学号 2013302580051

密级 公 开

武汉大学本科毕业论文

基于 WiFi 和手机传感器的大型商场消费群组行为预测

院（系）名 称：国际软件学院

专 业 名 称 ：软件工程

学 生 姓 名 ：张泽宇

指 导 教 师 ：朱卫平 副教授

二〇一七年四月

BACHELOR'S DEGREE THESIS OF WUHAN UNIVERSITY

Behavior Prediction of Consumer Groups Based on WiFi and Mobile Sensor

College : International School of Software

Major : Software Engineering

Name : Zeyu Zhang

Supervisor : Dr. Weiping Zhu

April 2017

郑 重 声 明

本人呈交的学位论文，是在导师的指导下，独立进行研究工作所取得的成果，所有数据、图片资料真实可靠。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确的方式标明。本学位论文的知识产权归属于培养单位。

本人签名：_____　　日期：_____

摘　要

为了便捷商场消费者在商场内的购物行为以及最大化商场的利润，人们希望能对一个消费群组进行位置预测，并在得到预测位置的同时，判断消费群组在该预测位置可能出现的行为动作（静止、行走或者奔跑），以用于后期的分析。消费群组每经过一个商场区域，系统就会将该区域记录下来，以形成一条区域序列作为历史记录。对一个消费群组的所有区域序列进行关联规则挖掘后，得到的关联规则可以用来预测群组的位置。论文在分析得出以前许多使用 Apriori 算法进行序列模式挖掘（Sequential Pattern Mining）的研究无法对区域序列进行关联规则挖掘的结论后，便改动 Apriori 算法的 apriori-gen()和关联规则生成部分，使之能专门对区域序列进行关联规则挖掘。挖掘出来的关联规则存放在专门设计的树形结构里，以用在本研究里的预测查询过程中。对消费群组进行定位的部分使用位置指纹 WiFi 定位算法，并为了防止生成的区域序列“失真”，还要对 WiFi 定位记录进行异常数据剔除操作。论文使用移动智能手机的三轴加速度传感器进行消费群组的行为动作检测与记录。在对消费群组进行位置预测之后，使用数据库中的历史信息判断消费群组在预测位置可能出现的行为动作，以确定该预测区域对消费群组的吸引力大小。

关键词：行为预测；Apriori 变体；消费群组；WiFi 定位；手机传感器

ABSTRACT

People hope to predict the future location and motion state (e.g. staying in stillness, walking or running etc.) of a consumer group in a shopping mall for the purpose of taking convenience to those consumers shopping in the mall and boosting the earnings of stores. When a consumer group passes through a region in the mall, the system records this region to form an ordered region sequence in which the same region can repeat several times. The association rules mined from all of the ordered region sequences of a consumer group using Apriori algorithm can be used to predict the future location of this consumer group. Former researches that proposed kinds of Apriori-based approaches for mining sequential patterns cannot be used to mine ordered region sequences in my work, so I modify the function apriori-gen() and the association rule discovering step in Apriori algorithm to make it able to mine ordered region sequences. The association rules mined above are stored in a tree structure specifically designed for the association rule indexing and the location prediction process in this work. The fingerprinting-based WiFi positioning approach is used for locating the consumer group, and in order to avoid the "data distortion" of the ordered region sequence, I take measure to eliminate outliers in the positioning data. Then, I use 3-axis acceleration sensor in mobile phone to detect the motion state of the consumer group and record it into the database. After predicting the location of the consumer group, the historical data in the database can be used to decide the motion state of the consumer group in the predicted region for estimating the attraction of the predicted region to the consumer group.

Key words: Behavior Prediction, Apriori Variant, Consumer Group, WiFi Positioning, Mobile Sensor

目　录

1 绪论

在中国，随着百姓生活水平的提高，越来越多人会选择去大型商场消费购物。

为了便捷商场消费者在商场内的购物行为（通过软件指导消费者消费，或者为消费者推荐消费商品来实现）以及最大化商场的利润，商场需要根据消费者的历史行为来预测他们在接下来的时间里可能会出现的消费行为。因此，商场希望能够预测：

①一个商场中的消费群组在接下来的时间里可能出现的位置。

②群组在预测出的位置上的行为动作（如静止、慢走、快走或者跑动等）。

为了预测消费群组的位置，历史数据库中的历史数据可以使用有序的区域序列。事先将商场划分成多个区域。比如一个商场中的一个小商店可以成为一个区域，大一些的商店根据实际情况也可以分成多个区域，还有商店外的过道、楼梯等，也可以划分成一个一个的区域。一般来说，一个区域的最大直径约 4-10 米。当然也可能更长，也可能更短（不过最好不低于 3 米，因为后面提到的 WiFi 定位算法的精度为 2-3 米）。一个群组的历史运动数据可以由多个有序的区域序列L组成，其中$L = (r_1, r_2, r_3, \ldots, r_n)$。$r_n$代表着一个区域。一个$L$也可以被称作一条运动轨迹数据。一个有序区域序列$L$记录的是一个消费群组从进入商城到离开商场进行一次完整消费行为所经过的区域。一个消费群组G的所有的历史区域序列L组成一个历史数据集合H_G，$H_G = \{L_1, L_2, L_3, \ldots, L_n\}$。对$H_G$使用算法即可得到该群组$G$的运动模式。群组的运动模式会有很多种表现方法，而在本篇文章中，群组的运动模式以特殊的关联规则方式呈现，如$(r_1, r_2, r_3) \xrightarrow{0.6} (r_4, r_5, r_6, r_7)$。左侧的$(r_1, r_2, r_3)$代表群组按照$r_1, r_2, r_3$的顺序经过了这三个区域，右侧的$(r_4, r_5, r_6, r_7)$表示依据左侧所经区域进行预测得到的结果，且这个结果说明群组接下来有 0.6 的概率会以r_4, r_5, r_6, r_7的顺序依次经过这四个区域（关联规则中的这个概率被称为置信度“Confidence”）。当有了一个消费群组的所有的关联规则之后，在进行预测的时候，将群组最近经过的区域序列与关联规则的左部区域序列进行对比，如果吻合，就可以把这个关联规则当作候补的预测规则。具体的预测方法将会在文章后面阐述算法具体步骤的地方进行具体讲解。

当预测出群组接下来会出现的区域时，商场会希望知道消费群组在这个预测区域内会进行怎样的行为动作。比如，如果行为是基本趋于静止的，那么这个区域可能有什么东西在吸引着消费群组，而如果行为被识别为慢走或者快走，那么这个区域对于消费者来说可能没什么吸引力。如何识别群组内组员的运动行为，考虑到硬件成本和硬件的普及性，我决定在本论文中使用移动设备（如智能手机或者平板电脑）的传感器进行识别。一个消费群组在某一个区域内可能会留下数个在该区域的历史行为动作数据。这些数据在未来可以用作参考数据，以判断消费群组在以后再次进入该区域时，可能会出现的行为动作。

如何确定消费群组所在的区域，肯定最好是使用室内定位算法（一般来说，室内的 GPS 信号弱，不适合用来定位，而且即使有 GPS 信号也无法进行不同楼层的定位）。本篇文章考虑到定位设施硬件成本和系统实现的可能性，最后决定使用室内 WiFi 定位来确定消费群组的位置。室内 WiFi 定位最高能达到的精确度一般在 2-3 米，足以确定一个消费群组所在的商场区域，因此 WiFi 定位是一个不错的选择。具体使用的室内 WiFi 定位算法在后面的相关部分会有具体详细的阐述。

2 相关研究

第二章将介绍一些现有的针对 WiFi 定位的研究，以及进行运动物体轨迹预测的研究（运动函数、基于模式的位置预测），还有用移动传感器识别人的行为动作状态的研究。

2.1 室内定位

常用的室内定位技术有基于超声波，基于红外线，基于超宽带，以及基于射频识别（WLAN、ZigBee）的定位技术，另外，使用的算法有起源蜂窝小区技术、时间到达法（TOA）、时间到达差法（TDOA）、信号强度法（RSSI）和到达角度差法（AOA）[32]。

考虑到硬件设施的成本和室内定位系统构建的可行性及难易程度，在商场内进行定位可供选择的最好的方法是基于信号强度法（RSSI）的室内 WiFi 定位技术。WiFi 定位最常使用的算法是三角形算法[30]和位置指纹定位算法[33]。文献[31]对比了这两种算法，结论是使用位置指纹定位技术的 WiFi 定位法的定位精度，要远远高于基于三角形算法的 WiFi 定位法的定位精度。而且由于基于位置指纹定位技术的无线定位方法并不需要知道 AP 接入点（Access Point）的位置以及准确的信道模型，因此不管在具体的实施上还是在定位的性能上，其相比于基于三角形算法的定位方法它都具有较大的优越性。所以我在本篇文章中选择使用基于位置指纹定位技术的室内 WiFi 定位方法。文献[29]提出了一种优化的位置指纹定位算法，可以消除 RSS 数据中异常值对定位结果的影响，最终获得更高的定位准确度。位置指纹定位方法的精度一般在 2-3 米之间，微软研发的 RADAR 原型系统就使用了基于位置指纹定位的 WiFi 定位技术[32]。

2.2 运动函数

早期的关于运动物体的位置预测的研究中，人们都是使用运动函数（Motion Function）来进行位置预测。那时，运动函数分成两大类，一个是线性的运动函数

（Linear Motion Function），另一个是非线性的运动函数（Non-Linear Motion Function）。文献[12-15]中将运动物体的运动设想成线性运动，文献[16，17]不仅考虑了物体的线性运动，还考虑了它们的非线性运动。假设一个运动物体在t_0时刻所在的位置为l_0，且此时它的运动速度为v_0，那么使用线性运动函数的方法就会使用线性函数来预测物体在t_q时刻的位置l_q: $l_q = l_0 + v_0 \times (t_q - t_0)$。而使用非线性运动函数的方法则使用更加复杂的数学公式来进行运动物体的位置预测。也因此，非线性预测方法要比线性预测方法更加准确。另外，在非线性运动函数中，预测准确度最高的方法是“递归运动函数”（RMF – Recursive Motion Function）[16]。但是，递归运动函数仅仅适用于近未来的预测（Near Future Prediction），而不适合对未来长远时间（Distant Time）下的一个时刻进行预测。也就是说，如果t_0代表当前的时刻，而我们要预测运动物体在t_q时刻的位置，当$t_q - t_0$较小的时候，这种预测就属于近未来预测，反之，就是非近未来预测（Non-Near Future Prediction）。为了能够进行非近未来预测，后来，就有了基于模式（Pattern）的位置预测方法。

2.3 基于模式的位置预测方法

首先提及隐马尔可夫模型。隐马尔可夫模型（Hidden Markov Model）[34]，文献[18]使用转移概率（Transition Probability）来代表从一个特定状态转移到另一个特定状态的概率。它把一个总随机过程看成一系列状态的不断转移。因此，可以通过构建隐马尔可夫模型，在一系列的区域之间建立起彼此的转移概率，然后利用这些转移概率来预测运动物体未来可能出现的区域。文献[19，20]实现了使用隐马尔可夫模型进行位置预测的方法。但是，根据历史数据构建隐马尔可夫模型的过程较为复杂（文献[34]详细阐述了隐马尔可夫模型三类问题的基本解法），而且随着历史数据的增加，旧的隐马尔可夫模型肯定要被更新，且并没有一个合适且高效的更新算法用来更新隐马尔可夫模型（重新构建隐马尔可夫模型是相当低效且愚蠢的做法）。所以，使用隐马尔科夫模型进行消费群组的位置预测在本篇论文中是不太合适的。

另外，序列模式（Sequential Pattern）这个概念也可以用来呈现物体的运动模式。文献[21，22]定义了序列模式（Sequential Pattern）这个概念，并且提供了用来挖掘

序列模式的比较好的算法。但是，序列模式在本篇论文的研究中并不会被使用。

最后，关联规则就是在本篇文章中被使用的用来呈现物体运动模式的概念。关联规则（Association Rule）也可以用来进行运动物体的位置预测，如文献[23-25]。文献中使用的关联规则形如$(r_i,\ t_1) \xrightarrow{c} (r_j,\ t_2)$。其中，$c$是置信度（Confidence），$r_i$表示物体在$t_1$时刻所在的区域，$r_j$表示物体在$t_2$时刻所在的区域。这个关联规则表示，在$t_1$时刻处在区域$r_i$的运动物体有$c$概率的可能性在$t_2$时刻处于区域$r_j$。在本篇文章中使用的关联规则的形式与$(r_i,\ t_1) \xrightarrow{c} (r_j,\ t_2)$有稍许的不同。本文中使用的关联规则形如$(r_i,\ r_{i+1},\ldots,\ r_{i+m}) \xrightarrow{c} (r_j,\ r_{j+1},\ldots,\ r_{j+n})$，其中，$c$同样为置信度。整个关联规则表示：如果运动物体按照$r_i,\ r_{i+1},\ldots,\ r_{i+m}$的顺序依次经过这$m+1$个区域，那么接下来，该群组有$c$概率的可能性按照$r_j,\ r_{j+1},\ldots,\ r_{j+n}$的顺序依次经过这$n+1$个区域。

2.4 挖掘关联规则的方法

既然已经决定使用关联规则来表示运动物体的运动模式，那么也就需要一个能够从历史数据中挖掘出关联规则的挖掘算法。一个最为经典的也最常用的挖掘关联规则的方法是 Apriori 算法[9]。然而，在本论文的研究中直接使用文献[9]中提到的 Apriori 算法是不可行的，原因在于：

①、文献[9]中原始的 Apriori 算法是对元素集合进行处理的，因为集合中的元素是无序的，因此计算后得到的关联规则是无序的。比如，一个交易（Transaction）是由多个商品（Item）组成的无序集合。多个交易会组成一个交易集合（Transaction Set）。Apriori 算法对交易集合中的每个交易都进行处理和分析，以挖掘出关联规则。这种关联规则是无序的。比如，一个交易是{商品 A，商品 B，商品 C，商品 D}，从其中挖掘出来的关联规则可能有{商品 A，商品 B} $\xrightarrow{c}$ {商品 D}，也可能有{商品 C，商品 D} $\xrightarrow{c}$ {商品 A}等。这种关联规则是子集合向子集合的映射，且左右两部的两个子集合也不存在先后关系。

②、但是在我的研究内容中，我需要对由区域组成的有序且连续的轨迹序列进行处理。比如，对于一个消费群组G的历史数据集合H_G，我对H_G中所有的有序轨迹序列L进行计算和分析，以得到关联规则。这种关联规则是有序的，

是一个子序列向一个子序列的映射，而且左右两部的两个子序列不仅有先后顺序关系，还必须连续。假若有一条轨迹序列$L = (r_1, r_2, r_3, r_4, r_5)$，那么从其中挖掘出来的关联规则可能有$(r_2, r_3) \xrightarrow{c} (r_4, r_5)$或者$(r_1, r_2) \xrightarrow{c} (r_3, r_4, r_5)$，而绝对不可能有$(r_4, r_5) \xrightarrow{c} (r_2, r_3)$或者$(r_1, r_2) \xrightarrow{c} (r_4, r_5)$。因为对于前面两个有效的关联规则，群组$G$是先依次经过区域$r_2$和$r_3$（$r_1$和$r_2$），紧接着再依次经过区域$r_4$和$r_5$（$r_3$，$r_4$和$r_5$）的。所以结果中不可能出现类似$(r_4, r_5) \xrightarrow{c} (r_2, r_3)$，或者$(r_1, r_2) \xrightarrow{c} (r_4, r_5)$的关联规则。因此，直接使用 Apriori 算法是不适合解决我的研究问题的。

值得注意的是，在以前的很多研究中，无论是早期的还是近期的，都有针对有序的元素进行关联规则挖掘的方法被提出。文献[1-8]都是针对有序元素进行关联规则挖掘研究的。但是，它们中却没有一个能针对有序且连续的轨迹区域序列进行挖掘。

比如，在文献[1，2]的研究场景里，商品项（Item）组成的集合S上定义了一个全序关系，不同 Item 之间有着先后关系。一个交易记录T（Transaction）是S的一个子集合，也就是交易记录中的 Item 也是有序的。因此交易记录<Item1, Item2>与交易记录<Item2, Item1>是不同的两个交易。

在[3-7]的研究场景里，商品项（Item）组成的集合S中元素是无序的，因为一个交易记录T也S的一个子集合，因此交易记录T中的元素也是无序的。交易记录(Item1, Item2)与交易记录(Item2, Item1)是相同的两个交易。但是不同交易记录T之间是有先后关系的。形如<$T1$, $T2$, $T3$>的集合是这五篇文献的主要研究对象，它是S的子集，也是一个 Item 集合，但集合<$T1$, $T2$, $T3$>与集合<$T1$, $T3$, $T2$>不是同一个集合，虽然他们包含着完全一样的 Item。

而以上这 7 篇文献[1-7]的研究内容都有一个共同点，那就是，对于它们研究的有序序列来说，一个父序列的子序列元素需要保证先后顺序，但不需要连续。比如，文献[1，2]中，5 阶序列<a, b, c, d, e>的 3 阶子序列不仅可以是<a, b, c>或者<c, d, e>，还可以是<a, c, e>或者<a, b, d>，但不可以是<c, b, a>或者<e, c, a>。序列<a, b, c, d, e>和序列<a, c, f, e>存在着相同的子模式<a, c, e>。文献[3-7]中，父序列<$T1$={a, b, c}, $T2$={d, e}, $T3$={f, g}>的子序列可以是<$t1$, $t2$>，也可以是<$t1$, $t3$>，但不可能是<$t3$, $t1$>或者<$t2$, $t1$>（其中，$t1 \subseteq T1$，$t2 \subseteq T2$，$t2 \subseteq T2$）。序列<$T1$, $T2$, $T3$,

$T4$, $T5$>与序列<$T3$, $T2$, $T1$, $T4$, $T5$>存在着相同的子模式<$t2$, $t4$, $t5$>（其中，$t2 \subseteq T2$，$t4 \subseteq T4$，$t5 \subseteq T5$）。

但是，在我的研究里，要进行关联规则挖掘的有序区域序列的子序列的元素不仅要保证先后顺序，还要保证连续。比如区域序列<r_1, r_2, r_3, r_4, r_5>的子序列可以是<r_1, r_2, r_3>、<r_2, r_3, r_4>、<r_3, r_4, r_5>、<r_2, r_3>或者<r_2>，但不可能是<r_1, r_2, r_5>或者<r_4, r_3, r_2>。区域序列<r_1, r_2, r_3, r_2, r_5, r_4>与区域序列<r_1, r_2, r_3, r_2, r_4, r_5>存在着相同的子模式<r_1, r_2, r_3, r_2>，但不存在相同的子模式<r_1, r_2, r_3, r_2, r_5>。

另外，文献[8]在使用 AprioriTid 算法[9]进行关联规则挖掘后（AprioriTid 算法与 Apriori 算法的应用场景一样，最终同样产生无序子集合向无序子集合映射的关联规则），还多加了一个步骤：删除所有不符合频现区域（Frequent Region）[8]先后顺序的无效的关联规则。但其实，这种两步到位的方法是比较耗时间和空间的，因为算法中计算出冗余的无效关联规则既耗时间又耗空间，删除无效的关联规则也要花时间。那么，相比于这种两步到位的方式，我更希望使用一种能在计算关联规则的过程中，直接过滤无效关联规则的一步到位的方法。

因此，我对原始的 Apriori 算法的“Apriori Candidate Generation”部分和“Discovering Rules”部分进行了一些改动，使之不仅能对我的有序区域序列进行关联规则挖掘，而且还能一步生成子序列向子序列映射的所有有效的关联规则。至于具体的改动内容，其将会在阐述该 Apriori 算法变体的部分被具体地讲解。

2.5 关联规则更新

随着消费群组G的历史数据集合H_G的不断扩充，新的轨迹序列L被加入。这时候，新的关联规则可能就会产生，而旧的关联规则中，可能会有一些不再满足条件而需要被删除。如果重新执行一次 Apriori 算法来计算所有的关联规则，将会极耗时间。文献[10]提出 FUP 算法（Fast Update Algorithm），用来快速更新关联规则。因此，为了提高效率，FUP 算法可以用在本研究更新关联规则的部分。

2.6 关联规则的存储与查询

由 Apriori 算法变体挖掘出来的关联规则可以代表一个消费群组的运动模式，运动模式作为历史参考数据可以用来进行群组的位置预测。那么，用什么样的数据结构来存储消费群组的运动模式，以及如何查询运动模式就成为了一个要解决的问题。

在本篇论文中，运动模式在本质上就是关联规则，存储运动模式就是把挖掘出来的关联规则存放到一种数据结构当中，然后使用一种算法从该数据结构中查询出需要使用的关联规则。文献[8]设计了 TPT（Trajectory Pattern Tree），一种签名树（Signature Tree）[11]的变体，并用 TPT 来存储关联规则。使用签名树可以高效地对关联规则的左部进行相似查询。但是，签名树仅仅适合存储集合向集合映射的关联规则，因为集合中不会有重复的元素出现。而我的研究中要对序列向序列映射的关联规则进行存储。对于序列来说，其中可能会有重复的元素，比如序列(r_1, r_2, r_3, r_1)中r_1就重复了。所以签名树并不适合用在我的论文中。为了能解决我的问题，我需要设计另外一套数据结构进行关联规则存储。受到签名树的启发，如果想要提高查询效率，我也可以选择使用树形结构来存储关联规则。同时为了使该结构能够适用于后期关联规则的特定的查询方法，我设计了一个只针对论文中的查询方法的树形结构构建方式。具体的关联规则树形存储结构和查询方法将会在后面相关部分进行阐述。

2.7 移动手机传感器识别消费者行为动作

一个人的行为动作可以有静止、行走、奔跑、站立、坐下等。早期的研究为了减少计算资源的使用，低计算的复杂性，以及提高行为动作识别的准确度，就会使用多个传感器来记录人物的运动数据。但是，这种传统的基于多传感器的识别系统，不仅笨重、复杂，而且不适用于移动消费环境（现实中一个消费者的身上不可能携带多个传感器，即使能携带，也必须把不同的传感器固定在不同的位置，而且固定后就不能再移动这些传感器了）。文献[27]提出了一种基于单传感器的动作识别方法，它可以解决传统的多传感器识别的问题。这个方法仅仅使用一个三轴加速度传

感器作为动作数据记录工具（目前的移动智能手机都具有三轴加速度传感器），不仅可以判断传感器在人体的哪个部位（比如在裤子口袋，在包里，在上衣的胸前口袋，或者在手里等），而且还能以超过 96%的精确度分辨出一个人坐下、站起、行走、跑动这些不同的动作。

2.8 论文的各部分内容

在第三部分中，我会阐述我所使用的位置指纹 WiFi 定位算法的具体细节，以及记录消费群组轨迹区域序列的具体方法。对生成的轨迹区域序列进行关联规则挖掘的具体方法将会在第四部分进行阐述。论文第四部分阐述针对商场消费群组的位置预测方法。先针对只有一个人的群组，依次提出改动的 Apriori 关联规则挖掘算法，和适用于本研究的关联规则的存储及查询方法（提到查询方法的部分也会直接阐述位置预测的具体内容），然后再将这套预测方法演化为针对包含多人的群组的方法。论文第五部分则阐述判断消费群组在预测位置的行为动作的方法。第六部分则是实验部分，用来分析评估消费群组的位置预测方法和行为动作预测方法的实际效果。最后一部分第七部分用来总结。

3 使用 WiFi 定位生成区域序列

相比于位置指纹 WiFi 定位算法，基于三角形算法的 WiFi 定位方法不需要建立庞大的位置指纹数据库。对于位置指纹 WiFi 定位算法来说，其需要进行大量的位置指纹采集工作。而且，为了使位置指纹定位更加准确，决定如何合理地设置无线接入点（Access Point）的位置以及如何测算接入点信号并进行指纹对比的过程自然又使工作更加繁琐。另外，如果对一个区域进行三角形定位的无线接入点的个数在三个以上，那么该方法不仅能得到更高的精度，而且还更能抗干扰——如果有一个或多个接入点无法正常工作，只要对此区域进行三角形定位的无线接入点的个数仍然大于三的话，那么该系统仍能正常地进行工作。而对于位置指纹 WiFi 定位算法来说，如果有一个或多个无线接入点无法正常工作，那么在一个位置点测得的位置指纹和实际的位置指纹会有较大的出入，以至于影响最终的定位结果。

但也不能说三角形定位算法就一定比位置指纹定位算法要好。三角形定位算法也有自己的缺点，比如，WiFi 信号的强度在不同环境中（不同的温度、不同的空气湿度、不同的磁场强度、不同的无线接入点摆放方向等）的损耗量是不尽相同的，这大大影响了三角形定位算法的精确度。虽然文献[30]提出的基于 RSSI 的三角形质心定位算法可以忽略 WiFi 信号的具体的损耗模型，以减少基于 RSSI 的定位算

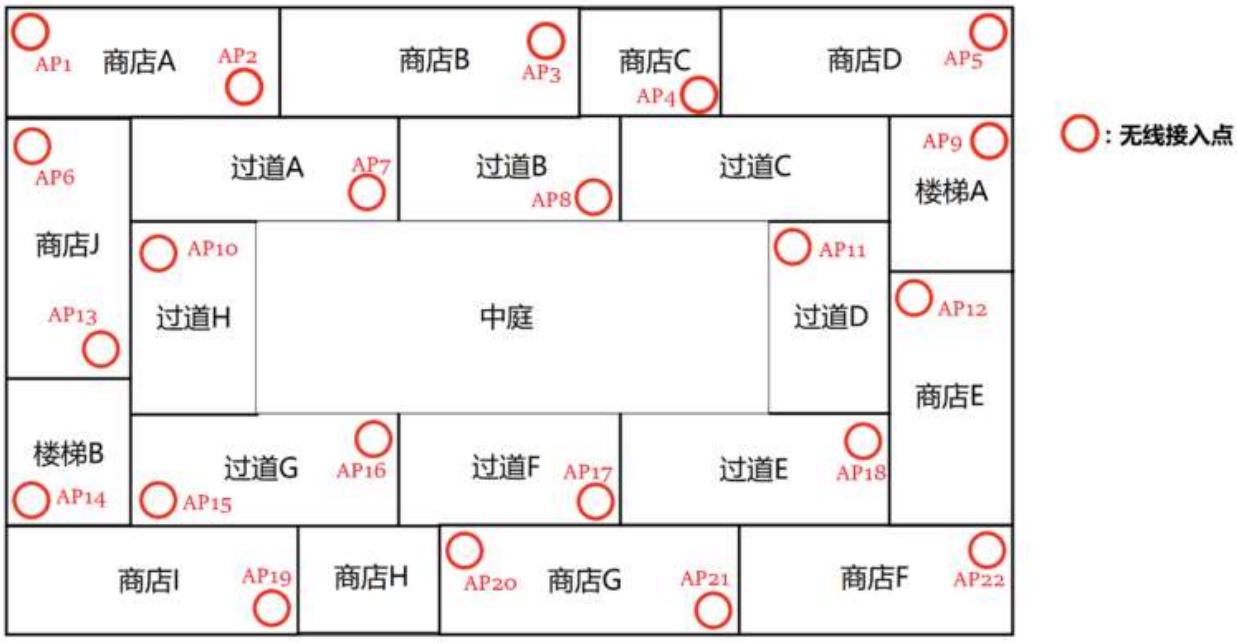

图 **1** 商场无线接入点布局图

法的测量误差，但是该 RSSI 三角形质心定位算法实际上假定了每个无线接入点的信号强度损耗模型是完全一样的。在现实的大型商场中，任何一个无线接入点都可能受到墙壁、商店、楼层支柱的遮挡而拥有与其他接入点完全不同的信号损耗模型。所以三角形定位算法并不适合用在商场内的定位方法里。我在本研究中还是应该选择使用位置指纹 WiFi 定位算法。

在使用位置指纹 WiFi 定位算法的研究中，微软研发的 RADAR 原型系统的定位准确度一般在 2-3 米之间[32]，文献[31]的定位算法实验结果表明其定位精确度一般在 2-7 米（实验结果范围在 1.96-6.84 米之间）之间，文献[29]提出了改良的位置指纹定位算法并得到了定位精度在 1.07-3.49 米之间的实验结果。由于本论文中要给商场划分的各个区域的最大直径长度一般在 3-10 米左右，所以位置指纹 WiFi 定位算法足够确定一个商场中的区域。

为了方便对位置指纹 WiFi 定位方法进行说明，这里我只假定对商场中的一层楼层进行无线接入点的布置。如图 1 所示。

图 2 这个商场内的无线接入点 AP 共有 22 个，它们每个接入点的信号强度可以组成一个向量fp = vec(AP1, AP2, AP3, …, AP22)。这个信号强度向量就是一个位置指纹，也可以叫它位置指纹向量。在商场的每个区域中，都要选择多个参考点进行参考指纹的采集。一个商场区域内有多个位置不同的参考点，每个参考点都有一个参考指纹向量，且一个参考指纹向量中的每个接入点的参考信号强度都是多次采集该接入点的信号强度并取均值得到的。对于一个指纹向量中的所有无法侦测的无线接入点的信号强度，我规定将它们均设置为-100dBm（信号强度的单位是 dBm，而且，-10dBm 相比于-50dBm 来说，-10dBm 表示距离无线接入点近，信号强度大，而-50dBm 表示距离无线接入点远，信号强度小）。

3.1 K 最近邻匹配算法实现位置指纹 WiFi 定位

在商场的指纹数据库中，假设无线接入点 AP 的个数为n，而参考点的个数为m，每个参考点都仅仅属于一个区域r_i，那么可以使用 K 最近邻匹配算法来进行位置指纹的匹配，然后实现定位。

首先，定义位置指纹向量fp_1与位置指纹向量fp_2之间的距离 Distance 为：

$$\text{Distance}(fp_1,\ fp_2) = \sum_{j=1}^{n}(\ | \ s_j - \ S_j \ |^{\omega} \ )^{1/\omega} \qquad (1)$$

其中s_j是fp_1的第j个无线接入点 AP 的信号强度，S_j是fp_2的第j个无线接入点 AP 的信号强度。当ω等于 1 时，距离 Distance(fp_1, fp_2)为曼哈顿距离；当ω等于 2 时，距离 Distance(fp_1, fp_2)为欧几里得距离。

当对一个人所在的位置进行位置指纹定位时，首先测定在该位置上各个无线接入点的信号强度，以组成一个位置指纹向量fp。然后将位置指纹向量fp与所有的参考点的参考指纹向量fp_{ref_m}进行比较，计算所有的距离 Distance(fp, fp_{ref_m})。接着，从小到大排列这些距离，选择前 K 个最小的距离 Distance(fp, fp_{ref_m})，于是，与这 K 个距离相关的 K 个参考点就成为了备选参考点。最后，K 个备选参考点中，在同一商场区域内个数最高的参考点的所在区域r_i就是定位的结果。也就是说，要定位的人他现在被定位到区域r_i内。

另外，文献[29]提出了一种优化的位置指纹定位算法，可以消除 RSS 数据中异常值对定位结果的影响，最终能够获得更高的定位准确度。其使用均值平滑操作对位置指纹向量fp进行处理。该文献提到的优化方法可以直接在我的研究中使用。文献[28]提出了一种可以将高维度的非线性空间里的数据转化成低维度空间数据的方法 ISOMAP。在我的研究里，无线接入点 AP 的个数n就是输入数据空间的维度。位置指纹向量fp就是一个n维空间的数据。当n过大的时候，为了提高位置指纹存储和匹配的效率，可以使用 ISOMAP 方法对位置指纹向量fp进行降维操作。ISOMAP 在需要的时候也可以用在我的研究中。

3.2 过滤定位异常记录，防止轨迹区域序列“失真”

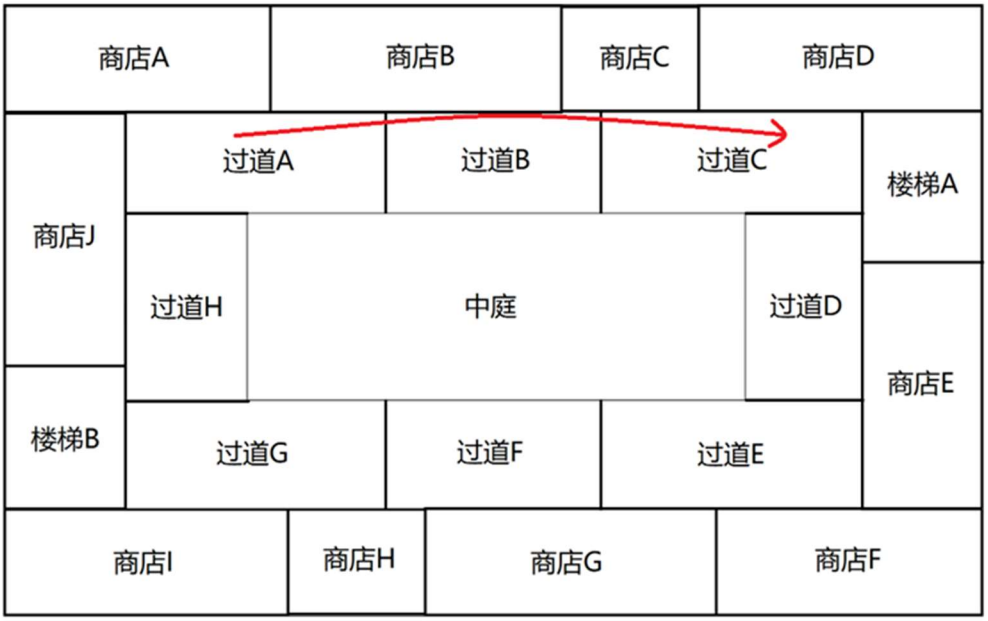

图 **2** 可能产生异常定位数据的场景图

当能够对一个消费者G_1进行室内定位后，就可以在其逛商场的时候记录他的轨迹区域序列L了。在记录消费者G_1的轨迹区域序列L时，可能会遇到以下问题：商场的定位系统是每隔一定的时间间隔（这个时间间隔叫做时间步长 time step）对消费者G_1进行一次定位的。如果在连续的几次定位中消费者G_1都处在同一个区域r_i，那么可以说消费者G_1是一直处在区域r_i的。当下一次定位发现消费者G_1出现在了区域r_{i+1}的时候，就说明G_1离开了区域r_i进入了区域r_{i+1}。这时，区域序列L就可以被记录为(r_i, r_{i+1})。但是，在这种记录方式之下，定位系统带来的误差很可能就会使区域序列L“失真”。如图 2。

消费者G_1按照图中红色箭头的方向依次经过了过道 A、过道 B 和过道 C。在经过过道 B 的时候假设他为了躲避人群而偏向了左侧，以至于过于靠近商店 B。如果定位系统的时间步长 time step 为 2 秒，且在对消费者G_1进行定位的过程中，定位系统出现了一次失误，得到了（过道 A，过道 A，过道 A，过道 A，过道 A，过道 B，过道 B，过道 B，**商店 B**，过道 B，过道 B，过道 B，过道 C，过道 C，过道 C，过道 C，过道 C）这样的一条定位记录。因为这条定位记录中出现了一个异常记录——商店 B，所以，根据这条错误的定位记录得到的轨迹区域序列L“(过道

A, 过道 B, **商店 B**, 过道 B, 过道 C)”也会存在异常元素“商店 B”。这严重与事实不符。因为实际的轨迹区域序列*L*应该是——(过道 A, 过道 B, 过道 C)。错误的轨迹区域序列*L*不仅多出了元素“商店 B”，还把“只经过一次过道 B”的事实变成了“经过两次过道 B”。所以，有必要解决这个问题。解决该问题的方法如下：

给定阈值 min_num。如果一条定位记录中，某个区域连续出现的次数小于 min_num，则将该区域连续出现的这部分直接从定位记录中删除。比如，令 min_num = 3，定位记录（A，A，A，A，B，B，B，B，B，**C**，**C**，D，D，D，D，D，C，C，C，C，C）中加粗部分的连续两个区域 C 的连续个数为 2，小于 min_num，因此需要将这两个区域 C 从定位记录中删除，从而得到处理后的定位记录——（A，A，A，A，B，B，B，B，B，D，D，D，D，D，C，C，C，C，C）。这条处理后的定位记录会得到正确的轨迹区域序列(A, B, D, C)，而不会得到错误的轨迹区域序列(A, B, C, D, C)。

由位置指纹 WiFi 定位获得的定位记录，经过一遍异常数据过滤后，就可以转换成区域序列。得到的区域序列会被放入数据库中，等待之后的关联规则挖掘。在下一章，就会详细阐述针对区域序列进行关联规则挖掘的具体方法。

4 使用区域序列进行位置预测

第三章中生成的区域序列会在本章中被进行关联规则挖掘。挖掘出来的关联规则可以用来进行位置预测。本章先针对一个人的群组提出位置预测法，然后再提出针对多人群组的位置预测法。

4.1 针对一人群组的位置预测

位置预测分为两个部分，一个是用改动的 Apriori 算法挖掘区域序列中的关联规则，另一个是存储关联规则用于预测查询。

4.1.1 改动的 Apriori 算法处理区域序列

针对只有一个人的群组进行的位置预测，可以看成只针对单个运动物体（一个消费者）的位置预测。为了简化算法的研究，我先从对单个运动物体进行位置预测的方式入手。

以下先解释说明与算法相关的一些概念。

●将一个大型商场中消费者任何可能到达的地方划分成多个大小合适的区域r，所有的区域r可以组成一个集合R_{ALL}。

●含有n个成员的消费群组G可以表示为G_n。这里先假设一个群组里只有一个人，所以可以用G_1表示。同时，也仅仅只对这种情况，可以用G_1来表示一个消费者。

●一个消费者G_1的一条运动轨迹可以表示为一个有序区域序列L，$L = (r_1, r_2, r_3, \ldots, r_n)$。$r_n$代表着一个商场区域。一个$L$记录的是消费者$G_1$从进入商城到离开商场进行一次完整消费行为所经过的区域。

●消费者G_1的所有的运动轨迹L能组成一个历史数据集合H_{G_1}，$H_{G_1} = \{L_1, L_2, L_3, \ldots, L_n\}$。

●历史数据集合H_{G_1}中，消费者G_1经过的所有区域r组成一个区域集合R_{G_1}，$R_{G_1} = \{r | \exists L \in H_{G_1}, r \in L\}$。

●关联规则：$(r_i, r_{i+1}, \ldots, r_{i+m}) \xrightarrow{c} (r_j, r_{j+1}, \ldots, r_{j+n})$。左侧的$(r_i, r_{i+1}, \ldots, r_{i+m})$代表消费者$G_1$依次按照$r_i, r_{i+1}, \ldots, r_{i+m}$的顺序经过了这$m+1$个区域，右侧的$(r_j,$

$r_{j+1}, \ldots, r_{j+n}$)表示依据左侧所经区域进行预测得到的结果，且这个结果说明群组接下来有c的概率会以r_j, r_{j+1}, …, r_{j+n}的顺序依次经过这$n+1$个区域。概率c也叫置信度。

●支持度（Support）：①关联规则(r_i, $r_{i+1}, \ldots, r_{i+m}$) $\xrightarrow{c}$ (r_j, $r_{j+1}, \ldots, r_{j+n}$)的支持度s为——H_{G_1}中包含子序列(r_i, r_{i+1}, …, r_{i+m}, r_j, r_{j+1}, …, r_{j+n})的L的个数。②一个子区域序列(r_i, r_{i+1}, …, r_{i+m})的支持度s为——H_{G_1}中包含子区域序列(r_i, $r_{i+1}, \ldots, r_{i+m}$)的L的个数。

●置信度（Confidence）：关联规则(r_i, r_{i+1}, …, r_{i+m}) $\xrightarrow{c}$ (r_j, r_{j+1}, …, r_{j+n})的置信度c为——H_{G_1}中包含子序列(r_i, r_{i+1}, …, r_{i+m}, r_j, r_{j+1}, …, r_{j+n})的L的个数，与H_{G_1}中包含子序列(r_i, $r_{i+1}, \ldots, r_{i+m}$)的L的个数的比值。

●算法需要事先设定两个经验参数，一个是最小支持度 minsup，一个是最小置信度 minconf。

●大子区域序列（这个概念类似于原始 Apriori 算法中的 large itemset 的概念）：如果子区域序列(r_i, $r_{i+1}, \ldots, r_{i+m}$)的支持度s大于等于最小支持度 minsup，那么这个子区域序列就可以被称作“大子区域序列”，反之则被称为“小子区域序列”。

●K 阶大子区域序列：含有 K 个区域元素的大子区域序列被称为 K 阶大子区域序列。

●所有的 K 阶大子区域序列可以组成 K 阶大子区域序列集合LS_K。

●算法中，在由 K-1 阶大子区域序列生成 K 阶大子区域序列的过程中，会产生 K 阶大子区域序列候补项cd_K。这些 K 阶候补项cd_K组成的集合，我用C_K表示。整个改动后的 Apriori 算法可以分为两大步骤：①先计算出所有的大子区域序列，②再根据求出的所有大子区域序列计算出所有的关联规则。

整个改动后的 Apriori 算法可以分为两大步骤：①先计算出所有的大子区域序列，②再根据求出的所有大子区域序列计算出所有的关联规则。

4.1.1.1 计算大子区域序列

（算法的输入参数为消费者G_1的历史数据集合H_{G_1}）

1) LS_1 = {消费者G_1的所有 1 阶大子区域序列};

2) **for** (k = 2; $LS_{K-1} \neq \emptyset$; k++) **do begin**

3) C_K = candidate-gen(LS_{K-1}); // 第一大步骤中对原始算法进行改动的部分就是在这个 candidate-gen()函数里。函数 candidate-gen()的阐述会在后面进行。

4) **forall** $L \in H_{G_1}$ **do begin**

5) **forall** $cd_K \in C_K$ **do begin**

6) **if** L中存在子序列cd_K **then**

7) cd_K.count++; // cd_K.count 在cd_K初始化的时候置为 0。

8) **end**

9) **end**

10) $LS_K = \{cd_K \in C_K | cd_K.\text{count} \geq \text{minsup}\}$;

11) **end**

12) **return** Answer = $\bigcup_K LS_K$;

函数 candidate-gen()的具体改动说明如下：

文献[9]提出的原始的 Apriori 算法中，函数 apriori-gen()就是对本论文里 candidate-gen()函数进行改动前的函数。函数 apriori-gen()分为两步骤：

①、一个是将 K-1 阶 large itemset 组成的集合进行自身与自身的连接（Join）运算，得到所有的 K 阶备选 large itemset。

②、另一个是 prune 步骤，用来删除所有的 K 阶备选 large itemset 中不满足以下条件的 large itemset——备选 large itemset 的任意一个包含 K - 1 个元素的子集，都必须是已经确认的 K-1 阶 large itemset。

在原始的 Apriori 算法中，由 K-1 阶 large itemset 生成的 K 阶备选 large itemset 需要经过两个过滤操作才能最终获得所有有效的 K 阶 large itemset。

①、第一个过滤操作就是上面提到的 prune 步骤，经过该过滤操作后，备选的 K 阶 large itemset 就变成了候补的 K 阶 large itemset。

②、第二个过滤操作是对经过 prune 步骤后得到的候补 K 阶 large itemset 进行支持度检查，所有小于最小支持度的 K 阶候补 large itemset 都会被删除。剩下的所有 K 阶候补 large itemset 就是最终的结果，它们就是所有有效的 K 阶 large itemset。

而函数 apriori-gen()之所以要进行第二步 prune 过程的理由是，一个 K 阶的 large itemset，其包含的任意一个 K-1 阶的子集，也一定是一个 large itemset，即 K-1 阶 large itemset。在第一步 Join 连接操作中，K-1 阶的集合$\{a_i, a_{i+1}, \ldots, a_{i+K-3}, b\}$与 K-1 阶的集合$\{a_i, a_{i+1}, \ldots, a_{i+K-3}, c\}$都是已经确定的 K-1 阶 large itemset，由这两个 K-1 阶集合可以生成备选 K 阶集合$\{a_i, a_{i+1}, \ldots, a_{i+K-3}, b, c\}$。但是这个 K 阶集合$\{a_i, a_{i+1}, \ldots, a_{i+K-3}, b, c\}$的 K-1 阶子集合不再只有$\{a_i, a_{i+1}, \ldots, a_{i+K-3}, b\}$与$\{a_i, a_{i+1}, \ldots, a_{i+K-3}, c\}$这两个 large itemset 了，还有其他 K - 2 个 K-1 阶子集合，它们中有很多不一定是 K-1 阶的 large itemset。因此，函数 apriori-gen()需要进行 prune 操作以删除那些无效的 K 阶备选项。

但是，在改动 Apriori 算法使之能对序列进行关联规则挖掘的过程中，值得注意的一件事是：对于一个 K 阶大子区域序列，其任意的一个 K-1 阶子序列也一定是一个大子区域序列。而且，K 阶序列$(r_i, r_{i+1}, \ldots, r_{i+K-1})$的 K-1 阶子序列只有两个，一个是$(r_i, r_{i+1}, \ldots, r_{i+K-2})$，另一个是$(r_{i+1}, r_{i+2}, \ldots, r_{i+K-1})$。因此，在用两个 K-1 阶大子区域序列生成一个 K 阶区域序列后，就不需要使用 prune 步骤对备选项进行过滤了。生成的 K 阶区域序列可以直接放到函数 candidate-gen()的结果中作为候补项。生成过程为——对于 K-1 阶大子区域序列$L_1 = (r_i, r_{i+1}, \ldots, r_{i+K-2})$与$L_2 = (r_j, r_{j+1}, \ldots, r_{j+K-2})$，如果$L_1$的子序列$(r_{i+1}, r_{i+2}, \ldots, r_{i+K-2})$与$L_2$的子序列$(r_j, r_{j+1}, \ldots, r_{j+K-3})$完全相同（或者$L_1$的子序列$(r_i, r_{i+1}, \ldots, r_{i+K-3})$与$L_2$的子序列$(r_{j+1}, r_{j+2}, \ldots, r_{j+K-2})$完全相同），那么就可以生成 K 阶的候补项序列$(r_i, r_{i+1}, \ldots, r_{i+K-2}, r_{j+K-2})$（或者生成 K 阶候补项序列$(r_j, r_i, r_{i+1}, \ldots, r_{i+K-2})$）。

函数 candidate-gen()的伪代码如下：

1) Answer = ∅;

2) **for** (m = 1; m <= LS_{K-1}.size; m++) **do begin**

3) **for** (n = m + 1; n <= LS_{K-1}.size; n++) **do begin**

4) 由序列$(r_i, r_{i+1}, \ldots, r_{i+K-2})$来表示$LS_{K-1}$.get(m); // LS_{K-1}.get(m)用来获取LS_{K-1}中的第 m 个元素。LS_{K-1}中的元素下标从 1 开始。

5) 由序列$(r_j, r_{j+1}, \ldots, r_{j+K-2})$ 来表示LS_{K-1}.get(n);

6) **if** LS_{K-1}.get(m)的子序列(r_{i+1}, r_{i+2}, …, r_{i+K-2})与LS_{K-1}.get(n)的子序列(r_j, r_{j+1}, …, r_{j+K-3})完全相同 **then**

7) Answer.append((r_i, r_{i+1}, …, r_{i+K-2}, r_{j+K-2}));

8) **end if**

9) **if** LS_{K-1}.get(m)的子序列(r_i, r_{i+1}, …, r_{i+K-3})与LS_{K-1}.get(n)的子序列(r_{j+1}, r_{j+2}, …, r_{j+K-2})完全相同 **then**

10) Answer.append((r_j, r_i, r_{i+1}, …, r_{i+K-2}));

11) **end if**

12) **end**

13)**end**

14)**return** Answer;

接下来将举一个例子解释生成所有大子区域序列的过程：

假设消费者G_1的历史数据集合H_{G_1}中有三个轨迹区域序列(a, b, c, d)、(a, b, c, e, f, g, a)和(a, b, c, e, h, g)，最小支持度 minsup 为 2，那么首先求出其所有的 1 阶大子区域序列。所有的 1 阶大子区域序列为 a、b、c、e 和 g。由这五个 1 阶大子区域序列可生成 25 个 2 阶候补序列 aa、bb、cc、ee、gg、ab、ac、ae、ag、bc、be、bg、ce、cg、eg、ba、ca、ea、ga、cb、eb、gb、ec、gc 和 ge，其中只有 ab、bc、ce 三个是 2 阶大子区域序列。由 ab、bc、ce 可生成 2 个 3 阶候补序列 abc 和 bce。经最小支持度验证，abc 和 bce 均为 3 阶大子区域序列。由 abc 和 bce 可生成 4 阶候补序列 abce。经检验 abce 确实是 4 阶大子区域序列。接下来因为只有一个 abce，所以无法再生成 5 阶候补序列。由此得到所有的大子区域序列 a、b、c、e、g、ab、bc、ce、abc、bce 和 abce。

4.1.1.2 计算关联规则

任意一个已经确定的K（K > 1且K为整数）阶大子区域序列(r_i, r_{i+1}, …, r_{i+K-1})，其都可以生成 K - 1 个候补的关联规则。这些候补关联规则可以表示为(r_i, r_{i+1}, …, r_{i+m}) $\xrightarrow{c}$ (r_{i+m+1}, r_{i+m+2}, …, r_{i+K-1})，其中$m \in [0, K-2]$且m为整数。计算候补关联规则(r_i, r_{i+1}, …, r_{i+m}) $\xrightarrow{c}$ (r_{i+m+1}, r_{i+m+2}, …, r_{i+K-1})的置信度c，如果置信度c

小于最小置信度 minconf，那么就要删除这个候补关联规则。剩下的所有关联规则都是有效的关联规则。

计算关联规则的算法步骤如下：

（算法的输入参数为消费者G_1的所有大子区域序列组成的集合$\bigcup_K LS_K$）

1) Answer = $\emptyset$;

2) **forall** 大子区域序列 $l \in \bigcup_K LS_K$ **do begin**

3) K = l.length;

4) 用序列$(r_i, r_{i+1}, \ldots, r_{i+K-1})$来表示大子区域序列$l$;

5) **for** (m = 0; m < K - 1; m++) **do begin**

6) 关联规则 ar = $(r_i, r_{i+1}, \ldots, r_{i+m}) \xrightarrow{c} (r_{i+m+1}, r_{i+m+2}, \ldots, r_{i+K-1})$;

7) c = 关联规则 ar 的置信度c;

8) **if** c >= minconf **then**

9) Answer.append(关联规则 ar);

10) **end if**

11)**end**

12)**end**

13)**return** Answer;

在挖掘关联规则的过程中需要注意两个比较有用的经验知识。一个是：如果关联规则$(r_i, r_{i+1}, \ldots, r_{i+m}) \xrightarrow{c} (r_j, r_{j+1}, \ldots, r_{j+n})$是有效的关联规则（即它的置信度大于等于 minconf），那么关联规则$(r_i, r_{i+1}, \ldots, r_{i+m}, r_j) \xrightarrow{c} (r_{j+1}, \ldots, r_{j+n})$也一定是有效的。另一个是：如果关联规则$(r_i, r_{i+1}, \ldots, r_{i+m}) \xrightarrow{c} (r_j, r_{j+1}, \ldots, r_{j+n})$是有效的关联规则（即它的置信度大于等于 minconf），那么关联规则$(r_i, r_{i+1}, \ldots, r_{i+m}) \xrightarrow{c} (r_j, r_{j+1}, \ldots, r_{j+t})$（其中$t \in [0, n-1]$且$t$为整数）也一定是有效的。以上两个经验知识在一些给定的应用情景下能对计算关联规则的过程实现较好的速度优化。比如，在不需要知道关联规则置信度c的确定值而只需要知道c的大致范围的情况下，或者在除了给定的关联规则外其他关联规则一律舍弃的情况下，以上两个经验知识能够大大地减少计算过程的计算量。

4.1.2 关联规则存储与预测查询

存储结构的设计是用来提高后期关联规则查询的效率的。

在得到消费者G_1的所有关联规则之后，就可以利用这些关联规则实现位置预测。比如，如果消费者G_1在当前时间之前已经依次区域r_1，r_2，r_3，r_4和r_5，那么，离当前时间最近的经过的区域就是r_5。先在所有关联规则中查找左部为(r_5)的所有关联规则（有则记录下来，没有则继续），然后再找左部为(r_4，r_5)的所有关联规则，接着再找左部为(r_3，r_4，r_5)的所有关联规则……直到之前经过的所有区域记录已经

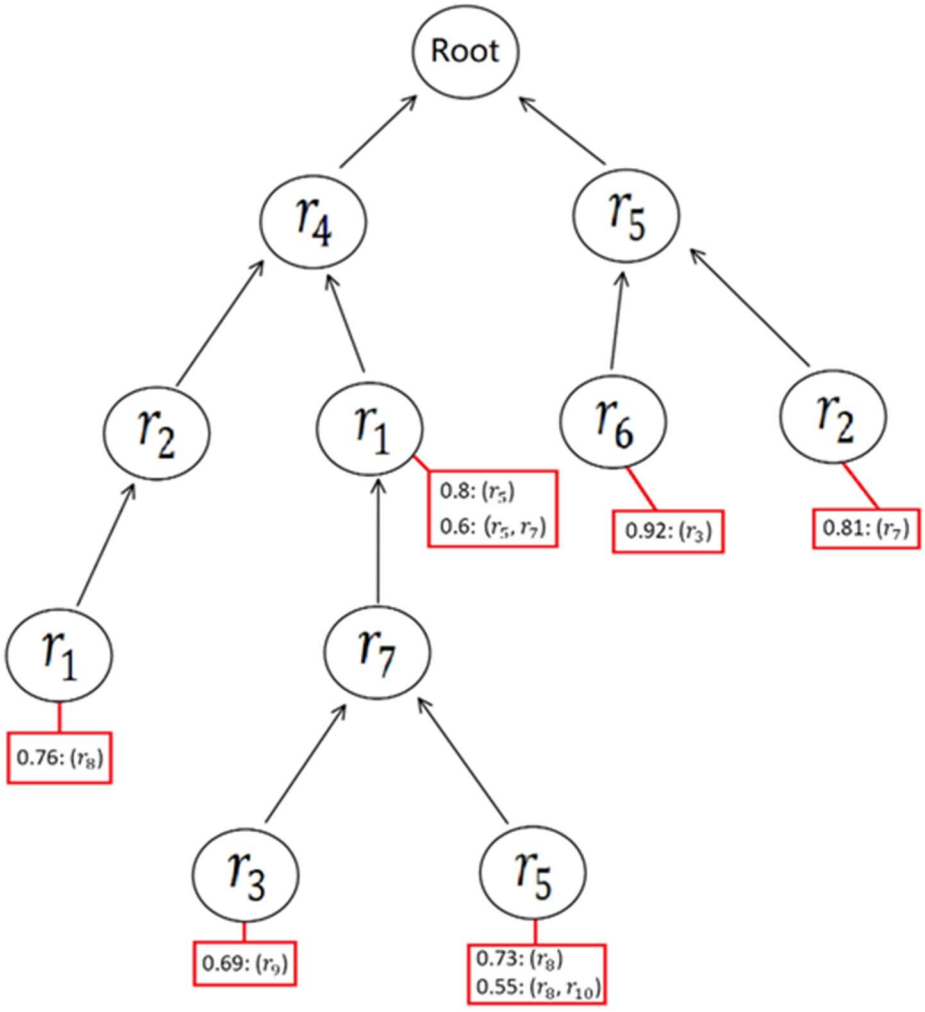

图 **3** 存储挖掘出来的关联规则的树形结构

被查询完，或者关联规则数据库中不再有符合条件的关联规则。在所有的被记录下来的合格的关联规则中，按照给定的查询条件（比如要向后预测至少一定数量的区域）再筛选出符合查询条件的关联规则。然后选择其中置信度c最大的一个关联规

则作为预测标准，比如关联规则$(r_2,\ r_3,\ r_4) \xrightarrow{0.96} (r_8,\ r_9,\ r_{12})$。那么，根据这个预测标准，可以得到预测结果：消费者G_1接下来很有可能会依次经过r_8，r_9和r_{12}这三个区域。

针对以上的查询过程，我设计的关联规则树形存储结构将在下面进行具体的阐述：

如果消费者G_1有关联规则$(r_1,\ r_2,\ r_4) \xrightarrow{0.76} (r_8)$、$(r_1,\ r_4) \xrightarrow{0.8} (r_5)$、$(r_1,\ r_4) \xrightarrow{0.6} (r_5,\ r_7)$、$(r_3,\ r_7,\ r_1,\ r_4) \xrightarrow{0.69} (r_9)$、$(r_5,\ r_7,\ r_1,\ r_4) \xrightarrow{0.73} (r_8)$、$(r_5,\ r_7,\ r_1,\ r_4) \xrightarrow{0.55} (r_8,\ r_{10})$、$(r_6,\ r_5) \xrightarrow{0.92} (r_3)$和$(r_2,\ r_5) \xrightarrow{0.81} (r_7)$共 8 条，那么可以观察到这 8 条关联规则的左部有 6 种完全不同的区域序列。它们分别是区域序列$(r_1,\ r_2,\ r_4)$、$(r_1,\ r_4)$、$(r_3,\ r_7,\ r_1,\ r_4)$、$(r_5,\ r_7,\ r_1,\ r_4)$、$(r_6,\ r_5)$和$(r_2,\ r_5)$。进行关联规则查询的时候，也首要是进行关联规则左部区域序列的完全匹配。提高查询关联规则的效率，也就是要提高查询关联规则左部序列的效率。使用树形结构将所有不同的关联规则的左部序列进行存储是一个可行的选择。于是我可以构建如图 3 的树形结构来存储上面的 8 条关联规则。

我假定根结点 Root 为第 0 层。那么第 1 层的树结点就是关联规则左部区域序列的最后一个元素，而树的叶子结点其实就是关联规则左部区域序列的第一个元素。叶子结点一定是关联规则左部区域序列的第一个元素，但是，关联规则左部区域序列的第一个元素却并不一定是叶子结点。比如，第 2 层从左往右数的第二个树结点就是关联规则$(r_1,\ r_4) \xrightarrow{0.8} (r_5)$左部序列的第一个元素，但它却不是叶子结点。实际上，正如图中所示，关联规则左部区域序列的第一个元素所在的树结点不仅记录了这个区域元素，还会存储红色方框内的内容——关联规则的置信度c和关联规则的右部区域序列。我将这种还会存储关联规则的右部区域序列的树结点称为“关联规则起始树结点”。从任意一个关联规则起始树结点出发，沿着树中箭头的方向，一层一层地连接经过的所有结点直到第一层，所得到的序列就是一个关联规则的完整的左部区域序列。

利用这种树形结构，可以很方便很有效率地实现我所要实现的查询方法，具体方法如下：

在得到消费者G_1的所有关联规则并将它们存储后，我继续假设消费者G_1在当前的希望进行位置预测的时刻t_c以前，依次经过了 6 个区域r_2，r_3，r_5，r_7，r_1和r_4。这

六个区域按顺序组成的序列，是记录下来的已知的经过的区域。区域r_2之前可能还有经过的区域，但是它们没有被记录下来的原因可能是因为区域r_2是消费者G_1刚进入商场时所经过的区域，也有可能是因为服务器中存储的临时历史记录条目数达到上限而需要遗弃、删除这些记录。开始进行查询的时候，先从距离当前时刻t_c最近的区域r_4开始，在树形结构的第 1 层中进行查询，也就是：

首先：在树形结构的第 1 层（即根结点 Root 的所有子结点）中查找有没有含有区域r_4的树结点。显然左边第一个结点就是区域r_4结点。

接着：再在第一层的r_4结点的所有子结点中查找含有区域r_1的树结点。显然，r_4结点的子结点只有一个，而且它刚好是含有区域r_1的树结点，同时它还是关联规则起始树结点。找到了第一个关联规则起始树结点，也就意味着第一次找到可供之后进行位置预测的关联规则。结合已经查找的两个区域r_4和r_1，以及关联规则起始树结点存储的红色方框里的内容，我可以得到两条可供位置预测的关联规则——$(r_1,\ r_4) \xrightarrow{0.8} (r_5)$和$(r_1,\ r_4) \xrightarrow{0.6} (r_5,\ r_7)$。将这两条关联规则存储到结果集中。

接下来：在刚才的关联规则起始树结点的子结点中查找含有区域r_7的树结点。显然存在r_7结点。

然后：在r_7结点的子结点中查找含有区域r_5的树结点。显然从左往右数第二个子结点就是r_5树结点，而且该r_5树结点也是一个关联规则起始树结点。所以，根据已查找的区域和关联规则起始树结点存储的内容，我又可以得到两条可供位置预测的关联规则——$(r_5,\ r_7,\ r_1,\ r_4) \xrightarrow{0.73} (r_8)$和$(r_5,\ r_7,\ r_1,\ r_4) \xrightarrow{0.55} (r_8,\ r_{10})$。也将这两条关联规则存储到结果集中。

最后：继续查询r_5关联规则起始树结点的子结点中有没有包含区域r_3的子结点。显然它并不存在。于是，查询算法到此结束。结果集中最后总共得到了 4 条可供进行位置预测的关联规则。

查询存储关联规则的树形结构以得到可供进行位置预测的关联规则的算法的伪代码如下：

1) Answer = $\emptyset$;

2) T = 储关联规则的树形结构;

3) 消费者G_1在当前时刻之前依次经过的已知的区域用序列$(r_i, r_{i+1}, \ldots, r_{i+m})$来表示;

4) node = T.Root; // node 存储T的根结点。

5) **for** ($k = i+m$; $k >= i$; k--) **do begin**

6) **if** node 的子结点存在包含区域r_k的结点n **then**

7) node = n;

8) **if** node 是关联规则起始树结点 **then**

9) Answer.append(由关联规则左部序列$(r_k, r_{k+1}, \ldots, r_{i+m})$和node 中存储的内容生成的可供进行位置预测的关联规则);

10) **end if**

11) **else**

12) break;

13) **end if**

14)**end**

15)**return** Answer;

由此算法得到的所有可供后期进行位置预测的关联规则，接下来可拿来进行位置预测了。我仍然以上面提到的例子为例阐述位置预测的方法。上面通过关联规则查询已经查到消费者G_1的 4 个可用来进行位置预测的关联规则$(r_1, r_4) \xrightarrow{0.8} (r_5)$、$(r_1, r_4) \xrightarrow{0.6} (r_5, r_7)$、$(r_5, r_7, r_1, r_4) \xrightarrow{0.73} (r_8)$和$(r_5, r_7, r_1, r_4) \xrightarrow{0.55} (r_8, r_{10})$。在进行位置预测的时候，需要提供一个参数$f$来表示希望向前预测的区域的个数。同时，用$mr$表示所有的可用来进行位置预测的关联规则的右部序列中，最长的右部序列的元素个数（比如，在本例中，最长的右部序列为(r_5, r_7)和(r_8, r_{10})，那么mr =2）。如果 1 <= f <= mr，则寻找关联规则右部序列长度为f的所有关联规则，选择其中置信度最大的一条作为位置预测的预测标准，且该预测标准关联规则的右部序列即为预测结果。比如，在本例里，如果f = 1，则有关联规则$(r_1, r_4) \xrightarrow{0.8} (r_5)$和$(r_5, r_7, r_1, r_4) \xrightarrow{0.73} (r_8)$可供选择，且其中$(r_1, r_4) \xrightarrow{0.8} (r_5)$的置信度最大，所以选择它作为预测标准直接得出预测结果——消费者G_1接下来很有可能经过区域r_5；如果f = 2，则有关联规则$(r_1, r_4) \xrightarrow{0.6} (r_5, r_7)$和$(r_5, r_7, r_1, r_4) \xrightarrow{0.55} (r_8, r_{10})$可供选择，且其

中$(r_1,\ r_4) \xrightarrow{0.6} (r_5,\ r_7)$的置信度最大，所以选择它作为预测标准直接得出预测结果——消费者G_1接下来很有可能依次经过区域r_5和r_7。如果$f > mr$，则直接令$f = mr$，接下来的步骤和上面的一样，寻找关联规则右部序列长度为f的所有关联规则，选择其中置信度最大的一条作为位置预测的预测标准。另外，如果在查询关联规则时找不到可用来进行位置预测的关联规则（即历史数据不足），那么可直接使用非线性运动函数中的“递归运动函数”（RMF – Recursive Motion Function）[16]来进行位置预测。

4.2 针对多人群组的位置预测

当消费群组G_n的人数n大于 1 的时候，如何在数据库中存储和表示消费群组G_n的历史数据就成了一个问题。一方面，可以为每个消费者p的个人所有的历史轨迹区域序列数据进行存储。这样，在对一个消费群组G_n进行位置预测的时候，可以根据群组内的每个人的历史数据集合H_p先各进行一次位置预测，然后根据每个人的预测结果，选择结果的相同数量最高的那个结果作为消费群组G_n的位置预测结果。另一方面，也可以单独为消费群组G_n进行历史轨迹区域序列数据的记录。群组G_n作为一个单独的个体，拥有仅属于自己的区域序列L组成的历史数据集合H_{G_n}。

实际上，对于上面提及的两种方式来说，前者并不能有效地对消费群组G_n进行位置预测。原因是，在一个消费群组中，每一个消费者在群组中的影响力都是不同的。大多数组员的共同的习惯并不一定能代表整个消费群组的习惯。领导整个消费群组的人并不一定是这“大多数组员”中的一员。比如，举个典型的例子——一家三口逛商城。假设有出入口区域 A、服饰售卖区域 B 和零食玩具售卖区域 C，对于父亲或者母亲来说，他们各自的习惯一般都是按照 A→B→A 的路径进行运动，然而当他们和年幼的孩子三人一起逛商场时，经常会出现 A→C→A 的运动模式。因为此时整个消费群组的领导消费者是年幼的孩子。所以，一般来说，人们很难根据各组员的个人习惯来分析出整个消费群组的习惯。但是，如果专门为群组G_n记录历史数据的话，那么仅仅属于消费群组G_n的历史数据集合H_{G_n}（H_{G_n}只属于群组G_n，不属于其中的任何一个人或者任何一部分人）就能很好地代表整个群组G_n的习惯。

文献[26]的研究使用了位置指纹 WiFi 定位对室内的徒步群组（Pedestrian Flock）

进行识别。其对徒步群组的定义为：一个徒步群组为一个聚簇，这个聚簇的存留时间$t \geq \tau$，并且该聚簇在存留时间内包含的个体数$n \geq \nu$（其中参数τ和参数ν是人为给定的，针对不同的应用场景，它们会取得不同的值）。因为本篇论文的重点是对消费群组进行位置和行为动作的预测，而不是识别商场内的消费群组，因此如何识别徒步群组（也就是消费群组）的具体算法在本篇论文里不予讨论。不过，文献[26]对“Pedestrian Flock”的定义对于我在本论文中设计记录消费群组G_n的历史数据和查询G_n的历史数据并对其进行位置预测的方法有较大的指导作用。

我们知道，无论是记录消费群组G_n的历史数据还是查询G_n的历史数据，都需要一个群组G_n的 id 或者标签来代表该消费群组。这种 id 或者标签可以使用群组G_n中所有消费者的 id 的集合来表示。表示方法可以有两种，一个是群组完全匹配表示，一个是群组相似匹配表示。比如，对于群组完全匹配表示来说，如果消费群组 A 与消费群组 B 的人员完全相同的话，那么群组 A 和群组 B 就是同一个群组；对于群组相似匹配表示来说，如果消费群组 A 与消费群组 B 的相同人员的个数占各自总人数的百分比大于给定的阈值时，那么群组 A 和群组 B 就代表着同一个群组。针对不同的消费群组规模，就要使用不同的表示方法。比如，对于群组规模在 2-5 人的消费群组来说，少 1-2 个或者多 1-2 个人就很可能会使整个消费群组的习惯发生改变（如上面提到的一家三口逛商场的例子），所以对于这种小规模群组适合使用群组完全匹配表示法；而对于群组规模在 10 人左右的消费群组来说，少 1-2 个或者多 1-2 个人一般不会对该群组的习惯造成影响，因为大规模群组一般都是集体活动，集体性习惯一般不会因为个别人的加入或者离开而受到影响，并且因为大规模群组人数众多，在每次这种集体活动发生的时候，很难保证其所有成员和以前完全相同，因此，针对这种大规模群组，适合使用群组相似匹配表示法。

于是，给定阈值ts，当消费群组G_n的人数$n \leq ts$时，使用群组完全匹配表示法；当群组人数$n > ts$时，使用群组相似匹配表示法。一般，阈值ts的取值范围在 4 到 8 之间。

①、当$n \leq ts$时：

1、记录消费群组G_n的运动数据：

当在商场中识别出消费群组G_n（$n \leq ts$）时，从此时开始记录它的一条轨迹区域序列，直到该群组的人员出现了任何变动。群组内的所有成员的

id 集合代表着此群组的 id。

2、查询消费群组G_n的历史数据：

当要对一个消费群组G_n（$n \leq ts$）进行位置预测的时候，先将群组内的所有成员的 id 集合与数据库中的进行对比。如果找到成员 id 集合完全一样的群组，则将该群组的历史数据取出来进行关联规则挖掘。如果找不到，可以对所有成员的历史数据进行分析并预测他们的个人位置（数据记录时，除了要记录群组G_n的数据，实际上每个人的数据也要记录，只不过记录群组数据和记录个人数据的过程是分开的），然后根据各个成员的预测结果分析群组未来可能出现的位置；或者使用“递归运动函数”直接对消费群组G_n进行位置预测。

②、当$n > ts$时：

1、记录消费群组G_n的运动数据：

给定参数pc。当在商场中识别出消费群组G_n（$n > ts$）时，从此时开始记录它的一条轨迹区域序列，直到群组中剩下的的属于原群组G_n的成员之数，与原群组G_n的人数n的比值，还有与当前群组的人数的比值均小于参数pc。原群组G_n内的所有成员的 id 集合代表着此群组的 id。

2、查询消费群组G_n的历史数据：

使用上面的参数pc。当要对一个消费群组G_n（$n > ts$）进行位置预测的时候，先将群组内的所有成员的 id 集合与数据库中的进行对比。如果，该成员 id 集合与数据库中群组G的成员 id 集合的相同成员 id 个数，与各自成员 id 集合的元素总数的比值均大于参数pc，那么就将该群组G的历史数据取出来。当所有满足以上条件的群组G的历史数据被取出并放在一起之后，接着就可以对其进行关联规则的挖掘。如果找不到任何一个能满足以上条件的群组G，那么可以对G_n的所有成员的历史数据进行分析并预测他们的个人位置（数据记录时，除了要记录群组G_n的数据，实际上每个人的数据也要记录，只不过记录群组数据和记录个人数据的过程是分开的），然后根据各个成员的预测结果分析群组未来可能出现的位置；或者使用“递归运动函数”直接对消费群组G_n进行位置预测。

5 预测消费群组的行为动作

在利用 Apriori 算法的变体对一个消费群组G_n进行位置预测后，便得到了预测的位置区域。接着要预测判断群组G_n在该区域可能会有什么行为动作（比如静止、慢走、快走还是跑动），可以参考群组G_n在该区域的历史行为动作数据。如何记录这些历史行为动作数据，以及如何分析这些行为动作数据，将在下面进行阐述。

为了简化对问题的阐述，这里假设群组里只有一个成员（在对含有多人的消费群组进行行为动作数据的记录时，可以根据每个成员的行为动作的判断结果综合分析生成仅属于该消费群组的行为动作数据）。

要判断消费群组G_1，也就是消费者G_1的行为动作，低成本也最可行的方法是使用消费者G_1的移动手机传感器。文献[27]提出了一种基于单传感器的动作识别方法。这个方法仅仅使用一个三轴加速度传感器作为动作数据记录工具（目前的移动智能手机都具有三轴加速度传感器），不仅可以判断传感器在人体的哪个部位（比如在裤子口袋，在包里，在上衣的胸前口袋，或者在手里等），而且还能以超过 96% 的精确度分辨出一个人坐下、站起、行走、跑动这些不同的动作。该方法主要分为三部分：第一部分是是预处理阶段，从加速度传感器数据中抽取特征值；第二部分系统要判断移动手机在人身上的位置，以选择最合适的行为动作推断方案；第三部分是根据选定的行为动作推断方案识别人的行为动作。因为本论文的研究重点不是如何用传感器识别行为动作，因此文献[27]中的内容不会在此进行赘述。

每隔一个时间间隔对消费者G_1的移动手机的三轴加速度传感器进行采样，取固定数量的采样点（每个采样点都有三个数据分别代表加速度传感器三个轴上的数据）形成一个时间窗口，对时间窗口内的所有采样点进行计算，分别得到这些采样点的方差和其快速傅里叶变换功率谱的最大值。方差主要用来判断人是否静止，而快速傅里叶变换功率谱的最大值则用来判断人是处于行走还是奔跑的状态，甚至可以估算人的奔跑速度。

对消费者G_1处在某个商场区域r_i的时间段内的所有时间窗口里的采样数据进行计算，然后为每个窗口打上“静止”、“行走”和“奔跑”中的一个标签。所有时间窗口内所占数量比例最大的那个标签即为消费者G_1本次在该区域r_i的主要行为动作状态。假设那个标签为“行走”，于是，系统就会添加一条新的记录：(G_1, r_i,

行走, 动作的持续总时间)。动作的持续总时间就是本次在区域r_i里打上“行走”标签的所有时间窗口所占的全部时间之和。

当位置预测系统判断消费者G_1接下来可能经过区域r_i，于是就从数据库中找出消费者G_1在区域r_i的所有记录，选择标签数量最多的那个标签$state$作为G_1行为动作的预测判断结果，并将其所有的动作持续总时间从大到小进行排列。接着对排列好的动作持续总时间的列表进行一次分析，分析多数的动作持续总时间处在哪个值附近。如果值过小，说明消费者G_1在区域r_i以行为状态$state$进行运动的时间比较短；如果值较大，说明消费者G_1在区域r_i以行为状态$state$进行运动的时间较长。分析多数的动作持续总时间处在哪个值附近因为：假设$state$为“行走”，如果值过小，说明在消费者G_1只是匆匆走过该区域；如果值较大，说明消费者G_1在该区域走了很长时间，而且如果该区域是个商店，这说明G_1对该商店会有很大的兴趣。

6 实验

模拟实验部分对模拟商场中的位置指纹定位方法进行了准确度计算，并设计消费者购物场景，模拟生成定位记录，根据定位记录生成轨迹区域序列，接着对其进行关联规则挖掘，然后根据关联规则进行位置预测，之后得出模拟实验结果。最后，本部分还设计了群组性购物情景示例，对含有多人的消费群组的群组匹配方法的合理性做了说明。

6.1 模拟商场中位置指纹 WiFi 定位的准确度

在商场情景中，由于墙体、物品的遮挡，各个无线接入点的信号强度损耗模型均不一样。在模拟时，给定每个无线接入点一个信号强度衰减能力权值。假设无线接入点 A 的信号强度衰减能力权值为 2，无线接入点 B 的信号强度衰减能力权值为 3。在定位检测点 s，s 与 A 的距离为 4，与 B 的距离为 7，则 A 的信号强度从 A 到 s 衰减了 2 * 4 = 8，而则 B 的信号强度从 B 到 s 衰减了 3 * 7 = 21。每个接入点在自身位置的信号强度相同。参考点设置如下图：

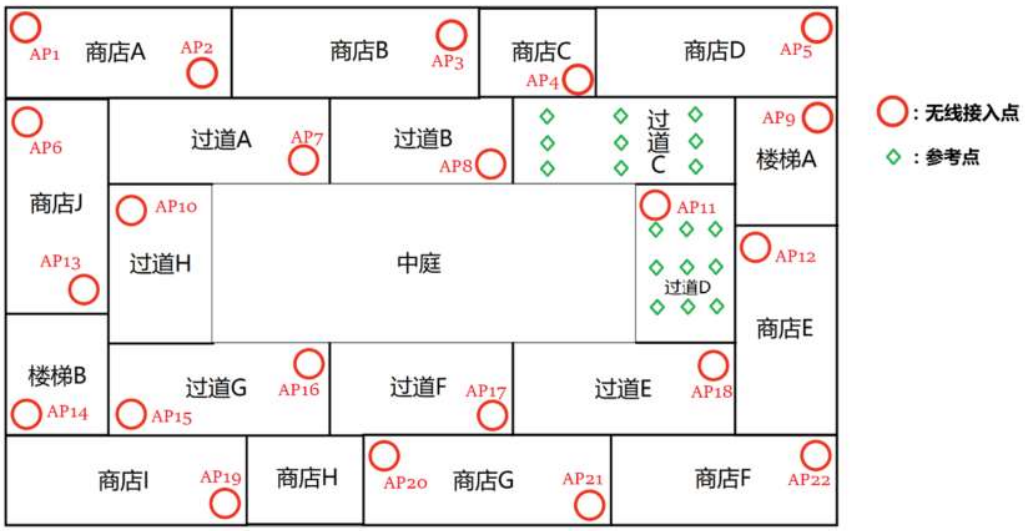

图 **1** 商场无线接入点和参考点布局图

为每个区域等距离设置 9 个参考点，如过道 C 和过道 D 两个区域。其他区域的参考点设置相同。模拟实验测算了所有参考点的位置指纹并将它们放入模拟指纹数据库里。所以，该模拟商场中无线接入点共 22 个，参考点共 20 * 9 = 180 个。

在定位准确度的测试过程中，模拟实验做了五组测试，每组对每个区域的 10 个随机的不同位置进行定位，一组要做 20 * 10 = 200 次定位。成功定位点个数与总定位次数（200 次）的比值即为定位准确度。五组模拟实验的结果见下表：

表 1 商场位置指纹 WiFi 定位模拟实验结果

实验组次	成功定位点个数	总定位次数	定位准确度
实验一	193	200	96.5%
实验二	191	200	95.5%
实验三	198	200	99%
实验四	193	200	96.5%
实验五	194	200	97%

6.2 针对一人群组的位置及行为动作预测模拟

模拟实验需要生成历史数据，并对历史区域序列数据进行关联规则挖掘，对行为动作数据进行查询，以用来预测消费者未来的位置和行为动作。

针对一个消费者的预测系统的结构如图 5：

根据该预测系统的结构图，我设计了一个预测模拟系统和一个消费者消费场景，用来评估针对区域序列的 Apriori 算法变体的效果和判断消费者在特定区域行为动作的系统的效果。

先设置人工数据记录，分别为历史定位记录文件和历史行为动作状态记录文件。打开消费者的历史定位记录文件后，模拟系统便开始按照文件内容模拟定位过程，同时执行异常定位数据过滤操作，生成消费者的区域序列。再对所有的区域序列进行关联规则挖掘，存储在设计好的树形结构里。之后便等待预测系统查询树形结构里的关联规则以便进行位置预测。打开消费者的历史行为动作状态记录文件后，模

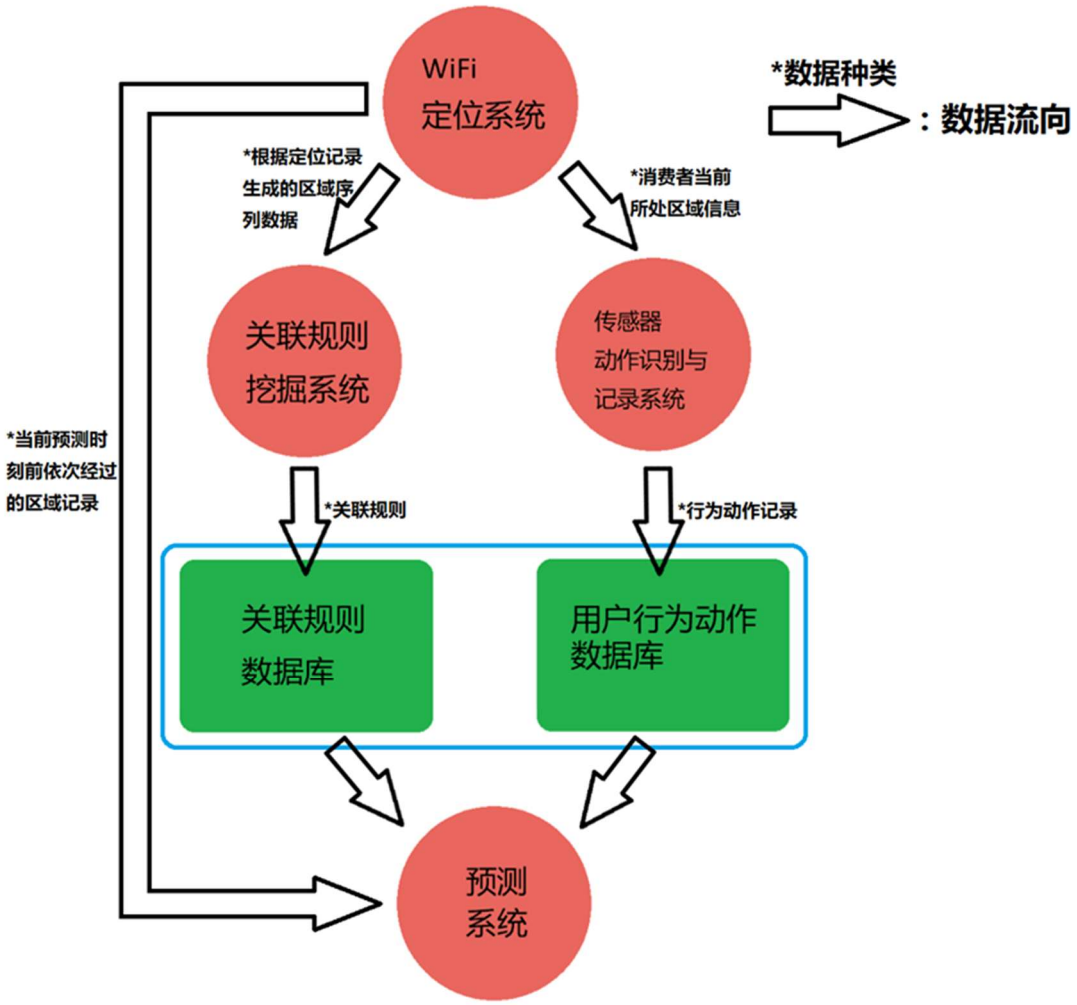

图 **2** 针对一个消费者的预测系统结构图

拟系统便开始按照文件的内容模拟传感器识别过程和行为动作记录过程。接着，等待预测系统查询行为动作数据库，判断在指定区域内消费者最可能出现的行为动作以及持续时长。

这套针对单个消费者的预测模拟系统界面如图 6：

打开历史定位记录文件后，便开始执行区域序列生成模拟和关联规则挖掘模拟。如图 7。

实验设定了以下一个消费者消费场景：

假设一个消费者 A 一年中基本上每周逛两次商场 B。这样，就设定消费者 A 逛商场 B 总次数为 100 次。因此可以得到 100 条 A 的历史区域序列记录。

以图 8 为例，100 条历史区域序列中，30 条以“楼梯 A-过道 D-过道 E-过道 F-过道 G-楼梯 B”的路径完成消费，且其中部分序列会有途经商店 G 的序

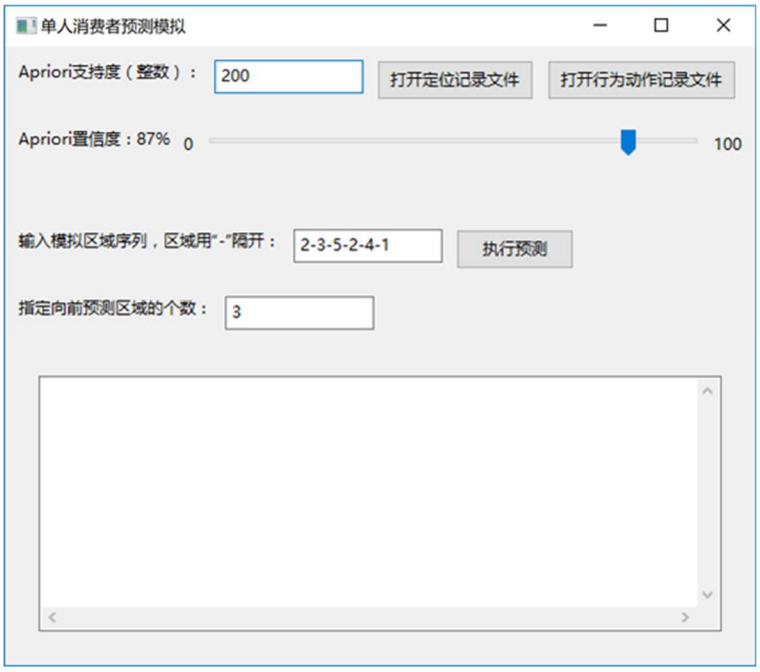

图 **3** 针对单个消费者的预测模拟系统 UI 图

列。70 条以“楼梯 A-过道 C-过道 B-过道 A-过道 H-楼梯 B”，且其中部分序列会有途经商店 B 或商店 J 或同时两个商店的序列。

于是，设定历史区域序列记录：

- “楼梯 A-过道 D-过道 E-过道 F-过道 G-楼梯 B”：10 条。
- “楼梯 A-过道 D-过道 E-商店 G-过道 F-过道 G-楼梯 B”：20 条。
- “楼梯 A-过道 C-过道 B-过道 A-过道 H-楼梯 B”：5 条。
- “楼梯 A-过道 C-过道 B-商店 B-过道 A-过道 H-楼梯 B”：30 条
- “楼梯 A-过道 C-过道 B-过道 A-商店 J-过道 H-楼梯 B”：20 条
- “楼梯 A-过道 C-过道 B-商店 B-过道 A-商店 J-过道 H-楼梯 B”：15 条。

以及各区域行为动作记录：

- 商店 B 被打上“静止”标签 35 次，“行走”标签 10 次。
- 商店 G 被打上“静止”标签 15 次，“行走”标签 5 次。
- 商店 J 被打上“静止”标签 10 次，“行走”标签 25 次。
- 过道 G 被打上“静止”标签 25 次，“行走”标签 5 次。
- 其他每个区域的所有标签均为“行走”。

根据以上的设定，模拟系统在支持度为 10，置信度为 60%，向前预测区域数量为 3 的情况下，输入模拟的行进区域序列“楼梯 A-过道 C”，无法预测。

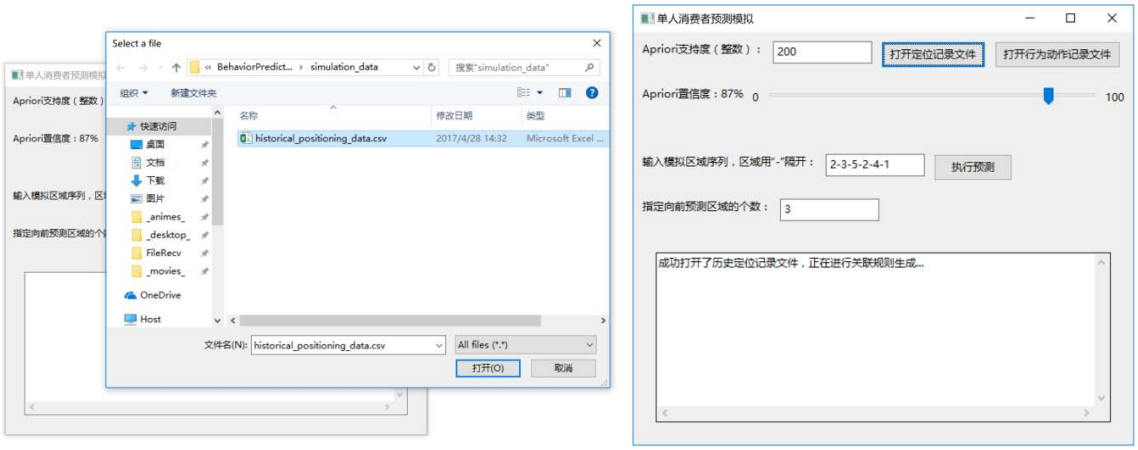

图 **4** 打开历史定位记录文件示例图

将向前预测区域数量降为 1，便可预测到未来出现区域为过道 B。在过道 B 其行为动作判断为“行走”。

如果输入模拟的行进区域序列信息更充足，如“楼梯 A-过道 C-过道 B-过道 A”，支持度和置信度不变，向前预测区域数量为 3，则直接能以置信度 80%预测消费者 A 接下来的行进序列为“商店 J-过道 H-楼梯 B”。在楼梯 B，其行为动作被判断为“行走”。

支持度和置信度不变，向前预测区域数量为 3，输入行进区域序列“过道 D-过道 E”，则以置信度 66.6%预测消费者接下来的行进序列为“商店 G-过道 F-过道 G”。在过道 G，其行为动作被判断为“静止”。当发现一个消费者在过道时的行为动作被判断为静止时，系统就要注意在该过道周围是不是有什么东西在吸引着消费者。

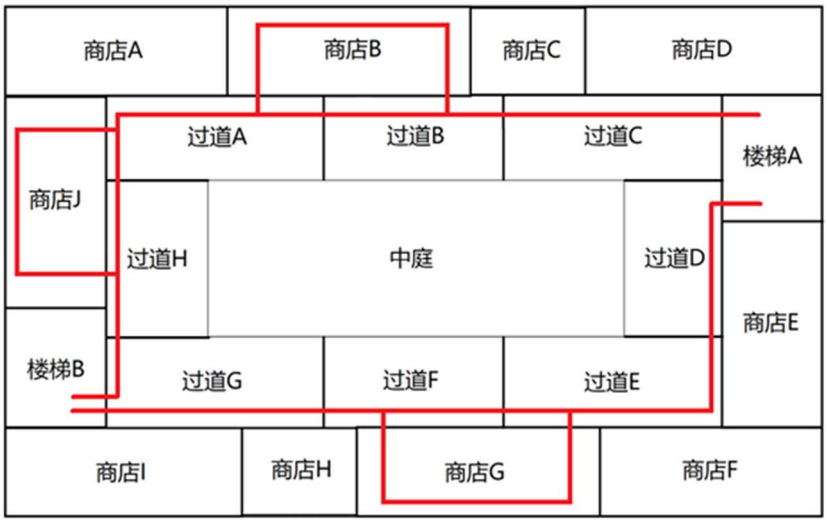

图 **5** 商场内消费者 A 的习惯路径示例图

以上实验的结果见下表：

表 2 预测模拟实验结果

输入的模拟行进序列	支持度	置信度	向前预测区域数量	结果	结果可信度	结果处的行为动作
楼梯 A-过道 C	10	60%	3	无	无	无
楼梯 A-过道 C	10	60%	1	过道 B	100%	过道 B：行走
楼梯 A-过道 C-过道 B-过道 A	10	60%	3	商店 J-过道 H-楼梯 B	80%	楼梯 B：行走
过道 D-过道 E	10	60%	3	商店 G-过道 F-过道 G	66.6%	过道 G：静止

6.3 多人群组群组匹配法的合理性说明

在进行对含多人消费群组的位置及行为动作预测时，除了群组匹配部分外，进行关联规则挖掘和行为动作记录与查询的部分和对单个消费者进行的该部分是相同的。因此，在对多人群组进行预测时，需要单独说明群组匹配方法的合理性。

首先是群组完全匹配。群组人数一般小于 5-8 时采用群组完全匹配。

设定场景：一家五口（孩子、爸爸、妈妈、爷爷和奶奶）。同时设定：

●群组{孩子、爸爸、妈妈、爷爷和奶奶}的路线一般经过餐饮区和儿童或成人服饰区。

●群组{爷爷和奶奶}的路线一般经过保健品区。

●群组{爸爸、妈妈、爷爷和奶奶}的路线一般经过保健品区、成人服饰区。

●群组{爸爸和妈妈}的路线一般经过服饰或者化妆品区。

●群组{孩子、爸爸和妈妈}的路线一般经过餐饮区和儿童服饰区。

那么，在商场中发现其中的一个群组时，选择完全匹配能够使该群组的习惯路线被正确匹配。若是采用相似匹配则会把多种不同的习惯路线混合在一起，以至于

影响预测结果。

接着是群组的相似匹配。群组相似匹配一般是针对 5-8 人以上的群组。

设定场景：班级同学。

对于一个班级的同学，集体活动的情况一是群组人数多，二是集体活动的记录次数少（因为集体活动本身次数就很少），三是组员很难保持完全一致（但是大部分人还是基本不变的）。比如：

●假设群组{1-13}、{1-12}、{2-12}、{2-13}、{1-7, 9, 11-13}都是一个班级的群组，且在某商场内活动时它们的习惯经路线是基本一致的。

如果，在商场中发现其中的一个群组时，只用完全匹配的该群组的历史数据进行预测可能会因为历史数据不足而无法预测；或者新的群组如{2-12}、{2-4, 6-13}被发现，但数据库中并没有它们的历史数据。因此，针对这种大群组，适合使用群组相似匹配法。

群组的完全匹配和相似匹配法的具体内容已经在论文的相关部分进行了具体阐述，在此处则不需赘述。

7 总结与展望

先前使用 Apriori 进行序列模式挖掘的研究仅仅考虑了 Item 的先后关系但并没考虑其是否相邻，因此针对区域序列进行关联规则挖掘时，它们的方法都没法直接使用。而本论文中，在改动了原始 Apriori 的 apriori-gen()和 discovering rules 步骤后，Apriori 就可以用来对区域序列进行关联规则挖掘。专门为本论文的预测方法设计的树形结构用来存储关联规则，可以提高预测算法的效率。在对多人消费群组进行预测时，还要根据群组规模，选择使用群组完全匹配法或者群组相似匹配法。室内 WiFi 定位使用位置指纹定位算法可以消除商场中的物品或者墙壁对接入点信号强度的影响，以提高定位准确度。由定位记录生成区域序列的过程中，还要对定位记录中的异常记录进行过滤，防止生成的区域序列“失真”。使用手机三轴加速度传感器的数据可以判断消费者的行为动作状态，将消费者在某区域的行为状态记录到数据库中，可以在后期使用这些历史数据判断消费者在一个预测区域可能出现的行为动作。

未来，还可以提出新的方法，改变该 Apriori 算法变体的数据存储结构，以提高算法的运算效率。也可以结合消费群组识别系统，设计更加合理的群组匹配算法，以增大群组位置预测和行为动作预测的准确度。

参考文献

[1] Ji C R, Deng Z H. Mining Frequent Ordered Patterns Without Candidate Generation[C]// International Conference on Fuzzy Systems and Knowledge Discovery. IEEE Xplore, 2007:402-406.

[2] Patil K S, Patil S S. Sequential Pattern Mining Using Apriori Algorithm & Frequent Pattern Tree Algorithm[J]. IOSR Journal of Engineering (IOSRJEN), 2013, 3(1): 26-30.

[3] Yonggang W. Sequential Association Rules Based on Apriori Algorithm Applied in Personal Recommendation[J]. International Journal of Database Theory and Application, 2016, 9(6): 257-264.

[4] Yuvanesh P, Arunkumar S. A Horizontal Formatting of Partially-Ordered Sequentialmining with Apriori-Based Algorithm[J]. Middle-East Journal of Scientific Research, 2016, 24(S2): 360-365.

[5] Agrawal R, Srikant R. Mining sequential patterns[C]// Eleventh International Conference on Data Engineering. IEEE Xplore, 1995:3-14.

[6] Nguyen T T, Nguyen P K. A New Approach for Problem of Sequential Pattern Mining[M]// Computational Collective Intelligence. Technologies and Applications. Springer Berlin Heidelberg, 2012:51-60.

[7] Vijayarani S, Deepa S, Vijayarani S, et al. Sequential Pattern Mining - A Study[C]// International Conference on Research Trends in Computer Technologies (ICRTCT), 2013:14-18.

[8] Jeung H, Liu Q, Shen H T, et al. A Hybrid Prediction Model for Moving Objects[C]// IEEE, International Conference on Data Engineering. OAI, 2008:70-79.

[9] Agrawal R, Srikant R. Fast Algorithms for Mining Association Rules[J].

Journal of Computer Science & Technology, 2000, 15(6):619-624.

[10] Cheung W L, Han J, Ng V, et al. Maintenance of Discovered Association Rules in Large Databases: An Incremental Updating Technique[C]// Twelfth International Conference on Data Engineering. IEEE, 1996:106-114.

[11] Mamoulis N, Cheung D W, Lian W. Similarity Search in Sets and Categorical Data Using the Signature Tree[J]. 2003:75-86.

[12] Patel J M, Chen Y, Chakka V P. STRIPES:an efficient index for predicted trajectories[C]// ACM SIGMOD International Conference on Management of Data, Paris, France, June. DBLP, 2004:637-646.

[13] Jensen C S, Lin D, Ooi B C. Query and Update Efficient B+-Tree Based Indexing of Moving Objects[C]// (e)Proceedings of the Thirtieth International Conference on Very Large Data Bases, Toronto, Canada, August 31 - September 3 2004. 2004:768－779.

[14] Šaltenis, Simonas, Jensen C S, Leutenegger S T, et al. Indexing the positions of continuously moving objects[J]. Acm Sigmod Record, 2000, 29(2):331-342.

[15] Tao Y, Papadias D, Sun J. The TPR*-Tree : An Optimized Spatio-Temporal Access Method for Predictive Queries[C]// International Conference on Very Large Data Bases. VLDB Endowment, 2003:790-801.

[16] Tao Y, Faloutsos C, Papadias D, et al. Prediction and indexing of moving objects with unknown motion patterns[C]// ACM SIGMOD International Conference on Management of Data. ACM, 2004:611-622.

[17] Kollios G, Papadopoulos D, Gunopulos D, et al. Indexing mobile objects using dual transformations[J]. The VLDB Journal, 2005, 14(2):238-256.

[18] Rabiner L R. A tutorial on hidden Markov models and selected

applications in speech recognition[J]. Readings in Speech Recognition, 1990, 77(2):267-296.

[19] Ishikawa Y, Tsukamoto Y, Kitagawa H. Extracting Mobility Statistics from Indexed Spatio-Temporal Datasets[C]// Spatio-Temporal Database Management, International Workshop Stdbm'04, Toronto, Canada, August. DBLP, 2004:9-16.

[20] Bhattacharya A, Das S K. LeZi-update:an information-theoretic approach to track mobile users in PCS networks[C]// 1999:1-12.

[21] Yang J, Hu M. TrajPattern: Mining Sequential Patterns from Imprecise Trajectories of Mobile Objects[J]. 2006, 3896:664-681.

[22] Jeung H, Man L Y, Jensen C S. Trajectory Pattern Mining[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2007:330-339.

[23] Yavaş G, Katsaros D, Özgür Ulusoy, et al. A data mining approach for location prediction in mobile environments ☆[J]. Data & Knowledge Engineering, 2005, 54(2):121-146.

[24] Verhein F, Chawla S. Mining Spatio-temporal Association Rules, Sources, Sinks, Stationary Regions and Thoroughfares in Object Mobility Databases[J]. 2006, 3882:187-201.

[25] Tao Y, Kollios G, Considine J, et al. Spatio-Temporal Aggregation Using Sketches[J]. Icde, 2004, 20:214-214.

[26] Kjærgaard M B, Wirz M, Roggen D, et al. Mobile sensing of pedestrian flocks in indoor environments using WiFi signals[C]// IEEE International Conference on Pervasive Computing and Communications. IEEE, 2012:95-102.

[27] Kawahara Y, Kurasawa H, Morikawa H. Recognizing User Context Using Mobile Handsets with Acceleration Sensors[C]// IEEE International Conference on Portable Information Devices, 2007. Portable. IEEE,

2007:1-5.

[28] Tenenbaum J B, Silva V D, Langford J C. A Global Geometric Framework for Nonlinear Dimensionality Reduction[J]. Science, 2000, 290(5500):2319.

[29] 杨帆, 赵东东. 基于 Android 平台的 WiFi 定位[J]. 电子测量技术, 2012, 35(9):116-119.

[30] 林玮, 陈传峰. 基于 RSSI 的无线传感器网络三角形质心定位算法[J]. 现代电子技术, 2009, 32(2):180-182.

[31] 卢恒惠, 刘兴川, 张超,等. 基于三角形与位置指纹识别算法的 WiFi 定位比较[J]. 移动通信, 2010, 34(10):72-76.

[32] 汪苑, 林锦国. 几种常用室内定位技术的探讨[J]. 中国仪器仪表, 2011(2):54-57.

[33] 李昊. 位置指纹定位技术[J]. 山西电子技术, 2007(5):84-87.

[34] 刘河生, 高小榕, 杨福生. 隐马尔可夫模型的原理与实现[J]. 国际生物医学工程杂志, 2002, 25(6):253-259.

致谢

感谢所有帮助过我的同学，所有对我指导过的老师，一直支持着我的家人。是你们对我的教诲、帮助和鼓励，让我才能一直坚持至今。

首先，感谢朱卫平老师对我在论文选题、文献查找以及论文写作和修改方面的各种指导，有了朱老师的详细指导我才可以顺利地完成论文撰写。而且，在这一过程中，我不断学习、领悟到了做学术时应该具有的基本素养和需要的能力。这次的论文撰写对我的创新能力具有非常大的帮助。朱老师的学术能力令我深感敬佩。在其对我指导的过程中，我明白了汲取他人意见的重要性，这能使我及时地改正自己的错误。

另外，还要感谢国际软件学院细心教学的全体老师们，是你们让我体会到只有踏实求学，严谨治学，才能迅速有效地吸收更广泛的专业知识，并将其运用到实际中。你们教会我的不仅仅是专业知识，更多的还是对待学习以及对待生活的正确态度。

然后，感谢武汉大学良好的教育，它培养了我主动学习的能力。感谢学校为我们创造了十分优越的学习环境，有了这个环境我们才可以借阅到很多丰富的参考资料。

最后，感谢大学四年给我的所有，感谢这段美好的时光让我成熟，让我长大。我以我最诚挚的心再次对老师、同学和家人表示感谢。