基于出行模式子图的城市功能区域发现方法

肖 飞 王 悦 梅逸男 白 璐 崔丽欣
(中央财经大学信息学院 北京 100081)

摘 要 城市的功能区域是指在城市的发展过程中逐渐形成的功能(如工业、商业、居住、教育等)相对固定的地理区域。这些区域间的位置结构影响着城市中居民的出行模式,与此同时,城市居民的出行模式也客观地反映了城市不同区域的真实的功能定位。文中以出租车运行轨迹数据为基础,研究城市居民的出行模式,并根据所得模式实现城市功能区域的自动化发现。主要思路及贡献包括:1)使用车辆轨迹及路网结构数据构造区域模式图(region pattern graph)结构,并提出区域模式图构建算法,采用图结构将城市的不同地理区域连接起来;2)提出自底而上的功能区域发现算法(Bottom-Up Functional Region Discovering,BUFRD)框架及基本实现思路,包括提出频繁出行模式子图挖掘算法,发现区域模式图中频繁出现的出行模式;3)提出功能区域聚类算法,聚类已获取的出行模式子图集,并最终实现城市功能区域的发现。实验结果表明,通过所提方法发现的城市功能区域较传统方法所得结果的功能纯度更高,其熵值比传统方法降低了至少 10%。

关键词 城市大数据,数据挖掘,城市功能区域,出行模式子图
中图法分类号 TP311 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2018.12.044

City Functional Region Discovery Algorithm Based on Travel Pattern Subgraph

XIAO Fei WANG Yue MEI Yi-nan BAI Lu CUI Li-xin
(School of Information, Central University of Finance and Economics, Beijing 100081, China)

Abstract City's functional regions refer to the geographical regions with relatively fixed functions(such as industry, commerce, housing, education, etc.) in the development of city. The position structure of these functional regions affect people's travel patterns, and these travel patterns also objectively reflect the real function of regions. This paper focused on the travel patterns of urban residents by using the taxicabs' trajectory data, and obtained functional regions according to these travel models. The main contributions of this paper are as follows. Firstly, this paper constructed the region pattern graph by using the taxicabs' trajectories and the road network structures, and then connected different geographical regions via the graph structure created by region graph pattern construcing algorithm. Secondly, this paper proposed the framework and basic implementation idea of bottom-up functional region discovering algorithm, including mining the frequent travel pattern subgraphs and discovering frequent travel pattern from these subgraphs. Thirdly, this paper proposed a functional region cluster algorithm to cluster the obtained travel pattern graph set, thus discovering the functional regions according to the clustering results. The experimental results show that this method is effective and achieves higher purity of the function compared with traditional methods, and the entropy is decreased by 10%.

Keywords City big data, Data mining, City functional region, Travel pattern subgraph

随着城市的不断发展,城市中区域的功能也逐渐集中,可能出现如工业区、商业区、居住区等功能较集中的区域。这些区域的地理位置结构影响着人们生活的各方面,最直接的是对人们的日常出行模式产生影响。因此,分析人们的日常出行模式,会获得与城市区域功能相关的知识。这种根据客观日常出行模式发现的功能区域能加深我们对城市各部分功能的理解,为管理机构和部门提供有价值的城市规划、交通治理等方面的决策支持意见。目前,大多数城市的出租车都安装了成熟的 GPS 监控系统,主要目的是记录出租车运营的基本状态,如:载客与否、行驶速度、路径坐标点等。这些数据客观地记录了城市中乘坐出租车人群的出行状态以及城市道路的运行状态。这些数据汇总之后,不仅可用于监管出租车本身,还常常被交通管理部门结合其他数据一起用于分析实时路况或其他问题。

到稿日期:2017-10-13 返修日期:2018-01-11 本文受国家自然科学基金(61503422,61602535),北京市社会科学基金项目(15JGC150),中央财经大学科研创新团队支持计划资助。

肖 飞(1994—),男,硕士生,主要研究方向为数据挖掘、数据库;**王 悦**(1981—),男,副教授,CCF 会员,主要研究方向为图数据挖掘、机器学习,E-mail:wangyuecs@cufe.edu.cn(通信作者);**梅逸男**(1996—),男,主要研究方向为数据挖掘;**白 璐**(1983—),男,讲师,主要研究方向为机器学习、模式识别;**崔丽欣**(1986—),女,讲师,主要研究方向为特征提取、智能优化算法。

政府在城市建设和管理过程中,不能完全采用自顶向下的方式规划城市功能和建设,应结合城市已经形成的各区域功能,在此基础上,提供人性化服务。同时,随着社会的发展,区域的功能也可能随着时间发生变化。因此,自动化发现城市发展过程中形成的功能区域,对城市建设的管理、规划非常重要。

基于前述分析,本文将尝试结合城市人群的日常出行模式及城市道路结构数据,发现实际发展过程中自发形成的功能区域。本文的主要贡献包括:

1)根据出租车载客状态和采样时间信息,从出租车数据中提取车辆轨迹数据。提出区域模式图构建算法(Region Pattern Graph Construct,RPGC),并使用车辆轨迹数据及路网结构构造区域模式图结构,采用图结构将城市的不同地理区域连接起来,形式化地给出功能区域发现的相关概念及定义。

2)提出频繁出行模式子图挖掘算法(Travel Pattern Subgraph Discover,TPSubgraph_discover),找出区域模式图中频繁出现的出行模式子图。

3)提出功能区域聚类算法,通过图核方法,计算各频繁出行模式之间的相似度,得出相似度矩阵。使用层次聚类方法聚类已发现的出行模式子图。最终根据频繁出行模式,实现功能区域的发现。

4)基于北京市约 3 万辆出租车实际运营的真实 GPS 数据,测试并对比了本文方法及前人的工作。实验证明:本文方法所发现的功能区域较传统方法结果的功能纯度更高(其熵值比传统方法降低了 10%)。

本文第 1 节主要介绍和分析与本文研究相关的工作的研究现状;第 2 节形式化地给出区域模式图、频繁出行模式子图等概念的相关定义以及本文主要解决的问题;第 3 节给出本文主要研究框架,并提出面向单一图结构的自底而上的功能区域发现算法、频繁出行模式子图挖掘算法等相关算法;第 4 节使用真实的北京市数据测试了本文方法,并与相关工作进行对比,实验证明本文算法在功能纯度上更高;最后总结全文并给出未来工作的方向。

1　相关工作

近年来,随着 GPS 技术的发展与广泛应用,各种功能更强、成本更低、安装更便捷的 GPS 设备不断出现,相关的 GPS 监控设备已被用到城市公共交通的各个方面(如公交车、地铁、出租车甚至共享单车等),人们可以通过相关设备获取大量与出行相关的 GPS 数据,并用于各种研究中[1]。由于出租车的出行更具有个体代表性且更能反映公路的状态,很多文献尝试使用出租车 GPS 监控数据分析城市运行及人类行为的相关问题[2-4]。文献[5]最早提出城市计算(Urban Computing)的概念,该领域的相关研究也逐渐兴起。城市计算即通过传感设备、个人、交通工具等手段探测城市的动态变化,并进一步通过计算获得知识,运用所得知识更好地服务社会。目前,城市计算领域的研究都基于对各种定位数据的处理与分析,通过不同的算法实现相应的研究目的。根据研究目的不同,该领域研究大致可以分为以下两种类型:1)功能区发现研究[2];2)城市规划研究[3]。微软亚洲研究院的郑宇等人已经做了大量的相关研究[2-3,5-6],文献[6]根据北京市的道路结构信息,将北京市地理范围划分为不同区域,同时结合北京市的 POI 数据以及北京市的出租车出行轨迹数据,基于主题模型,将区域对应于主题模型中的文档,区域功能对应于模型的主题,POI 类别对应于主题模型的元信息,使用基于潜狄利克雷分配(LDA)的方法实现对城市功能区域的分析。文献[2]在文献[6]的基础上提出了潜在活动轨迹(Latent Activity Trajectory)的概念,并在原有出租车轨迹数据的基础上,增加了城市居民出行的公交车刷卡数据,从定位数据和迁移数据结合的角度实现了功能区域发现,同时更加详细地介绍了如何将北京市地理地图信息划分成一个个不同区域,在区域的基础上进行相应分析。文献[3]根据出租车在不同地区之间的迁移数据,从地区交通问题和地区之间的连通结构及相关关系两个维度分析城市规划中存在的问题,同时在城市规划方面提供相应建议。除此之外还有其他相关研究,如文献[7]基于用户手机一个月的定位数据,提出了一种基于亲和度(一种对复杂网络中两点相似度的度量)的计算方法,通过计算人们出发、到达分布的相对熵值(Relative Entropy)识别出不同的功能区域;并提出核密度评估的方法,以衡量功能区域内部的分布密度情况。文献[8]基于 Petri 网模型优化城市交通流状态,将 Petri 网分析作为遗传算法的适应度函数,对整个城市路网进行实时控制,从而提供一种最好的路线,优化交通流量,提高城市运行效率。文献[9]在出租车轨迹数据的基础上,通过出租车的起点和终点,采用带噪音的、基于密度的空间聚集算法来对城市交通模型进行相关研究,并证明了该算法的有效性。文献[10]根据出租车数据提取了 6 个特征,分析了土地使用和功能区域问题。还有将出租车轨迹数据映射成图结构的相关研究[11],文献[4]基于图建模,并结合可视化的工具,在出租车轨迹数据的基础上研究了城市出行模式,但未进一步研究区域分析和城市规划等。

然而,现有研究存在以下两方面的问题:1)只考虑出行的起点和终点,没有考虑中间途经点,忽略了途经点所包含的一些重要信息(如经验丰富的司机会故意绕开某些拥堵路段等);2)虽然使用图结构表示出行模式,但很少基于图结构进行相关分析。这些问题限制了已有研究对城市人群出行模式的进一步分析。

频繁子图挖掘是图挖掘领域的重要分支,主要研究集中于从图集合中挖掘频繁子图结构[12],或从单一大图结构中挖掘出频繁子图[13],又或是针对查询图在单一大图结构下的频繁度挖掘[14]。本文认为,不同的交通出行模式最终可展示为区域模式图所包含的子图,因此可应用图挖掘算法进行相应分析。

图核函数[15]是一种基于图结构的核函数,直观上可以理解为是计算图结构之间的相似度。图核函数由于具有可在低维空间计算图的拓扑结构及将附加信息映射到高维空间关联状态的优良特性,近年来被应用到很多不同的研究领域[16-18],本文将使用图核函数来解决子图聚类问题。

本文尝试结合车辆轨迹 GPS 数据、区域数据,以发现城市区域的不同功能。本文提出了区域模式图的概念,并基于频繁子图挖掘算法的思路提出了区域功能发现算法。具体地,该方法采用类似于文献[6]的区域标记方法,将北京市的

地理信息映射成不同区域。但本文划分的区域粒度更细，即保留的路网信息更加详细，区域更小、更多。文献[6]只保留了城市主干道，而本文在区域划分时不仅保留了城市主干道，同时保留了第二干道、第三干道等道路。然后本文根据轨迹信息，将划分出的区域映射成图结构，基于频繁子图挖掘理论，挖掘频繁出行模式图结构，并进一步计算图核相似度[19]，采用聚类算法对频繁模式进行聚类，确认并发现相关功能区域。最后，通过详实的实验证明该算法的有效性。

2 基本概念及问题定义

为了实现城市功能区域的自动化发现，本文采用的具体思路为：将城市的地理数据及车辆轨迹建模成为图结构，并尝试从该图结构中搜索频繁出现的出行模式；将所得频繁模式聚类，与城市地理区域相结合，最终发现城市的功能区域。为了更清晰地描述我们的问题及后续工作，本节给出：1)地理信息区域划分的方法；2)区域模式图的相关概念及构造方法；3)频繁出行模式子图的概念；4)城市功能区域发现问题的形式化定义。

2.1 区域划分

城市错综复杂的道路网络系统将城市划分成不同的区域，人们在不同区域之间的迁移反映了不同区域功能对人们行为的影响。在地理信息系统中，表示空间数据的方法有两种[2]：基于向量的模型和基于栅格的模型。基于向量的模型一般通过点、线和多边形的方式表示空间对象，而基于栅格的模型将空间分成网格数据，并以栅格为研究对象。本文针对功能区域进行研究，根据道路结构信息将地理数据划分成不同区域，因此采用基于栅格的模型来表示道路网络更加合适，每个区域由多个栅格组成，如图 1 所示。

图 1 北京市的路网信息

Fig. 1 Information of road network of Beijing

从图 1 可以看出，北京市的道路网络结构将北京市分割成不同的区域。通过基于栅格的表示方法，将地图映射成二进制的图片格式，图中每个像素点的值为 0 表示该点在道路上，值为 1 表示该点在某一区域内。但道路之间存在缝隙或者环路的情况(如图 2(a)所示)，为了移除这些不必要的区域细节，本文采用像素扩充的方法覆盖这些区域，结果如图 2(b)所示；然后采用文献[20]中的方法获取道路框架结构，结果如图 2(c)所示；最后，根据细化之后的结果，采用文献[21-22]提出的方法进行连通区域的识别，并标注不同颜色，如图 2(d)所示。至此，完成对城市范围的区域划分。

定义 1(区域划分(Region Division)) 给定城市地图 M，其道路集合 $Road=\{road_1,road_2,\cdots,road_x\}$，其中 $x=|Road|$，根据道路信息可以将 M 划分为区域集合 $R=\{r_1,r_2,\cdots,r_n\}$，使得 $M=r_1\cup r_2\cup\cdots\cup r_n$。

其中，x 表示地图 M 包含的道路总数。在对地图数据进行预处理后，x 应该为图 2(c)所示的道路框架结构的道路数，最终的 r_i 为图 2(c)中的一个封闭区域。

(a) (b) (c) (d)

图 2 区域划分

Fig. 2 Region segmentation

2.2 区域模式图构建

在完成 2.1 节的城市区域路网划分工作后，我们将各区域作为节点，使用车辆出行的轨迹将这些节点连接起来，最终形成图结构，称为区域模式图。后续的相关工作均在该结构及相关概念中展开。由此，本节提出区域模式图及相关概念，并给出其构建算法。

定义 2(轨迹(Trajectory)) 轨迹 $T_r=\{n_1,n_2,\cdots,n_k\}$，有 $k=|T_r|$，其中 $n_i=\{lon,lat,t,a\}(1\leqslant i\leqslant k)$ 表示该轨迹的一个采样点。

其中，k 表示该轨迹的采样长度，即该轨迹采样次数；采样点 n_i 中，属性 lon 和 lat 表示该点的经、纬度信息；属性 t 表示该点的采样时间；i 表示采样点在轨迹中的顺序值。属性 a 有 3 种可能取值：1)当 $i=1$ 时，其值为“起始点”；2)当 $i=k$ 时，其值为“终点”；3)当 $1<i<k$ 时，其值为“途经点”。

定义 3(区域模式图(Region Pattern Graph)) 区域模式图 $G=\{V,E,L\}$，其中 V 是点集合 $V=\{v_1,v_2,\cdots,v_m\}(m=|V|)$，$E$ 是边集合，L 是标签函数。

在区域模式图中，节点代表一个区域，边代表两个区域之间有轨迹经过；对于点 v_i 来说，其标签与属于该区域的采样点的属性 a 有关。$v_i=\{vid,s\_num,m\_num,e\_num,label\}$，其中 s_num 表示该区域中起始点的数目，m_num 表示该区域中途经点的数目，e_num 表示该区域中终点的数目。而属性 $label$ 是根据该点在 s_num,m_num,e_num 3 个维度上的值，通过 k-means 算法聚类之后的类标签。对于边，其标签为经过该边两端顶点的轨迹数量。例如：对于边 $\langle v_1,v_2\rangle$ 而言，存在轨迹 T_r，其中 $n_i\in v_1$ 并且 $n_{i+1}\in v_2$ 或 $n_i\in v_2$ 并且 $n_{i+1}\in v_1$ $(1\leqslant i\leqslant k-1)$，则该边权重加 1。

例 1 已知轨迹集 $S=\{T_{r0},T_{r1}\}$。其中，轨迹 $T_{r0}=\{n_1,n_2,n_3,n_4,n_5\}(n_1\in v_0,n_2\in v_1,n_3\in v_2,n_4\in v_3,n_5\in v_4)$，$T_{r1}=\{n_1,n_2,n_3,n_4\}(n_1\in v_3,n_2\in v_2,n_3\in v_5,n_4\in v_1)$。则轨迹集合 S 所对应的区域模式图如图 3 所示，图中顶点内的数字为 s_num,m_num,e_num 的值。

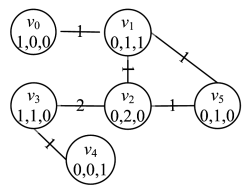

图 3 区域模式图实例

Fig. 3 Example of region pattern graph

在构造区域模式图的过程中需要用到车辆轨迹数据,但由于原始出租车数据按各车传感器返回时间整体排序存储,若把所有车辆信息混杂在一起,将难以处理。为获取按车辆分组的单独轨迹数据,需对原始数据进行分组(group by)处理,将其转化为针对各车辆分组的轨迹数据,因此需做如下预处理:

首先,通过遍历所有轨迹数据,根据车牌信息定位是否为同一辆车,接着通过载客状态判断轨迹的起始位置,再通过载客状态或者采样时间间隔判断轨迹终点,最终提取出全部车辆的所有载客数据,同时剔除采样时间异常的轨迹,达到数据清洗效果。本系统单独使用了轨迹数据预处理算法对数据进行预处理,具体步骤将在后文给出。

对轨迹数据进行预处理后,将构建区域模式图,具体步骤如算法 1 所示。

算法 1　区域模式图构建算法(Region Pattern Graph Construct,RPGC)

```
输入:轨迹数据 T_r 集,栅格点与区域对应 Hash 表 /* key=栅格 id,
      value=区域 id */
输出:区域模式图 G
1. for each t_r in T_r:
2.    for i in range(len(t_r)-1):
3.       if Hash[n_i] not in G :
4.          G.add_node(Hash[n_i])
5.       if Hash[n_{i+1}] not in G :
6.          G.add_node(Hash[n_{i+1}])
7.       Edge(n_i, n_{i+1}).w+=1
8.    End for
9. End for
10. Return G.
```

在区域模式图构建算法执行之前,需采用 2.1 节的方法,用栅格点表示地理数据信息,然后确定栅格点与划分区域之间的对应关系,得到相应的 Hash 表。

该算法遍历所有轨迹数据,且对于每条轨迹,获取轨迹点所在区域,进而向区域模式图中添加点和边,最终构造出完整的区域模式图结构。

在执行完该算法之后,需要对图结构做进一步处理,即运用标签函数得出点的最终标签 *label* 值。

对于该算法来说,时间复杂度为 $O(n*(m-1))$,其中 n 为轨迹数量,m 表示平均轨迹长度。根据统计可知,每天的轨迹数据量大约为 700000 条。实验中,生成的区域模式图的节点数(区域数)约为 836,边数超过 10000。

2.3　频繁出行模式子图

本节将在区域模式图的基础上,给出频繁出行模式子图的概念及相关介绍。

定义 4(子图同构(Subgraph Isomorphism))　设图 $S=\{V_s,E_s,L_s\}$ 是图 $G=\{V,E,L\}$ 的子图,S 在图 G 中的子图同构是内射函数 $f:V_s\to V$ 满足

1) $L_s(v)=L(f(v))$ 对于所有的 $v\in V_s$ 成立;

2) $(f(u),f(v))\in E$,并且 $L_s(u,v)=L(f(u),f(v))$ 对于所有的 $e\in E_s$ 成立。

定义 5(频繁子图(Frequent Subgraph))　用 $f_1,f_2,\cdots,f_m$ 表示子图 $S=\{V_s,E_s,L_s\}$ 在图 $G=\{V,E,L\}$ 中的同构图,给定最小支持度 τ,若 $m>\tau$,则认为图 $S=\{V_s,E_s,L_s\}$ 是图 $G=\{V,E,L\}$ 中的一个频繁子图。

上述定义有助于对本文所提的出行模式子图概念、频繁出行模式挖掘算法进行理解。本文以区域为研究对象,生成从该区域出发的出行模式子图,从区域的角度自底而上地在整个区域模式图中查找频繁出行模式子图。其相关定义如下。

定义 6(出行模式子图(Travel Pattern Subgraph))　出行模式子图 $Q=\{V_q,E_q,L_q\}$,其中 $V_q\subseteq V$,$E_q\subseteq E$ 且 $G=\{V,E,L\}$,Q 对应从固定起始点出发的出行模式。

频繁出行模式子图是频繁子图的扩展,因此频繁度 τ 的设定决定了最终可获取到不同频繁出行模式的数量。当 τ 设定过低(如 $\tau<5$)时,频繁出行模式的数量过多,区域功能划分会受到大量偶发、低阈值频繁出行模式的影响,导致最终结果的随机性较大;反之,当 τ 值较大(如 $\tau>15$)时,频繁出行模式的数量又会过少,功能区域划分会因缺少某些典型模式而无法反映区域的真实功能。因此,为了降低结果的随机性,同时也为了不丢失某些重要信息,通过实验测试,我们最终将频繁度 τ 设定为 10,即认为在区域模式图 G 中,同构体数目 $m>10$ 的出行模式子图均为频繁出行模式子图。

2.4　功能区域发现

上述内容介绍了与本文主要问题相关的定义,本节将给出功能区域及相关问题的定义。

定义 7(功能区域(Functional Region))　给定区域功能集合 $T=\{t_1,t_2,\cdots,t_k\}$,给定区域 R,其区域功能向量 $FR=\{R_{t1},R_{t2},\cdots,R_{tk}\}$,其中 R_{ti} 表示第 i 个功能在该区域所占比例,当某个功能的占比明显高于其他功能时,该区域功能相对集中,形成功能区域。

根据前述概念,本文主要解决的问题可以定义如下。

问题 1:城市功能区域发现。设区域集 $R=\{r_1,r_2,\cdots,r_n\}$ 为城市地图 M 上按路网结构的一个划分($M=r_1\cup r_2\cup\cdots\cup r_n$),其中 $n=|R|$,给定车辆行驶轨迹集合数据 $T_r=\{n_1,n_2,\cdots,n_k\}$,则本文需解决的重要问题是如何结合 T_r 及 R 数据,最终获取 M 的新的划分 $R'=\{r_1',r_2',\cdots,r_n'\}$,以满足对于 $\forall r_i'\in R'$,r_i' 中包含的城市功能达到单一。

由问题 1 的描述可知,若要获得该问题的最优解,需要枚举全部区域的组合并检查其相关的功能单一性。由此可知,该问题是难度为 NP-hard 的组合优化问题。为解决该问题,本文融合了车辆行驶轨迹及区域集数据,构造区域模式图,并基于频繁子图挖掘思路,设计出行模式子图发现算法,挖掘频繁出行模式。本文方法不仅考虑了出行轨迹的起点与终点,同时综合考虑了出行轨迹的途经点信息,用图核算法计算图与图之间的相似度,并根据层次聚类算法对频繁出行模式子图聚类,最终将具有相同或相似功能的区域聚集,进而识别出该类区域的功能,解决功能区域发现问题。

3　自底而上的城市功能区域发现算法

本文实现了自底而上的城市功能区域发现系统。该系统的主要思路为:结合车辆轨迹数据及城市地理空间数据构造

区域模式图，并采用数据挖掘手段发现区域模式图中频繁出现的出行模式子图，聚类频繁出行模式子图，最终获取城市不同区域的功能细分。本系统共包含 3 个重要模块：1）模式图生成模块；2）出行模式子图挖掘模块；3）功能区域发现模块。本节详细介绍该系统的框架及实现原理。系统框架如图 4 所示。

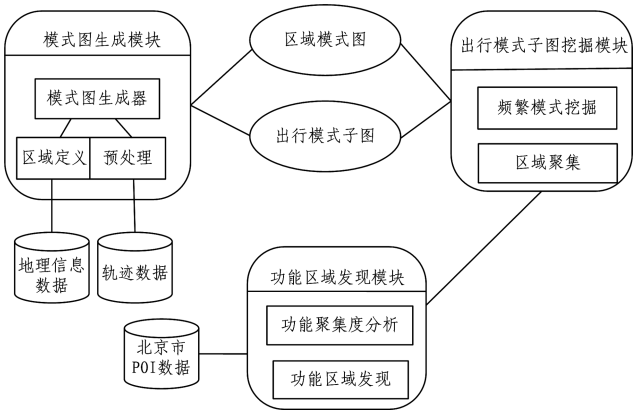

图 4 功能区域自动发现系统

Fig. 4 Automatic discovering system for functional region

由图 4 可以看出，模式图生成模块的输入数据包括两部分：地理信息数据（即 OpenStreetMap 网站五环内的地图数据）和轨迹数据（即北京市出租车在 2015 年 9 月 1 日至 30 日的所有定位及状态数据）。该模块的主要任务是完成区域模式图和出行模式子图的构建。出行模式子图挖掘模块首先对输入图结构应用频繁子图挖掘算法，得出频繁出行模式子图；然后运用图核相似度算法，计算出图与图之间的相似矩阵，并采用层次聚类算法将出行模式相似的功能区域聚类。功能区域发现模块在出行模式子图挖掘的结果上，应用北京市 POI 数据，对区域聚类结果进行相应分析与验证。

综合来说，本文使用城市 GIS 数据及 GPS 数据发现城市区域功能的过程可以归结为算法 2。

算法 2 自底而上的功能区域发现算法（Bottom-Up Functional Region Discovering，BUFRD）

输入：北京市地理信息数据，出租车轨迹数据

输出：发现的功能区域

1. 从原始 GPS 记录数据中分离轨迹数据；/* 即轨迹数据预处理（见算法 3）*/
2. 构建区域模式图（见算法 1）及出行模式子图；/* 即出行模式子图生成算法（见算法 4）*/
3. 计算出行模式子图频繁度，得到频繁出行模式子图；/* 即出行模式子图挖掘（见算法 6）*/
4. 对频繁出行模式子图聚类，得到聚类区域结果；/* 即功能区域聚类算法（见算法 9）*/
5. 根据 POI 数据及图聚类结果，判断各聚类区域的具体功能.

从算法中可以看出，第 1－2 步属于模式图生成模块的功能，具体地理信息数据的预处理如 2.1 节所示，轨迹数据预处理如算法 3 所示。第 3－4 步属于出行模式子图挖掘模块的功能，算法 6 完成频繁模式挖掘（其中算法 7、算法 8 辅助算法 6 完成设计功能），算法 9 完成区域聚类功能。第 5 步属于功能区域发现模块，具体内容参见 3.3 节。

下面展示及分析算法 2 中各模块的设计和实现。

3.1 模式图生成模块

该模块相当于本文所提方法的数据预处理模块，主要对地理信息数据和轨迹数据进行预处理，经过图生成器生成相应的区域模式图和出行模式子图，以便下一模块的进一步处理与分析。

该模块首先通过区域定义及相应算法，获得区域与栅格点以及经纬度信息与栅格点的对应关系。由于无法直接使用从 OpenStreetMap[23] 下载的数据，因此需对该数据进行预处理，获取道路信息。本文通过墨卡托投影方法将地理信息数据映射成一张二维图片，图片中每个像素点代表一个栅格数据点。由于直接映射之后的图包含过多细节信息（如图 2(a) 所示），因此需要对图像数据根据 2.1 节所述方法进行处理，最终获得各区域所包含栅格点的范围，并建立栅格点与区域标记的一一对应关系。同时建立经纬度与栅格点的对应关系，方便后续通过轨迹数据点的经纬度确定其栅格点信息。至此，完成了区域标记以及经纬度信息与区域的对应关系。

由于原始轨迹数据按时间顺序排列，同一车辆的定位数据没有连续存放，不便于分析及后续区域模式图的生成，因此需要对轨迹数据进行预处理操作，提取出同一车辆的连续轨迹数据信息。具体方法如算法 3 所示。

算法 3 轨迹数据预处理（Trajectory Preprocessing，TrajPre）

输入：按时间顺序排列的轨迹数据 D

输出：分离出的轨迹数据集 D_s

```
1. 读取轨迹数据 D，D_s=∅，Texi_n={}
2. for each n in D：
3.    If 载客状态=1 and 出租车 not in Texi_n：
4.       Texi_n[出租车]=[n]
5.    elif 载客状态=1 and Texi_n[出租车][size-1].time-n.time<120：
6.       Texi_n[出租车].append(n)
7.    elif 载客状态=1：
8.       D_s+=Texi_n[出租车]，Texi_n[出租车]=[n]
9. End for
10. Return D_s.
```

该算法遍历所有轨迹数据，首先判断载客状态，确定轨迹起点，然后通过载客状态或者时间间隔判断终点，从而得到完整的轨迹数据。其时间复杂度为 $O(n)$，其中 n 为轨迹文件的定位记录数。

将预处理后的地理信息数据和轨迹数据输入模式图生成器，生成相应的区域模式图和出行模式子图。区域模式图和出行模式子图的定义，以及区域模式图生成算法已经在第 2 节做了详细介绍。出行模式子图一般为区域模式图的子图，出行模式子图生成算法与区域模式图生成算法类似，主要区别在于出行模式子图的起始点是指定的。本方法将每个区域均作为起始点，且每个起始点对应唯一的出行模式子图，即出行模式子图个数为区域模式图中的节点个数，且等于划分区域个数。生成出行模式子图的算法如算法 4 所示。

算法 4 出行模式子图生成算法（Travel Pattern Subgraph Generate，TPSubgraph_gen）

输入：轨迹数据 T_r，所有区域点集 V，栅格点与区域对应 Hash 表

输出:从区域点出发的出行模式子图集合 Q_s

1. for each v inV:
2. trs=∅
3. for each t_r in T_r:
4. if t_r. start_node=v:
5. trs+=t_r
6. end for
7. Q_s+= RPGC (trs,Hash) /* 区域模式图构建算法(见算法 1)*/
8. end for
9. return Q_s.

该算法的具体思路为从每个点出发,遍历所有轨迹数据,通过结合从该点出发的所有轨迹,构造出行模式子图。但实验发现,该算法每生成一个出行模式子图集合 Q 均需遍历一次轨迹数据库,效率很低。本文对现有算法进行相应改进,以空间效率换取时间效率。改进后的生成算法如算法 5 所示。

算法 5　改进的出行模式子图生成算法(Enhanced Travel Pattern Subgraph Generate,ETPSubgraph_gen)

输入:轨迹数据 T_r,所有区域点集 V,栅格点与区域对应的 Hash 表

输出:从区域点出发的出行模式子图集合 Q_s

1. Tr_Hash=∅
2. for each t_r in T_r:
3. if v=t_r. start_node not in Tr_Hash:
4. Tr_Hash[v]=[t_r]
5. else:Tr_Hash[v]+=t_r
6. end for
7. for eachkey,value in Tr_Hash:
8. Q_s+=RPGC (trs,Hash) /* 区域模式图构建算法(算法 1)*/
9. end for
10. return Q_s.

改进后的算法大幅减少了扫描轨迹数据库的次数,时间复杂度从原来的 $O(m*n)$降低到 $O(n)$(其中 m 表示总区域个数,n 表示轨迹数据文件长度),降低了 I/O 消耗,提高了算法的时间效率。

3.2　出行模式子图挖掘模块

该模块是整个算法的核心模块,同时也是本文的创新模块,其主要包括两部分:频繁模式挖掘和区域聚集。

3.2.1　频繁模式挖掘

该部分的输入为区域模式图和出行模式子图,该模块首先需要对出行模式子图进行频繁度计算,即在区域模式图中找出其同构图,如果同构图数大于 10,则认为该图在区域模式图中是频繁图。

定义 8(深度优先编码(DFSCode))　给定出行模式子图 Q,其起点为 v,从起始点出发,采用深度优先遍历(Depth-First Search)算法所得到的点和边的序列即为深度优先编码。

对于出行模式子图 Q,其 *DFSCode* 为边集序列,即该图从起始点出发,经过深度优先搜索算法所经过的边的序列。获取 *DFSCode* 之后,计算出行模式子图频繁度,具体算法如算法 6 所示。

算法 6　出行模式子图挖掘(Travel Pattern Subgraph Discover,TPSubgraph_discover)

输入:区域模式图 G,出行模式子图集合 Q_s

输出:频繁出行模式子图集合 F_s

1. F_s=∅
2. for Q in Q_s:
3. 采用深度优先遍历算法获取 Q 的 DFSCode
4. freq=TF_collect(G,DFSCode)/* 出行模式频繁度计算(见算法 7)*/
5. if freq>10:
6. F_s+=Q
7. end for
8. return F_s.

在频繁子图挖掘算法中,对于每个 Q,首先获取其深度优先编码值,如算法 6 第 3 行所示。接着通过子图频繁度计算算法,获取该出行模式子图在区域模式图 G 中同构子图的数量 $freq$,若 $freq>10$,则该图为频繁图,该出行模式为频繁出行模式,具体代码如算法 6 第 4—6 行所示。子图频繁度计算算法如算法 7 所示。

算法 7　出行模式频繁度计算(Travel Frequency Collect,TF_collect)

输入:区域模式图 G,出行模式子图深度优先编码 DFSCode

输出:出行模式子图 Q 在图 G 中同构体的个数 num

1. num=0,start_node. candidate=[]
2. 对 DFSCode 出行模式子图节点按度数排序
3. 选取度数最大点作为起始查询点 start_node
4. 按照每次选度数最大的子节点规则重新获取 DFSCode′
5. for each node n in G:
6. if n. degree>start_node. degree and n. label=start_node. label
7. start_node. candidate+=n
8. end for
9. for each node n in start_node. candidate:
10. num=Iso_search (G,start_node,n,num) /* 同构体搜索算法(见算法 8)*/
11. end for
12. return num.

在出行模式频繁度计算算法中,首先获取出行模式子图 Q 度数最大的点作为起始查询点,如算法 7 第 2—3 行所示。接着按照每次选度数最大的子节点的规则重新获取出行模式子图 *DFSCode′*,从而使得每次进行匹配时从度数大的点开始。然后从区域模式图 G 中找出起始查询点的同构点,构成候选集 *start_node. candidate*,如第 5—8 行所示。最后从每个候选点出发,寻找出行模式子图同构体,获取频繁度并返回,如第 9—11 行所示。其中第 10 行的同构体搜索算法 Iso_search 如算法 8 所示。

算法 8　同构体搜索算法(Subgraph Isomorphism Search,Iso_search)

输入:区域模式图 G,当前查询点 start_node,当前出行模式子图同构点 n,频繁度 num

输出:出行模式子图当前频繁度 num

1. if start_node is last node of DFSCode′:
2. num+=1
3. return num

```
4. for each unvisited child n_c of n:
5. if n_c. degree > start_node. child. degree and n_c. label = start_node. child. label
6. num = Iso_search (G, start_node. child, n_c, num) / * 递归本算法 * /
7. end for
8. return num.
```

算法 8 主要解决同构体搜索，即给定起始匹配点，判断从该点出发是否存在出行模式子图同构体。该算法存在递归操作，首先进行退出递归条件的判断，即该查询点是否为 *DFSCode* 的最后一个点，若是，则退出递归；否则继续对该查询点的子节点进行进一步的递归判断，直到所有点均被遍历或者所有同构体全部找出为止。算法 6－算法 8 共同完成频繁子图挖掘任务，假设频繁子图数目为 N，M 表示出行模式图中点在区域模式图中的同构点数目，则算法 6 的复杂度为 O(N)，算法 7 的复杂度为 O(K)，K 表示出行模式图中的节点数目，对于算法 8，假设 n 为出行模式图中点的平均度数，m 表示区域图中点的平均度数，则算法 8 的复杂度为 O(m^n)，复杂度的具体计算过程参见文献[14]。

经过以上算法，返回所有频繁出行模式子图集合 F_s，即获得人们的频繁出行模式。接下来从频繁模式出发，进一步分析这些出行模式的相似之处，进而获得区域的聚类信息。

3.2.2 区域聚集

通过频繁模式挖掘处理之后，返回所有频繁查询子图。由于每个出行模式图代表一种出行模式，因此本文对所有频繁出行模式采用层次聚类算法，获取频繁出行模式的聚类结果。具体地，图聚类算法如算法 9 所示。

算法 9 功能区域聚类算法(Functional Region Cluster，FR_cluster)

```
输入：所有频繁子图 F_s
输出：k 个聚类簇
1. matrix = Ø / * n×n 图相似度矩阵，n = |F_s|，记录图与图之间的相似度 * /
2. adj_matrixs = Ø / * 存放所有图的邻接矩阵 * /
3. label_matrixs = Ø / * 存放所有图的标签集合 * /
4. for each g in F_s:
5. adj_matrixs + = g. adj_matrix
6. label_matrixs + = g. labels
7. end for
8. matrix = get_kernel_similarity(adj_matrixs, label_matirxs) / * 采用文献[19]提供的图核算法 * /
9. k_cluster = hierarch_clustering(matrix) / * 对相似度矩阵进行层次聚类 * /
10. return k_cluster.
```

图聚类算法的核心是获取图相似度矩阵和使用层次聚类算法。在进行相似度计算之前，需获取每个聚类图的邻接矩阵和标签列表，如算法 9 第 4－7 行所示。get_kernel_similarity 算法主要依据线性时间图核算法[19]的思想，计算图核相似度矩阵。具体方法为：首先对每个节点定义 0-1 二进制标签，然后通过一系列逻辑运算将图节点进行 hash 映射，再对映射后的标签集计算 Jaccard 系数，得出图与图之间的相似度，如算法第 8 行所示；再采用层次聚类的方法对相似度矩阵进行处理，得到聚类的最终结果，如算法第 9 行所示。

完成图聚类算法后，便完成了所有的区域聚类操作，即将功能相同或相似的区域聚集在一起。接下来进一步分析各类别区域的具体功能。

3.3 功能区域发现模块

该模块的主要任务是功能区域的分析，即根据上一模块的聚类结果，分析每个区域的所属类别。本模块的主要输入数据为北京市 POI 数据和上一节图聚类算法的返回结果。

从上一步返回的聚类结果可知每个区域的区域功能向量 FR，本文选取区域功能占比最高的功能作为该区域的最终功能。

在确定了每个区域所属类别之后，根据北京市 POI 数据，获取每个区域不同 POI 类型的数量分布，具体分析各类型区域功能。相应的分布密度(Distribution density)值的计算公式为：

$$Dd_i = \frac{POI\_num\_i}{Size\_of\_region} \tag{1}$$

其中，Dd_i 表示该类型区域第 i 类别 POI 分布密度值，其中 $Size\_of\_region$ 表示该类型区域的大小(区域包含的总栅格数)，POI_num_i 表示该类型区域中第 i 个 POI 类型的兴趣点数目。对于某类型区域，其分布密度值为向量 $Dd = (Dd_1, Dd_2, \cdots, Dd_F)$，其中 F 表示 POI 类别的数量。然后根据各类区域向量 Dd 值进行区域功能类别的分析。

4 实验结果与讨论

本节在北京市的真实数据上对比提出的城市功能区域发现算法。在比较过程中，根据算法最终划分的区域类别结果及 POI 数据来计算相应的统计值，并通过对比统计值的熵值，证明本方法划分区域的准确性；同时对比了本文划分区域的粒度，证明了该方法比传统方法考虑得更加细致。

4.1 实验数据及实验基本说明

实验使用的数据主要包括：1)OpenStreetMap[23]提供的地理地图数据；2)北京市出租车 2015 年 9 月一个月的轨迹数据集，包括时间、经纬度、载客状态等数据，数据采集时间间隔为 60 s；3)高德地图的北京市所有 POI 数据，包括经纬度、POI 类别等信息。由于高德地图和 OpenStreetMap 的数据在经纬度可以匹配，我们的 POI 数据可以精确反映指定区域的微观细化功能。相关实验均在一台 CPU 为 Intel(R) Core(TM) i5-2450M @ 2.50GHz，4GB 内存的 PC 上完成。为验证本文所提方法的有效性，将实验结果同随机确认区域所属类别方法及文献[6]所提供的结果进行对比。

实验所用到的 POI 数据由高德提供，包括整个北京市的所有兴趣点的数据，一共是 2046466 个点，每个点有 14 个属性，包含地理位置、更新时间、兴趣点类型等信息。由于兴趣点数据描述了某一位置的具体功能，如“市政府”“博物馆”等，因此可用于客观地对已划分的城市功能区域的真实功能进行验证。本实验主要考虑 POI 的类型，经统计，POI 类型主要分为 22 种(该分类方法及标记均是由高德地图提供)，具体如表 1 所列。

表 1　POI 类型

Table 1　Types of POI

ID	POI 类型	ID	POI 类型	ID	POI 类型	ID	POI 类型
1	汽车销售	7	购物服务	13	公共设施	19	餐饮服务
2	摩托车服务	8	汽车服务	14	汽车维修	20	体育休闲服务
3	医疗保健服务	9	室内设施	15	公司企业	21	交通设施
4	地名地址信息	10	住宿服务	16	科教文化服务	22	道路附属设施
5	商务住宅	11	政府机构及社会团体	17	交通设施服务		
6	风景名胜	12	金融保险服务	18	生活服务		

4.2　模型有效性验证

首先,对于出行模式子图生成算法来说,由于极大地减少了整个数据文件的遍历次数,即从原来的 m 次降低为 1 次(其中 m 表示区域模式图的所有节点数目),因此改进后的算法效率大约提高了 m 倍,具体的效率对比结果如表 2 所列。

表 2　算法效率对比

Table 2　Comparison of algorithm efficiency

算法	效率/s
改进前	34400
改进后	43

在区域发现结果分析方面,本文主要基于 POI 数据,根据每个区域内各 POI 类型的数目,计算每一类别功能区域的分布密度值 Dd。通过密度值的信息熵判断区域功能聚集程度,并将结果与文献[6]的结果进行对比,实验表明本文方法具有较好的结果。但由于文献[6]中的 POI 有 30 种类型,所以需一一对应进行比较,对应关系如表 3 所列。在表 3 的对应关系下,根据文献[6]计算区域聚集度的公式(同式(1)),每类型区域的 Dd_i 值分布如表 4 所列,其中 $C_i(i=0,1,\cdots,6)$ 表示第 i 种区域类别。由于 POI 类型地名地址信息在文献[6]中没有对应,所以在后续实验比较中忽略了该类型 POI 的相关计算和比较,仅比较其他 21 种 POI 类型。

表 3　本文与文献[6]的 POI 类型对应情况

Table 3　Corresponding situation of POI types between this paper and ref.[6]

本文	文献[6]	本文	文献[6]
汽车销售	car sales	购物服务	shopping mall/supermarket
摩托车服务	motorcycle service	汽车服务	car service
医疗保健服务	hospital	室内设施	furniture building materials market
地名地址信息	无对应	住宿服务	hotel
商务住宅	Residence	政府机构及社会团体	governmental agencies and public organizations
风景名胜	scenic spot	金融保险服务	banking and insurance service
公共设施	public utilities	餐饮服务	Chinese restaurant/foreign restaurant/fastfood restaurant
汽车维修	car repair	体育休闲服务	sports/sports/stationery shop
公司企业	corporate business	交通设施	entrance/bridge
科教文化服务	science and education	道路附属设施	street furniture
交通设施服务	transportation facilities	生活服务	living service/convenience store

表 4　功能区域的 Dd 值

Table 4　Dd values of different functional regions

	C_0		C_1		C_2		C_3		C_4		C_5		C_6	
	Dd	R	Dd	R	Dd	R	Dd	R	Dd	R	Dd	R	Dd	R
汽车销售	0.019861	3	0.025303	1	0.02196	2	0.01892	4	0.01523	6	0.01429	7	0.01692	5
摩托车服务	0.004945	4	0.003991	6	0.00341	7	0.00543	2	0.00511	3	0.00444	5	0.00661	1
医疗保健服务	0.296862	2	0.296259	3	0.30859	1	0.28131	5	0.22355	7	0.26960	6	0.29328	2
商务住宅	0.684274	5	0.813545	3	0.84953	1	0.69899	4	0.62787	6	0.53570	7	0.84231	3
风景名胜	0.114464	2	0.088938	3	0.13830	1	0.07027	5	0.06688	6	0.03016	7	0.07759	4
购物服务	3.021229	5	3.765065	3	4.10103	2	3.40398	4	2.32689	7	4.18594	1	2.78890	6
汽车服务	0.098576	4	0.127495	2	0.09703	5	0.12193	3	0.16016	1	0.07811	7	0.08751	6
室内设施	0	3	0	3	0.08829	1	0	3	0	3	0	3	0.00525	2
住宿服务	0.234928	4	0.272236	2	0.27702	1	0.22118	5	0.20425	6	0.16131	7	0.26605	3
政府机构及社会团体	0.310156	5	0.414793	2	0.42098	1	0.33611	4	0.30532	6	0.26229	7	0.38955	3
金融保险服务	0.294511	4	0.349000	3	0.41778	1	0.26782	5	0.24135	6	0.23657	7	0.39985	2
公共设施	0.340718	4	0.378269	1	0.37512	2	0.32952	5	0.21995	6	0.20990	7	0.36699	3
汽车维修	0.041506	7	0.050004	2	0.03892	6	0.05668	1	0.04164	3	0.03937	5	0.03325	7
公司企业	1.39919	5	1.860693	2	1.87532	1	1.42057	6	1.44359	4	1.42516	5	1.77505	3
科教文化服务	0.627528	4	0.724306	4	0.73650	3	0.59473	7	0.77757	1	0.61033	6	0.74584	2
交通设施服务	0.664656	5	0.754655	3	0.82404	2	0.70140	4	0.64939	6	0.59254	7	0.85592	1
生活服务	1.940545	4	2.271947	2	2.27657	1	1.87967	5	1.65272	6	1.51788	7	2.22645	3
餐饮服务	2.372543	4	2.867477	2	2.78371	3	2.32443	5	2.09972	6	2.03263	7	2.89567	1
体育休闲服务	0.31421	4	0.383314	2	0.37438	3	0.30411	5	0.27392	6	0.23339	7	0.44050	1
通行设施	0.298969	4	0.34054	3	0.34836	2	0.29794	5	0.28067	6	0.21117	7	0.36485	1
道路附属设施	0.001783	4	0.002636	3	0.00287	2	0.00167	5	0.00127	6	0.00127	7	0.00350	1

表 4 给出了每个类型区域的 POI 类型分布密度(Dd 值),R 值代表该类型 Dd 值在所有区域类型值中的 $rank$ 排序值。如:C_0 的汽车销售 Dd 值为 0.019861,$rank$ 值为 3,说明 C_0 的汽车销售 Dd 值在所有区域类型 C 中的排序值为 3。

定义 9(城市区域功能纯度熵)　为分析区域功能聚集度,本文从 Dd_i 值分布的纯度出发,分析各类区域的具体熵值(Ent),计算公式如下:

$$Ent(C_i)=-\sum_{k=1}^{21}p_k * \log p_k \tag{2}$$

其中,p_k 表示第 k 类 POI 类型的 Dd_i 值在该类型区域所有

Dd_i 值总和中所占的比例，log 是指取底为 2 的对数函数。熵值反映了 C_i 类型区域的 POI 分布纯度。本文还将本算法的结果同随机划分方法和文献[6]所提供的结果在城市区域功能纯度熵上进行了相应比较(随机划分方法是指随机给每个区域规定所属类别)，具体结果如图 5 所示。

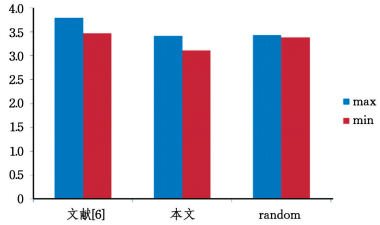

图 5 各方法区域聚集度的最大/最小熵值

Fig. 5 Max/min entropy value of region cluster degree of different methods

从图 5 中可以看出，本文方法表现出最低的熵值，即本文的区域聚集方法的区域聚集度最高。

图 6 显示了聚集之后的区域，颜色相同的区域代表同一功能区域，不同颜色对应不同的功能区域。通过对比图 6 和图 7(图 6 是北京市五环内区域划分结果，图 7 是包括六环外一部分区域的区域划分结果)可以看出，在两图中五环内的区域，图 6 所包含的区域数明显多于图 7，即本文的区域划分粒度更加细致。

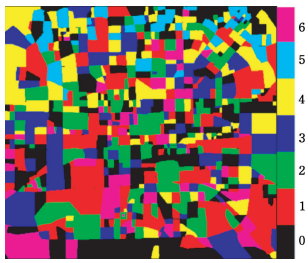

图 6 区域发现结果

Fig. 6 Results of region discovery

图 7 文献[6]的功能区域图

Fig. 7 Function region graph from ref. [6]

4.3 功能类型标识

文献[6]最终将区域分为 9 个类别，分别为：Diplomatic and embassy areas(外交与大使馆区域)、Education and science areas(教育科技区域)、Developed residential areas(发达居民区)、Emerging residential areas(正在形成中的居民区)、Developed commercial/entertainment areas(发达商业休闲娱乐区)、Developing commercial/business/entertainment areas(发展中的商业休闲娱乐区)、Regions under construction(正在建设中的区域)、Areas of historic interests(历史景区)、Nature and parks(自然和公园)。为进一步分析本文算法功能区域识别结果的实际意义，我们根据 GIS 地图各不同的实际区域(如天安门、天坛公园等)范围，在所得区域功能分析结果的基础上做了进一步确定。具体的分析结果如下。

对于本文来说，将地理区域分为 7 个类别(C_0－C_6)。从表 4 可以看到各 POI 类别在每种功能区域的 *Dd* 值占比及排名情况，从排名情况可以看出：C_1，C_2 和 C_6 区域的大多数 POI 类型的 *rank* 值较高，可以判断这些区域相对较为发达，同时从图 6 中也可以看出相应的地理区域位置。具体的每个聚类区域功能的分析如下。

旅游景区(C_0)：从表 4 中的 POI 分布来看，该类型区域的各个排名都处于中等水平，从图 6 中可以看出，天安门、故宫博物院、奥体中心等旅游景区属于该类型区域，因此可以认为该区域为旅游景区。

商业休闲娱乐一体化区域(C_1)：从表 4 中可以看出该类型区域各类型 POI 排名都较高，尤其是餐饮、休闲、金融服务等，形成了工作娱乐一体的商业区模式。从图 6 中可以看出，金融街、各办公大楼等均在该类型区域内。

公园风景区(C_2)：从表 4 中可以看出，该功能区域的风景名胜、住宅等 POI 类型占比较高，同时从图 6 中可以看到，天坛公园、玉渊潭公园等都在该类型功能区域内。

教育科技区(C_4)：从表 4 中可以看出，该区域的科教文化等排名较高，同时从图 6 中可以看出该区域包括了中关村、人民大学等。

发达居民区(C_6)：从表 4 中可以看出，该区域的各 POI 类别排名都很靠前，各种住宿设施、生活设施均齐全，满足人们日常生活的需求。

C_3 与 C_5 比较类似，所有的 POI 类型占比都比较低，但是 C_3 的汽车服务比较多，可能是属于 4S 店或者加油站之类的地区，C_5 的购物服务较高，可能是正在发展的区域。

结束语 本文首次结合车辆行驶轨迹数据及城市 GIS 数据尝试发现城市中细粒度的功能区域划分。具体来说，我们将相关数据预处理为区域模式图，并设计了区域模式图构建算法的整体框架及其内部的各种处理过程。其中，频繁出行模式子图挖掘算法基于单一图结构，采用自底而上的方法，挖掘出频繁的出行模式子图。本文提出频繁出行模式子图挖掘算法，根据图核方法计算出行模式之间的相似度，依据层次聚类法对频繁出行模式聚类，从而依据人们的出行目的，分析出各聚类簇区域的功能。实验中，根据 POI 数据计算聚类区域的 POI 分布 *Dd* 值，并计算相应熵值，与文献[6]的结果进行对比，熵值下降了 10%，表明空间聚集度更高，验证了本文所提算法的有效性。

图 7 是文献[6]所提算法划分的最终的区域标记图，从区域划分的粒度可以看出，本文的划分粒度更细，考虑的功能区域更精确。如图 7 中功能区域 C_0，文献[6]指出其属于外交与大使馆区域，但该区域包含有其他功能区域，如日坛公园和团结湖公园等。因此，从粒度上来看，文献[6]所划分的粒度过于粗略。本文从更加细致的粒度出发，定义了更加细致的功能区域。

文献[2]在文献[6]的基础上进行了相应的改进，并在原有出租车轨迹数据的基础上，增加了城市居民出行的公交车刷卡数据，从定位数据和迁移数据相结合的角度实现了功能

区域发现,其具体功能区域图如图 8 所示,其区域划分粒度与文献[6]相同。通过与文献[2]进行对比,可以得出本文的区域划分粒度依然比其细致,且区域功能聚集度依然更优。

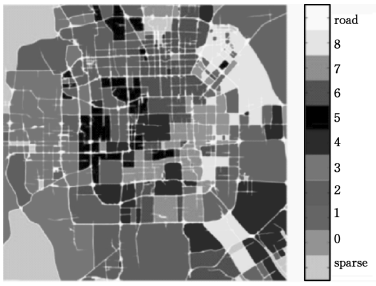

图 8　文献[2]的功能区域图

Fig. 8　Function region graph from ref. [2]

由于本文在研究出行模式时没有考虑具体时段,后续工作可以从时间细化的角度进一步完善该领域的研究;同时本文分析的北京市地理范围有限,没有考虑整个城市的所有数据,后续可以将该算法扩展到整个北京市范围的地理区域;本文只得出粗略的城市功能区域划分图,并没有将该图与真实的北京市地图相结合,给出直观的可视化结果;本文目前仅将该方法与文献[6]和文献[2]进行了相应的对比,后续将与城市规划领域专家进行相关讨论,联合城市规划领域的研究工作做进一步分析,进而得出更符合实际意义的结论。本文未来的工作还包括与北京市市政规划研究的专家合作,进一步梳理区域的真实意义并尝试发现城市功能区域演化相关的动态规律。

致谢　由衷感谢中央财经大学政府管理学院魏海涛老师在整篇论文工作过程中给予的关于城市规划领域的建议和支持!

参 考 文 献

[1] FURLETTI B,CINTIA P,RENSO C,et al. Inferring human activities from GPS tracks[C]// Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing. ACM, 2013.

[2] YUAN N J,ZHENG Y,XIE X,et al. Discovering urban functional zones using latent activitytrajectories[J]. IEEE Transactions on Knowledge and Data Engineering, 2015, 27(3): 712-725.

[3] ZHENG Y,LIU Y,YUAN J,et al. Urban computing with taxicabs[C]// Proceedings of the 13th International Conference on Ubiquitous Computing. ACM,2011:89-98.

[4] HUANG X,ZHAO Y,MA C,et al. TrajGraph: A graph-based visual analytics approach to studying urban network centralities using taxi trajectory data[J]. IEEE Transactions on Visualization and Computer Graphics,2016,22(1):160-169.

[5] ZHENG Y,CAPRA L,WOLFSON O,et al. Urban computing: concepts, methodologies, and applications[J]. ACM Transactions on Intelligent Systems and Technology (TIST),2014, 5(3):38.

[6] YUAN J,ZHENG Y,XIE X. Discovering regions of different functions in a city using human mobility and POIs[C]// Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM,2012:186-194.

[7] YAJING X,GONGFU L,CHAO X,et al. Affinity-based human mobility pattern for improved region function discovering[J]. The Journal of China Universities of Posts and Telecommunications,2016,23(1):60-67.

[8] DEZANI H,BASSI R D S,MARRANGHELLO N,et al. Optimizing urban traffic flow using Genetic Algorithm with Petri net analysis as fitness function[J]. Neurocomputing,2014,124:162-167.

[9] TANG J,LIU F,WANG Y,et al. Uncovering urban human mobility from large scale taxi GPS data[J]. Physica A: Statistical Mechanics and its Applications,2015,438:140-153.

[10] PAN G,QI G,WU Z,et al. Land-use classification using taxi GPS traces[J]. IEEE Transactions on Intelligent Transportation Systems,2013,14(1):113-123.

[11] ZHENG Y. Trajectory data mining:an overview [J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2015,6(3):1-41.

[12] YAN X,HAN J. gspan:Graph-based substructure pattern mining[C]// IEEE International Conference on Data Mining(ICDM 2003). IEEE,2002:721-724.

[13] ELSEIDY M,ABDELHAMID E,SKIADOPOULOS S,et al. Grami:Frequent subgraph and pattern mining in a single large graph[J]. Proceedings of the VLDB Endowment,2014,7(7): 517-528.

[14] HaN W S,LEE J,LEE J H. Turbo iso:towards ultrafast and robust subgraph isomorphism search in large graph databases[C]// Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data. ACM,2013:337-348.

[15] VISHWANATHAN S V N,BORGWARDT K M,KONDOR R,et al. Graph Kernels[J]. Journal of Machine Learning,2008, 11(2):1201-1242.

[16] FOUSS F,FRANCOISSE K,YEN L,et al. An experimental investigation ofkernels on graphs for collaborative recommendation and semisupervised classification[J]. Neural Networks, 2012,31(none):53-72.

[17] GAÜZERE B,BRUN L,VILLEMIN D. Two new graphs kernels in chemoinformatics[J]. Pattern Recognition Letters,2012, 33(15):2038-2047.

[18] BARRA V,BIASOTTI S. 3D shape retrieval using kernels on extended Reeb graphs[J]. Pattern Recognition,2013,46(11): 2985-2999.

[19] HIDO S,KASHIMA H. A linear-time graph kernel[C]// 9th IEEE International Conference on Data Mining. 2009:179-188.

[20] ZHANG T Y,SUEN C Y. A fast parallel algorithm for thinning digital patterns[J]. Comm Acm,1984,27(3):236-239.

[21] FIORIO C,GUSTEDT J. Two linear time Union-Find strategies for image processing[J]. Theoretical Computer Science,1996, 154(2):165-181.

[22] WU K,OTDO E. Optimizing connected component labeling algorithms[C]// Medical Imaging:Image Processing. International Society for Optics and Photonics. 2005.

[23] OpenStreetMap Foundation. Beijing 3th ring osm data [DB/OL]. [2016-06-23]. http://www.openstreetmap.org.

[24] CASTRO P S,ZHANG D,CHEN C,et al. From taxi GPS traces to social and community dynamics: A survey[J]. Acm Computing Surveys,2013,46(2):1-34.

[25] SNYDER,JOHN P. Map projections[M]. Springer Netherlands. 1997.

[26] ROSENFELD A, DAVIS LS. A Note on Thinning[J]. IEEE Transdactions on Systems, Man and Cybernetics, 1976, SMC-6 (3): 226-228.

[27] YAN C, WANG P, SUN L. Sensing Urban with Wi-Fi and Satellite: Functional Region Discovery across Cities[C]// On Thematic Workshops of Acm Multimedia. ACM, 2017: 314-322.

[28] LIU X, GONG L, GONG Y, et al. Revealing travel patterns and city structure with taxi trip data[J]. Journal of Transport Geography, 2015, 43: 78-90.

[29] FENG Z, ZHU Y. A Survey on Trajectory Data Mining: Techniques and Applications[J]. IEEE Access, 2017, 4: 2056-2067.

[30] YU Y, CHEN X. A survey of point-of-interest recommendation in location-based social networks[C]// Twenty-Ninth AAAI Conference on Artificial Intelligence. 2015.

[31] LIU X P, HE J L, YAO Y, et al. Classifying urban land use by integrating remote sensing and social media data[J]. International Journal of Geographical Information Science, 2017(1): 1675-1696.

(上接第 254 页)

[3] GREEN R, EASTWOOD M L, SARTURE C M, et al. Imaging spectroscopy and the airborne visible/infrared imaging spectrometer(AVIRIS)[J]. Remote Sensing of Environment, 1998, 65(3): 227-248.

[4] LEE D D, SEUNG H S. Learning the parts of objects by non-negative matrix factorization[J]. Nature, 1999, 401(6755): 788-791.

[5] TONG L, ZHOU J, QIAN Y T. Nonnegative Matrix Factorization Based Hyperspectral Unmixing with Partially Known Endmembers[J]. IEEE Transactions on Geoscience and Remote Sensing, 2016, 54(11): 6531-6544.

[6] YU Y, GUO S, SUN W D. Minimum distance constrained nonnegative matrix factorization for the endmember extraction of hyperspectral images[C]// Proceeding of Remote Sensing and GIS Data Processing and Applications. Wuhan, 2007: 6790151-6790159.

[7] JIA S, QIAN Y T, JI X, et al. Hyperspectral Unmixing Algorithm Based on spectral and spatial characteristics[J]. Journal of Shenzhen University (Science & Engineering), 2009, 26(3): 162-167. (in Chinese)
贾森, 钱沄涛, 纪霞, 等. 基于光谱和空间特性的高光谱解混方法[J]. 深圳大学学报(理工版), 2009, 26(3): 162-167.

[8] YANG S Y, ZHANG X T, YAO Y G, et al. Geometric Nonnegative Matrix Factorization (GNMF) for Hyperspectral Unmixing [J]. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sening, 2015, 8(6): 2696-2703.

[9] WANG W H, QIAN Y T, TANG Y Y. Hypergraph-Regularized Sparse NMF for Hyperspectral Unmixing[J]. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sening, 2016, 9(2): 681-694.

[10] YUAN Y, FU M, LU X Q. Substance Dependence Constrained Sparse NMF for Hyperspectral Unmixing[J]. IEEE Transactions on Geoscience and Remote Sensing, 2015, 53(6): 2975-2986.

[11] ADAMS J B, SABOL D E, KAPOS V, et al. Classification of multispectral images based on fractions of endmembers: Application to land-cover change in the Brazilian Amazon[J]. Remote Sensing of Environment, 1995, 52(2): 137-154.

[12] ADAMS J B, SMITH M O, JOHNSON P E. Spectral mixture modeling: of rock and soil types at the Viking Larder 1 site[J]. Journal of Geophysical Research: Solid Earth (1978-2012), 1986, 91(8): 8098-8112.

[13] DIAS J B, PLAZA A. Hyperspectral unmixing geometrical, statistical and sparse regression-based approaches[C]// Proceedings of SPIE: Image and Signal Processing for Remote Sensing XVI. Toulouse, France: SPIE Press, 2010.

[14] ZHAO C H, CHENG B Z, YANG W C. A hyperspectral unmixing algorithm based on the constraint nonnegative matrix decomposition[J]. Journal of Harbin Institute of Technology, 2012, 33(3): 378-382. (in Chinese)
赵春晖, 成宝芝, 杨伟超. 利用约束非负矩阵分解的高光谱解混算法[J]. 哈尔滨工业大学学报, 2012, 33(3): 378-382.

[15] SONG Y G, WU Z B, WEI Z H, et al. Survey of sparsity constrained hyperspectral unmixing[J]. Journal of Nanjing University of Science and Technoloogy, 2013, 37(4): 486-492. (in Chinese)
宋义刚, 吴泽彬, 韦志辉, 等. 稀疏性高光谱解混方法研究[J]. 南京理工大学学报, 2013, 37(4): 486-492.

[16] KONG F J, BIAN C D, LI Y S, et al. Hyperspectral unmixing method for non-convex and low rank constraints[J]. Journal of Xi'an Electronic and Science University (Natural Science Edition), 2016, 43(6): 116-121. (in Chinese)
孔繁锵, 卞陈鼎, 李云松, 等. 非凸稀疏低秩约束的高光谱解混方法. 西安电子科技大学学报(自然科学版), 2016, 43(6): 116-121.

[17] WANG T C, LIU X Z, DONG Z Z, et al. An adaptive robust minimum volume hyperspectral unmixing algorithm[J]. Journal of Automation, 2017, 43(2): 1-19. (in Chinese)
王天成, 刘相振, 董泽政, 等. 一种自适应鲁棒最小体积高光谱解混算法[J]. 自动化学报, 2017, 43(2): 1-19.

[18] LIU H, WU Z, CAI D, et al. Constrained non-negative matrix factorization for image representation[J]. IEEE Transactions Pattern Analysis and Machine Intelligence, 2012, 34(7): 1299-1311.

[19] SHU Z Q, ZHAO C X. Constrained nonnegative matrix decomposition algorithm based on graph regularization and its application in image representation[J]. Pattern Recognition and Artificial Intelligence, 2013, 26(3): 300-306. (in Chinese)
舒振球, 赵春霞. 基于图正则化的受限非负矩阵分解算法及其在图像表示中的应用[J]. 模式识别与人工智能, 2013, 26(3): 300-306.

[20] PLAZA A, MARTINEZ P, PEREZ R, et al. A quantitative and comparative analysis of endmember extraction algorithms from hyperspectral data[J]. IEEE Geoscience and Remote Sensing Letters, 2004, 42(3): 650-663.

[21] KESHAVA N, MUSTARD J F. Spectral unmixing[J]. IEEE Signal Process Mag, 2002, 19(1): 44-57.

[22] LANDGREBE D. Multispectral data analysis: a signal theory perspective[D]. West Lafayette: Purdye University, 1998.

[23] SWAYZE G. The hydrothermal l and structural history of the cuprite mining district, southwestern Nevada: an integrated geological and geophysical approach[D]. Boulde: University of Colorado, 1997.