计算机科学
COMPUTER SCIENCE

面向自动化集装箱码头的 AGV 行驶时间估计

冷典典，杜鹏，陈建廷，向阳

引用本文

冷典典，杜鹏，陈建廷，向阳. 面向自动化集装箱码头的 AGV 行驶时间估计[J]. 计算机科学, 2022, 49(9): 208-214.

LENG Dian-dian, DU Peng, CHEN Jian-ting, XIANG Yang. Automated Container Terminal Oriented Travel Time Estimation of AGV[J]. Computer Science, 2022, 49(9): 208-214.

相似文章推荐（请使用火狐或 IE 浏览器查看文章）

Similar articles recommended (Please use Firefox or IE to view the article)

密码学智能化研究进展与分析

Research Progress and Analysis on Intelligent Cryptology

计算机科学, 2022, 49(9): 288-296. https://doi.org/10.11896/jsjkx.220300053

基于大数据的进化网络影响力分析研究综述

Survey of Influence Analysis of Evolutionary Network Based on Big Data

计算机科学, 2022, 49(8): 1-11. https://doi.org/10.11896/jsjkx.210700240

基于多尺度的稀疏脑功能超网络构建及多特征融合分类研究

Construction and Multi-feature Fusion Classification Research Based on Multi-scale Sparse Brain Functional Hyper-network

计算机科学, 2022, 49(8): 257-266. https://doi.org/10.11896/jsjkx.210600094

基于 N-Gram 静态分析技术的恶意软件分类研究

Study on Malware Classification Based on N-Gram Static Analysis Technology

计算机科学, 2022, 49(8): 336-343. https://doi.org/10.11896/jsjkx.210900203

联邦学习攻防研究综述

Survey on Attacks and Defenses in Federated Learning

计算机科学, 2022, 49(7): 310-323. https://doi.org/10.11896/jsjkx.211000079

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.210700028

面向自动化集装箱码头的AGV行驶时间估计

冷典典　杜　鹏　陈建廷　向　阳
同济大学电子与信息工程学院　上海 201800
(1933049@tongji.edu.cn)

摘　要　自动导引车(Automated Guided Vehicle,AGV)在自动化集装箱码头的水平运输中发挥了重要作用,对AGV行驶时间进行准确估计,有利于减少码头各作业环节的资源闲置,提高整体效率。针对AGV在自动化集装箱码头的行驶时间估计问题,提出了一种AGV行驶时间估计方法。首先,根据AGV的行驶模式将目标行驶路径切分为若干段,使用神经网络模型对其进行编码;其次,对该路径出发前后一段时间内的其他路径进行编码并将其作为环境信息,以通过模型预测其是否与目标路径发生冲突作为辅助任务;最后,综合两类信息对行驶时间进行估计。该方法引入了路径间冲突对时间估计造成的影响。基于自动化集装箱码头的历史数据的实验表明,相比AGV场景中常用的静态时间估计方法,所提方法能够将时间估计的误差降低18%以上,可以更准确地估计AGV的行驶时间。

关键词:自动化集装箱码头;行驶时间估计;AGV;机器学习
中图法分类号　TP23

Automated Container Terminal Oriented Travel Time Estimation of AGV

LENG Dian-dian,DU Peng,CHEN Jian-ting and XIANG Yang
College of Electronic and Information Engineering,Tongji University,Shanghai 201800,China

Abstract　Automated guided vehicles(AGV)are crucial for the horizontal transportation of automated container terminals. Accurate estimation of the travel time of each AGV will reduce the number of idle AGV resources and increase the efficiency of the entire terminal. This paper proposes a method for travel time estimation of AGV in automated container terminals. Firstly,the target route of AGV is divided and encoded into several segments. Secondly,other routes are encoded as environment information, which depart before or after the target route. And the conflict between these routes and target route is estimated as an auxiliary task. Finally,the travel time with all encodings is calculated. The proposed method introduces the influence of path conflicts on time estimation. Experiments based on historical data of automated terminals show that,compared with static time estimation methods commonly used in AGV scenarios,the proposed method can reduce the time estimation error by more than 18%,and can estimate the travel time more accurately.

Keywords　Automated container terminal,Travel time estimation,AGV,Machine learning

1　引言

随着技术的进步和经济全球化的发展,以及国际贸易需求的增大,集装箱码头的自动化成为一种趋势。自动化集装箱码头就是将集装箱码头内,包括岸边与堆场之间的货物运输、堆场内的集装箱作业等工作完全实现自动化。集装箱码头的自动化,能够通过减少工人数量来降低人工成本,并能通过机械设备全天连续作业提高码头的工作效率,降低员工劳动强度[1]。

自动导引车在自动化集装箱码头中承担水平运输任务,穿梭于岸桥与堆场之间,在交换点(Transfer Point,TP)处与轨道吊、桥吊等设备进行交互,是衔接岸桥和堆场的重要设备。

AGV的行驶时间估计(Travel Time Estimation)指在给定起点TP和终点TP之后,根据路径规划算法给出的行驶路径,以及影响AGV速度的载重等任务特征,对该路径的行驶时间进行估计。在码头作业过程中,AGV行驶时间估计的应用场景十分广泛。AGV行驶时间是调度系统的关键基础数据,可应用于自动化码头设备调度、车辆与任务的匹配决策及AGV运行时交通冲突优先级的选取等方面,对于码头整体的效率优化有着重要意义。

当前自动化码头对AGV行驶时间进行估计时采用的是静态时间估计方法,即在给定起点与终点的情况下,单纯考虑车辆的加减速,以没有任何冲突和避让的策略直接计算从起点到达终点所需要的时间。该方法不考虑码头上的实时

到稿日期:2021-07-02　返修日期:2022-02-28
基金项目:国家重点研发计划(2019YFB1704402)
This work was supported by the National Key Research and Development Program of China(2019YFB1704402).
通信作者:向阳(shxiangyang@tongji.edu.cn)

环境变化和任务信息，直接使用物理模型对时间进行计算，估计结果与实际值偏差较大。过大的时间估计误差使得港口作业的各个环节出现了资源闲置或效率低下的现象，亟待改善[2]。

本文提出了一种面向集装箱自动化码头的 AGV 行驶时间估计方法，即基于码头的实时环境变化和 AGV 任务信息，使用神经网络对 AGV 行驶时间进行动态估计，并利用多任务学习框架，引入多辆 AGV 相互之间冲突和避让造成的影响，使行驶时间估计更加准确。首先，基于路径规划得出的行驶轨迹，根据 AGV 的行动模式（直行、转弯等），将其轨迹切分为若干子段，并融合任务特征输入神经网络，得到目标路径的编码；其次，为了引入其他 AGV 路径与目标路径冲突对时间估计造成的影响，本文将可能冲突的其他 AGV 路径轨迹编码为环境路径向量，使用注意力机制，判断其是否与目标路径发生冲突，以此作为多任务学习的辅助任务，使模型学习到路径冲突对行驶时间的影响；最后，综合目标路径的编码和环境信息的注意力权重，对行驶时间进行回归估计。

2 相关工作

目前，行驶时间估计问题主要采用物理模型计算和深度学习两类方法。

很多 AGV 相关的路径规划和调度问题都需要估计 AGV 的行驶时间，但大多采用简单的物理模型进行计算。Zhong 等[3]在自动化码头上的 AGV 路径规划研究中，以及 Yue 等[4]和 Guo 等[5]在柔性生产系统的 AGV 调度研究中，都假设 AGV 的行驶速度是固定的，并以固定速度计算行驶时间。一些研究虽然考虑了速度的变化，但仍理想化地假设加速度或减速度等是固定的，例如使用固定的直行、转向速度和加减速度等静态因素对 AGV 的行驶时间进行计算。Zhang 等[6]则将加减速细分为了更多的阶段。这类处理方法忽略或简化了实际工作中 AGV 复杂的运动状态变化过程，导致行驶时间估计不准确。

基于深度学习的行驶时间估计方法在城市交通领域得到了广泛应用，主要分为两类。第一类是基于路径的方法，即直接考虑整条路径的行驶时间估计。如 Wang 等[7]通过计算具有相似出发点和目的点的历史数据平均时间来进行行驶时间估计；Yuan 等[8]使用一个独特的神经网络，训练时从路网中获取信息而在预估时间时不依赖于轨迹；Zou 等[9]使用 GBDT 和深度神经网络集成，在不需要行车轨迹数据的情况下也能得到结果。该类方法虽然在缺少轨迹信息时也能得到结果，但无法有效利用车辆的轨迹信息，导致时间估计效果相对较差。第二类是基于路段的方法，这类方法将路网中的道路按照交会点切分为路段，计算出一条路径经过的所有路段中各自所需的时间，其总和即为该路径的行驶时间。基于路段的方法的主流做法是将行驶轨迹中坐标组成的轨迹序列转换为路网数据中以路段标号表示的路段序列，再进行后续的计算[10-12]。Wang 等[13]在此基础上，通过更复杂的神经网络引入更多的路段相关特征；Xu 等[14]通过 CNN 和 RNN 的组合来获取路网的时空特征；Liu 等[15]提出的 ST-DGI 模型在路径编码时引入了动态的空间信息；Fu 等[16]将路径标记到地图上后，用图像相关的方法对路段进行编码。

在 AGV 的行驶时间估计问题中，由于自动化集装箱码头本身的一些特性，在应用深度学习方法时，与城市道路环境相比存在以下难点。1）集装箱码头上的交换点十分密集，导致 AGV 的行驶路径灵活且多变。而城市道路由于车道及路口的限制，行驶车辆的路径相对固定，因此城市道路的路径表示方法难以直接应用于 AGV 路径。2）码头面积比城市小很多，时间表示的尺度也更小，而 AGV 的时间估计失准会影响到与其任务相关的其他设备的效率，因此 AGV 行驶时间比城市道路的精度要求更高。3）由于港口作业繁忙，多辆 AGV 同时执行任务时存在路径交叉，不可避免地会发生冲突和避让，从而影响最终的到达时间，而冲突对行驶时间造成的影响又具有较大的不确定性，因此行驶中的冲突和避让是 AGV 行驶时间估计的一项关键难点。

3 AGV 行驶时间估计

3.1 问题描述

自动化集装箱码头上，AGV 作业过程中与其他设备进行交互是在若干位置固定的交换点（TP）完成的。对于 AGV 行驶时间的估计，是由路径规划算法给出预定的目标行驶轨迹之后，在同一时间其他非目标 AGV 行驶路径，即环境路径信息，以及载重等任务特征的辅助下，估计两个 TP 之间的行驶时间，且要求与真实时间尽可能接近。

AGV 的路径规划是由直行、转弯等多种行动模式组合而成的。一条完整的 AGV 路径规划结果由多个直行或转弯的子段路径首尾顺次连接而成，如图 1 所示，从 A 点出发到 B 点，依次是直行、左转、直行、右转、直行的子段。

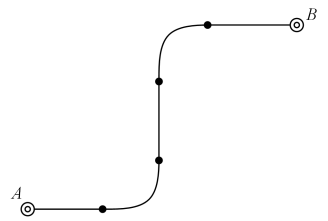

图 1 路径子段划分示意图

Fig. 1 Diagram of path subsegment

从数据角度，一条 AGV 路径规划数据被表示为一个由若干个轨迹坐标点组成的序列，其中任意相邻两点即为某子段路径的起点和终点，同时数据中还记录了相邻两点间的子段路径的具体行动模式，如直行、转弯等。因此，我们可以将 AGV 的路径依据行动模式切分为若干子段，将行驶路径表示为子段序列。对此，有以下定义。

定义 1 一条路径 $R=\{P_{\text{start}},P_{\text{end}},t_{\text{start}},T,F_R\}$。其中，$P_{\text{start}}$和$P_{\text{end}}$分别为路径的起点 TP 和终点 TP；t_{start}为路径出发时间；$P=\{x,y\}$由其坐标表示；$T=\{p_1,p_2,\cdots,p_n\}$为路径对应的子段序列，由一系列子段 $p_i=\{(x_{\text{start}},y_{\text{start}}),(x_{\text{end}},y_{\text{end}}),(x_{\text{dist}},y_{\text{dist}}),\cos\theta,m_{\text{start}},m_{\text{end}}\}$组成，子段中各项分别表示子段起点坐标、子段终点坐标、起点到终点间的坐标差、子段的方向向量余弦 $\cos\theta$ 以及该子段是否为路径的起始段m_{start}或终止段m_{end}的标记；F_R 为路径级特征，包括当前 AGV 所

承载的集装箱重量w和当前AGV正在执行的任务所属的类型，如装船或卸船任务。

定义 2 环境路径$R_e=\{R_{e_1},R_{e_2},\cdots,R_{e_n}\}$是在目标路径前后一段时间内出发的若干条其他路径。$R_{e_i}=\{P_{\text{start}},P_{\text{end}},t_{\text{start}},w\}$包含起点 TP、终点 TP、出发时间以及所装载集装箱的重量。

因此，最终的目标是给出一条路径R，根据在其对应的环境路径R_e影响下，估计出 AGV 完成路径R所需要的行驶时间。

3.2 模型整体结构

本文所提模型的整体结构如图 2 所示。R为需要对其进行行驶时间估计的目标路径，其中轨迹数据T被输入一个路径编码器中进行编码，得到一个固定长度的目标路径向量表示(Target Route Encoding)。路径级特征F_R通过一个全连接层，得到固定长度的路径级特征向量表示(Route-level Feature Encoding)，其中任务类型等离散特征被转换为一个$|O|$维的独热编码表示，其特定一位为 1，其余为 0。例如，假设共有 3 种任务类型，则它们分别用[1,0,0]，[0,1,0]，[0,0,1]表示，数值较大的箱重特征则通过归一化处理缩放为 0～1 之间的浮点数。其余的起点 TP、终点 TP、出发时间以及箱重与环境路径保持一致，共同输入环境路径编码器中，通过一个受注意力机制启发的模型，作为多任务训练的一个辅助任务，以目标路径是否与各条环境路径发生冲突作为标签，训练得到目标路径与各环境路径发生冲突的概率向量(Environment Route Attention Weight)。将目标路径向量表示、路径级特征向量表示以及环境路径冲突概率拼接，并通过全连接层回归估计行驶时间。

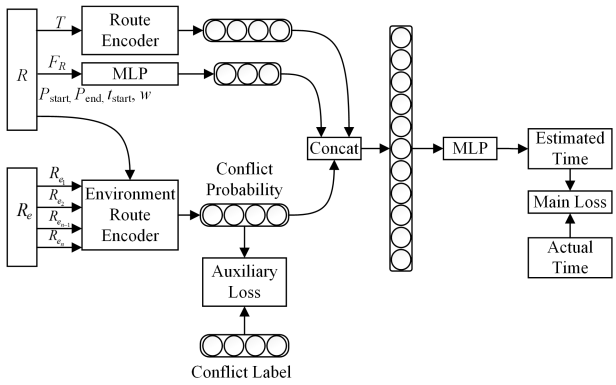

图 2 模型整体结构

Fig. 2 Overall model structure

3.3 目标路径编码

虽然自动化集装箱码头上缺少如城市道路一样的以路口为结点的自然划分，但在我们通过直行、转弯等不同的行动模式，将 AGV 的行驶路径切分为不同的子段后，可以部分参考公共交通领域对路径的编码方式。不同于公共交通领域常见的对路段进行嵌入(Embedding)得到路段编码的方式，本文提出的划分方式所得到的子段并不像城市道路一样是固定的，因此只能用定义 2 中所述的方法对各子段进行表示，再进行后续的处理。

常见的公共交通领域对路径的编码会将路径的子段向量序列输入 RNN 或 LSTM 模型中进行编码。本文也在所提出的模型中使用了类似的处理方式。模型的输入是路径轨迹中每个子段的特征序列。首先，将每一个路径中的n个子段特征都输入到全连接层中，并使用 ReLU 激活函数，得到它们各自的固定长度的向量表示，通过 LSTM 模型对它们进行编码，取最后的隐状态作为目标路径的轨迹数据编码。

由于卷积神经网络能够更好地获取到子段间的局部空间关系[17]，因此本文也尝试使用 CNN 来对路径轨迹数据进行编码。输入部分与 LSTM 的方法相同，通过全连接层将子段特征转换为特征向量，之后分别通过第一个卷积核大小为k的 1 维卷积层、最大池化和第二个卷积核大小同样为k的 1 维卷积层，最终将向量展平，得到轨迹数据的编码。其中，两个卷积层使用的激活函数都是 ReLU。

实验显示，相比使用 LSTM 对路径轨迹进行编码，使用 CNN 进行编码不仅能在估计行驶时间时取得更好的效果，训练的时间效率也更高，因此本文最终选择使用 CNN 的方法。两种方法的比较将在实验部分进行。

3.4 环境路径筛选

由于自动化集装箱码头任务繁忙，如果直接选取目标路径出发时间附近一定窗口内的路径作为环境路径，数量将会非常多，但其中可能会与目标路径发生冲突的却相对比较少，如果将时间窗口内的所有路径都输入模型，大量的环境路径中只有极少的路径会对目标路径的时间估计有影响，其余的大量路径成为噪声信息，不利于模型学习环境路径对目标路径之间的关系。因此，需要对目标路径出发时间的一定时间窗口内的其他路径进行筛选，减少实际输入神经网络模型的数据量。

首先是数据的获取。AGV 行驶过程中保存下的日志中记录了其行驶时是否发生冲突，以及与谁发生了冲突。该冲突是单向记录的，即当A车等待B车让出道路时，只有A车会记录自己与B车发生过冲突。通过对日志数据进行处理，从中找出所有发生过的冲突路径对(R_a^+,R_b^+)，表示路径R_a^+与路径R_b^+发生过冲突，标记为 1，每个路径都表示为$\{P_{\text{start}},P_{\text{end}},t_{\text{start}},w\}$。获取到正样本之后，再从路径数据中随机挑选与正样本数量相当的路径作为负样本的R_a^-，再随机挑选与R_a^-在时间上存在重叠但未发生冲突的路径作为R_b^-，构建出负样本，标记为 0。

针对此类结构化数据的机器学习任务，主流的 XGBoost[18]模型在准确率、训练速度等方面具有巨大优势，该模型有能力满足我们对冲突路径初步筛选的判断。因此在完成数据的准备后，我们使用 XGBoost 来训练筛选模型。将路径对中的 TP 名称映射为其对应的坐标，并加入两路径出发时间差、坐标差、夹角余弦等各类特征，输入模型中，训练出一个能够判断路径对是否会发生冲突的二分类模型。XGBoost 分类器模型使用 gpu_hist 的树方法以便于 GPU 加速，树的最大深度为 11，学习率为 0.04，训练 1200 轮。该模型的预测效果如表 1 所列，这说明 XGBoost 筛选模型有助于去除掉足够多的时间窗口内的非冲突路径，过滤掉大量的不相关的环境路径，有效地减少噪声的输入。

表 1 XGBoost 筛选模型的预测效果

Table 1 Prediction result of XGBoost filter model

（单位：%）

Accuracy	*Recall*	*F*1
86.92	94.64	89.44

在获取到目标路径出发时间的一定时间窗口内若干候选环境路径之后，将目标路径与其环境路径进行同样的特征处理并输入训练好的二分类模型中，输出其预测为“1”的概率，选择概率最高的d_{env}个候选环境路径作为最终输入到神经网络中的环境路径。其中，d_{env}的值应略大于或等于历史统计的单条路径发生的最多冲突次数，但d_{env}设置过大又会引入无关环境路径，从而影响学习效果。

筛选模型的目的是初步筛选环境路径，减少无关数据特征。之后，筛选过的环境路径还需编码后与主路径编码共同训练学习。因此，在筛选出的环境路径数量d_{env}大于或等于历史最大路径冲突次数的情况下，利用 XGBoost 筛选模型过滤环境路径，能够满足为后续的环境路径编码学习过滤无关特征的需求。

3.5 环境路径编码

AGV 作为自动化设备，其在两点间的行驶时间理论上应该是基本保持不变的，而对行驶时间估计造成影响的一大因素就是车辆行驶过程中发生冲突而产生的避让等待。通过 XGBoost 筛选模型，我们已经能够得到 K 条最可能与目标路径发生冲突的环境路径，但由于筛选出的环境路径数量固定为 K，而每条路径实际发生冲突的次数各不相同，并且从表 1 可以看出，即使 XGBoost 筛选模型有着比较可观的准确率，依然存在错误判断的可能，因此需要使用神经网络模型对筛选出的环境路径是否发生冲突做进一步判断。此外，筛选出的环境路径如何影响目标路径的时间依然是未知的，因此我们对环境路径编码的目的，就是要聚合目标路径与环境路径间的冲突对行驶时间的影响。这些信息有助于我们更加准确地估计出车辆间的冲突对行驶时间造成的影响。

本文提出的对路径编码的模型参考了注意力机制的思想。注意力机制在深度学习各领域中被广泛使用，可以被简单解释为一个元素与其他若干元素之间的重要性权重向量，在本文的问题中，即是目标路径与各环境路径之间是否会发生冲突的概率向量。

本文提出的环境路径编码模型如图 3 所示。模型的输入部分包括目标路径和若干环境路径的一些基本属性，以及通过目标路径编码器得到的目标路径编码。目标路径和环境路径的基本属性$\{P_{\text{start}}, P_{\text{end}}, t_{\text{start}}, w\}$在输入到特征处理模块之后进行了一些简单的处理，得到了一系列路径对特征：目标路径起止 TP 坐标，环境路径起止 TP 坐标，两者间的出发时间差，两路径装载的集装箱重，两路径自身的起止点间 XY 轴坐标差，两路径之间、起止点的 XY 坐标差，以及两路径各自的方向余弦与两路径之间的夹角余弦，这些特征与路径筛选时使用的特征是一致的。在得到路径对特征之后，将其输入全连接层，可得到与目标路径编码相同维度的环境路径特征向量$\boldsymbol{E}_{\text{rp}}$，接着计算目标路径编码与各环境路径特征向量的点积，得到注意力得分 S。在这之后，若按照注意力机制的做法，会使用一个 softmax 激活函数，将得分转化为和为 1 的概率分布。但在该问题中，一条目标路径会与多条环境路径发生冲突，因此本文使用了 sigmoid 激活函数 $S(x)=1/(1+e^{-x})$，让每一条环境路径都可能被判断为与目标路径发生冲突，如此一来便可依据其输出与真实的冲突情况计算损失值，从而作为时间估计模型的辅助任务。整个过程的形式化表示如下：

$$A=\text{sigmoid}(S) \tag{1}$$

$$S=\boldsymbol{E}_{\text{rp}} \cdot \boldsymbol{R}_{\text{main}} \tag{2}$$

$$\boldsymbol{E}_{\text{rp}}=\boldsymbol{W}_e^{(2)}\,\text{ReLU}(\boldsymbol{W}_e^{(1)}\boldsymbol{F}_{\text{rp}}+b_e^{(1)})+b_e^{(2)} \tag{3}$$

其中，$A\in\boldsymbol{R}^{d_{\text{env}}}$ 和 $S\in\boldsymbol{R}^{d_{\text{env}}}$ 是最终输出的模型判断的目标路径与环境路径是否冲突的结果及其注意力得分，d_{env}是经过筛选后剩下的环境路径数量；$\boldsymbol{E}_{\text{rp}}\in\boldsymbol{R}^{d_{\text{env}}d_{\text{main}}}$ 是经过全连接层之后的路径对特征向量，$\boldsymbol{F}_{\text{rp}}\in\boldsymbol{R}^{d_{\text{rp}}}$ 是路径对特征；$\boldsymbol{W}_e^{(1)}\in\boldsymbol{R}^{d_e^{(1)}\times d_{\text{rp}}}$，$\boldsymbol{W}_e^{(2)}\in\boldsymbol{R}^{d_e^{(2)}\times d_e^{(1)}}$，$b_e^{(1)}\in\boldsymbol{R}^{d_e^{(1)}}$，$b_e^{(2)}\in\boldsymbol{R}^{d_e^{(2)}}$ 都是可训练的全连接层参数。

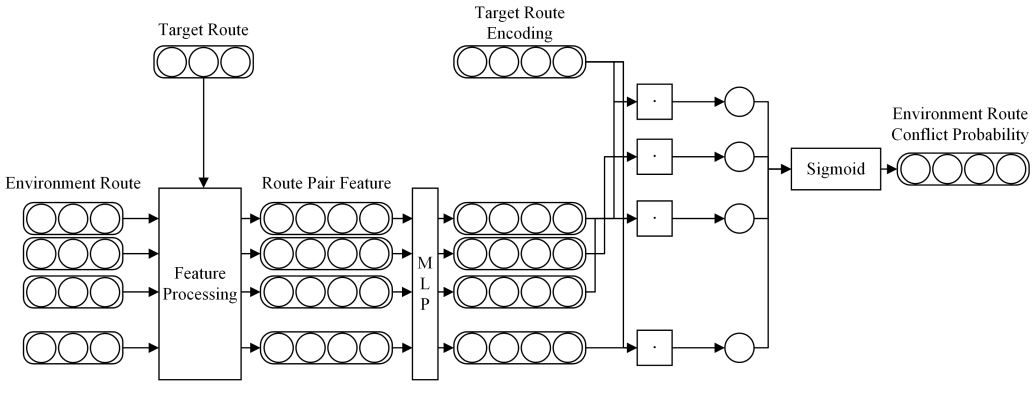

图 3 环境路径编码器结构

Fig. 3 Structure of environment path encoder

3.6 模型训练

对模型进行训练时，本文使用了多任务训练的方法。总共分为两个任务，第一个是作为主要任务的行驶时间估计，第二个是作为辅助任务的环境路径冲突估计。两个任务使用不同的损失函数，主任务基于模型估计的行驶时间与真实的行驶时间，使用均方误差（Mean Squared Error，MSE）计算主损失值（Main Loss），如式（4）所示。辅助任务使用模型估计的目标路径和各环境路径发生冲突的概率向量，与目标路径

和各环境路径之间是否实际发生冲突的 01 标签计算二元交叉熵(Binary Cross Entropy,BCE)作为辅助损失值(Auxiliary Loss),如式(5)所示。在训练时,将两个损失值的加权求和作为总损失值,其计算式如式(6)所示。

$$loss_m=\frac{1}{m}\sum_{i=1}^{m}(y_i^t-\hat{y}_i^t)^2 \tag{4}$$

$$loss_a=-\frac{1}{m}\frac{1}{n}\sum_{i=1}^{m}\sum_{j=1}^{n}y_{ij}^c*\log(\hat{y}_{ij}^c)+(1-y_{ij}^c)*\log(1-\hat{y}_{ij}^c) \tag{5}$$

$$loss=w*loss_m+(1-w)*loss_a \tag{6}$$

其中,y_i^t是第 i 条目标路径的实际行驶时间;$\hat{y}_i^t$是模型估计出的第 i 条目标路径的行驶时间;y_{ij}^c 是第 i 条目标路径与其对应的第 j 条环境路径是否发生冲突的标签;$\hat{y}_{ij}^c$ 是模型估计的第 i 条目标路径与其对应的第 j 条环境路径发生冲突的概率;m 是路径样本数;n 是一条目标路径对应的环境路径数;w 是计算总损失值的权重,为一个可调的超参数。

3.7 算法整体运行流程

综上所述,本文所提算法的整体运行流程如图 4 所示。首先,从 AGV 相关的历史数据中整理出 AGV 的路径数据及相关特征,同时根据行动模式划分路径子段;然后,根据路径数据与 AGV 行驶日志,构建 AGV 行驶路径的冲突路径对作为环境路径筛选模型的正样本,并按照 3.4 节中的方式构造冲突负样本,用于筛选模型的训练;完成筛选模型训练后,将每条目标路径与其出发时间窗口内的其他环境路径分别组合为路径对并输入筛选模型,筛选模型输出概率最高的d_{env}条路径与目标路径组合起来得到最终输入多任务深度学习模型的样本;最后,采用梯度下降算法训练由主路径编码器和环境路径编码器组成的多任务深度学习模型,实现对目标路径行驶时间的估计。

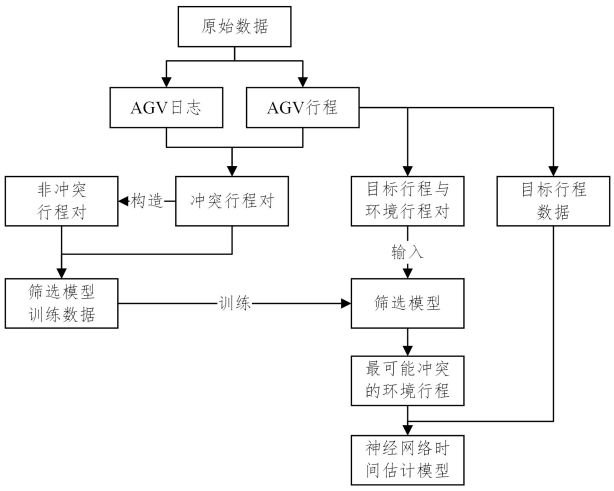

图 4 算法整体运行流程

Fig. 4 Overall flow of the proposed algorithm

4 实验

4.1 数据集

本文在上海港洋山四期自动化码头的真实历史数据上对所提出的模型进行了测试,使用的数据为 2020 年 4 月的数据。与 AGV 相关的数据分散在多个数据表中:指令表中记录了 AGV 指令的信息,包括任务的类型、目的 TP 的类型等可能影响时间估计的信息,约 43 万条;轨迹表中记录了所有 AGV 在各个时间所处的位置信息、行驶里程、直行或转向等,约 104 万条;此外,还有如集装箱重量等信息记录于作业指令数据表中。由于一条 AGV 指令会经过多个 TP 点并可能在其上停留以进行交互,而本文只关注任意两个 TP 之间的行驶时间,因此需要对以上数据进行匹配和重构。经过一些必要的处理和筛选之后,最终得到了约 37 万条路径,共涉及码头上的 90 辆 AGV 以及超过 1 200 个交换点。每条路径都包含了各自的轨迹数据与特征,环境路径则通过预处理完成了筛选与路径对特征处理等工作。数据被随机划分为了两部分:占 90%的训练集(约 33 万条路径)与占 10%的测试集(约 4 万条路径)。

4.2 对比方法

由于与该问题直接相关的研究较少,除了路径规划问题中常见的速度固定的估算方法之外,本文还选择了一个比较基础的神经网络回归模型和一个常见的机器学习模型 XGBoost 进行比较,以证明所提方法的有效性。

(1)速度固定的估算方法。很多 AGV 路径规划问题中都会使用该方法来预估 AGV 的行驶时间。由于 AGV 是一个自动化设备,其在正常直行时会保持相对恒定的速度,该算法虽然缺少对起步、停车与转弯等情况速度变化的考虑,但依然可以一定程度上估算出 AGV 的行驶时间。实验采用的固定速度是所有路径的平均速度。

(2)神经网络回归方法。该方法类似于公共交通领域行驶时间估计问题的基于路径方法的思想,即假设起点和终点确定后,其所行驶的路线便确定了,那么行驶时间也能够确定。将 TP 映射为固定的 ID,再通过 embedding 转化为 TP 的向量表示,拼接上F_R路径级特征的向量表示之后,通过全连接层进行回归得到时间估计值。

(3)XGBoost。XGBoost 是一个基于决策树的集成学习方法,具有高效、灵活的特点,能完成分类、回归等任务。在所进行的实验中,输入 XGBoost 的数据与本文提出的神经网络模型基本相同,即路径各子段的特征通过 XGBoost 模型估计出子段的行驶时间,再将其相加得到整条路径的行驶时间。

4.3 评价指标

规定$t_1,t_2,\cdots,t_n$是实际的行驶时间,$\hat{t}_1,\hat{t}_2,\cdots,\hat{t}_n$是估计出的时间,$n$ 是路径数量。对于此类问题,一般采用 3 个指标来评价:绝对百分比误差(Mean Absolute Percentage Error,MAPE)、平均相对误差(Mean Relative Error,MRE)和平均绝对误差(Mean Absolute Error,MAE)。它们的计算方法如下:

$$MAPE=\frac{1}{n}*\sum_{i=1}^{n}\left|\frac{t_i-\hat{t}_i}{t_i}\right|*100\% \tag{7}$$

$$MRE=\frac{\sum_{i=1}^{n}|t_i-\hat{t}_i|}{\sum_{i=1}^{n}|t_i|}*100\% \tag{8}$$

$$MAE=\frac{\sum_{i=1}^{n}|t_i-\hat{t}_i|}{n} \tag{9}$$

4.4 参数设置

所提模型基于 Keras 框架实现，使用 Adam 优化器，多任务训练的损失函数权重 w 为 0.9，*batch-size* 设置为 512，学习率为 0.000 05（过程中保持不变），共训练 250 个 epoch。此外，用于筛选环境路径的时间窗口设置为目标路径出发前后各 10 min，共约 400 条路径。图 5 给出了所用数据中各路径发生冲突次数的统计结果，可以看到，绝大多数路径发生冲突的次数都小于 10，因此我们选择与目标路径发生冲突概率最高的 10 条路径作为环境路径。

此外，模型中的各个超参数分别为：路径中子段数量 $d_t=21$，原始子段特征维度 $d_p=9$，$d_{\text{main}}=128$，$d_{\text{env}}=20$，$d_{\text{rp}}=22$，$d_e^{(1)}=128$，$d_e^{(2)}=256$，$k=3$。

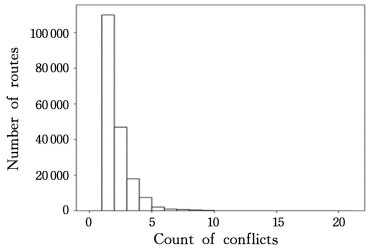

图 5 各路径发生冲突次数的统计

Fig. 5 Count of conflicts

4.5 多任务训练损失值权重的选择

为了微调多任务训练时的损失值权重 w，多次调整 w 的值并进行了对比实验。w 取值从 0.1 增长到 0.9，每次增加 0.1，共进行 9 次实验，每一次训练都记录下 3 项指标，结果如图 6 所示。

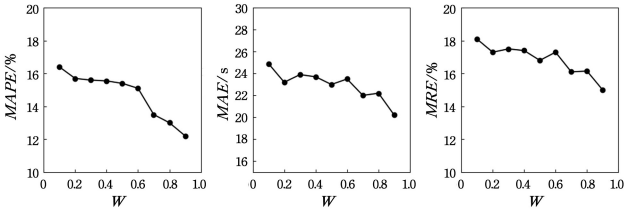

图 6 各权重下模型评价指标的变化

Fig. 6 Change of metrics with different loss weights

从图 6 中可以看出，3 项指标变化的总体趋势都是随着 w 的增大而逐渐变好，且都在 $w=0.9$ 时取得了最好的效果，因此在之后进行的各项实验中都将多任务训练权重 w 的取值设为 0.9。

4.6 性能比较

表 2 列出了所提模型与其他对比方法的性能比较，可以看出，传统路径规划与 AGV 调度问题中常用的固定车速进行估算的方法并不十分准确，这也印证了之前的假设，即 AGV 作为自动化设备，其车速在正常境况下趋于恒定，因此该方法的结果并非极端的糟糕。使用简单的神经网络进行回归自然能取得更好的结果，但由于没有使用到路径数据，无法捕捉到实际行驶中的各种情况对行驶时间造成的影响，其结果无法和另外两种使用了路径数据的方法相比较。XGBoost 和本文所提出的方法都使用了路径数据，而本文的方法在环境路径编码的辅助下，3 项指标都取得了最好的效果。

表 2 时间估计方法的性能比较

Table 2 Performance comparison of different time estimation methods

Method	*MAPE*/%	*MAE*/s	*MRE*/%
Static Velocity	39.316 7	55.352 1	41.485 6
Base-NN	22.197 4	26.351 9	19.459 7
XGBoost	14.701 3	20.780 2	15.311 6
Ours	**12.355 9**	**20.218 1**	**15.023 2**

4.7 目标路径编码器的比较与环境路径的影响

为了明确目标路径编码器的选择以及环境路径对结果的影响，我们对模型进行了修改，进行了 3 组额外的实验，包括：1）目标路径使用 CNN 编码，不使用环境路径；2）目标路径使用 LSTM 编码，使用环境路径；3）目标路径使用 LSTM 编码，不使用环境路径。使用到的数据以及参数都保持一致，实验结果如表 3 所列。

表 3 不同目标路径编码器以及环境路径编码器的性能比较

Table 3 Performance comparison of different destination route encoders and environment route encoders

Method	*MAPE*/%	*MAE*/s	*MRE*/%
LSTM	15.884 0	24.078 5	17.762 7
LSTM+ENV	14.367 6	22.185 7	16.354 7
CNN	13.617 4	22.363 3	16.483 7
CNN+ENV	**12.355 9**	**20.218 1**	**15.023 2**

从对比中可以看出：1）CNN 作为目标路径编码器，比 LSTM 有更好的效果（*MAPE* 分别提升 2.266 6% 和 2.011 7%），在 3 个指标上都取得了比较明显的提升，这是由于本文在对路径进行分段时，是以直行、转弯等模式来进行划分的，因此采用 CNN 作为目标路径的编码器能够更好地捕获到这类局部信息，如直行后转弯会存在减速；2）加入环境路径的编码之后，无论目标路径采用哪种方式编码，估计的效果都得到了明显的提升（*MAPE* 分别提升 1.516 4% 和 1.261 5%），可以看出路径间的冲突等待对时间估计的准确度有较大影响，而能否准确估计路径间的冲突，将极大地影响到时间估计的准确性。

此外，我们还对比了 4 种方法的训练时间，如表 4 所列。从表中可以看出，完成同样的 250 轮训练，使用 LSTM 作为目标路径编码器的模型所消耗的时间要远多于使用 CNN 的模型，可见使用 CNN 作为目标路径编码器不仅能取得更好的效果，在训练时间上也比 LSTM 的模型有着更大的优势。

表 4 不同方法的训练时间比较

Table 4 Training time comparison of different methods

（单位：s）

Method	Time
LSTM	8 012
LSTM+ENV	9 172
CNN	1 498
CNN+ENV	2 180

结束语 本文提出了一个使用真实历史数据来估计自动化集装箱码头上 AGV 行驶时间的神经网络模型。该模型使用 CNN 来对路径的轨迹数据进行编码，并引入了环境路径这一概念，以获取 AGV 行驶时相互之间发生冲突从而对

行驶时间造成的影响。通过与多种估计方法进行比较，证明了所提方法在3种指标上都取得了最好的结果，并通过额外的实验对比，证明了使用CNN作为目标路径编码器以及环境路径编码器的有效性。在进行实验时，我们曾经将路径实际发生冲突的次数作为特征直接输入到模型中，得到了更好的效果。然而，该数据在AGV出发前是无法得到的，因此最终并未使用该特征，但是这也说明，对路径间冲突的预测对最终结果会有较大影响且还有很大的提升空间，这也将是今后改进的方向。

致谢 感谢上海国际港务(集团)股份有限公司为本文设计及撰写提供业务指导和实验数据等相关支持。

参 考 文 献

[1] HUO K G, ZHANG Y Q, HU Z H. Research on scheduling problem of multiload AGV at automated container terminal[J]. Journal of Dalian University of Technology, 2016, 56(3): 244-251.

[2] WANG H C, ZHANG Z W, ZHAO Y, et al. A dynamic time prediction method of AGV for vertical transportation: CN CN202110027744.2[P]. 2021-04-30.

[3] ZHONG M, YANG Y, DESSOUKY Y, et al. Multi-AGV scheduling for conflict-free path planning in automated container terminals [J]. Computers & Industrial Engineering, 2020, 142(Apr.): 106371.1-106371.11.

[4] YUE X, XU X, WANG X. Research on MutiAGV Sechduling Algorithm Based on Improved Hybrid PSOGA for FMS[J]. Computer Science, 2018, 45(S2): 167-171.

[5] GUO K, ZHU J, SHEN L. An Improved Acceleration Method Based on Multi-Agent System for AGVs Conflict-Free Path Planning in Automated Terminals[J]. IEEE Access, 2020, 9: 3326-3338.

[6] ZHANG Z, WU L, ZHANG W, et al. Energy-efficient path planning for a single-load automated guided vehicle in a manufacturing workshop[J]. Computers & Industrial Engineering, 2021, 158: 107397.

[7] WANG H, TANG X, KUO Y H, et al. A simple baseline for travel time estimation using large-scale trip data [J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2019, 10(2): 1-22.

[8] YUAN H, LI G, BAO Z, et al. Effective travel time estimation: When historical trajectories over road networks matter[C]// Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data. 2020: 2135-2149.

[9] ZOU Z, YANG H, ZHU A X. Estimation of travel time based on ensemble method with multi-modality perspective urban big data [J]. IEEE Access, 2020, 8: 24819-24828.

[10] DE FABRITIIS C, RAGONA R, VALENTI G. Traffic estimation and prediction based on real time floating car data[C]// 2008 11th International IEEE Conference on Intelligent Transportation Systems. IEEE, 2008: 197-203.

[11] HOFLEITNER A, HERRING R, ABBEEL P, et al. Learning the dynamics of arterial traffic from probe data using a dynamic Bayesian network[J]. IEEE Transactions on Intelligent Transportation Systems, 2012, 13(4): 1679-1693.

[12] HUNTER T, HERRING R, ABBEEL P, et al. Path and travel time inference from GPS probe vehicle data[J]. NIPS Analyzing Networks and Learning with Graphs, 2009, 12(1): 2.

[13] WANG Z, FU K, YE J. Learning to estimate the travel time [C]// Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018: 858-866.

[14] XU J, ZHANG Y, CHAO L, et al. STDR: a deep learning method for travel time estimation[C]// International Conference on Database Systems for Advanced Applications. Cham: Springer, 2019: 156-172.

[15] LIU W, HE J, WANG H, et al. A Novel Road Segment Representation Method for Travel Time Estimation[C]// DASFAA (Workshops). 2021: 398-413.

[16] FU T, LEE W C. Deepist: Deep image-based spatio-temporal network for travel time estimation[C]// Proceedings of the 28th ACM International Conference on Information and Knowledge Management. 2019: 69-78.

[17] WANG D, ZHANG J, CAO W, et al. When will you arrive? estimating travel time based on deep neural networks[C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2018.

[18] CHEN T, GUESTRIN C. Xgboost: A scalable tree boosting system[C]// Proceedings of the 22ndACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016: 785-794.

LENG Dian-dian, born in 1996, postgraduate. His main research interests include big data and machine learning.

XIANG Yang, born in 1962, Ph.D, professor, Ph. D supervisor, is a senior member of China Computer Federation. His main research interests include natural language processing, data mining, knowledge graph, and so on.

(责任编辑：柯颖)