

电子科技大学

UNIVERSITY OF ELECTRONIC SCIENCE AND TECHNOLOGY OF CHINA

硕士学位论文

MASTER THESIS



论文题目 基于信息融合的目标聚集
预测算法研究

学科专业 计算机科学与技术

学 号 201921080327

作者姓名 袁铭潮

指导老师 田玲 教授

学 院 计算机科学与工程学院

分类号 _____ 密级 _____

UDC 注¹ _____

学 位 论 文

基于信息融合的目标聚集 预测算法研究

(题名和副题名)

袁铭潮

(作者姓名)

指导老师 _____ 田玲 教授

电子科技大学 成都

(姓名、职称、单位名称)

申请学位级别 _____ 硕士 学科专业 _____ 计算机科学与技术

提交论文日期 2022 年 3 月 18 日 论文答辩日期 2022 年 5 月 17 日

学位授予单位和日期 电子科技大学 2022 年 6 月 8 日

答辩委员会主席 刘峤

评阅人

注 1: 注明《国际十进分类法 UDC》的类号。

Research on Target Gather Prediction Algorithm based on Information Fusion

A Master Thesis Submitted to
University of Electronic Science and Technology of China

Discipline: **Computer Science and Technology**

Student ID: **201921080327**

Author: **Mingchao Yuan**

Supervisor: **Prof. Ling Tian**

School: **School of Computer Science and
Engineering**

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

作者签名: _____ 日期: 年 月 日

论文使用授权

本学位论文作者完全了解电子科技大学有关保留、使用学位论文的规定，有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人授权电子科技大学可以将学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

(保密的学位论文在解密后应遵守此规定)

作者签名: _____ 导师签名: _____

日期: 年 月 日

摘要

聚集预测是对城市中可能存在的人群聚集现象进行预测，能够帮助政府更好地应对城市中的聚集问题，增强各部门的应对能力，大幅提升城市治理水平。城市背景下生活规律性强，道路结构复杂，地理环境、假期、天气等因素对用户行为模式影响较大。且多目标交互复杂，需要同时考虑行人、车辆、公共交通等。现有方法存在对外部知识考虑不足、多用户关联关系刻画不准的问题，导致模型预测精度不高。

基于上述问题，本文围绕基于信息融合的目标聚集预测算法开展研究，具体研究内容如下：

1、提出了基于时空门控循环单元（STGRU）的轨迹预测模型。现有的算法模型忽略了地理环境信息、假期信息、天气信息等外部知识对运动模式的影响。为此，本文提出了融合路网门控结构，实现了对以路网结构为代表的地理环境特征提取和外部知识的融合，从而实现更准确的用户行为模式建模，并在一定程度上减少模型参数量。通过实验表明，考虑路网结构和外部知识可以有效提高轨迹预测性能，相较于现有方法的性能提升最高可达 18.1%。

2、提出基于空间特征增强的聚集预测模型。当前的时空特征挖掘算法不能同时建模多用户之间关联特征，本文提出了基于希尔伯特曲线与图嵌入的空间特征增强方法，通过使用希尔伯特曲线进行标签设计增强了标签间的空间关系，同时通过融合图嵌入方法实现了多用户场景下的实时路网结构特征提取。通过实验表明，能够有效提升模型对多用户时空特征的建模能力，性能提升幅度最高为 22.6%。

3、完成基于城市流量的聚集预测告警系统。为了验证算法的实用性，结合实际需求，设计并实现了基于城市流量的聚集预测告警系统，实现了城市级别的多目标聚集预测，为城市的管理管控提供有利支撑。

关键词：聚集预测，复杂场景，智慧城市，深度学习，时空特征挖掘

ABSTRACT

Gather prediction is to predict the possible crowd gathering phenomenon in the city. Gather prediction can help the government deal with the gathering problems, and enhance the coping ability of various departments. Gather prediction can significantly improve the level of urban governance. The urban scene has a strong regularity of life, complex road structure, and factors such as geography, holidays, and weather have a large impact on user behavior patterns. And the multi-objective interaction is complex, and pedestrians, vehicles and public transportation need to be considered simultaneously. The existing methods have the problems of insufficient consideration of external knowledge and inaccurate portrayal of multi-user association relationships, which lead to low model prediction accuracy.

Based on the above-mentioned problems, this paper conducts research around the target gather prediction algorithm based on information fusion as follows.

1. Thesis proposes a trajectory prediction model based on spatio-temporal gated recurrent unit (STGRU). The existing trajectory prediction algorithms ignore external knowledge such as geographic environment, holiday and weather information affects the movement patterns. To this end, a fused road network gating structure is proposed to achieve the fusion of geographic environment feature extraction and external knowledges, so as to achieve more accurate user behavior pattern modeling and reduce the number of model parameters. It is shown through experiments that considering the road network structure and external knowledge can effectively improve the trajectory prediction performance, and the performance improvement is up to 18.1% compared with the existing methods.

2. Thesis proposes a gather prediction model based on spatial feature enhancement. The spatial feature enhancement method based on Hilbert curves and graph embedding is proposed to address the problems of the existing spatio-temporal feature mining algorithms in label design and explicit feature extraction methods. The model enhances the spatial relationship between labels by using Hilbert curves for label design, and realizes real-time road network structure feature extraction in multi-user scenarios by fusing graph embedding methods. It is shown through experiments that the model can effectively improve the modeling ability of multi-user spatio-temporal features with the highest performance improvement of 22.6%.

ABSTRACT

3. Complete the gather prediction alert system based on city flow. In order to verify the practicality of the algorithm, the gather prediction alert system based on city flow is designed and implemented in conjunction with the actual demand, which realizes the multi-objective gather prediction at the city level and provides favorable support for the management control of the city.

Keywords: gather prediction, complex scenario, smart city, deep learning, spatio-temporal feature mining

目 录

第一章 绪 论	1
1.1 研究工作的背景与意义	1
1.2 国内外研究历史与现状	1
1.2.1 传统位置预测方法	2
1.2.2 基于神经网络模型的位置预测方法	2
1.2.3 现有工作总结	4
1.3 本文的主要贡献与创新	5
1.4 本论文的结构安排	5
第二章 相关理论基础	7
2.1 引言	7
2.2 相关技术	7
2.2.1 分类任务及其评价指标	7
2.2.2 循环神经网络	10
2.2.3 长短时记忆网络	11
2.3 时空特征建模方法	14
2.3.1 基于时空特征的循环神经网络	14
2.3.2 分层时空长短时记忆网络	16
2.3.3 时空门控网络	16
2.3.4 时空图卷积网络	18
2.4 数据集介绍	19
2.5 本章小结	19
第三章 基于信息融合的轨迹预测算法	20
3.1 引言	20
3.2 时空门控循环网络	21
3.2.1 轨迹预测问题定义	21
3.2.2 标准门控循环单元	22
3.2.3 时空门控循环单元	23
3.2.4 模型分析与训练	25
3.3 实验与分析	26
3.3.1 实验配置和基线方法	26

3.3.2 评价指标.....	27
3.3.3 对比实验.....	27
3.3.4 消融实验.....	30
3.3.5 应用分析.....	32
3.4 本章小节.....	33
第四章 基于空间特征增强的聚集预测方法	34
4.1 引言.....	34
4.2 基于空间特征增强的聚集预测模型	35
4.2.1 聚集预测问题定义及模型架构	35
4.2.2 希尔伯特曲线的区域标签生成	37
4.2.3 路网图构建及轨迹点匹配	41
4.2.4 基于图卷积网络的路网节点嵌入.....	43
4.3 实验与分析	45
4.3.1 实验配置.....	45
4.3.2 对比实验.....	45
4.3.3 消融实验.....	46
4.3.4 应用分析.....	48
4.4 本章小结.....	50
第五章 基于城市流量的聚集预测告警系统实现.....	51
5.1 引言.....	51
5.2 系统概述.....	51
5.3 需求分析.....	51
5.4 系统实现.....	54
5.4.1 资源管理模块	54
5.4.2 数据传输和处理模块	55
5.4.3 聚集预测模块	56
5.4.4 前端交互模块	57
5.4.5 聚集预测结果展示界面	57
5.5 系统展示	57
5.6 系统测试	58
5.6.1 功能测试.....	58
5.6.2 性能测试.....	60
5.7 本章小结.....	61

目 录

第六章 总结与展望	62
6.1 全文总结	62
6.2 后续工作展望	62
致 谢	63
参考文献	64
攻读硕士学位期间取得的成果	71

第一章 绪论

1.1 研究工作的背景与意义

随着城市中物联网设备的不断增加^[1] 和 5G^[2] 等技术的普及，城市场景下能够获取到的轨迹数据呈爆发式增长。海量的轨迹数据^[3] 为基于大数据技术的机器学习、深度学习发展提供了良好的数据支持，轨迹预测^[4]、聚集预测^[5,6]、交通流量预测^[7]、热点区域分析^[8]、人口转移预测^[9] 等研究方向受到了越来越多研究者的关注，其中聚集预测受到了越来越多城市管理者的重视。近年来由人群聚集所带来的问题对城市管理影响重大。首先是新型冠状病毒在全球的传播和蔓延^[10]，新冠疫情的传播和暴发很大程度上依赖于人群聚集。如今我国已经能够较好地控制新型冠状病毒疫情的传播，但仍需时刻保持警惕，避免聚集的发生，特别是节假日或大型活动的举办。其次，人群聚集极为容易发生踩踏伤人等事故，以及某些恶性事件^[11] 的发生，需要及时进行疏散。

当前聚集预测算法通常对用户轨迹预测结果进行融合得到聚集预测结果，不能很好地提取用户关联特征^[12]，在多轨迹任务^[13] 上仍存在较大局限性。一方面，现有的模型在提取时空特征时未考虑地理环境信息^[14]、假期信息、天气信息等外部知识^[15] 的影响。另一方面，当前方法在对用户轨迹进行预测时主要针对单一轨迹进行设计，建模单用户的行为模式^[16-19]，并未考虑用户关联特征。

为此，本文围绕基于时空轨迹预测的聚集预测算法进行研究，首先设计结合路网结构的时空门控网络，通过提取路网结构特征和融合多种外部知识的方法，提升了对时空特征的建模能力。其次通过空间特征增强方法，提取用户之间的关联特征，提升模型在多用户场景下的行为模式建模能力。并在多个数据集上设置和进行了多项实验，验证了模型的可行性和预测性能。

本文的研究能够根据城市中的用户分布情况以及采集的用户历史轨迹信息对每个用户的位置进行预测，对每个用户的位置预测结果通过统计的方法得到用户之间的聚集情况，聚集预测结果可以帮助城市管理部门提前做出反应，并大大减少恶性聚集造成的损失和风险。

1.2 国内外研究历史与现状

聚集预测是时空数据分析^[20] 和挖掘中^[21] 的重要组成部分，它包括很多应用场景，如热点区域分析、客流预测^[22]、人口转移预测等。Tomaharu 等人^[23] 提出了一个集体图模型来预测区域间的转移人口，Verma 等人^[24] 利用轨迹数据挖掘热

点,实现了非实时的粗粒度聚集预测。Ni 等人^[25]通过客流预测实现了城市间的聚集预测。Kumar 等人^[26]利用轨迹聚类^[16,27]和相似性分析进行聚集预测。聚集预测也可以转化为同一时间和空间下的多轨迹预测任务,即关注多个轨迹之间的时间和空间特征。时空数据的数据挖掘^[28]非常困难,也是目前的研究热点之一。

本节先介绍以马尔可夫链为主的传统位置预测算法,再对基于神经网络模型的位置预测方法进行阐述,最后对现有工作进行总结。

1.2.1 传统位置预测方法

位置预测的主要特征是地理信息和时间信息。位置预测还可以使用位置语义^[29]、速度、方向。在传统概率的基础上,矩阵分解法用低秩矩阵去组合矩阵,得到用户和轨迹的隐含特征向量。张量分解法扩展到三个维度,包括用户、时间信息和空间信息。

经典基于序列数据的预测模型通常采用马尔可夫链模型^[30]。马尔可夫链(Markov Chain, MC),具有离散时间的特性,可以表示为状态空间中多个状态之间转换的随机过程,且该过程具有独立性,即任一状态的概率分布只取决于前一个状态。根据下一时刻状态由当前时刻和前 $m - 1$ 个时刻决定,称为 m 阶马尔科夫链。基本马尔科夫链模型基于转移矩阵,通过上一时刻的状态来预测下一时刻的状态,比较适合建模个人的序列。但是比较容易受到最近的噪声影响^[31]。

在 MC 的基础上, FPMC^[32] 对每个用户建模了一个马尔可夫转移矩阵,组成了一个转移张量,来融合序列和个性化两个方面的信息,称为个性化马尔可夫链,通过成对交互的张量分解来解决数据稀疏的问题。对用户、当前状态、下一时刻状态之间的成对关系设置两方面的嵌入表示,通过点积的和来估算,降低了计算复杂度且参数量更少,同时显示建模成对的交互关系更为直接。但是 FPMC 中将每一个状态进行了两次嵌入,在轨迹预测中,将两个状态嵌入到两个相互独立的空间中是不恰当的,不能很好地捕捉两者转化的关系, FPMC 存在一些变种模型^[18],但都没有从根本上解决这个问题。

根据位置预测任务在 FPMC 的基础上进行了改进,提出了 PRME 模型^[33],将状态统一嵌入到一个欧式的度量空间中,使用向量间的欧式距离作为相似度的衡量方式。后续还有一部分基于图嵌入^[34]的工作,将其扩展到非欧式空间中。

1.2.2 基于神经网络模型的位置预测方法

随着近几年机器学习和深度学习的发展,部分任务需要处理序列数据,循环神经网络在序列处理上取得了巨大的提升,成为时间序列预测最好的方法。RNN 以及其变种模型 LSTM、GRU 的发展,使得神经网络模型在序列预测任务上保持

最好的效果。

之后的模型发展更专注于对时空特征的建模，更多考虑轨迹的上下文信息^[35,36] 和轨迹点之间间隔距离等信息，基于基础模型发展出了 STF-RNN、ST-RNN、HST-LSTM、STGN 等一系列效果更好或更符合实际情况的模型。

1.2.2.1 基于时间序列的神经网络模型

随着神经网络领域研究的深入，神经网络模型在大多数任务上取得了较好的效果。最传统的神经网络模型是前馈神经网络，通过层与层之间的全连接对数据进行拟合，这样的结构也导致了传统的神经网络模型对很多问题是无法解决的，最典型的就是序列数据。神经网络的优势在于能够对多维度信息之间的非线性关系进行建模，广泛应用于推荐领域^[37-40]。

为了能够处理序列数据又能拥有神经网络模型的建模能力，Wojciech 等人提出了循环神经网络（Recurrent neural network，RNN）^[41]。RNN 与前馈神经网络的区别在于对历史信息的记忆和计算。通过连接隐藏层中的相邻节点构建了重复模块链，借助上一时刻的输出计算当前状态。RNN 在位置预测任务上，一度是效果最好的模型。

但由于 RNN 本身结构的限制，为了适应现有的使用现状，一般都仅考虑当前状态只与前面的几个状态相关，这与马尔科夫链模型的本质上十分相近，仅是借助于神经网络强大的建模能力，但是 RNN 并未考虑到反向传播时的累乘问题，存在严重的梯度消失和梯度爆炸问题。

现实场景中序列数据的状态取决于其上下文状态以及较远的历史状态，而 RNN 对于历史状态并不能很好地建模。长短时记忆网络（Long Short-term Memory，LSTM）^[42] 通过结构优化实现了对长间隔事件的处理和预测。LSTM 相较于 RNN 最大的进步在于对长期依赖性的建模，长时间记住信息实际上是 LSTM 的默认行为，LSTM 精心设计了门控结构来选择性的添加或移除信息，同时大幅缓解了梯度爆炸和梯度消失问题。

门控循环单元（Gate Recurrent Unit，GRU）^[43] 是一种 LSTM 优化模型。GRU 在保留了 LSTM 结构的优点的前提下对结构进行了精简，较 LSTM 减少了一个门控，参数比 LSTM 少，更便于计算和训练。

LSTM 和 GRU 是目前通用序列数据处理建模最为先进的范式模型，但对于位置预测任务来说，还有很多不足之处能够进一步改进。例如 LSTM 并不能建模轨迹数据中的空间关系。

1.2.2.2 建模时空特征的神经网络模型

这部分工作是在 RNN 和 LSTM 的基础上增加了空间依赖性建模^[44]的方式，更适合用于位置预测等数据间存在强烈空间关系的任务。

针对 RNN 难以建模时间间隔信息以及位置之间的地理距离的问题，Liu 等人^[19]提出了时空循环神经网络 STRNN，分别设计了特定时间间隔的过渡矩阵和地理距离的过渡矩阵。在一定程度上对轨迹的时空信息进行了建模，但是较为生硬，特征之间不能很好地融合，且 STRNN 中的时空特征提取需要经过手工处理。基于时空特征循环神经网络 STF-RNN^[45]能够自动提取空间和时间特征内部表示，而不是依靠手工表示。这使模型能够更有效地发现有关人的行为的有用知识，保留了 RNN 结构来建模用户移动历史。相较于 ST-RNN 中过渡矩阵的方法，STF-RNN 能够根据数据本身特点学习到不同的时间和空间特征。

Geo 主题模型^[46]采样基于主体和用户与历史位置之间的距离。GMEL 模型^[47]结合位置语义来嵌入地理环境信息。上述研究表明，连续轨迹点之间的顺序在位置预测中起着至关重要的作用，在强实时轨迹数据场景下效果更为显著，因为人类行为模式^[48,49]是有顺序的。

ST-LSTM 模型^[50]同时考虑轨迹点之间的时间间隔和距离间隔，通过将时间间隔和地理距离划分为离散时隙并仅对这些时隙的上限和下限进行编码来解决间隔的稀疏性问题，将其单独进行建模并添加到经典 LSTM 模型每个门控中。相当于在原始输入的基础上添加了时间和距离的手工处理，但是并没有使用单独的门控结构进行显式特征提取。STGN 模型^[51]通过在 LSTM 中引入时空门控以捕获连续轨迹点之间的时空关系，设计了两对时间门控和距离门控，对时间间隔和距离间隔单独进行建模，而不是像 ST-LSTM 直接添加到输入门、输出门、遗忘门之中，能够对时间和距离更好地建模。STGN 是目前效果最好的模型，但同样存在一定的问题，过多地考虑相邻轨迹点之间的时间间隔和地理距离，并不能很好地与 LSTM 本质上长期记忆的轨迹点与当前轨迹点之间的关系，对时空关系的建模并不完善，且额外对当前轨迹点的建模进行差异的记忆处理，导致参数量成倍地增加。

1.2.3 现有工作总结

在位置预测相关领域中，传统方法通过求解状态空间中的转移概率建模数据，建模能力较弱且对位置预测任务中至关重要的语义特征难以融合。同时传统方法对噪声较为敏感，因此数据质量要求很高，不适用于大多数实际应用场景。

基于神经网络的方法能够凭借其技术特点大幅提升对时空数据的建模能力，但通用的基于时间序列的神经网络模型在时空特征建模时受到了较大的限制，难

以建模数据中的空间特征。

建模时空特征的神经网络模型在时间序列神经网络模型的基础上增加了对空间信息的建模，并增加了时空特征的权重得到更强的建模能力。目前这类方法已经能够实现时空特征的自动提取，并解决了固有的短期记忆问题，但带来了模型参数量的成倍增长。同时现有的方法中仅通过各点间距离信息建模空间特征，忽视了固有地理环境及其状态的影响。

1.3 本文的主要贡献与创新

聚集预测任务的主要研究难点在于对轨迹数据中时空特征的建模以及多用户之间关联特征的建模。时空特征建模主要通过扩展各类时间序列神经网络实现。然而目前的各类方法均忽略了路网等地理环境信息以及天气信息、假期信息等外部知识对用户行为模式的影响，且为了对时空特征进行建模导致参数量的成倍增长。

针对上述问题，本文主要研究模型在聚集预测场景下的时空特征提取建模能力，主要创新点与贡献如下：

1、针对现有的算法模型忽略了地理环境信息、假期信息、天气信息等外部知识对运动模式的影响问题，本文提出了时空门控循环单元（Spatio-Temporal Gated Recurrent Unit, STGRU），实现了对以路网结构为代表的地理环境特征提取和外部知识的融合，从而实现更准确的用户行为模式建模，并在一定程度上减少模型参数量。通过实验表明，考虑路网结构和外部知识可以有效提高轨迹预测性能，相较于现有方法的性能提升最高可达 18.1%。

2、针对当前的时空特征挖掘算法不能同时建模多用户之间关联特征的问题，本文提出了基于空间特征增强的的聚集预测模型（Spatial Enhanced Gather Prediction Model, SEGPM），通过使用希尔伯特曲线进行标签设计增强了标签间的空间关系，同时通过融合图嵌入方法实现了多用户场景下的实时路网结构特征提取。通过实验表明，能够有效提升模型对多用户时空特征的建模能力，性能提升幅度最高为 22.6%。

3、完成基于城市流量的聚集预测告警系统。为了验证算法的实用性，结合实际需求，设计并实现了基于城市流量的聚集预测告警系统，实现了城市级别的多目标聚集预测，为城市的管理管控提供有力支撑。

1.4 本论文的结构安排

本文的后续章节结构安排如下：

第二章，相关理论基础。首先介绍了本文中涉及的各类深度学习算法模型的基础知识，再介绍了近年来的时空特征建模相关算法，最后介绍了本文对数据集的选择及所使用的轨迹数据集的整体情况。

第三章，基于时空特征的轨迹预测方法。首先根据当前轨迹预测方法存在的问题提出解决思路。然后详细介绍了本文中所提出的时空门控循环单元 (STGRU) 的结构与原理。最后，通过实验验证所提出模型建模时空特征的有效性。

第四章，基于希尔伯特曲线与图嵌入的空间特征增强方法研究。首先分析轨迹预测方法在聚集预测任务下的局限性，并详细介绍了希尔伯特曲线和图卷积算法的原理。然后详细阐述了增强轨迹数据空间特征的聚集预测模型。最后，并通过实验验证该方法对于聚集预测的有效性。

第五章，基于城市流量的聚集预测告警系统实现。结合本文研究内容和实际场景，设计并实现了一个聚集预测告警系统。并从需求分析、功能设计实现并通过可视化 demo 对系统进行展示。

第六章，总结与展望。对本文所做的工作及存在的缺点进行了总结，并构思了后续的研究可能。

第二章 相关理论基础

本章主要介绍后文中所用到的序列神经网络相关理论基础以及三种基于深度学习的时空特征建模方法，并对本文所使用的数据集进行了介绍。

2.1 引言

时空数据是人类活动中产生和采集到的最多的数据，从时空数据中发现知识对于解决现实世界中的诸多问题有很大的意义。随着深度学习算法的发展，越来越多的研究者开始关注基于时空数据的应用和挖掘。其中，最核心的问题就是如何对时空特征进行建模分析，近年来在不同应用领域都针对该问题开展了研究。

本章首先介绍本文提出模型所涉及的序列神经网络相关理论基础，然后阐述了时空特征建模方法的发展，最后对本文中所有使用的数据集进行分析。

2.2 相关技术

本文所涉及的相关技术面较广，为了方便理解，本节对本文所提出模型相关的技术要点进行详细介绍。

2.2.1 分类任务及其评价指标

与兴趣点推荐 (Point-of-Interest recommendation) 任务类似，本文将聚集预测任务等价为用户的区域预测，也就是无标签的多分类任务。本小节对分类任务的定义及相关评价指标进行介绍。

2.2.1.1 分类任务

在机器学习和深度学习的各项任务主要分为分类任务和回归任务。其中分类任务主要对离散值进行预测，输出结果为是、否或所属类别；回归任务则对连续值进行预测，得到的输出为确定数值。

分类就是将要预测的数据进行建模，针对给定的输入数据预测其所属类别。从模型角度看，所学习的就是如何将输入数据映射到特定类别。因此对所有类别需要将其映射为数值，如较为常用的 one-hot 标签、数值标签等，为每个类别分配唯一的对应数值或表示方式。对类别标签的设置目前没有通用的理论支撑，不同的标签映射会影响模型性能，本文第四章通过设计类别标签映射方式取得了一定的性能提升。

分类任务主要分为二分类任务和多分类任务，某些特殊场景下还会用到多标

签分类任务、样本不平衡分类任务。

二分类任务是指只有两个类别标签的分类任务，通常视为正类和负类。如常见的垃圾邮件检测，正类为非垃圾邮件，负类为垃圾邮件。二分类通常将类别标签设置为 0 和 1，使用每个样本的伯努利概率分布模型进行建模。输出结果则为样本对应的每个类别的概率，所属类别通过预测概率判断。

多分类任务则具有两个以上类别标签，多分类中类别之间并非对立关系。多分类任务同样是预测每个类别的范畴分布概率对多分类任务进行建模。

多分类任务的类别标签设置通常采用连续数值或 one-hot 表示。其中 one-hot 使用一对映射，在评价指标计算时更加方便，适用于类别数量较少的情况，当类别数量较多时对计算资源和内存资源要求较高；连续数值则适用于类别数量较多的情况，能够较大程度节省资源要求，但数值之间的关系容易对建模结果造成一定影响。

在深度学习中，常常使用 Logistic 回归^[52]作为最后一层进行二分类，多分类则使用 softmax 回归。

虽然名字中包含回归，但是 Logistic 回归是一种分类模型。Logistic 回归的假设为给定数据集线性可分，对于 n 个随机变量 $x = (x_1, x_2, \dots, x_n)$ ，于是决策边界可以表示为 $w_1x_1 + w_2x_2 + \dots + w_nx_n + b = 0$ ，根据输入 x 计算出 y ，点 (x, y) 落在分界线之上，则判定类别为 1，反之为 0。

Logistic 回归需要找到分类概率 $P(Y = 1)$ 与输入向量 x 的直接关系，然后通过根据概率值来判断类别。为了能够拟合离散变量并保持可微，使用 sigmoid 函数进行拟合：

$$y = \frac{1}{1 + e^{-(w^\top x + b)}} \quad (2-1)$$

$$\ln \frac{y}{1 - y} = w^\top x + b \quad (2-2)$$

条件分布概率计算为：

$$\ln \frac{P(Y = 1|x)}{1 - P(Y = 1|x)} = w \cdot x \quad (2-3)$$

$$P(Y = 1|x) = \frac{\exp(w \cdot x)}{1 + \exp(w \cdot x)} \quad (2-4)$$

$$P(Y = 0|x) = \frac{1}{1 + \exp(w \cdot x)} \quad (2-5)$$

Softmax 回归是在 Logistic 回归的基础上将离散型随机变量 Y 的取值集合推广到 $\{1, 2, \dots, k\}$ ，使用 softmax 函数计算输出，多分类 softmax 函数的分布概率计算公式为：

$$P(Y = k|x) = \frac{\exp(w_k \cdot x)}{1 + \sum_{k=1}^{K-1} \exp(w_k \cdot x)}, d = 1, 2, \dots, K-1 \quad (2-6)$$

$$P(Y = K|x) = \frac{1}{1 + \sum_{k=1}^{K-1} \exp(w_k \cdot x)} \quad (2-7)$$

2.2.1.2 常用评价指标

分类任务常用的评价指标为准确率、精确率、召回率、F1 分数、ROC、AUC 等，针对不同任务常使用不同的评价指标，本文的主要评价指标为多分类^[53] 召回率 $recall@k$ 和 AUC。

为了能够更好的介绍评价指标的定义，先对于二分类任务中常用的 $recall$ 指标进行介绍。

在二分类任务中所有样本存在四种情况：真正例 (TP) 为模型预测为正的正样本；假正例 (FP) 为模型预测为正的负样本；假负例 (FN) 为模型预测为负的正样本；真负例 (TN) 为模型预测为负的负样本。

最为直观的评价指标为准确率，表示预测正确的样本占总样本的百分比：

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2-8)$$

但在样本数据的类别不均衡的情况下，占比比较大的类别对准确率的影响更大，此时准确率结果并不能起到评价分类器好坏的作用。

在准确率的基础上，精确率计算的是自所有被预测为正的样本中实际为正样本的概率，比较适合进行挑选的任务。

$$Precision = \frac{TP}{TP + FP} \quad (2-9)$$

召回率 $recall$ 则针对所有样本，计算样本的所有正例中有多少被正确预测。

$$Recall = \frac{TP}{TP + FN} \quad (2-10)$$

由于本文的任务中区域之间并没有直接关联，模型的出发点为尽可能准确预测在目标区域的用户，因此使用召回率更为合适。虽然 F-1 分数同时兼顾了精确率和召回率，但在本文场景中的适用性不如召回率。

而在多分类任务中召回率 $recall$ 的计算相对较为复杂，首先需要根据所有预

测结果计算每个类别的对应的 $recall$ 值为 $(recall_0, recall_1, \dots, recall_n)$ 。同时为了能够评估模型的整体性能，需要对所有类别进行综合考虑。Macro-recall 指标直接对所有 $recall$ 值求平均，但会在一定程度上受到数据集分布的影响。Micro-recall 指标则是对数据集中的所有样本计算全局混淆矩阵，虽然 Micro-recall 会在一定程度上受到不均衡类别影响，但更符合聚集预测任务中聚集区域样本数量更多的诉求。

$$Recall_{micro} = \frac{\bar{TP}}{\bar{TP} + \bar{FN}} = \frac{\sum_{i=1}^n TP_i}{\sum_{i=1}^n TP_i + \sum_{i=1}^n FN_i} \quad (2-11)$$

由于分类数量较多，为了能够更好地衡量模型性能，本文考虑使用 $recall@k$ ，即当被预测样本的标签被预测的前 k 个结果命中时认为其被正确预测了。能够根据不同的 k 值对模型性能进行多维度的衡量。

同时为了能够更好地描述模型整体性能，本文使用 AUC 指标（Area under the Curve of ROC）^[54] 作为辅助，AUC 表示每一类别中正例排在负例前面的概率，在大多数情况下能够较好的衡量模型的整体建模性能。由于其基于 ROC 曲线进行计算，就不再次展开介绍。

2.2.2 循环神经网络

2012 年，AlexNet 模型^[55] 在计算机视觉领域首次达到了人类的精度，之后随着各领域数据的增加极大的推动了深度学习与神经网络的发展。其中，序列数据的建模在各领域都得到了广泛地研究。

时间序列数据^[56] 是指拥有不同时间戳的数据，根据时间的天然连续性这类数据能够反映出事物或现象随时间变化的过程。前馈神经网络^[57] 会给每个输入维度分配一个单独的参数，在每个时间点都是一个单独的参数，对于变长序列泛化能力极差，且难以实现不同位置的参数共享。循环神经网络（RNN）通过设计循环计算结构实现了不同方式的参数共享，能够适用于各类序列数据的建模，并成为了应用最广泛的神经网络模型之一。

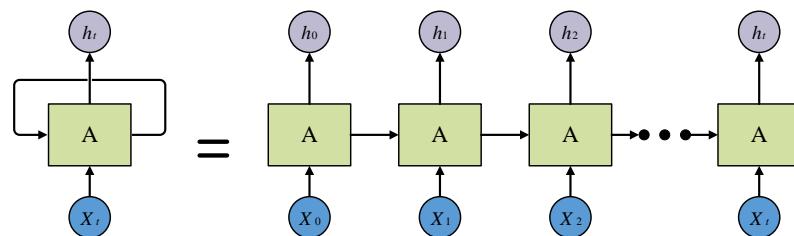


图 2-1 循环神经网络 RNN

如图2-1所示为 RNN 的网络结构，由输入层、隐藏层和输出层构成。其中最基

本的单层网络结构的定义为输入 x ，经过变换和激活函数 f 得到输出 y ：

$$y = f(x; \theta) \quad (2-12)$$

而对于序列数据 (x_1, x_2, \dots, x_n) ，在 t 时刻时所包含向量为 $x^{(t)}$ 。为了能够实现不同时间步之间能够进行循环计算，RNN 引入了隐藏状态 h 的概念：将前一次的输出结果作为下一次的输入一起训练。

$$h^{(t)} = f(h^{(t-1)}, x^{(t)}; \theta) \quad (2-13)$$

循环结构的引入使得循环神经网络能够对不同长度的序列都具有相同的输入大小，以及在每个时间步使用相同的参数，而不需要为每个时间步学习独立的模型。根据使用场景的不同，循环神经网络能够实现单一输入多输出、多输入单一输出、多输入多输出以及作为编解码器等多种方式。

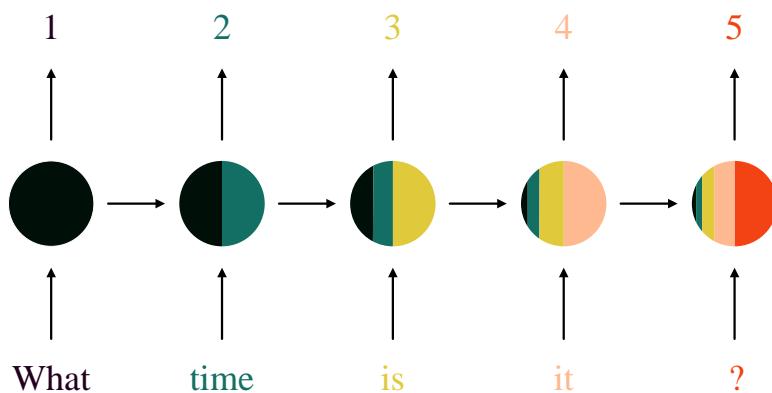


图 2-2 循环神经网络的短期记忆问题

然而 RNN 未能解决长期依赖的挑战，如图2-2所示为 RNN 在文本预测任务中的应用，短期记忆对计算结果影响较大，长期依赖所占的权重比例很小，这也就导致了 RNN 存在短期记忆问题。同时在循环神经网络的训练过程中，随着序列长度的增加，取决于梯度的幅值，梯度会不可避免的趋向于 0 或者无穷，这就导致了梯度消失和梯度爆炸问题。

简单地通过限制参数空间是不能避免梯度消失和梯度爆炸问题的，为了能够表示长期依赖关系并使模型具有一定程度的鲁棒性，RNN 必然会进入参数空间中的梯度消失区域^[58]。

2.2.3 长短时记忆网络

长短时记忆网络（LSTM）是一种特殊的循环神经网络，最早由 Hochreiter 等人^[42] 提出，最大的特点是能够学习序列数据中的长期依赖关系。

这类基于循环结构设计的神经网络模型都能表示为重复模块链的形式，如标准 RNN 中的重复模块链如图2-3所示，仅有单个 tanh 层。

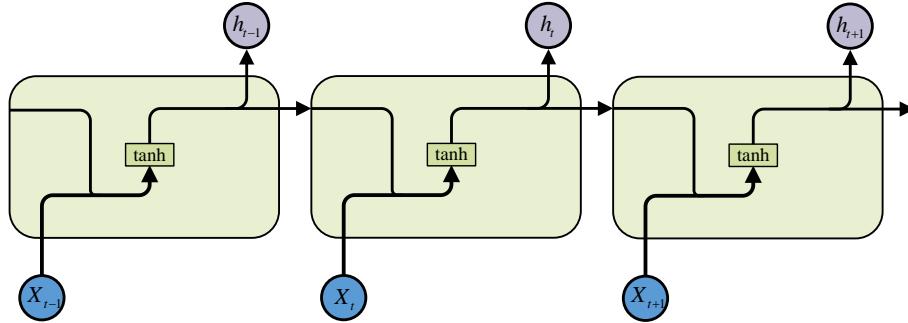


图 2-3 标准 RNN 的重复模块

LSTM 的重复模块结构较为复杂，如2-4所示通过设计了三个门控结构实现了长期依赖，并大幅缓解了梯度消失和梯度爆炸问题。

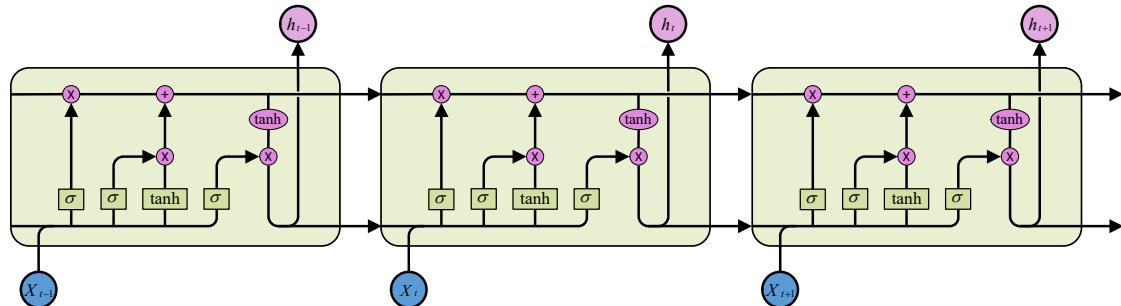


图 2-4 LSTM 的重复模块

LSTM 中最核心的结构是单元状态 cell status，单元状态贯穿整个 LSTM 模型，如图2-5所示，每个重复模块中仅有一些较小的线性相互作用。这样就能够使得信息不变的流动，从而实现长期依赖关系的学习。

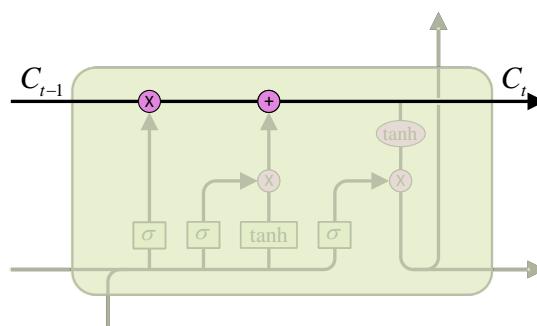


图 2-5 LSTM 的单元状态

LSTM 通过每个单元的隐藏状态实现长期记忆，设计了三个门控结构控制单

元状态的计算，分别为遗忘门控、记忆门控和输出门控。门控结构是为了实现信息的选择性通过，由以 sigmoid 函数为激活函数的简单神经网络以及矩阵的逐点运算构成。

遗忘门控的结构如图2-6所示，目的是接受上一个单元状态 C_{t-1} 并通过一个 sigmoid 层决定保留或者遗忘 C_{t-1} 中的哪些信息。

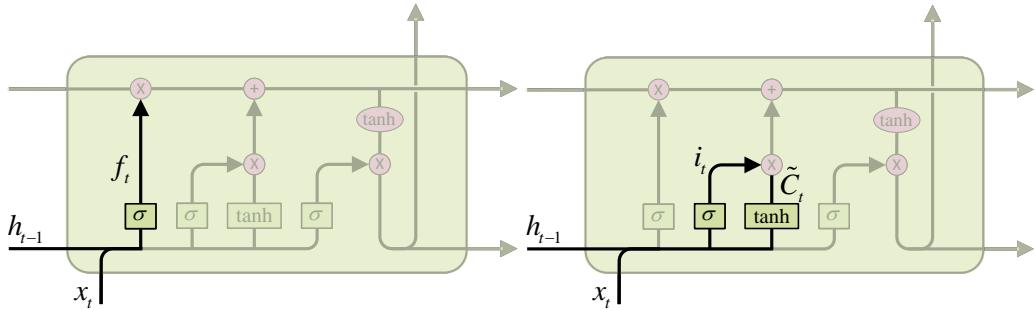


图 2-6 LSTM 的遗忘门控

图 2-7 LSTM 的记忆门控

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (2-14)$$

其中所使用的激活函数为 sigmoid 函数，将遗忘因子 f_t 值被限制在 $[0, 1]$ 之间表示对 C_{t-1} 的保留程度，0 表示完全遗忘，1 表示完全保留。 f_t 与 C_{t-1} 通过逐元素相乘的哈达玛积进行计算。哈达玛积是矩阵乘法的一种，用符号 \odot 表示，对于两个同阶矩阵 $A = (a_{ij})$ 和 $B = (b_{ij})$ ，其哈达玛积为 $C = A \odot B = (c_{ij})$ ，且 $C_{ij} = a_{ij} \times b_{ij}$ 。

记忆门控则是为了确定该将什么样的新信息存放到单元状态 C_t 中，因此记忆门控包括了一个 sigmoid 层和一个 tanh 层，如图2-7所示。sigmoid 层的作用与遗忘门控中相同，计算得到记忆因子 i_t 来决定需要更新的信息，tanh 层则是根据当前时间步的输入信息 x_t 得到新的候选值向量。

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2-15)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (2-16)$$

其中短期记忆 \tilde{C}_t 使用 tanh 作为激活函数，与 RNN 中的 tanh 层作用相同。

在计算得到遗忘因子 f_t 、记忆因子 i_t 和短期记忆 \tilde{C}_t 后，两种因子通过哈达玛积分别作用在上一个单元状态 C_{t-1} 和短期记忆 \tilde{C}_t 上，得到的结果通过逐元素相加得到当前的单元状态 C_t ，如图2-8所示。

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \quad (2-17)$$

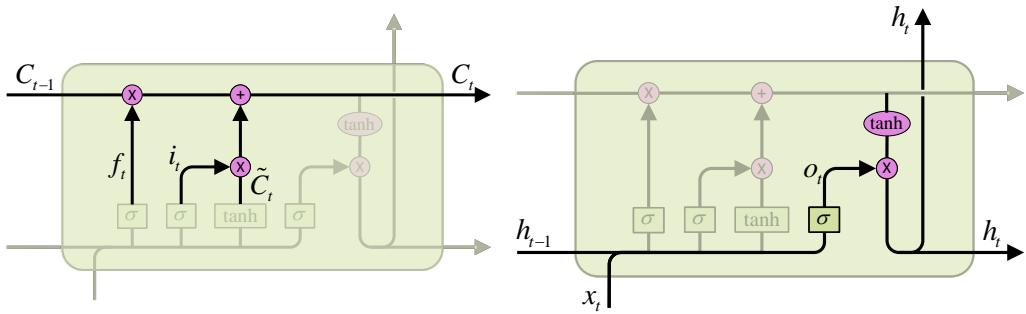


图 2-8 LSTM 中更新单元状态

图 2-9 LSTM 的输出门

最后根据当前单元状态 C_t 以及当前时间步的输入信息 x_t 计算输出向量。如图2-9所示，一个 sigmoid 层计算输出因子 o_t ，一个 tanh 层根据单元状态给出候选值向量，最后相乘得到输出向量 h_t ，计算公式为：

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (2-18)$$

$$h_t = o_t \odot \tanh(C_t) \quad (2-19)$$

LSTM 模型改善了 RNN 中存在的长期依赖问题，通过结构设计在一定程度上避免了 RNN 中连乘带来的梯度消失和梯度爆炸问题。在实际应用中的表现通常能够比 RNN 及 HMM 模型好，且因为其非线性的特性能够用于复杂深度神经网络的构造，在各领域均有着广泛的应用。但 LSTM 并没有完全解决长期依赖问题，在 10^3 量级以上的序列上长期依赖问题仍然存在。同时因为结构较为复杂，参数量较 RNN 成倍增加，模型训练计算量很大，训练难度较大。

针对 LSTM 存在的问题，cho 等人^[43] 在 LSTM 上进行改进提出了门控循环单元 (GRU)，优化了模型参数量以及训练难度。为了更好地阐述本文提出的模型结构，GRU 模型将在第三章中进行介绍。

2.3 时空特征建模方法

本节对后续章节中所涉及的位置预测中的时空特征提取和基于图网络的交通流量预测模型进行介绍，对算法特点和局限性进行分析，为本文所提出的模型方法的介绍做铺垫。

2.3.1 基于时空特征的循环神经网络

在轨迹预测任务中对所有用户建立准确的预测模型是较难的，这是因为轨迹预测任务中使用的数据集较为稀疏，用户轨迹之间相互影响较小。但对每个用户

建立一个预测模型是比较可行的方案。因此，大部分轨迹预测任务基于单个用户建模，基于时空特征的循环神经网络（STF-RNN）就是在此基础上通过对 RNN 模型进行扩展，结合查找表层自动提取时间和空间特征。

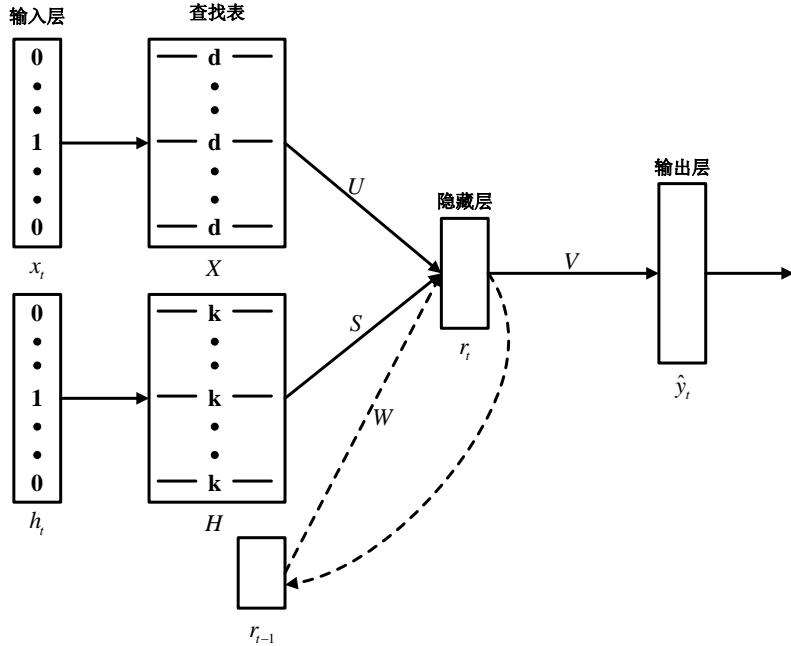


图 2-10 STF-RNN 模型结构

STF-RNN 主要考虑两类特征，一是用户每个位置对应的 ID 信息，使用 1-of-N 进行编码；二是 t 时间时离开兴趣点的小时数，同样使用 1-of-N 进行编码。整体结构如图2-10所示，由两个查找表层和一个 RNN 层构成。

首先通过查找表层将中心 ID 的向量和离开时间映射为实值向量。目的是学习兴趣点和离开时间的输入特征中的有效表示。这种表示方式使模型能够捕捉到用户行为的语义信息。

在 RNN 层中当前的输入以及隐藏层的上一个状态用于计算隐藏层的下一个状态。因此，下一个位置预测不仅取决于当前所处的位置，还受到历史轨迹序列的影响。RNN 结构有助于根据用户的移动历史并发现移动之间存在的依存关系，从而提高模型的性能。

目前看来这样的处理方式无疑是较为粗糙的，但 SF-RNN 模型是最先使用 RNN 建模时空特征的工作之一。摆脱了诸多马尔可夫链通过手工构建转移矩阵提取时空特征的方式，实现了自动的时空特征提取。充分发挥了神经网络模型的优势，通过训练能够自动发现一些用户行为模式中的依存关系。

2.3.2 分层时空长短时记忆网络

大多数顺序模型仅强调整节点之间的依赖性，而忽略了空间和时间关系之列的其他对应关系，因此大多数深度学习模型中均存在数据稀疏性问题。分层时空长短时记忆网络（HST-LSTM）尝试在 LSTM 上进行扩展，实现了对时空特征的提取，并验证了时空依赖性对缓解数据稀疏性问题至关重要。

HST-LSTM 的核心改进为在 LSTM 的基础上引入了空间影响因子和时间影响因子，对 LSTM 中三个门控进行了如下设计：

$$i_t = \sigma(W_i l_t + W_{h_i} h_{t-1} + F_i(s_{t-1}, q_{t-1}) + b_i) \quad (2-20)$$

$$f_t = \sigma(W_f l_t + W_{h_f} h_{t-1} + F_f(s_{t-1}, q_{t-1}) + b_f) \quad (2-21)$$

$$o_t = \sigma(W_o l_t + W_{h_o} h_{t-1} + F_o(s_{t-1}, q_{t-1}) + b_o) \quad (2-22)$$

时空影响因子的计算函数 $F(\cdot)$ 基于加法算子实现：

$$F_k(s_{t-1}, q_{t-1}) = W_{sk} s_{t-1} + W_{qk} q_{t-1}, k = i, f, o \quad (2-23)$$

其中 $W_{sk}, W_{qk} \in \mathbb{R}^{|c|*d}$ 是关于时空因素的线性转换矩阵， $|c|$ 是单元数。

此外在使用时间间隔 v_q 或者地理距离 v_s 计算相应的向量时通过将数值划分为离散间隙并仅对其上下限进行编码来降低对间隔建模时的稀疏性问题。

虽然 HST-LSTM 针对时空特征进行了单独的处理，仍然只是简单地融合到各个门控中，并未对其进行显式建模。这样的方式并不能充分发挥时空特征对建模行为模式的作用，HST-LSTM 通过会话场景等方案设计进行改进，但适用性较差。

2.3.3 时空门控网络

时间门控网络（STGN）是为了解决更充分提取并使用时空特征而提出的。RNN 扩展模型中均不适用于长期兴趣依赖，又如 HST-LSTM 这类并未对 LSTM 结构针对时空特征建模进行设计的模型同样难以建模长期时空特征。STGN 通过增强 LSTM 设计了两对时间门控和距离门口，分别控制短期兴趣和长期兴趣依赖。

如图2-11所示，STGN 设计了两对两个时间门控和两个距离门控，分别表示为 $T1_t, T2_t, D1_t, D2_t$ 。 $T1_t, D1_t$ 用于控制最近访问的地点对下一步行动的影响。STGN 在 LSTM 基础上的公式修改为：

$$T1_t = \sigma(x_t W_{xt_1} + \sigma(\Delta t_t W_{t_1}) + b_{t_1}) \quad (2-24)$$

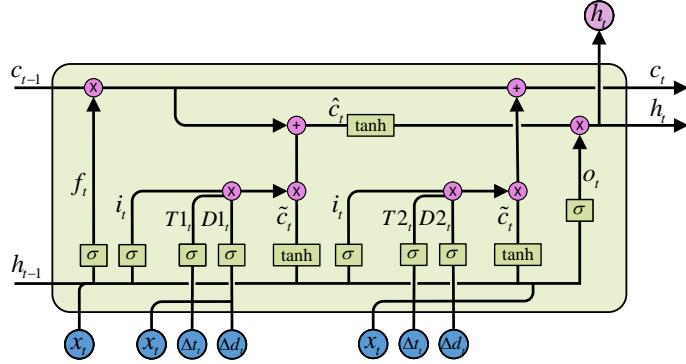


图 2-11 STGN 单元结构

$$T2_t = \sigma(x_t W_{xt2} + \sigma(\Delta t_t W_{t2}) + b_{t2}) \quad (2-25)$$

$$D1_t = \sigma(x_t W_{xd1} + \sigma(\Delta d_t W_{d1}) + b_{d1}) \quad (2-26)$$

$$D2_t = \sigma(x_t W_{xd2} + \sigma(\Delta d_t W_{d2}) + b_{d2}) \quad (2-27)$$

$T2_t, D2_t$ 两个门控的目的是利用时间间隔和距离间隔模拟用户的长期兴趣。并添加了一个新的单元状态 \hat{c}_t 记忆短期兴趣。 c_t 不仅可以记住用户的历史访问的顺序，还可以记住相邻 POI 之间的时间和距离间隔信息，从而捕捉用户的长期兴趣。 \hat{c}_t 和 c_t 的计算公式为：

$$\hat{c}_t = f_t \odot c_{t-1} + i_t \odot T1_t \odot D1_t \odot \tilde{c}_t \quad (2-28)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot T2_t \odot D2_t \odot \tilde{c}_t \quad (2-29)$$

o_t 和 h_t 也进行了相应的修改：

$$o_t = \sigma(W_o[h_{t-1}, x_t] + \Delta t_t W_{to} + \Delta d_t W_{do} + b_o) \quad (2-30)$$

$$h_t = o_t \odot \tanh(\hat{c}_t) \quad (2-31)$$

对距离间隔进行建模可以帮助捕获用户的总体空间兴趣，而对时间间隔进行建模则可以帮助捕获用户的周期性访问行为，通过对时间间隔和距离间隔的显式建模，放大了时空特征对整体行为模式的影响。

STGN 模型虽然考虑了时空特征但仍仅适用于弱实时轨迹和对单个用户进行建模，难以捕获用户之间的时空关系，且因为其扩展了两对门控和一个单元状态，

参数量较 LSTM 成倍地增加。本文基于 STGN 的局限性进行优化，提出了更适用于强实时轨迹建模的 STGRU 模型。

2.3.4 时空图卷积网络

上述时空特征建模的方法均基于单个序列进行建模，但在聚集预测任务中需要考虑用户之间的时空依赖性。交通流量预测任务需要同时兼顾群体之间关系及时空特征，时空卷积网络（STGCN）通过构建基于交通流量路网图实现交通流量的预测，结合任务特点将图拓扑结构视为空间特征，并巧妙设计了时间维度特征的提取。

交通流量预测是较为典型的时间序列预测问题，即给定先前的流量观察值作为预测，在接下来的时间步骤中预测最可能的流量测量值。因此，流量的变化受其周边流量的影响很大，具有天然的时空关联性。

为了充分利用空间信息，STGCN 通过图来建模交通网络，而不是对单独节点进行处理。为了处理 RNN 的固有缺陷，在时间轴上采用卷积结构建模时间特征。

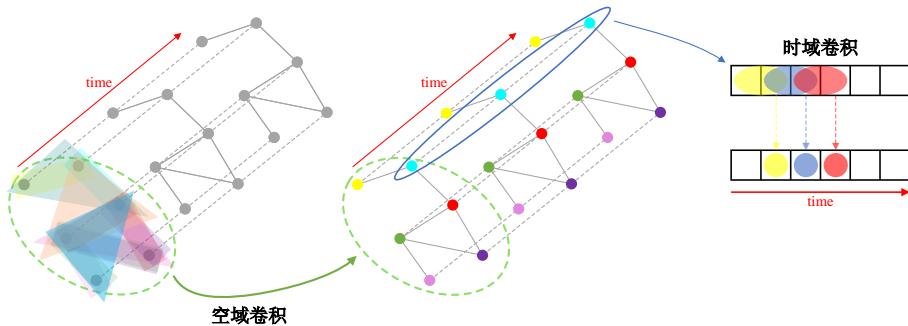


图 2-12 STGCN 原理示意图

如图2-12所示，STGCN 由空间维度上的图卷积和沿时间维度的一维卷积组成。根据交通网络构建图结构，每个节点的特征可以看作图上的信号。

为了充分利用交通网络的拓扑特性，在每个时间片上都采用图卷积网络 GCN 直接处理信号，提取图中的拓扑属性，等价于空间特征提取。时间维度上则是通过标准卷积层提取节点相邻时间片的信息，再与图卷积的结果进行合并更新。

STGCN 能够很好地捕捉交通数据的时空特征，且能够同时考虑用户之间的关系，本文第四章中所提出的聚集预测模型便是基于这一思路对用户之间行为模式的建模方式进行设计。

2.4 数据集介绍

由于聚集预测任务的特殊性，对数据集有较高的要求，常规的出租车数据集、少数目标的轨迹数据集等并不适用，因此本文使用了东京大学空间信息科学中心(CSIS)发布的三个基于SNS的人流数据集，人流数据集目标更多和实时性更强，更符合聚集预测任务的应用场景，统计数据中如表2-1所示。

表 2-1 三个基于 SNS 的人流数据集信息

数据集名称	覆盖时间（天）	用户数量	轨迹点数量
NPF	6	2387	1,068,064
OPF	6	4924	2,552,883
TPF	6	11536	6,883,245

名古屋人流数据集 (Nagoya People Flow, NPF) 包含 2013 年名古屋地区共 2387 个用户的轨迹，共有 1,068,064 个轨迹点。轨迹覆盖时间为 6 天，分别为 7 月 22 日、7 月 28 日、9 月 16 日、9 月 22 日、12 月 24 日和 12 月 29 日。每条轨迹覆盖时间为 24 小时，轨迹点采样时间间隔为 5 分钟。

大阪人流数据集 (Osaka People Flow, OPF) 包含 2013 年大阪地区的 4924 个用户的轨迹，轨迹覆盖时间为 7 月 22 日、7 月 28 日、9 月 16 日、9 月 22 日、12 月 24 日和 12 月 29 日，轨迹点总数为 2,552,883。

东京人流数据集 (Tokyo People Flow, TPF) 是三个数据集中最大的一个，包含了 2013 年东京地区 11536 个用户的轨迹数据，数量为 6,883,245 个轨迹点。时间为 7 月 1 日、7 月 7 日、10 月 7 日、10 月 13 日、12 月 16 日和 12 月 22 日。

上述三个数据集的覆盖时间均包含夏、秋、冬三个季节，每个季节中的两天分别为周末和工作日，包含了不同时间的多种行为模式，能够为后续章节提供较好的实验支撑。

2.5 本章小结

本章对本文中所使用的序列神经网络和时空特征建模算法进行了介绍。首先介绍了序列神经网络相关算法与概念，主要为循环神经网络、长短时记忆网络、分类任务及其评价指标三部分。其次对时空特征建模方法的发展进行阐述，分别介绍了基于时空特征的循环神经网络、分层时空长短时记忆网络、时空门控网络、时空图卷积网络等方法的特点及发展。最后对本文中所使用数据集的特点与详细情况进行介绍，并分析了数据集的选择原因。

第三章 基于信息融合的轨迹预测算法

本章首先分析了当前轨迹预测方法存在的问题，并阐述解决问题的思路。其次详细介绍了本章所提出的时空门控循环单元（STGRU）的结构与原理。最后在真实世界的人流轨迹数据集上进行评估，通过实验验证所提出模型的有效性。

3.1 引言

现有聚集预测方法多基于覆盖区域的人数变化进行预测，缺少对用户行为的建模，且在目标用户少或用户分布不均匀的实际场景下难以发挥作用。为了增强聚集预测方法的鲁棒性，本文先对每个目标用户进行轨迹预测，再根据轨迹预测结果统计得到聚集预测结果。

聚集预测主要利用各种物联网传感设备（如手机、汽车和其他 GPS 设备）所收集的轨迹数据。这些轨迹数据包括多种类型的模式，如步行轨迹、驾驶轨迹和公共交通轨迹等，因此聚集预测需要适用于多类轨迹场景。同时，有别于其他轨迹预测任务，如 POI（Point of Interest）推荐，聚集预测任务更关注实时轨迹数据，并具有较强的空间依赖性。但是现有方法主要基于轨迹数据进行建模，在多类轨迹上适用性较差，忽略了地理环境对行为模式的巨大影响。

为了实现更高精度的轨迹预测，目前的方法主要集中在对轨迹数据的顺序以及相邻轨迹点之间的时间间隔和距离间隔进行建模^[59]。其主要目的是结合时间和空间特征对用户行为模式建模。典型技术如循环神经网络（RNN）、长短时记忆（LSTM）和门循环单元（GRU）已成功应用于各种类型的序列数据建模，并大大改善了性能。然而，上述方法都没有考虑轨迹数据中的地理信息。最近的一些工作致力于扩展 RNN 和 LSTM，以实现对邻点之间的时间和距离间隔的建模。例如，ST-RNN 试图通过扩展 RNN 来建模时空特征，HST-LSTM 将时空影响融入 LSTM 中。最近，STGN 通过设计两对时间门控和距离门控，分别对时间间隔和距离间隔进行建模，实现了当时的最好结果，但性能提升的同时也带来了模型参数量成倍增加的问题。

然而，由于不均匀的采样间隔和传感设备的分布，应用于聚集预测的轨迹数据通常存在数据稀少的问题。以前的工作试图通过时空关系来缓解数据的稀疏性问题，但效果并不明显。受 Li 等人^[60]的启发，地理环境信息（如路网结构）和外部知识（如天气信息和节假日信息）可以有效缓解数据稀少的问题。其中地理环境的影响对于用户短期和长期行为模式的建模至关重要，天气和假日信息则影响

用户的整体行为。例如，如果用户的连续轨迹在同一路段，可以判断其当前的行为模式是相似的。同时，长期历史轨迹的路网信息可以很好地辅助建模用户的长期行为模式。此外，周末时愿意远行的人比平时更多，或者会在某个地方长时间停留。以上这些信息都有利于缓解数据稀少的问题，提高轨迹预测的性能。

为了充分利用外部知识，本章通过整合路网结构和外部知识，提出了一种新的时空门控网络，即时空门控循环单元（Spatio-Temporal Gate Recurrent Unit, STGRU）。设计了一对时间门控和距离门控，通过利用时间和距离间隔来捕捉短期行为模式，并引入了一个路网门控记忆路网结构来建模地理环境约束。该模型将所有轨迹点对应到相应的道路节点上，并将天气和假日信息整合到轨迹信息中进行输入。此外，STGRU 可以对用户的长期和短期行为模式进行建模，并在一定程度上减少模型参数的规模。实验表明，考虑路网结构和外部知识可以有效提高模型的性能。

本章主要工作总结如下：

- (1) 在标准门控循环单元的基础上，我们提出了时空门控循环单元（STGRU）模型，该模型引入了路网结构和外部知识，并在一定程度上减少了模型参数量。
- (2) 提出了一种创新的门控机制，增加了路网门控，它可以为学习用户轨迹的时空关系建立路网结构模型。
- (3) 在三个真实世界的数据集上评估了所提出的方法，包括名古屋、大阪和东京的人流数据。与最先进的方法进行的综合比较显示了我们的模型的有效性。

3.2 时空门控循环网络

本节首先给出了聚集预测场景下轨迹预测的定义，并介绍了标准门控循环单元（GRU）的实现。然后详细阐述了时空门控循环单元（STGRU），STGRU 通过建模时间、距离间隔以及路网机构来建模用户的短期和长期行为模式。

3.2.1 轨迹预测问题定义

首先需要对聚集预测场景下轨迹预测任务的问题进行定义。

假设需要进行轨迹预测的用户有 M 个，可以将 M 个用户的集合表示为 $\mathbb{U} = \{u_1, u_2, \dots, u_M\}$ 。随后根据边长 a ，将城市划分为若干个网格并进行编号，其中每个网格的面积为 a^2 。且每个区域有对应的唯一区域编号 r 。

对于用户 u 来说，他在当前时刻 t 之前的历史轨迹信息可以表示为 $H_i^u = r_{t_1}^u, r_{t_2}^u, \dots, r_{t_{i-1}}^u$ ，其中 $r_{t_i}^u$ 表示用户 u 在 t_i 时刻所在的区域。

轨迹预测目标为预测用户在 t_i 时刻所在的区域，也就是对于用户 u 在 t_i 时刻

在区域 r 的预测分数 s_{r,t_i}^u 越高, 就意味着用户 u 在 t_i 时刻位于区域 r 的概率越高。

在本章中, 对于地图区域的划分仅采用网格划分的方式, 这是因为网格划分方式的适用性最强, 能够广泛地在各类场景下使用。

3.2.2 标准门控循环单元

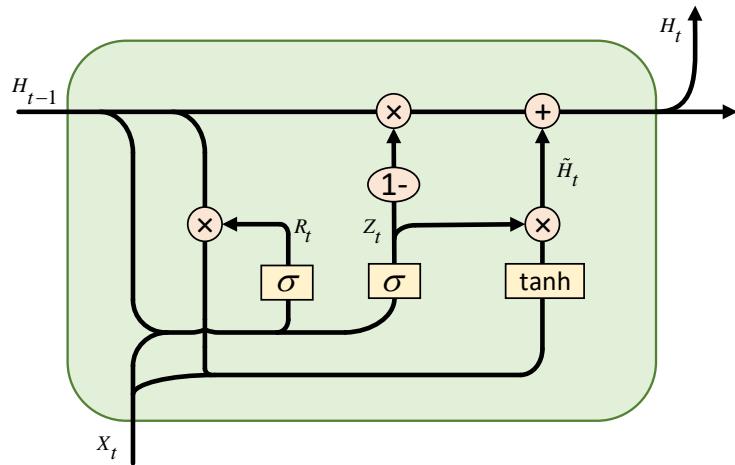


图 3-1 标准 GRU 模型

GRU 模型是 LSTM 模型的一个变种, 继承了 LSTM 模型能够学习序列数据中长期依赖性的特性。GRU 相较于 LSTM 模型结构更加精简, 并且在大多数任务上效果出色。由于 GRU 模型内部门控结构较少, 在对时空特征进行融合时难度较大, 但是 GRU 模型相较于 LSTM 存在天然的参数量上的优势。本章中的 STGRU 模型通过扩展 GRU 模型实现时空特征和地理结构特征的提取, 并大幅降低了参数量。

标准的 GRU 模型如图3-1所示, 将 LSTM 中的遗忘门控和输入门控融合成一个更新门控, 并去掉了单元状态, 使用隐藏状态来传递信息。GRU 的门控更新公式如下:

$$R_t = \sigma(W_{xr}[H_{t-1}, X_t] + b_r) \quad (3-1)$$

$$Z_t = \sigma(W_{xz}[H_{t-1}, X_t] + b_z) \quad (3-2)$$

$$\tilde{H}_t = \tanh(W_{xh}X_t + W_{hh}(R_t \odot H_{t-1}) + b_h) \quad (3-3)$$

$$H_t = Z_t \odot H_{t-1} + (1 - Z_t) \odot \tilde{H}_t \quad (3-4)$$

其中隐藏单元数量为 h ，在给定时间步 t 时，输入为 $X_t \in \mathbb{R}^{n \times d}$ ， d 表示批量大小，上一时间步的隐藏状态为 H_{t-1} 。 $R_t, Z_t \in \mathbb{R}^{n \times h}$ 表示重置门控和更新门控，其中 $\sigma(\cdot)$ 是 Sigmoid 函数。 $\tilde{H}_t \in \mathbb{R}^{n \times h}$ 表示时间步 t 时的隐藏状态，其中 $\tanh(\cdot)$ 是双曲正切函数。 $W_{xr}, W_{xz}, W_{xh}, W_{hr}, W_{hz}, W_{hh} \in \mathbb{R}^{d \times h}$ 则是各个门控的权重参数， b_r, b_z, b_h 是对应的偏置向量。 \odot 表示矩阵的哈达玛积。

3.2.3 时空门控循环单元

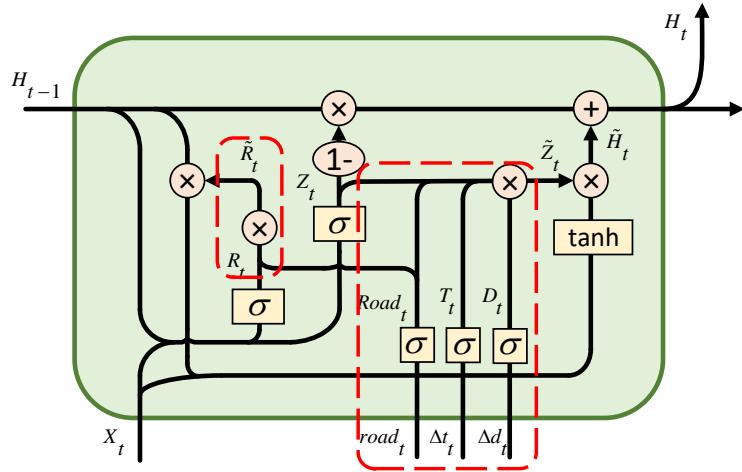


图 3-2 STGRU 模型

如图3-2中的两个红色虚线框所示，STGRU 增加了时间门控、距离门控和路网门，分别表示为 T_t 、 D_t 和 $Road_t$ 。 T_t 、 D_t 分别用于建模时间间隔和距离间隔对轨迹预测的影响， $Road_t$ 用于捕获路网信息对行为模式的影响。STGRU 是基于 GUR 进行扩展的，因此上述三个门控的计算公式也与 GRU 中的重置门控和更新门控类似，如下所示：

$$T_t = \sigma(W_{xt}X_t + \sigma(\Delta t_t W_t) + b_t) \quad (3-5)$$

$$D_t = \sigma(W_{xd}X_t + \sigma(\Delta d_t W_d) + b_d) \quad (3-6)$$

$$Road_t = \sigma(W_{xroad}X_t + \sigma(road_t W_{road}) + b_{road}) \quad (3-7)$$

为了利用 T_t 、 D_t 和 $Road_t$ 三个门控捕获的特征信息，需要将其与重置门控和更新门控进行融合，修改后的计算公式为：

$$\tilde{R}_t = R_t \odot Road_t \quad (3-8)$$

$$\tilde{Z}_t = Z_t \odot T_t \odot D_t \odot Road_t \quad (3-9)$$

对应的公式3-3和3-4修改为：

$$\tilde{H}_t = \tanh(W_{xh}X_t + W_{hh}(\tilde{R}_t \odot H_{t-1}) + b_h) \quad (3-10)$$

$$H_t = \tilde{Z}_t \odot H_{t-1} + (1 - \tilde{Z}_t) \odot \tilde{H}_t \quad (3-11)$$

其中 Δt_t 表示时间间隔, Δd_t 表示距离间隔, $Road_t$ 表示路网信息。 T_t 相当于输入信息中时间间隔特征的捕获器, D_t 则为输入信息中距离间隔特征的捕获器, $Road_t$ 用于捕获输入信息中的路网结构特征。通过增加一个新重置门控状态 \tilde{R}_t 和新的更新门控状态 \tilde{Z}_t 用以计算路网门控对重置门控的影响以及三个门控同时对更新门控的影响。上述所有修改, 共同确定新的隐藏状态 H_t 。

候选的隐藏状态 \tilde{H}_t 是由输入信息、重置门控和前一个时间步的隐藏状态决定的。重置门控的新状态 \tilde{R}_t 被设计用来记忆用户的长期路网访问信息。计算方式为先由 $Road_t$ 记忆路网信息 r_t , 然后与输入信息一起融入 \tilde{R}_t , 再进一步融合到 \tilde{H}_t 。

标准 GRU 中的更新门控 Z_t 可以捕获短期依赖性, 因此, 设计了一个时间门控和距离门控, 结合上述路网门控来控制更新门控的状态。 T_t 用于记忆轨迹点之间的时间间隔 Δt_t , 并参考 LSTM 中, 使用逐元素相乘的哈达玛积将其融入新的更新门控状态 \tilde{Z}_t 中。同样的, D_t 用于记忆轨迹点之间的距离间隔 Δd_t , $Road_t$ 记忆路网信息, 并均与 Z_t 通过哈达玛积得到 \tilde{Z}_t 。上述对于距离间隔以及时间间隔的建模可以帮助捕获用户的行为模式, 如当前的移动速度、所处状态等。对路网信息的建模可以帮助捕获用户的短期行为约束和长期的目标, 以及捕获用户之间的空间关系, 这部分内容会在下一章中详细阐述。

根据上述模型结构, 对于轨迹数据需要进行一定的处理。首先, 需要计算轨迹点之间的时间间隔和距离间隔, 于是 H^u 可以转化为:

$$[(r_1^u, 0, 0), (r_2^u, t_2^u - t_1^u, d(l_1, l_2)), \dots, (r_n^u, t_n^u - t_{n-1}^u, d(l_{n-1}, l_n))] \quad (3-12)$$

然后, 增加天气、假期信息和路网信息, 并进一步处理为:

$$(r_n^u, weather_n, date_n, t_n^u - t_{n-1}^u, d(l_{n-1}, l_n), road_n) \quad (3-13)$$

其中 r_t^u 包含经纬度信息和所在区域信息, $weather_t$ 则包括了最高温度、最低温度和平均温度, $date_t$ 根据轨迹点所属日期判断是否为节假日, 用 0, 1 区分标记。于是 STGRU 中的 X_t 就等价于 $(r_t^u, weather_t, date_t)$, Δt_t 由 $t_t^u - t_{t-1}^u$ 计算得到, Δd_t 则

是两个轨迹点之间的距离间隔 $d(l_{t-1}, l_t)$, $d(\cdot, \cdot)$ 是欧式距离的计算公式。而 $road_t$ 是一个向量, 由轨迹点所在的道路节点以及其相邻节点编号拼接而成, 节点编号为公开地图 OpenStreetMap 中对应路段的编号。例如, 轨迹点 i 所处的道路节点为 $node_i$, $node_i$ 的 c 邻居节点分别表示为 $node_{i+1}, node_{i+2}, \dots, node_{i+c}$, 那么 $road_t$ 就可以表示为:

$$road_t = (node_i, node_{i+1}, \dots, node_{i+c}) \quad (3-14)$$

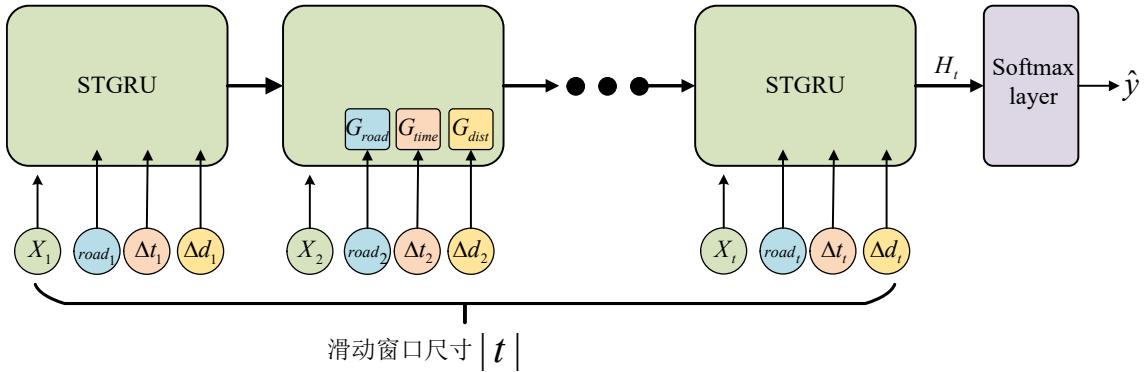


图 3-3 基于 STGRU 的轨迹预测模型

如图3-3所示, 轨迹预测任务整体的模型架构通过堆叠一个 STGRU 层和一个 softmax 层得到, 目的是为了对比实验的公平性, 减少其他因素的影响。采用单层网络模型进行对比实验, 以 softmax 作为分类器得到输出结果, 使用多分类交叉熵作为损失函数:

$$J = - \sum_{i=1}^K y_i \log(p_i) \quad (3-15)$$

3.2.4 模型分析与训练

STGRU 模型相较于最新的时空特征模型, 在一定程度上减少了模型参数的规模。单个 STGRU 单元的参数量可以计算为 $3 \times d_h^2 + 6 \times d_h \times d_x + 6 \times d_h$, 其中 d_h 是隐藏单元数, d_x 是输入的维度。同理, 一个 LSTM 单元的参数量计算可得为 $4 \times d_h^2 + 4 \times d_h \times d_x + 4 \times d_h$ 。GRU 单元相较于 LSTM 单元减少了一个门控结构, 因此参数量也较少, 为 $3 \times d_h^2 + 3 \times d_h \times d_x + 3 \times d_h$ 。本章中的基准模型 STGN 是目前通过扩展循环网络建模时空特征的模型中效果最好的, STGN 基于 LSTM 进行了改进, 参数量为 $5 \times d_h^2 + 8 \times d_h \times d_x + 10 \times d_h$ 。对比可以发现, STGRU 模型的参数量不足 STGN 模型的三分之二。

在模型训练中, 使用 Adam 优化器。Adam 优化器同时考虑了一阶和二阶梯

度，能够自动调整模型学习率，适用于大规模数据和参数量较大的场景。

3.3 实验与分析

本节首先对于本章的实验配置信息进行简单介绍。然后通过多组的对比实验、消融实验、应用分析，验证本章提出的 STGRU 模型在聚集预测场景轨迹预测任务上的有效性。

3.3.1 实验配置和基线方法

本文所有实验所使用的实验环境如表3-1所示。

表 3-1 实验环境配置表

软硬件	配置
CPU	Intel(R) Core(TM) i5-8600K CPU @ 3.60GHz
内存	32GB DDR4
GPU	GeForce RTX™ 2080 Ti × 2
GPU 显存	11GB GDDR6
操作系统	Ubuntu 16.04
Cuda 版本	11.4
cuDNN 版本	V10.1.243
anaconda 版本	4.5.11
python 版本	3.6.3
tensorflow 版本	2.1.0

本章实验基于章节 2.3 中所介绍的三个数据集展开。首先剔除了三个数据集中轨迹长度小于 30 的用户数据，然后将剩余数据的 70% 作为训练集，20% 作为验证集，10% 作为测试集。

由于三个数据集中均不包含轨迹点对应的区域信息，仅包含较为稀疏的 POI 信息。因此以 $1km$ 为边长对所拥有覆盖区域进行了划分，每个区域面积为 $1km \times 1km$ ，并以常规的蛇形编号方式对区域进行编号。对于数据集中的轨迹点信息，通过经纬度计算对应的区域，增加相应的区域标签。在所有数据集上均采用滑动窗口的方法生成样本。由于数据集本身质量较高，采样间隔较短，为了模拟现实场景下出于隐私保护考虑、设备精度不足等问题导致的轨迹点缺失问题，在滑动窗口内部增加了随机采样机制，即生成样本时每个轨迹点之间的时间间隔并不均匀，而是随机决定的，目的是增加轨迹的复杂性，更符合现实场景。例如，生成样本时第一个轨迹点的时刻为上午 8:00，随机区间为 $[1 - 6]$ 。假设生成的随机数为 3，样本的第二个轨迹点和第一个轨迹点之间的采样间隔为 $3 \times 5min = 15min$ ，

即为上午 8:15。

本章选取了五种具有代表性的轨迹预测模型与 STGRU 进行对比实验，模型的详细介绍如下：

- **循环神经网络 RNN**^[41]: 循环神经网络在内部循环传递状态，因此适用于广义上的时间序列模型，并被广泛应用于各类时间序列预测任务中。
- **长短时记忆网络 LSTM**^[42]: LSTM 在大幅缓解了 RNN 存在的梯度消失和梯度爆炸问题，适用于处理和预测时间序列中时间跨度较长的事件。
- **门控循环单元 GRU**^[43]: GRU 是 LSTM 模型的一个变种，参数量上比 LSTM 略少，性能上与 LSTM 相近，某些小数据集上甚至表现出更好的性能。
- **分层时空长短时记忆网络 HST-LSTM**^[50]: HST-LSTM 尝试将时空影响融合到 LSTM 的三个门控中，使用 LSTM 本身结构来尝试提取时空特征。由于本章使用的数据集中并没有会话信息，因此采用其通用版本 ST-LSTM 进行对比实验。
- **时空门控记忆网络 STGN**^[51]: 通过增强 LSTM 结构，引入两对时空门控结构来捕获时空关系，是目前性能最好的模型之一。

3.3.2 评价指标

为了评估 STGRU 模型的性能，并与上述五个基线方法进行比较，本章使用 AUC 和 Recall@k 两个指标作为评价标准。基于区域的轨迹预测任务本质上是一个多分类任务，AUC 指标可以较好地评估模型分类效果，具体原理见第二章。Recall@k 则是正确预测数与总预测数的比例。在本章中，使用 $K = \{1, 5, 10, 15, 20\}$ 来计算 Recall@K 的不同结果。多分类交叉熵得到的结果是对应每个区域一个概率，所有区域的概率和为 1。对所有区域按照其概率从高到低排列。recall 分数计算为概率最大的前 K 个区域中发现真实区域的次数的百分比。 U 是用户集合，假设 L_u 代表测试数据中用户 u 的真实区域集合， $P_{K,u}$ 表示前 K 个预测区域的集合，则 Recall@K 的计算公式可以表示为：

$$\text{Recall}@K = \frac{1}{|U|} \sum_{u \in U} \frac{|L_u \cap P_{K,u}|}{|L_u|} \quad (3-16)$$

3.3.3 对比实验

方法比较。表3-2、3-3和3-4中给出了 STGRU 模型和五个基线模型在三个数据集上通过 Recall@K 和 AUC 评估的性能。在所有对比实验中，隐藏状态的大小设置为 32，迭代次数设置为 200，批量大小设置为 512，滑动窗口大小设置为 10，名古屋人流数据集和大阪人流数据集的随机时间间隔为 1 到 3，东京人流数据集的

随机时间间隔为 1 到 5，原因是东京人流数据集中的数据密度更高，通过增加随机时间间隔可以提高数据的复杂性，后续对与不同随机时间间隔也进行了对比实验，验证随机时间间隔大小对实验的影响。本章的所有实验都使用相同的参数设置。

表 3-2 在 NPF 数据集上 Recall@K 和 AUC 指标的结果

NPF	recall@1	recall@5	recall@10	recall@20	AUC
LSTM	0.0428	0.1677	0.2894	0.4300	0.6951
GRU	0.0712	0.2372	0.3421	0.4771	0.7778
RNN	0.0809	0.2580	0.3570	0.4865	0.8018
ST-LSTM	0.0621	0.2438	0.3663	0.5050	0.7976
STGN	0.0762	0.2728	0.3734	0.5061	0.8177
STGRU	0.0920	0.2829	0.3891	0.5231	0.8290

表 3-3 在 OPF 数据集上 Recall@K 和 AUC 指标的结果

OPF	recall@1	recall@5	recall@10	recall@20	AUC
LSTM	0.0383	0.1638	0.2634	0.4375	0.7140
GRU	0.0455	0.1979	0.3007	0.4656	0.7611
RNN	0.0588	0.2258	0.3260	0.4881	0.8324
ST-LSTM	0.0502	0.2107	0.3174	0.4890	0.7817
STGN	0.0633	0.1699	0.2833	0.4920	0.8292
STGRU	0.0681	0.2439	0.3464	0.5116	0.8545

表 3-4 在 TPF 数据集上 Recall@K 和 AUC 指标的结果

TPF	recall@1	recall@5	recall@10	recall@20	AUC
LSTM	0.0752	0.3331	0.4849	0.6428	0.8317
GRU	0.0789	0.3319	0.4793	0.6384	0.8451
RNN	0.0856	0.3634	0.5066	0.6561	0.8645
ST-LSTM	0.0864	0.3699	0.5213	0.6682	0.8727
STGN	0.0900	0.3327	0.5103	0.6672	0.8734
STGRU	0.0933	0.3795	0.5263	0.6730	0.8755

如表中实验结果所示，本章提出的 STGRU 模型在三个数据集的所有指标上都明显优于现有的最先进方法 STGN。在名古屋、大阪和东京数据集的 Recall@1 指标上，STGRU 模型相较于五个基线模型的性能提升分别为 18.1% - 110.2%，5.7% - 74.7% 和 3.7% - 24.1%。同时在其他 Recall@k 指标上相较于最新的 STGN 模型最

大提升比例分别达到了 43.5%，22.3% 和 3.9%。结果表明，STGRU 模型中对于路网结构特征提取的机制可以更好地建模用户行为模式，对短期时空上下文特征的建模提升了在强实时数据下的效果，以及对轨迹预测的任务有效性。这是因为添加的路网门控与更新门控相结合，将短期路网特征融合到模型中，而重置门控则与长期路网特征相结合。

此外，RNN 模型在这三个数据集上的表现比 LSTM 模型略好。这是因为 RNN 模型具有短期记忆的特点。时间越近，轨迹点的权重就越大。因此即使加入了随机区间，得到的样本仍然具有较强的实时性，所以 RNN 模型的性能更好。同样，GRU 模型在对强实时数据进行建模时也优于 LSTM 模型。HST-LSTM 模型和 STGN 模型的性能优于以上三种模型，这验证了时空特征对轨迹预测的重要性。其中，STGN 模型的性能优于 HST-LSTM 模型，验证了通过特定门控结构提取时空特征的方式比在 LSTM 门控基础上的融合改进更为有效，其原因可能是参数的增加带来的模型建模能力的提升。

在进行对比实验的三个数据集中，每个数据集涵盖了日本一个地区总共 6 天的轨迹数据，相邻轨迹点之间的时间间隔为 5 分钟，因此时间上的刻度总数为 $6 \times 24 \times 60 \div 5 = 1728$ 。数据集所对应的每个地区可分为约 5000 至 10000 个区域。以 NPF 数据集为例，区域的数量约为 5000。计算可以得到，该数据集的时空矩阵大小约为 5000×2000 。然而，NPF 数据集中的轨迹点的数量只有 100 万。在去除重复的时空区域后，轨迹点覆盖矩阵的大小小于数据集的时空矩阵大小的 1%。RNN、LSTM 和 GRU 等模型直接对时空矩阵进行训练，这将导致严重的数据稀疏问题。STGRU 模型通过时间间隔、距离间隔和路网结构在轨道点之间增加约束。在训练 STGRU 模型时，只需要考虑每个样本所覆盖的局部区域以及轨迹点的周边区域，这大大缓解了数据集的数据稀少问题，与上述三个模型相比，可以更好地建模用户行为模式，有效的提升了模型对轨迹数据的预测性能。

表 3-5 STGRU 模型在不同隐藏状态大小下的性能

Cell Size	recall@1	recall@5	recall@10	recall@20
32	0.0933	0.3795	0.5263	0.6730
64	0.0922	0.3836	0.5253	0.6754
128	0.0942	0.3774	0.5217	0.6736
256	0.0923	0.3761	0.5238	0.6704
512	0.0913	0.3760	0.5190	0.6658

参数影响。在标准的 RNN 模型中，不同的单元大小将导致不同的性能。于是进行了不同隐藏状态大小对 STGRU 模型性能的影响试验。通过改变隐藏状态

大小为 32, 64, 128, 256, 512 来比较不同隐藏状态大小对 STGRU 模型性能的影响。从表3-5可以看出，在一定程度上增加隐藏状态尺寸可以提高 STGRU 模型的性能。但是过大的隐藏状态尺寸会增加训练时间，导致性能下降。当模型的单元数量确定后，隐藏状态大小决定了模型的复杂性，适当地增大隐藏状态尺寸能够更好地拟合数据，得到更好的性能表现。

3.3.4 消融实验

时间门控和距离门控的有效性。 STGRU 使用了一个时间门控和一个距离门控与更新门控相结合来捕获短期的依赖关系。时间门控和距离门控对建模时间间隔和距离间隔的有效性是很重要的。如公式3-11所示，时间门控和距离门控可以通过设置 $T_t = 1$ 和 $D_t = 1$ 来关闭。同时为了消除路网门控的干扰，STGRU 模型中的路网门控也被关闭。因此在名古屋人流数据集上进行了三组实验，分别为关闭时间门控、关闭距离门控和同时关闭两个门控，以比较和验证时间门控和距离门控的有效性。

表 3-6 时间门控和距离门控对模型性能的影响

NPF	recall@1	recall@5	recall@10	recall@20
GRU	0.0712	0.2372	0.3421	0.4771
GRU+ D_t	0.0887	0.2763	0.3778	0.5080
GRU+ T_t	0.0844	0.2744	0.3773	0.5099
GRU+ $D_t + T_t$	0.0909	0.2801	0.3805	0.5116

从表3-6中可以发现，时间门控和距离门控在实验数据集上的重要性相似。与 GRU 模型相比，同时增加时间门控和距离门控的 $GRU + D_t + T_t$ 在四个评价指标上的性能提升分别为 27.67%、18.04%、11.22% 和 7.23%。而仅增加一个门控结构的 $GRU + T_t$ 和 $GRU + D_t$ 之间性能差异非常小，说明距离间隔和时间间隔对行为模式的建模有类似的效果。而且， $GRU + D_t + T_t$ 相较于 $GRU + T_t$ 和 $GRU + D_t$ 的性能提升很小，说明在测试数据集上，时间间隔和距离间隔的特征存在很大程度的重叠。

路网门控的有效性。 在 STGRU 中设计了一个路网门控，它与更新门控和重置门控结合在一起，以捕捉长期和短期的路网依赖性。本组实验的动机是通过实验研究路网门控在更新门控和复位门控中的作用。如公式3-10和3-11所示，在 \tilde{R}_t 和 \tilde{Z}_t 中分别设置 $Road_t = 1$ ，即可以关闭路网门控。通过设置在名古屋人流数据集上的三组实验，即关闭所有路网门控和分别关闭单个路网门控，以验证路网闸门在捕捉长期和短期依赖关系方面的有效性。

表 3-7 路网门控对模型性能的影响

NPF	recall@1	recall@5	recall@10	recall@20
STGRU- R_t	0.0909	0.2801	0.3805	0.5116
STGRU- R_{2t}	0.0867	0.2776	0.3786	0.5101
STGRU- R_{1t}	0.0862	0.2795	0.3796	0.5093
STGRU	0.0920	0.2829	0.3891	0.5231

如表3-7所示，关闭单个路网门控的性能不如关闭所有路网门控。这可能是由于路网结构的长期特征和短期特征需要一起使用。关闭单个路网门控会导致路网信息对预测结果无效。同时，一些参数会被用来对路网特征进行建模，这一原因导致了模型的性能下降。因此，单独关闭一个路网门控的性能几乎是一样的。

滑动窗口大小的影响。在上述对比实验中，训练样本和测试样本均是通过滑动窗口采样取得。滑动窗口的大小限制了单个样本的长度。为了比较 STGRU 模型在不同大小的滑动窗口中的性能，通过设置不同尺寸的滑动窗口来获取不同长度样本进行实验，分别为 10、15、20、25 和 30。为了保证较大滑动窗口尺寸下的样本数量，在东京人流数据集上进行实验，因为该数据集的平均轨迹长度最长。

表 3-8 不同长度滑动窗口对模型性能的影响

Window Size	recall@1	recall@5	recall@10	recall@20
10	0.0933	0.3795	0.5263	0.6730
15	0.0929	0.3897	0.5331	0.6828
20	0.0949	0.3624	0.5174	0.6786
25	0.0905	0.3794	0.5267	0.6831
30	0.1014	0.3611	0.5842	0.6957

样本长度决定了模型的单元数量，以及模型的参数。如表3-8所示，随着滑动窗口长度的增加模型参数量会同比增加，模型的整体性能有一定的提高。当滑动窗口大小为 30 时，模型参数量是滑动窗口大小为 10 时的 3 倍，四个指标的性能提升分别为 8.68%、-4.85%、11% 和 3.37%。虽然样本长度的增加可以提高模型的性能，但出于聚集预测任务的实际应用场景考虑，样本长度在 10 以内更有意义，这也是在与基线模型进行对比实验时滑动窗口大小设置为 10 的主要原因。

随机区间大小的影响。另一个重要参数是滑动窗口中随机区间的大小。通过对滑动窗口中的轨迹点之间的时间间隔进行随机采样，目的是提高轨迹样本的复杂性。为了比较不同的随机区间对轨迹样本的复杂性和模型性能的影响，在东京人流数据集上设置了不同的随机区间进行对比实验，分别为 3、5、7、9、11。选择

东京人流数据集是因为该数据集的轨迹密度较大，轨迹点之间的连续性很强，需要增加一定的复杂性才更符合实际应用场景下的轨迹数据分布。

表 3-9 不同随机时间间隔对模型性能的影响

Rand Interval	recall@1	recall@5	recall@10	recall@20
3	0.0889	0.3725	0.5148	0.6636
5	0.0933	0.3795	0.5263	0.6730
7	0.0920	0.3835	0.5300	0.6776
9	0.0822	0.3677	0.5196	0.6752
11	0.0747	0.3512	0.5104	0.6702

由表3-9，可以看出随机区间大小为5和7时，STGRU模型性能最好，随机区间大小过大和过小都会导致模型性能下降。在另外两个数据集上，随机区间大小为3时，模型性能较好，这就是在与基线模型的对比实验中三个数据集分别使用不同的随机区间的原因，使用随机区间可以使样本更接近于真实世界的数据。

3.3.5 应用分析

最后，为了验证STGRU模型能够有效处理和预测短轨迹数据和长轨迹数据，与基线模型进行了两组应用分析实验。

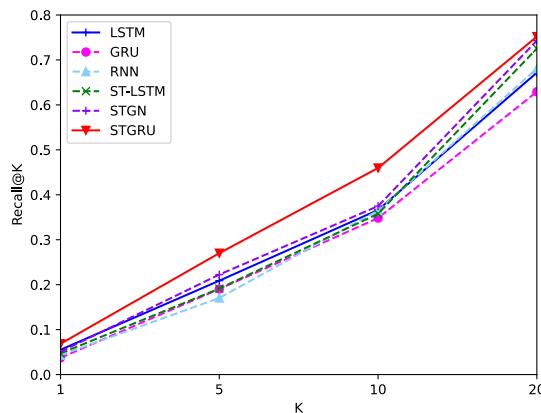


图 3-4 STGRU 模型在轨迹长度小于 30 的数据上的性能

如果用户的轨迹数据很稀少，就意味着很难理解用户的行为模式，这就对模型的性能要求更高，第一组实验基于东京人流数据集，在轨迹长度小于30的数据上进行实验，由于数据集过短没有使用随机间隔，使用最原始的滑动窗口生成样本，评价指标依旧为 $recall@k$ 。如图3-4所示，STGRU模型在 $recall@1$ 和 $recall@5$ 上的表现最好，这说明了STGRU模型能更好地处理稀疏数据。

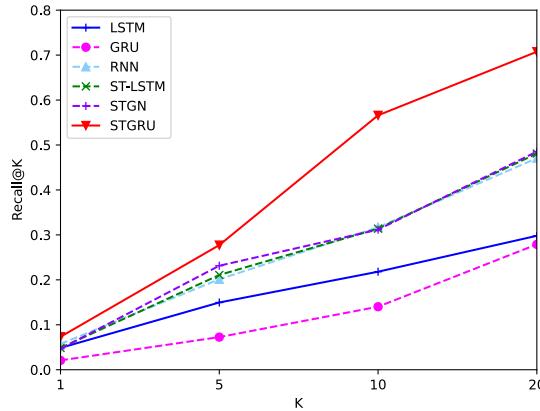


图 3-5 STGRU 模型在轨迹长度大于 200 的数据上的性能

在另一组实验中，使用的是轨迹长度大于 200 的数据，并沿用了对比实验中的参数设置。如图3-5所示，在长轨迹数据上，STGRU 模型的性能也优于所有基线模型，这验证了 STGRU 模型能够较好地提取和使用长期特征，特别是长期路网特征对强实时数据建模依然是有效的。

3.4 本章小节

在本章中，通过增强门控循环单元（GRU），提出了一个时空门控循环单元（STGRU）模型。在 STGRU 中，引入了时间门控和距离门控来捕获连续轨迹点之间的时间间隔和距离间隔特征，这对于建模用户的短期行为至关重要；然后引入了路网门控来捕获长期和短期路网结构。路网结构所代表的地理环境对用户的短期和长期行为都至关重要。上述三个门控结构与 GRU 中的更新门控相结合，用以建模用户的短期行为模式。只有路网门控与 GRU 中的重置门控相结合，建模用户的长期行为模式。并在三个真实世界的数据集上的实验结果验证了 STGRU 模型的有效性，它优于最新的方法。

第四章 基于空间特征增强的聚集预测方法

本章首先分析了轨迹预测方法在聚集预测任务下的局限性，并阐述解决问题的思路。然后详细介绍了本章所使用的希尔伯特曲线和图嵌入方式增强轨迹数据空间特征的原理。最后在真实世界的人流轨迹数据集上模拟现实场景下的聚集预测任务进行评估，并通过实验验证所提出模型的有效性。

4.1 引言

聚集预测能够对根据用户的历史轨迹信息及当前分布状态预测未来的用户位置，当前聚集预测算法通常对用户轨迹预测结果进行融合得到聚集预测结果，其主要难点如何建模在于同一时空下多轨迹之间关联特征。同时聚集预测任务对数据集要求较高，在用户数量和轨迹质量满足要求的前提下，还对数据的时空分布有所要求。但是现有方法主要针对单个用户的行为模式进行建模，忽略了用户之间的相互影响以及城市地理环境对用户行为模式的影响。

现有的轨迹预测方法通常使用用户时空特征和轨迹序列信息建模每个用户的行为模式，但并没有考虑用户之间的相互影响，如道路中的拥堵情况、多用户的趋同性等。目前只有交通流量预测对用户之间关联性的时空特征进行挖掘建模。交通流量预测对各个路段的流量进行分时段统计，再根据各路段之间的结构关系以及交通流量的变化模式进行建模，实现当前路段的流量预测。如时间图卷积网络 (T-GCN)^[61] 对每个时间片按快照的形式进行输入，在时间维度上使用 GRU 模型进行建模的方式，以及时空图卷积网络 (STGCN)^[62] 在时间维度上使用单维卷积网络建模时间特征。

然而，聚集预测需要同时建模多用户轨迹^[63] 的时空特征和用户之间的关联特征。受上述两个模型的启发，历史的用户分布信息对多用户场景下特定用户的行为模式有较大影响。道路的拥堵情况，若当前部分可选择道路存在拥堵现象，其余可选道路的被选择概率就会上升。多用户的趋同性，大型活动的举办会吸引大量用户前往，其周围道路的流量将会出现持续的增长，此时周边某个用户以此地为目的地的概率也将大大增加。流量数据与道路结构信息都有利于建模多用户之间的行为模式。

为了建模同一时空下用户分布情况对行为模式影响以及进一步增强数据空间特征，本章提出了一种基于空间特征增强的聚集预测模型 (Spatial Enhanced Gather Prediction Model, SEGPM)。通过引入希尔伯特扫描算法进行区域编号，增强了区

域编号之间的空间特征。并设计了一种基于统计的路网图构建方式，将用户的分布信息和路网结构信息通过图嵌入得到，生成向量通过 STGRU 模型的路网门控进行建模。实验表明，考虑用户之间的相互影响和区域编号的空间特征增强能够有效提高模型性能。

本章主要工作总结如下：

- (1) 在轨迹区域的编号方式上引入希尔伯特扫描算法进行编号，可以增强标签之间的空间关联性。
- (2) 基于数据集统计信息构建以路段为节点的路网图，在 STGRU 中使用路网结构的嵌入表示建模同一时空下用户之间的关联特征和增强路网的结构特征。
- (3) 在三个真实世界数据集上评估了空间特征增强方法的有效性。

4.2 基于空间特征增强的聚集预测模型

本节首先给出了聚集预测的定义，并介绍了基于空间特征增强的聚集预测模型，然后介绍了希尔伯特曲线进以及基于希尔伯特扫描算法生成区域标签的方法，最后阐述了基于统计的路网图构建方法和使用的图嵌入方法。

4.2.1 聚集预测问题定义及模型架构

首先对聚集预测问题进行定义：

假设有 M 个用户的轨迹信息，用户的集合表示为 $\mathbb{U} = \{u_1, u_2, \dots, u_M\}$ ，每个用户的轨迹信息表示为 $Traj_u = \{(lat_1, lon_1, t_1), (lat_2, lon_2, t_2), \dots, (lat_n, lon_n, t_n)\}$ ，其中 n 为每个用户的轨迹长度。于是，用户 u 在当前时刻 t 之前的历史轨迹信息可以表示为 $Ht_i^u = \{(lat_{t_1}^u, lon_{t_1}^u, t_1), (lat_{t_2}^u, lon_{t_2}^u, t_2), \dots, (lat_{t_{i-1}}^u, lon_{t_{i-1}}^u, t_{i-1})\}$ 。

区域划分方式为网格划分，所有区域的集合表示为 $Region = \{r_1, r_2, \dots, r_{a^2}\}$ ，其中 r_i 表示对应区域编号。用户 u 在当前时刻 t 之前的历史访问信息可以表示为 $Hr_i^u = \{r_{t_1}^u, r_{t_2}^u, \dots, r_{t_{i-1}}^u\}$ ，其中 $r_{t_i}^u$ 表示用户 u 在 t_i 时刻所在的区域。

通过聚集预测模型预测用户 u 在 t_i 时刻所在的区域，输出结果为用户在所有区域的可能性向量 $[s_{r_1, t_i}^u, s_{r_2, t_i}^u, \dots, s_{r_{a^2}, t_i}^u]$ 。对于用户 u ，在 t_i 时刻出现在区域 r' 的概率为对应的预测分数 s_{r', t_i}^u 。

根据所有用户轨迹的预测结果，可以得到 $M \times k$ 个用户分布的区域预测，其中 k 表示模型输出结果的前 k 个区域。对每个用户的前 k 个区域的预测结果进行归一化处理，保证每个用户的权重相等。用户 u 在 t_i 时刻可能出现在区域 r' 中的

概率为：

$$\hat{s}_{r',t_i}^u = \frac{s_{r',t_i}^u}{\sum_{j=0}^k s_{r_j,t_i}^u} \quad (4-1)$$

于是可以计算区域 r' 中在 t_i 时刻可能存在的人数 $number_{r',t_i}$ 。是否发生聚集则根据 $number_{r',t_i}$ 与阈值 η 的比较来判断。

$$number_{r',t_i} = \sum_{u \in U} \hat{s}_{r',t_i}^u \quad (4-2)$$

$$r'_{state} = \begin{cases} 1, & number_{r',t_i} \geq \eta \\ 0, & number_{r',t_i} < \eta \end{cases} \quad (4-3)$$

其中 r'_{state} 表示区域 r' 是否发生聚集，1 表示区域 r' 发生聚集，0 表示区域 r 没有发生聚集。 $number_{r',t_i}$ 则表示区域 r' 中在 t_i 时刻可能存在的人数。

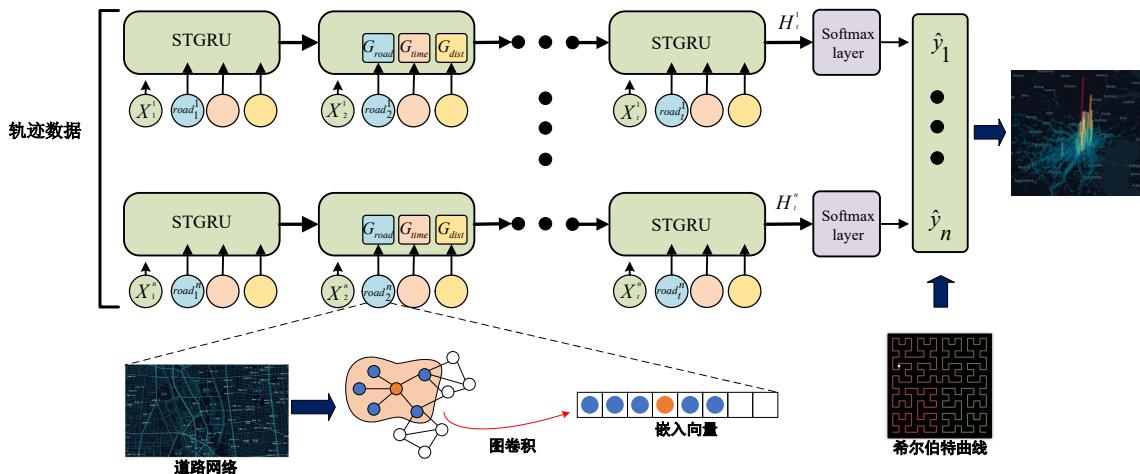


图 4-1 基于空间特征增强的聚集预测模型

如图4-1所示为本章提出的基于空间特征增强的聚集预测模型，使用单层 STGRU 模型对每个用户的位置进行预测，然后对同一时刻所有用户的位置预测结果融合得到得到聚集预测结果。在 STGRU 模型上堆叠了一个图卷积层提取路网图的节点嵌入向量作为路网门控的输入。其中路网图基于路网结构和用户分布情况构建，路网图的节点嵌入向量包含了路网的空间特征以及用户之间的分布特征。并使用希尔伯特扫描算法生成包含空间特征的区域标签，提升模型的时空特征提取能力。

4.2.2 希尔伯特曲线的区域标签生成

在本节中，首先分析了常用标签生成方法的问题，并介绍了希尔伯特曲线的定义及特性。然后阐述了使用希尔伯特扫描算法生成区域标签的方法。

4.2.2.1 常用标签生成方法存在的问题



图 4-2 名古屋市部分区域划分情况

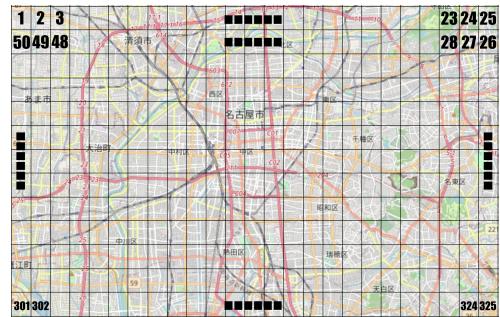


图 4-3 蛇形标签生成示例

在位置预测任务中，部分工作的目标是预测下一个轨迹点的经纬度，其损失函数为预测值与真实值之间的欧式距离，这类任务属于回归任务。其他工作是预测用户下一个可能出现的地点，如 POI 推荐，通过计算所有 POI 的概率进行预测，属于分类任务。本文中将聚集预测转换为基于区域的轨迹预测任务，输出为所有区域的概率值，本质上为多分类任务。POI 推荐任务中的标签包含兴趣点的语义特征^[64,65]，即使经过 one-hot 或者数值化操作，POI 标签仍然具有较强的指向型。而区域预测任务中的标签值并不具备语义，因此十分依赖于区域的划分方式以及标签生成的方法。

好的标签生成方法能够大幅减少模型建模难度，提升模型性能。差的标签生成方法会增加数据的复杂性，增加模型建模难度。以最常用的将蛇形编号为例。首先对本文中所使用的名古屋人流数据集^[66] 所覆盖的区域按前文中使用的参数进行划分，划分后总区域数量为 1786 个。为了更为清晰的展示，如图4-2所示截取了 $25 \times 13 = 325$ 个区域的局部图作为示例，其中每个区域对应一个编号。从左上角开始按蛇形递增，如图4-3所示。蛇形编号的方式符合直观的编号逻辑，但是忽视了区域之间的空间关系，隔行相邻的编号之间的空间关系难以表示，且随着划分区域大小的变化编号会随之发生较为复杂的变化。

在讨论区域编号方案的优劣性之前，需要对编号任务进行形式化表示。被编号区域是一个矩形或者正方形区域 R ，需要将其划分为 N 个子区域 $Regions = \{r_1, r_2, \dots, r_N\}$ ，对每个子区域 r_i 都关联到唯一的编号值。由于编号值具有唯一性，编号 $N_{regions} = \{1, 2, \dots, N\}$ 可以视作对区域 R 的压缩，从二维的地理空间压缩成

了一维的编号空间。由于编号是一维的，对被编号区域进行编号的过程就相当于一条线迂回穿过每一个子区域，再把线拉伸为得到一条直线，这条直线就是对应的编号空间。

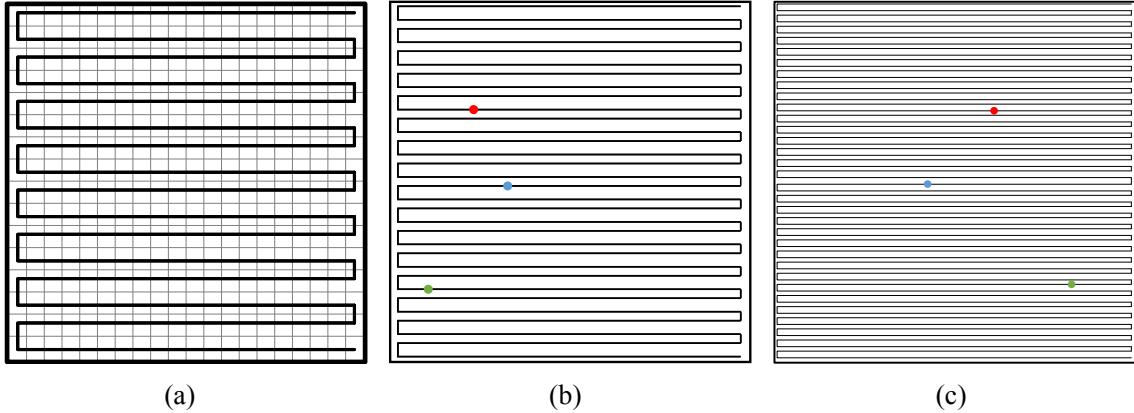


图 4-4 使用蛇形曲线进行编号。(a) 蛇形曲线填充 16×16 比例的正方形区域；(b) 蛇形曲线填充 32×32 比例的正方形区域，并标记了三个点；(c) 蛇形曲线填充 64×64 比例的正方形区域时被标记点的位置

最符合直觉的编号方案就是 Z 字形或者蛇形，此处以蛇形曲线为例，如图4-4(a)所示，一次对一行进行编号，在区域空间中交替推进。但是蛇形曲线存在一个严重的问题，当使用不同粒度的子区域进行区域划分时，编号空间中的许多点就会映射到地理空间中完全不同的部分，如图4-4(b)和(c)所示，这也就意味着编号与地理空间特征之间的映射较弱。

综上所述，编号方案的好坏取决于根据编号结果对被编号区域空间特征的保留程度，任意两个相邻数值的编号所关联的子区域的距离应该尽可能的小，这样即使对空间特征进行建模的时候存在一定误差，两个值相近的编号也会指向空间中邻近的区域。

4.2.2.2 希尔伯特曲线

希尔伯特曲线 (Hilbert curves, HC)^[67] 是一族无限多条的曲线，特定阶数的希尔伯特曲线被称为伪希尔伯特曲线 (Pseudo Hilbert curves, PHC)，1 阶和 2 阶的希尔伯特曲线如图4-5和4-6所示。希尔伯特曲线的定义为：

1) 希尔伯特曲线是良定义的 (well-defined)，伪希尔伯特曲线上的点是收敛的，对于所有 x ，对应的 $HC_n(x)$ 是收敛的。

$$\forall t \in [0, 1], HC_n(t) = \lim_{n \rightarrow \infty} PHC_n(t) \quad (4-4)$$

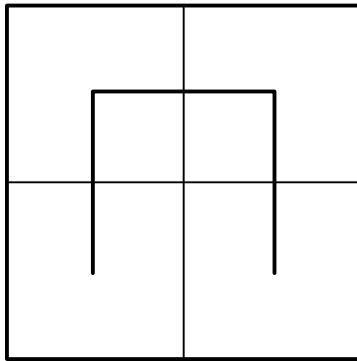


图 4-5 1 阶希尔伯特曲线

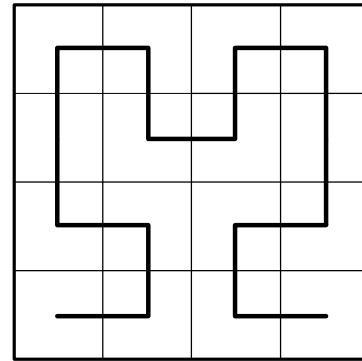


图 4-6 2 阶希尔伯特曲线

2) 希尔伯特曲线是连续的。对于值域中的点任意点 (x, y) ，选择一个任意小的 ε 邻域，都可以在其中找到更小的 $1/4^k \times 1/4^k$ 的闭区域，对应到定义域是一个闭区间 I ，可以在 I 中找到更小的 δ 开区间 I' ， I' 中所有点都会映射到值域的 ε 邻域中。

3) 希尔伯特曲线能够填充整个空间，即单位正方形中每个点都是希尔伯特曲线的输出。希尔伯特曲线 $HC(t)$ 是从实数 $[0, 1]$ 区间到实数平面 $[0, 1] \times [0, 1]$ 区域上的满射。也就是说给定值域中任意点 (x_i, y_i) ，都能找到对应参数 t_i 使得 $HC(t_i) = (x_i, y_i)$ 。

空间填充曲线的最大优势在于连续性，即输入平稳变化时，输出空间中不会发生跳跃。希尔伯特曲线除连续性之外还具有良定义，即给定一个输入点，随着希尔伯特曲线阶数的上升，对应的输出趋于空间中的某个点。因此使用希尔伯特曲线进行区域编号能够较大程度的满足相近编号区域在空间中邻近，保留编号之间的空间特征。可以证明希尔伯特曲线可以定义出极限曲线，而前文中的蛇形曲线就无法定义出极限曲线。

4.2.2.3 使用希尔伯特扫描算法生成区域编号

在对数据集中覆盖区域进行编号时发现大多数情况下编号区域是矩形区域。根据希尔伯特曲线能够填充整个空间的特性，只要子区域的大小足够小，理论上是可以使用标准希尔伯特曲线进行填充，但不能满足所有使用场景。在数字图像处理领域，Kamata 等人^[68,69] 和 Zhang 等人^[70] 先后提出了多种希尔伯特扫描算法，首先使用将矩形块划分为块的分割方法，然后基于查找表的方法提出每个块中地址的配置方法。

在对矩形区域进行编号时，需要先生成自定义大小的矩形区域的希尔伯特曲线填充数组，再根据经纬度及划分粒度计算出待编号区域的下标，并根据希尔伯特曲线填充数组中的对应值进行编号。本章实现了二维希尔伯特扫描算法

generate2d, 伪代码如算法4-1所示, 其中 sgn 函数为符号函数:

$$\text{sgn}(x) = \begin{cases} 1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0 \end{cases} \quad (4-5)$$

算法 4-1 二维希尔伯特扫描算法 generate2d

Data: (x, y, ax, ay, bx, by)
 其中 x, y 表示初始位置坐标, ax, ay, bx, by 分别用于确定区域分割线的位置
 第一次调用时的参数为: $x = 0, y = 0, ax = width, ay = 0, bx = 0, by = height$

Result: 编号后的二维数组 \mathcal{A}

```

1 begin
2    $w = |ax + ay|, h = |bx + by|$ 
3   计算希尔伯特曲线填充时的主方向  $(dax, day) = (\text{sgn}(ax), \text{sgn}(ay))$ 
4   正交方向为  $(dax, day) = (\text{sgn}(bx), \text{sgn}(by))$ 
5   if  $h == 1$  then
6     foreach  $i \in [0, w)$  do
7       yield  $(x, y) = (x + dax \times i, y + day \times i)$ 
8       return
9     end
10    if  $w == 1$  then
11      foreach  $i \in [0, h)$  do
12        yield  $(x, y) = (x + dbx \times i, y + dby \times i)$ 
13        return
14      end
15       $(ax_2, ay_2) = (ax // 2, ay // 2), (bx_2, by_2) = (bx // 2, by // 2)$ 
16       $w_2 = |ax_2 + ay_2|, h_2 = |bx_2 + by_2|$ 
17      if  $2w > 3h$  then
18        分为两个部分进行递归填充
19        yield  $\text{generate2d}(x, y, ax_2, ay_2, bx, by)$ 
20        yield  $\text{generate2d}(x + ax_2, y + ay_2, ax - ax_2, ay - ay_2, bx, by)$ 
21      else
22        标准情况: 分为上、下、水平三部分
23        yield  $\text{generate2d}(x, y, bx_2, by_2, ax_2, ay_2)$ 
24        yield  $\text{generate2d}(x + bx_2, y + by_2, ax, ay, bx - bx_2, by - by_2)$ 
25        yield  $\text{generate2d}(x + (ax - dax) + (bx_2 - dbx), y + (ay - day) +$ 
26           $(by_2 - dby), -bx_2, -by_2, -(ax - ax_2), -(ay - ay_2))$ 
27    end
28 end

```

4.2.3 路网图构建及轨迹点匹配

路网结构信息与用户的分布状态存在十分直观的联系^[71]，为了能够更充分利用路网结构信息以及建模用户之间的关联性，需要充分考虑道路之间的拓扑结构^[72]。路网拓扑结构通过路网图表示，构建路网图^[73]最直观的方式就是将道路交叉点视为节点，道路视为图的边，这样的构建方式更侧重于交通情况的建模，不符合本章构建路网图的出发点。受 Jiang 等人^[74]的启发，利用图对偶的方式，用节点表示道路，道路之间的连通性为边构建路网图。

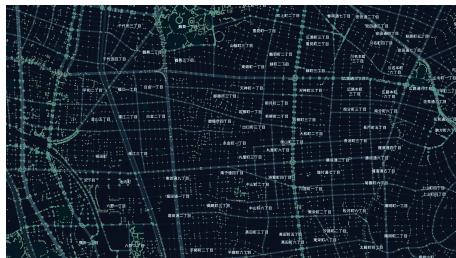


图 4-7 部分道路节点图



图 4-8 部分路网图

如图4-7和图4-8所示，分别为名古屋市道路网络中部分节点信息和道路网络的可视化，在地理信息领域^[75]通常采用 GeoJson 进行表示，在 GeoJson 中的道路由 string 类型组成，称之为路段。每个路段通常为一条直线，由多个离散的节点按顺序连接得到，路段之间的连接关系则表示为头尾节点的连接。因此，可以通过判断两个路段是否存在重叠节点判断连接与否。在地理信息领域中的道路有国道、省道、县道等級別，长度也是在数十公里到数百上千公里不等。如果将一条道路视为一个节点，那么得到的路网图会是一个节点较少且节点度较大的稠密图，这很难对用户在城市场景下的局部路网结构进行表示。

本章采用路段为节点，路段之间的连通关系为边的方式构建路网图，即两个有公共节点的路段之间视为存在边，主要原因基于路段的路网图能够更好的与用户分布轨迹点进行匹配。基于路段的路网图节点数量较多，以名古屋市为例，节点数约为 1,500,000 个，东京市路网中的路段数甚至达到了 2,500,000 个，且每个节点边数较为稀少，能够更精确的表示用户的分布信息和更方便建模局部区域的路网结构。于是路网图可以定义为 $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{A})$ ，其中 $\mathcal{V} = \{v_0, v_1, \dots, v_n\}$ 是由 n 个节点构成的集合，包含了路段的 ID、名称、类别等信息， \mathcal{E} 是图中所有边的集合， \mathcal{A} 是一个 $1 \times n$ 的向量，表示每个节点上存在的用户数量。

由于构建的路网图中包含用户的分布信息，每个时刻路网图节点上的值是变化的。根据所用人流数据集的采样时间，每个数据集共会产生 1728 个路网图快照与轨迹的采样时间一一对应。且为了防止用户分布信息统计对模型训练带来的标

签泄露问题，仅对训练集和验证集中的轨迹点数据进行统计。

由于采集设备的精度问题以及当前路网的表示原因，数据集中的大部分轨迹点与路网节点之间是存在一定偏差的，因此需要将轨迹点与路网图中的路段节点进行一一对应，先计算轨迹点与路网节点之间的距离，再取距离最近轨迹点与路网节点进行匹配。由于路网图节点数量众多，暴力匹配算法的时间复杂度将会指指数级增加，为了减少匹配过程中的运算量，本章提出了基于哈希思想的轨迹点匹配算法。

基于哈希思想的轨迹点匹配算法先根据所有待匹配轨迹点的经纬度确定完整覆盖区域并对覆盖区域进行粗粒度划分，再将所有路网图节点坐标分配到对应的粗粒度区域中，对每个待匹配轨迹点根据其经纬度确定对应区域，并计算与该区域中所有路网图节点的距离并排序，具体如算法4-2所示。

算法 4-2 基于哈希思想的轨迹点匹配算法

```

Data: 轨迹点坐标  $(x, y)$ ;  

所有路网图节点坐标及编号  $\{(x_1, y_1, l_1), (x_2, y_2, l_2), \dots, (x_n, y_n, l_n)\}$ ;  

路网图节点坐标的极值  $\text{Lat}_{max}, \text{Lat}_{min}, \text{Long}_{max}, \text{Long}_{min}$ ;  

区域划分粒度  $\delta$   

Result: 与轨迹点匹配的节点编号  $\mathcal{L}$ 
1 begin
2   初始化  $\lceil (\text{Lat}_{max} - \text{Lat}_{min}) \div \delta \rceil \times \lceil (\text{Long}_{max} - \text{Long}_{min}) \div \delta \rceil = \Theta$  个数
   组  $Array = \{A_1, A_2, \dots, A_\theta\}$ 
3   while  $\tau \in (0, n]$  do
4     | 计算每个节点对应的数组并修改对应数组
5     | 使用粒度  $\delta$  和最大经纬度  $\text{Lat}_{max}, \text{Long}_{max}$  计算对应的数组编号  $\theta$ 
6     | 修改对应数组  $A_\theta$  中的值，增加对应节点信息
7   end
8   计算轨迹点坐标对应的数组编号  $\theta'$ 
9   foreach  $(x_\tau, y_\tau) \in A_{\theta'}$  do
10    | 计算轨迹点  $(x, y)$  与  $(x_\tau, y_\tau)$  的距离，在对应数组  $A_{\hat{\theta}}$  中保存距离值和
       | 编号  $l_\tau$ 
11  end
12  对数组  $A_{\hat{\theta}}$  中的值进行排序
13  输出最短距离的节点编号  $\mathcal{L}$ 
14 end

```

设轨迹点数为 m 个，路网图节点数为 n 个，暴力匹配算法需要计算 $m \times n$ 次节点距离，每次计算两个节点之间的距离需要两次减法、两次乘法运算、一次加法运算以及一次开方运算，并需要对每个轨迹点与所有节点的距离进行排序，以快速排序为例，算法复杂度为 $O(n \log n)$ ，总的算法复杂度为 $O(m \times n^2 \log n)$ 。优

化后, 根据哈希表长度 k , 每个轨迹点匹配时仅需要与 $\frac{n}{k}$ 个节点进行计算, 总需计算 $\frac{mn}{k}$ 次节点距离, 每次排序比较次数约为 $\frac{n}{k}(\log n - \log k)$, 总的算法复杂度为 $O(\frac{mn^2}{k^2}(\log n - \log k))$, 大幅降低了轨迹点匹配的计算复杂度。

4.2.4 基于图卷积网络的路网节点嵌入

标准的卷积网络并不能在图等结构化数据上使用, 目前的解决方法主要有两种。一种是在空域上对卷积进行扩展^[76], 将顶点变换到欧几里得空间, 可以通过正常的卷积操作进行处理。另一种是在谱域空间进行图傅里叶变换^[77], 借鉴了信号处理领域的傅立叶变化及信号卷积定理, 将图转化为谱域空间中的点积来实现。目前图卷积方法均仅适用于无向图 $G = (V, E)$ 。

基于频域的图卷积模型引入了图卷积算子 $*_{\mathcal{G}}$ 的概念作为图信号 $x \in \mathbb{R}^n$ 与卷积核 Θ 之间的乘法:

$$\Theta *_{\mathcal{G}} x = \Theta(L)x = \Theta(U\Lambda U^T)x = U\Theta(\Lambda)U^T x \quad (4-6)$$

其中图傅里叶基 $U \in \mathbb{R}^{n \times n}$ 是图的归一化拉普拉斯矩阵 L 的特征向量矩阵。 $L = I_n - D^{-\frac{1}{2}}AD^{-\frac{1}{2}} = U\Lambda U^T \in \mathbb{R}^{n \times n}$, I_n 的单位矩阵, $D \in \mathbb{R}^{n \times n}$ 是图的度矩阵, A 是图的邻接矩阵, Λ 是 L 特征值的对角矩阵。根据图卷积算子的定义, 一个图信号 x 通过卷积核 Θ 与图傅里叶变换后的 $U^T x$ 的乘法实现过滤^[78]。

未进行优化的图卷积模型计算存在一些弊端, 每次向前传播都需要计算三个矩阵的乘积, 算法复杂度较高, 且局部性较弱。为了解决计算复杂度过高的问题, Kipf 等人^[79] 提出使用切比雪夫多项式的近似算法 ChebNet。

ChebNet 实现了滤波器的局部化并优化了参数数量。通过将图卷积核 Θ 中的 Λ 限制为多项式, $\Theta(\Lambda) = \sum_{k=0}^{K-1} \theta_k \Lambda^k$, 其中 $\theta \in \mathbb{R}^K$ 是一个多项式系数的向量, K 是图卷积核大小, 决定了每次算计所覆盖的最大半径。

切比雪夫多项式为: $T_k(X) = 2xT_{k-1}(X) - T_{k-2}(X)$, 其中 $T_0(X) = 1$, $T_1(X) = X$ 。使用切比雪夫多项式的 $K-1$ 阶截断来近似图卷积核, 重新调整 $\tilde{\Lambda} = 2\Lambda/\lambda_{max} - I_n$ 后得到 $\Theta(\Lambda) \approx \sum_{k=0}^{K-1} \theta_k T_k(\tilde{\Lambda})$, λ_{max} 表示 L 的最大特征值^[80]。图卷积公式可以重写为:

$$\Theta *_{\mathcal{G}} x = \Theta(L)x \approx \sum_{k=0}^{K-1} \theta_k T_k(\tilde{L})x \quad (4-7)$$

其中 $T_k(\tilde{L}) \in \mathbb{R}^{n \times n}$ 是使用切比雪夫 k 阶近似后的拉普拉斯矩阵 $\tilde{L} = 2L/\lambda_{max} - I_n$ 。

Kipf 等人^[79] 基于神经网络的缩放和归一化在一阶 ChebNet 的基础上进一步

近似。假设 $\lambda_{max} \approx 2$, 图卷积公式可简化为:

$$\Theta *_{\mathcal{G}} x \approx \theta_0 x + \theta_1 \left(\frac{2}{\lambda_{max}} L - I_n \right) x \quad (4-8)$$

$$\approx \theta_0 x - \theta_1 (D^{-\frac{1}{2}} A D^{-\frac{1}{2}}) x \quad (4-9)$$

其中 θ_0 和 θ_1 是卷积核中的两个共享参数。为了约束参数和稳定性数值性能将其设置为统一参数 $\theta = \theta_0 = -\theta_1$, 并对 A 进行重新规范化为 $\tilde{A} = A + I_n$, 于是可以进一步简化为:

$$\Theta *_{\mathcal{G}} x = \theta (I_n + D^{-\frac{1}{2}} A D^{-\frac{1}{2}}) x \quad (4-10)$$

$$= \theta (D^{-\frac{1}{2}} \tilde{A} D^{-\frac{1}{2}}) x \quad (4-11)$$

图卷积网络模型在各类图嵌入表示任务上取得了很好的表现, 本章使用的图卷积网络模型为一阶近似的 ChebNet 生成路网图中对应节点的嵌入表示。图卷积网络由于其计算方式的限制, 仅适用于静态图的计算, 能够较好的对路网图中节点进行嵌入表示, 且最大程度保留路网的拓扑结构以及邻接节点中的用户数等特征, 如图4-9所示。

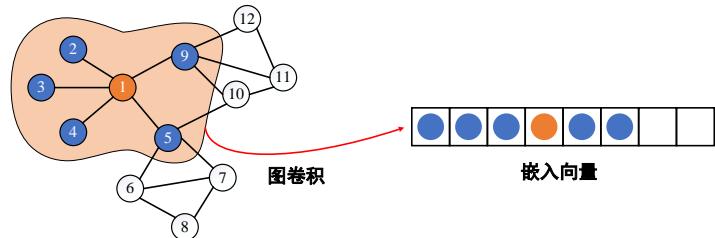


图 4-9 使用 GCN 生成路网节点嵌入

使用生成的路网节点嵌入表示作为道路节点信息 $road_t$, 使用 GCN 提取路网图空间结构特征和用户分布特征后的整体模型架构, 通过 STGRU 层和 GCN 层的堆叠来实现。路网中的节点由向量表示, 且会生成不同维度特征向量。将输入向量定义为 $X \in \mathbb{R}^{n \times C_i}$, 输出向量维度定义为 C_o 。于是 $road_t$ 的计算公式修改为:

$$road_t = \sum_{i=1}^{C_i} \Theta_{i,j}(L) x_i \in \mathbb{R}^n, 1 \leq j \leq C_o \quad (4-12)$$

其中 X 为 t 时刻轨迹点所在的路网节点。

4.3 实验与分析

本节首先设计多组对比实验和消融实验验证了本章提出的空间特征增强方法以及各模块的有效性，然后通过应用分析得到不同长度数据在聚集预测任务中的性能表现。

4.3.1 实验配置

本章所使用的实验环境配置、数据集、评价指标以及基线方法均第三章一致，详见章节3.3.1。

4.3.2 对比实验

方法比较。表4-1、4-2和4-3中给出了SEGPM模型、STGRU模型和六个基线模型在三个数据集上的性能表现。在对比实验中，所使用的参数设置参照章节3.3.3所示，GCN嵌入向量维度设置为8，目的是维持模型参数量的一致性，减少由参数量变化带来的影响。

表 4-1 SEGPM 与基线模型在 NPF 数据集上 Recall@K 和 AUC 指标的结果

NPF	recall@1	recall@5	recall@10	recall@20	AUC
LSTM	0.0428	0.1677	0.2894	0.4300	0.6951
GRU	0.0712	0.2372	0.3421	0.4771	0.7778
RNN	0.0809	0.2580	0.3570	0.4865	0.8018
ST-LSTM	0.0621	0.2438	0.3663	0.5050	0.7976
STGN	0.0762	0.2728	0.3734	0.5061	0.8177
STGRU	0.0920	0.2829	0.3891	0.5231	0.8290
SEGPM	0.1002	0.2966	0.3936	0.5256	0.8413

表 4-2 SEGPM 与基线模型在 OPF 数据集上 Recall@K 和 AUC 指标的结果

OPF	recall@1	recall@5	recall@10	recall@20	AUC
LSTM	0.0383	0.1638	0.2634	0.4375	0.7140
GRU	0.0455	0.1979	0.3007	0.4656	0.7611
RNN	0.0588	0.2258	0.3260	0.4881	0.8324
ST-LSTM	0.0502	0.2107	0.3174	0.4890	0.7817
STGN	0.0633	0.1699	0.2833	0.4920	0.8292
STGRU	0.0681	0.2439	0.3464	0.5116	0.8545
SEGPM	0.0835	0.2509	0.3542	0.5207	0.8538

表 4-3 SEGPM 与基线模型在 TPF 数据集上 Recall@K 和 AUC 指标的结果

TPF	recall@1	recall@5	recall@10	recall@20	AUC
LSTM	0.0752	0.3331	0.4849	0.6428	0.8317
GRU	0.0789	0.3319	0.4793	0.6384	0.8451
RNN	0.0856	0.3634	0.5066	0.6561	0.8645
ST-LSTM	0.0864	0.3699	0.5213	0.6682	0.8727
STGN	0.0900	0.3327	0.5103	0.6672	0.8734
STGRU	0.0933	0.3795	0.5263	0.6730	0.8755
SEGPM	0.1090	0.4060	0.5589	0.6993	0.8804

如表中实验结果所示，本章提出的 SEGPM 模型在三个数据集上的绝大多数指标均优于基线模型。在 NPF、OPF、TPF 三个数据集上，SEGPM 模型 *Recall@1* 指标的提升依次为 8.9% - 134.1%，22.6% - 118% 和 16.8% - 44.9%。实验结果表明，SEGPM 模型中对路网结构进行嵌入表示以及基于希尔伯特曲线的标签生成方法能够大幅提升模型建模时空特征的能力以及对聚集预测任务的有效性。因为这两种方式均有效增强了模型的时空特征提取能力，带来了一定的性能提升。同时路网结构的嵌入表示中考虑了用户之间的分布特征，提升了聚集预测建模多用户行为模式的能力。

基于希尔伯特曲线的标签生成方法保留了标签之间的空间特征，极大程度降低了标签数值之间的稀疏性问题，大幅降低了 SEGPM 模型对标签数值拟合的复杂度。对单用户进行建模的轨迹预测模型难以考虑到对应的其余用户的分布对用户行为模式的影响，在聚集预测任务中并不完全适用。路网结构嵌入时考虑了用户的分布情况，能够更准确的建模多用户场景下的行为模式。基于空间特征增强的聚集预测模型能够有效地对多用户之间的相互影响。

4.3.3 消融实验

为了验证本章提出的使用希尔伯特曲线的标签生成方法和基于 GCN 的路网结构嵌入方法的有效性，本节设置了两组使用单个方法的消融实验，并在三个数据集上都进行了实验，实验结果如表4-7、4-5和4-6所示。其中 $STGRU_1 = STGRU + HC$ 为仅使用希尔伯特曲线进行区域编号， $STGRU_2 = STGRU + GCN$ 表示仅使用 GCN 生成路网图的节点嵌入。

由实验结果可知，本章提出的使用希尔伯特曲线和 GCN 嵌入增强 STGRU 模型的空间特征方法在三个数据集的绝大多数指标上取得了较为明显的提升。以 *Recall@1* 指标为例， $STGRU+HC+GCN$ 模型相较于基线模型 STGN 模型性能提升

分别为 5.2% - 27.03%，22.6% - 31.9% 和 16.8% - 21.1%。结果表明，单独使用希尔伯特曲线生成区域编号或 GCN 生成路网图的节点嵌入，均能够较好的增强对数据集中空间特征的增强。同时考虑上述两者时的模型性能提升较为明显。

表 4-4 路网特征嵌入和标签特征增强在 NPF 数据集上的性能

NPF	recall@1	recall@5	recall@10	recall@20	AUC
STGN	0.0762	0.2728	0.3734	0.5061	0.8177
STGRU	0.0920	0.2829	0.3891	0.5231	0.8290
STGRU ₁	0.0974	0.2867	0.3881	0.5161	0.8356
STGRU ₂	0.0968	0.2881	0.3891	0.5179	0.8259
SEGPM	0.1002	0.2966	0.3936	0.5256	0.8413

表 4-5 路网特征嵌入和标签特征增强在 OPF 数据集上的性能

OPF	recall@1	recall@5	recall@10	recall@20	AUC
STGN	0.0633	0.1699	0.2833	0.4920	0.8292
STGRU	0.0681	0.2439	0.3464	0.5116	0.8545
STGRU ₁	0.0819	0.2436	0.3516	0.5193	0.8519
STGRU ₂	0.0700	0.2458	0.3516	0.5158	0.8542
SEGPM	0.0835	0.2509	0.3542	0.5207	0.8538

表 4-6 路网特征嵌入和标签特征增强在 TPF 数据集上的性能

TPF	recall@1	recall@5	recall@10	recall@20	AUC
STGN	0.0900	0.3327	0.5103	0.6672	0.8734
STGRU	0.0933	0.3795	0.5263	0.6730	0.8755
STGRU ₁	0.1010	0.3780	0.5326	0.6646	0.8783
STGRU ₂	0.1039	0.3776	0.5308	0.6651	0.8835
SEGPM	0.1090	0.4060	0.5589	0.6993	0.8804

且在不同数据集上两者的性能提升有所差别。在 NPF 和 OPF 数据集中，使用希尔伯特曲线的性能提升相较于路网图节点嵌入更大，原因是这两个数据集中用户数量较少，轨迹点较为稀疏，路网图节点嵌入对整体用户行为模式的建模影响较小。路网图中包含了当前分户分布信息，较为稀疏的轨迹点分布对建模多用户之间的行为模式的影响减弱。而 TPF 数据集中轨迹点分布更加密集，用户分布情况的建模对当前用户的行为模式影响较大，因此使用路网图节点嵌入的 STGRU

模型性能较仅增强区域编号之间的空间特征有更好的性能表现。

两种方法同时使用的性能表现优于单独考虑其中一种方法。这是因为使用希尔伯特曲线进行区域编号能够增强预测区域间的空间特征，更有助于提升 $Recall@1$ 性能。在 OPF 数据集上体现的较为明显， $STGRU_1$ 和 $STGRU_2$ 在其余指标上相差无几，但 $STGRU_1$ 的 $Recall@1$ 提升了 20.3%，提升十分明显。而使用 GCN 模型生成路网图节点嵌入与长短期路网特征的提取相结合，能够更好的提取用户之间的时空特征，建模多用户场景下的用户行为模式。两种方法对模型性能提升的作用域重叠较少，几乎不存在性能相互抵消或冲突的现象，因此同时考虑两种方法能够相较于单独使用任一方法均有所提升。

向量维度影响。图卷积网络生成节点嵌入向量的维度对的嵌入表示的性能影响较大，于是设置了使用不同维度节点嵌入向量的 STGRU+GCN 模型进行实验，通过生成并使用不同维度的节点嵌入为 8, 16, 32, 64 来完成。从表4-7可以发现，随着向量维度的增加各项性能指标均有所提升。当向量维度增加到 64 时，出现了性能的整体下滑，且下降明显，原因是当向量维度较大时，路网门控结构不能很好的提取和建模向量中的路网结构特征，甚至有一定程度的反作用。

表 4-7 使用不同维度节点嵌入向量的性能

Vector Dimension	recall@1	recall@5	recall@10	recall@20
8	0.1039	0.3776	0.5308	0.6651
16	0.1054	0.3757	0.5300	0.6707
32	0.1081	0.3903	0.5405	0.6800
64	0.1037	0.3806	0.5387	0.6735

4.3.4 应用分析

聚集预测对轨迹数据的质量要求较高，在分布时间相同的轨迹中，不同轨迹长度对建模用户行为模式影响较大。本小节通过设置不同长度的轨迹数据进行实验，一共在三个数据集上设置了 14 组实验，所使用的轨迹长度分别为 $(0, 50)$ 、 $[50, 100)$ 、 $[100, 150)$ 、 $[150, 200)$ 以及 $[200, \infty)$ 。由于 NPF 数据集中长度在 200 以上的数据极少，因此在 NPF 数据集上仅设置了 4 组实验，其余两个数据集上均为 5 组实验。如图4-10所示为基于 NPF 数据集的 4 组实验结果，实验设置与前文一致，评价指标为 $recall@k$ ，其中 $k = \{1, 5, 10, 20\}$ 。在所有的四个指标上 100 ~ 150 区间和 150 ~ 200 区间的性能表现最好。此外，轨迹长度小于 50 时的性能表现较好，因为 NPF 数据集中该区间的轨迹数据较少，大大降低了建模难度。

如图4-11所示为 OPF 数据集上的 5 组实验结果，其中性能表现最好的区间为

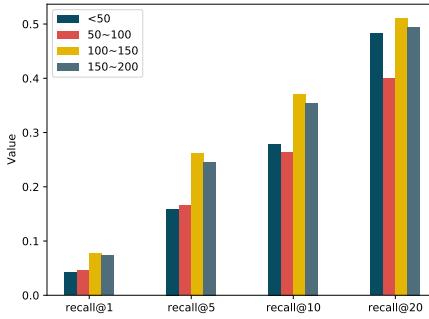


图 4-10 NPF 中的性能表现

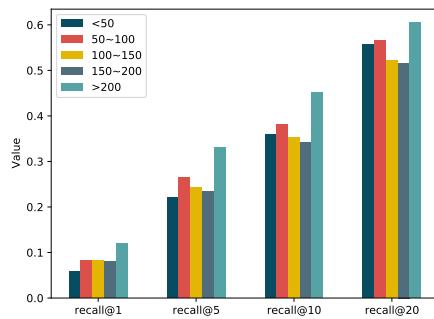


图 4-11 OPF 中的性能表现

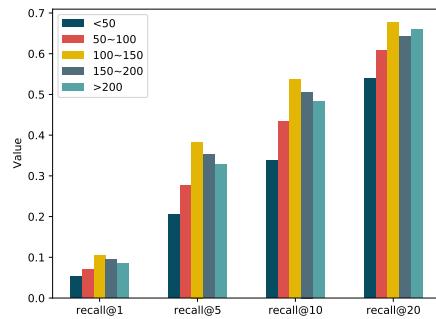


图 4-12 TPF 中的性能表现

50~100，其余区间稍弱。在轨迹长度大于 200 时性能指标高于前文中对比实验的结果，同样为数据量较少所致。

最后是在 TPF 数据集上的实，如图4-12所示，性能表现从高到低分别为 100~150 区间、150~200 区间、大于 200 区间、50~100 区间和小于 50 区间。其中轨迹长度大于 200 的 *recall@20* 指标仅低于 100~150 区间，说明了轨迹长度对粗粒度的预测结果存在正向驱动。

综合上述实验，相同分布时间的不同长度的轨迹数据意味着完全不同的行为模式以及样本质量，如果用户的轨迹较短，意味着该用户的移动次数较少，虽然在建模用户行为模式时较为困难，但这类用户更偏向于在某处长期停留，对整体结果影响较小。反之当用户轨迹过长时，模型精度并不足以对其意图进行精确预测，反而会对整体的聚集预测造成影响。因此在聚集预测中，使用移动次数较多并且由一定停留的轨迹数据更容易进行预测。

4.4 本章小结

在本章中，通过增强轨迹数据中的空间特征，在第三章 STGRU 模型的基础上提出了一个聚集预测模型。在聚集预测模型中，融合了希尔伯特曲线增强标签之间的空间特征和；然后引入图嵌入结构建模路网图的结构信息以及用户之间的相互影响，这对于建模多用户行为模式有很大的帮助；使用希尔波特曲线增强区域标签能够增强标签之间的语义关系，降低模型建模的难度；图嵌入结构所生成的向量作为输入的一部分并通过路网门控进行建模，且其中包含了用户之间的分布信息，能够建模用户在不同用户分布情况下的行为模式。最后在三个真实世界的数据集上进行多项实验证明聚集预测模型的有效性，它在第三章 STGRU 模型的基础上取得了较好的性能提升。

第五章 基于城市流量的聚集预测告警系统实现

为了帮助更好的掌握城市中的聚集情况，为城市中对聚集情况的管理管控提供支撑，本章介绍了聚集预测告警系统的设计与实现，并进行了功能和性能测试。

5.1 引言

随着智慧城市建设的推进和各类智能设备的推广，城市管理能够采集到用户在公共场所的位置信息与日俱增，这些数据中包含用户的移动轨迹、分布情况等。在人口密集的现代化城市中，由于人口聚集导致的各类恶性事件是城市管理者一直面临的难题。在人工智能、大数据技术不断发展的当下，如何利用好这些数据帮助城市管理者掌握城市中的聚集情况并基于此做出合理反映，能够极大程度提升城市的管理管控能力。在该任务中，如何提取数据中的时空特征对用户行为模式进行建模，实现更精确的用户位置预测是需要解决的核心问题。

在本文三、四章中提出的聚集预测模型的基础上，结合实际场景下的轨迹数据分析和聚集预测需求，本章实现了基于城市流量的聚集预测告警系统的设计与开发。首先，结合任务进行需求分析和系统的整体设计。然后，简单介绍了系统的实现，并通过可视化方案对系统的功能和界面布局进行展示。最后，对系统的功能和性能进行了测试。

5.2 系统概述

本章中设计实现的聚集预测告警系统整体采用 B/S 架构，web 端开发及 UI 设计基于 HTML5、CSS3 和 Kepler.gl 库实现，后端采用 python 的 Falsk 框架，相关服务部署环境如表3-1所示，数据库采用 SQLite。主要开发工具为 Microsoft 公司的 Visual Studio Code 软件。

基于城市流量的聚集预测告警系统在前两章所提出的聚集预测模型上进行开发，主要面向的是城市场景下的聚集预测问题。系统包括数个前端界面、一个由 Flask 和 SQLite 实现的后端数据处理程序以及一个基于 tensorflow 的模型实现，基于 Unix 操作系统开发实现。

5.3 需求分析

本节对聚集预测告警系统进行简单的需求分析，并通过系统用例图和数据流图进行需求建模分析。该系统需要实现数据展示和基于模型推理的聚集预测结果

呈现，对其进行分析后绘制了如图5-1所示为聚集预测告警系统的系统用例图，主要用例为数据选择、数据处理、数据传输、聚集预测、数据存储和结果展示等。

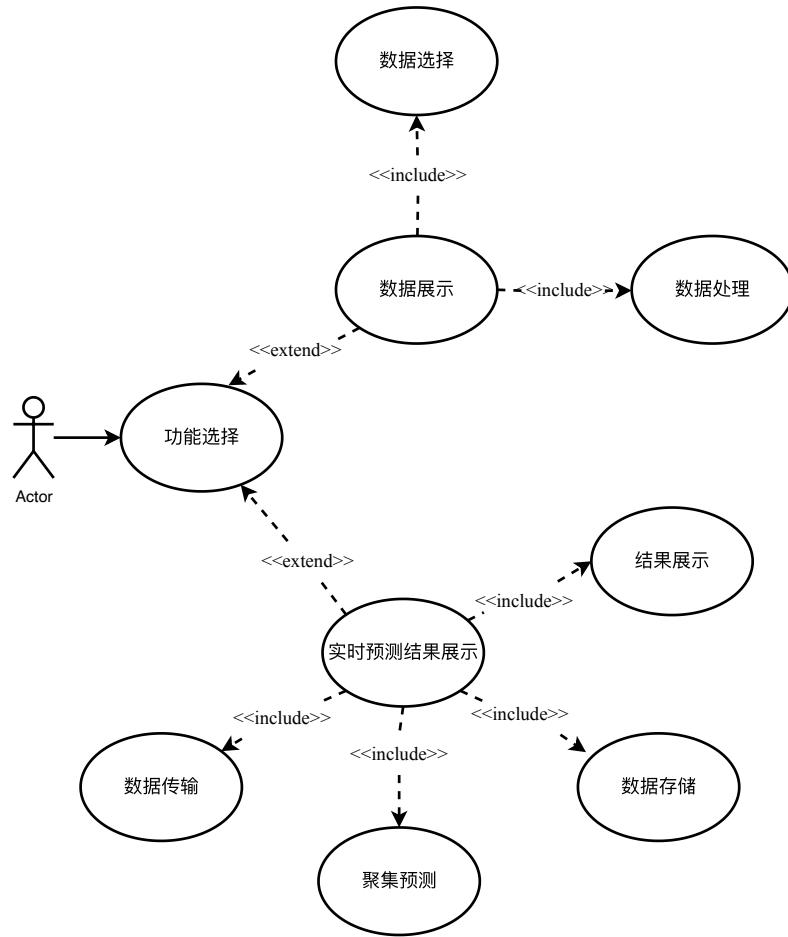


图 5-1 聚集预测告警系统用例图

如图5-2所示为经过分析后得到的本系统数据流图。其中的数据展示事务，主要由资源管理模块对用户所选择的关键字通过数据库查询得到对应的轨迹数据和路网数据，然后根据预设的模型在前端界面上进行展示，并支持用户对数据的可视化交互。对于聚集预测结果展示事务，资源管理模块控制数据传输接口的开放，数据传输和处理模块实时接收并结合历史轨迹信息生成样本，样本数据由聚集预测模块进行预测推理，所得到的结果在前端界面上进行展示，同时将预测结果存入数据库中。本章开发的聚集预测告警系统功能主要包括：

- (1) 对用户选择的轨迹数据和路网数据进行可视化展示。
- (2) 实时接收移动端传输的数据。
- (3) 实现对实时数据进行聚集预测。

(4) 对实时数据和预测结果进行可视化展示。

(5) 将轨迹数据和预测结果存入数据库。

(6) 在前端界面中提供可视化交互能力。

在性能需求方面，本章提出的聚集预测预警系统需要满足同时接收多设备数据，模型推理耗时应在小于轨迹采样的时间间隔（通常为 5 分钟），且聚集预测的结果需要达到一定精度。

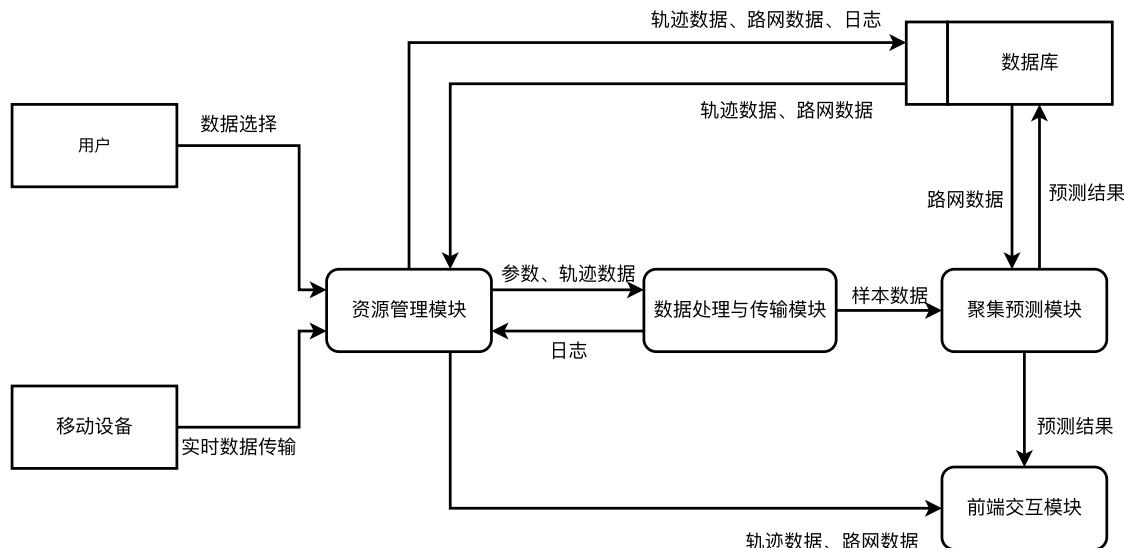


图 5-2 聚集预测预警系统数据流图

聚集预测预警系统整体为四层架构，分别为表现层、业务层、服务层和数据库。其中表现层主体为 UI 界面，基于 HTML5 和 CSS3 语言开发，在页面中通过基于 Flask 的 Dash 框架嵌入了 Uber 公司开发的 Kepler.gl 库实现简单的交互能力；业务层包括计算资源调度、数据处理和异常处理，实现了除聚集预测模型外的所有计算功能；服务层主要提供对外的输出传输接口和模型调用接口，实现基于移动终端设备的试试数据获取以及充分利用远程计算资源；数据库中主要存储系统所产生的日志信息、历史轨迹信息以及路网结构信息；

从功能划分角度出发，系统主要包含四个功能模块，如图5-3所示，分别是资源管理模块、数据传输和处理模块、聚集预测模块以及前端交互模块。

1、资源管理模块。主要包括流程控制、计算资源的管理、异常的捕获与处理，负责整个系统运行过程中的资源管理调度，保障系统的正常运行，并具备处理系统运行时可能发生的多种异常的能力。

2、数据传输和处理模块。包括数据传输、数据处理、日志数据管理和数据库读写等，其中数据传输主要为提供并运行数据接口，通过端口监听实现与移动终端之间的信息传输功能；以及在数据样本生成后通过模型调用接口与聚集预测模

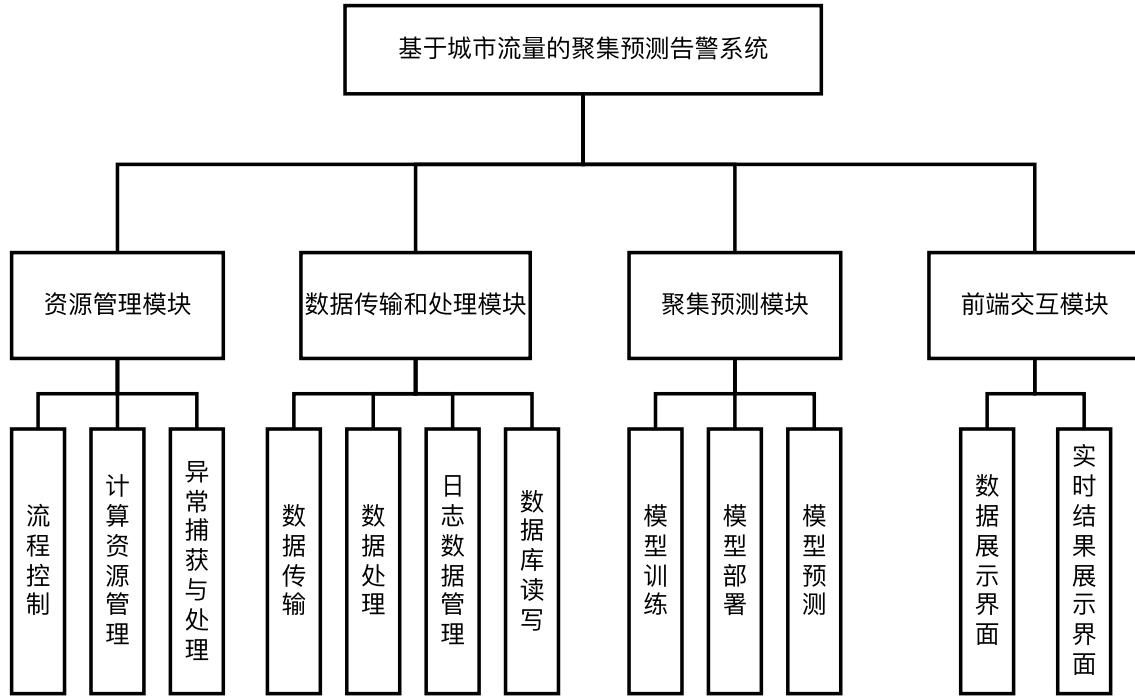


图 5-3 聚集预测告警系统模块图

块进行交互；并且能够对其他本地数据进行统一管理。

3、聚集预测模块。该模块主要负责模型训练、模型部署、模型预测，以及与数据传输和处理模块之间进行交互。分为模型训练、模型调用和数据交互三个部分。

4、前端交互模块。主要包括 UI 界面的设计与实现以及通过嵌入 Kepler.gl 库提供简单的自主交互能力，包括数据展示界面和实时结果展示界面。在数据展示界面中分别展示了轨迹数据的分布以及路网结构；实时结果展示界面同时对实时轨迹信息以及经过模型预测的结果进行展示。

5.4 系统实现

本节根据前文所述的整体设计思路对每个模块进行了详细的介绍。

5.4.1 资源管理模块

资源管理模块是整个系统的管理中枢，为系统的运行提供保障。系统流程图如图5-4所示，主要包含了对数据展示和聚集预测两个事务的管理。对于数据展示事务，基于用户选择的区域与轨迹数据，通过数据选择-数据处理-数据展示-界面交互的流程，对用户选择的轨迹数据的整体分布以及所处区域的路网结构的可视化信息通过两个 Kepler.gl 界面进行展示，并对路网结构进行了动态可视化。对于聚集预测事务，首先端口实时获取数据并对数据进行处理，通过调用部署的算法

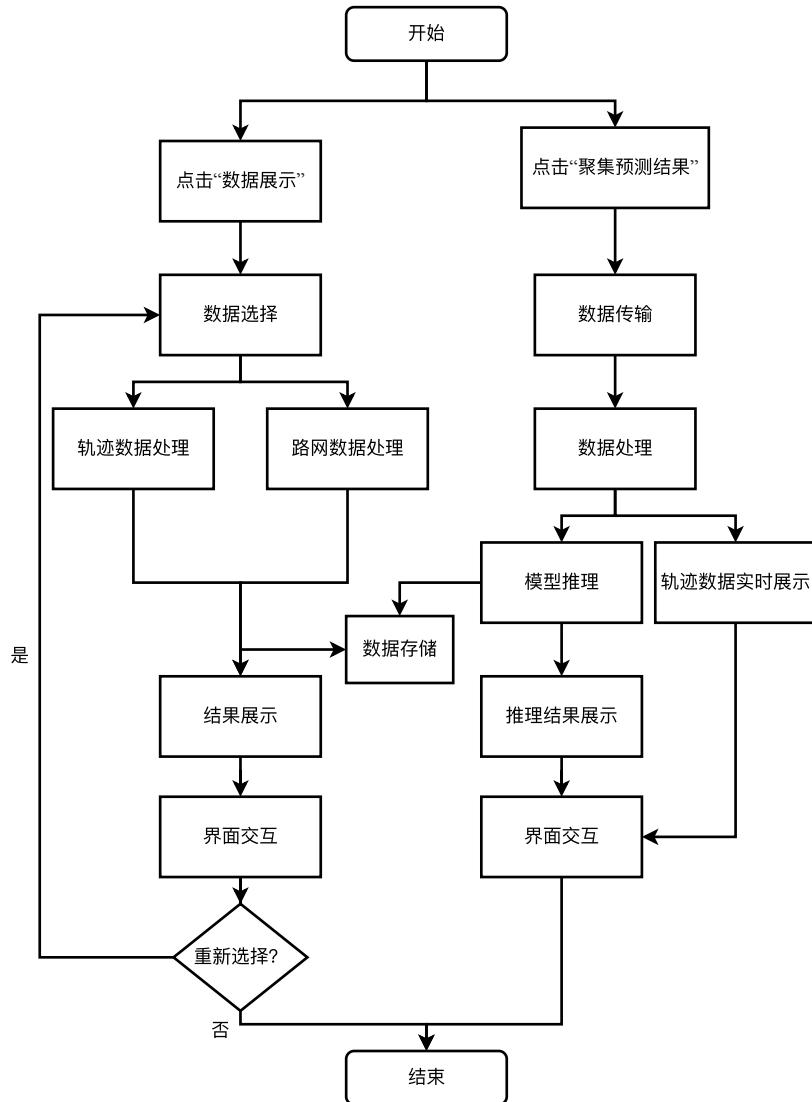


图 5-4 聚集预测警系统整体流程图

模型推理得到预测结果，同样在两个 Kepler.gl 界面中分别展示实时获取的轨迹信息以及处理后的聚集预测结果。

5.4.2 数据传输和处理模块

数据传输和处理模块主要负责数据传输、数据处理、日志数据管理和数据库读写。其中日志信息基于 Logtail 实现，主要记录系统的运行状态、用户操作、接口使用情况等。为后续系统使用中的产生的 bug 进行溯源以及根据使用情况进行性能优化等。

对于实施数据的获取由数据传输接口实现，通过开放端口和端口监听获取移动端的位置信息和时间信息。并与部署模型的服务器之间构建了固定连接以传输

处理好的轨迹数据并接收模型的预测结果。基于数据传输接口能够实现模型部署和系统部署的相互独立，对硬件要求更低，实际应用中部署更加灵活。由于测试环境配置了公网 IP，测试时通过建立 TCP 连接进行移动端数据传输，开放端口数量根据实际需求设置。对于没有配置公网 IP 的特殊情况，可以使用 http 服务实现相同效果。

对与获取到的轨迹数据，需要结合路网图进行处理，以及对其进行样本划分以满足模型的输入要求，并提供了自主的参数设置能够满足不同场景下的应用需求。并将处理完的数据和模型推理结果在前端界面进行展示并存入数据库保存。

系统部署与模型部署端的输出传输通过 sftp 协议实现，并借助 rsync 实现断点续传，增强了数据传输模块的鲁棒性。

由于本系统中所涉及数据较为简单，同时为了降低系统部署端的计算资源需求，采用轻量化的开源数据库 SQLite 对数据进行存储。如表5-1所示为本文中所使用的数据库设计文档。

表 5-1 数据库设计文档

字段	存储类	描述
id	INTEGER	用户编号，根据传输设备自动生成
time	TIMESTAMP	时间戳信息
lat	REAL	轨迹点纬度
lng	REAL	轨迹点经度
road	INTEGER	所处路段编号
label	INTEGER	所处区域编号

5.4.3 聚集预测模块

聚集预测模块采用的是第四章中的基于空间特征增强的聚集预测模型，骨干网络为第三章中的时空间门控循环单元（STGRU）。

在应用中首先需要对模型进行预训练，根据需要部署的区域获取对应的路网结构，并结合历史轨迹信息生成路网图。然后在对应的人流数据集上进行预训练，因为引入了路网结构知识，城市间的路网结构各不相同，不同城市数据集训练的模型在其他数据集上的泛化能力较差。

为了保证聚集预测结果的实时性，将模型部署在多张运算显卡上来增加数据吞吐量，大幅减少了模型推理部分的耗时。模型推理所需的路网图数据以及预训练模型存储在服务器上，根据模型实际大小以及数据吞吐量设置适当的推理 batch size，充分利用显卡的内存与计算能力，本系统测试部署时的 batch size 大小为 32，每张 GeForce RTX™ 2080 Ti 显卡能够同时部署 2 个预训练模型，测试环境共有 2

张显卡，能部署 4 个模型，单次吞吐量为 128。

5.4.4 前端交互模块

前端交互模块主要包含 UI 界面的设计与实现以及简单的自主交互能力。所有的 UI 设计使用侧边导航栏的方式，并将剩余空间均分为两部分以展示更多的内容并实现了一定的数据对比效果。事务切换通过侧边导航栏的 NavLink 组件实现。

5.4.5 聚集预测结果展示界面

用户点击 **【数据展示】** 可切换至数据展示界面，其中左侧为所选择数据的展示区域，数据通过 Radio 组件进行选择，右侧为对应区域的路网结构的展示区域，根据所选择数据自动适配。两个展示区域均基于 Kepler.gl 实现，支持简单的缩放、视角转换等功能，能够实现对数据进行全方位的展示。

点击 **【聚集预测结果】** 可切换至实时结果展示界面，此界面的数据传输与选择需在部署时进行配置。左侧为对实时获取的数据进行动态展示，右侧为根据模型预测结果进行聚集预测情况的可视化展示。

在数据展示时需要预先加载对应地区的道路网络贴图，再加载需要展示的轨迹数据和聚集结果数据。其中轨迹数据经过预处理后与路网节点信息一对一匹配，只需要通过过滤器对显示时间进行过滤，再利用时间戳进行排序显示，即可实现轨迹的动态展示。聚集预测结果展示则需要根据超参数设置提前对被预测区域进行划分，以及对数值进行色阶设置，动态展示与轨迹数据类似，通过过滤器实现。

5.5 系统展示

通过系统部署服务器的 IP 地址和设置的 web 服务端口可以在任意浏览器中访问本系统。进入本系统时，默认进入数据展示界面，加载为 **【day1】** 标签所对应的数据。右侧加载对应的路网数据，并通过模拟时间戳实现了动态展示。通过对顶部 Radio 组件选择不同时间的历史数据，轨迹数据展示部分和路网数据展示部分会同时重新加载对应的轨迹数据和路网数据。当用户处于其他界面时，可以通过点击侧边导航栏的 **【数据展示】** 进入数据展示界面。

用户通过点击侧边导航栏的 **【聚集预测结果】** 进入实时结果展示界面。实时结果展示界面不提供数据选择功能，会根据配置开放数据传输接口，并实时接收数据。对接收到的轨迹数据根据其时间戳展示在左侧区域。右侧为根据接收到的轨迹数据以及历史轨迹数据得到的聚集预测结果的可视化展示。其中展示的图层及颜色等参数可以通过 Kepler.gl 提供的交互接口实现简单的交互，自定义显示样例。

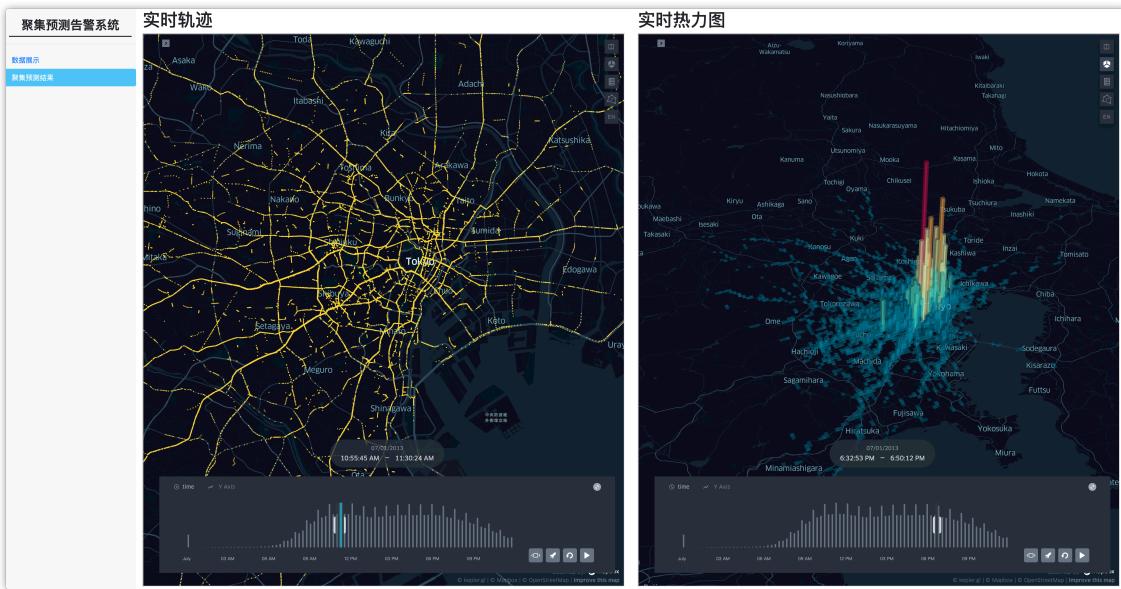


图 5-5 聚集预测结果展示界面

如图5-5所示为使用数据集中数据进行模拟演示的结果，颜色样例等为预设参数。

5.6 系统测试

本节对基于城市流量的聚集预测警系统的各项功能和性能进行了测试分析，明确系统是否满足需求。

系统模型部分与系统服务部分部署环境并不相同，系统模型部分的测试环境如表3-1所示，系统服务部分的测试环境如表5-2所示。

表 5-2 系统测试环境配置表

软硬件	配置
芯片	Apple M1 芯片
内存	8GB 统一内存
操作系统	macOS Monterey 12.2.1
SQLite 版本	3.38.0
Flask 版本	2.0.3
kepler.gl 版本	0.2.2

5.6.1 功能测试

根据系统模块设置与功能分割，分别对系统的接受移动端数据、模型推理、文件传输、数据展示和结果展示五个功能进行测试。

接受移动端数据。如表5-3所示，服务器能够通过 TCP 协议开放的端口正常接

受移动端发送的位置信息。后续根据数据接收时间添加时间戳信息以及根据移动端的 ip 地址生成对应编号。

表 5-3 接受移动数据测试用例表

用例编号	1	功能名称	接受移动端数据
测试方法	黑盒测试	测试时间	2022/2/25
测试说明	服务器端开放端口与移动端通信, 移动端定期发送位置信息		
判断标准	服务器端能正常接受到位置信息, 格式为文本, 且内容符合经纬度格式, 精确度高于 10^2		
测试环境	服务器端 ip: 113.54.199.129, 端口号: 1234		
测试输出	lat:30.749,lon:103.924		
测试评价	测试通过, 接受数据符合预期		

文件传输。服务器与模型部署端的数据交互通过文件传输完成, 通过 sftp 协议完成, 如表5-4所示, 能够正确传输文件, 并根据时间戳生成用户名。

表 5-4 文件传输测试用例表

用例编号	2	功能名称	文件传输
测试方法	黑盒测试	测试时间	2022/2/25
测试说明	服务器与模型部署端通过 sftp 协议共享数据文件夹, 并生成唯一文件名		
判断标准	服务器端在固定时间间隔生成数据文件, 文件权限为 666 命名格式为 in-xxx-yyyy-mm-dd-number.csv, 模型部署端针对输入文件生成输出结果, 文件权限为 666, 命名格式为 out-xxx-yyyy-mm-dd-number.csv		
测试输出	输入文件: in-chengdu-2022-02-25-1.csv, 文件权限 rw-rw-rw- 输出文件: out-chengdu-2022-02-25-1.csv, 文件权限 rw-rw-rw-		
测试评价	测试通过, 输出传输、数据文件命名及权限设置符合预期		

模型推理。结合历史轨迹信息生成样本, 预测用户在一段时间后的所处区域, 默认时间间隔为半小时, 如表5-5所示, 输入时需要给定用户 id, 模型输出为预测的区域编号。

表 5-5 模型推理测试用例表

用例编号	3	功能名称	模型推理
测试方法	黑盒测试	测试时间	2022/2/25
测试说明	模型部署端根据数据库中历史轨迹信息生成样本并进行预测		
判断标准	预测结果有时间戳, 且区域编号符合生成规则		
测试输入	(1,103.924,30.749,2022-2-25-14:30)		
测试输出	(1,817,2022-2-25-15:00)		
测试评价	测试通过, 模型推理结果符合预期		

数据展示。提供历史数据和路网结构可视化展示功能，测试结果如表5-6所示。

表 5-6 数据展示测试用例表

用例编号	4	功能名称	数据展示
测试方法	黑盒测试	测试时间	2022/2/25
测试说明	数据展示界面通过 Radio 组件选择对应日期		
判断标准	数据展示界面能够正常显示数据，支持简单的缩放和拖拽		
测试输出	界面左右分别显示对应日期的所有历史轨迹信息和路网结构信息		
测试评价	测试通过，能够根据所选日期正确显示		

结果展示。读取模型推理结果并进行结果的可视化展示，同时对接受到的数据进行实时展示，测试结果如表5-7所示。

表 5-7 结果展示测试用例表

用例编号	5	功能名称	结果展示
测试方法	黑盒测试	测试时间	2022/2/25
测试说明	根据接收数据的时间戳筛选并显示用户分布，读取预测结果通过 3d 网格可视化		
判断标准	结果展示界面能够正常显示实时轨迹数据与预测结果，支持简单的缩放和拖拽		
测试输出	能够显示所有用户的实时分布并使用 3d 网格显示预测结果		
测试评价	测试通过，能够根据所选日期正确显示		

根据上述功能测试结果，系统能够所有的相关功能测试，符合系统需求，已达到预期功能目标。

5.6.2 性能测试

系统对数据传输和模型推理上有一定的性能需求，分别测从接收移动端数据、模型推理和系统的兼容性方面进行测试。

接受移动端数据。系统在对实时获取目标移动端的位置信息时，需要满足一定量的并发能力，在当前测试环境下需要满足 500 个移动客户端同时传输数据。测试时开放了 50 个通信端口，通过端口扫描模拟移动端通信，结果如表5-8所示，能够满足上述要求。

模型推理耗时。默认情况下的间隔时间为 5 分钟，因此需要在 5 分钟内完成对所有样本的模型推理。测试时根据数据接受的峰值传入 500 个轨迹点信息，模

表 5-8 移动端并发性能测试结果

并发量	正确接受数	耗时 (秒)	成功比例	测试结果
1	1	1.02	100%	测试通过
50	50	1.25	100%	测试通过
100	100	2.89	100%	测试通过
500	497	20.58	99.4%	测试通过

型部署端的运算显卡为两张 GeForce RTX™ 2080 Ti，每张运算显卡上部署了两个聚集预测模型，生成并完成 500 个样本推理的计算时间为 2 分 35 秒，能够满足性能要求。

兼容性测试。测试了系统对操作系统以及浏览器兼容性，结果如表5-9所示，满足正常使用需求。

表 5-9 兼容性测试结果

测试对象	测试内容	软件版本	测试结果
操作系统	能否正常开放对应端口以及系统部署	Chrome:99.0.4844.51 (64 位) FireFox:89.0 Microsoft Edge:99.0.1150.39	测试通过
浏览器	能否正常访问系统界面	Ubuntu 16.04+ CentOS 7+ MacOS 10.05+ Windows 10+	测试通过
移动端	能否正常传输数据	HarmonyOS 2.0.0 EMUI 10+ MIUI 11+	测试通过

5.7 本章小结

本章详细介绍了基于城市流量的聚集预测预警系统的设计与开发，包括需求分析、系统设计、系统实现、系统展示和相关测试。在完成系统设计后，本章在时空控循环单元和基于空间特征增强的聚集预测模型的基础上，对系统功能进行了可视化展示并对系统相关功能性能进行测试。通过系统的设计开发，将本文所提出的聚集预测模型应用到城市场景下的实际任务中，实现了对日常生活中人群聚集的预测，为城市管控提供一定帮助。

第六章 总结与展望

6.1 全文总结

聚集预测任务作为城市管理管控的重要组成部分，近年来正发挥着越来越重要的作用。本文针对现有聚集预测算法忽略了路网等地理环境信息以及天气信息、假期信息等外部知识以及对时空特征建模能力较弱参数量需求较大等问题，分别从引入并有效建模外部知识和通过空间特征增强辅助建模多用户之间的相互影响两个角度进行研究，实现了多用户和强实时轨迹场景下的聚集预测。

本文主要完成的工作有：

1、针对现有方法忽略了路网等地理环境信息以及天气信息、假期信息等外部知识的问题和参数量较大的问题，设计了路网门控结构，并通过扩展 GRU 模型提出了时空门控循环单元（STGRU）模型。

2、针对聚集预测多用户关联特征建模难的问题，提出了一个基于空间特征增强的聚集预测模型，使用希尔伯特扫描算法和基于统计的路网结构嵌入方法实现对多用户关联特征的建模。通过实验，本文提出的聚集预测模型能够显著提升时空特征建模能力，能够有效提升模型的聚集预测精度，在多个真实世界数据上取得了当前最优性能。

3、以基于空间特征增强的聚集预测模型为核心，设计并实现了基于城市流量的聚集预测预警系统，对系统设计和实现过程进行了详细介绍，还对聚集预测进行了可视化。

6.2 后续工作展望

本文提出的聚集预测模型针对现有方法存在的一些问题进行一定的优化和改进。但是，由于实验环境和数据集的限制，该任务还存在很大的研究和创新空间。

1. 基于图网络的聚集预测。由于数据集的限制，当前只有很少部分轨迹数据集能用于聚集预测任务，也一定程度限制了所能使用的方法范围。若是能够针对聚集预测任务构建数据集，能够利用图网络的结构优势能够更好的提取数据中的时空特征，算法精度会有较大提升空间。

2. 时空序列神经网络的堆叠。多种扩展序列模型的方法均不能很好的进行模型的堆叠或者仅使用标准模型进行堆叠，这会对显式提取的时空特征有一定程度的弱化。针对这类扩展序列模型提取的时空特征设计能够进行多层堆叠的结构，或许可以充分利用深度学习模型的优势进一步提升对时空特征的建模能力。

致 谢

研究生的这段经历是追求自己梦想的开始，自本科毕业后经历了两年军旅生活，离开部队时毅然决定继续深造，有幸能在成电度过我的研究生三年。在研究生期间，曾有过自我怀疑和自我否定，但终究时坚持下来了。本文总结了我研究生期间所有工作，在此，对本文成文过程中所有鼓励、帮助过我的老师、同学、朋友和亲人表示最真诚的感谢。

首先，最感谢的是我的导师田玲老师。田老师（人美心善）为人严谨，做事认真，在科研生活中给予了我宝贵的帮助与建议。在田老师的帮助和细心教导下，我的逻辑思维能力得到了较大的提升，对自己的要求也变得更高了，这是研究生生涯中我最大的收获。另，田老师平时还是有点严肃，希望田老师多笑点！

其次对郑旭老师从我入学至今的指导和帮助表示感谢。郑老师在学习生活的各方面都给予了我很大的帮助，从入学时的暑期教程到学术论文的撰写发表以及本文的撰写修改都是在郑老师的指导和帮助下完成的。研究生三年里，在郑老师悉心教导下，学习阅读文献、设计实验、撰写论文，最宝贵的是学会了发现问题和独立解决问题的能力。不可否认，郑老师是实验室最帅的老师！

然后还要对实验室的闫科老师和康昭老师等老师对我的帮助，在我完成论文期间给予了宝贵的建议。

同时对实验室的师兄师姐们，以及张谨川、张天舒、张雨林、张勇、曾翰林等同学的关心与帮助。

最后要感谢我的家人对我的支持，他们是我求学道路上最坚强的后盾。

参考文献

- [1] Sejdiu B, Ismaili F, Ahmed L. Iotsas: An integrated system for real-time semantic annotation and interpretation of iot sensor stream data[J]. Computers, 2021, 10(10): 127.
- [2] Sharma P, Jain S, Gupta S, et al. Role of machine learning and deep learning in securing 5g-driven industrial iot applications[J]. Ad Hoc Networks, 2021, 123: 102685.
- [3] Chao F, He Z, Feng R, et al. Predictive trajectory-based mobile data gathering scheme for wireless sensor networks[J]. Complexity, 2021, 2021: 3941074:1-3941074:17.
- [4] Rossi L, Paolanti M, Pierdicca R, et al. Human trajectory prediction and generation using LSTM models and gans[J]. Pattern Recognition, 2021, 120: 108136.
- [5] Osaragi T, Hayasaka R. Prediction of spatiotemporal distributions of transient urban populations with statistics gathered by cell phones[C]. Proceedings of the 6th International Conference on Geographical Information Systems Theory, Applications and Management, GISTAM 2020, Prague, Czech Republic, 2020: 33-44.
- [6] Wang S, Cao J, Chen H, et al. Seqst-gan: Seq2seq generative adversarial nets for multi-step urban crowd flow prediction[J]. ACM Transactions on Spatial Algorithms and Systems, 2020, 6(4): 22:1-22:24.
- [7] Liu X, Qin X. A neighborhood information utilization fireworks algorithm and its application to traffic flow prediction[J]. Expert Systems with Applications, 2021, 183: 115189.
- [8] Preto A J, Matos-Filipe P, de Almeida J G, et al. Predicting hot spots using a deep neural network approach[M]. Springer, 2021: 267-288
- [9] Liu Z, Wang H. Improved population prediction strategy for dynamic multi-objective optimization algorithms using transfer learning[C]. IEEE Congress on Evolutionary Computation, CEC 2021, Kraków, Poland, 2021: 103-110.
- [10] Alassafi M O, Jarrah M, Alotaibi R. Time series predicting of COVID-19 based on deep learning[J]. Neurocomputing, 2022, 468: 335-344.
- [11] Sreenivasulu M, Sridevi M. Comparative study of statistical features to detect the target event during disaster[J]. Big Data Mining and Analytics, 2020, 3(2): 121-130.
- [12] Wu Z, Ismail M, Serpedin E, et al. Artificial intelligence for smart resource management in multi-user mobile heterogeneous rf-light networks[J]. IEEE Wireless Communications, 2021, 28(4): 152-158.

- [13] Lu F, Maggioni M, Tang S. Learning interaction kernels in heterogeneous systems of agents from multiple trajectories[J]. *Journal of Machine Learning Research*, 2021, 22: 32:1-32:67.
- [14] Lian D, Wu Y, Ge Y, et al. Geography-aware sequential location recommendation[C]. *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, 2020*: 2009-2019.
- [15] Yuan J, Zheng Y, Xie X, et al. Driving with knowledge from the physical world[C]. *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA, 2011*: 316-324.
- [16] Lian D, Zhao C, Xie X, et al. Geomf: joint geographical modeling and matrix factorization for point-of-interest recommendation[C]. *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA, 2014*: 831-840.
- [17] Li X, Cong G, Li X, et al. Rank-geofm: A ranking based geographical factorization method for point of interest recommendation[C]. *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, 2015*: 433-442.
- [18] Cheng C, Yang H, Lyu M R, et al. Where you like to go next: Successive point-of-interest recommendation[C]. *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, 2013*: 2605-2611.
- [19] Liu Q, Wu S, Wang L, et al. Predicting the next location: A recurrent model with spatial and temporal contexts[C]. *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, Phoenix, Arizona, USA, 2016*: 194-200.
- [20] 刘大有, 陈慧灵, 齐红, 等. 时空数据挖掘研究进展 [J]. *计算机研究与发展*, 2013, 50(2): 225.
- [21] Wu Y, Lian D, Jin S, et al. Graph convolutional networks on user mobility heterogeneous graphs for social relationship inference[C]. *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, 2019*: 3898-3904.
- [22] Jin K, Sun S, Li H, et al. A novel multi-modal analysis model with baidu search index for subway passenger flow forecasting[J]. *Engineering Applications of Artificial Intelligence*, 2022, 107: 104518.
- [23] Iwata T, Shimizu H. Neural collective graphical models for estimating spatio-temporal population flow from aggregated data[C]. *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, Honolulu, Hawaii, USA, 2019*: 3935-3942.

- [24] Verma N, Balaian N. Pam clustering based taxi hotspot detection for informed driving[C]. 2017 8th International Conference on Computing, Communication and Networking Technologies (IC-CCNT), 2017: 1-7.
- [25] Ni M, He Q, Gao J. Forecasting the subway passenger flow under event occurrences with social media[J]. IEEE Transactions on Intelligent Transportation Systems, 2017, 18(6): 1623-1632.
- [26] Kumar D, Bezdek J C, Rajasegarar S, et al. A visual-numeric approach to clustering and anomaly detection for trajectory data[J]. The Visual Computer, 2017, 33(3): 265-281.
- [27] Ye M, Yin P, Lee W, et al. Exploiting geographical influence for collaborative point-of-interest recommendation[C]. Proceeding of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2011, Beijing, China, 2011: 325-334.
- [28] Hamdi A, Shaban K B, Erradi A, et al. Spatiotemporal data mining: a survey on challenges and open problems[J]. Artificial Intelligence Review, 2022, 55(2): 1441-1488.
- [29] Guo Q, Sun Z, Zhang J, et al. An attentional recurrent neural network for personalized next location recommendation[C]. The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, NY, USA, 2020: 83-90.
- [30] Moreno-Díaz A, Virto M A, Martín J, et al. Approximate solutions to semi markov decision processes through markov chain montecarlo methods[C]. Computer Aided Systems Theory - EU-ROCAST 2003, 9th International Workshop on Computer Aided Systems Theory, Las Palmas de Gran Canaria, Spain, 2003: 151-162.
- [31] Begleiter R, El-Yaniv R, Yona G. On prediction using variable order markov models[J]. Journal of Artificial Intelligence Research, 2004, 22: 385-421.
- [32] Rendle S, Freudenthaler C, Schmidt-Thieme L. Factorizing personalized markov chains for next-basket recommendation[C]. Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, 2010: 811-820.
- [33] Feng S, Li X, Zeng Y, et al. Personalized ranking metric embedding for next new POI recommendation[C]. Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, 2015: 2069-2075.
- [34] Xie M, Yin H, Wang H, et al. Learning graph-based POI embedding for location-based recommendation[C]. Proceedings of the 25th ACM International Conference on Information and Knowledge Management, CIKM 2016, Indianapolis, IN, USA, 2016: 15-24.

- [35] Song J, Xiao J, Wu F, et al. Hierarchical contextual attention recurrent neural network for map query suggestion[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2017, 29(9): 1888-1901.
- [36] Sordoni A, Bengio Y, Vahabi H, et al. A hierarchical recurrent encoder-decoder for generative context-aware query suggestion[C]. *Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, 2015*: 553-562.
- [37] Wang X, Wang Y. Improving content-based and hybrid music recommendation using deep learning[C]. *Proceedings of the ACM International Conference on Multimedia, MM '14, Orlando, FL, USA, 2014*: 627-636.
- [38] Zhang F, Yuan N J, Lian D, et al. Collaborative knowledge base embedding for recommender systems[C]. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 2016*: 353-362.
- [39] He X, Liao L, Zhang H, et al. Neural collaborative filtering[C]. *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, 2017*: 173-182.
- [40] Cui Q, Wu S, Huang Y, et al. A hierarchical contextual attention-based network for sequential recommendation[J]. *Neurocomputing*, 2019, 358: 141-149.
- [41] Mikolov T, Karafiat M, Burget L, et al. Recurrent neural network based language model[C]. *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, 2010*: 1045-1048.
- [42] Hochreiter S, Schmidhuber J. Long short-term memory[J]. *Neural Computation*, 1997, 9(8): 1735-1780.
- [43] Cho K, van Merriënboer B, Gülcühre Ç, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[C]. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, 2014*: 1724-1734.
- [44] Sun K, Qian T, Chen T, et al. Where to go next: Modeling long- and short-term user preferences for point-of-interest recommendation[C]. *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, NY, USA, 2020*: 214-221.
- [45] Al-Molegi A, Jabreel M, Ghaleb B. STF-RNN: space time features-based recurrent neural network for predicting people next location[C]. *2016 IEEE Symposium Series on Computational Intelligence, SSCI 2016, Athens, Greece, 2016*: 1-7.

- [46] Kurashima T, Iwata T, Hoshide T, et al. Geo topic model: joint modeling of user's activity area and interests for location recommendation[C]. Sixth ACM International Conference on Web Search and Data Mining, WSDM 2013, Rome, Italy, 2013: 375-384.
- [47] Liu Z, Miranda F, Xiong W, et al. Learning geo-contextual embeddings for commuting flow prediction[C]. The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, USA, 2020: 808-816.
- [48] Feng J, Li Y, Zhang C, et al. Deepmove: Predicting human mobility with attentional recurrent networks[C]. Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, 2018: 1459-1468.
- [49] Tang J, Wang K. Personalized top-n sequential recommendation via convolutional sequence embedding[C]. Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, 2018: 565-573.
- [50] Kong D, Wu F. HST-LSTM: A hierarchical spatial-temporal long-short term memory network for location prediction[C]. Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, Stockholm, Sweden, 2018: 2341-2347.
- [51] Zhao P, Zhu H, Liu Y, et al. Where to go next: A spatio-temporal gated network for next POI recommendation[C]. The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, Honolulu, Hawaii, USA, 2019: 5877-5884.
- [52] Hastie T, Tibshirani R, Friedman J H. The elements of statistical learning: Data mining, inference, and prediction, 2nd edition[M]. Springer, 2009
- [53] Yang Y, Lin Y, Chu H, et al. Deep learning with a rethinking structure for multi-label classification[C]. Proceedings of The 11th Asian Conference on Machine Learning, ACML 2019, Nagoya, Japan, 2019: 125-140.
- [54] Fawcett T. An introduction to ROC analysis[J]. Pattern Recognition Letters, 2006, 27(8): 861-874.
- [55] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[C]. Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Lake Tahoe, Nevada, United States, 2012: 1106-1114.
- [56] Torres J F, Hadjout D, Sebaa A, et al. Deep learning for time series forecasting: A survey[J]. Big Data, 2021, 9(1): 3-21.

- [57] Rosenblatt F. The perceptron: a probabilistic model for information storage and organization in the brain.[J]. Psychological review, 1958, 65(6): 386.
- [58] Bengio Y, Simard P Y, Frasconi P. Learning long-term dependencies with gradient descent is difficult[J]. IEEE Transactions on Neural Networks, 1994, 5(2): 157-166.
- [59] Li J, Wang Y, McAuley J J. Time interval aware self-attention for sequential recommendation[C]. WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, 2020: 322-330.
- [60] Li Y, Yu R, Shahabi C, et al. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting[C]. 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, 2018: 1-16.
- [61] Zhao L, Song Y, Zhang C, et al. T-GCN: A temporal graph convolutional network for traffic prediction[J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 21(9): 3848-3858.
- [62] Yu B, Yin H, Zhu Z. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting[C]. Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, Stockholm, Sweden, 2018: 3634-3640.
- [63] Lin X, Niu S, Wang Y, et al. K-plet recurrent neural networks for sequential recommendation[C]. The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, 2018: 1057-1060.
- [64] Ruiz C K V, Quercia D, Bonchi F, et al. Taxonomy-based discovery and annotation of functional areas in the city[C]. Proceedings of the Ninth International Conference on Web and Social Media, ICWSM 2015, University of Oxford, Oxford, UK, 2015: 445-453.
- [65] Yuan N J, Zheng Y, Xie X, et al. Discovering urban functional zones using latent activity trajectories[J]. IEEE Transactions on Knowledge and Data Engineering, 2015, 27(3): 712-725.
- [66] Sekimoto Y, Shibasaki R, Kanasugi H, et al. Pflow: Reconstructing people flow recycling large-scale social survey data[J]. IEEE Pervasive Computing, 2011, 10(4): 27-35.
- [67] Hilbert D. Mathematical problems[J]. Bulletin of the American Mathematical Society, 1902, 8(10): 437-479.
- [68] Kamata S, Bandoh Y. An address generator of a pseudo-hilbert scan in a rectangle region[C]. Proceedings 1997 International Conference on Image Processing, ICIP '97, Santa Barbara, California, USA, 1997: 707-710.
- [69] Kamata S, Eason R O, Bandoh Y. A new algorithm for n-dimensional hilbert scanning[J]. IEEE Transactions on Image Processing, 1999, 8(7): 964-973.

- [70] Zhang J, Kamata S, Ueshige Y. A pseudo-hilbert scan algorithm for arbitrarily-sized rectangle region[C]. International Workshop on Intelligent Computing in Pattern Analysis/Synthesis, IWIC-PAS 2006, Xi'an, China, 2006: 290-299.
- [71] Wang D, Jiang M, Syed M, et al. Calendar graph neural networks for modeling time structures in spatiotemporal user behaviors[C]. KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, 2020: 2581-2589.
- [72] Boyaci B, Dang T H, Letchford A N. Vehicle routing on road networks: How good is euclidean approximation?[J]. Computers & Operations Research, 2021, 129: 105197.
- [73] Owaki T, Machida T. Roadnetgan: Generating road networks in planar graph representation[C]. Neural Information Processing - 27th International Conference, ICONIP 2020, Bangkok, Thailand, 2020: 535-543.
- [74] Lin J, Ban Y. Comparative analysis on topological structures of urban street networks[J]. ISPRS International Journal of Geo-Information, 2017, 6(10): 295.
- [75] Yang L, Zhai J, Wang X. A graph-cut-based method for road labels making with OSM data[J]. IEEE Geoscience and Remote Sensing Letters, 2022, 19: 1-5.
- [76] Niepert M, Ahmed M, Kutzkov K. Learning convolutional neural networks for graphs[C]. Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, 2016: 2014-2023.
- [77] Defferrard M, Bresson X, Vandergheynst P. Convolutional neural networks on graphs with fast localized spectral filtering[C]. Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, Barcelona, Spain, 2016: 3837-3845.
- [78] Shuman D I, Narang S K, Frossard P, et al. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains[J]. IEEE Signal Processing Magazine, 2013, 30(3): 83-98.
- [79] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[C]. 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, 2017: 1-14.
- [80] Hammond D K, Vandergheynst P, Gribonval R. Wavelets on graphs via spectral graph theory[J]. Applied and Computational Harmonic Analysis, 2011, 30(2): 129-150.

攻读硕士学位期间取得的成果

- [1] **Mingchao Yuan**, Tian L, Yan K, et al. A road network enhanced gate recurrent unit model for gather prediction in smart cities[J]. Wireless Communications and Mobile Computing, 2021, 2021: 1-11.
- [2] 物联网感知数据隐私保护关键技术研究, 61802050, (参研)
- [3] 面向多接入点的大容量长距离安全传输网络关键技术及应用, 2021YFG0018, (参研)
- [4] 人工智能概念体系架构图谱软件, 200978, (参研)
- [5] 学业奖学金三等奖, 2020 年 10 月
- [6] 学业奖学金二等奖, 2021 年 10 月