计算机科学
COMPUTER SCIENCE

结合时空关键字的轨迹范围查询混合索引结构

孟祥福, 李天朔, 张霄雁

引用本文

孟祥福, 李天朔, 张霄雁. 结合时空关键字的轨迹范围查询混合索引结构[J]. 计算机科学, 2024, 51(11A): 240200114-8.

MENG Xiangfu, LI Tianshuo, ZHANG Xiaoyan. Hybrid Index Structure for Trajectory Range Query Combined with Spatio-Temporal Keywords [J]. Computer Science, 2024, 51(11A): 240200114-8.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.240200114

结合时空关键字的轨迹范围查询混合索引结构

孟祥福　李天朔　张霄雁
辽宁工程技术大学电子与信息工程学院　辽宁 葫芦岛 125105
(mengxiangfu@lntu.edu.cn)

摘　要　对于路网上广泛的轨迹数据集,传统结合关键字特征的时空范围查询方法存在存储结构冗余和查询效率低下的问题,同时这些方法忽视了文本特征对优化查询结果个性化方面的潜在影响。为此,提出了一种结合文本特征的时空轨迹索引结构,称为 IG-Tree。其基本思想是将道路网络图划分为分层子图,并据此构建一个平衡的树结构,其中每个树节点均关联并存储其特定的轨迹数据。此外,设计的查询算法利用与 IG-Tree 节点相关联的子路网图的文本特征,筛选并提出范围边界处的不相关轨迹,实现高效且精准的文本空间范围查询。这种索引结构不仅有效集成了时间、空间和文本 3 个维度的信息,而且基于这种结构的查询方法能够支持基于时空关键字的轨迹范围查询,从而极大地满足用户查询的个性化需求。在 Porto 和 LA 数据集上的实验证明,IG-Tree 索引结构不仅在查询精度上表现出色,而且在响应速度上也具有显著优势,这进一步验证了其处理大规模轨迹数据集时的有效性和实用性。

关键词:时空关键字查询;轨迹数据;范围查询;混合索引结构
中图分类号　TP311

Hybrid Index Structure for Trajectory Range Query Combined with Spatio-Temporal Keywords

MENG Xiangfu, LI Tianshuo and ZHANG Xiaoyan
School of Electronic and Information Engineering, Liaoning Technical University, Huludao, Liaoning 125105, China

Abstract　For a wide range of trajectory datasets on the road network, the method of spatial-temporal range query combined with keyword features has redundant storage structure and low query efficiency. In this paper, a spatial-temporal trajectory index structure combining text features, called IG-Tree, is proposed. The basic idea is to divide the road network graph into hierarchical subgraphs and generate a balanced tree structure, in which each tree node maintains its associated trajectory. In addition, the query algorithm designed in this paper utilizes the text features of sub-images associated with IG-Tree nodes and deletes irrelevant trajectories at range boundaries to realize text space range query. Experimental results show that the proposed IG-Tree index structure shows high accuracy and fast response speed on Porto & LA dataset.

Keywords　Spatio-temporal keyword query, Trajectory data, Scope query, Hybrid index structure

1　引言

随着智能移动手机等携带全球定位设备的普及和基于位置的服务的快速发展,收集到的轨迹数据大幅增多。近期在时空轨迹的研究内容包括旅行时间评估[1]、轨迹数据压缩[2]、最优路径推荐[3]、高频路径检索[4]等。除了研究时空轨迹外,近些年越来越多的科研人员着重于研究将文本信息与各个空间位置相结合的语义轨迹[5]。基于位置的社交网络服务产生了大量语义轨迹,比如 Foursquare 和 Twitter,这些轨迹数据为这个研究方向提供了大量数据基础。其中的代表性工作包括语义轨迹中的模式挖掘[6]和活动轨迹检索[7]。

近年来的研究发现,传统的查询方法在距离搜索过程中仅通过几何属性衡量空间关系,而忽视了底层道路网络。Zhong 等[8]针对基于位置的查询提出了一种用于道路网络上的高度平衡且可拓展的索引,称为 G-Tree。这种方法是将路网递归划分为多个子路网,并在子路网构造一个树形结构索引。这种索引结构可以高效地检索路网中的顶点,但并不适用于轨迹数据。为此,Wang 等[9]提出了一种用于轨迹范围查询的道路感知索引,这种索引结构用于实现轨迹数据中的时空范围查询,即找到与给定范围相交的所有轨迹。这种结构能够支持路网中的时间-空间维度的查询,然而查询方法忽视了文本特征的约束。针对轨迹数据上的空间关键字查询,Cong 等[10]提出了一种混合索引结构,称为单元关键字意识(Cell-Keyowrd conscious B^+-Tree,B^{ck}-Tree)。这种结构支持利用文本相关性和位置邻近性来促进高效且有效的查询处理。

以往的研究倾向于仅仅关注轨迹的时空特征[11-12]或空间和文本方面[7,10]。然而,需要特别强调的是,在现实的许多场景中,用户往往需要同时考虑 3 个方面,才能获得最优的轨迹查询结果。具体来说,用户可能对特定时期(比如上周或过去 10 天)在特定区域(如附近地区或郊区)内的轨迹点(如活动)感兴趣。此外,轨迹点上标注的关键字包含了相关活动和用户经历等丰富信息。因此,将查询关键字作为轨迹检索的基础是该背景下很值得关注的一个研究方向。

例 1　图 1 显示了 7 条轨迹$\{t_1, t_2, t_3, t_4, t_5, t_6, t_7\}$和每条轨迹的属性(时间范围和文本特征),以及其经过的路网。

通信作者:李天朔(lee_tuffy@163.com)

查询内容为“找到在 10:00—12:00 时间范围经过范围 r 内且包含文本特征{'restaurant','shopping'}的轨迹”，得到的结果为{t_5}。

trajectory	time	keywords
t_1	0:00—24:00	{'hospital','coffee', 'government','shopping'}
t_2	13:00—22:00	{'market','cinema', 'shopping','restaurant','hospital'}
t_3	6:00—22:00	{'hospital','library', 'government','museum'}
t_4	0:00—12:00	{'hospital','shopping','market'}
t_5	8:00—12:00	{'shopping','restaurant','palace'}
t_6	6:00—20:00	{'market','hospital','shopping', 'bank'}
t_7	8:00—20:00	{'market','shopping','restaurant', 'hospital'}

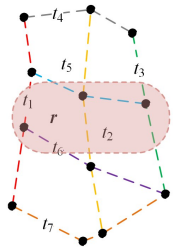

图 1 结合时空关键字查询的轨迹范围查询

Fig. 1 Trajectory range query combining spatio-temporal keyword search

难点：处理大量的轨迹数据需要高效的索引技术，从而实现轨迹范围搜索。当前研究存在的主要困难在于有效整合目标的时空和关键字特征，以过滤掉大量冗余和不相关的轨迹数据。根据 Christforaki 等[13]的实验结果可知，用户在现实生活中通常只标记少量关键字，一般在 2 到 5 个之间。在这项工作中，本文使用倒排索引技术，在查询处理过程中采用关键字优先修剪策略。此策略只加载包含查询关键字的相关轨迹。同时，倒排索引通过按照指定顺序遍历索引表实现关键字集合的顺序查询。此外，还需要一个有效的时空索引结构来组织每个关键字相关的轨迹点。

2 相关工作

2.1 轨迹范围查询

轨迹范围查询是指根据给定的时间范围和地理位置范围查询轨迹数据中的所有路径点的操作。轨迹范围查询是轨迹数据分析与挖掘的重要任务之一，被广泛应用于交通、物流、公共安全等领域。

轨迹范围查询的基本流程如图 2 所示。

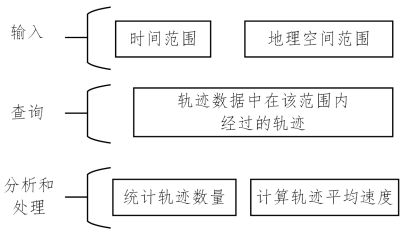

图 2 轨迹范围查询的基本流程图

Fig. 2 Basic flow chart of trajectory range query

目前，结合空间关键字的轨迹查询研究主要集中在以下几个方面。

1)轨迹数据的表示和存储：有效表示和存储运动物体的轨迹数据是高效查询的基础。研究人员提出了各种数据结构和技术来支持轨迹数据的存储和查询，如 R-Tree，Quad-Tree 和线段树等。

2)轨迹相似度测量：对于给定的轨迹查询，如何确定两条轨迹之间的相似度是一个关键问题。研究人员提出了各种相似性测量方法，以支持轨迹的精确和近似匹配，例如动态时间规整、最长公共子序列等。

3)轨迹索引与查询：为了提高查询效率，研究人员提出了各种轨迹索引与查询方法。例如，基于 R-Tree 和线段树的索引结构可以有效支持轨迹数据的范围查询和最近邻查询。此外，研究人员还提出了基于机器学习的轨迹查询方法，例如基于深度学习的轨迹匹配和推荐算法。

为了提高查询的效率和准确性，在查询轨迹范围时，本文考虑了如何快速有效地处理和检索轨迹数据。

2.2 范围索引结构

空间关键字查询的探索范围广泛，主要是由基于位置的服务衍生的空间文本对象驱动。在各类查询中，top-k 空间关键字查询是一项重要的研究，旨在识别具有最高空间接近度的 k 个对象，同时合并所有查询关键字。为了应对这一挑战，已经提出了几种有效的索引结构，如倒 R-tree[14] 和 IR^2-tree[15]。

常见轨迹索引结构的分类和特征如表 1 所列。

表 1 常见的轨迹索引结构

Table 1 Classification and characteristics of common trajectory indexing structures

类别	特点
基于时间范围的索引	基于时间范围的索引按照时间戳进行排序，能够快速查询某个时间范围内的数据点
基于空间范围的索引	基于空间范围的索引则按照地理位置信息进行排序，能够快速查询某个地理区域内的数据点
基于静态轨迹的索引	静态轨迹是指在一段时间内位置信息不发生变化的轨迹，例如交通摄像头的视频轨迹
基于动态轨迹的索引	动态轨迹则是指位置信息随时间变化的轨迹，例如车辆行驶轨迹
基于空间划分的索引	基于空间划分的索引结构将地理空间划分为若干个区域，每个区域内的数据建立索引
基于网格的索引	基于网格的索引结构则将地理空间划分为相邻且唯一的网格，在每个网格中建立索引
基于树的索引	基于树的索引结构利用树形数据结构进行组织，例如 Quad-Tree 和 R-Tree 等
基于链表的索引	基于链表的索引结构则利用链表等数据结构进行组织

为了高效处理轨迹相似性查询，Zheng 等[7]提出了一种名为 GAT 的索引结构，即混合网格索引。GAT 通过距离和活动相关性来删除搜索空间，用于按层次组织轨迹段和活动。这种方法能够实现轨迹数据的空间查询内容，但并未考虑其他维度对查询结果的影响。为了解决轨迹数据库中的时空范围查询问题，Wang 等[9]引入了一种新的索引技术，称为 RP-Tree，通过对道路网络进行划分来实现。RP-Tree 能够有效地处理轨迹数据的时空范围查询，并且将轨迹数据映射到道路网络中，然而，它忽视了数据中的文本特征。为了有效地处理轨迹上的空间关键字范围搜索及其涉及顺序敏感关键字的变体，Han 等[16]设计了一种新的索引结构，并将其标记为 IOC-Tree，专门用于组织轨迹数据，这种结构支持轨迹数据中的关键字查询。同时，Cong 等[10]开发了一种混合索引 B^{ck}-

Tree，该索引结构旨在处理查询和轨迹之间的文本相关性和位置接近性。IOC-Tree 和 B^{ck} Tree 支持轨迹数据的空间关键字查询，同时考虑了空间因素和文本特征，但并未应用到路网上，空间距离的计算仅仅使用简单的欧氏距离计算方法。

3 问题定义

本节给出研究问题的形式化定义以及相应的符号表示。

3.1 符号定义

研究问题所涉及的符号及其含义如表 2 所列。

表 2 符号定义

Table 2 Definition of symbols

符号	含义
$G=(V,E)$	路网
n	IG-Tree 的节点
$G_N=(V_N,E_N)$	子路网
N_l,N_r	节点 N 的左右子节点
L_N	N_l 和 N_r 连接的路径集合
$keys(N)$	节点 N 包含的特征关键字
$tr(e)$	在路径 e 上经过的轨迹集合
η	叶节点大小阈值

3.2 问题描述

定义 1 路网可表示为一种无向图 $G=\langle V,E\rangle$，其中 V 是一组顶点 v 的集合，E 是一组边 e 的集合。如图 3 所示。

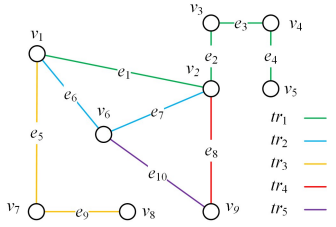

图 3 路网中的轨迹

Fig. 3 Trajectories in road network

定义 2 轨迹数据是指在路网中根据节点之间的某种顺序关系，获得其依次相连的路段序列，表示为 $tr=\{e_1,e_2,\cdots,e_n\}$（$e_i$ 为 E 中的某条道路，$i\in[1,n]$）。

结合空间关键字的轨迹查询是近年来的研究热点之一，主要涉及运动物体轨迹数据的处理和查询。该查询方法可以提供位置、时间、轨迹方向等多种查询，对轨迹分析、用户行为分析、城市规划等领域具有重要意义。

结合空间关键字进行轨迹查询的基本流程如图 4 所示。

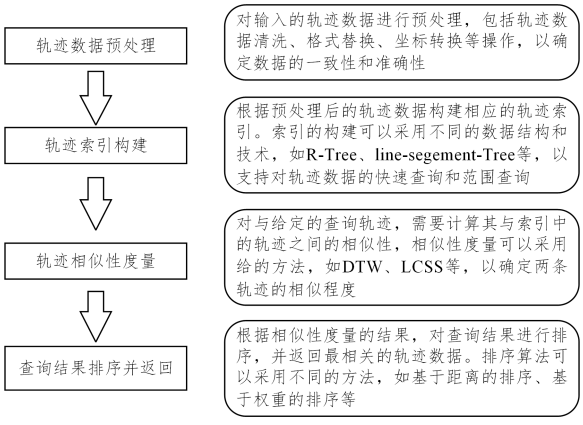

图 4 结合空间关键字的轨迹查询基本流程

Fig. 4 Basic flow of trajectory query combining spatial keywords

定义 3 结合时空关键字的轨迹查询是指根据给定的时间范围和空间范围以及文本特征，检索满足以上 3 种属性的轨迹数据集合。

给定轨迹数据集 $D=\{T_i\mid i=1,2,\cdots,n\}$ 包含 n 条轨迹，其中还每条轨迹包含 3 个属性，即路径信息、时间信息和文本特征。用户输入查询条件，其中包括空间范围 $S=\{lat_{\min},lat_{\max},lon_{\min},lon_{\max}\}$，时间范围 $T=\{time_{start},time_{end}\}$，一个关键字集合 $K=\{keywords_1,keywords_2,\cdots,keywords_n\}$。查询结果返回满足这些条件的所有轨迹 $t\in D$。具体来说，轨迹 t 必须在指定的空间范围 S 和时间范围 T 内，同时满足关键字 K 定义的文本特征。

4 理论技术

本文提出了一种高效的索引，该索引为文本轨迹的搜索和排序提供了以下功能。

空间过滤：过滤掉与查询范围无关的所有轨迹，减小搜索空间[17]。

文本过滤：丢弃与查询关键字无关的所有曲目，降低搜索成本[18]。

关联计算和排序：由于只返回匹配效果最好的前 k 个结果，且 k 值的设置远小于相关轨迹总数，因此采用增量搜索过程，将联合相关性的计算与文档排序的计算进行无缝集成。一旦查询到满足条件的前 k 个最优轨迹，搜索进程将停止[19]。

4.1 索引结构

为了实现对轨迹数据的时空关键字查询，索引结构需要满足以下特点。

1）路网感知：根据道路限制对路网数据进行剪枝操作。

2）层次化结构：层次树索引结构可以实现对较高层次的节点进行搜索和剪枝，对较低层次的节点进行查找，进一步达到精确搜索。

3）平衡结构：平衡结构可以降低树的整体高度，处理大数据集的效果优于不平衡结构。

4）反向文件索引：引入了反向文件索引，进一步实现了时间和空间搜索中的关键字检索。

为了用路网来索引轨迹，本文根据每条边包含的轨迹密度将路网划分为平衡的层次结构[20-21]。同时为了实现矩形的范围查询，保留了矩形的空间信息。基于这两个标准，本文将道路网络划分为一个层次树，其中每个节点包含一个子图（道路网络）及其最小边界矩形，两个节点之间的边表示从属关系[22-24]。

本文提出了一种带有倒排文件索引的道路分区树索引结构，其定义如下。

定义 4 IG-Tree 是一种平衡二叉搜索树，是一种基于路网的图划分结构。已知网络 $G=(V,E)$ 和叶节点大小阈值 η，其中每个节点 N 代表相应的子图 G_n。每个非叶节点 N 具有左子节点 N_l 和右子节点 N_r，分别用 G_{Nl} 和 G_{Nr} 表示。每个叶子节点 N 代表一个子图 G_N，其中 $|V_N|\leqslant\eta$。在构建树形索引结构时，对于层次结构不小于阈值 θ_H 的每个节点 N，根据出发时间 $idx(N)$ 对与 N 中道路相交的轨迹进行排序和索引。根节点不包含任何单词，充当将搜索范围分为两部分的分界点。叶子节点包含一个〈关键字-轨迹〉列表，该列表包含该节

点下索引轨迹的特征关键字，用 $keys(N)$ 表示。除根节点和叶节点外的内部节点存储一个词向量，该词向量将子树的范围分为两部分。左子树存储的词向量的模小于该节点包含的词向量，而右子树存储的词向量大于该节点包含的词向量。

IG-Tree 的结构如图 5 所示。

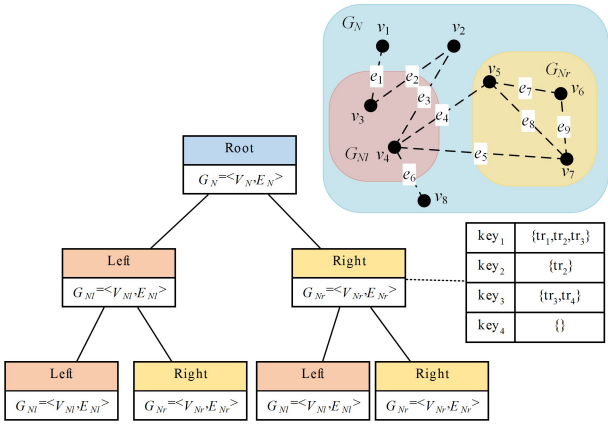

图 5 IG-Tree 结构

Fig. 5 IG-Tree structure

为了找到空间维度上的范围，本文将穿过道路 $e\in E$ 的轨迹表示为 $tr(e)$。在时间维度上，按照路径 e 的起始时间对轨迹进行索引，按照时间顺序对轨迹进行排序，表示为 $idx(e)$。为了进一步实现文本维度上的查询，根据轨迹中包含的关键字，一条轨迹可以包含多个关键字，因此一条轨迹可以索引到多个关键字分类。在本文中，$tex(key)$ 用于表示包含特征关键字 key 的轨迹。

4.2 构建索引

本节介绍了用于构建索引结构时使用的技术方法。

4.2.1 空间划分

定义 5 空间划分指路网被分为几个分区，每个分区是一个连续的区域。

图 $G_N=(V_N,E_N)$ 被分为 $G_{Nl}=(V_{Nl},E_{Nl})$ 和 $G_{Nr}=(V_{Nr},E_{Nr})$，这两部分由不存在任何重叠的矩形包围，它们之间连接的道路记为 L_N。分区的目的是获得边平衡的一对子图，即：

$$\min\left|\left|\sum_{e\in E_{N_l}}tr(e)\right|-\left|\sum_{e\in E_{N_r}}tr(e)\right|\right| \tag{1}$$

定理 1 在图 G_N 中，边平衡划分等价于顶点平衡划分。

证明：根据 $w(v)=\sum_{e\in v.edges}|tr(e)|$，可以得到：

$$\begin{cases}\left|\sum_{v\in V_{N_l}}\right|=\sum_{e\in E_{N_l}}|tr(e)|+\sum_{e\in L_N}|tr(e)|\\ \left|\sum_{v\in V_{N_r}}\right|=\sum_{e\in E_{N_r}}|tr(e)|+\sum_{e\in L_N}|tr(e)|\end{cases} \tag{2}$$

因此，可得出：

$$\left|\left|\sum_{v\in V_{N_{N_l}}}w(v)\right|-\left|\sum_{v\in V_{N_r}}w(v)\right|\right|=\left|\left|\sum_{e\in E_{N_{N_l}}}tr(e)\right|-\left|\sum_{e\in E_{N_r}}tr(e)\right|\right| \tag{3}$$

根据式(2)的最小值划分路网图 G_N。本文在算法 1 中提出了一种动态划分方法，该算法可以构建平衡度更高的索引结构。基于分割算法，递归划分路网图，将其分为子图 G_{N_l} 和 G_{N_r}，直到达到节点中的顶点数小于阈值 η 这一条件时停止。例如，在图 6 中，计算出路网 G 中顶点 $v_1,v_2,\cdots,v_9$ 经过的边数，算法尝试多种划分方案，得到满足式(1)的最优道路划分矩形 G_1 和 G_2，即满足以下条件：

$$\begin{aligned}&\min\left|\left|\sum_{v\in V_{N_l}}w(v)\right|-\left|\sum_{v\in V_{N_r}}w(v)\right|\right|\\&=\min\left|\left|\sum_{v\in G_1.V}w(v)\right|-\left|\sum_{v\in G_2.V}w(v)\right|\right|\\&=\min|9-11|\\&=2\end{aligned} \tag{4}$$

根据这个计算规律，继续计算得出矩形 G_3,G_4,G_5,G_6。

本文选择了图中较长的一边作为分区边，同时根据计算获得式(1)中的最小值进行划分。

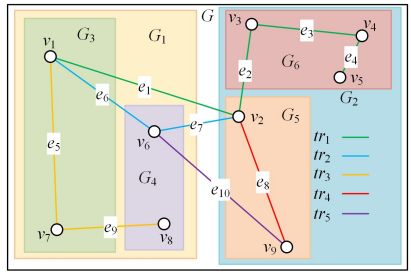

图 6 路网划分示例

Fig. 6 Example of road network division

4.2.2 时间分割

定义 6 时间分割是根据一定的标准或模式，将一个连续的时间序列划分为不同的片段或间隔。

对于给定的数据集 D，每条轨迹所经过的顶点可以看作是一个对象 $o\in D$。它是一个包含顶点属性的三元组 $(o.pos,o.time,o.key)$，其中 $o.pos$ 表示空间坐标位置，$o.time$ 表示对象的时间信息，$o.key$ 记录对象的文本特征信息。

对象 o 的时间特征由间隔 $(o.time_{start},o.time_{end})$ 表示，其中 $o.time_{start}$ 和 $o.time_{end}$ 分别表示开始和结束时间戳。为了实现标准化的时间索引，本研究根据统一的时间间隔对时间跨度进行划分。如图 7 所示，研究采用每天 24 h 为周期，时间间隔设定为 1 h。例如，范围(8:00,9:00)内的任何时间戳都可以表示为第 8 个单位，而时间跨度(12:00,17:00)对应于时间单位(12,17)。或者，若时间间隔单位为 30 min，则将(8:00,8:30)表示为第 16 个时间单位，将时间跨度(12:00,17:00)映射为时间单位(24,34)。

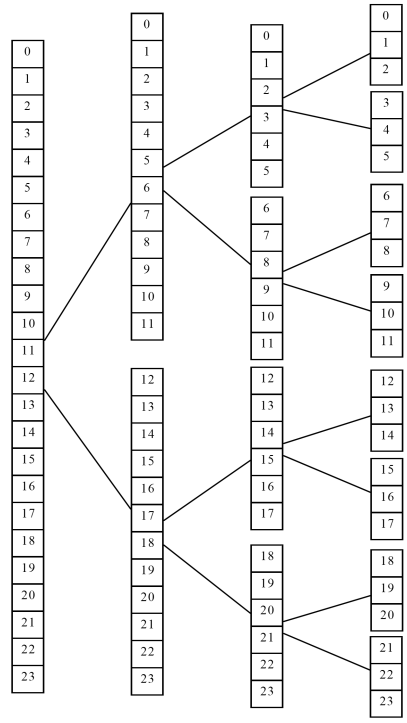

图 7 时域分割示例

Fig. 7 Example of temporal segmentation

4.2.3 文本特征提取与编码

为了将文本特征与时空轨迹数据相结合,从每个对象的轨迹中提取关键字特征,并使用文本向量表示[25]。

在初始阶段,计算每个对象的关键字的 TF-IDF 值,即获得所有关键字的权重值。假设数据集 K 包含所有关键字,对于一个对象内的多个关键字,根据每个关键字在关键字集合 K 中的频率进行排序。例如,如果对象 o_1 的关键字集为{k_1,k_2,k_3},出现频次分别为 56,198,3,则排序后的关键字集合变为{k_2,k_1,k_3}。

随后,将排序后的关键字集合排列形成词汇表,并利用词向量化模型提取关键字特征,将文本特征转化为数值特征。

处理低频词依赖于 K 中关键字的上下文信息。例如,"coffee"和"starbucks"经常同时出现,"coffee"的频次为 290,"starbucks"的频次为 15,可以将"starbucks"用"coffee"的本文向量来表示。同时,为了减少存储空间,本文省略了出现频次在指定阈值以下的低频词。

最后应用 K-Means 算法对集合中的关键字进行聚类,将相关关键字分配到同一聚类。

构建 IG-Tree 的具体步骤如算法 1 所示。

算法 1 构建 IG-Tree

输入:路网 G=(V,E),阈值 η,θ_H

输出:IG-Tree

1. index=Split(G,η,θ_H,1)
2. index 即为 IG-Tree 的根节点

函数 1 Split

输入:路网 G=(V,E),阈值 η,θ_H,当前树深度 h

输出:节点 N

1. 初始化节点 N,保存当前路网图覆盖的分割矩形范围
2. if d$\geqslant\theta_H$ then
3. 在节点 N 中索引轨迹
4. end if
5. if |V|$<\eta$ then
6. 构建叶节点
7. else
8. 根据坐标对候选点进行排序
9. $p=\text{argmin}\left|\sum_{j\in[0,i]} w(v_j)-\sum_{j\in[i+1,|V|-1]} w(v_j)\right|$
10. 根据数值 p 将图 G 划分为子图 G_{N_l} 和 G_{N_r}
11. N_l=Split(G_{N_l},η,θ_H,1)
12. N_r=Split(G_{N_r},η,θ_H,1)
13. 返回节点 N
14. end if

4.3 更新索引

轨迹数据集更新后,首先将未处理的轨迹数据映射匹配到路网中,每个匹配的轨迹段将被索引到其经过路网的边上。随着每个节点上轨迹数量的变化,IG-Tree 的结构可能变得不平衡。本文采用以下方法应对以上问题。

根据分区优化指标获取子图权值更新后不平衡节点 N^* 的集合。筛选出 N^* 中的最小公共祖先,即如果存在 $N\in N^*$,且 N 是 N^* 的祖先,则从 N^* 中去掉 N。文中根据分区优化指标,对于每个 $N'\in N^*$,将 N 分割为子节点的分区位置 p(即函数 $Split$)。具体为,若 N_l 的权重大于 N_r 的权重,则降低 p 的值,否则增加 p 的值,直到找到一个可以分配平衡的 p 值。最后递归划分,直到达到阈值 η 时停止。

4.4 结合时空关键字的轨迹查询

本节首先讨论了检索范围以及路网中道路、顶点和轨迹之间的位置关系,之后介绍了如何使用 IG-Tree 实现结合时空关键字的轨迹查询。

4.4.1 准备工作

在本文的算法中,将轨迹划分为由任意两个路网中的顶点连接的多个边[8]。根据边 e 与查询范围 r 之间的关系,将它们分为 3 类:内部(完全在查询范围内的边),外部(在查询范围外的边),以及交叉(与查询范围相交的边)。

内部:对于边 $e(v_i,v_j)$,v_i 和 v_j 都在 r 内,即 $v_i\in r$,$v_j\in r$。在本文中,完全在查询范围 r 内的边集表示为 E_{ri}。

外部:对于边 $e(v_i,v_j)$,v_i 和 v_j 都在 r 之外,即 $v_i\notin r$ 并且 $v_j\notin r$。在本文中,查询范围 r 之外的边集合表示为 E_{ro}。

交叉:对于边 $e(v_i,v_j)$,v_i 在 r 内,而 v_j 在 r 外,或者 v_j 在 r 内,而 v_i 在 r 外,即 $v_i\notin r$ 且 $v_j\notin r$,或者 $v_j\notin r$ 且 $v_i\notin r$。本文将与查询范围 r 相交的边集表示为 E_{rc}。

4.4.2 结合时空关键字的轨迹查询

结合时空关键字的轨迹查询的概要如算法 2 所示。对于查询范围 r、查询时间段(t_{qs},t_{qe})和查询关键字{key_1,key_2,…,key_n}的查询,本文通过递归遍历 IG-Tree 来检索满足查询条件的道路 e,并匹配通过道路 e 的所有轨迹。在本文中,查询结果表示为:

$$T_e=\bigcup_{e\in E,\{E.range\subset r\&E.time_{start}<t_{qe}\&E.time_{end}>t_{qe}\&E.keys\subset\{key_1,key_2,\cdots,key_n\}\}} tr(e) \tag{5}$$

其主要思想是利用 IG-Tree 结构,基于时空和文本信息,尽可能多地对轨迹数据做剪枝。

算法 2 结合时空关键字的轨迹查询

输入:查询范围 r,时间范围 t={t_{qs},t_{qe}},关键字集合 k={key_1,key_2,…,key_n}

输出:轨迹集合 T

1. 初始化道路集合 E,轨迹集合 T
2. n=root(IG-Tree)
3. E=Proofread(n,r,t,k)
4. for e in E do
5. T.add(tr(e))
6. end for

函数 2 Proofread

输入:节点 N,查询范围 r,时间范围 t={t_{qs},t_{qe}},关键字集合 keylist={key_1,key_2,…,key_n}

1. 阈值 η,θ
2. if N.range$\cap$r$\neq\varnothing$ then
3. if $\frac{\min(N.time_{end},t_{qe})-\max(N.time_{start},t_{qs})}{t_{qe}-t_{qs}}>\eta$ then
4. for k in keylist do
5. v_1=word2vec(k)
6. v_2=NearestVec(v1,N.key)
7. if consine_similarity(v1,v2)$>\theta$ then
8. continue
9. else
10. Prune(N)
11. end if
12. end for
13. if 节点 N 是叶子节点 then
14. return N.edge
15. else

16. return Proofread(N. left_child, r, t, keylist) ∪ Proofread(N. right_child, r, t, keylist)
17. end if
18. end if
19. end if

例 2 在图 8 和图 9 中，考虑带有范围 r、有效时间段(20:00,22:00)和关键字{'shopping'}的查询 Q。r 与 G_2 和 G_3 有重叠部分，因此可以遍历 G_2 和 G_3。这两个节点也符合查询的有效时间间隔和关键字特征，继续遍历到下一层；r 与 G_5 和 G_6 有重叠，但不与 G_4 和 G_7 重叠。因此，搜索 G_5 和 G_6，并通过两个最小边界矩形通过的轨迹 t_1，t_2，t_3 来满足有效时间间隔的条件，然后继续过滤文本特征，发现轨迹 t_1 和 t_2 满足查询条件。最终，得到查询结果 Q 的轨迹 t_1 和 t_2。

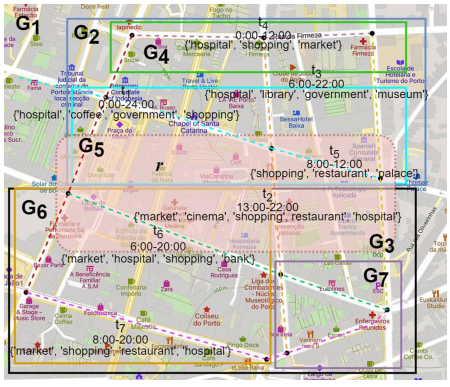

图 8 路网示例

Fig. 8 Example of road network

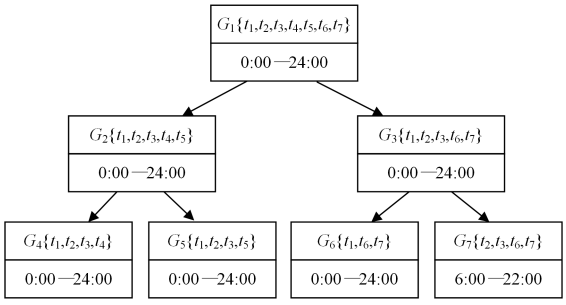

图 9 IG-Tree 示例

Fig. 9 Example of IG-Tree

5 实验评估

本节对几种索引结构进行效率和准确性的综合实验评估。

5.1 实验设置

所有算法均使用 Python 实现，并在一个拥有 2 个核心 CPU 和 2GB 内存的云服务器上执行，运行的操作系统是 Ubuntu。本文使用了两个真实的轨迹数据集，分别是 Kaggle 上的 Porto 数据集[8](葡萄牙波尔图的出租车轨迹)和 Foursquare 上的洛杉矶(LA)签到记录[26]。这两个城市的道路网络数据是从 OpenStreetMap[1)] 下载得到的。

为了更方便地匹配轨迹数据与路网数据，本文根据 K-Means 聚类对路网数据进行了网络分区处理。然后，利用隐式马尔可夫模型(ST-Matching)[23]将轨迹映射到路网中。

5.2 对比模型

B^{ck}-tree：用于解决轨迹上的空间关键字问题的索引结构，它基于一种新的混合索引，即单元关键字意识 B+树(B^{ck}-Tree)。它将空间区域划分为 4 个单元。利用文本相关性和位置邻近性实现高效且有效的查询处理。为了结合时间信息，根据轨迹上的顶点中包含的对应关键字的时间戳对轨迹进行排序。首先，B^{ck}-Tree 通过空间文本约束修剪轨迹，然后通过时间约束修剪轨迹。

RP-tree：一种支持有效范围查询的道路感知分区树。其基本思想是将路网图划分为层次化的子图，生成一个平衡的树结构，其中每个树节点都保持其相关的轨迹。我们紧凑地索引相应道路网络边缘轨迹的时空信息。对于文本信息，我们结合倒排文件(IF)。

GAT：它分层组织轨迹段和活动，以便可以同时通过位置邻近度和活动包含来修剪搜索空间，并提出一种有效的算法来计算最小值匹配距离和最小序列敏感匹配距离。对于每个网格，我们构建一棵 B+树，根据时间戳对轨迹上的顶点进行索引，然后为网格中的每个关键字构建点的倒排索引来记录文本信息。

5.3 性能评价

为了评估不同算法的效果，本文考虑了两个主要指标：查询结果准确率和算法的运行时间。

$$accuracy=\frac{All\cap result}{result}\times 100\% \tag{6}$$

其中，All 表示所有真正相关的时空对象，$result$ 表示查询返回的时空对象，$All\cap result$ 表示查询返回的结果中真正相关的时空对象。

在第一组实验中，使用不同数量的关键字来评估 4 种算法针对特定任务的时间消耗和准确性。如图 10 所示，随着关键字数量的增加，IG-Tree 算法的效率在两个不同的数据集中表现出最小的变化。同时，所有 3 种算法的准确性都出现了不同程度的下降。

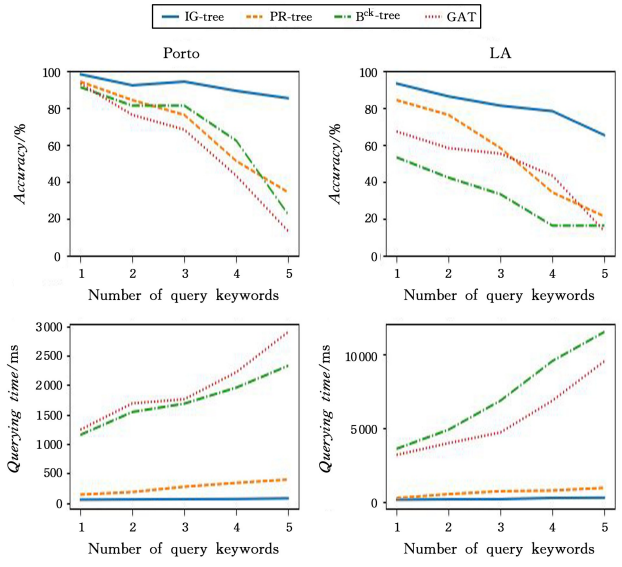

图 10 查询关键字数量变化的实验效果

Fig. 10 Experimental effect of varying number of query keywords

以 Porto 数据集为例，RP-Tree 和 IG-Tree 的查询精度相

1) https://www.openstreetmap.org/

当,但 IG-Tree 的效率超过了 RP-Tree。这种差异可以归因于 RP-Tree 在处理固定查询范围和有效时间段时性能最优,而在检索文本信息时,RP-Tree 需要遍历所有倒排文件索引,从而影响了效率。在 LA 数据集的背景下,我们发现,除了 IG-Tree 之外,GAT 被证明是最有效的。此外,随着关键字数量的增加,由于文本数据的矢量化,IG-Tree 的剪枝操作表现出更快的速度。因此,尽管关键字和轨迹查询都增加了,但 IG-Tree 的查询效率和准确性仍然值得称赞。

然后深入研究数据集大小对每种查询方法的影响,实验结果如图 11 所示。显然,随着读取数据量的增加,查询效率会下降。数据集大小对于索引构建过程中路网的划分和数据库的整体效率起着至关重要的作用。

更大的数据集与索引结构规模的增加相关,导致检索期间对修剪操作的需求更高,从而导致运行时间延长。在本实验中,IG-Tree 算法的索引结构在划分轨迹数据和路网方面表现出优越性,从而有助于提高检索效率。

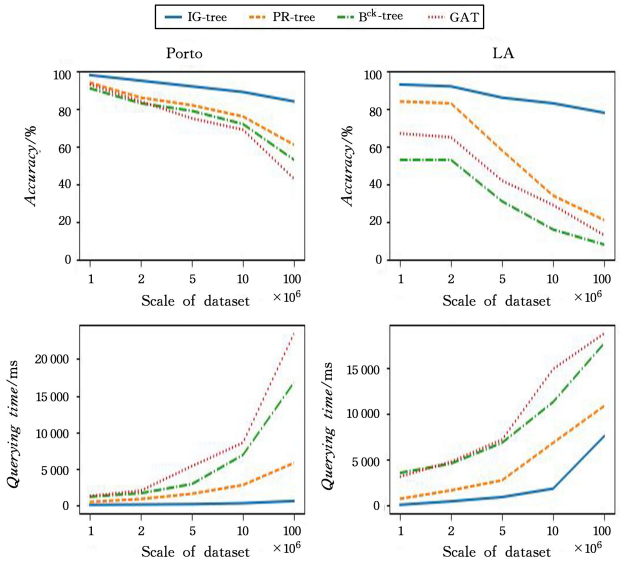

图 11　数据集规模变化的实验效果

Fig. 11　Experimental effect of varying scale of dataset

最后,我们研究了不同查询空间大小对 TQSK 上 4 种算法有效性的影响,实验结果如图 12 所示。

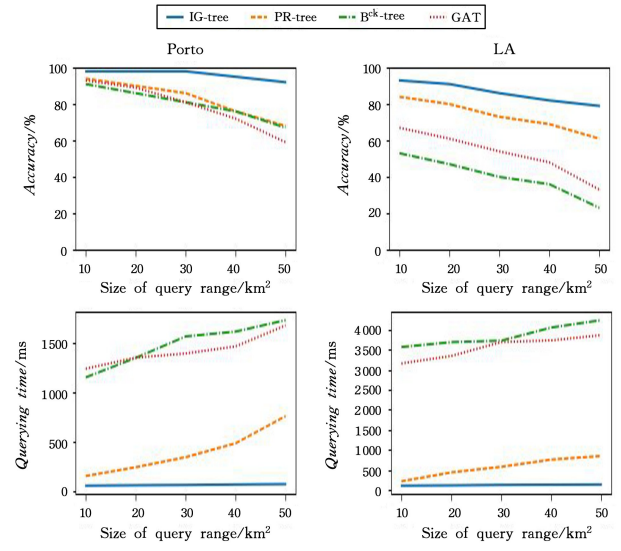

图 12　查询范围变化的实验效果

Fig. 12　Experimental effect of varying size of query range

可以看出,查询空间的增加会产生更大的搜索空间,从而导致检索到的轨迹数据量更大。所有算法的性能准确性都会随着查询空间的扩大而降低。

在这 3 个不同的基线中,随着查询空间的扩大,RP-Tree 始终优于 B^{ck}-Tree 和 GAT。这种优越性可归因于 RP-Tree 将道路网络图划分为分层子图,并附带修剪不相关轨迹以支持范围查询的搜索算法。同时,IG-Tree 始终表现出最佳性能。这归因于其对文本信息的量化,将关键字转换为机器可读的数字信息,并整合空间距离和时间有效性。

结束语　本文解决了轨迹数据处理中的一个重要问题,特别关注轨迹范围查询和关键字查询的高效执行。首先,本文详细探讨了当前轨迹数据处理方法固有的挑战和局限性,包括数据量大、复杂性高和查询效率低等问题。随后,作者提出了一种名为 IG-Tree 的新型索引结构,它同时解决了轨迹范围查询和关键字查询,有效克服了现有方法的缺点。

在论文中,作者精心设计并描述了 IG-Tree 索引结构。道路网络最初被划分为不同的子网,提高了查询效率和准确性。随后,轨迹数据与关键字特征一起被分配到这些子网,以改进数据组织和管理。IG-Tree 索引结构采用高效的查询算法,在实时性和准确性之间取得了平衡。

除了介绍索引结构和算法外,本文还进行了大量的实验来验证 IG-Tree 索引结构在处理真实轨迹数据方面的有效性和优越性。实验结果表明,与现有方法相比,轨迹范围查询和关键字查询的性能和效率都有显着提高。

总之,本文介绍了一种创新的轨迹数据处理方法。通过结合 IG-Tree 索引结构,同时解决了轨迹范围查询和关键字查询,为轨迹数据处理领域的研究和实际应用提供了有价值的见解和指导。

参 考 文 献

[1] WANG Y,ZHENG Y,XUE Y. Travel time estimation of a path using sparse trajectories[C]// Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2014:25-34.

[2] SONG R,SUN W,ZHENG B,et al. Press:A novel framework of trajectory compression in road networks[J]. arXiv: 1402.1546,2014.

[3] SU H,ZHENG K,HUANG J,et al. Crowdplanner:A crowd-based route recommendation system[C]// 2014 IEEE 30th International Conference on Data Engineering. IEEE,2014:1144-1155.

[4] LUO W,TAN H,CHEN L,et al. Finding time period-based most frequent path in big trajectory data[C]// Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data. 2013:713-724.

[5] ALVARES L O,BOGORNY V,KUIJPERS B,et al. Towards semantic trajectory knowledge discovery[J]. Data Mining and Knowledge Discovery,2007,12.

[6] ZHANG C,HAN J,SHOU L,et al. Splitter:Mining fine-grained sequential patterns in semantic trajectories[C]// Proceedings of the VLDB Endowment. 2014:769-780.

[7] ZHENG K,SHANG S,YUAN N J,et al. Towards efficient search for activity trajectories[C]// 2013 IEEE 29Th Interna-

tional Conference on Data Engineering(ICDE). IEEE, 2013: 230-241.

[8] ZHONG R, LI G, TAN K L, et al. G-tree: An efficient and scalable index for spatial search on road networks[J]. IEEE Transactions on Knowledge and Data Engineering, 2015, 27(8): 2175-2189.

[9] WANG Y, LI K, LI G, et al. Road-aware indexing for trajectory range queries[J]. IEEE Transactions on Knowledge and Data Engineering, 2023, 35: 8476-8489

[10] CONG G, LU H, OOI B C, et al. Efficient spatial keyword search in trajectory databases[J]. arXiv: 1205. 2880, 2012.

[11] CHAKKA V P, EVERSPAUGH A, PATEL J M, et al. Indexing large trajectory datasets with seti[C]// CIDR. 2003: 76.

[12] PFOSER D, JENSEN C S, THEODORIDIS Y. Novel approaches to the indexing of moving object trajectories[C]// Very Large Data Bases Conference. 2000.

[13] CHRISTOFORAKI M, HE J, DIMOPOULOS C, et al. Text vs. space: efficient geo-search query processing[C]// Proceedings of the 20th ACM International Conference on Information and Knowledge Management. 2011: 423-432.

[14] ZHOU Y, XIE X, WANG C, et al. Hybrid index structures for location-based web search[C]// Proceedings of the 14th ACM International Conference on Information and Knowledge Management. 2005: 155-162.

[15] DEFELIPE I, HRISTIDIS V, RISHE N. Keyword search on spatial databases[C]// 2008 IEEE 24th International Conference on Data Engineering. IEEE, 2008: 656-665.

[16] HAN Y, WANG L, ZHANG Y, et al. Spatial keyword range search on trajectories[C]// International Conference on Database Systems for Advanced Applications. 2015.

[17] FAN X, LI S, LAFORGE P D, et al. Em-based design approach for multi-band filters by reflected group delay method and cascade space mapping[C]// 2019 IEEE MTT-S International Microwave Symposium(IMS). 2019: 1035-1037.

[18] CHANDRIKA G N, REDDY E S. An efficient filtered classifier for classification of unseen test data in text documents[C]// 2017 IEEE International Conference on Computational Intelligence and Computing Research(ICCIC). 2017: 1-4.

[19] WANG Y, LIU Y, BLASCH E, et al. Simultaneous trajectory association and clustering for motion segmentation[J]. IEEE Signal Processing Letters, 2018, 25(1): 145-149

[20] HSUEH Y L, CHEN H C. Map matching for low-sampling-rate GPS trajectories by exploring real-time moving directions[J]. Information Sciences, 2018, 433-434: 55-69.

[21] LIAO C, CHEN C, XIANG C, et al. Taxi-passenger's destination prediction via GPS embedding and attention-based bilstm model[J]. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(5): 4460-4473.

[22] YUAN J, ZHENG Y, XIE X, et al. T-drive: Enhancing driving directions with taxi drivers' intelligence[J]. IEEE Transactions on Knowledge and Data Engineering, 2013, 25(1): 220-232.

[23] YANG Z, SUN H, HUANG J, et al. An efficient destination prediction approach based on future trajectory prediction and transition matrix optimization [J]. IEEE Transactions on Knowledge and Data Engineering, 2020, 32(2): 203-217.

[24] YU D, SHI X, CHAI L, et al. Balancing localization accuracy and location privacy in mobile cooperative localization [J]. IEEE Transactions on Signal Processing, 2023, 71: 2804-2818,

[25] DZISEVIČ R, SĔOK D. Text classification using different feature extraction approaches[C]// 2019 Open Conference of Electrical, Electronic and Information Sciences(eStream). 2019: 1-4.

[26] ZHENG Y, CHEN Y, XIE X, et al. Geolife2. 0: A location based social networking service[C]// 2009 tenth International Conference on Mobile Data Management: Systems, Services and Middleware. IEEE, 2009: 357-358.

MENG Xiangfu, born in 1981, Ph. D, professor, Ph. D supervisor. His main research interests include spatio-temporal big data analysis, medical image analysis and artificial intelligence.

LI Tianshuo, born in 1999, master. His main research interests include spatio-temporal big data analysis and visualization.