

路网空间下基于马尔可夫决策过程的 异常车辆轨迹检测算法

毛江云 吴 昊 孙未未

(复旦大学计算机科学技术学院 上海 201203)

(上海市数据科学重点实验室 上海 201203)

摘 要 随着 Internet、移动通信、空间定位和 LBS 技术的发展,越来越多的车辆轨迹被收集,如何从大量的车辆轨迹中高效检测出异常轨迹逐渐引起人们的关注. 研究人员提出了许多针对车辆轨迹的异常检测方案,从采用的算法来划分,这些方案被分为三类:基于度量的算法、基于统计的算法和基于监督与半监督学习的算法. 三类算法都各自存在不足:第一类的计算量随轨迹数据量的增长而增长,对异常特征的刻画不完整;第二类严重依赖历史数据,因此没有办法解决轨迹稀疏问题;第三类需要大量的人工标注. 该文提出了一套路网空间下基于马尔可夫决策过程的异常车辆轨迹检测算法,该算法总共分为预处理、离线训练和在线检测三个阶段. 预处理阶段采用了隐马尔可夫地图匹配算法作为核心,将原出租车轨迹转化为由路网空间中路段边序列表示的轨迹集合. 离线训练阶段采用了马尔可夫决策过程模型对车辆驾驶行为进行建模,深入讨论了模型中路段奖励函数的设计规则,并提出采用无监督的贝叶斯反向增强学习算法配合蒙特卡罗采样算法训练历史车辆轨迹数据学习得到模型参数. 在线检测阶段中,实时计算待检测的轨迹的异常度,通过用户指定的异常度阈值判断该轨迹是否为异常车辆轨迹. 最后,在真实数据集上进行实验,同时实现了 iBOAT 算法和 MEX 算法,并作为对比算法. 正确性实验中,该算法在 NDCG 评测指标中达到了 99.3% 的正确率;在算法的运行时间上,该算法的单条轨迹在线检测时间能够做到仅耗时 0.012 ms,较已有算法提升百倍到千倍的效率. 在稀疏数据下进行结果正确性实验,在对比算法的效果严重受影响的情况下,该算法依然展现出很强的鲁棒性. 在样例分析中可以看到通过该算法计算得到的路段奖励函数数值和对真实驾驶行为的评估高度一致.

关键词 异常检测;轨迹计算;马尔可夫决策过程;增强学习;基于位置的服务

中图法分类号 TP311 **DOI 号** 10.11897/SP.J.1016.2018.01928

Vehicle Trajectory Anomaly Detection in Road Network via Markov Decision Process

MAO Jiang-Yun WU Hao SUN Wei-Wei

(School of Computer Science, Fudan University, Shanghai 201203)

(Shanghai Key Laboratory of Data Science, Fudan University, Shanghai 201203)

Abstract With the bloom of the Internet and the Location-based Services (LBSs) as well as the development of positioning technologies, more and more vehicles are equipped with positioning devices such as GPS devices, which make it possible to collect and process such an increasing number of trajectory data. It also raises interest for many researchers to study how to effectively detect the anomalies in trajectory database as well as maintain the efficiency when the data volume grows larger. This paper analyzes abnormal vehicle trajectory detection literatures at home and abroad and finds that in the current literature, the trajectory anomaly detection research can be

收稿日期:2016-12-13;在线出版日期:2017-03-22. 本课题得到国家自然科学基金(61772138)和上海市自然科学基金(14ZR1403100)资助. 毛江云,女,1990年生,硕士研究生,主要研究方向为移动数据管理和异常检测. E-mail: jyao14@fudan.edu.cn. 吴昊,男,1992年生,博士研究生,主要研究方向为轨迹计算. 孙未未(通信作者),男,1973年生,博士,教授,中国计算机学会(CCF)高级会员,主要研究方向为时空数据处理和分析.

divided into three categories: metric-based approaches, statistics-based approaches as well as supervised and semi-supervised learning-based approaches. However, after making a comprehensive discussion and analysis on the correctness of trajectory representation, the performance on sparse data and the feasibility among the existing approaches, we find that these above approaches all have disadvantages. For the metric-based approaches, the computation cost will be growing with the increase of data volume. Moreover, they cannot fully characterize the trajectory since only distance and density features are taken into consideration. The statistics-based approaches highly rely on the historical data which means that the performance will suffer a large drop when the data sparsity occurs. While the problem of supervised and semi-supervised learning-based approaches lay on the troublesome and impractical manual annotation. Motivated by the disadvantages of existing works, this paper proposes a Markov Decision Process (MDP) based trajectory anomaly detection approach in the road network space, and this architecture can be divided into three phases: preprocessing, offline training and online detection. In the preprocessing phase, a hidden Markov model based approach is adopted to convert the spatial trajectories into the sequence of road segments which are based on the origin and destination. In the offline training phase, the MDP model is adopted to model the routing behavior of drivers. As for the reward function of road segment, it is delicately designed to fit the anomaly detection in the road network space. And the Bayesian inverse reinforcement learning (BIRL) algorithm along with the Markov Chain Monte-Carlo (MCMC) sampling approach is leveraged to learn the reward function by the historical trajectory data. In online detection phase, it computes the abnormal score from the output of the MDP model and judges whether the trajectory is an anomaly based on the threshold given by the user. At last, extensive experiments are conducted by using real world dataset which is generated from a city called Porto. In addition, two approaches, namely iBOAT and MEX are implemented as baselines. The proposed approach of this paper achieves 99.3% for NDCG metric which outperforms the baselines. What's more, the approach only cost 0.012ms for online anomaly detection while the baselines are several hundreds to thousands much slower. And in the experiment of sparse data, the approach shows the strong robustness against data sparsity in contrast to the significant performance drop of baselines. Finally, the case study further justifies the correctness of the reward function estimated by the model and the modeling for driving behavior.

Keywords anomaly detection; trajectory computing; Markov decision process; reinforcement learning; location-based services

1 引言

随着移动通信、空间定位和位置服务等技术的不断发展,越来越多的移动对象,尤其是私家车、出租车等车辆,配备了诸如 GPS 等定位设备,使得人们可以收集和存储更多的车辆轨迹数据。如何快速处理和有效利用大量的车辆轨迹数据以服务于智能交通、智慧城市等领域引起了大量研究人员的兴趣^[1-2]。其中车辆的异常轨迹检测工作,即检测起终点 (Origin-Destination, OD) 相同的轨迹集合下“少而不同”^[3-4]的轨迹工作是一个重要的研究点。具

体而言,这里讨论的异常轨迹是指:(1)在轨迹集合中相同轨迹数量少,(2)其行为特征不同于大部分正常的轨迹,也就是说,其行驶路线往往与主流轨迹的路线有较大区别。异常轨迹检测有很多应用场景,如交通事故中肇事方追责、出租车司机绕路欺诈检测、城市电子地图更新、人群行为研究等,近年来逐渐引起人们的关注。

异常检测在数据挖掘中被称为离群点检测问题。离群点是数据集中少量、显著不同于其他数据点的对象。常见的离群点检测算法大体可以分为基于统计的算法、基于深度的算法、基于距离的算法、基于密度的算法和面向高维数据的算法等^[5]。但是车

辆轨迹异常检测工作不同于传统的异常检测工作. 它面临着更大的挑战: (1) 车辆轨迹数据不同于常见的结构化数据, 它是多维非结构化数据, 同时包含空间、时间及其他多维度信息(如速度、角度); (2) 随着车载定位设备的普及化, 车辆轨迹数据与日俱增, 呈现出爆发式增长, 因而积累了海量的历史数据, 并且同时还拥有着大量源源不断的实时数据; (3) 车辆轨迹数据在宏观上呈现出海量性, 但是在微观局部中仍体现出稀疏性, 尤其是在长距离轨迹情况下. 我们通过对葡萄牙出租车数据集进行统计, 发现轨迹总长度大于 15 km 的轨迹总数只占总数据集的 0.7%. 这也就意味着, 在长距离轨迹情况下, 拥有相同行驶路径的车辆轨迹变得稀少, 这个被称为轨迹稀疏的问题在处理车辆轨迹工作中不容忽视. 特别地, 异常轨迹在数据稀疏情况下“少而不同”的特征表现弱, 加大了误判的概率. 因此针对该问题需要提出一种适用于车辆轨迹的快速、准确性高且能应对数据稀疏的异常检测算法.

目前已有许多针对车辆轨迹的异常检测工作^[3-4, 6-15], 它们大致可以分为基于度量的方法、基于统计的方法和基于监督与半监督学习的算法. 综合这些方法, 存在以下缺陷:

(1) 对于基于度量的方法而言, 首先其计算量随着轨迹数据量的增长而增长, 其次上述工作只考虑了轨迹的距离、密度等特征, 对异常特征的刻画不完整.

(2) 基于统计的方法则严重依赖于历史数据的支持, 当历史数据稀少时直接影响检测结果的正确性. 也就是说, 该方法没有办法解决轨迹稀疏尤其是数据缺失的问题.

(3) 对于基于监督与半监督学习的算法, 不足之处是需要对训练集数据进行人工标注. 第一, 人工标注的结果因人而异; 第二, 真实道路网络日新月异, 因此标注工作必须持续更新; 第三, 为了达到良好的效果, 大量的人力标注工作是不可避免的. 因此基于人工标注的方法不具有实用性.

本文提出一套基于路网空间的在线异常车辆轨迹检测算法. 在解决问题时, 我们用概率模型对轨迹进行建模, 异常轨迹的问题被形式化为轨迹的似然概率. 具体地, 采用马尔可夫决策过程对驾驶员的驾驶行为进行建模, 接着使用无监督的贝叶斯反增强学习对历史轨迹数据进行训练学习, 配合蒙特卡罗采样算法学习出路段的潜在开销, 进而得到路段之

间的转移概率, 为异常轨迹检测提供支撑. 本文的贡献在于:

(1) 不同于已有的工作, 我们从驾驶行为建模的角度研究轨迹异常, 使用无监督学习的方法在路网空间上解决在线车辆轨迹的异常检测工作问题.

(2) 利用马尔可夫决策过程以及反向增强学习的方法, 学习出路段的潜在开销, 解决了数据稀疏的问题.

(3) 结合真实出租车轨迹数据, 进行了大量的实验, 验证了算法的高效性和鲁棒性.

本文第 1 节介绍针对车辆轨迹的异常检测研究现状; 第 2 节对相关工作进行分析; 第 3 节将给出本算法框架的系统概览图; 第 4 节对问题进行描述, 并提出异常度的判断标准和马尔可夫决策过程的准确定义; 第 5 节将详细展开技术细节的介绍, 包括离线模型学习阶段和在线的异常检测过程; 第 6 节将对所提到的方法进行实验分析; 最后, 总结全文并展望后续工作.

2 相关工作

近年来, 针对车辆轨迹的异常检测工作取得了长足的发展^[3-4, 6-15], 总结已有方法, 大致可以分为以下 3 类: (1) 基于距离、密度、误差等信息的基于度量的方法; (2) 基于轨迹数量, 车流量等信息的基于统计的方法; (3) 通过提取和训练轨迹特征信息的基于监督与半监督学习的算法.

第一类是基于度量的方法. 以 Knorr 和 Ng 等人^[6]为代表, 提出了“离群点”的概念, 并提出了借助网格索引将 $O(kN^2)$ 的时间复杂度优化至线性的算法. Ge 等人^[7]基于 Dempster-Schafer 理论, 提出结合距离和密度的模型来识别欺诈出租车轨迹, 首先采用独立成分分析方法计算一组轨迹的独立成分, 其次计算每条轨迹的 coding cost, 于是得到了在起始点之间的“最常见路线”, 最后只需要计算对应的每条轨迹和这个“最常见路线”的相异度即得到每条轨迹的异常度. Lee 等人^[8]则提出一种基于轨迹划分的异常轨迹检测框架, 他们首先将轨迹划分成等长的线段集合, 接着采用基于距离和密度混合的 Hausdorff 距离标准进行检测. Yu 等人^[9]提出了一套多参数的分别基于“离群点”和“离群轨迹”的异常检测算法, 计算实时流“窗口”(固定长度的子轨迹)中每个采样点的时空近邻点或者轨迹数量的支持度

进而判断子轨迹异常与否。

第二类是基于统计的方法, Li 等人^[10]对比当前数据和历史在时间维度上的发展趋势, 构造 N 的相似向量, N 为道路网络路段数量, 通过检测趋势的剧变来确定离群路段。Pan 等人^[11]提出结合车辆轨迹数据和社交网络数据来检测异常并进一步分析引起异常的原因, 在检测异常车辆轨迹阶段, 以路径模式 (routing pattern) 为单位计算其车流量的占比比率, 通过对比比率的显著变化来检测异常, 接着利用社交网络来挖掘分析引起该异常的原因。Zhang 等人^[3]则基于异常车辆轨迹“少 (few)”和“不同 (different)”两个显著特征, 提出了基于隔离点的异常轨迹检测系统 (iBAT)。该方法首先将空间划分成等尺度的网格, 接着把轨迹转换成网格表示的网格序列, 最后通过计算网格序列的频度来检测异常轨迹, 不过该方法需要多次遍历在海量的历史轨迹数据集, 耗时过大; 在此基础上, Chen 等人^[4]提出了一套基于隔离点的异常轨迹在线检测系统 (iBOAT), 解决了 iBAT 在线检测和寻找局部异常轨迹的问题。该方法通过对起始点一致的轨迹集合建立倒排索引来索引历史轨迹, 接着维护一个 k 长度的滑动窗口, 实时计算窗口中的网格序列的异常值, 从而检测出异常轨迹。Lei^[12]综合考虑轨迹在空间、时间、行为三个维度的特征, 建立异常轨迹检测系统, 同样的, 他首先将轨迹转化为网格序列, 接着构建概率后缀树 (PST-Tree) 进而检测异常轨迹, 不过该方法的处理对象是船舶的轨迹数据, 该类轨迹和车辆轨迹的区别在于前者基于欧式空间, 而后者则基于路网空间。

第三类是基于监督与半监督学习的算法。Li 等人^[13]提出了一套基于分类的异常轨迹检测算法, 在该算法中, 首先提取出共同模式 (被称为 motifs), 接

着这些共同模式组成一个特征空间, 轨迹数据继而 被转换到特征空间中, 形成特征向量, 之后可用分类器进行分类, 检测出异常轨迹。Liao 等人^[14]设计了一套可视化分析系统来检测租车的异常轨迹。他们利用条件随机场建立模型, 标注得到异常轨迹, 再加上主动学习的场景, 即利用用户交互逐步提高他们的模型。Sillito 和 Fisher^[15]开发出一套增量半监督学习方法来检测异常, 他们把已标注的数据和未标注的数据糅合在一起, 统一运用分类器分类, 未被标注的数据可以借助人后期修正后再加入训练以提高模型的准确性。然而该方法处理的对象是视频流数据而非轨迹数据。

3 系统概览

本节将介绍我们提出的基于路网空间的在线异常车辆轨迹检测算法。如图 1 所示, 算法分为三个阶段: 预处理、离线训练和在线检测阶段。在预处理阶段, 首先采用基于隐马尔可夫模型 (Hidden Markov Model, HMM) 的地图匹配算法^[16], 将由点序列表示的轨迹转化为由路段边序列表示的轨迹, 接着根据相同的起始路段 (OD) 提取出 OD 轨迹集。在离线训练阶段, 我们使用马尔可夫决策过程 (Markov Decision Process, MDP) 模型对驾驶行为建模, 根据 OD 轨迹集数据, 采用贝叶斯反向增强学习 (Bayesian Inverse Reinforcement Learning, BIRL) 算法不断训练道路网络中的路段潜在开销函数。而在在线检测阶段, 输入一条待检测轨迹, 通过离线训练得到的路段潜在开销函数构建的完整的 MDP 异常检测模型计算得到异常度, 配以按需缓存策略的加速, 高效地在线给出轨迹的异常度 (或根据异常阈值判断异常与否)。

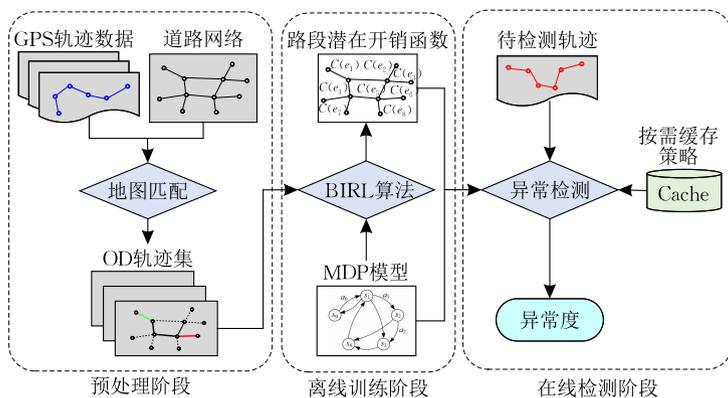


图 1 异常车辆轨迹检测系统流程图

4 基本概念

本节将给出介绍具体算法细节之前所需要介绍的基本定义、概念以及涉及到的算法模型。

定义 1. 道路网络(road network). 道路网络(简称路网) $G(V, E)$ 是一个有向图,其中 V 是结点集(即路口), E 是边集(即路段). 对于路段 $e \in E$, 记 $e.s \in V$ 为路段的起点, $e.d \in V$ 为路段的终点.

定义 2. 轨迹(trajjectory). 受限与路网空间运动的轨迹 $T = e_1 \rightarrow e_2 \rightarrow \dots \rightarrow e_k$ 由有序的边序列组成, 满足相邻边连续性质, 即 $\forall i$ 有 $e_i.d = e_{i+1}.s$.

定义 3. 问题定义. 给定路网 $G(V, E)$, 以及轨迹集合 \mathcal{T} . 对于待检测的轨迹 T , 设计异常检测算法鉴别 T 是否是异常.

定义 4. 最优轨迹. 对于给定的起始路段 e_s , e_d , 定义其最优轨迹 T^* 为

$$T^* = \underset{\{T | T \text{ is from } e_s \text{ to } e_d\}}{\operatorname{argmax}} P(T | e_s, e_d),$$

即在所有从 e_s 出发到 e_d 的可能路径中似然(likelihood)最高的轨迹.

定义 5. 异常度. 定义一条起始路段为 e_s , e_d 的轨迹的异常度为

$$\xi(T) = 1 - \frac{\log P(T^* | e_s, e_d)}{\log P(T | e_s, e_d)} \quad (1)$$

显然有 $\xi(T) \in [0, 1)$, 当轨迹为最优轨迹时(即 $T = T^*$)异常度取到最小值 0. 对概率加上对数是为了将概率的指数级变化映射到线性级变化.

定义 6. 异常轨迹的判断标准. 给定阈值 λ , 对于一条以 e_s, e_d 为起始路段的待测轨迹 T :

$$\text{如果有 } \xi(T) \begin{cases} \geq \lambda, & \text{则 } T \text{ 为异常轨迹} \\ < \lambda, & \text{则 } T \text{ 为正常轨迹} \end{cases},$$

对于一条轨迹 $T = e_1 \rightarrow e_2 \rightarrow \dots \rightarrow e_k$, 进行一阶马尔可夫假设的轨迹似然为

$$\begin{aligned} P(T | e_1, e_k) &= P(e_1 | e_1, e_k) \prod_{i=1}^{k-1} P(e_{i+1} | e_i, e_1, e_k) \\ &= 1 \cdot \prod_{i=1}^{k-1} P(e_{i+1} | e_i, e_k) \end{aligned} \quad (2)$$

需要注意, 在概率条件中省略了起点路段 e_1 而保留了终点路段 e_k . 原因是, 转移概率 $P(e_{i+1} | e_i)$ 为从路段 e_i 行驶到路段 e_{i+1} 的概率, 它受终点影响而并不受起点影响. 以图 2 为例, 驾驶员在路段 e_i , 如果目的地为 $e_D^{(1)}$, 则行驶到 $e_{i+1}^{(1)}$ 的概率更大; 若目的地为 $e_D^{(2)}$, 那么行驶到 $e_{i+1}^{(2)}$ 的概率更大. 而显然, 无论起点是在 $e_s^{(1)}$ 还是在 $e_s^{(2)}$, 并不会影响当前的决策. 下面将

介绍如何使用马尔可夫决策过程和反向增强学习算法对有目标条件限制的转移概率 $P(e_{i+1} | e_i, e_k)$ 进行建模.

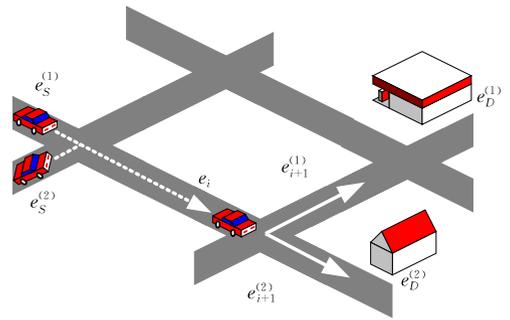


图 2 路段转移示意图

定义 7. 马尔可夫决策过程(Markov Decision Processes, MDP). MDP^① 是一个四元组 (S, A, γ, R) , 其中

(1) S 是状态(state)集合.

(2) A 是动作(action)集合, 定义 $a(s) \in S$ 为状态 s 经过动作 a 后转移到的状态.

(3) $\gamma \in [0, 1]$ 是一个衰减系数(discount factor).

(4) $R: S \mapsto \mathbb{R}$ 是奖励(reward)函数, $R(s)$ 表示状态 s 对应的奖励.

一个 MDP 的动态过程如下: 某个代理者(agent)的初始状态为 s_0 , 最初选择动作 a_0 执行, 随后 agent 转移到了下一个状态 s_1 , 故有 $a_0(s_0) = s_1$. 接着再执行动作 a_1 , 就转移到了 s_2 , 有 $a_1(s_1) = s_2$. 以此类推.

如图 3 所示, 我们得到了过程: $s_0 \xrightarrow{a_0} s_1 \xrightarrow{a_1} s_2 \rightarrow \dots$.

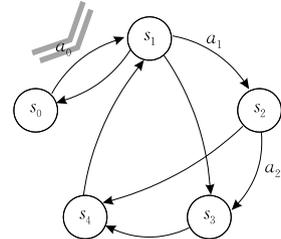


图 3 马尔可夫决策过程示意图

请注意, MDP 过程中的动作决策满足一阶马尔可夫性, 即只取决于当前状态, 与前面状态无关.

定义 8. 策略(policy). 记 π 为一个策略, π 是一个 $S \mapsto A$ 的函数, $\pi(s) = a$ 表示当 agent 的状态为 s 时, 策略 π 所给出的决策是执行动作 a .

定义 9. 值函数(value function). 给定一个策略 π , 奖励函数 R , 其对应的值函数为当前状态下的

① MDP 可以分为确定性和非确定性, 由于在驾驶行为决策中, 司机做出了决策后状态的转移确定的, 因此本文只讨论确定性的 MDP.

奖励以及未来根据该策略进行决策后得到的总奖励的期望,即

$$V^\pi(s_0 | R) = E[R(s_0) + \gamma R(s_1) + \gamma^2 R(s_2) + \dots | \pi] \\ = R(s_0) + \gamma R(s_1) + \gamma^2 R(s_2) + \dots,$$

其中未来的奖励会随时间乘以一个时间衰减项 γ . 本文讨论的是确定性 MDP, 因此给定初始状态 s_0 和策略 π 后, agent 采取的动作以及状态的转移都是确定的, 对应式子中的 s_0, s_1, s_2 等状态转移的序列是确定的, 因此期望记号可以去掉. 同时, 根据贝尔曼方程可得

$$V^\pi(s_0 | R) = R(s_0) + \gamma V^\pi(s_1).$$

在所有的策略 π 中, 使得系统中每个状态对应的值函数都能达到最大的策略则为最优策略 (optimal policy), 也就是说, 最优策略 π^* 满足:

$$\pi^* = \arg \max_{\pi} V^\pi(s), \quad \forall s \in S.$$

定义 10. Q-函数. 给定一个策略 π 和奖励函数 R , Q 函数是定义在 $S \times A$ 的函数, 定义为

$$Q^\pi(s, a | R) = Q^\pi(s \rightarrow s' | R) \\ = R(s) + \gamma V^\pi(s' | R) \quad (3)$$

为了方便描述, 用 $(s \rightarrow s')$ 来表示 (s, a) 二元组, 其中 $s' = a(s)$. Q 函数的值的意义为, 当前状态 s (不受限于策略 π) 通过采取某个动作 a 之后, 依照策略 π 能够得到的奖励期望.

定义 11. 增强学习 (reinforcement learning). 增强学习是智能系统从环境到行为的映射的学习, 目的是使奖励信号函数值最大化. 即给定一个 MDP, 增强学习的目标是得到最优策略 π^* , 使得 agent 在任何状态下, 根据 π^* 的行动得到的奖励 (值函数) 是最大的. 一般的做法是通过值迭代 (value iteration) 或策略迭代 (policy iteration) 来求得最优策略 π^* [17].

定义 12. 值迭代 (value iteration). 值迭代是一种在 MDP 中求得最优策略的算法, 具体算法步骤见算法 1.

算法 1. 值迭代.

输入: MDP $M(S, A, \gamma, R)$

输出: 最优策略 π^*

1. FOR $s \in S$ DO
2. $V(s) \leftarrow 0$;
3. WHILE 未收敛 DO
4. FOR $s \in S$ DO
5. $V(s) \leftarrow R(s) + \max_{a \in A} V(a(s))$;
6. FOR $s \in S$ DO

$$7. \quad \pi(s) \leftarrow \arg \max_{a \in A} V(a(s));$$

8. RETURN π ;

在给定 MDP 的情况下, 值迭代算法不断更新值函数 V (第 5 行), 直至值函数 V 收敛后利用最优策略的定义 (第 7 行) 求得每个状态 s 下的最优策略. 可以证明, V 一定能收敛^①.

定义 13. 策略迭代 (policy iteration). 策略迭代是另外一种在 MDP 中求得最优策略的算法, 具体算法步骤见算法 2.

算法 2. 策略迭代.

输入: MDP $M(S, A, \gamma, R)$

输出: 最优策略 π^*

1. FOR $s \in S$ DO
2. $\pi(s) \leftarrow \text{Random}(a \in A)$;
3. WHILE 未收敛 DO
4. $V \leftarrow V^\pi$;
5. FOR $s \in S$ DO
6. $\pi(s) \leftarrow \arg \max_{a \in A} V(a(s))$;
7. RETURN π ;

策略迭代首先对每个状态随机分配一个策略 (第 1、2 行), 之后每次迭代前, 先通过求解贝尔曼方程组, 根据当前求得的策略 π 来得到对应的值函数 V (第 4 行), 随后对于每个状态 s , 根据贪婪原则得到新一轮的策略 $\pi(s)$, 再进行下一轮的迭代直到策略 π 收敛为止. 可以证明, π 一定能收敛^[18].

需要注意的是, 值迭代和策略迭代都是解决 MDP 的常用方法, 并无特别优劣之分. 一般而言, 在小规模的 MDP 问题中, 策略迭代收敛速度快; 而在较大规模的 MDP 问题中, 由于策略迭代需要解线性方程组, 因此当状态空间非常大时, 解方程组的开销将会变得很大, 而此时值迭代方法更适合.

5 基于马尔可夫决策过程的转移概率建模

5.1 驾驶行为与转移概率

我们采用 MDP 对车辆驾驶行为进行建模. 本节主要介绍两者之间的关联性.

具体地, 路网中路段集合 G, E 对应状态空间, 驾驶员则对应 agent, 驾驶员在路口进行的决策 (如直行、左转、右转等) 对应动作 A . 车辆在路网上的行驶过程可以认为是状态之间的转移过程, 根据常识,

① Convergence of Value Iteration Algorithm. <http://www.math.tau.ac.il/~mansour/rl-course/scribe6/node4.html>, 1999, 12, 18

在不相邻的路段之间转移是不合法的,因此限制该动作的发生.表 1 总结了 MDP 中的元素与驾驶行为场景中的对应关系,其中关于奖励函数将在 5.2 节中介绍.

表 1 驾驶行为与 MDP 对应关系

MDP 元素	驾驶行为场景中对应的元素
状态集 S	路段 $e \in G, E$
动作集 A	路口决策, 决定转移至哪一个路段
奖励函数 R	路段潜在开销的负数

使用 MDP 对驾驶行为建模是合理的,原因是基于 MDP 的增强学习得到的最优策略具有未来前瞻性(最大化当前与未来得到的奖励),这与驾驶行为是一致的.因为驾驶过程中,驾驶员的决定会结合未来的考量.如驾驶员某时刻的决策是“右转上高架”,做出决策的原因是考虑到这样做后,之后的驾驶体验会较优(如时间较短、道路通畅、红绿灯较少等),这也就是最大化当前与未来得到的奖励的过程.可见,MDP 模型非常好地符合了该问题背景.

在最优决策的前提下,agent 采取的行为是确定的,而现实生活中,人们的驾驶决策是不确定的.直观上地,人们的决策概率与做了该决策后的预期奖励正相关,即与 Q 函数正相关.更具体地,如果决策 a_1 的 Q 函数的值与决策 a_2 的 Q 函数值非常接近,那么人们选择 a_1 或者 a_2 的概率应该也是非常接近的,因为不论做哪一种决策,带来的预期奖励都是差不多的.因此,在给定奖励函数 R 的情况下,我们对驾驶员在路段 e_i 的情况下做出“转移到 e_j ”的概率使用下式表示:

$$P(e_j | e_i, R) = \frac{\exp Q^*(e_i \rightarrow e_j | R)}{Z(e_i | R)} \quad (4)$$

为了满足概率定义,分母为归一化因子,即 $Z(e_i | R) = \sum_{e_n \in \text{adj}(e_i)} \exp Q^*(e_i \rightarrow e_n | R)$, $\text{adj}(e_i)$ 为与 e_i 邻接的路段集合.从式(4)中可以看出,驾驶员做的决策概率与决策对应的 Q 函数的值成指数正比.需要注意的是,通常可以对目标状态的奖励函数值进行一些特殊的设置(将会在 5.2 节中进行阐述),诱导最优策略朝向目标状态进行转移.在驾车情形下,奖励函数由驾驶的最终目的地决定,这样驾驶员的决策将会诱导状态向目标状态进行转移.因此随机变量 R 与目的地路段 e_D 在概率条件中是等同的,所以式(4)可以用于式(2)中的转移概率 $P(e_{i+1} | e_i, e_k)$ 的建模,下文将介绍如何计算奖励函数 R .

5.2 奖励函数与路段潜在开销

由前文可知,如果 MDP 的所有元素都已给出,那么最优策略即可求出,对应的 Q 函数也可以求出,随后也可以根据式(4)求得式(2)中的转移概率而最终得到整条轨迹的异常度.而实际上,我们并不能直接获得 MDP 中的奖励函数 R .本文采用的解决方法是利用机器学习的思想,根据历史数据的分布来自动地学习 R ,使得模型在满足当前奖励函数下,历史数据的似然最大.

从前文分析可以看出 MDP 中决定 agent 的策略的关键因素是奖励函数 R 和目标状态 e_D 的设置.由第 4 节对图 2 中的例子分析可知,在车辆驾驶行为建模的场景下,不同的驾驶目的地在同一路段上对应着不同的路口决策,也即对应了不同的奖励函数.一种简单的方法是,可以直接将历史数据根据不同的目的地分成若干个集合,对每个集合内的数据训练一个独立的奖励函数 R_D .而这种方法有个严重的问题是,由于实际中状态空间(路段数量)很大,将历史数据根据目标状态进行划分后,每个状态对应的数据集则会变得非常稀疏,这样通过训练得到的奖励函数将会非常不准确.

但是同时,路段的等级、长度、车道数和限速等静态的外部特征往往和目的地无关,驾驶员在进行决策时这些特征实际上起着重要的作用.考虑到这些因素,我们定义每条路段 e 存在着一定的潜在开销(latent cost) $C(e) > 0$,该潜在开销对于每条路段而言是一个固定的值,与目标路段无关.对于在同一个路网中的所有不同目标的 MDP,除了目标路段以外都共享同样的路段潜在开销.所以,对于一个目标路段为 e_D 的 MDP,我们通过以下规则构造对应的奖励函数 R

$$R(e) = \begin{cases} -C(e), & \text{如果 } e \neq e_D \\ 0, & \text{如果 } e = e_D \end{cases}$$

由于驾驶员的目的是最小化开销,也就是说,最大化“负”开销,根据增强学习中的最优策略的定义,我们将路段的潜在开销的负值作为路段状态在 MDP 中的奖励函数.除了目标路段以外的状态,其余对应的奖励都是小于 0 的,而将目标路段设置为 0 的原因是,将决策引向目标状态,这是因为如果不朝着目标状态做决策,那么不管如何进行转移,奖励的数值都是负的,只会越转越少.因为对于目标状态 e_D 而言,最优策略 $\pi^*(e_D)$ 应该为留在原地才能使得累计奖励最大化,这样实现了目的地的“吸引”特性,

使得值函数不会无穷地计算下去。

如图 4 所示,虚线平行四边形框内的为路网,每条路段具有一个不变的潜在开销 $C(e)$ 。对于不同目标的 MDP,我们认为这些 MDP 共享除了各自对应

的目标路段以外的奖励函数,即路段潜在开销的负值。将目标路段的奖励置为 0 则体现了 MDP 的不同目标的特性。最终,具有不同目标的驾驶员们根据对应的 MDP 产生了现实生活中的轨迹。

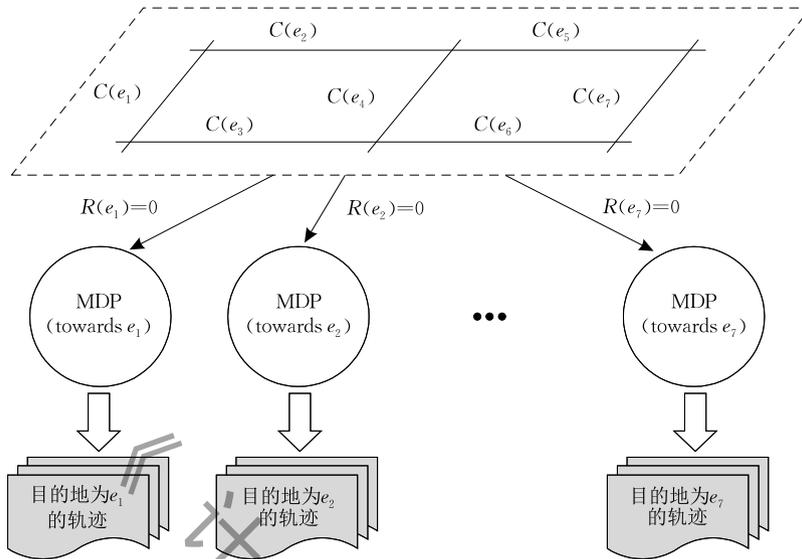


图 4 共享路段潜在开销的驾驶行为建模示意图

5.3 模型学习

本节将介绍如何学习找到路段的奖励分配方式(即 MDP 模型的参数 R),根据这个路段的奖励分配方式,使得在该方式下生成的估计的概率最贴近真实生活中的轨迹,这样我们就能够利用 MDP 模型来求得以任意路段为终点的任意两个相邻路段之间的转移概率了。我们采用 BIRL 算法来完成这一任务^[19],该算法是一种无监督的学习算法,根据一系列由 MDP 产生的轨迹来学习出 MDP 中的奖励函数 R 。具体地,对于一个目标路段为 e_i 的 MDP $M=(S, A, \gamma)$,其奖励函数 R_i 未知,给出根据该 MDP 产生的若干轨迹集 \mathcal{T}_i , R_i 的后验分布记作 $P(R_i | \mathcal{T}_i)$ 。根据贝叶斯定律,并将式(4)代入有

$$\begin{aligned} P(R_i | \mathcal{T}_i) &\propto P(\mathcal{T}_i | R_i) P(R_i) \\ &= P(R_i) \prod_{T \in \mathcal{T}_i} P(T | R_i) \\ &= P(R_i) \prod_{T \in \mathcal{T}_i} \prod_{e_j \in T} \frac{\exp Q^{\pi^*}(e_j \rightarrow e_{j+1} | R_i)}{Z(e_j | R_i)} \end{aligned} \quad (5)$$

在无先验信息的情况下 $P(R_i)$ 设置为均匀分布。

记真实的奖励函数为 R_i ,算法估计的奖励函数为 \tilde{R}_i ,为了最小化平方误差 $\|R_i - \tilde{R}_i\|_2^2$,等价于从 R_i 的后验分布 $P(R_i | \mathcal{T}_i)$ 中返回均值^[20]。

由于总数据集是由不同的 e_i 为目标路段的 MDP 产生的,而每个 R_i 都共享了除了目标路段以外相同的路段潜在开销,因此可以联立所有 R_i 的

验分布得到 C 的后验分布

$$P(C | \mathcal{T}) = \prod_i P(R_i | \mathcal{T}_i) \quad (6)$$

所以实际上我们想要求得的是路段的潜在开销 C ,同样地,使用 C 的后验分布 $P(C | \mathcal{T})$ 的均值 \tilde{C} 作为 C 的估计同样可以满足最小化平方误差 $\|C - \tilde{C}\|_2^2$ 。

直接计算后验分布比较困难,文献[19]提出了一种基于马尔可夫链蒙特卡罗(Markov Chain Monte-Carlo, MCMC)采样的方法。MCMC 方法可以对一个分布进行采样,当马尔可夫链收敛时,采样出的数据的均值则是分布的均值。算法流程在算法 3 中有具体描述。首先使用路段的长度作为路段的开销 $C(e)$ 的初始值且使用开销的负数作为奖励函数的初始值(第 2 行)。然后每次采样都从当前 C 的 δ -邻域中进行均匀地采样。即对于第 $t+1$ 次的采样,采样器从 $[C(t) - \delta, C(t) + \delta]$ 中服从均匀分布地采一次样,得到 C' (第 5 行)。然后根据式(5)计算轨迹集 \mathcal{T} 中的每条轨迹的似然,并根据式(6)得到新采样到的路段潜在开销函数 C' 的后验概率 $P(C' | \mathcal{T})$ (第 6~10 行),服从 $\min\left\{1, \frac{P(C' | \mathcal{T})}{P(C^{(t)} | \mathcal{T})}\right\}$ 的概率接受 C' 成为 $C^{(t+1)}$ (第 11 行),并进行下一轮的采样。当马尔可夫链达到收敛后,采样器采得的样本(包含接受步骤)将服从目标分布 $P(C | \mathcal{T})$,此时对最后若干个采样的 C 样本求其均值(第 12 行)即可作为目标分布

$P(C|T)$ 的均值的一个估计. 最后, 将开销函数取负数即得到最终的奖励函数 R .

采样过程中马尔可夫链的收敛是很难控制和检测的, 因此我们将在实验过程中针对不同的步长 δ 进行实验, 找出收敛最快的 δ .

算法 3. BIRL 算法.

输入: MDP $M(S, A, \gamma)$, 轨迹集合

输出: 奖励函数 R

1. FOR e in G, E DO
2. $C(e) \leftarrow e.length, R(e) \leftarrow -C(e)$;
3. $\pi \leftarrow value_iteration(M, R)$;
4. WHILE 未收敛 DO
5. $C' \sim Uniform[C - \delta, C + \delta]$;
6. FOR T in \mathcal{T} DO
7. $R'(e_D) \leftarrow 0$;
8. $\pi \leftarrow value_iteration(M, R')$;
9. 根据式(5)计算 $P(T|R')$;
10. 据式(6)计算 $P(C'|T)$;
11. 服从概率 $\min\left\{1, \frac{P(C'|T)}{P(C|T)}\right\}$ 接受 C' 成为 C ;
12. $\tilde{C} \leftarrow$ 循环过程中最后得到的若干个 C 的均值;
13. $\tilde{R} \leftarrow -\tilde{C}$;
14. RETURN \tilde{R} ;

稀疏性问题的解决: 由于模型假定了在路网运动的车辆共享统一的路段潜在开销, 因此, 即使 e_s, e_D 距离很远导致历史轨迹稀疏的情况下, 我们依然可以通过其他的密集轨迹学习到每条路段的潜在开销 $C(e)$ (或 $R(e)$), 对稀疏轨迹也能正确地计算出它的似然, 从而进一步地判断其是否异常.

5.4 在线检测过程

求得每条路段的奖励 $R(e)$ 后, 对一条终点为 e_D 的待检测轨迹 T , 算法 4 给出了检测异常的流程.

算法 4. 异常检测在线算法.

输入: 轨迹 T , MDP $M(S, A, \gamma)$, 开销函数 C , 终点路段 e_D

输出: T 是否为异常轨迹

1. FOR $e_i \neq e_D$ DO
2. $R(e_i) \leftarrow -C(e_i)$;
3. $R(e_D) \leftarrow 0$;
4. $\pi \leftarrow value_iteration(M, R)$ 或 $policy_iteration(M, R)$;
5. 计算 $P(T^* | e_s, e_D)$;
6. $P(T|R) \leftarrow 1$;
7. FOR e_i in T DO
8. 根据式(3)计算 $Q^\pi(e_i \rightarrow e_{i-1} | R)$;
9. 根据式(4)计算 $P(e_i | e_{i-1}, R)$;
10. $P(T|R) \leftarrow P(T|R) \times P(e_i | e_{i-1}, R)$;
11. 根据式(1)计算 $\xi(T)$;
12. IF $\xi(T) > \lambda$ THEN

13. RETURN true;

14. ELSE

15. RETURN false;

算法的输入是待检测的轨迹 T 和终点路段 e_D 以及对应的 MDP 和通过历史数据训练得到的路段潜在开销函数 C ; 输出是一个布尔值, 标识检测轨迹 T 是否为异常. 首先将所有的路段的奖励函数 $R(e_i)$ 赋值成训练得到的路段负开销—— $C(e_i)$, 而单独将目标路段设置为 0, 从而构建目标状态为目标路段的 MDP (第 1~3 行). 注意到 T 的似然的计算需要计算 Q 函数, 也就是计算 R 对应的值函数, 因此第 4 行需要进行值迭代或策略迭代求得值函数. 接着使用动态规划算法可求得最优轨迹的概率值 (第 5 行). 第 6~10 行计算待检测轨迹 T 对应的概率值. 第 11 行通过异常度定义公式计算该轨迹的异常度. 第 12~15 行, 根据用户定义的异常度阈值 λ 标识该轨迹异常与否.

注意到, 由于每条轨迹需要将对应的终点奖励值设置成 0 后单独做一次值迭代或策略迭代, 如果轨迹条数多的时候, 需要每条做轨迹条数次值迭代或策略迭代, 这显然会使得时间开销变得非常大. 优化方法是首先离线地将每条路段作为终点做好值迭代或策略迭代, 并计算每个状态下每个动作对应的值函数与 Q 函数, 并求出状态之间的转移概率并保存下来, 进而优化线上判断的过程. 或者可以采取“按需计算并缓存”的思路. 即每次需要对某个路段作为终点计算对应的 MDP 的值迭代或者策略迭代时, 首先去事先缓存的区域查看是否之前已有缓存, 如果有则直接使用之前计算过的结果 (因为只要两条轨迹的终点是一致的, 那么他们对应的 MDP 的奖励函数也是一致的); 如果事先没有计算过且缓存下来, 那么在进行一次值迭代或策略迭代算法计算后将结果缓存下来供之后的数据使用. 类似的, 第 5 行最优路径 T^* 的计算也可以利用线下计算或按需计算并缓存的策略进行加速.

6 实验结果与分析

6.1 实验设置

我们使用了葡萄牙波尔多市的公开出租车轨迹数据集^①, 包含从 2013 年 7 月到 2014 年 6 月共 442

① ECML/PKDD 15: Taxi Trajectory Prediction (I). <https://www.kaggle.com/c/pkdd-15-predict-taxi-service-trajectory-i>, 2015, 04, 20

辆出租车, 约 3 GB 的轨迹数据, 经过数据清洗与切割后共选取了 20 万条轨迹作为实验数据集. 轨迹的平均采样率为 15 s. 由于原始的数据是欧几里得空间上的点序列(即经纬度), 我们首先采用地图匹配算法^[16], 将其映射到路网从而转化成了边序列. 根据引言部分所述, 目前 iBOAT^[4] 与 MEX^[9] 是现有的即支持在线地检测异常又是适合于车辆轨迹进行的研究工作. 因此我们将 iBOAT 与 MEX 算法作为对比方法, 由于 MEX 算法提出了两套不同的异常判别机制, 分别是 MEX-PN 与 MEX-TN, 因此我们分别对这两种机制都进行了实现.

6.2 异常度实验分析

首先, 我们针对该异常值的准确程度进行实验分析. 本实验从轨迹数据中抽取若干对起始路段分别为 e_s, e_d , 且每对起始路段相同的轨迹集合满足:

- (1) 不少于 100 条;
- (2) 至少有 4 种不同的轨迹模式.

为了方便描述, 本文将抽取的起始路段记作 (e_s, e_d) , 并将给定起始路段 (e_s, e_d) 之间的轨迹记为 T_1, T_2, \dots, T_k (没有重复), 每种轨迹在数据集中出现的频数记为 F_1, F_2, \dots, F_k , 不妨规定 $F_1 \leq F_2 \leq \dots \leq F_k$. 这里假定在数据密集的情况下, 一种轨迹经过的次数与轨迹的异常程度是正相关的. 一个好的轨迹异常检测算法对所有以 (e_s, e_d) 为起始路段的不同轨迹的异常度打分得到的降序排序应该与实际中按照轨迹的频数的升序排序是一致的(即异常度越高, 轨迹频数应该越低). 因此, 我们使用信息检索中常用的 $NDCG$ (Normalized DCG) 标准对排序的性能进行评测^[21]. $NDCG$ 是一种比较排序效果的度量标准. 它的特点是, 对待测算法得到的排序靠前的对象加上较大的权重, 而减少靠后的对象权重. 对于异常检测问题, $NDCG$ 指标对排序靠前的对象更敏感, 例如, 把实际中频繁的正常轨迹误判为异常, 体现在把排序靠后的轨迹置前, 这样会带来的惩罚将比把异常轨迹排序的位置置后更高. 原因在于, 轨迹异常检测的目的在大部分情况下是去除异常轨迹, 将正常轨迹误判为异常通常会造成长量频繁正常的轨迹被删除的严重后果, 因此必须要避免这样的情况发生.

具体地, 令一个算法对给定 (e_s, e_d) 下的不同轨迹根据其计算得到的异常度进行降序排序后得到的轨迹顺序为 $\hat{T}_1, \hat{T}_2, \dots, \hat{T}_k$, 记 $Rank(\hat{T}_i)$ 为轨迹 \hat{T}_i 在真实排序下的排名, \hat{T}_i 对应的实际的频数记

$$\text{作 } \hat{F}_i = F_{Rank(\hat{T}_i)}. \text{ 则有 } DCG = \sum_{i=1}^k \frac{2^{rel(i)} - 1}{\log_2(i+1)}, \text{ 其中}$$

$$rel(i) = \frac{\log \sum_{j=1}^k \hat{F}_j - \log \hat{F}_i}{\log \sum_{j=1}^k \hat{F}_j} \in (0, 1] \text{ 是相关度的计算函}$$

数, 轨迹的真实排名越靠后(即频数越高), 则对应的相关度越小, 反之相关度越接近 1. DCG 中的分母项 $\log_2(i+1)$ 则是对排名进行了权重衰减, 对靠后的排名, 降低其权值, 而增加排名靠前的重要程度. 记 $iDCG$ 为真实排序的 DCG 值, 则

$$NDCG = \frac{DCG}{iDCG} \in [0, 1],$$

其中当算法排序与真实排序一样时, $NDCG$ 值为 1. 算法与真实排序差异越大, $NDCG$ 值则越低.

在实验数据中, 我们根据之前的选取规则, 共选出了 129 对起始路段对, 表 2 展示了四种算法在这些测试样本下的平均 $NDCG$ 值和每条测试轨迹的平均的运行时间. 运行时间已扣除算法的预处理时间, 只计算在线执行的时间.

表 2 算法平均性能结果

	平均 $NDCG$	平均时间/ms
iBOAT	0.951	474.931
MEX-PN	0.935	31.878
MEX-TN	0.943	7895.880
Ours	0.993	0.012

图 5 中每个点的横坐标表示某一个起始路段 (e_s, e_d) 里的不重复的轨迹模式数量, 纵坐标表示这些轨迹通过算法排序后得到的 $NDCG$ 值.

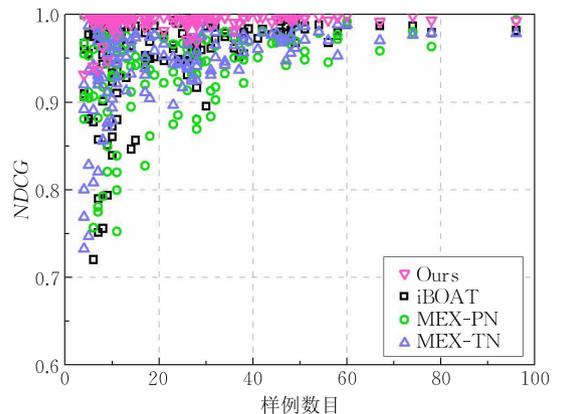


图 5 密集数据集下的 $NDCG$ 结果

可以看出, 我们的 $NDCG$ 指数在大部分情况下非常接近 1, 即我们使用轨迹似然对轨迹的异常度进行打分得到的排序与真实的排序几乎一样, 也就是说无论异常度的阈值设为多少, 我们的算法都能

很好地鉴别异常轨迹,而对比的三个算法它们的 $NDCG$ 值则有较大的差距,主要原因是, $iBOAT$ 是基于网格的方法,有些异常轨迹在局部产生了异常,而这些轨迹却都落在一个或相邻的网格内, $iBOAT$ 是无法发现这类异常的;而 $MEX-PN/TN$ 两个算法由于涉及到过多参数(如近邻范围、窗口大小、异常点判定数量阈值等等),而这些参数与实际数据密度与分布密切相关,对不同的地方使用同样的参数必然会使得一些场景的效果变差.此外,从图中点的分布趋势可以看出,当 (e_s, e_D) 的轨迹模式变多的情况下,差距变小了.其主要原因是,当给定起始路段 (e_s, e_D) 的需要排序的轨迹模式很多的情况下,由于 $NDCG$ 对靠后的排名进行权重衰减,因此即使出现对靠后的排名排序错误的情况,也不会对 $NDCG$ 有很大的影响,而从数据分布来看,也可以看出,大部分数据的轨迹模式数量还是位于 $4 \sim 40$ 内,因此我们的算法在实际中的优势还是很大的.

在算法性能方面,我们的方法由于不需要扫描数据集中的其他数据,因此效率非常高.而其他三种算法,由于都需要对原始数据集进行扫描,因此效率相比我们要差很多.图 6 展示了每个测试样本下对比的三个算法的时间开销与我们算法的时间开销的比值,为了便于展示,我们对统计结果根据我们的算法的绝对时间开销进行了升序排序.

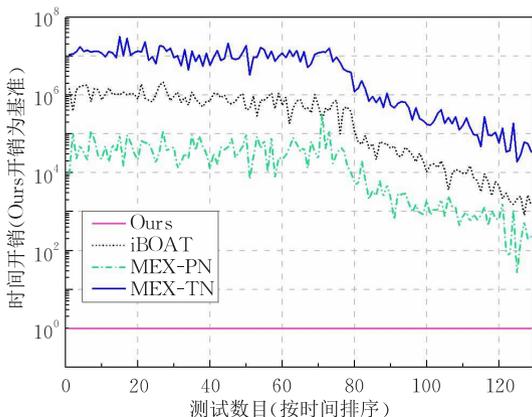


图 6 密集数据集下的时间开销与本文方法的比值

可以看出 $MEX-PN$ 在近邻查询中有优化剪枝策略,一旦搜索到满足给定阈值的点的数量则停止搜索,在三个对比算法中最快,但仍比我们的算法慢了 $2 \sim 4$ 个数量级; $iBOAT$ 由于一开始的近邻查询必须需要完全查询,因此比 $MEX-PN$ 还要慢了 1 个数量级;而 $MEX-TN$ 也与 $iBOAT$ 一样,需要进行完全查询,但是却不像 $iBOAT$ 会自动缩小检测窗口,因此更是比 $iBOAT$ 慢了 1 个数量级.三个对比

算法在我们算法开销的时间增长的情况下,与我们的差距有所缩小,主要原因是,我们的时间开销是正比于轨迹长度的,也就是说横坐标越靠后轨迹长度越长,而当轨迹很长的情况下,三个对比算法都有对应的剪枝策略起到了作用,因此与我们的差距有所缩小.需要注意的是,我们的模型虽然有训练部分,但是该模块只需要针对历史数据离线训练一次即可,且在在线部分利用缓存优化或按需计算缓存思想加速可以保证在在线检测部分具有很快的速度.

6.3 数据稀疏情况下的实验分析

为了观察在长距离历史数据稀疏的情况下异常检测算法是否依旧能具有鲁棒性,我们通过一定规则构造了一些稀疏情况下的测试集.需要注意的是,由于人工标注具有主观性(尤其实验区域在国外),此外能标注的数据量也很有限,人工标注不宜实施.相对地,我们设置了一定的规则构造了稀疏且有标记的测试样本.首先,从历史数据中抽取满足以下条件的 (e_s, e_D) 对以及对应的轨迹:

- (1) 轨迹不少于 100 条;
- (2) 最短的路径至少经过 8 个路段.

记 (e_s, e_D) 中所有轨迹的频数之和为 $F_{(e_s, e_D)}$,我们将轨迹中频数低于 $3\% \times F_{(e_s, e_D)}$ 的轨迹标记为异常轨迹,将频数高于 $3\% \times F_{(e_s, e_D)}$ 的轨迹标记为正常轨迹.同时,对这些正常轨迹稀疏化,每种只保留 $3\% \times F_{(e_s, e_D)}$ 条.

至此,我们得到了满足(1)长距离(2)有标记(3)稀疏的轨迹数据集了.将构造后的数据集作为异常轨迹检测算法的输入,进行实验.

首先,在稀疏化实验数据中,我们根据异常度实验分析实验的选取规则,同样分析了两个算法在这些测试样本下的 $NDCG$ 值的分布情况,如图 7 所示.

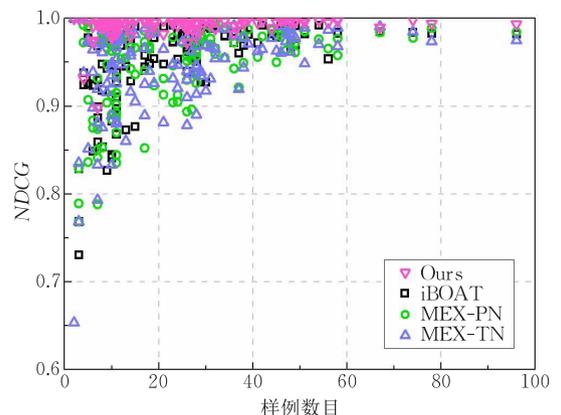


图 7 稀疏数据集下的 $NDCG$ 结果

需要注意的是,本实验中 $NDCG$ 的评测结果没有受到特别大的影响. 原因是, $NDCG$ 的评测对象为给定起始路段后的所有不同的路线的排序情况,而在本实验中,给定起始路段,正常轨迹的路线种类是远小于异常轨迹的路线种类的,即使将正常轨迹错误地排在异常轨迹之间,因此对总 $NDCG$ 的影响并不显著. 此外,在稀疏的情况下,我们更需要知道的是正常轨迹是否会被错误地判断成了异常轨迹,因此综合考虑,我们认为用于评判分类标准的 ROC 曲线更适合用来描述稀疏情况下的评测标准.

我们根据表 3 统计对应的数量,然后以 x 轴为 $FPR = \frac{FP}{TN+FP}$, y 轴为 $TPR = \frac{TP}{TP+FN}$, 根据不同的预测的阈值 λ 绘制 ROC (Receiver Operating Characteristic) 曲线^[22]. ROC 曲线能够体现在不同任务下的泛化性能的好坏,同时也能客观地应对正反例数量不均的情况. 一个理想的分类器的 ROC 曲线应为左上角,一个随机分类器的 ROC 曲线为对角线. 若一个分类器的 ROC 曲线被另一个分类器的曲线完全“包住”,则可断言后者的性能优于前者.

表 3 分类结果混淆矩阵

	预测为异常	预测为正常
标记为异常	TP	FN
标记为正常	FP	TN

根据规则,我们从数据中构造了 6165 条轨迹用于测试,其中 2215 条标记为正常轨迹,3950 条标记为异常轨迹. 由于正常轨迹都被减少到了总数量的 3%,因此正常轨迹的数量略少于异常轨迹.

图 8 为所有算法在稀疏情况下对应的 ROC 曲线.

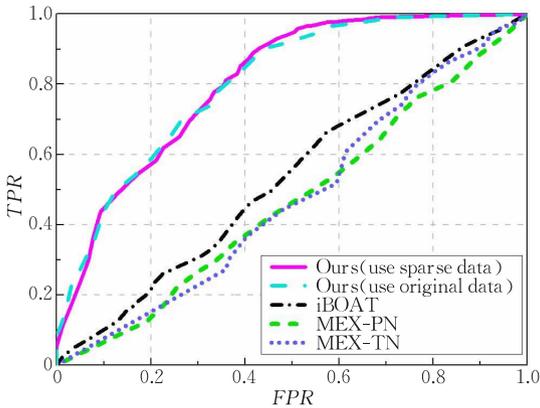


图 8 稀疏数据集下的 ROC 曲线

可以看出,我们的方法是大幅度优于对比的三种. 原因在于这三种算法都是基于统计的方法,根据轨迹数据集中与待测数据相近的轨迹或轨迹点数量进行判断. 而一旦长距离下数据稀疏后,历史数据

集中完全相同的轨迹数量非常少,根据这些算法的判断规则,这些轨迹都会被错判成异常轨迹,因而效果急剧变差,退化到了接近随机判断的结果. 而我们的方法,虽然轨迹变少了,但是通过使得每条路段共享同一个潜在开销后,利用其它密集的数据进行训练后,依然能够学习出路段的潜在开销. 即使历史数据中不存在与测试的正常轨迹一样的轨迹,依然可以正确地计算出它的似然.

更进一步地,图中长虚线对应的是我们使用未稀疏化的数据集训练得到的结果. 通过对比使用稀疏化后的数据和稀疏化前的数据的结果,可以发现即使将部分长距离的频繁轨迹稀疏化后,我们的模型依然能够学习出稀疏化前的效果,这归功于我们模型中提出的开销(奖励)函数的共享机理. 即使当长距离下历史轨迹较少,模型可以通过数据量较大的短距离下的轨迹分布情况,学习出路段的潜在开销,并将其共享到长距离的数据稀疏情况下,这也进一步证明了模型的鲁棒性.

6.4 步长选取实验分析

回想 5.3 节模型学习中, BIRL 算法在 MCMC 采样过程中,每次会对当前的路段潜在开销函数 C 的 δ -邻域 $[C-\delta, C+\delta]$ 内进行均匀采样. 很明显, δ 将会直接影响模型学习的收敛速度与效果.

对于 BIRL 算法中的步长 δ , 我们对该参数的选择进行了实验. 我们通过在模型学习过程中记录根据式(6)计算路段潜在开销 C 相对于数据集中所有轨迹的后验分布,即 MCMC 采样算法所需要模拟的目标分布 $P(C|T)$, 来观察不同的 δ 下模型的收敛速度. 由于概率数据过于小,我们将其对数化后对结果进行观察. 我们分别将 δ 设置为了 0.1、0.2、0.4 和 0.8, 结果展示在图 9 中,横坐标为 MCMC 采样的次数.

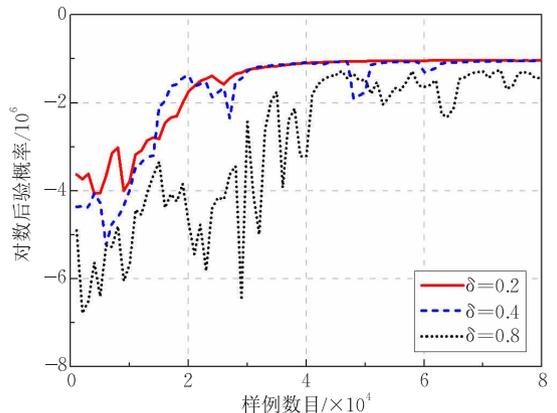


图 9 训练步长 δ 与算法收敛关系实验结果图

从图 9 中可以看出,当 δ 设置为 0.1~0.4 的时候,算法收敛效果较好,且 δ 设置为 0.2 与 0.4 的马尔可夫链收敛速度要略快于 0.1 的情况,主要原因是最终的 C 离初始值较远,因此 δ 大的情况下,每一步调整的步伐变大,也更容易达到最优解。同时地,随着 δ 的增大,每次从 C 的 δ -邻域 $[C-\delta, C+\delta]$ 中采样后的 C' 被接受的概率将会变低,且接近最优值的附近时,由于步长变长的原因,更加进一步地接近最优值的难度也加大了,因此 δ 设置为 0.2 与 0.4 的时候不如 $\delta=0.1$ 的曲线那么平缓稳定。当 δ 继续变大,设置成 0.8 时可以看出波动变大,因为每次采样得到的新的 C 可能接近最优值较远,算法收敛明显变困难。事实上我们也做了 1.6~6.4 的收敛实验,发现当 $\delta>1.6$ 时,算法已无法收敛。由于如果同时放上 1.6~6.4 的结果会对的结果展示产生影响,故未在实验图中展示。

6.5 训练时间实验分析

本实验将分析异常车辆轨迹检测系统的离线训练的时间开销与数据规模的关系。本实验的运行环境为 Intel i7-4790 3.6 GHz 的 CPU, 32 GB DDR3 内存, Windows 7 系统;使用 C++ 代码在 Visual Studio 2015 下编译。

图 10 展示了 BIRL 算法每轮 MCMC 采样需要的时间开销与数据规模的关系,我们通过变化训练数据的数量(从 5 万条到 35 万条)来观察算法时间与数据规模的变化关系。可以看出,随着数据规模的增长, BIRL 算法每轮采样以及计算后验概率所需的时间近似线性增长,这与预想是一致的。因为 BIRL 算法每轮采样需要计算数据集中所有数据的后验概率,因此时间开销正比于数据的规模。以 2×10^5 条轨迹(数据集中约 3 个月的跨度)为例,由图 10 可知, BIRL 算法的 MCMC 采样在约 3×10^4 轮采样后

开始收敛,因此约需要 $3 \times 10^4 \times 2 \div 3600 = 16.67$ h 在一台桌面 PC 上即可完成训练,这相对于 3 个月的轨迹数据时间跨度而言是可接受的。

6.6 样例分析

为了更好地展示我们的方法的有效性,我们选取了一个样例进行了分析。图 11 为给定起始路段下(起点路段标记为 e_s ,终止路段标记为 e_D)数据集中的所有轨迹。

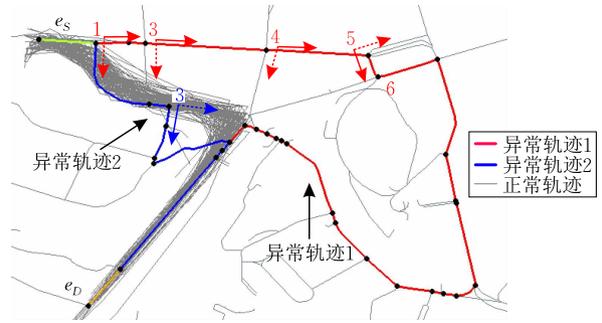


图 11 异常检测样例

为了说明正常轨迹的频繁特征,我们并未对匹配后的轨迹进行可视化(匹配后的轨迹由于都是边序列,所以可视化后重叠在了一起不容易观察),而是对正常轨迹使用原始采样数据进行可视化(即进行地图匹配前,欧几里得空间中的 GPS 采样点);而为了研究异常轨迹的路段转移过程,我们对异常轨迹使用匹配到路网空间中的路段进行可视化。其中明显可以看出两条异常轨迹,图中分别标识为异常轨迹 1 和异常轨迹 2。

图 12 显示了这两条异常轨迹在每个路口采取的行为对应的转移概率(已对数化)。可以看出,对于异常轨迹 1,在第一个路口处,绝大部分的历史轨迹都选择了向下行驶,而这条轨迹则选择了右行,从图 12 中则体现为其第 1 个路口的转移概率很低。在第 3 个路口时,向下走能够再次返回正常的路线,而

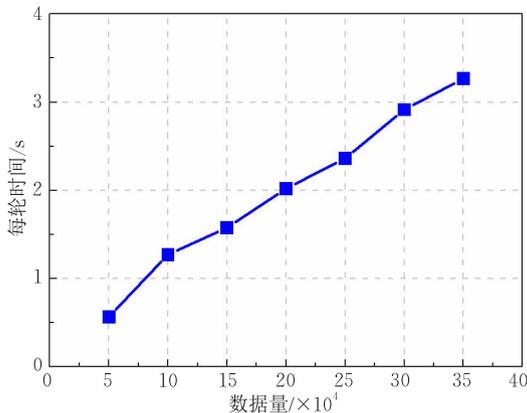


图 10 BIRL 算法每轮采样时间开销与数据规模结果图

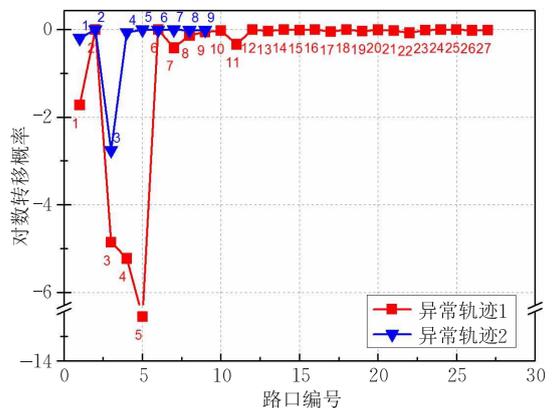


图 12 样例对应的两条异常轨迹的对数转移概率

其依然向右直行,对应的转移概率也很低. 同样地,对于第 4 个路口,向下转弯更为合理,而其依然向右直行,其对应的转移概率也很低. 第 5 个路口也同理. 在之后的状态下(第 6 个路口),异常轨迹 1 的后半部分应该和合理的行驶路线是一样的,对应的对数转移概率也都稳定在 0 附近. 对于异常轨迹 2,在第 3 个路口,其行为(向下)与大部分轨迹(向右)产生了偏差,对应的转移概率也立刻变低.

从此例可以看出,我们使用 MDP 对驾驶行为的概率建模是非常合理的,使用 BIRL 对路段潜在开销的学习也是很准确的. 此外,我们的方法不仅能够检测出异常轨迹,还能准确定位到所有异常发生的位置.

7 结束语

本文从驾驶行为建模的角度出发,研究异常轨迹,提出一种在路网空间上基于无监督学习解决在线车辆轨迹的异常检测算法. 算法利用马尔可夫决策过程以及反向增强学习的方法,学习出路段的潜在开销,解决了数据稀疏的问题. 并结合真实的出租车轨迹数据,与已有的方法 iBOAT 对比. 我们的方法在一般情况下表现出更高的准确性以及在数据稀疏情况下表现出高效性和鲁棒性. 同时通过样例研究进一步地展示了本文提出的算法的有效性与优越性,为异常轨迹检测工作提供了新的思路.

在接下来的工作中,我们将考虑如何利用时间维度的信息使得模型能够适用于动态变化的道路网络状况,设计一套具有动态增量式的异常检测模型.

参 考 文 献

- [1] Song Ying, Li Qing-Quan, Zheng Nian-Bo. Distributed vehicle monitor information service platform based on LBS. *Computer Engineering*, 2007, 33(6): 242-244(in Chinese)
(宋莺, 李清泉, 郑年波. 基于 LBS 的分布式车辆监控信息服务平台. *计算机工程*, 2007, 33(6): 242-244)
- [2] Zheng Yu. Methodologies for cross-domain data fusion: An overview. *IEEE Transactions on Big Data*, 2015, 1(1): 16-34
- [3] Zhang Da-Qing, Li Nan, Zhou Zhi-Hua, et al. iBAT: Detecting anomalous taxi trajectories from GPS traces//*Proceedings of the IEEE 13th International Conference on Ubiquitous Computing*. Beijing, China, 2011: 99-108
- [4] Chen Chao, Zhang Da-Qing, Castro P S, et al. iBOAT: Isolation-based online anomalous trajectory detection. *IEEE International on Intelligent Transportation Systems*, 2013, 14(2): 806-818
- [5] Li Yan, Li Hao, Qian Xiao-Lu, et al. A review and analysis of outlier detection algorithms. *Computer Engineering*, 2002, 28(6): 5-6(in Chinese)
(李炎, 李皓, 钱肖鲁等. 异常检测算法分析. *计算机工程*, 2002, 28(6): 5-6)
- [6] Knorr E M, Ng R T, Tucakov V. Distance-based outliers: Algorithms and applications. *The International Journal on Very Large Data Bases*, 2000, 8(3-4): 237-253
- [7] Ge Yong, Xiong Hui, Liu Chuan-Ren, et al. A taxi driving fraud detection system//*Proceedings of the IEEE 11th International Conference on Data Mining*. Vancouver, Canada, 2011: 181-190
- [8] Lee Jae-Gil, Han Jia-Wei, Li Xiao-Lei. Trajectory outlier detection: A partition-and-detect framework//*Proceedings of the IEEE 24th International Conference on Data Engineering*. Cancun, Mexico, 2008: 140-149
- [9] Yu Yan-Wei, Cao Lei, Wang Qin. Detecting moving object outliers in massive-scale trajectory streams//*Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, USA, 2014: 422-431
- [10] Li Xiao-Lei, Li Zhen-Hui, Han Jia-Wei, et al. Temporal outlier detection in vehicle traffic data//*Proceedings of the IEEE 25th International Conference Data Engineering*. Shanghai, China, 2009: 1319-1322
- [11] Pan Bei, Zheng Yu, Wilkie D, et al. Crowd sensing of traffic anomalies based on human mobility and social media//*Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. Orlando, USA, 2013: 344-353
- [12] Lei Po-Ruey. A framework for anomaly detection in maritime trajectory behavior. *Knowledge and Information Systems*, 2016, 47(1): 189-214
- [13] Li Xiao-Lei, Han Jia-Wei, Kim S, et al. ROAM: Rule-and motif-based anomaly detection in massive moving object data sets//*Proceedings of the 7th SIAM International Conference on Data Mining*. Minnesota, USA, 2007: 273-284
- [14] Liao Zi-Cheng, Yu Yi-Zhou, Chen Bao-Quan. Anomaly detection in GPS data based on visual analytics//*Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*. Salt Lake City, USA, 2010: 51-58
- [15] Sillito R R, Fisher R B. Semi-supervised learning for anomalous trajectory detection//*Proceedings of the British Machine Vision Conference*. Leeds, UK, 2008: 1-10
- [16] Song Ren-Chu, Lu Wei, Sun Wei-Wei, et al. Quick map matching using multi-core CPUs//*Proceedings of the 20th International Conference on Advances in Geographic Information Systems*. Redondo Beach, USA, 2012: 605-608

- [17] Sutton R S, Barto A G. Reinforcement Learning: An Introduction. London, UK; MIT Press, 1998
- [18] Santos M S, Rust J. Convergence properties of policy iteration. *SIAM Journal on Control and Optimization*, 2004, 42(6): 2094-2115
- [19] Ramachandran D, Amir E. Bayesian inverse reinforcement learning//Proceedings of the 20th International Joint Conference on Artificial Intelligence. Hyderabad, India, 2007: 2586-2591
- [20] Berger J O. Statistical Decision Theory and Bayesian Analysis. New York, USA; Springer, 2013
- [21] Järvelin K, Kekäläinen J. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 2002, 20(4): 422-446
- [22] Tan P N, Steinbach M, Kumar V. Introduction to Data Mining. Boston, USA; Pearson Addison Wesley, 2006



MAO Jiang-Yun, born in 1990, M. S. candidate. Her research interests include mobile data management and anomaly detection.

WU Hao, born in 1992, Ph. D. candidate. His research interests focus on trajectory computing.

SUN Wei-Wei, born in 1973, Ph. D., professor. His research interests include spatial-temporal data processing and analysis.

Background

This paper mainly focused on the field in the research on detecting the anomaly trajectory given its origin and destination information based on the road network. This field belongs to a classical problem in the field of data mining, called outlier detection. Outlier is regarded as a data object that is grossly different from or inconsistent with the remaining set of data. And there have already many outlier detection algorithms reported in the literature. Despite its importance, it is until Knorr tried his first attempts in 1998 that anomaly trajectory detection attracted attention of the public. Previous studies in detecting anomaly trajectory can be classified into three methods: utilizing distance or density of trajectories to detect anomaly, directly or indirectly doing statistics on the number of trajectories, transforming trajectories into the feature space to compute their “distance” through model and then classifies the normal ones and anomaly ones. However, as already stated in the previous paper, they have some

limitations such as suffering low efficiency, lacking in specification and failure to handle data sparsity problems.

Motivated by this, we propose an abnormal vehicle trajectory detection algorithm in road network space via Markov decision process model. Our approach supports returning the degree of abnormal and even is able to find out exactly where the abnormal behavior occurs. We conduct comprehensive experiments via real world taxi trajectory dataset. The results show that our approach outperforms current work not in both efficiency and effectiveness when the dataset is dense. We also conduct experiment with respect to long-trip data sparsity, the result again justifies the robustness of our approach.

This paper offers an extensive view on detecting anomaly trajectory, and we also provide a new perspective that will be helpful to the future researches in this field.