



DOI:10.12404/j.issn.1671-1815.2307832

引用格式:蒋挺,杨柳,刘亚林,等.基于分解动态时空分解框架预测交通流量[J].科学技术与工程,2025,25(7):3007-3017.

Jiang Ting, Yang Liu, Liu Yalin, et al. Traffic flow prediction based on the dynamic spatial-temporal decomposition framework [J]. Science Technology and Engineering, 2025, 25(7): 3007-3017.

# 基于分解动态时空分解框架预测交通流量

蒋挺<sup>1</sup>, 杨柳<sup>1\*</sup>, 刘亚林<sup>2,3</sup>, 张邵华<sup>2,3</sup>, 石硕<sup>2,3</sup>

(1. 西南交通大学信息科学与技术学院, 成都 611756; 2. 极端环境岩土和隧道工程智能建养全国重点实验室, 西安 710043; 3. 中铁第一勘察设计院集团有限公司, 西安 710043)

**摘要** 近几年,时空图卷积网络(spatial-temporal graph convolutional network, STGCN)被引入交通流量预测中,具有良好的时空交通数据建模能力,取得了先进的性能,但是仍存在两个问题:①交通流量数据具有很强的时空相关性;②静态的预定义图难以捕获交通流随时间动态变化的时空依赖关系。为解决以上问题,提出了一种新的时空分解框架(spatial-temporal decomposed framework, STDF),它使用了残差连接、遗忘门、更新门,将时间模块和空间模块有机连接起来,以将输入信息进行多层次双维度的分解和预测。此外将STDF进行实例化,提出一种新的基于输入交通信号分解的动态时空融合的交通预测模型(decomposed dynamic spatial-temporal graph convolutional network, DDSTGCN),它捕捉了交通的时空相关性,并设计了一个动态图学习模块,考虑了空间依赖的动态性质。最后利用两个真实交通流量的数据(在PEMS04和PEMS08的数据集),与现有的交通流量预测算法进行对比。实验结果证明,所提方法在交通流量预测的准确率有良好的性能表现,能够有效地完成真实场景下的交通流量预测。

**关键词** 交通流量预测; 时空图卷积网络(STGCN); 时空相关性; 时空融合; 动态图学习

**中图分类号** U491.1 TP183; **文献标志码** A

## Traffic Flow Prediction Based on the Dynamic Spatial-temporal Decomposition Framework

JIANG Ting<sup>1</sup>, YANG Liu<sup>1\*</sup>, LIU Ya-lin<sup>2,3</sup>, ZHANG Shao-hua<sup>2,3</sup>, SHI Shuo<sup>2,3</sup>

(1. School of Information Science and Technology, Southwest Jiaotong University, Chengdu 611756, China;

2. Key Laboratory of Geotechnical and Tunnel Engineering in Extreme Environments, Xi'an 710043, China;

3. China Railway First Survey and Design Institute Group Co., Ltd., Xi'an 710043, China)

**[Abstract]** In recent years, spatial-temporal graph convolutional network (STGCN) has been introduced into traffic flow prediction, which has good spatial-temporal traffic data modeling ability and has achieved advanced performance, but there are still two problems; ①Traffic flow data have strong temporal and spatial correlation; ②Static pre-defined graphs are difficult to capture the spatio-temporal dependence of dynamic changes in traffic flow over time. To solve the above problems, a new spatial-temporal decomposed framework (STDF) was proposed, which used residual connection, forgetting gate and update gate to organically connect time module and space module to decompose and predict input information in multiple dimensions. In addition, by instantiating STDF, a new traffic prediction model based on input traffic signal decomposition decomposed dynamic spatial-temporal graph convolutional network (DDSTGCN) was proposed. It captured the spatiotemporal dependencies of traffic and designed a dynamic graph learning module that takes into account the dynamic nature of spatial dependencies. Finally, two real traffic flow data were used to compare with the existing traffic flow prediction algorithms. The experimental results show that the proposed method has good performance in the accuracy of traffic flow prediction and can effectively complete the traffic flow prediction in the real scenario.

**[Keywords]** traffic flow forecast; spatiotemporal graph convolutional network (STGCN); spatiotemporal correlation; space-time fusion; dynamic graph learning

当前,全球城市交通系统普遍面临着安全隐患、拥堵问题和环境负荷等多重挑战。然而,数字化技术的崛起,其中包括云计算、物联网、移动互联网和人工

智能等,为城市交通管理注入了智能化的可能性。这种智能化转型的核心在于运用先进的数据分析方法,为决策者提供科学、全面的决策支持。

收稿日期:2023-10-09 修订日期:2024-07-09

基金项目:四川省科技创新基地(平台)和人才计划(2022JDR0356);四川省科技计划(软科学项目)(2021JDR0101);中铁一院科研项目(2022KY49ZD(ZNGZ)-01)

第一作者:蒋挺(1999—),男,汉族,四川成都人,硕士研究生。研究方向:深度学习。E-mail:923895809@qq.com。

\* 通信作者:杨柳(1978—),女,汉族,四川达州人,博士,高级工程师。研究方向:移动通信与工程信息化。E-mail:yangliu@swjtu.edu.cn。

投稿网址:www.stae.com.cn

随着城市规模的不断膨胀,空气质量下降和交通流量过大成为普遍现象。预测性交通管理或早期干预是解决这些问题的有效手段。这些手段一般在智能交通系统(intelligent traffic system, ITS)中得以实施,其依托于道路传感器、公共交通系统的数据日志以及实时监控视频等多源数据<sup>[1]</sup>。

以2015年的统计数据为例,美国驾驶人员的日均驾驶时长约为48 min,这表明了实时交通信息的不可或缺性。不论是个体驾驶者、商业运输组织还是政府管理部门,都极需对交通状态进行实时而准确的评估。从流量控制到路线优化,高质量的交通数据为这些应用提供了基础。在学术研究和实际应用中,交通流预测和状态评估被视为关键要素。研究者通常会选取如车流速度、交通流量和密度等基础变量作为研究对象。根据预测的时效性,这些模型可以进一步被分类为短期与中长期预测模型。根据预测时间长短,交通预测一般分为两个尺度:短期(5~30 min),中长期(30 min以上)<sup>[2]</sup>。

实时交通数据分析的准确性对于交通管理有着至关重要的作用。这样的数据可以被运用于交通规划、实时流量控制和指导驾驶者等多个方面。通过精准的预测和实时响应,不仅可以解决当前的交通问题,还能显著提升交通系统的整体运行效率,从而达到缓解城市交通压力的目的<sup>[3]</sup>。

目前国内外对于交通流量预测的研究方法主要分为以下三类。

(1) 基于统计分析的预测模型。主要包括有:历史平均分析预测法、时间序列分析预测法、卡尔曼滤波分析预测法等。这些方法的前提是假设将来要预测的数据具有与过去相同的特征。Ahmed等<sup>[4]</sup>提出了ARIMA(autoregressive integrated moving average),它常用于交通预测。Williams等<sup>[5]</sup>设计了季节ARIMA,它考虑到季节性因素,是对ARIMA的改进。但是它只适用于稳定路段而无法应对突发交通情况,并且这些方法无法很好地处理高非线性的时间序列,忽略了时间序列之间的复杂相关性,限制了交通预测的有效性。

(2) 基于非线性理论模型,包括基于小波分析、混沌理论和深度学习的模型,是用于短时交通流预测的主要方法。尽管短时交通流数据呈现非线性特征,这些方法从理论上更适合应对这种情况。然而,它们在计算上相对复杂,数学运算较为困难,因此实现完美的预测十分具有挑战性。它的优点在于它们能够提供高精度的预测结果,并且能够有效挖掘非线性关系。然而,随着智能交通技术的快速发展,现实条件变得更加复杂,需要考虑更多的影响因素。因此,单

一的非线性理论模型已经无法满足当前交通预测需求,需要进一步考虑综合多种方法和维度的综合模型来提高预测的准确性和可应用性。

(3) 基于多模型融合的方法。通过将多种模型结合起来,充分发挥各自模型的优势,从而提高性能。这种方法可以有效降低误差,并更好地挖掘交通数据的时空特征,提高交通流预测的准确性,满足智能交通快速发展的要求。

近几年来,预测交通流量的主流方法是基于单一深度学习模型或多种深度学习模型的融合。常用神经网络有反向传播神经网络(back propagation neural network, BPNN)、小波神经网络和长短时记忆神经网络等。

Qiao等<sup>[6]</sup>提出了并行的长短期记忆网络(long short-term memory, LSTM)提取交通流的短时和周期特征,取得良好结果。任艺柯<sup>[7]</sup>通过融合单向和双向LSTM交通预测模型考虑了交通流上下文的相关性。Arif等<sup>[8]</sup>基于非参数回归的深度学习方法进行交通流的预测。Koesdwiady等<sup>[9]</sup>使用深度信念网络(deep belief network, DBN)考虑进天气因素,并用真实数据集进行了验证。但它们只考虑了时间序列,没有考虑空间上的联系。而不同区域之间的空间关系是复杂的,难以明确地描述。

其次,时间维度的时间依赖可能相互关联。挖掘潜在的时空模式是实现精确的交通预测的关键。于是随着近几年,图卷积网络(graph convolutional networks, GCNs)被广泛认为是交通流的空间依赖性建模,基于时空图神经网络(spatial-temporal graph neural network, STGNN)的模型被不断提出,它是构建邻接矩阵来建模道路网复杂的空间拓扑结构,将交通数据用时空图进行表示,是一种将时间和空间信息结合的一种有效的方法。STGCN<sup>[10]</sup>用定义在无向图上的谱图卷积来模拟空间依赖关系,而扩散卷积递归神经网络(diffusion convolutional recurrent neural network, DCRNN)<sup>[11]</sup>则采用有向图上的扩散图卷积来适应交通流的方向。然而,他们忽略了交通状况的动态变化(如高峰期和交通事故),因为空间依赖关系一旦被训练好就固定了。空间依赖关系是随着空间和时间块的深度动态生成的,而不是实际的时间步骤<sup>[12]</sup>。Graph WaveNet<sup>[13]</sup>通过对图中每个节点的可学习嵌入来提高交通预测的准确性。Guo等<sup>[14]</sup>提出基于注意力机制的时空图卷积模型基于注意力的时空图卷积网络(attention based spatial-temporal graph convolutional network, ASTGCN),捕获时间和空间的动态相关性。

虽然它们都取得了一定的成果,但是这些方法

存在以下两种问题:①它们没有充分建模交通运输中时间和空间信息的相关性;②采用了预定义的邻接矩阵,忽略了交通状况空间上的变化,而静态的预定义图难以捕获交通流随时间动态变化的时空依赖关系。为了解决①的问题,Shao 等<sup>[15]</sup>提出了一种新的时空解耦框架,将输入数据不断进行逐步分解,但它没有进行残差连接,输入下一层时空块的信息越来越少,本文中认为它设计的框架缺乏可解释性,且时空模块中仍还有更大的改良空间。为了解决问题②,Sun 等<sup>[16]</sup>提出了一种自适应时空图神经网络 Ada-STNet,该网络首先在节点属性的引导下获得最优的图结构,然后通过一个专门的时空卷积结构捕捉复杂的时空相关性,用于多步交通状况预测,但它设计的时空模块过于简单,仍有很大的优化价值。

基于对各种预测方法的优劣性分析,提出一种交通预测时空分解框架(spatial-temporal decomposed framework, STDF),它使用了残差连接、遗忘门、更新门,将时间模块和空间模块有机连接起来,以将输入信息进行多层次双维度的分解和预测。这使得对交通数据的不同部分进行更精确的建模,以提高预测精度。并在 STDF 的基础上,将其时间模块和空间模块进行实例化,依次设计了基于扩散图卷积的空间模块以捕捉空间信息,并设计了基于残差块(residual block)、门控循环单元(gated recurrent units, GRUs)、自注意力机制(self-attention)的时间序列模型捕捉时间信息,将此模型称为分解动态时空图卷积网络(decomposed dynamic spatial-temporal graph convolutional network, DDSTGCN),还提出了一个动态图结构学习组件,从宏观和微观的角度获得最优的图邻接矩阵。以此更充分地利用各种信息,来动态地调整预定义图结构。

通过分析交通数据的时空关系来提高预测准确性。通过重视不同地点间的相互作用来捕捉全面的空间信息,并强化时间信息提取,实现短长期预测的平衡。同时,利用路网的空间结构,完善交通数据的空间特征分析,并提高交通预测的准确度。

通过在 PEMS04、PEMS08 上的实验证明,所提方法较 DGCRN 等方法具有更好的预测效果。随后又进行了消融实验,证明了模型各结构的有效性。

通过城市交通预测可以提前了解未来的交通状况,从而帮助交通管理部门制定相应的措施,优化交通流量,减少交通拥堵,提高交通效率。此外,交通预测还可以改善出行体验,出行者可以根据预测结果选择最佳路线和出行方式,避免拥堵和堵

车。同时,通过交通预测还可以提前预警可能导致交通事故的隐患,及时采取相应的管控措施,降低交通事故发生的概率,增强道路安全性。

## 1 问题定义及时空分析

交通网络可以被抽象为有向或无向图  $G = (V, E, A)$ ,其中  $V$  是指  $|V|$  节点的集合,每个节点对应一个交通道路部署的传感器,它是部署在交通系统中的传感器,如道路网络,记录着交通信息。 $E$  是  $|E|$  的边的集合,  $A \in \mathbf{R}^{N \times N}$  代表节点间的可达性,表示加权邻接矩阵,存储了任意两个节点间位置的邻近程度。

假设输入  $X_t \in \mathbf{R}^{N \times C}$  表示对交通网络  $G$  在  $t$  时刻第  $N$  个节点观察到的交通特征(如过往车辆的流量或车辆的速度等),其中  $C$  表征观测交通流特征的维度。 $T_h$  是指过去时间步数,  $T_f$  是指未来时间步数,而交通流预测是旨在通过历史交通信号  $\chi = [X_{t-T_h+1}, \dots, X_{t-1}, X_t] \in \mathbf{R}^{T_h \times N \times C}$  预测最近的未来时间步长  $T_f$  的未来交通信号  $\gamma = [X_{t+1}, X_{t+2}, \dots, X_{t+T_f}]$ 。

图 1(a) 显示了不同时段交通流的空间影响,每个节点代表不同地域空间,之间的粗线代表它们之间互相影响的强度,线条越深影响越大,可见相同时间不同空间的影响强度是不同的,并且随着时间的改变影响强度也会发生改变。图 1(b) 是不同位置的历史观测数据对于节点 A 的交通状态有着不同的影响。在相邻地点和时间戳的观测结果不是独立的,而是相互动态关联的。由此可见如何捕捉交通流的时空相关性是至关重要的问题。

图 2 展现了一个交通流量数据实例,可以得出观察结果:同一道路网内的交通流量可能随时间而变化,即空间依赖性也是动态的。

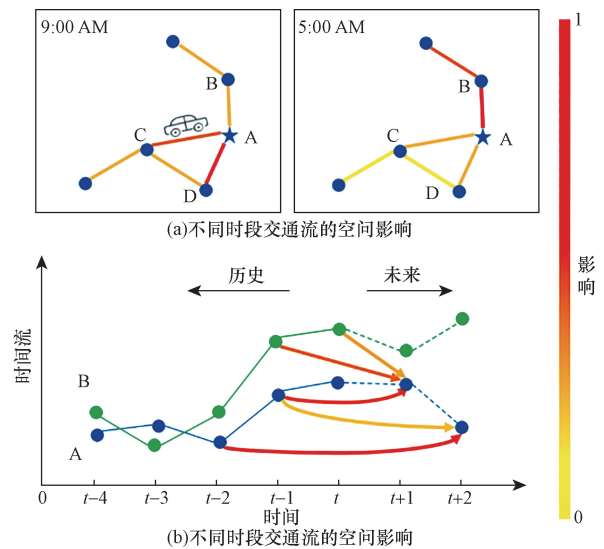


图 1 交通流时空相关图

Fig. 1 The spatial-temporal correlation diagram of traffic flow

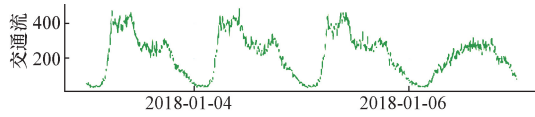


图2 交通流量数据

Fig. 2 The traffic flow data

## 2 时空分解框架

### 2.1 框架总体架构

本文中提出的时空分解框架 STDF 如图 3 所示。首先将原始交通信号  $\mathbf{R}^{T_h \times N \times C}$  通过线性层转换到潜在空间  $\mathbf{R}^{T_h \times N \times d}$ , 为了简单起见本文中  $\chi^l \in \mathbf{R}^{T_h \times N \times d}$  作为第  $l$  步输入的默认值, 本文中称为第  $l$  步的循环传递信息, 并且  $\chi^0 = \chi$ 。STDF 包含有遗忘门和更新门, 并且进行残差连接, 遗忘门旨在将输入  $\chi$  进行分解, 将其进行动态调整, 以丢弃不重要信息; 此后将通过空间模块和时间模块进行信息的更新, 其中空间模块将学习循环传递信息中关于传感器拓扑空间上的信息, 时间模块学习循环传递信息关于时序上的信息。最后将输入信号  $\chi^l$  与该步通过时空块后得到结果进行残差连接, 获得  $\chi^{l+1}$ , 以此作为第  $l+1$  步的循环传递信息进行输入, 以此循环。

### 2.2 循环残差分解机制

为充分捕捉到时空相关性, 首先设计了循环残差分解机制。通过将循环传递信息  $\chi^l \in \mathbf{R}^{T_h \times N \times d}$  进行遗忘、更新和保留, 不断循环地去除非重要的信息, 并将每个模块学到的有用信息进行连接, 得到多步预测值以进行汇总得到最终预测结果。如图 3 所示, 绿色部分为分解模块, 它将传递的信息进行遗忘、更新和保留(通过残差连接); 粉红色的部分为空间模块; 蓝色的部分为时间模块, 每个都包含三个组成部分, 分别为: ①某个主要模型, 将学习输入信息  $\chi^* \in \mathbf{R}^{T_h \times N \times d}$  并生成隐藏信息  $H^* \in \mathbf{R}^{T_h \times N \times d}$  作为候选值; ②一个预测分支的生成模块, 可得到

预测隐藏状态  $H_t^*$ ; ③一个更新门, 将该模块学到的有用信息作为更新的反馈值  $\chi_r^* \in \mathbf{R}^{T_h \times N \times d}$ 。其中 \* 表示空间或时间块信号。

本文将信号在通过遗忘门后得到的保留信息称为  $\chi^{rem}$ , 此后信息将通过空间模块和时间模块的更新门作为更新, 更新门可以将隐藏状态进行重构, 通过非线性层能够很好地近似输入信号的部分, 因此, 在通过第一个更新门后, 本文中得到时间模块的输入, 即时间信号为

$$\chi^{time} = \chi^{rem} + \chi_u^{spa} = \chi^{rem} + \sigma(H^{spa} W_u^{spa}) \quad (1)$$

式(1)中:  $\chi^{spa}$  和  $\chi^{time}$  分别为空间和时间信息。本文中使用时空非线性层来实现更新门;  $W_u^{spa}$  为网络可学习参数训练权重矩阵;  $u$  为更新门参数;  $\sigma$  为 relu 激活函数。类似地本文在时间模块引入第二个更新连接, 并将其与第  $l$  层输入的循环传递信息进行残差连接, 以此避免随模型深度增加带来的梯度消失和模型退化问题。表达式为

$$\begin{aligned} \chi^{l+1} &= \chi^l + \chi^{time} + \chi_u^{time} \\ &= \chi^l + \chi^{time} + \sigma(H^{time} W_u^{time}) \end{aligned} \quad (2)$$

式(2)中:  $\chi^{l+1}$  表示通过第  $l$  层后得到的余留的更新信号。如图 3 所示本文通过叠加多个时空层以此增强模型的学习能力, 每个模块将根据各自负责方面进行相关方面的学习。

值得注意的是, 图 3 例子中是先通过空间模块再通过时间模块, 实际它们是可互换的, 并在后续进行了实验, 以验证在性能上没有明显差异。

尽管可以通过空间/时间模块进行信息的更新, 但它并不是足够的, 还需要在每步丢弃掉其中的非重要信息, 以免造成误差的累计, 为此设计了遗忘门。首先为每个节点初始化两个空间 embedding 矩阵  $E^u, E^d \in \mathbf{R}^{N \times d}$ , 前者表示 source 状态下的 embedding 用于节点向相邻节点传递消息、后者表示 target 状态下的 embedding 用于节点从相邻节点聚

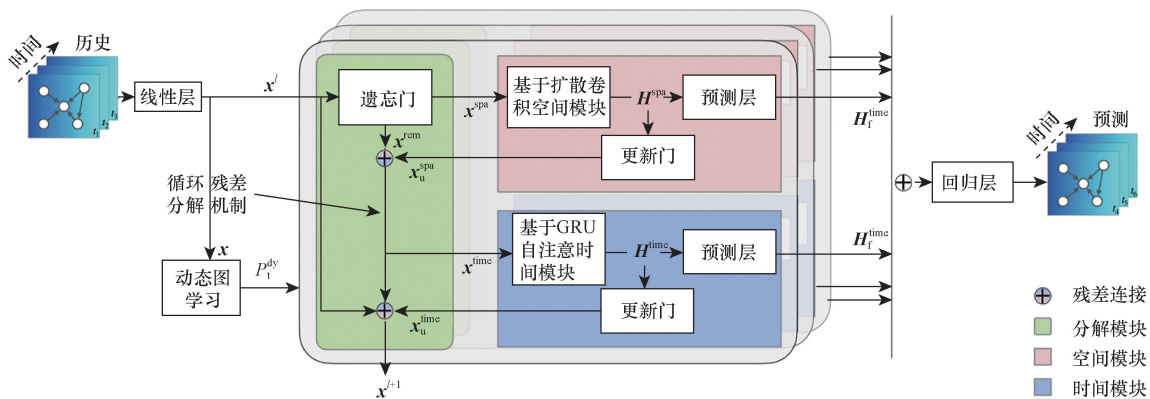


图3 提出的 STDF 的总体架构

Fig. 3 The overall architecture of the proposed STDF

集信息。为考虑交通信号的周期性,而对于每个输入的时间片,可以判断它是属于周几和属于该天的哪个时间段,由此引入了这两个时间特征作为周期特征,并且将它们通过 embedding 层进行初始化以获得  $T^D \in \mathbf{R}^{N_d \times d}$ ,  $T^W \in \mathbf{R}^{N_w \times d}$  矩阵( $D$ 指 day,  $W$ 指 week)。其中  $N_d$  为一天内 time step 的数量,  $N_w$  为一周内的天数为 7。将这些特征拼接到一起,通过两个全连接层和 sigmoid 激活函数,以产生对应阈值。其公式为

$$\Omega_{i,t} = \text{Sigmoid}[\sigma(T_i^D \parallel T_i^W \parallel E_i^u \parallel T_i^l) W_1] W_2 \quad (3)$$

$$\chi^{\text{rem}} = \chi^{\text{spa}} = \Omega \odot \chi^l \quad (4)$$

式中:  $\Omega \in \mathbf{R}^{T_h \times N \times d}$ ,  $\Omega_{i,t} \in (0, 1)$  为节点  $i$  时隙  $t$  的输入交通数据到遗忘门后保留有用信息的比例;  $\odot$  广播到输入信号  $\chi^l \in \mathbf{R}^{T_h \times N \times d}$  每个信道的元素乘积;  $W_1 \in \mathbf{R}^{4d \times d}$  和  $W_2 \in \mathbf{R}^{d \times 1}$  是可学习参数;  $\sigma$  为 relu 激活函数;  $\chi^l$  为第  $l$  步输入的循环传递信息;  $\chi^{\text{spa}}$  为输入进空间模块的输入值;  $\chi^{\text{rem}}$  为通过遗忘门后获取的余留信息。

### 3 分解动态时空图卷积网络

下面将介绍 STDF 实例化中的空间模型和时间模型,它们将重点放在各自擅长学习的地方,分别捕捉传感器拓扑空间上的信息和循环传递信息关于时序上的信息。此外介绍了用于空间模型的动态图学习模块,它将从宏观和微观的角度获得最优的图邻接矩阵。以此更充分地利用各种信息,来动态地调整预定义图结构。

#### 3.1 基于扩散卷积的空间模块

时空图神经网络的核心思想是在交通流空间上建模为一个扩散过程,并使用扩散卷积进行捕捉,此后再通过时间模块捕捉时序上的信息,从而捕捉交通数据的时空相关性。扩散过程和扩散卷积最先在 DCRNN 中被提出<sup>[11]</sup>,扩散过程是指目标节点的未来扩散信号取决于相邻节点的近期值,即设置具有时空局部性,为此本文中假设只有过去  $k_t$  时间步长的相邻  $k_s$  个节点可以影响目标节点,为捕捉这样的扩散过程本文设计了一个空间模型。首先,定义一个时空局部转移矩阵,公式为

$$(P^{\text{local}})^k = [P^k \odot (\mathbf{1} - I_N)] \parallel \cdots \parallel [P^k \odot (\mathbf{1} - I_N)] \quad (5)$$

式(5)中:  $P$  为邻接矩阵的转移矩阵包括前向转移矩阵和后向转移矩阵将在后面提到;  $P^k$  为  $k$  阶转移矩阵,  $\parallel$  是指将它进行拼接。  $I_N$  为  $N \times N$  单位矩阵,该公式进行拼接的次数为  $k_t$ 。  $[i, i + k'N]$  ( $k' = 0, 1, \dots, k_{t-1}$ ) 将被屏蔽为零,它代表节点本身固有的信息,将

由时间模块进行学习。为将扩散过程进行建模,将扩散方向分为前向和后向,分别对应前向转移矩阵  $P_f = A / \text{rowsum}(A)$  和后向转移矩阵  $P_b = A^T / \text{rowsum}(A)$ , 将其统称为转移矩阵  $P$ , 其中  $A$  为输入的预定义的邻接矩阵。但本文还利用了一个动态图学习模块,将转移矩阵  $P$  转化为自适应转移矩阵,为简单起见,3.2 节提到的  $P$  都为经过动态图学习后的自适应邻接矩阵,其包含两个自适应邻接矩阵为自适应前向转移矩阵  $P_f$  和自适应后向转移矩阵  $P_b$ 。

类似地,将局部特征矩阵定义为

$$X_t^{\text{lc}} = \underbrace{[\sigma^T(X_{t-k_t+1}^{\text{spa}} W_{k_t-1}) \parallel \cdots \parallel \sigma^T(X_t^{\text{spa}} W_0)]^T}_{k_t} \quad (6)$$

式(6)中:  $W_k$  为可学习的参数;  $\sigma$  为可学习的激活函数。这里使用非线性变换旨在增强模型的表现力。

因此,基于上面提到的转移矩阵  $(P^{\text{local}})^k$  和特征矩阵  $X_t^{\text{lc}}$ , 定义了空间核大小为  $k$  的时空局部图卷积算子,表达式为

$$H_t^{\text{spa}} = \sum_{k=1}^{k_s} (P^{\text{lc}})^k X_t^{\text{lc}} W_k \quad (7)$$

式(7)中:  $H_t^{\text{spa}} \in \mathbf{R}^{N \times d}$  为时间步长  $t$  的局部图卷积算子的输出,它考虑了来自  $k$  阶邻居的空间信息;  $W_k$  为  $k$  阶图卷积参数;  $H_t^{\text{spa}}$  为后续时隙的隐藏状态,可以用来预测空间部分。

由此计算获得隐藏输出值,公式为

$$H_t^{\text{spa}} = \sum_{k=1}^{k_s} [(P_f^{\text{lc}})^k X_t^{\text{lc}} W_{k_1} + (P_b^{\text{lc}})^k X_t^{\text{lc}} W_{k_2}] \quad (8)$$

$$H^{\text{spa}} = \Theta G(X^{\text{spa}}) = [\cdots, H_{T-2}^{\text{spa}}, H_{T-1}^{\text{spa}}, H_T^{\text{spa}}] \quad (9)$$

预测层:通过自回归的方法,用最后一个隐藏状态  $H_T^{\text{spa}}$  生成后面多步的隐藏状态即  $H_t^{\text{spa}} = [H_{T+1}^{\text{spa}}, H_{T+2}^{\text{spa}}, \cdots, H_{T+T_t}^{\text{spa}}]$ , 将收集每一步的预测状态进行汇总,生成最终的预测结果。

更新门:更新门可以将隐藏状态进行重构,通过非线性层能够很好地近似输入信号的部分,  $\chi_u^{\text{spa}} = \sigma(H^{\text{spa}} W_u^{\text{spa}})$ , 从而反馈到循环信号使其状态得到更新。

#### 3.2 时间模块

本章设计的时间模型基于残差块 (Residual Block)、门控循环单位 (GRUs)、自注意力机制 (self-attention), 其中 Residual Block, 由一个 Dense + ReLU 层、一个 Dense 线性层、一个 dropout 层、一个 Add&LayerNorm 组成,它最先由 Das A 等提出且取得了良好的效果<sup>[17]</sup>。其中 Residual Block 是允许信息在深层神经网络中更顺畅地传递,降低梯度消失的风险,加速训练过程,提高模型的泛化能力,同时通过正则化层 (Dropout) 减少过拟合的风险。GRU

善于捕捉短期依赖,而自注意力层善于捕捉长期依赖。时间模型如图4所示。

本文中使用的残差块作为本文的时间模型的基础层。它是一个带有 ReLU 激活函数隐藏层的 MLP。它还有一个完全线性的残差连接。在将隐藏层映射到输出的线性层上使用 dropout,并在输出处使用 layer norm。dropout 和 layer norm 都可以通过模型超参数调整开关。

门控循环单元(gated recurrent unit, GRU)是循环神经网络(RNN)的变体,专用于处理序列数据如时间序列。与传统的 RNN 相比,GRU 引入了门控机制,以解决长期依赖问题,同时减少了梯度消失的风险。GRU 在很多序列建模任务中表现出色,是深度学习领域中的一个重要组件。可以循环地保存历史数据的隐藏状态,并控制流向下一个时间步骤的信息。如图4所示,对于每个节点*i*,它在某时间*t*时,输入为 $X_t^{time} \in \mathbf{R}^{N \times d}$ ,在通过 GRU 时,将进行如下操作,表达式为

$$\begin{cases} z_t = \sigma(\mathbf{W}_z X_t^{time}[i, :] + \mathbf{U}_z \mathbf{H}_{t-1}^{time}[i, :] + bz) \\ r_t = \sigma(\mathbf{W}_r X_t^{time}[i, :] + \mathbf{U}_r \mathbf{H}_{t-1}^{time}[i, :] + br) \\ \tilde{\mathbf{H}}_t^{time}[i, :] = \tanh[\mathbf{W}_h X_t^{time}[i, :] + r_t \odot (\mathbf{U}_h \mathbf{H}_{t-1}^{time}[i, :] + bh)] \\ \hat{\mathbf{H}}_t^{time}[i, :] = (1 - zt) \odot \hat{\mathbf{H}}_{t-1}^{time}[i, :] + zt \odot \tilde{\mathbf{H}}_t^{time}[i, :] \end{cases} \quad (10)$$

式(10)中: $\tilde{\mathbf{H}}_t^{time}[i, :]$ 表示节点*i*在时间步长*t*上的更新隐藏状态; $\odot$ 表示元素乘积; $\mathbf{W}_z, \mathbf{W}_r, \mathbf{W}_h$ 和 $\mathbf{U}_z, \mathbf{U}_r$ 表示可学习参数。

为了捕捉全局信息,本文中引入了多头自注意力层,多头自注意力(multi-head self-attention)是深度

学习领域中在自然语言处理等任务中广泛使用的一种注意力机制。它是通过并行地在不同的表示子空间中进行自注意力计算,来捕捉不同的语义信息,并将这些信息整合到一个更丰富的表示中。但由于多头自注意力机制会忽略序列中的相对位置,所以在 GRU 层和多头自注意力层中间加入了一个位置编码器层,公式为

$$\begin{cases} \tilde{\mathbf{H}}_t^{time}[i, :] = \hat{\mathbf{H}}_t^{time}[i, :] + \mathbf{e}_t \\ \mathbf{e}_{t,i} = \begin{cases} \sin(t/10000^{2i/d}), & i = 0, 2, 4, \dots \\ \cos(t/10000^{2i/d}), & \text{其他} \end{cases} \end{cases} \quad (11)$$

式(11)中: $\mathbf{e}_t \in \mathbf{R}^d$ 将时间步长进行嵌入,它是不可训练的。

给定注意力,可学习矩阵 $\mathbf{W}_s^Q, \mathbf{W}_s^K, \mathbf{W}_s^V \in \mathbf{R}^{d \times d}$ ,和输出矩阵 $\mathbf{W}^O$ ,节点*i*的注意力函数为

$$\begin{aligned} \mathbf{H}^{time}[:, i, :] &= \text{Multihead}(\mathbf{H}^{vi}) \\ &= \text{Concat}(\text{head}_1, \dots, \text{head}_s) \mathbf{W}^O \text{ where } \text{head}_s \\ &= \text{Attention}_s(\mathbf{H}^{vi}) \\ &= \text{softmax} \left[ \frac{\mathbf{H}^{vi} \mathbf{W}_s^Q (\mathbf{H}^{vi} \mathbf{W}_s^K)^T}{\sqrt{d}} \mathbf{H}^{vi} \mathbf{W}_s^V \right] \end{aligned} \quad (12)$$

式(12)中: $\mathbf{H}^{vi} \in \mathbf{R}^{T \times d}$ 为节点在每个时隙的特征,如图4所示每个节点将被单独并行计算。

预测层:这里也通过自回归的方法,用最后一个隐藏状态 $\mathbf{H}_T^{spa}$ 生成后面多步的隐藏状态即 $\mathbf{H}_f^{time} = [\mathbf{H}_{T+1}^{time}, \mathbf{H}_{T+2}^{time}, \dots, \mathbf{H}_{T+T_f}^{time}]$ ,将收集每一步的预测状态进行汇总,生成最终的预测结果。

更新门:更新门可以将隐藏状态进行重构,通过非线性层能够很好地近似输入信号的部分, $\chi_u^{time} = \sigma(\mathbf{H}^{time} \mathbf{W}_u^{time})$ ,从而反馈到循环信号使其状态得到更新。

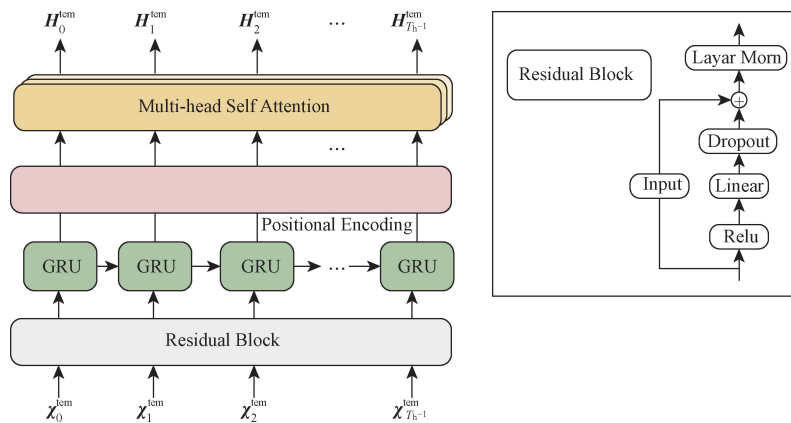


图4 时间模型

Fig. 4 The time model

### 3.3 动态图学习

近年来时空图神经网络(ST-GNNs)的模型在交通预测问题上取得了很大的进步。它的模型的核心思想,就是将交通系统在空间维度上的依赖建模为一个扩散过程,并使用扩散卷积<sup>[11]</sup>——图卷积网络(graph convolution network, GCN)<sup>[18]</sup>的一种变体对其进行捕捉。他们再进一步将扩散卷积和其他序列模型结合(例如RNN、CNN、Transformer等),从而捕捉交通数据中复杂的时空关联。

ST-GNN通常设计用于空间关系学习的GNN,以及用于时间依赖性提取的RNN或CNN。通过GCN提取了交通状况的空间关系,并通过因果卷积捕获了时间依赖性。可以得出结论,大多数现有的ST-GNN首先构建由预定义测量确定的图,然后对构建的图进行研究。

然而,预定义的图可能不足以包含复杂的交通预测场景中的所有有用信息。在大多数情况下,这样的图由每对区域之间的欧几里得距离确定。但是节点之间的扩散关系是动态变化的,早8点的早高峰时期节点之间的扩散关系和早10点节点之间的扩散关系肯定是不同的。这意味着本文需要基于路网图,进行对空间依赖(即图结构)进行调整。由此,本课题的做法是更充分地利用各种信息,来动态地调整预定义图地图结构。

具体来说,提出了一种自适应图结构学习组件,分别从宏观和微观角度获得最优图结构。具体而言,宏观视角描述了长期复杂的空间关系,微观视角反映了短期时变的空间关系。并且构建了一个时空卷积网络来预测多个时段的未来交通状况,它可以捕获交通预测的空间关系和时间依赖性。

#### 3.3.1 宏观特征学习

(1)从宏观上看,图中节点的空间关系相对稳定,反映了它们内在的相关性。然而,预定义的邻接矩阵 $\mathbf{A}$ 只能反映单侧性质,这是由欧氏距离确定的。因此,预定义邻接矩阵 $\mathbf{A}$ 不足以包含所有有价值的信息,因为与交通状况相关的一些隐含因素[如POI(point of interest distribution)分布、区域功能]被预定义规则忽略。为此,设计了一个宏观层次的图结构学习模块,以描述更全面的长期信息。

本文中目标不是在没有任何先验信息的情况下直接设计一个可学习的,而是学习难以被预定义规则捕获的隐含因素,然后将学习到的隐藏关系注入到预定义的邻接矩阵 $\mathbf{A}$ 中,以实现信息的补充。为将扩散过程进行建模,将扩散方向分为前向转移和后向转移,分别对应前向转移矩阵 $\mathbf{P}_f = \mathbf{A}/\text{rowsum}(\mathbf{A})$ 和后向转移矩阵 $\mathbf{P}_b =$

$\mathbf{A}^T/\text{rowsum}(\mathbf{A}^T)$ ,并将其统称为转移矩阵 $\mathbf{P}$ ,其中 $\mathbf{A}$ 为输入的预定义的邻接矩阵。

具体地,本文旨在获得的宏观邻接矩阵 $\mathbf{A}_{\text{Ma}}$ 被定义为

$$\mathbf{A}_{\text{Ma}} = \mathbf{P} + \Delta\mathbf{A} \quad (12)$$

式(12)中: $\Delta\mathbf{A}$ 表示图中节点的可训练隐藏关系。通过将可训练节点嵌入字典空间维度的特征( $\mathbf{E}^u, \mathbf{E}^d \in \mathbf{R}^{N \times d}$ )进行以下计算获取 $\Delta\mathbf{A}$ ,即

$$\Delta\mathbf{A} = \text{softmax}[\text{relu}(\mathbf{E}^u \mathbf{E}^d{}^T)] \quad (13)$$

这样不同于直接定义可学习矩阵 $\mathbf{E} \in \mathbf{R}^{N \times N}$ 的好处在于可以节省更多参数,加快学习效率。

#### 3.3.1 微观特征学习

从微观角度来看,图中节点的空间关系会由于交通事件而发生剧烈变化,这些交通事件包括常规的和意外的,例如高峰时间,天气状况或交通事故。由于节点与记录交通状况的属性相关联,假设通过挖掘节点属性中的相关信息来描述突发波动。具体来说,本文使用每天时间序列一段时间内的历史信息、时间维度的特征( $\mathbf{T}^d \in \mathbf{R}^{N_d \times d}, \mathbf{T}^w \in \mathbf{R}^{N_w \times d}$ )和空间维度的特征( $\mathbf{E}^u, \mathbf{E}^d \in \mathbf{R}^{N \times d}$ )作为节点的动态的特征,首先在通过embedding参数对齐后,将它们进行拼接。公式为

$$\mathbf{M}_t^u = \text{Concat}[\text{FC}(\bigparallel_{c=1}^C \mathbf{X}_c), \mathbf{T}_t^d, \mathbf{T}_t^w, \mathbf{E}^u] \quad (14)$$

$$\mathbf{M}_t^d = \text{Concat}[\text{FC}(\bigparallel_{c=1}^C \mathbf{X}_c), \mathbf{T}_t^d, \mathbf{T}_t^w, \mathbf{E}^d] \quad (15)$$

式中: $\mathbf{M}_t^u, \mathbf{M}_t^d \in \mathbf{R}^{N \times d'}$ ,分别包含了前向转移矩阵和后向转移矩阵的特征信息; $\bigparallel_{c=1}^C$ 表示通道 $c$ 从1到 $C$ 的连接。

在跨时间维度的聚合之后,为简单起见将 $\mathbf{M}_t^u, \mathbf{M}_t^d$ 统称为 $\mathbf{M}$ ,设计度量学习方法以经由如下学习成对节点表示的度量函数 $\varphi(\cdot, \cdot)$ 来导出节点之间的关系,公式为

$$\mathbf{A}_{\text{Mi}}[i, j] = \varphi(\mathbf{M}_i, \mathbf{M}_j) = \mathbf{M}_i \cdot \mathbf{M}_j^T, \quad 1 \leq i, j \leq N \quad (16)$$

式(16)中: $\mathbf{A}_{\text{Mi}}[i, j]$ 表示节点 $i$ 和节点 $j$ 之间的学习关系。

由于本文中采用点积来表示节点的接近度,因此度量学习函数可以用矩阵乘法格式表示为

$$\mathbf{A}_{\text{Mi}} = \mathbf{M}\mathbf{M}^T \quad (17)$$

通过以上两个模块,可以同时得到宏观层次的图邻接矩阵 $\mathbf{A}_{\text{Ma}}$ 和微观层次的图邻接矩阵 $\mathbf{A}_{\text{Mi}}$ 。为了便于交通量预测任务,需要一种最优的图结构,通过将两个邻接矩阵用relu激活函数求和,并通过归一化得到图的邻接矩阵,即

$$\mathbf{A}^* = \text{Norm}[\text{relu}(\mathbf{A}_{\text{Ma}} + \mathbf{A}_{\text{Mi}})] \quad (18)$$

在上述融合过程中可以利用更复杂的设计,如注意力机制。在本文中,选择简单的求和运算,因为它在实验中已经可以取得令人满意的性能。

### 3.4 输出和训练方法

通过第  $l$  层两个模块预测层,可以得到  $\mathbf{H}_t^{\text{spa},l} = [\mathbf{H}_{T+1}^{\text{spa}}, \mathbf{H}_{T+2}^{\text{spa}}, \dots, \mathbf{H}_{T+T_f}^{\text{spa}}]$  和  $\mathbf{H}_t^{\text{time},l} = [\mathbf{H}_{T+1}^{\text{time}}, \mathbf{H}_{T+2}^{\text{time}}, \dots, \mathbf{H}_{T+T_f}^{\text{time}}]$ ,先收集每层的预测隐藏状态进行相加,即

$$\begin{aligned} \mathbf{H} &= \mathbf{H}_t^{\text{spa}} + \mathbf{H}_t^{\text{time}} = \sum_{l=0}^{L-1} \mathbf{H}_t^{\text{spa},l} + \sum_{l=0}^{L-1} \mathbf{H}_t^{\text{time},l} \\ &= \left[ \sum_{l=0}^{L-1} (\mathbf{H}_{T+1}^{\text{spa},l} + \mathbf{H}_{T+1}^{\text{time},l}), \right. \\ &\quad \left. \sum_{l=0}^{L-1} (\mathbf{H}_{T+2}^{\text{spa},l} + \mathbf{H}_{T+2}^{\text{time},l}), \dots \right] \end{aligned} \quad (19)$$

然后本文中采用两层全连接层作为回归层,将其应用于  $\mathbf{H}$ ,生成最终的预测。回归层将每个时间步的输出串联起来,形成最终的输出:  $\hat{\mathbf{y}} \in \mathbf{R}^{T_f \times N \times C_{\text{out}}}$ 。给定真实值  $\mathbf{y} \in \mathbf{R}^{T_f \times N \times C_{\text{out}}}$ ,本文中使用了 MAE 作为损失函数,用梯度下降的方法来优化模型,即

$$L(\hat{\mathbf{y}}, \mathbf{y}; \theta) = \frac{1}{T_f N C_{\text{out}}} \sum_{i=1}^{T_f} \sum_{j=1}^N \sum_{k=1}^{C_{\text{out}}} | \hat{y}_{i,j,k} - y_{i,j,k} | \quad (20)$$

式(20)中:  $N$  为节点的数量;  $T_f$  为预测步骤的数量,  $C_{\text{out}}$  为输出维度。

## 4 实验

### 4.1 数据集描述

为验证模型的有效性,本文在两个公开的真实交通数据集<sup>[11]</sup>上进行了实验,两个数据集介绍如下:

PEMS04 包含旧金山湾区(美国加利福尼亚州旧金山大湾区)29 条高速公路 307 个探测器,2018 年 1 月 1 日—2 月 28 日这 2 个月的数据。

PEMS08 包含圣贝纳迪诺市(美国加利福尼亚州南部一座城市)8 条高速公路 170 个探测器,2016 年 7 月 1 日—8 月 31 日这 2 个月的数据。

PEMS04 和 PEMS08 数据集都以 5 min 的时间间隔汇总交通流速数据,这两个数据集的详细统计数据如表 1 所示。

表 1 本文数据集的统计信息

Table 1 Statistics of Datasets Used in This Paper

数据集	节点	边	样本量
PEMS04	307	680	16 992
PEMS08	170	548	17 856

### 4.2 基线方法

选择一批具有代表性的用于交通流预测任务的

算法。

(1) HA (historical average model)<sup>[19]</sup>: 历史平均模型。

(2) VAR (vector auto-regression)<sup>[20]</sup>: 假设传递的时间序列是平稳的,并估计时间序列与其滞后值之间的关系。

(3) SVR (support vector regression)<sup>[21]</sup>: 将线性支持向量机用于经典的时间序列回归任务。

(4) FC-LSTM (fully connected long short-term memory unit)<sup>[22]</sup>: 全连接长短时记忆单元构成的循环神经网络,仅建模时间依赖。

(5) DCRNN<sup>[11]</sup> (diffusion convolutional recurrent neural network): 扩散卷积循环神经网络,集成了扩散卷积和门控循环单元。

(6) Graph WaveNet<sup>[13]</sup>: 将门控 TCN 和 GCN 层进行逐层堆叠,以同时捕获时空相关性。

(7) ASTGCN<sup>[14]</sup>: 基于时空注意力机制捕捉交通数据的动态时空特征。

(8) STSGCN<sup>[23]</sup>: 它有效地捕捉局部化的时空相关性,并考虑时空数据的异质性。

(9) GMAN<sup>[24]</sup>: 是一个基于注意力的模型,它将空间、时间和变换的注意力叠加在一起。

(10) MTGNN<sup>[25]</sup>: 通过空间模块中的 mix-hop 传播层、时间模块中的 dilated inception 层和更细致的图学习层扩展了 Graph WaveNet。

(11) DGCRN<sup>[26]</sup>: 对动态图进行建模,设计了一种新的动态图卷积递归模块来捕获交通信号的时空相关性。

### 4.3 实验环境和参数设置

配备 RTX 3090 显卡,将 Epoch 设为 300,数据集为 PEMS04 和 PEMS08,用前一个小时 12 个点预测后一个小时 12 个点,批量大小 batch\_size 统一设置为 32,预测时间为 15、30、60 min,对于 PEMS04 和 PEMS08,本文使用约 60% 的数据用于训练,20% 的数据用于测试,其余 20% 用于验证,比较了 15 min (horizon3), 30 min (horizon6) 和 1 小时 (horizon12) 在 MAE、RMSE 和 MAPE 指标上进行预测。

### 4.4 实验结果与分析

结果如表 2 所示,容易看到:

(1) DDSTGCN 在所有数据集上的每个 horizon 都保持着最好的性能,说明了模型的有效性。

(2) 基于深度学习的方法普遍优于基于传统学的方法 (HA、VAR、SVR),说明对于预测交通流问题,深度学习的方法更加有效。

(3) 在深度学习的方法中,同时对时空建模的方

表2 模型预测精度对比

Table 2 Comparison of model prediction accuracy

数据集	模型	@ Horizon3			@ Horizon6			@ Horizon12		
		MAE	RMSE	MAPE/%	MAE	RMSE	MAPE%	MAE	RMSE	MAPE/%
PEMS04	HA	28.92	42.69	20.31	33.73	49.37	24.01	46.97	67.43	35.11
	VAR	21.94	34.30	16.42	23.72	36.58	18.02	26.76	40.28	20.94
	SVR	22.52	35.30	14.71	27.63	42.23	18.29	37.86	56.01	26.72
	FC-LSTM	21.42	33.37	15.32	25.83	39.10	20.35	36.41	50.73	29.92
	DCRNN	20.34	31.94	13.65	23.21	36.15	15.70	29.24	44.81	20.09
	STGCN	19.35	30.76	12.81	21.85	34.43	14.13	26.97	41.11	16.84
	Graph WaveNet	18.15	29.24	12.27	19.12	30.62	13.28	20.69	33.02	14.11
	ASTGCN	20.15	31.43	14.03	22.09	34.34	15.47	26.03	40.02	19.17
	STSGCN	19.41	30.69	12.82	21.83	34.33	14.54	26.26	40.11	14.71
	MTGNN	18.22	30.13	12.47	19.27	32.21	13.09	20.93	34.49	14.02
PEMS08	GMAN	18.28	29.32	12.35	18.75	30.77	12.96	19.95	<b>30.21</b>	12.97
	DGCRN	18.27	28.97	12.36	19.39	30.86	13.42	21.09	33.59	14.94
	DDSTGCN	17.48*	28.76*	11.77*	18.25*	30.25*	12.21*	19.42*	32.16	12.89*
	HA	23.52	34.96	14.72	27.67	40.89	17.37	39.28	56.74	25.17
	VAR	19.52	29.73	12.54	22.25	33.30	14.23	26.17	38.97	17.32
	SVR	17.93	27.69	10.95	22.41	34.53	13.97	32.11	47.03	20.99
	FC-LSTM	17.38	26.27	12.63	21.22	31.97	17.32	30.69	43.96	25.72
	DCRNN	15.64	25.48	10.04	17.88	27.63	11.38	22.51	34.21	14.17
	STGCN	15.30	25.03	9.88	17.69	27.27	11.03	25.46	33.71	13.34
	Graph WaveNet	14.02	22.76	8.95	15.24	24.22	9.57	16.67	26.77	10.86
PEMS08	ASTGCN	16.48	25.09	11.03	18.66	28.17	12.23	22.83	33.68	15.24
	STSGCN	15.45	24.39	10.22	16.93	26.53	10.84	19.50	30.43	12.27
	MTGNN	14.24	22.43	9.02	15.30	24.32	9.58	16.85	26.93	10.57
	GMAN	13.80	22.88	9.41	14.62	24.02	9.57	15.72	26.00	10.56
	DGCRN	13.89	22.07	9.19	14.92	23.99	9.85	16.73	26.88	10.84
	DDSTGCN	13.39*	21.72*	8.68*	14.25*	23.67*	9.22*	15.54*	25.99*	10.15*

注: \* 表示最优性能。

法优于只对时间依赖进行建模的方法(FC-LSTM),说明要对时间和空间同时进行建模。

#### 4.5 消融实验

为证明模型框架 STDF 和时间模型、空间模型的有效性,本文中分以下三个方面进行了消融实验。

设计了一个不带 STDF 的模型,在每个时空块,它去除掉遗忘门和更新门,转而直接将时间模型和空间模型进行连接,命名为 W/o dec;将时空模块中的时间模型和空间模块顺序进行置换,以验证它们是否可替换,命名为 switch。

设计模型 W/o gate 去除了模型的遗忘门, W/o res 去除了模型的残差连接。

设计 W/o apt 用预先定义的静态图替换动态图(消除了动态图学习模块),设计 W/o timeRes 去除了时间模型中的残差块,设计 W/o GRU 去除了时间模型中的 GRU 模块,设计 W/o MSA 去除了时间模型中的多头自注意层, W/o AR 去掉了预测门中的自回归策略,而是直接基于隐藏状态来一次多步预测。

将这些模型在 PEMS08 进行实验,结果如表 3 所示,可以看出:

在框架结构方面,DDSTGCN 比起未使用 STDF 的 W/o dec 有着显著的优势,这证明 STDF 的有效性。switch 表明转换时间模块和空间模块也没有显著的不同。W/o gate 和 W/o res 表明遗忘门和残差连接对于 STDF 至关重要。

在重要组件方面, W/o apt 表明动态图学习模块对于空间模块扩散图卷积的有效性。W/o time Res 表明本文所设计的残差块对于交通预测的有效性, W/o GRU 和 W/o MSA 表明短期和长期相关性对于预测交通流至关重要, W/o AR 表明自回归预测方法更加契合本文的模型。

## 5 结论

通过实验证明后可以得到以下结论。

(1) 首先提出了一种新的时空框架 STDF,将时间模块和空间模块进行更加合理地连接,从而更好地捕捉交通信号的时空相关性。

(2) 在此基础上,更加细致地设计了时间模块和空间模块,在时间模块中对长短期时间序列进行了特征学习,在空间模块中基于扩散过程提出了一个扩散卷积网络。

表3 基于 PEMS08 的消融实验  
Table 3 Ablation study on PEMS08

模型	@ Horizon3			@ Horizon6			@ Horizon12		
	MAE	RMSE	MAPE/%	MAE	RMSE	MAPE/%	MAE	RMSE	MAPE/%
DDSTGCN	<b>13.39</b>	<b>21.72</b>	<b>8.68</b>	<b>14.25</b>	<b>23.67</b>	<b>9.22</b>	15.54	<b>25.99</b>	<b>10.15</b>
switch	13.40	21.74	8.68	14.26	23.71	9.27	<b>15.53</b>	26.03	10.20
W/o dec	14.15	22.41	9.51	15.32	25.06	10.86	16.89	27.36	11.42
W/o gate	13.85	22.21	9.01	14.83	24.06	9.66	16.54	26.79	10.96
W/o res	13.59	22.14	9.05	14.52	24.42	9.74	15.93	26.71	10.62
W/o apt	13.51	22.17	9.31	14.39	24.21	9.76	15.97	26.53	10.46
W/o timeRes	13.29	21.84	8.85	14.39	24.22	9.54	15.77	26.60	10.50
W/o GRU	14.11	22.21	8.96	14.93	24.14	9.82	16.27	26.53	10.53
W/o MSA	13.80	22.14	8.91	14.51	23.83	9.41	15.64	25.81	10.21
W/o ar	14.31	22.47	8.92	15.39	24.47	9.72	16.44	26.72	10.56

(3) 根据现实生活的周期性和动态性,引入了周期特征,并且设计了一个动态图学习模块,从交通系统的宏观和微观角度进行图学习,从而动态调整道路网络的空间依赖性。

(4) 在 PEMS04 和 PEMS08 的数据集上进行的大量实验表明,本文设计的模型起到了良好的性能表现。

#### 参 考 文 献

- [1] 陈悦, 杨柳, 李帅, 等. 基于 Softmax 函数增强卷积神经网络-双向长短期记忆网络框架的交通拥堵预测算法[J]. 科学技术与工程, 2022, 22(29): 12917-12926.  
Chen Yue, Yang Liu, Li Shuai, et al. Traffic congestion prediction algorithm based on Softmax function enhanced convolutional neural network and bidirectional long short term memory network framework [J]. Science Technology and Engineering, 2022, 22(29): 12917-12926.
- [2] 方方, 王昕. 基于小波分析和集成学习的短时交通流预测[J]. 科学技术与工程, 2022, 22(1): 383-392.  
Fang Fang, Wang Xin. Short-term traffic flow prediction based on wavelet analysis and ensemble learning [J]. Science Technology and Engineering, 2022, 22(1): 383-392.
- [3] 代亮, 梅洋, 钱超, 等. 基于深度学习的短时交通量预测研究综述[J]. 计算机科学, 2019, 46(3): 39-47.  
Dai Liang, Mei Yang, Qian Chao, et al. A review of short-term traffic volume prediction research based on deep learning [J]. Computer Science, 2019, 46(3): 39-47.
- [4] Williams B M, Hoel L A. Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: theoretical basis and empirical results[J]. Journal of Transportation Engineering, 2003, 129(6): 664-672.
- [5] Xie Y, Zhang Y, Ye Z. Short-term traffic volume forecasting using Kalman filter with discrete wavelet decomposition[J]. Computer-Aided Civil and Infrastructure Engineering, 2007, 22(5): 326-334.
- [6] Qiao S, Sun R, Fan G, et al. Short-term traffic flow forecast based on parallel long short-term memory neural network[C]//2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS). New York: IEEE, 2017: 253-257.
- [7] 任艺柯. 基于改进的 LSTM 网络的交通流预测[D]. 大连: 大连理工大学, 2019.  
Ren Yike. Traffic flow prediction based on improved LSTM network [D]. Dalian: Dalian University of Technology, 2019.
- [8] Arif M, Wang G, Chen S. Deep learning with non-parametric regression model for traffic flow prediction[C]//2018 5th International Conference on Information Science and Control Engineering (ICISCE). New York: IEEE, 2018: 6-10.
- [9] Koesdwiady A, Souza R, Karray F. Improving traffic flow prediction with weather information in connected cars: a deep learning approach[J]. IEEE Transactions on Vehicular Technology, 2016, 65(12): 9508-9517.
- [10] Yu B, Yin H, Zhu Z. Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting[J]. arXiv preprint arXiv: 1709.04875, 2017.
- [11] Li Y, Yu R, Shahabi C, et al. Diffusion convolutional recurrent neural network: data-driven traffic forecasting[J]. arXiv preprint arXiv: 1707.01926, 2017.
- [12] Yao H, Tang X, Wei H, et al. Modeling spatial-temporal dynamics for traffic prediction [J]. DOI: 10.48550/arXiv.1803.01254, 2018.
- [13] Wu Z, Pan S, Long G, et al. Graph wavenet for deep spatial-temporal graph modeling [J]. arXiv preprint arXiv: 1906.00121, 2019.
- [14] Guo S, Lin Y, Feng N, et al. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2019, 33(1): 922-929.
- [15] Shao Z, Zhang Z, Wei W, et al. Decoupled dynamic spatial-temporal graph neural network for traffic forecasting[J]. arXiv preprint arXiv: 2206.09112, 2022.
- [16] Sun J, Li J, Wu C, et al. Ada-STNet: A dynamic AdaBoost spatio-temporal network for traffic flow prediction[C]//ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). New York: IEEE, 2022: 5478-5482.
- [17] Das A, Kong W, Leach A, et al. Long-term forecasting with TiDE: time-series dense encoder[J]. arXiv preprint arXiv: 2304.08424, 2023.
- [18] Wu Z, Pan S, Chen F, et al. A comprehensive survey on graph neural networks[J]. IEEE Transactions on Neural Networks and

- Learning Systems, 2020, 32(1): 4-24.
- [19] Liu J, Guan W. A summary of traffic flow forecasting methods [J]. Journal of Highway and Transportation Research and Development, 2004, 21(3): 82-85.
- [20] Holden K. Vector auto regression modeling and forecasting [J]. Journal of Forecasting, 1995, 14(3): 159-166.
- [21] Smola A J, Schölkopf B. A tutorial on support vector regression [J]. Statistics and Computing, 2004, 14: 199-222.
- [22] Sainath T N, Vinyals O, Senior A, et al. Convolutional, long short-term memory, fully connected deep neural networks [C]// 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). New York: IEEE, 2015: 4580-4584.
- [23] Song C, Lin Y, Guo S, et al. Spatial-temporal synchronous graph convolutional networks: a new framework for spatial-temporal network data forecasting [J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(1): 914-921.
- [24] Zheng C, Fan X, Wang C, et al. Gman: A graph multi-attention network for traffic prediction [J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(1): 1234-1241.
- [25] Wu Z, Pan S, Long G, et al. Connecting the dots: multivariate time series forecasting with graph neural networks [C]// Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM, 2020: 753-763.
- [26] Li F, Feng J, Yan H, et al. Dynamic graph convolutional recurrent network for traffic prediction: benchmark and solution [J]. ACM Transactions on Knowledge Discovery from Data, 2023, 17(1): 1-21.