using LSTMs with spatial-temporal attention mechanisms[J]. IEEE Intelligent Transportation Systems Magazine, 2021, 14(2): 197-208.

[12] Li L, Sui X, Lian J, et al. Vehicle interaction behavior prediction with self-attention[J]. Sensors, 2022, 22(2): 429, doi: 10.3390/s22020429.

[13] Punzo V, Borzacchiello M T, Ciuffo B. On the assessment of vehicle trajectory data accuracy and application to the Next Generation SIMulation (NGSIM) program data[J]. Transportation Research Part C: Emerging Technologies, 2011, 19(6): 1243-1262.

[14] Ma Y, Zhu X, Zhang S, et al. Trafficpredict: trajectory prediction for heterogeneous traffic-agents[C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2019, 33(1): 6120-6127.

[15] HAN H, XIE T. Lane change trajectory prediction of vehicles in highway interweaving area using Seq2Seq-attention network[J]. China Journal of Highway and Transport, 2020, 33(6): 106-118.

附中文参考文献:

[5] 高振海，鲍明喜，高 菲，等. 智能驾驶系统对周边交通车辆预期轨迹的单模态网络预测方法[J]. 汽车技术，2022，(11)：1-9，doi：10.26914/c.cnkihy.2022.090385.

[15] 韩 皓，谢 天. 基于注意力 Seq2Seq 网络的高速公路交织区车辆变道轨迹预测[J]. 中国公路学报，2020，33(6)：106-118.