

基于网络的时空同现模式挖掘算法

张永梅¹ 郭 莎¹ 季 艳² 马 礼¹ 张 睿³

(北方工业大学计算机学院 北京 100144)¹ (北京遥感信息研究所 北京 100011)²

(太原科技大学计算机科学与技术学院 太原 030024)³

摘 要 大多数数据库都不能有效地处理数据的时间维度,时空同现模式挖掘有利于提取隐含在时空数据集中有价值的信息,目前已经成为研究热点。针对现有同现模式发现方法挖掘效率较低的问题,采用双层网络对时空数据进行初始化建模,针对传统方法在进行时空兴趣度计算时未考虑对象类型存在有效周期的问题,改进了现有兴趣度计算方法,引入了权重特征值,并提出了基于网络的时空同现模式挖掘算法。实验表明,在使用不同数据量的测试集中挖掘同现模式集时,新算法的运行效率优于不对数据集进行建模的方法以及仅对实例层进行建模的方法。

关键词 同现模式,时空关系网络,时空兴趣度,有效周期

中图法分类号 TP399 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2018.03.035

Spatial-Temporal Co-occurrence Pattern Mining Algorithm Based on Network

ZHANG Yong-mei¹ GUO Sha¹ JI Yan² MA Li¹ ZHANG Rui³

(College of Computer Science and Technology, North China University of Technology, Beijing 100144, China)¹

(Beijing Institute of Remote Sensing, Beijing 100011, China)²

(School of Computer Science and Technology, Taiyuan University of Science and Technology, Taiyuan 030024, China)³

Abstract Most databases cannot effectively deal with time dimension of data, the spatial-temporal co-occurrence pattern mining is helpful to extract implicit valuable information from large spatio-temporal dataset, and it has become a hot research topic at present. To overcome lower mining efficiency of current co-occurrence pattern discovery methods, a double-level network model was used to initialize spatio-temporal dataset. In the calculation of spatial-temporal interestingness, traditional methods ignore the fact that every object-type has effective lifecycle. Thus, the current computation of interestingness was improved in this paper. We introduced weight eigenvalue and proposed a new spatial-temporal co-occurrence pattern mining algorithm based on network. Experiment results show that the proposed algorithm is more effective to calculate co-occurrence patterns in test sets with different data volumes than the methods without modeling or modeling instance layer only.

Keywords Co-occurrence pattern, Spatial-temporal relation network, Spatial-temporal interestingness, Effective lifecycle

1 引言

随着时空数据的海量生成,时空数据挖掘已经引起了国内外学者的广泛关注^[1]。时空同现模式发现是当前时空数据领域的关键技术之一,旨在从包含多个时空对象类型的时空数据集中发现在邻近位置频繁同现且在连续时间槽内频繁共现的类型集合,给出时空对象间比较典型的同现规律。时空同现模式挖掘在现实应用中非常有意义,如根据商家的交易数据,分析在不同时间段客户交易的不同商品类型同现的时空频繁程度,挖掘商品类型的时空同现模式,利用这些模式就可以在不同时空领域指定更合理的推销策略等;或者从大量

社交网络中,发现社交群体在既定时间段内在各社交平台之间隐含的同现模式,同现模式反映了社交对象的一些共现规律,可以利用这些规律分析不同类型的社交群体对于某些事件的舆情等。时空同现模式在气象研究、网络舆情、体育赛事、位置推荐服务等领域也具有很高的应用价值^[2]。

由于时空数据的复杂性,目前时空同现模式挖掘方法的研究都是在空间同位模式挖掘的基础上进行探索的,将时空数据库离散化为一系列时间槽上的空间数据库,给空间同位模式加上时间维度,使其拓展到时空数据库中,从而生成时空同现模式。

现有的时空同现模式挖掘可以简化为时空维下的关联规

到稿日期:2016-11-20 返修日期:2017-04-04 本文受国家自然科学基金项目(61371143),北方工业大学基于内容感知的最优图像缩放技术研究与应用科研平台(XN054),北方工业大学优势学科项目(XN078),太原科技大学校博士科研启动基金(20162036)资助。

张永梅(1967—),女,博士,教授,CCF会员,主要研究方向为数据挖掘、图像处理,E-mail:zhangym@ncut.edu.cn;郭莎(1992—),女,硕士生,主要研究方向为数据挖掘、图像处理;季艳(1975—),女,博士,高级工程师,主要研究方向为数据挖掘、图像处理;马礼(1968—),男,博士,教授,主要研究方向为高性能计算、无线传感器网络;张睿(1987—),男,博士,主要研究方向为动态测试技术与智能仪器、智能信息处理。

则发现,为了提高关联规则的挖掘效率,国内外学者给出了大量的研究思路^[3-6]。在时空同现模式挖掘方面,Yoo等人^[7]基于邻近和实例查找方式生成时空同位模式,将传统数据挖掘方法用于划分好的每个时间片,再基于每个时间片生成时空同位模式。HUANG等人^[8]曾提出基于连接的时空序列模式挖掘方法,虽然该方法并不针对时空同现模式的挖掘,但它将时间维度作为一个空间维度来处理,经过改进后可用于时空同现模式挖掘。CELIK等人提出了混合类型的时空同现模式的概念,依据同现模式在足够多的时间片内要保持一定的频繁性的特点,给出了时空兴趣度的计算方法,并设计了同现模式的挖掘算法,后来他们重新定义了模式的时间频繁度的计算方法,提出了一种局部时空同现模式的挖掘算法^[9]。PILLAI等人^[10]将同现模式的挖掘扩展到对时空事件的同现模式挖掘上,提出了时空同现规则。国内学者田晶等人^[11]将欧氏空间中的方法扩展到网络空间,利用同现关系来计算和推断同现模式。王占全等^[12]提出了一种基于时间汇总图的同现模式挖掘方法,该方法对时空实例进行时空关系建模,采用top-k%策略解决时间阈值的预先设定问题。

目前,针对时空同位模式的挖掘方法已经有了许多相关研究^[13-14],大多方法基于连接和阈值驱动。这种方式需要多次访问数据库来建立实例集,容易导致候选集增多,使时空兴趣度的计算量增大,同时使挖掘效率降低。另外,传统方法多采用统一的有效周期进行时空兴趣度计算,挖掘结果的有效性低,而且计算出的时空频繁度可能是无效的。因此,在筛选空间同位模式及发现时空同现模式的过程中,现有方法主要存在如下两个问题。

(1)缺乏有效的时空数据建模方式,同现模式的计算效率比较低。现有方法多利用连接方法来生成候选集,很少对时空数据集中的时空对象之间进行有效建模,没有很好地表示时空实例间的关联关系,而且由于直接采用连接的方式产生了大量重复或无用的计算,降低了计算效率,影响了时空同现模式的挖掘效率。

(2)计算时空兴趣度时忽略了时空对象的有效周期。在现有方法对时空对象间距及时空兴趣度的计算方法中,采用统一的时间框架作为时空对象的有效周期,而实际上对象的有效周期可能是不一致的,采用传统方式可能会带来大量无效计算,影响结果的有效性。

为提高同现模式的计算效率,本文对时空数据集进行初始化建模,建立了一种可以有效表示对象实例及对象类型之间时空关系的时空网络。为挖掘出更有效的时空同现模式集,本文重新设定了时空对象间的时空距离及相关时空兴趣度的计算方法。实验表明,本文提出的时空同现模式挖掘方法提高了挖掘方法的计算效率和时空同现模式的有效性。

2 基于网络的时空同现模式挖掘算法

2.1 时空兴趣度的相关计算

在时空数据挖掘领域中,表现显著的时空同现模式与其他同现模式相比,具有更高的频繁度值,即时空兴趣度较高的同现模式具有较高的时间频繁度和空间频繁度^[15]。本文首先从时空数据集的空间维度来分析模式的频繁性,得出时空同位模式后,对时空数据的时间维度也进行频繁性分析,引入

模式的权重特征值作为挖掘有趣时空同现模式的标准,从而获得时空同现模式。相关时空计量度的定义及计算方式如下。

1)时空距离。在时空同现模式挖掘过程中,首先需要计算时空对象的邻近关系。在计算任意两个或多个时空对象的距离时,欧氏距离是最常用的一种距离表示法。欧氏距离是指在n维空间中两个时空对象之间的真实距离,以点 $x=(x_1, \dots, x_n)$ 和 $y=(y_1, \dots, y_n)$ 为例,两者之间的距离如式(1)所示:

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2} \\ = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (1)$$

在现实生活中,尤其是在移动目标存在较多的道路网中,由于路网多以网格结构形成,并且存在大量建筑物及障碍物,两个移动目标之间的实际距离(从一个对象的位置到达另一个对象的位置所行走的路程)大多不是直线距离,因此在计算两个对象之间的实际距离时,使用欧氏距离表示时空对象的距离就显得不太合理。因此,本文使用另外一种距离表示法——曼哈顿距离。曼哈顿距离可以很好地表示实际道路距离,在欧几里得空间的固定直角坐标系上,两点在标准坐标系上的绝对轴距总和也可以表示为两个点所形成的线段对轴产生的投影的距离总和。以n维空间中点 $x=(x_1, \dots, x_n)$ 和 $y=(y_1, \dots, y_n)$ 为例,其曼哈顿距离计算公式如式(2)所示:

$$d(x, y) = |x_1 - y_1| + |x_2 - y_2| + \dots + |x_n - y_n| \\ = \sum_{i=1}^n |x_i - y_i| \quad (2)$$

使用曼哈顿距离进行计算时,为避免丢失使用欧氏距离计算所得的挖掘结果,将其邻近距离阈值的值设置为原欧氏阈值的 $\sqrt{2}$ 倍,容易证明:在直角三角形中,若斜边确定,两条直角边在相等时二者之和最大,和为斜边的 $\sqrt{2}$ 倍。因此在使用曼哈顿距离进行计算时,针对同一数据集,在实际测试时设置邻近距离的值为使用欧氏距离设置的邻近阈值的 $\sqrt{2}$ 倍。

曼哈顿距离不仅可以更真实地表示时空距离,而且在计算时采用加减法,其计算速度要比以乘法运算为主的欧氏距离的计算速度快很多,因此本文采用曼哈顿距离作为时空对象间的距离。

2)模式支持度。模式支持度是指构成模式的所有对象类型对该模式的支持程度,由于每一对象类型在时空数据集中具有很多对象实例,参与到模式中的实例数与该类型的实例数总和之比就是该对象类型的模式支持度,计算方式如式(3)所示:

$$PS = \frac{I}{A} \quad (3)$$

其中,PS(Pattern Support)是模式支持度,I表示支持当前模式且处于有效周期内的对象实例数,A表示该对象类型的总实例数。

3)模式的时空频繁度。在模式中所有类型都有效的时间槽下,模式的时空频繁度是模式中时空对象实例处于邻近关系的频繁程度,模式中各对象类型的实例在时间槽下处于邻近关系的频繁程度表征了该模式在当前时间槽的时空频繁程度。假定某一模式p有k个元素,即 p_0, p_1, \dots, p_{k-1} ,模式p的时空频繁度就是包含在模式中的所有时空对象类型的模式

支持度的最小值,如式(4)所示:

$$PS(p) = \min\{PS(p_i)\} \quad (4)$$

其中, $PS(p)$ 表示模式 p 的空间频繁度, $PS(p_i)$ 表示元素 p_i 的模式支持度, $0 \leq i \leq k-1$ 。

4)模式的时间频繁度。时间频繁度是指同现模式在时间上的频繁同现程度。模式 p 中所有类型都有效的时间槽片段为该模式的时间框架,该时间框架就是该模式的有效时间槽段,也即求出所有元素的元素周期交集,若在该交集内所有元素都是有效的,则该模式处于有效计算状态。假定模式 p 中有 n 个元素 p_0, p_1, \dots, p_{n-1} ,且各元素对应的时间框架为 T_0, T_1, \dots, T_{n-1} ,则该模式 p 的时间频繁度的计算方式如式(5)所示:

$$TP(p) = \frac{\text{模式 } p \text{ 同现的所有时间槽}}{T_0 \cap T_1 \dots \cap T_{n-1}} \quad (5)$$

其中, $TP(p)$ (Time Support of Pattern)是模式 p 的时间频繁度, T_i 表示 p 中某一元素 p_i 所对应的时间框架。

5)模式的权重特征值。时空同现模式不仅表示模式中各元素的时空邻近关系,也表示这种关系的时空频繁程度,包括时间维度和空间维度。因此,为了简化计算,在传统的同现模式挖掘过程中通常预先设定了表示时间频繁最低程度的时间阈值以及表示空间频繁最低程度的空间阈值,而这两个阈值处于不同的领域时,需要各领域的专家通过多次实验才能给出较合理的阈值,较大或较小的阈值参数都会引起高估或低估时空同现模式的问题。为了便于解决这个问题,本文定义了模式的权重特征值,该特征值将同现模式在各个活动时间框架下的空间频繁度和时间频繁度作为特征。具体描述如下:假定模式 p 中有 n 个元素 p_0, p_1, \dots, p_{n-1} ,且各元素对应的时间框架为 T_0, T_1, \dots, T_{n-1} ,则同现模式 p 的时间框架为 T_0, T_1, \dots, T_{n-1} 的交集,设该交集的各时间槽为: TF_0, \dots, TF_w 。若在该时间槽段内,模式 p 在 m 个时间槽内出现, $0 \leq m \leq w-v+1$,则各时间槽的空间频繁度也就是模式的各空间特征值,记为 $PS(p)_i, 0 \leq i \leq m-1$ 。同现模式的权重特征值计算方式如式(6)所示:

$$\begin{aligned} W(p) &= \frac{\sum_{i=0}^{m-1} PS(p)_i}{w-v+1} \\ &= \frac{\sum_{i=0}^{m-1} PS(p)_i}{m} \times \frac{m}{w-v+1} \\ &= PS(p)_i \times TP(p) \end{aligned} \quad (6)$$

其中, $W(p)$ 表示模式 p 的权重特征值,该特征值将同现模式 p 的空间频繁度分散到时间框架内的各个时间槽中,这样更能表示同现模式的实际空间频繁程度。从式(6)的最终结果可以看到,该值等于 p 频繁的所有时间槽的空间频繁度的均值与 p 时间频繁度之积。若同现模式的空间频繁度均值一定,则权重特征值随着模式的时间频繁度单调递增;若时间频繁度一定,则该值随着模式的空间频繁度均值单调增加。

2.2 时空数据集建模方法

时空数据集由于其自身特性而构成了一个大的时空网络。随着时间的推移,时空网络的拓扑关系和相关属性信息不断变化,在交通路网规划、气象推测、群体转移、最佳路径选择等领域都用到了时空网络。针对大量的时空数据,需要建

立查询高效的时空网络模型。现有时空网络大多采用时间拓扑图对时空数据集中的每个时间槽分别建立空间网络,但这种建模方法造成了大量节点信息的重复。随着时间槽数量的增多,建立时间拓扑图的时间消耗明显增大,而且在读取多个时间槽的时空网络时耗费的时间非常多。George 等人针对在 TEG 中需要为每个时间槽建立空间网络的缺点,提出一种更有效的时空网络建模方式——时间汇总图,解决了对每个时间槽重复建模的问题。

本文借鉴时间汇总图的模式,提出一种双层网络建模方式,在初始化过程中对时空数据集进行网络建模。该双层网络分别对时空对象的实例之间以及时空对象的类型之间建立关系网络。为方便计算,本文对所有实例对象及元素类型进行编号,双层网络分别采用邻接矩阵的方式进行存储,由于每层网络都是无向图,其存储在邻接矩阵中的信息是对称的,因此只存储上三角(或下三角)矩阵信息即可保留全部的同现信息,建模伪代码如算法 1 所示。

1) 双层网络建模的伪代码

算法 1 双层网络建模

输入:SD,时空数据集;TP,元素及元素有效周期列表;Dis,元素对象之间的邻近距离

输出:实例网络层及元素网络层

伪代码:

1. 遍历元素及元素周期列表,计算总体时间框架
2. 创建初始实例网络层的邻接矩阵 IM,并初始化矩阵为 -1
3. For 每个时间槽 in 元素列表 TP
4. 计算当前时间槽下有效时空对象之间的距离 d
5. IF $d \leq \text{Dis}$ THEN 将 IM 中对应时间槽位设为 1
6. ELSE 将 IM 中当前时间槽位设为 0
7. Return 实例网络层
8. 创建初始元素网络层邻接矩阵 TM,并初始化矩阵为 -1
9. For 每个时间序列 ts in 实例网络层
10. IF ts 的某个时间位等于 1 THEN 将 TM 中对应位设为 1
11. IF ts 的某个时间位等于 0
12. THEN IF TM 中当前位等于 1 THEN 保留原值
13. ELSE 将 TM 中对应位设为 0
14. Return 元素网络层

2) 具体建模方法

①如算法 1 所示,对时空数据集中各对象类型进行一次建模,形成实例网络层;

②从实例网络层出发,根据实例的类型属性形成元素网络层,两层网络之间的类型和类型的实例之间存在映射关系;

③在建立实例网络的过程中,以类型实例作为网络中的结点,结点之间在首次出现邻近时进行连接,结点之间的边用序列表示结点之间在各时间槽是否邻近,实例之间若满足邻近关系则设为 1,不满足邻近关系则设为 0,边连接的两实例不处于有效周期内则设为 -1。在元素网络层,结点类型之间在首次出现同现时连接,两个元素类型若在某一时间槽内同位,则对应时间槽标记为 1,不同位则标记为 0;若某一时间槽不在两个类型构成的模式的时间框架内则设为 -1。给定一个仿真数据集,如图 1 所示。

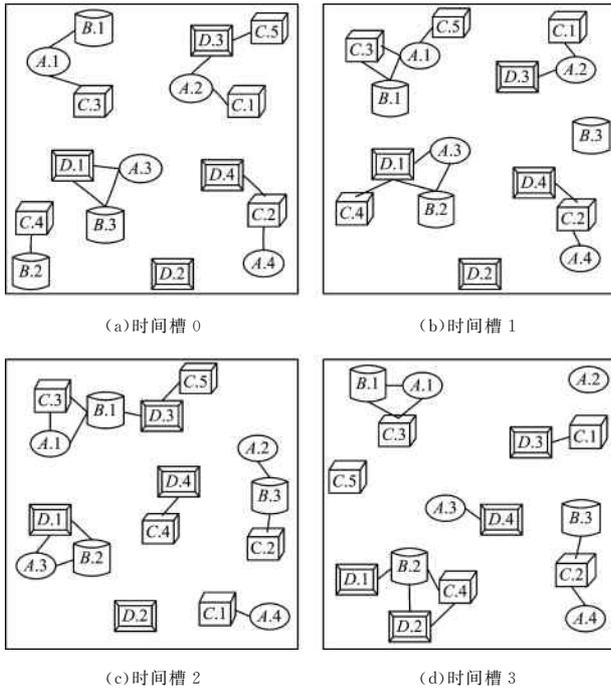


图1 仿真时空数据集

Fig. 1 Simulation spatial-temporal data set

图1所示的时空数据集包含A,B,C,D 4种对象类型。A类型有4个实例,有效周期为时间槽0—时间槽2;B类型有3个实例,有效周期为时间槽0—时间槽3;C类型有5个实例,有效周期为时间槽0—时间槽3;D类型有4个实例,有效周期为时间槽1—时间槽3。图中连线表示在有效周期内两者的曼哈顿距离满足邻近关系。对图1中的数据建立实例网络层,如图2所示。

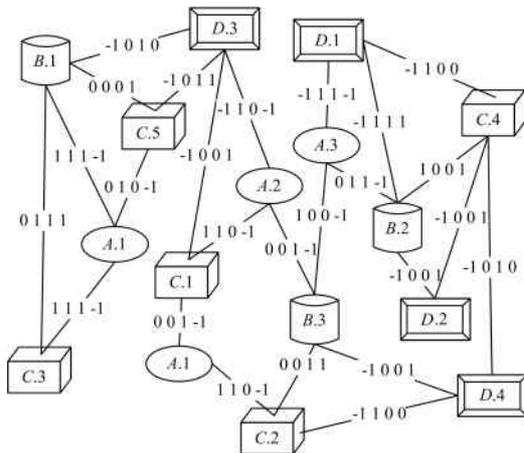


图2 时空数据集实例网络层

Fig. 2 Instance network layer for spatial-temporal data set

在图2中,实例之间在各时间槽的邻近关系组成序列,其作为图中结点之间边的属性,从该网络层中可以高效地获取实例在所有时间槽中与其他实例的空间邻近关系。结点之间边上的序列值为1,表示所对应的时间槽中边所连接的两个实例是邻近关系,某一类型的模式支持实例数可以很容易地从网络中读取出来,从而计算出模式支持度以及空间频繁度。根据图2的实例网络层建立的元素网络层如图3所示。

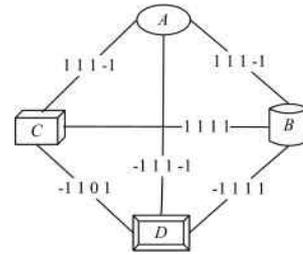


图3 时空数据集的元素网络层

Fig. 3 Element network layer for spatial-temporal data set

图3中的模式的时间框架可以很容易地从元素网络中获得。对于一个模式,若是时空同现模式,则模式中的所有类型在元素网络层中都相连,并在某一时间槽内所有类型之间边上的值都为1,而且至少存在一组实例支持该同现关系。依据此原则,还可以直接从元素网络层生成候选模式集,从而避免大量的连接操作。依据元素网络层保存的同现关系可以快速计算出时间频繁度。

2.3 时空同现模式挖掘算法

本文提出了基于网络的时空同现模式挖掘算法,首先初始化时空数据集,对各时间槽内的实例之间的时空距离进行计算,满足给定距离阈值的实例对符合邻近关系,对这些关系进行建模,形成双层时空网络;读取网络中的邻近关系序列,快速计算出各模式的模式支持度、空间频繁度及时间频繁度,然后计算同现模式的权重特征值。将同现模式按照权重特征值进行排序,获得时空同现模式集。该算法的伪代码及具体步骤如下。

1) 同现模式挖掘的伪代码

算法2 时空同现模式挖掘算法

输入:SD,时空数据集;TP,元素及元素有效周期列表;Dis,元素对象之间的邻近距离

输出:时空同现模式链

伪代码:

1. 初始化网络,生成实例网络层与元素网络层
2. For 每个时间槽 in 总时间框架
3. 遍历元素网络层的时间序列 TE_k
4. IF TE_k 中存在值为1的位
5. THEN 将对应元素对添加到候选模式链表中
6. 根据候选模式链表使用连接方式产生多元候选模式
7. For 每个时间槽 ts in 总时间框架
8. For 每个候选模式 p in 候选模式链中
9. IF p 处于其模式框架
10. THEN For 各元素类型 e in 当前模式 p
11. 读取时空网络并计算 e 的模式支持度
12. 计算 p 在当前时间槽下的空间频繁度
13. For 每个模式 p in 候选模式链表
14. 求出 p 的时间框架中的时间槽总数 n
15. 统计 p 的空间频繁度不为0的时间槽数 m
16. 求出 p 的时间频繁度 m/n
17. For 每个模式 p in 候选模式链表
18. 根据公式计算 p 的权重特征值
19. 使用快排算法根据权重特征值对候选模式排序形成模式链
20. 从时空同现模式链中删除值为0的模式
21. Return 时空同现模式链

2) 同现模式挖掘的具体步骤

①初始化时空网络。如算法1所示,首先读取数据集,确定数据集中各时空对象类型的存在周期;然后将数据格式处理成{实例编号,元素类型,位置信息,时间信息}的格式;再根据给定时空距离阈值,确定时间槽中各实例之间的邻近关系;最后遍历各时间槽的邻近关系,建立实例网络层,并根据实例网络层中各结点的类型属性,构建元素网络层。实例网络层使得模式支持度以及空间频繁度的计算速度加快,元素网络层支持时间频繁度的快速计算,从而加快了整体计算速度。

②计算候选模式的模式支持度。模式支持度是衡量一个模式所包含的所有元素类型对该模式的支持程度。一个模式包含多个元素类型,每个元素类型有多个对象实例,而在元素的所有实例中只有部分实例对象支持该模式同现,支持当前模式的实例数与元素的实例总数之比就是当前元素对该模式的支持度,计算出的所有元素的模式支持度就是当前候选模式的模式支持度,显然,某一候选模式的元素数目决定了该模式的模式支持度的数量。

若某一元素中的实例对象与该模式中其他元素的实例有同现关系,则表示该实例对象支持该模式。在计算元素的模式支持度时,需要统计支持当前模式同现关系的实例个数,即需要判定元素中某一实例是否与模式中其他元素的实例之间是同现关系。在初始化过程中,本文采用时空双层网络对数据进行建模,其中实例网络层保存了各实例对象之间的同现关系。图2所示为基于图1中的时空数据所建立的实例网络层,在保存所建立的网络时对所有元素及实例进行编号,如按字典序给对象B.1和C.1编号为3和5,此时若需要判断实例对象B.1与C.1在时间槽2中是否同现,则只需查询两实例之间所连接的边上的序列中对应时间槽位的值是否为1,也就是查询保存网络的邻接矩阵中[3][5][2]所对应的位是否为1即可。统计出元素的所有实例中支持模式的实例数,根据模式支持度计算公式就可求出当前元素对模式的模式支持度。依次计算出所有时间槽下候选模式中所有元素的模式支持度,从而便完成了对该模式的模式支持度的计算。

元素网络层中保存了时空对象类型之间是否有过同现的关系,如图3所示的元素网络层。以时间槽0中的候选模式集为例,可以直接从图中得出存在同现关系(序列中时间槽0上对应的值为1)的候选模式有:{AB,AC,BC},而传统方法直接从所有元素类型集{A,B,C,D}出发,采用连接或组合方式得到候选集{AB,AC,AD,BC,BD,CD},因此直接从元素网络层获取的候选模式的数量要比直接通过组合方法产生的候选集更少,而且多余的候选模式不存在同现关系,其模式支持度的计算是无效的,但在计算这些模式的模式支持度时需要多次查询时空网络,这会带来较多的时间开销。因此直接采用从元素网络层获得的候选模式集,再对各模式逐次进行模式支持度计算,可以提高算法的运行效率。

③计算各模式在各时间槽的空间频繁度。在步骤2)中计算出了模式支持度,候选模式在某一时间槽下的空间频繁度就是在该时间槽下模式中所有元素的模式支持度中的最小值。在图1给出的时空数据集中,每个元素都存在有效周期,如对于元素A,其有效周期为时间槽0—时间槽2,在时间槽3

中A无效,A的所有实例对象也是无效的,此时包含元素A的候选模式在时间槽3下就是无效的。模式中所有元素都处于有效状态时,该模式才是有效的,当候选模式处于有效状态时的所有时间槽就构成了模式的时间框架,即模式的时间框架就是模式中所有元素的有效时间槽的交集。

从元素网络层可以看到,每个模式并不是在所有时间槽下都是有效的,任一候选模式在其时间框架内的时间槽下是有效的,而在模式时间框架外的时间槽中至少存在一个元素是无效的,此时针对该模式的计算是无意义的,空间频繁度记为-1。对于某一候选模式,首先求出该模式的时间框架;然后依次对时间框架中的各时间槽求出所有元素的模式支持度中的最小值,即该模式在当前时间槽下的空间频繁度,模式支持度在模式的时间框架下大于或等于0。由于模式只有在在其时间框架内的时间槽中,才能保证模式中所有元素都处于有效状态,若在某一时间槽下模式中存在无效的元素,则该模式不满足时空同现关系的条件,其相关计算其实是多余的。因此,模式时间框架的引入使得模式不再计算该框架外的时间槽内的模式支持度及空间频繁度,减少了整体的计算量;而且只计算有效时间槽内的兴趣度,增加了结果的有效性。

4)计算模式的时间频繁度。时间频繁度表征一个模式在其时间框架下存在同现关系的时间频度。在建立时空双层网络时,任意两个元素之间若存在同现关系,则将在这两个元素结点之间的边的序列上对应的时间槽位设为1,如图3中的A元素与C元素,由于A与C在时间槽0—时间槽2下存在同现关系,因此在连接节点A与节点C的边上,前3个时间槽序列都为1,而在时间槽3中因A元素无效,故AC之间不存在同现,对应的第4个时间槽位为-1。因此,对于任意一个模式,若该模式在某一时间槽下同现,则在元素网络层中的对应时间槽下,该模式中所有元素之间的时空关系位上对应的值同时为1。以模式{ABC}为例,在元素网络层中的时间槽1下,A与B之间、A与C之间、B与C之间第一个时间槽上都是1,在实例网络层中存在{A.1,B.1,C.1}实例组在时间槽1上同时为1,那么在时间槽1上模式{ABC}的空间频繁度就不为0。

由此可以看出,若模式在某一时间槽下存在同现关系,则其空间频繁度大于0;若在其时间框架下的时间槽内不存在同现关系,则其空间频繁度为0。空间频繁度的结果也表示了候选模式在当前时间槽下是否同现。时间频繁度是模式出现同现关系的时间槽数与该模式的时间框架下总时间槽数的比值,因此模式的时间频繁度可以根据空间频繁度来计算。若空间频繁度为0,则表示该模式在当前时间槽下不满足时空同现关系;若不为0,则表示满足时空同现关系。统计空间频繁度不为0的时间槽数,将其与模式的时间框架下的总时间槽数相比,计算结果就是时间频繁度。

5)计算模式权重特征值,并据此特征值对同现模式集排序,筛选符合要求的模式集。权重特征值基于模式的时空特征,将模式的各空间频繁度作为模式的时空特征,将模式的时间频繁度作为模式的时间特征,根据式(6)计算出所有模式的权重特征值。模式的权重特征值越大,表示模式的时空同现特征越明显。实际上,可能仅需输出时空特征明显的模式,或

者只需分析时空特征不明显的模式,因此本文按照特征权重值对所有模式进行排序,以便选择同现频繁度较高或较低的模式集。由于本文计算出的各模式的权重特征值并不具备有序特征,而且特征值数据类型并不复杂,因此使用快速排序算法进行排序,并将排序后的模式保存并形成模式链。若在实际需求中只需要时空同现关系很明显的模式集,如需要同现关系较为明显的前20%的模式集,则可以直接从模式链中筛选出权重特征值排名在前20%的模式;若需要同现关系不明显但很可能存在较大价值的模式集,则可以筛选出排名靠后的模式集;若需要查看全部的模式集,则可以输出全部时空同现模式。

在传统方法中,通过设定时空阈值去除了大量不满足阈值的候选模式,也这些频繁度较低的候选模式也出现了同现关系,直接剔除这些频繁度较低的模式可能会丢失某些具有较高价值的同现模式,因此本文采用模式链表来保存所有同现模式,若在实际中需要这些频繁度较低但价值较高的模式,也可以很方便地从链表中获取。

2.4 时空同现模式挖掘算法分析

初始化时空网络后,时空同现模式挖掘算法主要进行候选模式集生成、模式支持度及空间频繁度计算、时间频繁度及权重特征值计算。

候选模式集从元素网络层出发遍历元素网络层生成二元候选模式,由于元素网络层中保存的元素间的时间序列依据实例网络层而生成,而在实例网络层,各对象之间的时间序列通过直接计算原始时空数据的时空关系决定,另外在计算过程中对各时间槽内的全部元素进行了计算,因此实例网络层存储的时空信息是可靠、全面的。实例网络层中的时间序列中为1的位是在各时间槽内存在同现关系的位,因此实例网络层生成的元素网络层所对应的时间序列也是元素之间在各时间槽是否同现的具体体现,实例网络层的可靠性保证了元素网络层存储的同现信息的可靠性。在各时间槽内,元素网络层根据时间序列中时间位为1的同现元素筛选出的候选模式是当前时间槽内的全部二元同现模式,多元模式在二元模式集的基础上通过连接生成,合并各时间槽内的同现模式就构成了全部的候选模式。时空双层网络的可靠性保证了此候选模式集的完整性,候选模式集中的各模式只要在某一个时间槽内出现过,同现关系就会被存储在候选集中,不会遗漏部分模式,而且各模式都是有效的。

传统方法直接从原始时空数据集中通过连接生成候选模式集,需要在全部的时间槽下计算相关时空计量度,并默认在所有时间槽下候选模式集是相同的,而本文产生的候选模式集在不同时间槽下可能是不同的,因为在不同时间槽下,时空对象的同现关系可能不同,剔除了在各时间槽下没有同位关系的候选模式,这样就减少了候选模式的数量。在时空计量度的计算过程中,因为双层网络已经保存了对象之间及元素之间的时空同现关系,因此直接读取对应矩阵中的时间序列即可,提高了计算效率。

相比,传统同现模式挖掘方法,本文所提的基于网络的时空同现模式挖掘算法的优越性体现在:1)采用双层网络的建模方式,提高了候选时空同现模式的时空兴趣度的计算效率;

2)引入模式时间框架,减少了模式的模式支持度及空间频繁度的计算量,使得空间频繁度及时间频繁度的计算更有效,挖掘得到的时空同现模式更贴近实际应用;3)采用权重特征值及模式链表,完整地保存了所有时空同现模式,避免了传统方法中使用阈值而造成的部分同现模式丢失的问题。

3 实验结果及分析

1)性能分析

针对本文提出的建模方法,分别从时间复杂度和空间复杂度进行分析。建模过程包括实例网络层初始化及元素网络层初始化两部分。在形成实例网络层的过程中,遍历元素及元素有效周期列表,其只与元素个数有关,用 OT 表示元素个数,其时间复杂度为 $O(OT)$ 。在空间消耗方面,只需要两个表示时间框架起始与结束的变量,故空间复杂度为 $O(1)$ 。对于用于保存同位信息的邻接矩阵,用 T 表示时间槽个数,用 N 表示所有对象的个数,需占据空间 $O(N * N)$ 。在创建实例网络层的过程中,需要计算每个时间槽下对象之间的距离,并判断是否同位,该环节在最坏情况下的时间复杂度为 $O(T * N * (N+1)/2)$,且该过程只需要一个额外的存储空间来暂存计算距离,空间复杂度为 $O(1)$ 。因此实例网络层形成的时间复杂度为: $O(OT + T * N * (N+1)/2)$,空间复杂度为 $O(N * N)$ 。在元素网络层的生成过程中,需要遍历一次实例网络层,根据实例网络层的时间序列设置元素网络层的时间序列,其时间复杂度为 $O(T * N * (N+1)/2)$,保存元素网络层时间序列的矩阵需占据空间 $O(OT * OT)$ 。

初始化过程中,总的时间复杂度为: $O(OT + T * N * (N+1)/2) + O(T * N * (N+1)/2) = O(OT + T * N * (N+1))$,总的空间复杂度为: $O(N * N) + O(OT * OT) = O(N * N + OT * OT)$,时间复杂度与元素类型个数、时间槽数、对象实例个数有关,空间复杂度与元素类型个数及对象实例个数有关。在对象个数远大于元素类型数时,时间复杂度约为 $O(N^2)$,空间复杂度约为 $O(N^2)$ 。

在最坏情况下,候选模式生成的时间复杂度为 $O(OT * OT)$,空间复杂度为 $O(L)$, L 表示模式链表的长度。在计算模式支持度及空间频繁度的过程中,直接从实例网络层读取相应的同位信息,并进行统计计算,不需要遍历全部数据集,其时间复杂度为 $O(L)$,在计算过程中最多需要 T 个存储空间来保存各时间槽的空间频繁度,因此空间复杂度为 $O(T * L)$;在计算时间频繁度时,需要访问各时间槽下模式的空间频繁度,其时间复杂度为 $O(T * L)$;需要为各模式开辟存储空间来存储模式的时间频繁度,空间复杂度为 $O(L)$;最后计算模式的权重特征值,将前面的时间频繁度及空间频繁度作为特征组,其计算时间复杂度为 $O(L)$,而且链表中各模式的计算结果都需要保存,因此其空间复杂度为 $O(L)$,在算法实现过程中采用高效的快速排序算法进行模式的排序,其时间复杂度为 $O(L \log L)$,空间复杂度为 $O(\log L)$ 。从初始化到时空同现模式链表的最终生成,忽略较小时间和空间,总的时间复杂度约为: $O(OT^2 + T * N^2)$,总的空间复杂度约为 $O(OT^2 + N^2 + T * L)$,在对象实例数远大于元素类型数的情

况下,总的时间复杂度为 $O(N^2)$,空间复杂度为 $O(N^2)$ 。由此可以看出,算法的时空消耗主要发生在初始化建模过程中。相比于传统方法时间复杂度 $O(2^N)$ 以及空间复杂度 $O(2^N)$,在数据量较大时,本文所提方法明显提高了效率。

图 4 给出了建模时间与时空同现模式挖掘总时间的对比结果,在数据量较大时,挖掘算法的性能消耗主要在于建模过程,这与上述分析结果一致,当对象个数远大于类型数时,算法的总时空复杂度与建模过程的时空复杂度是一个数量级,而实际时空计量度的计算所消耗的时间占比很小。另外,建模结果可以保存再复用,当有新的对象加入数据集中时,不需要重新建模,只需要在网络中增加相应节点,并更新时间序列即可。

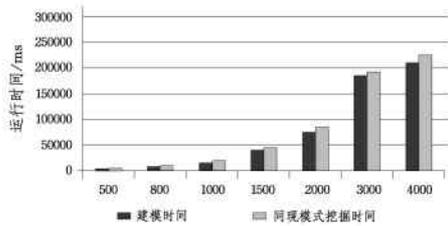


图 4 建模时间与时空同现模式挖掘总时间的比较

Fig. 4 Comparison between modeling time and total spatial-temporal co-occurrence pattern mining time

为验证本文所提方法的有效性,分别采用北京车辆 GPS 轨迹数据(简称北京数据集)、深圳市车辆数据集(简称深圳数据集)、旧金山车辆 GPS 轨迹数据(简称旧金山数据集)及仿真数据集进行了实验。北京数据集来源于微软亚洲研究院发布的 Geolife 项目数据集,在文献[16-17]中公开发表,可在微软研究院官方网站¹⁾获取到相应的数据集,该数据集包含总时间长度为 6 天的 10357 辆车的轨迹点数据,每条数据包含车辆编号、时间信息及经纬度,轨迹点数约为 1500 万。深圳数据集来源于同济大学数学系数学建模大赛²⁾,其包含了 15206 辆出租车,每辆车每 20 秒产生一个 GPS 采样点,每个采样点包含车牌号、采集时间、经度、纬度、车辆状态、车速及行车方向信息。在旧金山数据集中包含 500 辆车超过 30 天的轨迹点数据,每条数据包含编号、经度、纬度、是否载客及采样点时间信息,该数据集已公开发表,可以从国内数据堂官方网站免费³⁾获取。除了真实数据集,本文还使用空间坐标中的仿真数据集进行了测试,仿真数据集中每条数据以{类型,编号,时间槽,x 坐标,y 坐标}格式形成,生成的仿真数据集中总共包含 18 个时间槽和 35 个元素类型,每个元素类型的实例数在 1~55 之间不等,平均实例数为 19。在时空同现模式挖掘中,需要数据集中的类型、编号、时间及经纬度信息,由于不同数据集的格式及所包含的信息不同,在同现模式挖掘实验时,本文将不同数据集中的数据处理成统一的格式。表 1 列出了处理后的北京车辆数据集片段,该片段包含类型、实例编号、采样点时间、采样点经纬度,其他数据集的格式与之类似。

¹⁾ <https://www.microsoft.com/en-us/research>

²⁾ <http://math.tongji.edu.cn/model/camp2011D.html>

³⁾ <http://more.datatang.com/data/15731>

表 1 北京数据集处理后的数据片段

Table 1 Processed data fragment of Beijing data set

类型	实例编号	时间	经度	纬度
1	534	2008-02-02 13:31:06	116.45852	39.87680
1	534	2008-02-02 13:31:21	116.45852	39.87690
1	534	2008-02-02 13:31:36	116.45852	39.87700
2	650	2008-02-02 13:33:47	116.34772	39.87428
2	650	2008-02-02 13:33:50	116.34773	39.87425
2	650	2008-02-02 13:34:20	116.34782	39.87253
3	675	2008-02-02 13:31:41	116.36005	39.88818

如图 5 所示,对本文算法在不同数据集上选取不同对象类型数时的运行时间进行了测试比较。结果表明,在仿真数据集上的效果优于其他数据集,这是由于仿真数据集的数据点比真实数据集中的点更规律。在 3 个真实数据集中,从较小数据集到较大数据集,算法的运行效率相差不大,这表明本文方法具有一定的普适性。

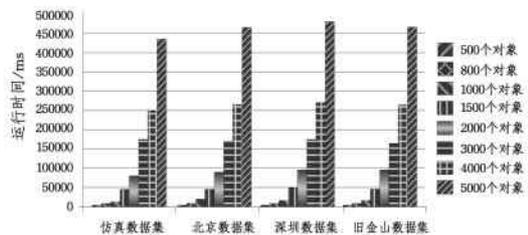


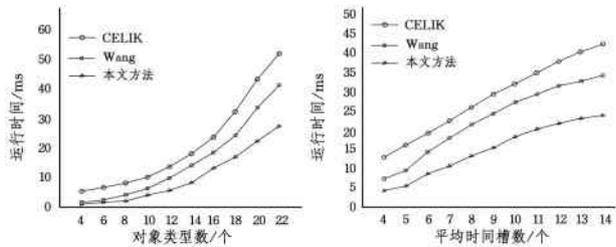
图 5 不同数据集上随着对象类型数增加的运行时间比较

Fig. 5 Comparison of running time with increasing number of object types on different data sets

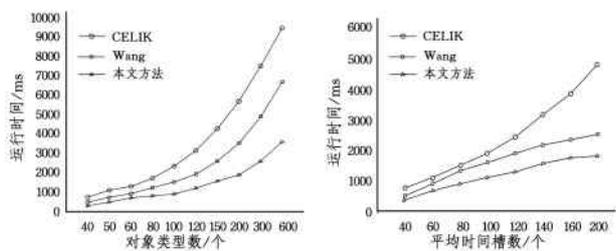
2) 本文算法与其他方法的比较

为验证本文方法的效率,从已有数据集中选取相同的测试数据集,在获取相同模式集结果的情况下将本文方法与 Wang 等人所提的方法以及 CELIK 所提的方法进行了比较。由于 CELIK 的方法在阈值确定后所包含的结果集是固定的,为保证获取相同的结果,以 CELIK 方法的结果集为准,调整本文方法中所要输出的模式比例值以及 Wang 方法中的 k 值以获取相同的模式集。CELIK^[9]提出的局部时空同现模式挖掘算法考虑了同现模式中不同目标类型的生命周期,重新定义了模式的时间频繁度的计算方法,通过这种方法挖掘出来的时空同现模式的适用性更强,是目前比较优越的算法之一。Wang 等人^[12]提出的 Top- k %混合时空同现模式挖掘方法对时空数据集进行了实例间空间关系的建模,一定程度上提高了同现模式挖掘效率并采用 top- k %方法选择得到时间维度下最频繁的时空同现模式集,解决了时间频繁度的设置问题。本文以 3 种方法获取到的全部时空同现模式集为结果,分别采用上述不同的数据集在小数据量以及较大数据量的情况下对 3 种方法进行测试,并取每类测试结果的平均值作为最终结果,运行结果如图 6 所示。从图 6(a)可以看到,在获取相同结果的情况下,随着对象类型的增加,本文所提方法比 Wang 等人的方法以及 CELIK 的方法的运行效率更高,而且类型数量越大,优越性更加显著。图 6(b)采用平均有效周期的时间槽个数作为测量指标,随着时间槽数的增加,3 种方法呈单调递增趋势,本文方法比其他两种方法的运行时间更少。

当数据量增大时,从图 6(c)和图 6(d)可以看到,本文方法比其他两种方法的运行效率更有优势。Wang 等人的方法仅对时空实例间的时空关系进行建模,相比本文的双层网络,单实例网络模式在计算时空同现模式的时间频繁度时效率较低,因此运行时间比本文方法更长。相比于 CELIK 方法,本文的候选集基于元素网络层而生成,其产生的候选模式的数量比从数据集中采用连接操作生成的候选模式集的数量少,从而减少了计算量,因此本文方法的运行效率更高。



(a)小数据量下对象类型数对算法的影响 (b)小数据量下平均时间槽数对算法的影响



(c)较大数据量下对象类型数对算法的影响 (d)较大数据量下平均时间槽数对算法的影响

图 6 3 种方法运行效率的比较效果

Fig. 6 Efficiency comparison of three methods

结束语 本文针对时空数据集提出双层网络的建模方式,保存了实例对象之间在各时间槽的同位关系,使得时空兴趣度的计算更便捷,减少了算法的计算量。有效周期及权重特征值的引入使得时空兴趣度的计算更实际,生成的模式集更有价值,而且采用模式链表较完整地保存了时空同现模式集。实验结果表明,本文的时空网络算法减少了算法的运行时间,提高了时空同现模式挖掘的计算效率。

参考文献

- [1] CAI J N, LIU Q L, XU F, et al. An Adaptive Method Mining Hierarchical Spatio Co-location Patterns[J]. Acta Geodaetica et Cartographica Sinica, 2016, 45(4): 474-485. (in Chinese)
蔡建南, 刘启亮, 徐枫, 等. 多层次空间同位模式自适应挖掘方法[J]. 测绘学报, 2016, 45(4): 474-485.
- [2] AKBARI M, SAMADZADEGAN F, ROBERT W. A generic regional spatial-temporal co-occurrence pattern mining model: a case study for air pollution[J]. Journal of Geographical Systems, 2015, 17(3): 249-274.
- [3] ZHAO X J, SUN Z X, YUAN Y. An Efficient Association Rule Mining Algorithm Based on Prejudging and Screening[J]. Journal of Electronics & Information Technology, 2016, 38(7): 1654-1659. (in Chinese)
赵学健, 孙知信, 袁源. 基于预判筛选的高效关联规则挖掘算法[J]. 电子与信息学报, 2016, 38(7): 1654-1659.
- [4] MAOLEGI M A, ARKOK B. An improved Apriori algorithm for association rules[J]. International Journal on Natural Language Computing, 2014, 3(1): 21-29.
- [5] TANK D M. Improved algorithm for mining association rules [J]. International Journal of Information Technology and Computer Science, 2014, 6(7): 15-23.
- [6] GE L, JI X S, JIANG T. Discovery of network information content security incidents based on association rules and its implementation in Map-Reduce[J]. Journal of Electronics & Information Technology, 2014, 36(8): 1831-1837. (in Chinese)
葛琳, 季新生, 江涛. 基于关联规则的网络信息内容安全事件发现及其 Map-Reduce 的实现[J]. 电子与信息学报, 2014, 36(8): 1831-1837.
- [7] YOO J S, SHEKHAR S. A Joinless Approach for Mining Spatial Colocation Patterns[J]. IEEE Transactions on Knowledge and Data Engineering, 2006, 18(10): 1323-1337.
- [8] HUANG Y, ZHANG L, ZHANG P. A framework for mining sequential patterns from spatio-temporal event databases [J]. IEEE Transactions on Knowledge and Data Engineering, 2008, 20(4): 433-448.
- [9] CELIK M. Partial spatio-temporal co-occurrence pattern mining [J]. Knowledge and Information Systems, 2015, 44(1): 27-49.
- [10] PILLAI K G, ANGRYK R A, BANDA J M, et al. Spatiotemporal co-occurrence rules[C]// New Trends in Databases and Information Systems; 17th East European Conference on Advances in Databases and Information Systems. Berlin, German; Springer International Publishing, 2014: 27-35.
- [11] TIAN J, WANG Y H, YAN F, et al. A New Method for Co-location Patterns Between Network Spatial Phenomena[J]. Wuhan University (Geomatics and Information Science), 2015, 40(5): 652-660. (in Chinese)
田晶, 王一恒, 颜芬, 等. 一种网络空间现象同位模式挖掘的新方法[J]. 武汉大学学报(信息科学版), 2015, 40(5): 652-660.
- [12] WANG Z Q, PENG X G, GU C H. Mining At Most Top-K% Mixed-drove Spatio-temporal Co-occurrence Patterns[C]// Proceedings of 2013 9th Asian Control Conference (ASCC). Piscataway, NJ: IEEE Press, 2013: 1-5.
- [13] BARUA S, SANDER J. Mining Statistically Significant Co-location and Segregation Patterns[J]. IEEE Transactions on Knowledge and Data Engineering, 2014, 26(5): 1185-1199.
- [14] QIAN F, CHIEW K, HE Q M, et al. Mining Regional Co-location Patterns with kNNG[J]. Journal of Intelligent Information Systems, 2014, 42(3): 485-505.
- [15] AKBARI M, SAMADZADEGAN F. Identification of air pollution patterns using a modified fuzzy co-occurrence pattern mining method[J]. International Journal of Environmental Science and Technology, 2015, 12(11): 3551-3562.
- [16] YUAN J, ZHENG Y, XIE X, et al. Driving with knowledge from the physical world[C]// Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'11). New York, USA: ACM, 2011: 316-324.
- [17] YUAN J, ZHENG Y, ZHANG C Y, et al. T-drive: driving directions based on taxi trajectories[C]// Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS'10). New York, USA: ACM, 2010: 99-108.