ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.21, No.3, March 2010, pp.438−451
doi: 10.3724/SP.J.1001.2010.03454

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

基于动态概率路径事件模型的 RFID 数据填补算法*

谷 峪[1,2], 于 戈[1,2+], 李晓静[2], 王 义[2]

[1](医学影像计算教育部重点实验室(东北大学),辽宁 沈阳 110004)

[2](东北大学 信息科学与工程学院,辽宁 沈阳 110004)

RFID Data Interpolation Algorithm Based on Dynamic Probabilistic Path-Event Model

GU Yu[1,2], YU Ge[1,2+], LI Xiao-Jing[2], WANG Yi[2]

[1](Key Laboratory of Medical Image Computing (Northeastern University), Ministry of Education, Shenyang 110004, China)

[2](School of Information Science and Engineering, Northeastern University, Shenyang 110004, China)

+ Corresponding author: E-mail: yuge@mail.neu.edu.cn

Gu Y, Yu G, Li XJ, Wang Y. RFID data interpolation algorithm based on dynamic probabilistic path-event model. *Journal of Software*, 2010,21(3):438−451. http://www.jos.org.cn/1000-9825/3454.htm

Abstract: Missing reads occur frequently during RFID (radio frequency identification) data collection, which will reduce the accuracy of query results in RFID applications. To solve this problem, the existing algorithms mainly take primitive RFID readings as granularity and adopt window smooth strategy based on tag historical readings, which may interpolate data that the query doesn't care about and incur inaccuracy when multiple logic areas are involved. In this paper, data are transformed from data level to logic area level as the interpolation granularity. Then three data interpolating algorithms based on the probabilistic path-event model are proposed, where the incoming events are judged and interpolated by mining the sequence correlation of known area events. Furthermore, the factor of time is considered, and thus probabilistic path-event model is developed. Abundant experiments prove the proposed algorithms have different performance advantages in different conditions and are predominant over the existing strategy in redundancy and accuracy.

Key words: RFID technology; data interpolation; probabilistic path-event model; area event; missing reads; redundant data

摘 要: RFID 数据采集过程中漏读现象频频发生,降低了 RFID(radio frequency identification)应用中查询结果的准确性.目前解决漏读问题的算法主要是以 RFID 原始读数为粒度,并基于标签自身历史读数进行窗口平滑,这种作法会填补许多与查询无关的冗余数据,并且在多逻辑区域参与的复杂应用中,填补准确率较差.为解决上述问题,首次将RFID数据从数据层抽象到逻辑区域层作为处理的粒度,提出3种基于动态概率路径事件模型的数据填补算法,通过挖掘已知的区域事件的顺序相关性来对后续发生的事件进行判断和填补.进一步,增加对时间因素的考虑,对概率路径事件模型进行扩展.大量实验证明,提出的各个算法在不同的情况下有着不同的性能优势,并且在精简性和准确性上要高于现有的策略.

* Supported by the National Natural Science Foundation of China under Grant Nos.60773220, 60873009 (国家自然科学基金); the National Basic Research Program of China under Grant No.2006CB303000 (国家重点基础研究发展计划(973))

Received 2008-07-09; Accepted 2008-08-28

关键词: RFID 技术;数据填补;概率路径事件模型;区域事件;漏读数据;冗余数据
中图法分类号: TP311 文献标识码: A

无线射频识别(radio frequency identification,简称 RFID)技术是一种非接触式的自动识别和数据获取技术.RFID 技术的基本工作原理是:阅读器广播式地向其周围发送能量,感应到能量的标签立即向阅读器返回自身携带的数据,阅读器对收到的数据进行解码,然后将数据传给主机进行处理.RFID 技术最早应用在雷达监测系统中对物体进行跟踪探测.但是随着无线通信技术和数据管理技术的发展,RFID 技术正广泛应用于供应链管理[1]、交通监控、智能展馆和医院等更多的领域.RFID 技术应用前景的广泛性使其受到了越来越多的关注.

RFID 技术采用无线射频信号进行数据通信,由于无线射频信号极易受环境影响,而且相互间干扰很大,尤其当标签和读器数量增多时,信号干扰加强,导致 RFID 数据的不可靠性很高.RFID 数据的不可靠性主要是数据漏读现象,当漏读现象严重时,查询结果的准确性急剧下降,极大地阻碍 RFID 技术的广泛推广,因此,本文选择解决 RFID 数据漏读问题为主要研究内容,结合路径和时间来填补对查询有用的数据.

例如,图 1 为某个公园的逻辑区域示意图,L_1,L_2,L_3,L_4 表示逻辑区域,每个逻辑区域都布置一个或多个阅读器进行标签探测.如果某个带有标签的游客连续在逻辑区域 L_1 和 L_4 被探测到,那么说明这位游客在经过 L_2 或 L_3 时出现了数据漏读现象,实际游玩路线是 $L_1L_2L_4$ 或 $L_1L_3L_4$;或者该游客连续在 L_1 和 L_2 被探测到,那么该游客实际游玩路径可能是 L_1L_2、$L_1L_2L_4$,或者 $L_1L_2L_4L_3$,可能没有发生漏读现象,也可能漏读现象在一个或多个逻辑区域发生.如何区分这些情况并对数据进行填补将是本文研究的主要问题.

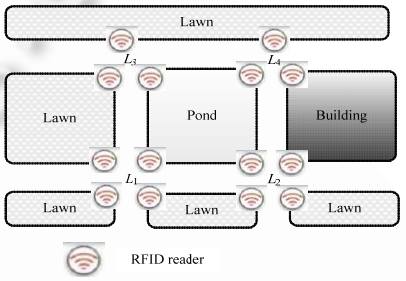


Fig.1 Logic areas illustration of a park
图 1 公园逻辑区域图

与现有的 RFID 和传感器数据清洗工作中针对漏读数据的处理不同,本文提出了一个全新的 RFID 数据填补模型.现有的数据清洗策略都是以 RFID 读数为粒度进行的,而对大多数应用来说,并不关心底层的读数是否丢失,查询需要的原始数据是经过抽象的逻辑区域信息.因此本文首次提出了以逻辑区域事件为粒度进行填补的策略,避免填补冗余的数据信息.现有的 RFID 数据清洗模型主要考虑基于历史读数的窗口平滑方法和时空关联策略,不适合有多逻辑区域参与的基于路径信息的应用场景.特别的,当某逻辑区域的读数全部漏读后,窗口平滑将无法修补该逻辑区域的数据.本文采用了新的概率路径事件模型,同时考虑了路径、区域漏读率和停留时间等重要的填补因素,在多逻辑区域参与的复杂应用中,填补的准确性要高于现有的策略.

本文第 2 节提出问题并定义模型.第 3 节基于概率路径事件提出 3 种数据填补策略.第 4 节通过考虑停留时间对算法进行优化.第 5 节用实验证明模型和算法的有效性.第 6 节总结全文.

1 相关工作

近几年来,RFID 数据管理[2]方面的研究主要集中在以数据为中心的建模[3]、以事件为中心的处理[4]和数据清洗机制.特别的,国外一些科研机构对 RFID 数据清洗策略已经进行了初步的研究,根据数据的历史读数信息和时空关联性[5−7]对数据进行清洗和填补,是一种常见的方法,同时也在传感器网络中得到应用[8].其主要思想是设置一个时间单位,在该时间单位内,传感器或阅读器读出的数值具有时间(或空间)关联性.例如,在一个时间单位的某一时刻,探测到某个标签的存在,则可以认为该标签在这个时间单位内一直存在.

文献[5]提出一种 ESP(extensible receptor stream processing)机制,可以清洗来自不同接收器的数据,并且可以针对各类型脏数据的特点进行清洗.该机制通过设置描述性的查询语句对脏数据进行处理,被设计成管道结构,便于数据处理和模型实现.文献[6]根据数据时间相关性提出了一种基于概率模型的数据清洗策略,主要用于解决数据漏读问题.该方法对静态平滑过滤器策略进行了改进,根据具体应用环境动态地调整窗口的大小.文献[7]则在保证流质量的前提下结合 pipeline 清洗算法[5]提出了一种数据清洗策略,主要解决数据漏读和数据多读

问题.通过计算 confidence 和 coverage 来评估流的质量.

上面提到的 3 种数据清洗策略都对 RFID 数据漏读问题进行了研究,但是这些算法是基于数据层的时空关联对数据进行填补的,当逻辑区域漏读率较大时,有可能标签经过该逻辑区域时一次也没有被探测到,这种情况下,上面的 3 种填补算法失效,不能对漏读标签进行填补.另外,这 3 种算法没有与应用结合在一起考虑,有可能填补的数据是冗余数据,没有任何实用价值,反而极大地浪费了系统资源.而本文则基于数据层的上层——逻辑区域层对数据进行填补,开拓了新的思路,解决了上述算法存在的问题.文献[7]重点关注单逻辑区域的漏读问题,与本文要解决的多逻辑区域漏读问题并不相同.文献[8]主要是根据数据层的大量数据对流质量进行评估,从而进行清洗,与本文研究的内容不在一个层面上.本文在实验中采用了文献[6]提出的 pipeline 方法进行了参照,说明了本文算法在多逻辑区域下的有效性.

文献[9]根据具体应用,提出了一种概率性的数据清洗算法,主要解决数据漏读和数据错读问题,但文献[9]提出的策略对约束性规则较多的应用比较适用,而且约束需要由用户定义,缺少通用性.而本文要解决的问题是以多路径区域应用为前提,不需要用户定义约束规则,所以二者的应用场景不同.文献[10]提出了一种通过统计学习和工作流建模来进行数据挖掘的方法,相关的模型可以用于简单情况的以数据为中心的清洗.但没有考虑 RFID 场景的漏读率等重要的因素,并且它是基于数据库,不是针对事件流进行处理的.

2 问题描述

一般情况下,RFID 数据表示为三元组 $o(T_{epc},R_{epc},t)$形式,为了统一模型,本文称其为读数事件,相当于基本事件,其中,T_{epc} 和 R_{epc} 分别表示标签和阅读器的 EPC 编码,它们均是唯一标识,t 是时间戳,表示阅读器探测到标签的时刻.进一步,本文要将 RFID 数据抽象到逻辑区域层,标记为 $o(T_{epc},L,t_{start},t_{end})$,称为区域事件,其中,$L$ 表示标签被探测到的逻辑区域,t_{start} 和 t_{end} 分别表示标签在 L 被探测到的开始时刻和结束时刻.本节在此数据抽象的基础上,详细描述了 3 种与事件填补算法相关的模型定义,即概率路径事件模型,相似路径事件模型和评价模型.

2.1 概率事件模型

定义 1(路径事件). 一个标签依次经过一个或多个逻辑区域,被称之为路径事件,记为 E,表示为 $l_\alpha L_\beta l_\gamma$,其中 $l_\alpha \in L_{start}, L_\beta \in 2^{L_{all}}, l_\gamma \in L_{end}, L_{start}$ 表示所有起点逻辑区域的集合,L_{end} 表示所有终点逻辑区域的集合,L_{all} 表示所有逻辑区域的集合,包含 L_{start} 和 L_{end},符号 $2^{L_{all}}$ 表示 L_{all} 的幂集.集合中的元素称为区域事件,如 l_α, l_γ.路径事件 E 中包含区域事件的个数称为路径事件的长度,记为 $L_{en}(E)$,长度为 n 的路径事件集合记为 S_n.

定义 2(发生率). 给定一时间段 T,在 T 内某一路径事件 E 的发生率定义为该路径事件发生次数总和与所有路径事件发生次数总和之比,记为 $P_O(T,E)$,如式(1)所示:

$$P_O(T,E)=Count(T,E)/Count(T,U) \tag{1}$$

其中,$Count(T,E)$表示在时间段 T 内,路径事件 E 发生次数总和,U 表示所有路径事件.由此易得,在某段时间 T 内,所有路径事件发生率之和为 1,如式(2)所示:

$$\sum_{E_i \in U} P_O(T,E_i)=1 \tag{2}$$

定义 3(逻辑区域漏读率). 某一逻辑区域 L 的漏读率是指在一给定时间段 T 内,经过该逻辑区域但未被探测到的标签个数总和与所有经过该逻辑区域的标签个数总和之比,记为 $P_{ML}(T,L)$.

逻辑区域漏读率一般受很多因素影响,如该逻辑区域的物理环境、阅读器的布置方位和探测顺序等等.例如在一个公园,水上乐园周围区域的漏读率多数情况下会大于草坪花圃附近区域的漏读率,因为水和金属对 RFID 无线信号传播干扰很强.由此可见,不同的逻辑区域通常会有不同的漏读率,本文将根据上面提到的影响漏读率的因素对每个逻辑区域的漏读率进行估算,即在具体应用下,逻辑区域的漏读率是可以统计的.

定义 4(概率路径事件模型). 概率路径事件模型是包括路径事件发生率和逻辑区域漏读率的路径事件统计模型.它可以用树或哈希表的结构来存储.

下面将给出以树的结构建立概率路径事件模型的例子.如图 2 所示,节点代表逻辑区域,边代表逻辑区域间

的可达性;每个节点有两个属性,即到该节点终止的路径事件发生率和该逻辑区域的漏读率,比如节点 l_5 上的数字分别表示路径事件 $l_1l_3l_4l_5$ 的发生率为 0.4,l_5 的漏读率为 0.3;节点 l_i 指向节点 l_j 的边上的权值 e_{ij} 表示依次经过逻辑区域 $l_1,l_2,\ldots,l_i,l_j$ 的事件概率,其中 l_k 为 l_{k+1} 的父节点,如 e_{34}=0.6 表示某一标签依次被逻辑区域 l_1,l_3,l_4 探测到的概率.

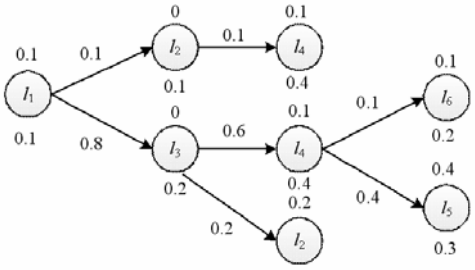


Fig.2 Probabilistic path event model

图 2 概率路径事件模型

此外,RFID 流数据持续到来并不断发生变化,为了保证概率事件模型能够准确预测后续到达的事件,需要对概率事件模型进行实时更新维护.本文采用滑动窗口技术[11]对概率事件模型进行增量维护与减量维护.滑动窗口模型始终维护最近时间段内固定长度的数据集,随着窗口的滑动,不断有新的数据加入,同时有过期的数据被删除.为了方便对数据集进行增减量维护,可将滑动窗口(sliding window)再细化为若干个跳数窗口(hop window),即跳数窗口是滑动窗口的基本组成单位,并且也是每次窗口滑动幅度的基本单位.

采用滑动窗口技术对概率事件模型进行维护时,滑动窗口大小以及每次窗口滑动幅度需要根据具体应用而定,以保证通过统计学习而获得的概率事件模型尽可能准确地预测后续到达的事件.本文将在给定应用条件的背景下,通过实验获得适当的滑动窗口大小和窗口滑动幅度.

2.2 相似路径事件模型

定义 5(相似路径事件). 给定两个路径事件 E_1 和 E_2,记为 $E_1=l_{i_1}l_{i_2}\ldots l_{i_m}$,$E_2=l_{j_1}l_{j_2}\ldots l_{j_n}$,其中 l_{i_k},l_{j_k} 代表某个逻辑区域,如果路径事件 E_1 和 E_2 同时满足下面两个条件:(1) $m=n$;(2) $l_{i_k}=l_{j_k}$,$k\in\{1,2,\ldots,m\}$,则称它们互为相同路径事件.如果路径事件 E_1 只要填补 k 个逻辑区域,就成为 E_2 的相同路径事件,则称 E_2 为 E_1 的 k-相似路径事件,记为 $E_1\overset{k}{\rightarrow}E_2$,由此,相同路径事件亦可称为 0-相似路径事件.符号 $S_i(E,k)$表示路径事件 E 的 k-相似路径事件集.

定义 6(最相似路径事件). 给出 t 时刻的观察值 $o(t)=l_{i_1}l_{i_2}\ldots l_{i_m}$,则其所有相似路径事件集合表示为 $S_{i\_all}=S_i(o(t),k),k\in0,1,2,\ldots,n$,其最相似路径事件定义如式(3)所示:

$$s_{i\_mostly}(o(t))=\left\{r'(t)\middle|P(r'(t)\mid o(t))=\max(P(r(t)=l_{j_1}l_{j_2}\ldots l_{j_n}\mid o(t)=l_{i_1}l_{i_2}\ldots l_{i_m}),r(t)\in S_{i\_all})\right\} \tag{3}$$

定义 7(公共子事件与最长公共子事件). 给定两个路径事件 E_1 和 E_2,如果 $\exists E\left(E\overset{k_1}{\rightarrow}E_1\wedge E\overset{k_2}{\rightarrow}E_2\right)$ 为真(k_1,k_2 是自然数),则 E 称为 E_1 和 E_2 的公共子事件,记为 $E=C(E_1,E_2)$.如果 $!\exists E'\left(E'\overset{k_1'}{\rightarrow}E_1\wedge E'\overset{k_2'}{\rightarrow}E\wedge k_1'<k_1\right)$ 也为真(k_1' 是自然数),则 E 称为 E_1 和 E_2 的最长公共子事件,记为 $E=\hat{C}(E_1,E_2)$.

2.3 评价模型

本小节将给出评价算法精简率和准确率的标准定义.

定义 8(精简率). 在给定时间段 T 内,精简率定义为原始数据经过数据抽象后减少的数据量与未经抽象的数据量之比,记为 $P_D(T)$,计算表达式为

$$P_D(T)=(Account(T,R_a)-Account(T,R_c))/Account(T,R_a) \tag{4}$$

其中,$Account(T,R_a)$和 $Account(T,R_c)$分别表示在时间 T 内到来的源数据量与经过抽象后剩余的数据量.

定义 9(准确率). 给定两个数据集,即真实数据集 R_e 和清洗过的数据集 R_c,在某个时间段 T 内,准确率 $P_A(T)$ 定义见式(5):

$$P_A(T)=R_e(T)\cap R_c(T)/R_e(T) \tag{5}$$

3 填补技术

3.1 数据填补模型

本文提出了一个 RFID 数据填补模型框架,如图 3 所示.

RFID 数据填补模型主要由数据抽象、漏读探测、数据填补机制三大部分组成.数据抽象机制是对 RFID 数据重新建模,即将三元组数据 $o(T_{epc},R_{epc},t)$建模成 $o(T_{epc},L,t_{start},t_{end})$.事件表用于存储所有可能发生的真实路径事件,它是可以预知的.如果发生的某个路径事件不在事件表内,则表示该路径事件被探测时发生区域事件漏读现象.漏读探测机制则是结合事件表,对正在发生的路径事件进行实时监测.如果探测到发生的路径事件存在漏读现象,则将其传送到数据填补机制;否则将事件直接输出,供查询层使用.需要强调的是,数据填补机制是对发生漏读现象的区域事件进行数据填补的,它主要由概率路径事件模型、匹配引擎和填补策略机制 3 部分组成.概率路径事件模型是在对历史事件集统计学习的基础上建立的,并按设定的周期不断进行更新维护.匹配引擎是根据具体的填补算法,搜索当前事件的相似路径事件集,然后将其传送到填补策略机制.填补策略机制在概率路径事件模型和相似路径事件集的基础上,对当前漏读事件进行填补.

本文提出的数据抽象策略考虑在数据抽象的同时对数据漏读有一定的容忍度,即设置一个阈值 *t_smooth*,表示时间平滑窗口大小,当前后两个数据时间间隔小于 *t_smooth* 个时间单位时,对其间数据进行平滑处理.图 4 通过示例描述了数据抽象过程.通过数据抽象,可以对查询有意义的区域事件进行填补,避免填补冗余的数据.

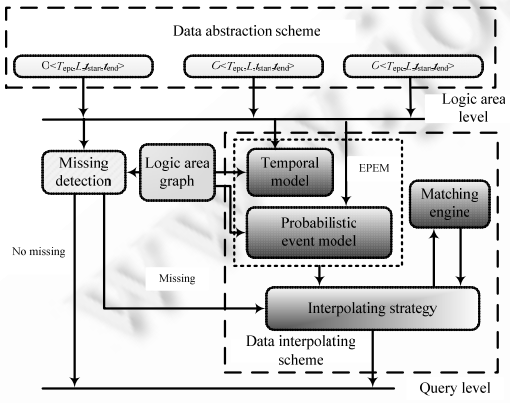


Fig.3 Data interpolation model framework
图 3 数据填补模型框架

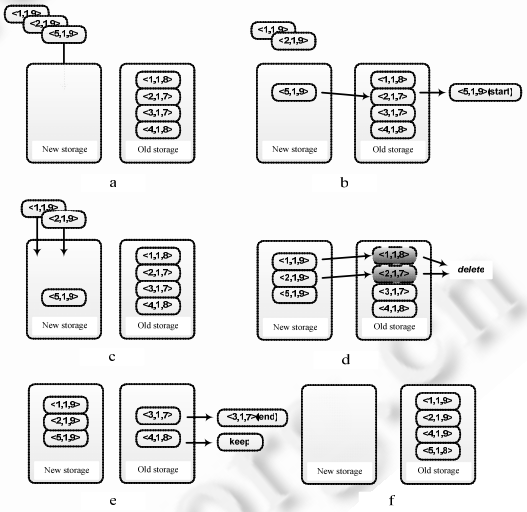


Fig.4 Data abstraction process
图 4 数据抽象过程

3.2 填补算法

3.2.1 理论依据

下面给出一些支持填补算法的理论依据.

引理 1. 设一个逻辑区域 L 的漏读率为 $P_{ML}(L)$,则一个标签经过 L 而未被 L 探测到的概率,即逻辑区域事件漏读率 $P_M(L)=\alpha\cdot(P_{ML}(L))^{w+\vartheta}(\alpha>1)$,其中,$w$ 表示数据建模时平滑窗口大小,ϑ表示逻辑区域 L 布置的阅读器天线个数,α为调整因子.

证明:当一个标签经过漏读率为 P_{ML} 的逻辑区域时,一个阅读器天线在平滑窗口内没有探测到该标签的概率为$(P_{ML})^w$,而 ϑ个阅读器天线均没有探测到该标签的概率 P_M 为$(P_{ML})^{w+\vartheta}$,但由于多个阅读器对标签同时探测时,相互间会有干扰,增大漏读可能性,而且阅读器不同的布置方位,干扰也会略有不同,在这里增加一个调整因子

$\alpha(\alpha>1)$进行调整,使 P_M 尽可能地与实际情况相拟合.

由此可得,逻辑区域 L 漏读该标签的概率,即逻辑区域事件漏读率为 $P_M(L)=\alpha\cdot(P_{ML}(L))^{w+\vartheta}(\alpha>1)$. □

下面根据实际应用中的普遍情况提出一个假设.

一般情况下,携带标签的物体在运动过程中,不会往复经过相同的逻辑区域,例如游客去博物馆参观,一般会按照路线行走,很少情况下会返回到已经参观过的地方.由此,给出下面的假设.

假设 1. 任何一个路径事件 E 不包含两个或两个以上相同的区域事件.

同时,定义一个函数 $F(o(t),r(t))$,见式(6):

$$F(o(t),r(t))=P_O(r(t)=l_{j_1}l_{j_2}...l_{j_n})\cdot\prod_{l_j\in L_\alpha}P_M(l_j) \tag{6}$$

其中,$o(t)$和 $r(t)$分别表示 t 时刻的观察值和真实值候选值,L_α表示出现在 $o(t)$中而未出现在 $r(t)$中的区域事件集合,$P_M(l_j)$表示逻辑区域 l_j 的事件漏读率.在此函数基础上提出定理 1.

定理 1(最相似路径事件定理). 给出观察值 $o(t)=l_{i_1}l_{i_2}...l_{i_m}$,可以求出它的相似路径事件集 S_i,则当且仅当 $F(o(t),r'(t))=\max(F(o(t),r(t)),r(t)\in S_i)$ 时,$r'(t)$是 $o(t)$的最相似路径事件.

证明:由式(3)可知观察事件 $o(t)$的最相似事件定义为

$$s_{i\_mostly}(o(t))=\left\{r'(t)\middle|P(r'(t)\mid o(t))=\max(P(r(t)=l_{j_1}l_{j_2}...l_{j_n}\mid o(t)=l_{i_1}l_{i_2}...l_{i_m}),r(t)\in S_{i\_all})\right\}.$$

根据贝叶斯定理计算后验概率如下:

$$P(r(t)=l_{j_1}l_{j_2}...l_{j_n}\mid o(t)=l_{i_1}l_{i_2}...l_{i_m})=\frac{P(r(t)=l_{j_1}l_{j_2}...l_{j_n})\cdot P(o(t)=l_{i_1}l_{i_2}...l_{i_m}\mid r(t)=l_{j_1}l_{j_2}...l_{j_n})}{P(o(t)=l_{i_1}l_{i_2}...l_{i_m})}=A\cdot F(o(t),r(t)),$$

其中, $A=\dfrac{\prod_{h=1}^{m}(1-P_M(l_{i_h}))}{P_O(o(l_{i_1}l_{i_2}...l_{i_m}))}$.当给定 $o(t)$时,A 可看作是一个常数,由此式(3)可等价变换为

$$\begin{aligned}s_{i\_mostly}(o(t))&=\left\{r'(t)\middle|P(r'(t)\mid o(t))=\max(A\cdot F(o(t),r(t)),r(t)\in S_i)\right\}\\&=\left\{r'(t)\middle|P(r'(t)\mid o(t))=\max(F(o(t),r(t)),r(t)\in S_i)\right\}\\&=\left\{r'(t)\middle|P(r'(t)\mid o(t))=F(o(t),r^*(t))\right\}=r^*(t).\end{aligned}$$

至此,定理得证. □

定理 2(少漏读区域事件定理). 假设正确事件集中区域事件的个数为 m,逻辑区域 l_i 的事件漏读率记为 $P_M(l_i)$,则当且仅当 $n<(1-\bar{p})/\bar{p}$ 时,$p(x=0)>p(x=1)>\ldots>p(x=n)$,其中 n 表示到达路径事件的真实长度(包括漏读的区域事件), $\bar{p}=\dfrac{1}{m}\sum_{i=1}^{m}P_M(l_i)$,$p(x=k)$表示到达路径事件漏读 k 个区域事件的概率.

证明:当一个标签依次经过 n 个逻辑区域时,没有探测到它的平均概率为 $\bar{p}$,探测到它的平均概率为 $1-\bar{p}$,并且某个逻辑区域漏读某个标签是随机现象,逻辑区域之间没有相关性,由此,可以将此过程看作是一个服从伯努利二项分布的过程,即 $p(x=k)=C_n^k\cdot\bar{p}^k\cdot(1-\bar{p})^{n-k}$,由此可以得到下面的推导过程:

$$p(x=k)>p(x=k+1)\Leftrightarrow C_n^k\cdot\bar{p}^k\cdot(1-\bar{p})^{n-k}>C_n^{k+1}\cdot\bar{p}^{k+1}\cdot(1-\bar{p})^{n-k-1}\Leftrightarrow\frac{1-\bar{p}}{\bar{p}}>\frac{n-k}{k+1}\Leftrightarrow(n+1)\cdot\bar{p}<k+1.$$

那么,原式可以变形如下:

$$p(x=0)>p(x=1)>\ldots>p(x=n)\Leftrightarrow\forall k,(n+1)\cdot\bar{p}<k+1\Leftrightarrow(n+1)\cdot\bar{p}<k_{\min}+1=1\Leftrightarrow n<\frac{1-\bar{p}}{\bar{p}} \tag{7}$$

至此定理得证. □

下面将给出 3 种数据填补策略的主要思想.限于篇幅,具体的算法描述不在本文给出.

3.2.2 贪婪算法

贪婪算法主要侧重事件填补时的实时性要求,对到达的数据边监控边填补,其基本思想是将概率事件模型

用树结构存储,当有区域事件到达时,根据广度优先原则与节点进行匹配.

贪婪算法可以保证很高的实时性,在线处理实时到来的区域事件,但它的策略是搜索当前最优解,不进行回溯遍历,因此不能保证全局最优,甚至会出现填补错误.例如在图 2 中,当探测到的路径事件为 l_1l_2 时,贪婪算法会判定路径事件 $l_1l_2l_4$ 发生,尽管路径事件 $l_1l_3l_2$ 发生的可能性更大.

3.2.3 最小 k-相似算法

由定理 2 可知,当应用条件满足式(7)时,探测到的路径事件更倾向于没有漏读或漏读个数较少的情况,由此,提出最小 k-相似算法.此算法的主要思想是当完整路径事件到达时,与正确事件集进行匹配,如果匹配成功,则认为该路径事件没有漏读;如果匹配不成功,则根据定理 1 从它的最小 k-相似路径事件集中求出它的最相似路径事件,据此对漏读区域事件进行填补.

最小 k-相似算法弥补了贪婪算法中的一些问题,但它采取的策略是从区域事件漏读最少的相似路径事件集中搜索,如果搜索到,就不再考虑区域漏读更多的情况,所以此算法仍没有进行全局遍历,有些情况不能达到全局最优.例如在图 2 中,当检测到路径事件 $l_1l_3l_4$ 时,最小 k-相似算法会判定路径事件 $l_1l_3l_4$ 发生,而此时路径事件 $l_1l_3l_4l_5$ 发生的可能性更大,由此会出现填补错误.而该算法是在式(7)的前提下提出的,当 n 和 $\bar{p}$ 满足关系式(7)时,此算法可以达到较优的效果.但如果不满足该关系式,效果会较差,因此提出下面的全相似算法.

3.2.4 全相似算法

全相似算法的主要思想是当完整路径事件到达时,求出它的所有相似路径事件集 S_{i_all},然后从中求出它的最相似路径事件集,对漏读区域事件进行填补.全相似算法进行的是全局遍历,对所有可能情况都进行检测,从中选出漏读区域事件的最相似路径事件,这种方法能够保证很高的准确性,但计算开销较大.

3.2.5 准确性定理

下面针对最小 k-相似算法和全相似算法,给出一个准确率方面的定理.

定理 3(最长公共子事件定理). 给定 η, $\eta=\max_{E_i,E_j\in\hat{E}}L_{en}(\hat{C}(E_i,E_j))$ 和 n,$n=L_{en}(E)$,E 为到来事件,当 $n>\eta$ 时, $\hat{P}_A(E)=1$, $\hat{P}_A(E)$ 表示相似度算法填补事件 E 的准确率.

证明:由最长公共子事件定义,可以得到下面推理:

$$\eta=\max_{E_i,E_j\in\hat{E}}(\hat{C}(E_i,E_j))\Leftrightarrow\forall E'(E'\xrightarrow{k_1'}E_i\wedge E'\xrightarrow{k_2'}E_j\wedge L_{en}(E')\le\eta)\Leftrightarrow!\exists E'(E'\xrightarrow{k_1'}E_i\wedge E'\xrightarrow{k_2'}E_j\wedge L_{en}(E')>\eta).$$

所以,当 $n>\eta$时,其中 $n=L_{en}(E)$,根据相似度算法原理可得:

$$!\exists E_i,E_j(E\xrightarrow{k_1}E_i\wedge E\xrightarrow{k_2}E_j\wedge E_i,E_j\in\hat{E})\Leftrightarrow|S_{i\_all}(E)|=1\Leftrightarrow S_{i\_mostly}(E)=\hat{E}.$$

$\hat{E}$ 为漏读事件 E 的正确事件.所以, $\hat{P}_A(E)=P(S_{i\_mostly}(E)=\hat{E})=1$.至此,定理得证. □

4 改进策略

在 RFID 应用中,获取数据的同时,也得到了一个比较重要的属性,就是时间戳.一般情况下,当一个携带标签的物体经过各个逻辑区域时,不仅在选择不同逻辑区域时存在一定的规律性,而且在不同逻辑区域停留的时间间隔也存在一定的规律性.如在上节提到的公园系统中,游客在小剧院的停留时间大概是戏剧的演出时间,在游乐场等趣味性强的场所,大部分游客会停留时间长一些,而在一些景观少的场所,大部分游客不会停留太长的时间.所以,可以将停留时间间隔加入到概率路径事件模型中,当标签在某个逻辑区域漏读时,不仅考虑逻辑区域顺序关联性,同时也增加对标签漏读时间间隔的考虑,从这两个方面权衡对漏读数据进行填补.

4.1 β^*改进算法

在考虑时间属性后,可以对最相似路径事件定义进行扩展.

定义 10(扩展最相似路径事件). 给出 t 时刻的观察值 $o(t)=l_{i_1}l_{i_2}...l_{i_m}$,及其所有相似路径事件集合 S_{i_all},则其扩展最相似路径事件定义见式(8):

$$S'_{i\_mostly}(o(t))=\left\{r'(t)\middle|P(r'(t)\mid o(t))=\max(P(r(t)=l_{j_1}l_{j_2}...l_{j_n}\mid o(t)\wedge t_k=\Gamma_k),k\in L_\zeta\right\} \tag{8}$$

其中,L_ζ表示出现在 $o(t)$中而未出现在 $r(t)$中的区域事件集合,t_k 表示标签在逻辑区域 k 的漏读时间间隔.此外,标签在漏读区域停留时间段的求解与 RFID 阅读器的布置有一定关系,如果阅读器布置在十字路口,如图 1 所示,那么标签在漏读区域的停留时间Γ即是标签在该路径景区的停留时间.假设某个标签先后经过两个不相邻逻辑区域的结束时刻和开始时刻分别是 t_1 和 t_2,两个逻辑区域的距离为 l,标签的平均速度为 v(可以根据已知数据计算求解),则可得 $\Gamma=t_2-t_1-l/v$.

假设 2. 标签先后经过逻辑区域 l_1 和 l_2 时的停留时间相互独立.即标签在逻辑区域 l_1 的停留时间不会影响在逻辑区域 l_2 的停留时间.

假设 3. 标签在某个逻辑区域停留时间与标签经历的逻辑区域序列相互独立.也就是说,在不同逻辑区域序列中,标签在某个逻辑区域停留时间符合同一个分布,即式(9)成立:

$$P(t_{l_i}=\Gamma\mid o=l_i)=P(t_{l_i}=\Gamma\mid o=L_\alpha l_i L_\beta),L_\alpha,L_\beta\in 2^{L_{all}} \tag{9}$$

其中,o 表示观察值,t_{l_i} 表示标签在逻辑区域 l_i 的漏读时间间隔.

以上两个假设在一般的应用中是完全合理的,在这两个假设的基础上,给出下面定理的证明.

首先,定义一个权值函数 $\beta(o(t),r(t))$,见式(10):

$$\beta(o(t),r(t))=\prod_{k\in L_\alpha}P(t_k=\Gamma_k\mid r(t)=l_k) \tag{10}$$

其中,$o(t)$和 $r(t)$分别表示 t 时刻的观察值和真实值候选值,L_α表示出现在 $o(t)$中而未出现在 $r(t)$中的逻辑区域集合,t_k 表示标签在逻辑区域 k 的漏读时间间隔.在此函数基础上提出定理 4.

定理 4(扩展最相似路径事件定理). 给出观察值 $o(t)=l_{i_1}l_{i_2}...l_{i_m}$ 和其相似事件集 S_i,当且仅当 $\beta(o(t),r'(t))F(o(t),r'(t))=\max(\beta(o(t),r(t))F(o(t),r(t)),r(t)\in S_i)$ 时,$r'(t)$是 $o(t)$的扩展最相似事件.

证明:由公式(8)可知,观察事件 $o(t)$的扩展最相似事件定义如下:

$$S'_{i\_mostly}(o(t))=\left\{r'(t)\middle|P(r'(t)\mid o(t))=\max(P(r(t)=l_{j_1}l_{j_2}...l_{j_n}\mid o(t)\wedge t_k=\Gamma_k),k\in L_\zeta\right\}.$$

根据贝叶斯定理计算后验概率如下:

$$\begin{aligned}
&P(r(t)=l_{j_1}l_{j_2}...l_{j_n}\mid o(t)=l_{i_1}l_{i_2}...l_{i_n}\wedge t_k=\Gamma_k),(k\in L_\alpha)\\
&=\frac{P(r(t)=l_{j_1}l_{j_2}...l_{j_n})\cdot P(o(t)=l_{i_1}l_{i_2}...l_{i_n}\wedge t_k=\Gamma_k\mid r(t)=l_{j_1}l_{j_2}...l_{j_n})}{p(o(t)=l_{i_1}l_{i_2}...l_{i_n}\wedge t_k=\Gamma_k)},(k\in L_\alpha)\\
&=\frac{P(r(t)=l_{j_1}l_{j_2}...l_{j_n})\cdot P(o(t)=l_{i_1}l_{i_2}...l_{i_n}\mid r(t)=l_{j_1}l_{j_2}...l_{j_n})\cdot P(t_k=\Gamma_k\mid r(t)=l_{j_1}l_{j_2}...l_{j_n})}{p(o(t)=l_{i_1}l_{i_2}...l_{i_n}\wedge t_k=\Gamma_k)},(k\in L_\alpha)\\
&=\frac{P(r(t)=l_{j_1}l_{j_2}...l_{j_n})\cdot\prod_{h=1}^{m}(1-P_M(l_{i_k}))\cdot\prod_{l_j\in L_\alpha}P_M(l_j)\cdot P(t_k=\Gamma_k\mid r(t)=l_{j_1}l_{j_2}...l_{j_n})}{p(o(t)=l_{i_1}l_{i_2}...l_{i_n}\wedge t_k=\Gamma_k)},(k\in L_\alpha)\\
&=A\cdot F(o(t),r(t))\cdot\prod_{k\in L_\alpha}P(t_k=\Gamma_k\mid r(t)=l_{j_1}l_{j_2}...l_{j_n})\ (\text{定理 1 和假设 2})\\
&=A\cdot F(o(t),r(t))\cdot\prod_{k\in L_\alpha}P(t_k=\Gamma_k\mid r(t)=l_k)\ (\text{假设 2})\\
&=A\cdot\beta(o(t),r(t))\cdot F(o(t),r(t))\ (\text{权值函数定义}).
\end{aligned}$$

由定理 1 知,A 是一个常量,由此扩展最相似事件变换如下:

$$\begin{aligned}
S'_{i_mostly}(o(t))&=\left\{r'(t)\middle|P(r'(t)\mid o(t))=\max(A\cdot\beta(o(t),r(t))\cdot F(o(t),r(t)))\right\}\\
&=\left\{r'(t)\middle|P(r'(t)\mid o(t))=\max(\beta(o(t),r(t))\cdot F(o(t),r(t)))\right\}\\
&=\left\{r'(t)\middle|P(r'(t)\mid o(t))=\beta(o(t),r^*(t))F(o(t),r^*(t))\right\}=r^*(t).
\end{aligned}$$

至此,定理得证. □

在定理 4 的基础上,本文提出了β*改进算法,即在求解漏读路径事件的最相似路径事件时,将标签漏读时间段作为条件加以考虑,用一个权值函数 $\beta(o(t),r(t))$ 对定理 1 的结果进行修正.由此,得到了 3 种基于β*改进的数据填补算法,即β*改进贪婪算法,β*改进最小 k-相似算法和β*改进全相似算法.改进算法是在定理 4 的基础上,对漏读路径事件按照扩展最相似路径事件进行填补,具体的算法可以通过修改 3 种基本填补策略得到.

由此可见,在β*改进算法中,一个比较关键的步骤就是对权值函数 $\beta(o(t),r(t))$ 进行求解,下面将给出求解 $\beta(o(t),r(t))$ 的算法.根据式(10)的定义可知,求解权值函数 $\beta(o(t),r(t))$ 的过程主要是求解标签在每个逻辑区域停留时间的分布情况,即建立近似时间模型,本文将采用直方图[12]的方法对其进行求解.

在算法 1 中,对建立近似时间模型的直方图算法过程进行了描述.其中给定的参数包括滑动窗口大小 S,窗口滑动的幅度 R 和直方图的组距 W.

算法 1. 直方图算法(histogram algorithm).

输入:逻辑区域图 G,经过数据抽象处理的源事件流;

输出:近似时间模型.

1. establish two-level hash table T_a by S, and the keys are separately L and t_{end}–t_{start}, and creat N node spaces, N is the number of logic areas;
2. initial time window whose size is S over event stream;
3. **while**($o(T_{epc},L,t_{start},t_{end})$==getSimpleEvent())
4. $T_a[f_1(L)][f_2(t_{start}-t_{end})][currentHop]++$;
5. **if** (newHopArrive()) **then**
6. creat N node spaces;
7. $currentHop=newHop$;
8. **end if**
9. **if** (slidingHopArrive()) **then**
10. delete expired hops and add newly arriving hops;
11. **end if**
12. **end while**

4.2 β+改进算法

Minkowski 距离[13]是欧几里德距离的推广,也称为 L_p 距离.给定时间序列 $X=\{x_1,x_2,\ldots,x_n\}$和 $Y=\{y_1,y_2,\ldots,y_n\}$,其 L_p 距离距离定义如公式(11)所示:

$$L_p(X,Y)=\left[\sum_{i=1}^{n}|x_i-y_i|^p\right]^{\frac{1}{p}},\ p\geq 1 \tag{11}$$

其中,当 p=2 时为欧氏距离(Euclidean distance).

当标签在逻辑区域的停留时间分布比较集中时,采用直方图方法对时间进行估计,准确率会下降,这时如果采用欧式距离对其进行评估,会取得更好的效果,β+改进算法则采用此方法,下面进行具体描述.

β+改进算法也是在扩展概率事件模型的基础上提出的,不过该算法是基于精确时间模型.在基于定理 4 对填补算法进行改进的同时,也可以在填补数据的时候并行考虑事件顺序性和标签停留时间间隔这两个方面的影响,并通过一个参数调节这两个影响的权重大小,基于这个思想,本文又给出了一个扩展最相似事件的定义方式,如式(12)所示:

$$S''_{i\_mostly}(o(t))=\alpha\cdot S_{i\_mostly}(o(t))+(1-\alpha)\cdot\beta'(o(t),r(t)) \tag{12}$$

其中,$S_{i_mostly}(o(t))$的定义在式(3)中给出,而 $\beta'(o(t),r(t))$ 的定义在式(13)中给出,α是一个调整因子,具体数值可由实验获得.

$$\beta'(o(t),r(t))=\sum_{l\in o\wedge l\notin r}g(L_2(t_l,D(t))) \tag{13}$$

其中,$g(x)$是一个单调递减函数,$D(t)$是精确时间模型在 t 时刻维护的每个区域事件的 t_{end}–t_{start} 集合.

β+改进算法是在式(12)的基础上提出来的,在对漏读路径事件进行填补时,它不仅考虑了路径事件发生率

和逻辑区域漏读率这两个因素,还考虑了标签在漏读区域停留时间这个因素,并且通过α来调整他们各自对判断标签属于哪个漏读区域的影响力大小,在不同的应用情况下,α取值会有所不同.由此,得到了 3 种基于β+改进的数据填补算法,即β+贪婪算法、β+最小 k-相似算法和β+全相似算法.3 种改进算法需要在求解漏读路径事件的最相似路径事件处对基本策略进行相应的修改.

由此可见,β+改进算法的一个关键步骤是对 $\beta'(o(t),r(t))$ 的求解.算法 2 对求解 $\beta'(o(t),r(t))$ 的欧氏距离算法进行了描述.其中给定的参数包括逻辑区域图 G,滑动窗口大小 S 和窗口滑动的幅度 R.

算法 2. 欧氏距离算法(euclidean distance algorithm).

输入:观察值 o,真实候选值 r;

输出: $\beta'(o(t),r(t))$.

1. establish accurate temporal model M by G and initial time window which size is S over event stream;
2. $sum \leftarrow 0$;
3. **for** $\forall l(l \in o \wedge l \notin r)$ **do**
4. $L_2(l,D(t)) = \left[\sum_{i=1}^{|D(t)|} | l - D_i(t) |^2\right]^{\frac{1}{2}}$;
5. $sum \leftarrow sum + g(L_2(l,D(t)))$;
6. **end for**
7. **if** (slidingHopArrive()) **then** delete expired data in M;
8. **end if**

5 实验评价

5.1 实验设置

由于场地和应用的限制,现有的 RFID 数据查询和清洗文献一般都采用模拟数据进行实验.为了保证模拟效果尽可能接近真实的情况,本文使用了著名的被广泛用于传感设备模拟的 NetLogo 仿真系统,根据 RFID 设备的特点进行了配置.为了考虑应用的复杂性和数据的多样性,本文用 NetLogo 模拟了某个公园的游玩场景:假设该公园共有 30 个景区(即逻辑区域),每个景区布置 2 个阅读器(阅读器的读速率为 5KBPS,每个阅读器配有两个天线),模拟数据在下面 3 种情况下采集得来:

- 半个月内,公园大部分景区正在维修,只有 10 个景区对外开放,其中两个景区上、下午均有节目演出 1 次;
- 工作日,只有 20 个景区全天开放,其他 10 个景区定点开放,开放时间为 8:00~10:00,12:00~14:00,16:00~18:00;
- 节假日期间,公园所有景区全天对外开放.

在上面 3 情况下采集的数据分别被记为 Data1,Data2,Data3.此外,公园的物理区域可以抽象地用图结构 $G(V,E)$来表示,节点表示逻辑区域,边表示逻辑区域间的直接可达性,该图被称为逻辑区域图.公园的入口和出口组成它的开始节点集 V_{start}和终止节点集 V_{end}.由此,当事件 $l_1l_2\ldots l_n$满足式 $\forall i, l_i l_{i+1} \in E, i \in \{1,2,\ldots,n\}$,则为正确事件.所有正确事件组成的集合称为正确事件集,记为 $\hat{E}$.实验设定,β+改进算法中的单调递减函数 $g(x)=1/x$.

本实验的硬件环境是兼容机:2.4GHz 的 Pentium 4 CPU、主存为 512MB、硬盘为 80GB.软件环境是:操作系统为 Windows XP, 编程环境 Visual C++6.0,抽象和填补算法采用 C 编写.实验衡量的算法包括原数据抽象算法 ODAA,改进的数据抽象算法 IDAA,贪婪算法 RA,最小 k-相似算法 MKA 和全相似算法 AKA,作为比较,还引入了目前相关工作中综合性能较好的文献[5]中的 pipeline 算法 PLA.

5.2 数据抽象算法评测

图 5 描述的是在一个布置两个阅读器的逻辑区域内,先后有 500 个标签经过时阅读器采集到的数据量与经过抽象算法处理后的数据量之间的比较.由图 5 可见,数据抽象算法可以将最初采集的数据量减少 40%~50%,

这将极大地降低系统开销,为后续的数据清洗工作创造了条件.

图 6 则分析了影响数据抽象算法的两个因素,即标签个数和标签在一个逻辑区域的停留时间.横坐标表示标签的平均停留时间,随着横轴坐标的增大,每条曲线都有上升的趋势,这是因为停留时间越长,冗余数据量越大,即经数据抽象删除的数据量越大,所以精简率也随之增大.从图中还可看出,在停留时间大致相同的情况下,标签数目越多,精简率越小.这是因为阅读器的读速率是不变的,即无论在它探测的范围内存在多少标签,在一段时间内它采集的数据量是一定的,当标签数目较多时,它随机采集到的有意义的数据量就多,经数据抽象删除的数据量就少,所以精简率小.当标签个数为 10 时,精简率可以高达 90%以上,极大地节省了系统开销.

图 7 分析了漏读率对适当窗口大小的影响.由图可见,随着漏读率的增加,各算法曲线均有上升的趋势,其中 ODAA 曲线上升速度最快,性能最差;而漏读率较高时,窗口较大的算法错误数少些,这是因为随着漏读率的增大,连续漏读的数据个数增多,较大的时间窗口可以对其进行填补;但窗口过大,同样会导致较高的错误数.该实验说明漏读率对适当窗口的选择有一定的影响,应根据具体的应用,选择适当的窗口,保证改进数据抽象算法的高性能.

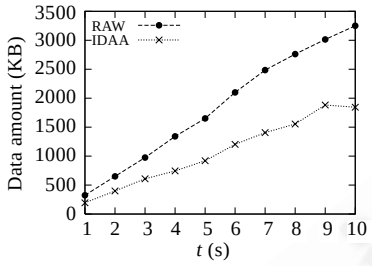


Fig.5 Data amount comparison-1

图 5 数据量比较-1

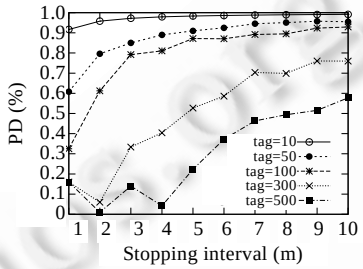


Fig.6 Analysis of effects on PD

图 6 影响精简度的因素分析图

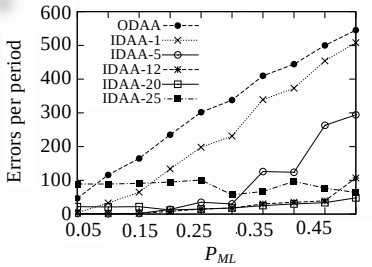


Fig.7 Effect of PM over proper WS

图 7 漏读率对适当窗口大小的影响

5.3 数据填补算法评测

为了说明数据填补策略,对 Data1,Data2 和 Data3 的适当窗口大小和滑动幅度进行了测试(实验图略去).下面的实验,针对不同的模拟数据,将按照实验测得的最适宜窗口大小 WS 和滑动幅度 SR 对窗口大小和滑动幅度进行设置.图 8 是在使用模拟数据 Data1 的情况下测得的,逻辑区域图中节点的出入度 θ=3,逻辑区域漏读率 P_{ML} 取值为 0.1~0.7.从图中可以看出,当 P_{ML}<0.3 时,4 种填补算法均可以达到很高的准确率,几乎接近 100%,但随着逻辑区域漏读率的继续增大,本文提出的 3 种填补算法准确率要明显高于 PLA 算法,这是因为 RA,MKA 和 AKA 算法是基于逻辑区域层对数据进行填补,即当一个标签经过某个逻辑区域,一次也没有被探测到时,这 3 种算法可以根据概率路径事件模型对它进行填补,但 PLA 算法是基于数据层的,当标签在某个逻辑区域完全没有被探测到时,PLA 算法填补不了该标签,所以当逻辑区域漏读率增大时,PLA 算法的准确率下降最快.而本文提出的 3 种算法,AKA 算法的准确率最高,当 P_{ML}<0.7 时,可以保证填补后的数据准确率高达 90%以上,MKA 算法的准确率稍低,RA 算法的准确率最低,与理论分析结果一致.同时,随着漏读率的增大,RA,MKA 和 AKA 算法的准确率均有下降的趋势.这是因为 3 种算法都是在概率路径事件模型的基础上提出的,当漏读率高时,RFID 数据的不可靠性增大,根据统计学习源数据而建立的概率路径事件模型准确率降低,导致算法准确率下降.

图 9 是在使用模拟数据 Data3 的情况下测得的,θ=2,P_{ML} 取值为 0.1~0.8.与图 8 对 Data1 进行测试的结果相比,4 种算法的准确率比较关系相同,但随着漏读率的增大,PLA 的准确率下降更快,而其他 3 种算法的准确率却下降缓慢,准确率很高.这是因为 Data3 引入了 30 个逻辑区域,路径事件长度在 20 以上,而 Data1 只引入了 10 个逻辑区域,路径事件长度小于 10,随着漏读率的增大,路径事件长度较大的 Data3 更容易发生标签被某个区域事件完全漏读的情况,导致 PLA 准确率下降.而当逻辑区域多、路径事件长度长时,标签选择逻辑区域的随机性变大,使得两个事件间的最长公共子事件的长度不会很长,由定理 3 可知,在这种情况下,最小 k-相似算法和全相似算法的准确率会很高.由此可见,在较大规模的应用背景下,本文提出的填补算法在准确率方面优于 PLA.

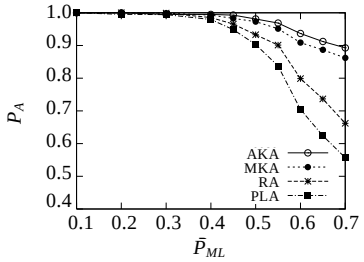


Fig.8 Accuracy comparison-1

图 8 准确率比较-1

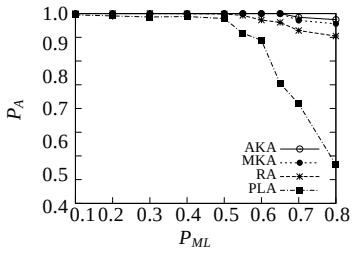


Fig.9 Accuracy comparison-2

图 9 准确率比较-2

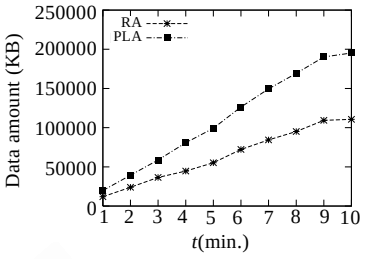


Fig.10 Data account comparison-2

图 10 数据量比较-2

然后,从数据冗余度方面对算法 PLA 和 RA 进行比较,结果如图 10 所示.该图是在使用模拟数据 Data1 的情况下测得的,标签数目为 300.可以看出,RFID 数据经 RA 算法填补后的数据量约是经 PLA 算法填补后数据量的 60%,数据量减少了 40%左右.这是因为 RA 算法是在对 RFID 数据重新建模的基础上进行填补的,即有一个数据抽象过程,对冗余数据进行删除,保证经 RA 算法填补后的数据冗余度很小;而 PLA 算法只是对漏读数据进行填补,不对冗余数据进行处理,所以经过 PLA 算法填补后的数据有很高的数据冗余度.经算法 MKA 和 AKA 填补后的数据,数据量与 RA 曲线也近似相同.由此可见,本文提出的 3 种算法在处理冗余数据方面优于 PLA 算法.

由于 PLA 算法是基于数据层进行数据填补的,没有涉及到事件的概念,与本文提出的实时性不在一个层面上,所以接下来只进行 RA,MKA 和 AKA 这 3 种算法的实时性比较.比较结果如图 11 所示,该图是在使用模拟数据 Data1 的情况下测得的,横坐标表示系统运行时间,纵坐标表示从开始时刻到当前时刻的累计延迟时间.从图中可以看出,贪婪算法的延迟时间最短,实时性最高,在固定时间段内,延迟时间几乎为 0;其次是最小 k-相似算法,表示它的曲线有时平缓,有时上升幅度较大,这说明每个单位时间段内该算法的延迟时间波动较大,这是因为它与具体的到达事件有关,当在正确事件集中找到到达事件的相同事件时,不用再遍历其他事件,认为该事件没有发生漏读,这时延迟时间最短;当遍历所有事件后,才找到与到达事件最相似的事件,这时延迟时间最长.而对于全相似算法来说,对于任何到达事件,它都需要遍历一遍正确事件集,从中找出最相似事件,所以全相似算法的延迟时间最大,实时性最差,曲线几乎呈直线上升,与理论分析结果一致.

以上的实验都是在应用满足式(7)的条件下测得的.可以看出,这种情况下 MKA 算法的性能很好.下面改变应用条件,使其不满足式(7),在此条件下,对算法 MKA 的准确率进行评价,实验结果如图 12 所示.该图是在使用 Data3 的情况下测得的,θ=5,P_{ML} 取值为 0.1~0.8.可以看出,当应用条件不满足式(7)时,算法 PLA,RA 和 AKA 的准确率几乎不受影响,只有 MKA 算法的准确率明显下降,甚至略低于 RA,这是因为当应用条件不满足式(7)时,漏读路径事件更倾向于多漏读区域事件,这时如果再根据最少填补原则进行填补,则会增大错误填补的概率,所以 MKA 算法的准确率下降.由此可见,式(7)是算法 MKA 应用的前提条件,当应用条件不满足该公式时,可以选取 RA 和 AKA 算法对数据进行填补.

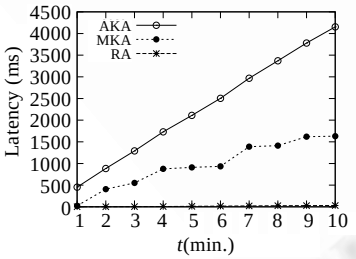


Fig.11 Real-Time comparison

图 11 实时性比较

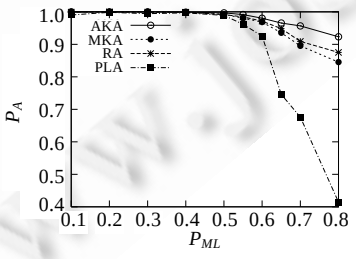


Fig.12 Accuracy comparison-3

图 12 准确率比较-3

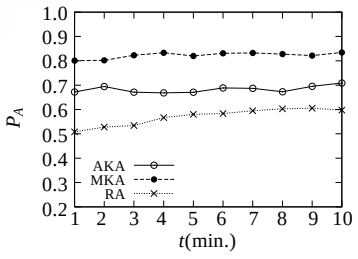


Fig.13 Missing rate factor effect

图 13 漏读率因素的影响

图 13 是分析逻辑区域漏读率对本文提出的 3 种填补算法准确率的影响,是在算法不考虑逻辑区域漏读率的情况下测量的,即发生区域事件漏读时,直接根据路径事件发生率对数据进行填补,使用的实验数据是 Data1,P_{ML}=0.6,其他参数与图 8 相同.与图 8 中 P_{ML}=0.6 时的 3 种算法相比,图 13 中相应算法准确率均有所下降,其中 AKA 算法准确率下降最快.由此可见,漏读率是影响算法准确率的关键因素,尤其对 AKA 算法影响最大.因为当不考虑漏读率时,全相似算法会把所有实时到达的路径事件按照发生率最大的路径事件进行填补.

5.4 改进填补算法评测

首先考察了直方图组距大小对β*改进算法准确率的影响.例如,对β*贪婪算法来说,取 W=5min 的组距会获得比较高的准确率.然后,考察调整因子α对β+改进算法的影响,实验结果如图 14 所示.该实验图测量了漏读率 P_{ML} 取 0.1,0.45,0.6,0.65 的情况.从图中可以看出,当α=0.6 或α=0.7 时,β+贪婪算法可以达到较高的准确率,这说明与时间因素相比,区域事件顺序相关性对填补漏读数据所起的作用更大.而α过大或过小都会降低算法的准确率,当α=0.1 时,算法性能很差,这说明过多根据时间因素对漏读数据进行填补,准确率不会很高;当α=1 时,算法即退化为前面提出的贪婪算法.由此可见,单一的考虑区域事件顺序性或时间因素都比较片面,填补效果不能达到最优,只有二者综合考虑,才能到达最好的效果,这也验证了提出β+改进算法的合理性.α对β+最小 k-相似算法和β+全相似算法的影响与此类似,这里予以省略,下面的实验中设α=0.6.

图 15 是对贪婪算法、β*贪婪算法和β+贪婪算法的准确率进行比较.从图中可以看出,改进的贪婪算法的准确率明显优于原有的贪婪算法,而β+贪婪算法的准确率略高于β*贪婪算法.这是因为改进的数据填补算法增加对时间因素的考虑,在对漏读数据进行填补时,依据更加全面,而此实验数据停留的时间分布服从正态分布,用欧氏距离对时间进行估计比用直方图估计更准确一些.其他两个填补算法与改进算法的比较与此相似,改进算法的准确率要高于原有的算法,而β+改进算法的准确率略高于β*改进算法(实验图省略).

图 16 是改变标签在逻辑区域停留时间的分布情况,对两种改进算法的性能进行衡量.此实验的时间分布服从近似均匀分布.从图中可以看出,β*改进算法的准确率略高于β+改进算法,这是因为当时间分布较为分散时,利用直方图对时间进行估计要比欧氏距离更为准确,与理论分析一致.由此可见,当应用环境不同时,应选择适合的改进算法对数据进行填补,这样才能够保证填补后数据的高准确率.

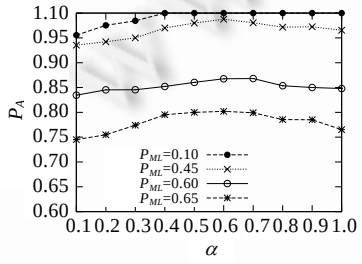


Fig.14 Effect of α over β*RA

图 14 α对β*RA 的影响

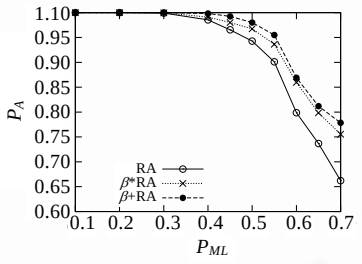


Fig.15 Accuracy comparison-4

图 15 准确率比较-4

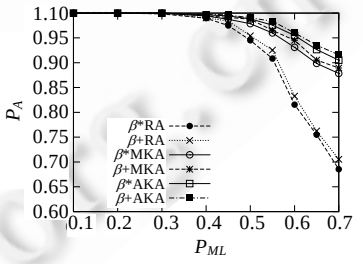


Fig.16 Accuracy comparison-5

图 16 准确率比较-5

6 结 论

一方面,RFID 技术凭借其自动和快速的特点正广泛地应用于越来越多的领域.另一方面, RFID 阅读器采集的数据具有很高的不可靠性和冗余,制约了 RFID 的应用场合,对数据管理提出了新的挑战.针对这些问题,本文对 RFID 数据填补算法进行了深入的研究.首次提出了以逻辑区域事件为填补粒度的基于动态路径事件模型的填补框架.在对 RFID 数据进行三元组模型的基础上,提出了一种数据抽象算法,将 RFID 数据从数据层抽象到逻辑区域层.然后,针对 RFID 应用中数据不可靠性的主要类型——漏读数据,在数据抽象的基础上,提出了 3 种数据填补算法,即贪婪算法、最小 k-相似算法和全相似算法,对实时性、准确性和维护代价进行了权衡.最后增加了对时间因素的考虑,对已提出的填补算法进行了改进,增加了填补结果的准确率.在不同的应用条件下,两种改进算法各有各的优势.大量实验证明了本文提出的数据抽象和各种数据填补策略在不同的情况下有着不同

的性能优势,对于多逻辑区域参与的,带有明显路径信息的应用场景,本文提出的框架要在数据精简性和填补准确性上好于现有的数据填补策略.

本文还有一些问题有待进一步研究:(1) 提出的 3 种填补算法解决的都是路径事件中不包含相同的区域事件问题,如果去掉假设 1,在查找漏读区域事件的相似路径事件时,一个区域事件可能会有多种匹配,如何设计一个优化的匹配算法,并从中求出最相似事件,有待进一步解决.(2) 在增加时间模型对填补算法进行改进时,没有考虑标签依次经过多个逻辑区域时,在每个逻辑区域停留时间的相关性.比如有的游客游玩时比较细致,在每个逻辑区域停留的时间都要比一般的游客长,相反也有的游客性子比较急,在逻辑区域停留的时间都要相对短一些,还有就是游客在开始游玩的时候比较细致,后来由于劳累等因素游玩的时间比较快.由此可见,这种相关性可能是正相关,也可能是负相关.如何确定这种相关性,量化这种相关性将是我们继续研究的一个方向.

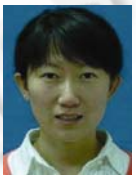
References:

[1] Asif Z, Mandviwalla M. Integrating the supply chain with RFID: A technical and business analysis. Communications of the Association for Information Systems, 2005,15:393−427.
[2] Gu Y, Yu G, Zhang TC. RFID complex event processing techniques. Journal of Frontiers of Computer Science and Technology, 2007,1(3):255−267 (in Chinese with English abstract).
[3] Wang FS, Liu PY. Temporal management of RFID data. In: Bohm K, Jensen CS, eds. Proc. of the 31st Int'l Conf. on Very Large Data Basesn. Trondheim: ACM, 2005. 1128−1139.
[4] Agrawal J, Diao YL, Gyllstrom D, Immerman N. Efficient pattern matching over event streams. In: Wang JT, ed. Proc. of the 2008 ACM SIGMOD Int'l Conf. on Management of Data. Vancouver: ACM, 2008. 147−160.
[5] Jeffrey SR, Alonso G, Franklin MJ, Hong W, Widom J. A pipelined framework for on line cleaning of sensor data streams. In: Liu L, Reuter A, *et al*, ed. Proc. of the 22nd Int'l Conf. on Data Engineering. Atlanta: IEEE Computer Society, 2006. 140−142.
[6] Jeffery SR, Garofalakis M, Franklin M J. Adaptive cleaning for RFID data streams.In: Whang KY, ed. Proc. of the 32nd Int'l Conf. on Very Large Data Bases. Seoul: ACM, 2006. 163−174.
[7] Sarma AD, Jeffery SR, Franklin MJ, Widom J. Estimating data stream quality for object-detection applications. Technical Report, No.UCB/EECS-2005-23, University of California at Berkeley, 2005.
[8] Zhuang YZ, Chen L. In-Network outlier cleaning for data collection in sensor networks. In: Proc. of the 1st Int'l VLDB Workshop on Clean Database. 2006. 41−48.
[9] Khoussainova N, Balazinska M, Suciu D. Towards correcting input data errors probabilistically using integrity constraints. In: Chrysanthis RK, Jensen CS, eds. Proc. of the 5th ACM Int'l Workshop on Data Engineering for Wireless and Mobile Access. Chicago: ACM, 2006. 43−50.
[10] Gonzalez H, Han JW, Li XL. Mining compressed commodity workflows from massive RFID data sets. In: Yu PS, Tsotras VJ, eds. Proc. of the 15th ACM Int'l Conf. on Information and Knowledge Management. Arlington: ACM, 2006. 162−171.
[11] Özsu MT, Golab L. Processing sliding window multi-joins in continuous queries over data streams. In: Freytag JC, Lockemann PC, eds. Proc. of the 29th Int'l Conf. on Very Large Data Bases. Berlin: Morgan Kaufmann Publishers, 2003. 500−511.
[12] Ioannidis Y. The history of histograms. In: Freytag JC, Lockemann PC, eds. Proc. of the 29th Int'l Conf. on Very Large Data Bases. Berlin: Morgan Kaufmann Publishers, 2003. 19−30.
[13] Keogh EJ, Pazzani MJ. Relevance feedback retrieval of time series data. In: Proc. of the 22th Int'l Conf. on Research and Development in Information Retrieval. Berkeley: ACM, 1999. 89−95.

附中文参考文献:

[2] 谷峪,于戈,张天成. RFID 复杂事件处理技术.计算机科学与探索,2007,1(3):255−267.

谷峪(1981－),男,辽宁鞍山人,博士生,CCF 会员,主要研究领域为 RFID 数据管理,数据流.



李晓静(1983－),女,硕士生,主要研究领域为数据流.

于戈(1962－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为数据库理论与技术.

王义(1961－),男,博士,教授,博士生导师,主要研究领域为实时系统.