International Journal of Distributed Sensor Networks, 3: 69–86, 2007

ISSN: 1550-1329 print/1550-1477 online
DOI: 10.1080/15501320601069499

Taylor & Francis
Taylor & Francis Group

On Mining Movement Pattern from Mobile Users

DAVID TANIAR and JOHN GOH

School of Business Systems, Monash University, Clayton, Vic, Australia

In the era in which activities performed by mobile users are tracked through various sensing mechanisms, the movement data collected through these sensors is submitted into a data mining algorithm in order to determine the movement pattern. The movement pattern refers to the pattern that mobile users generally take to move from one base location to another base location through multiple intermediate locations. This paper provides a proposal and case study on how the movement pattern can be extracted from mobile users through transforming the user movement database to the location movement database and subsequently transferred to an algorithm Apriori-like movement pattern (AMP) and movement tree (M-tree). The result is a list of sequences in which mobile users frequently go through that which satisfies min-support and min-confidence. The result of this movement pattern mining exercise opens up a new future for the prediction of the movement for the individual mobile user.

Keywords Mobile Data Mining; Mining Mobile Data; Mobile User Data Mining; Intelligent Mobile Mining; Location Technique; Network Resource Management; Data Mining

1. Introduction

A rapid adoption of information technologies in societies have improved many aspects of our daily lives. With the availability and ever increasing performance in communication technologies, the communication network is ever growing through strong support by consumers who wish to benefit from the ability to stay in touch anytime, anywhere, and the ability to retrieve useful information anytime and anywhere [2]. This led to the wide deployment of wireless service points around the geographical space in order to provide services to the mobile users through the mobile devices they carry [12].

Mobile commerce [26] has led to a wide deployment of mobile services which in turn has led to the need for mobile devices to communicate with the mobile service providers in order to authenticate themselves and receive services. The most widely used mobile service is the mobile phone service. The need to login and authenticate oneself and receive mobile services have led to the ability for mobile service providers to track the identity and location of mobile users, thus providing location dependent information [32] to these users.

Tracking the identity of a mobile user involves identifying the unique mobile user identification number and matches it against the customer database in order to retrieve details such as name, age, gender, and address. The retrieval of such personal data, also known as the mobile user profile, will lead to the identification of mobile users. Even if the mobile device is carried by a different mobile user at different times, the ability to authenticate oneself in the mobile device allows the specific identity to be retrieved.

The tracking location of the mobile user [1] involves the identification of the direction and signal strength from the wireless service station [18, 19, 20]. A more effective means

Address correspondence to David Taniar, School of Business Systems, Monash University, Clayton, VIC 3800, Australia. Tel: 613 9905 9693, Fax: 613 9905 5159. E-mail: David.Taniar@infotech.monash.edu.au

of tracking the location of a mobile user is through the global positioning system (*GPS*) device located in the newer generation of mobile phones. The mobile phone will activate its *GPS* device which will return the current coordinates on earth in latitude and longitude. This location data can then be sent to the wireless service provider to get the accurate location of mobile users [33].

Due to the ability for gathering data from mobile users that cover most aspects of the mobile users' life as mobile moves around the mobile environment, this presents a valuable opportunity for data miners to gather interesting patterns and knowledge through detailed analysis from the data collected from these users. Data mining [24, 30, 31, 34] itself is an established research field, while application of data mining into mobile users' data requires more investigations. Previous work in data mining includes examining the association rule [21], sequential pattern [22], periodic pattern [13–15, 27], spatial pattern [5, 6], spatial association rules [16], and surprising pattern [23] through various algorithms and data structures [17].

2. Background

The mobile environment refers to a boundary in a 3-dimensional geographical space, where the area is serviced by wireless service providers through wireless stations. Wireless service providers are companies which provide services through the mobile device carried by mobile users. These services are generally but not limited to mobile phone and infotainment services. Wireless stations are built around the mobile environment to ensure adequate coverage to meet the bandwidth, processing, and storage requirements for mobile users.

The mobile user in the mobile environment may carry multiple mobile devices. Each mobile device can be identified by a unique physical address. However, the mobile user himself might not be identifiable at current technology. A mobile user may allow his mobile device for a friend to use for two days. It is assumed that mobile users carry their mobile device most of the time.

A mobile device is any electronic device which can be carried by mobile users in which it interacts with the wireless stations in order to provide services to mobile users. This can be mobile phones or personal digital assistants, which provide telecommunication, internet and email, and infotainment services to mobile users. Mobile devices are growing at a rate in which a newer generation of mobile devices are equipped with better processing power, better memory, and better bandwidth access in the mobile network.

The user movement database (*UMD*) is the raw database for movement pattern mining. It consists of a 2-dimensional database in which each column represents the individual mobile user and each row represents individual time, and the duration between each time unit represents an equal duration, whether it is 1 second, 1 minute, or 1 hour. Each cell in the database represents a coordinate which identifies the current position of mobile user at the given time. Table 1 shows an example of the user movement database.

TABLE 1 User movement database (*UMD*)

Time	u_1	u_2	u_3	u_4	u_5
t_1	(x, y)	(x, y)	(x, y)	(x, y)	(x, y)
t_2	(x, y)	(x, y)	(x, y)	(x, y)	(x, y)
t_3	(x, y)	(x, y)	(x, y)	(x, y)	(x, y)
t_4	(x, y)	(x, y)	(x, y)	(x, y)	(x, y)
t_5	(x, y)	(x, y)	(x, y)	(x, y)	(x, y)

User location database (*ULD*) is the database after validation and conversion from user movement database (*UMD*). It is validated to ensure data are present in each cell, and irrelevant data are ignored. Irrelevant data are data, which shows drastic movement between t_n and t_{n+1}. Drastic movement can occur when mobile user is on a transportation vehicle. As the goal of converting *UMD* to *ULD* is to identify the locations which the mobile users have associated themselves with, mobile users sitting in a vehicle cannot be satisfied as properly associating themselves with the particular location they are in. If the mobile user is sitting in a vehicle waiting for a traffic light, it is quite likely that the location is not registered in location of interest database (*LOI-DB*) which will be explained later, thus not returning any location.

Time horizon is the term used to refer the time series in the user movement database (*UMD*). It is the total duration starting from $t=0$ to $t=n$. Time horizon represents the total time duration in which the raw data have been captured. The further the time horizon, the better knowledge the data mining algorithm can produce.

Movement refers to a concept where a subject such as the mobile user, moves from one base location to another base location. Two further concepts of location are the base location and intermediate location. Base location is identified by records that show that the mobile user stayed at a location of interest for a duration greater than max-duration. Intermediate location is identified by records that show that the mobile user stayed at a location of interest for a duration lesser than max-duration. A movement is defined as a sequence of events where a mobile user moves from a base location to another base location through a series of intermediate locations. Table 2 shows a location movement database.

Location movement database (*LMD*) is to sort out the movement sequence of mobile users into a structure suitable for data mining. As *ULD* contains only one sequence for each mobile user, it is necessary to divide this sequence into multiple sequences. In order to do this, a max-duration is defined in which it is the maximum duration in which a mobile user can remain in a particular location before the sequence is separated. Given a sequence for a mobile user before division is {*café* → *bookshop* → *food* → *home* → *café* → *bookshop* → *home*}, and this mobile user spent less than max-duration of time in all places except home, then this sequence will be divided into {*café* → *bookshop* → *food* → *home*} {*home* → *café* → *bookshop* → *home*}.

Location is divided into ***two*** types of location namely

a. ***generic location*** and
b. ***specific location***.

Generic locations are locations which carry names that apply to multiple places. Specific locations are locations which carry names that apply to a single place. Examples of generic location are {*café, mall, food, sports*, and *pool*}. Examples of a specific location are {(10, 20), (10 Smith Street)} which carries an identification code to determine the precise position on the geographical area in a two dimensional representation.

TABLE 2 Location movement database (*LMD*)

Time	u_1	u_2	u_3	u_4	u_5
t_1	*café*	*bookshop*	*bookshop*	*sports*	*cinema*
t_2	*café*	*bookshop*	*bookshop*	*sports*	*cinema*
t_3	*café*	*food*	*café*	*sports*	*café*
t_4	*bookshop*	*food*	*café*	*cinema*	*café*
t_5	*bookshop*	*food*	*café*	*cinema*	*café*

Locations of interest (*LOI*) are generic locations which are applicable for the data mining exercise. It is determined by generating a list of generic locations present in the geographical area which its relationship with other mobile users will be contributory to the generation of useful knowledge and better understanding of mobile users.

Location of interest database (*LOI-DB*) is the database which contains the list of location of interest along with the respective areas within the geographical boundary serviced by wireless points. Each location of interest may contain more than one area of coverage in the mobile environment because locations of interest are a generic location. For example, there can be more than one café around the mobile environment, but they all belong to the same generic location named cafe.

3. Related Work

Group Pattern [28, 29] is a previous work on mining knowledge of grouping relationships among mobile users from a given user movement database (*UMD*). In this study, it was considered that mobile users who frequently spend time with each other, by showing characteristics of being present within the distance of max-duration and for a duration of min-duration of time that occurs frequently, are considered to pose the same relationship among each other [3]. Another extension of the group pattern is through the trajectory approach [25] where partial data are gathered and movement location of mobile users is predicted through linear trajectories when the tracking mechanisms are not in ideal performance.

The static group pattern mining (*SGPM*) [8] is an improvement from the group pattern [28, 29] in which it not only tells the occurrence of a group of mobile users that are close together long and frequently enough, but it also tells the specific location where the group of mobile users meets. This is performed by the technique of first replacing the values of user movement database with a set of pre-defined locations of interests. Location of interests is pre-defined zones in the mobile environment where the location presents some interesting feature in which it is useful for knowledge extraction. For example, the neighborhood library would be a very good candidate of location of interest. The result of *SGPM* is as such: $\{u_1, u_2, u_3, u_4\}$ is a group of mobile users that frequently spends time together at the coffee shop on Blackburn road. It provides a specific location of group pattern occurring.

Mobile user database static object mining (*MUDSOM*) [7] is a method in which it takes into consideration static barriers in the mobile environment which fall into the category of intelligent mobile mining [9–11]. Quite often in the mobile environment there will be static barriers, such as walls, doors, and glass windows in which it separates the relationships among mobile users. Although the location tracking device can determine the current location by pinpointing the exact location that mobile users are currently located in, if they are separated by a wall, glass window or a door, or river, it would be obvious that although they are spending time together, physically near to each other, and satisfying the measurement requirements of the group pattern, it will certainly have no relationship among them at all. *MUDSOM* defines a set of static barriers in the mobile environment, and provides an algorithm in order to track and detect the presence of static barriers through a source to destination walking process. If a static barrier is detected that separates two mobile users, the distance measurement between the two mobile users is disregarded even if the distance is lesser than max-distance. The result of *MUDSOM* is a more accurate result of group pattern.

4. Movement Pattern Mining

The mining movement pattern from the user movement database (*UMD*) involves the following steps which will show in detail how data is transformed to knowledge from the

user movement database. The knowledge, which is the ultimate goal of this data mining exercise, will provide clues to decision makers who can tell the pattern of movement frequently conducted by mobile users to move from one base location to another base location through a set of intermediate locations.

Definition 1 (Mobile Users)

Let M be the list of mobile users as such $\{m_1, m_2, m_3, \ldots, m_n\}$ each element represents an individual mobile user. Each mobile user represents a physical person who carries a mobile device that has the capability of receiving service from the mobile environment, and also the capability of being identified and tracked.

Definition 2 (Location)

Let (x, y) be the coordinate structure where x represent the x axis value or the longitude and y represents the y axis value or the latitude. There is the generic location and the specific location. The specific location is the subset of the generic location where generic location is a collective term for one or multiple specific locations. Let L be the list of generic location of interest as such $\{l_1, l_2, l_3, \ldots, l_n\}$ each element represents a generic location of interest.

Definition 3 (Time Unit)

Let t be the time interval between each equal and uniform time unit between each of the rows in the user movement database and user location database. Variable *max-duration* is defined as such if m stays in location in location l for a duration of time greater than *max-duration*, the location l is deemed to be the base location for the mobile user, while other locations are the intermediate location for the mobile user.

Definition 4 (Sequence)

Let any given structure of $(s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow \ldots s_n)$ be the template for a sequence. A sequence is a list of events happening in the order of a sequence as such $t_{s1} < t_{s2} < t_{s3} < t_{sn.}$ The sequence can be separated into sub sequences when the criteria are met. It is done during the conversion from the user location database (*ULD*) into the location movement database (*LMD*). The separation criteria are such that:

1. If the mobile user m stays in a location l for a duration greater than *max-duration*, the separation will occur, where the current sequence will be terminated as $(\ldots x \rightarrow y \rightarrow z)$ and the next sequence begins with $(z \rightarrow a \rightarrow b \ldots)$. Note that a sub sequence ends with z and the next sub sequence begins with z, as z being the current location where the mobile user m stayed for a duration greater than the *max-duration*.
2. If the mobile user m stays in a location that is not a location defined by location of interest (*LOI*) in the database (*LOI-DB*), for a duration of time greater than *max-duration*, then the mobile user is deemed to have separated from the current sub sequence, and the separation of sequences is done in such manner as $(\ldots x \rightarrow y \rightarrow z)$ and the next sequence beings with $(a \rightarrow b \rightarrow c)$. Note that the ending location of interest of former subsequence is not the same as the beginning location of interest of the new subsequence.

Definition 5

Given the support for a pattern ($x \rightarrow y \rightarrow z$) is $x\%$, pattern ($x \rightarrow y \rightarrow z$) occurs in the given database for $x\%$ of the time. Given the support for a pattern ($x \rightarrow y \rightarrow z$) is x, pattern ($x \rightarrow y \rightarrow z$) occurs in the given database for x number of time.

Definition 6

Given the confidence for a pattern ($x \rightarrow y \rightarrow z$) is $x\%$, it is calculated by the occurrence of all occurrence of ($x \rightarrow y \rightarrow z$) divided by the occurrence of all occurrence of ($x \rightarrow y \rightarrow$*) where * is a wildcard representing any location of interest. The confidence measures the certainty of the pattern by using the ratio of the pattern occurring to the ratio of all potential situations that may lead to pattern occurring. In other words, if confidence is $x\%$ for ($x \rightarrow y \rightarrow z$), it is possible to say that among all occurrence of ($x \rightarrow y \rightarrow$ *), $x\%$ amount of time, it is occurrence of pattern ($x \rightarrow y \rightarrow z$).

Figure 1 shows a situation where in a mobile environment, there are 6 areas and a total of 4 locations of interest (*LOI*) and naturally they are generic locations. Each area is defined by its boundary and their names. Table 3 shows the location of interest database where interesting areas are covered through two set of coordinates that determines the lower left and upper right of the rectangle boundary zone. It is then subsequently named for their interestingness using generic location names, known as location of interest (*LOI*).

Table 4 shows the user movement database. Each row represents a sample of the position coordinate for mobile users. The time units between each row are equal and uniform. Each column in the database represents the individual mobile user. The time unit can be adjusted according to the nature of the data mining exercise. For this case, the time unit is adjusted to 15 minutes per sample. A different application may require more frequent sampling frequency, such as 1 second per sample, or 1 minute per sample.

Table 5 shows a user location database. From user movement database (*UMD*) converted to user location database (*ULD*), each cell in *UMD* is queried against *LOI-DB*, and if a match of location of interest is found, it is replaced with the particular location of interest. For those coordinates that do not match the location of interest, it will be as such

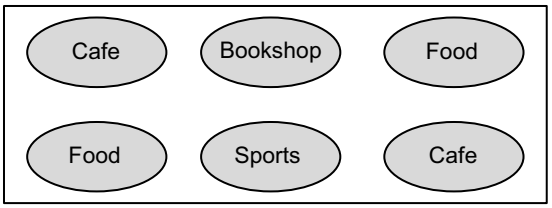

FIGURE 1 Mobile environment with 6 areas and 4 location of interest.

TABLE 3 Location of interest database (*LOI-DB*)

Area	Location of Interest
(1, 1) – (10, 10)	*Food*
(11, 1) – (20, 10)	*Sports*
(21, 1) – (30, 10)	*Café*
(1, 11) – (10, 20)	*Café*
(11, 11) – (20, 20)	*Bookshop*
(21, 11) – (30, 20)	*Food*

TABLE 4 User movement database (*UMD*)

Time	u_1	u_2	u_3	u_4	u_5
t_1	(21, 1)	(11, 11)	(21, 1)	(1, 1)	(1, 11)
t_2	(20, 2)	(12, 11)	(20, 2)	(1, 2)	(2, 11)
t_3	(11, 11)	(11, 11)	(11, 11)	(1, 1)	(11, 11)
t_4	(12, 11)	(12, 11)	(12, 11)	(1, 2)	(12, 11)
t_5	(21, 11)	(11, 11)	(1, 1)	(1, 1)	(1, 1)
t_6	(22, 12)	(12, 11)	(1, 2)	(1, 2)	(1, 2)
t_7	(11, 1)	(11, 11)	(11, 1)	(11, 11)	(11, 1)
t_8	(12, 2)	(12, 11)	(12, 2)	(12, 11)	(12, 2)
t_9	(35, 35)	(1, 1)	(35, 35)	(1, 1)	(35, 35)
t_{10}	(35, 35)	(1, 2)	(35, 35)	(1, 2)	(35, 35)
t_{11}	(35, 35)	(11, 11)	(35, 35)	(1, 1)	(35, 35)
t_{12}	(35, 35)	(12, 11)	(35, 35)	(2, 2)	(35, 35)
t_{13}	(21, 1)	(11, 11)	(21, 1)	(11, 11)	(1, 11)
t_{14}	(20, 2)	(12, 11)	(20, 2)	(12, 11)	(2, 11)
t_{15}	(11, 11)	(11, 11)	(11, 11)	(21, 11)	(11, 11)
t_{16}	(12, 11)	(12, 11)	(12, 11)	(22, 12)	(12, 11)
t_{17}	(21, 11)	(1, 11)	(21, 11)	(21, 11)	(21, 11)
t_{18}	(22, 12)	(2, 11)	(22, 12)	(22, 12)	(22, 12)
t_{19}	(11, 1)	(11, 11)	(11, 1)	(21, 11)	(11, 1)
t_{20}	(12, 2)	(12, 11)	(12, 2)	(22, 12)	(12, 2)

TABLE 5 User location database (*ULD*)

Time	u_1	u_2	u_3	u_4	u_5
t_1	*café*	*bookshop*	*café*	*food*	*café*
t_2	*café*	*bookshop*	*café*	*food*	*café*
t_3	*bookshop*	*bookshop*	*bookshop*	*food*	*bookshop*
t_4	*bookshop*	*bookshop*	*bookshop*	*food*	*bookshop*
t_5	*food*	*bookshop*	*food*	*food*	*food*
t_6	*food*	*bookshop*	*food*	*food*	*food*
t_7	*sports*	*food*	*sports*	*bookshop*	*sports*
t_8	*sports*	*food*	*sports*	*bookshop*	*sports*
t_9	–	*café*	–	*food*	–
t_{10}	–	*café*	–	*food*	–
t_{11}	–	*bookshop*	–	*café*	–
t_{12}	–	*bookshop*	–	*café*	–
t_{13}	*café*	*bookshop*	*café*	*bookshop*	*café*
t_{14}	*café*	*bookshop*	*café*	*bookshop*	*café*
t_{15}	*bookshop*	*bookshop*	*bookshop*	*food*	*bookshop*
t_{16}	*bookshop*	*bookshop*	*bookshop*	*food*	*bookshop*
t_{17}	*food*	*café*	*food*	*food*	*food*
t_{18}	*food*	*café*	*food*	*food*	*food*
t_{19}	*sports*	*bookshop*	*sports*	*food*	*sports*
t_{20}	*sports*	*bookshop*	*sports*	*food*	*sports*

that the coordinate is not a location of interest. This can occur if the mobile user is traveling on a road, waiting for a traffic light, or moving at quite a velocity.

User location database (*ULD*) now needs to be converted into location movement database (*LMD*) which is the suitable format for data mining. The step from converting

user location database to location movement database is to separate the sequences by each individual mobile users at situations where either there is a prolonged absence of location, which occurred 3 times between t_9 to t_{12}, or there is a prolonged presence at a location, which occurred for u_2 at t_1 to t_6, and t_{11} to t_{16}, u_4 at t_1 to t_6, and at t_{15} to t_{20}.

Figure 2 shows a location movement database which is a final product of data processing from the user movement database. This location movement database details the movement sequences that each mobile user has performed based on a sequence where the former occurs earlier than later. In this case, it has essentially summarized multiple traveling records for each mobile user to a point that is necessary and interesting locations are recorded. Each of the subsequence shows how mobile users move from one base location to another through a series of intermediate locations. At this point of time, there is a need to set and explain all the parameters involved in the data mining exercise.

Table 6 shows the parameters required for the mining movement pattern. These are variables in which they need to be pre-determined before movement pattern mining starts as these will be used to distinguish significant and non significant sequences and have a direct relationship with the final output of the algorithm. It provides control by the data miner to determine the level of sensitivity of the algorithm when dealing with a different application environment and a different source data.

Algorithm *AMP* and *W-Tree* differs totally in terms of the approach used to solve the movement pattern mining problem. However, the principle remains the same, where a sequence movement pattern can only occur when it satisfies the min-support and min-confidence criteria. *AMP* employs an *Apriori* approach by generating a list of candidate sequences through identification of frequent locations and performs permutation among those frequent locations and cross count from the location movement database in order to determine their individual support.

The process will continue as long as the frequent sequences which have *support* greater than *min-support* that have length *k* equal or greater than 2 are found. In the end of the mining process, each of the frequent sequences found have their *confidence* calculated in order to determine the level of certainty that the sequence is significant. If the sequence met with *min-confidence* requirement, then it will be present in the final output.

u_1: {(café → bookshop → food → sports), (café → bookshop → food → sports)}.
u_2: {(bookshop → food → café → bookshop), (bookshop → café → bookshop)}.
u_3: {(café → bookshop → food → sports), (café → bookshop → food → sports)}.
u_4: {(food → bookshop → food → café → bookshop → food)}.
u_5: {(café → bookshop → food → sports), (café → bookshop → food → sports)}.

FIGURE 2 Location movement database (*LMD*).

TABLE 6 Movement pattern parameters

Parameter	Description
min-support	Minimum frequency of occurrence required for a pattern.
min-confidence	Minimum certainty of a pattern based on the ratio of chance of pattern occurring and chance of condition occurring.
max-duration	Maximum duration of time before mobile user is considered stationed in a base location. If duration less than max-duration, mobile user is considered stationed in an intermediate location.

Movement tree (*M-Tree*) on the other hand uses a tree in order to transform the location movement database into tree. The rationale behind this is that tree data structures have good capability of being traversed and analyzed. This is because only important relationships are recorded. Important data and relationship are such as the nodes which represents the individual unique location of interest, and the edges which details the direction of the link of sequences among two locations of interest, and with each edge, an identification number and count figure is used to record the number of occurrences for such a relationship.

4.1 Algorithm: Apriori-like Movement Pattern (AMP)

Figure 3 shows the algorithm for Apriori-like Movement Pattern (*AMP*). The Algorithm *Apriori-like* Movement Pattern (*AMP*) exhibits the Apriori nature of solving the problem of mining. It begins by counting the total number of locations of interest (*LOI*) and then uses this list to generate a $k=2$ candidate sequence. This is a list of sequences generated by all the possible combinations derived from the location of interest database. It then filters

```
Input: Location movement database (LMD), min-confidence, min-support, location of interest
database (LOI-DB)
Output: List of movement pattern (M-Pattern)
Algorithm Apriori-like Movement Pattern
01 total-loi = count(LOI-DB);
02 for each loi in LOI-DB do
03   LOI-DB.loi.support = count-support(LMD, LOI-DB.loi);
04   if LOI-DB.loi.support ≥ min-support then
05     push (LOI-DB.loi, candidate);
06   end if
07 end for
08 let k = 1;
09 for each sequence in candidate do
10   generate-candidate-sequence(k + 1, candidate);
11   // generate candidate sequence with length k+1 and store in candidate;
12   for each candidate where k = k + 1
13     count-support(k+1, candidate, candidate.support);
14     if candidate.support < min-support then
15       candidate.current() = false;
16     else
17       candidate.current() = true;
18       found = true;
19     end if
20   end for
21   if found = false() then
22     break;
23   else
24     k = k + 1;
25   end if
26 end for
27 for each sequence in candidate
28   if sequence.status == true then
29     sequence.confidence = count(sequence-1) / count(sequence);
30     if sequence.confidence ≥ min-confidence then
31       output(sequence, support, confidence);
32     end if
33   end if
34 end for
```

FIGURE 3 Algorithm Apriori-like Movement Pattern (*AMP*).

out the frequently occurring sequences based on *min-support*, and removes all the non-frequently occurring sequences which have support lower than *min-support*.

Given a scenario where there are 4 locations of interest namely {*café, bookshop, food, sports*}. There are 4 independently unique locations of interest, while each location of interest may very well represent a few areas within the geographical environment. For each of the locations of interest, they are counted against min-support in order to confirm that they are frequent by themselves.

Let *min-support*=3

Table 7 shows the initial counting for support for each unique location of interest (*LOI*) found from location of interest database (*LOI-DB*). In this case, all {*café, bookshop, food, sports*} satisfies min-support. These will be used to generate candidate sequence with length k=2.

Table 8 shows the counting process for all combinations of candidate sequences where sequence length k=2. There are altogether 12 sequences generated. Note that the order of occurrence of location of interest matters in this mining exercise. In this case, only [{*café* → *bookshop*}, {*bookshop* → *food*}, {*food* → *sports*}] satisfies min-support. These will be used to generate candidate sequence with length k=3. In generation of candidate sequences for length k=3, the requirement is that for all $\{a \rightarrow b\}$ joining with $\{c \rightarrow d\}$, it is only possible to join when b=c such that it will form a proper sequence. If $b \neq c$, then it is not possible for the two length k=2 sequences to be joined to form candidate sequence for further mining.

Table 9 shows the counting process for candidate sequences where length k=3. In this case, both {*café* → *bookshop* → *food*} and {*bookshop* → *food* → *sports*} will be accepted as being part of the set of maximal frequent sequences as their support is greater than min-support. In other words, they have occurred frequently enough in the location movement

TABLE 7 Frequent location of interest

Location of interest	Support
café	9
bookshop	12
food	9
sports	6

TABLE 8 Candidate sequence length k=2

Candidate Sequence	Support
{*café* → *bookshop*}	9
{*café* → *food*}	0
{*café* → *sports*}	0
{*bookshop* → *café*}	1
{*bookshop* → *food*}	9
{*bookshop* → *sports*}	0
{*food* → *café*}	2
{*food* → *bookshop*}	1
{*food* → *sports*}	6
{*sports* → *café*}	0
{*sports* → *bookshop*}	0
{*sports* → *food*}	0

TABLE 9 Maximal sequence length $k=3$

Candidate Sequence	Support
$\{café \rightarrow bookshop \rightarrow food\}$	6
$\{bookshop \rightarrow food \rightarrow sports\}$	6

TABLE 10 Calculation of confidence

Candidate Sequence	Support	Confidence
$\{café \rightarrow bookshop\}$	9	9/9=100%
$\{bookshop \rightarrow food\}$	9	9/10=90%
$\{food \rightarrow sports\}$	6	6/9=66%
$\{café \rightarrow bookshop \rightarrow food\}$	6	6/9=66%
$\{bookshop \rightarrow food \rightarrow sports\}$	6	6/10=60%

database (*LMD*) and are objectively significant enough to be taken into consideration as part of the output of data mining. Until this point, the mining process stops because it is no longer possible to combine $\{café \rightarrow bookshop \rightarrow food\}$ and $\{bookshop \rightarrow food \rightarrow sports\}$ to form a length $k=4$ candidate sequence.

For each of the supported sequences that satisfied min-support requirement, each are checked again and have their confidence calculated. The confidence measure tells the degree of certainty that a particular sequences' occurrence is something that is useful for the decision maker for further investigation. In order to measure a degree of certainty, a measure by chance is used. Confidence is calculated by the ratio of the number of occurrences of the sequence in reference to the number of occurrences of the sequence in reference without the last item.

For example if $\{a \rightarrow b \rightarrow c\}$ is a sequence that is found to be frequent enough due to support greater than min-support, then the confidence of $\{a \rightarrow b \rightarrow c\}$ can be measured by all occurrence of $\{a \rightarrow b \rightarrow c\}$ / $\{a \rightarrow b \rightarrow *\}$ where * represents a wildcard that can be replaced by any location of interest (*LOI*). The following are a list of sequences that have satisfied min-support.

Table 10 shows the process of determining *confidence*. Confidence is determined for each frequent sequence where length is greater or equal to 2. In this case there are 5 frequent sequences with different length and each need to have their confidence determined. The confidence tells how certain it is that the frequent sequence is correct for the final output. It is based on chance of occurrence as such, given all the potential sequences that the frequent sequence may occur through, and there are x amount of occurrences for that particular type of sequence. A *confidence* of $\{bookshop \rightarrow food\}$ tells that given all the movement out from bookshop to something else, 90% of the time goes to food as the immediate next sequence.

4.2 Algorithm: Movement Tree (M-Tree)

The Algorithm Movement Tree (*M-Tree*) simulates the representation of the location movement database (*LMD*) sequences. This provides the opportunity to solve the data mining problem in another data structure which may be more efficient.

Figure 4 shows the algorithm of movement tree (*M-Tree*). The algorithm first builds up the tree by freeing up the w-tree memory space. It reads each individual subsequence

```
Input: Location movement database (LMD), min-confidence, min-support, location
of interest database (LOI-DB)
Output: List of movement pattern (M-Pattern)
Algorithm Movement Tree
01 free(w-tree);
02 for each sub-sequence in location movement database do
03    k = 0;
04    if read.sequence(a, b) ≠ nil then
05       if w-tree(k, a) == nil then
06          w-tree.create(k, a);
07       end if
08       if w-tree(k, b) == nil then
09          w-tree.create(k, b);
10       end if
11       if edge(k, a, b) == nil then
12          w-tree.create-edge(k, a, k+1, b);
13       end if
14       w-tree.edge(k, a, k++, b)++;
15    end if
16 end for
17 for all edges(a, b) in w-tree do
18    if edge.count < min-support then
19       w-tree.remove(b);
20       w-tree.remove(edge);
21    end if
22 end for
23 tree-confidence(w-tree).display();
```

FIGURE 4 Algorithm Movement Tree.

from location movement database (*LMD*) and gradually builds up the tree through examining whether the node has already been created. If it not has been created, relevant nodes are created at the relevant depth as indicated by k which can also be used to represent the relative position of the current location of interest in the subsequence. After the tree is built, it is processed through removing all nodes and edges where *count* < *min-support*. As a result, only significant nodes and edges remain. The final step of the mining process is to calculate the confidence and generate sequences that satisfy *min-support* and *min-confidence* as final output.

During the movement pattern mining process, each node in the tree represents a location of interest. The depth of the tree is important. The following tree has a *depth* level of 6. This shows that the longest sequence in location movement database have a length $k=6$. Each layer represents the sequential order or the position of the location of interest in the movement sequence for mobile users from base location to intermediate location.

The root of the following tree is "*nil*" and is linked to 3 nodes namely "*café, bookshop*, and *food*" at the first level of the tree. Each node is connected via an edge that is directional in order to show the next connection of the sequence. Each edge is associated with a counter which shows the frequency of occurrence of such sequences that the edge has linked.

The tree is built up from the location movement database (*LMD*), through submission of each sub sequences for each mobile users, and gradually all mobile users. Each time

when a sequence is encountered and depending on the position in the length of the sequence, new nodes may be created, and new edges may be created and counter of edges are incremented by 1. An initial location movement database is as such obtained from the location movement database in the background section.

Figure 5 shows an example of the movement tree building process. There are three sub figures from left to right named (*a*), (*b*), and (*c*). They are generated in sequence. Figure 5 (*a*) can be generated through parsing the following sequence {*bookshop* → *food* → *café*}. Figure 5 (*b*) can be generated through parsing the following sequence {*bookshop* → *food* → *sports*}. Figure 5 (*c*) can be generated through parsing the following sequence {*bookshop* → *food* → *café*}. For each time a sequence is read, there is an increment of 1 over the counter of each edge.

Figure 6 shows the raw movement tree generated by Movement tree algorithm. Movement tree algorithm takes the location movement database (*LMD*), *min-support, min-confidence* and transforms the location movement database into a tree representation. This tree does have a root which is "nil," similar to the hierarchical structure of the domain name system used on the Internet where the highest node is "*nil*." The tree has a hierarchical structure. In this tree, the depth of the tree is 6 as there are 6 layers in the tree. This also tells that the longest sequence have length $k=6$.

Figure 7 shows a processed movement tree in which those edges that do not have count greater than min-support are removed. Minimum support in this case is equal to 3, the same as the example used in Apriori-like Movement Pattern (*AMP*) algorithm. Nodes that are not connected by edges or connected by edges that have count lesser than *min-support* are also removed. This shows the processed movement tree where only the objectively significant nodes and edges are shown. Note that at this point confidence is not yet calculated. This processed tree directly tells that movement from *café* to *bookshop*, or the movement from *bookshop* to *food* or the movement from *food* to *sports* are prevalent in the mobile environment.

Figure 8 shows a portion of the raw movement tree with all the nodes restructured for better presentation. The edges connection and count of each edge remains the same. In this case, the *confidence* of {*bookshop* → *food* → *sports*} is calculated by the occurrence of {*bookshop* → *food* → ***} / {*bookshop* → *food* → *sports*}. It is calculated as 6 / 10=60%. This shows how confidence is calculated in a tree, where the values necessary to calculate the confidence can be derived directly from the variables stored on the edges of the movement tree.

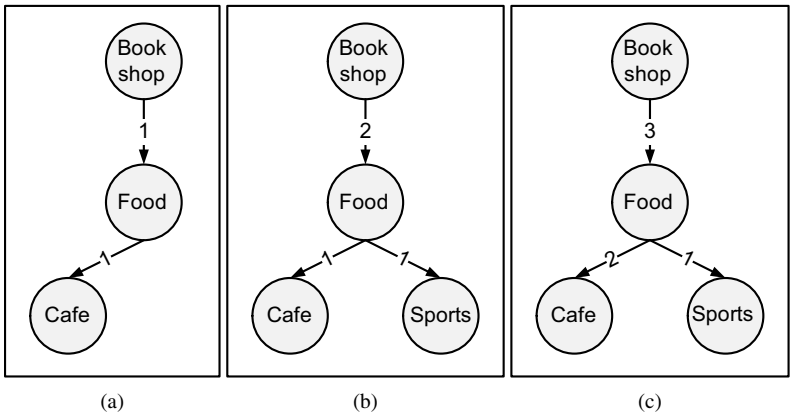

FIGURE 5 Example of movement tree building process.

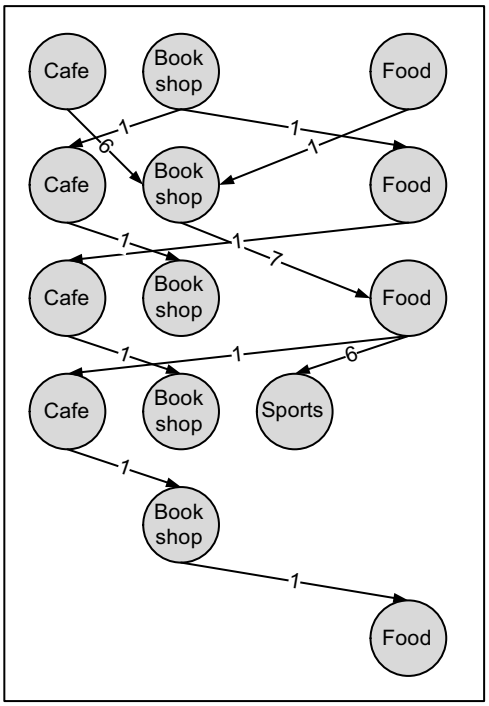

FIGURE 6 Raw Movement Tree.

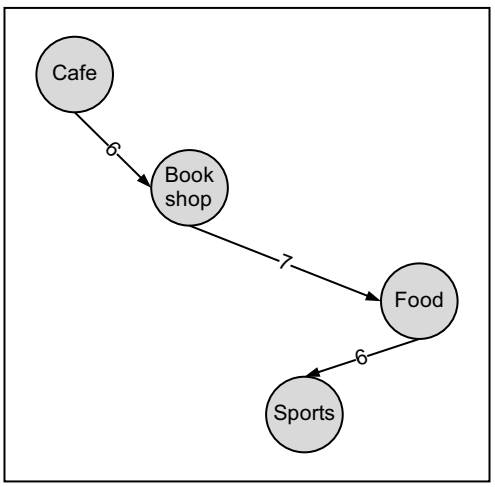

FIGURE 7 Processed Movement Tree.

5. Performance Evaluation

The performance evaluation was performed on three synthetic databases with varying degree of parameters in order to distinguish their difference in performance. Three parameters are used, namely *time horizon, location of interest*, and *k-average*. The performance evaluation was conducted on a Pentium IV platform with 384MB of main memory, using C programming language with data access through a file structure. Source data is synthetic

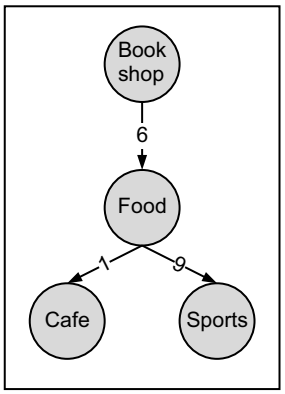

FIGURE 8 Calculating confidence.

data planned using Microsoft Excel and converted to plain text file and for reading in C programming language. Table 11 provides the parameters and description for each variable.

Table 12 shows the parameters for *DB-A, DB-B* and *DB-C*. Time horizon refers to the size of the time series in the user movement database (*UMD*). The longer the time horizon, the longer the data capture of the user movement database, thus the richer data have obtained, thus potentially better knowledge output. The *LOI* refers to the total number of locations of interest available in the database. The more the *LOI*, the greater the complexity of the database and therefore, the greater permutations required for each candidate sequences generation. Variable *k-average* refers to the average size of the sequences contained in the database, and the longer the average sequences, the more complex the mining problem.

Figure 9 shows the performance graph. The performance graph is measured by means of $k=2$ candidate size, count of frequent sequences, count of confident sequences, and finally the number of location movement database (*LMD*) passes. It is found that at increasing level of database complexity, $k=2$ candidate size and number of location movement database (*LMD*) passes increased near exponentially. It is also found that at an

TABLE 11 Performance Parameter

Parameter	Description
time horizon	Total duration which the synthetic database covers. Measured in terms of number of time units in total.
location of interest	Total number of unique location of interest. Measured in terms of natural number greater than 0.
k-average	The average size of individual movement. Measured in terms of number of steps. Calculated by averaging the number of steps for transaction in the source database.

TABLE 12 Input parameters

DB	Time Horizon	*LOI*	*k-average*
DB-A	10	5	3
DB-B	20	10	4
DB-C	30	15	5

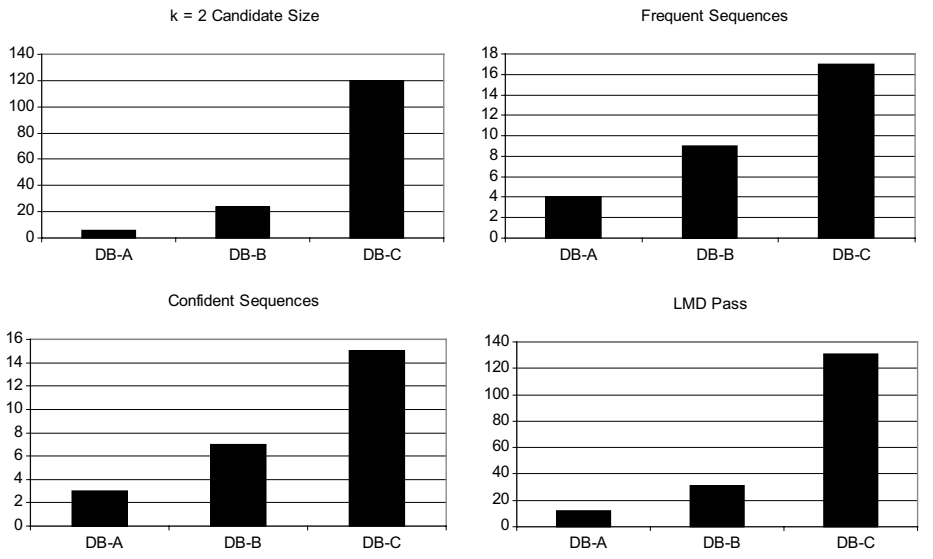

FIGURE 9 Performance Graph.

increasing level of database complexity, the total number of frequent sequences and confident sequences increases at a near linear increase.

6. Conclusion and Future Work

In conclusion, the movement pattern can be mined from the location movement database in order to determine how mobile users travel through the location of interest from base location to another base location via multiple intermediate locations. It is also possible to transform user movement database (*UMD*) which is raw data that records all grid coordinates of mobile users into a format called location movement database (*LMD*) that is suitable for data mining. For algorithm Apriori movement pattern (*AMP*), there is a requirement to go through the location movement database multiple times for counting, whereas for the algorithm movement tree (*M-Tree*) there is no requirement to traverse the location movement database for multiple times.

Future work is to look into predicting the next movement for individual mobile users based on their individual mobile user movement. In this paper, the movement pattern investigated is the overall movement pattern that is sourced from all mobile users from the mobile environment combined, and the output represents the movement pattern in the general population.

About the Authors

John Goh received his Bachelor of Information Technology with distinction, and subsequently Master of Information Technology at Monash University. He is currently finishing his PhD research in knowledge extraction from mobile users. He is interested in developing innovative methods, which have the ultimate result of discovering patterns that support decision makers to better understand the behavior of mobile users. John Goh has since published multiple conference and journal papers in this research area. He is also an assistant editor-in-chief of the Encyclopedia of Mobile Computing and Commerce, which

will be published by the IGI Publishers in 2007. John Goh teaches a Project Management course as an assistant lecturer, both at the undergraduate and the postgraduate level at Monash University.

David Taniar holds Bachelors (Honours), Masters, and PhD degrees—all in Computer Science/Information Technology, with a particular specialty in Databases. His research now expands to Data Mining, Mobile Information Systems, and Web Technology. He publishes extensively every year. He is currently a Senior Lecturer at the Faculty of Information Technology, Monash University, Australia. He is founding editor-in-chief of a number of international journals, including the International Journal of Data Warehousing and Mining, International Journal of Business Intelligence and Data Mining, Mobile Information Systems, Journal of Mobile Multimedia, International Journal of Web Information Systems, and International Journal of Web and Grid Services. He is also an editorial board member of numerous international journals. He was elected as a Fellow of the Institute for Management Information Systems (FIMIS).

References

1. B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins. *Global Positioning System: Theory and Practice*. Springer-Verlag Wien New York, 3rd revised edition, 1994.
2. D. L. Lee, M. Zhu, and H. Hu, "When Location Based Services Meet Databases", *Mobile Information Systems*, **1**, 2, pp. 81–90, 2005.
3. D. R. Forsyth. *Group Dynamics*. Wadsworth, Belmont, CA, 1999.
4. H. C. Tjioe, and D. Taniar. "Mining Association Rules in Data Warehouses", *International Journal of Data Warehousing and Mining*, **1**, 3, pp. 28–62, 2005.
5. J. F. Roddick and B. G. Lees. "Paradigms for Spatial and Spatio-Temporal Data Mining." *Geographic Data Mining and Knowledge Discovery*, Taylor and Francis. Research Monographs in Geographical Information Systems. Miller, H. and Han, J. Eds. pp. 1–14, 2001.
6. J. F. Roddick and M. Spiliopoulou. "A Survey of Temporal Knowledge Discovery Paradigms and Methods." *IEEE Trans. on Knowledge and Data Engineering*, **14**, 4, pp. 750–767, 2002.
7. J. Goh, and D. Taniar. "Mobile user data static object mining (MUDSOM)," *The IEEE 20th International Conference on Advanced Information Networking and Applications, AINA 2006*, IEEE Computer Press, 2006.
8. J. Goh, and D. Taniar. "Static Group Pattern Mining (SGPM)," *The 10th Pacific Asia Conference on Knowledge Discovery and Data Mining PAKDD 2006, Lecture Notes in Computer Science*, Springer-Verlag, 2006.
9. J. Goh, and D. Taniar. "Mining Frequency Pattern from Mobile Users," *Knowledge Based Intelligent Information & Engineering and Systems, Lecture Notes in Computer Science* Part III, Springer-Verlag, **3215**, pp. 795–801, 2004.
10. J. Goh, and D. Taniar. "Mining Parallel Pattern from Mobile Users," *International Journal of Business Data Communications and Networking*, **1**, 1, pp. 50–76, 2005.
11. J. Goh, and D. Taniar. "Mobile User Data Mining by Location Dependencies," *5th International Conference on Intelligent Data Engineering and Automated Learning, Lecture Notes in Computer Science*, Springer-Verlag, **3177**, pp. 225–231, 2004.
12. J. Häkkilä, and J. Mäntyjärvi. "Combining Location-Aware Mobile Phone Applications and Multimedia Messaging," *Journal of Mobile Multimedia*, **1**, 1, pp. 18–32, 2005.
13. J. Han and A. W. Plank. "Background for Association Rules and Cost Estimate of Selected Mining Algorithms," In *Proc. of the 5th CIKM*, pp. 73–80, 1996.
14. J. Han, G. Dong, and Y. Yin. "Efficient Mining of Partial Periodic Patterns in Time Series Database." In *Proc. of 15th ICDE*, pp. 106–115, 1999.
15. J. Han, J. Pei, and Y. Yin. "Mining Frequent Patterns without Candidate Generation." In *Proc. of ACM SIGMOD*, pp. 1–12, 2000.
16. K. Koperski and J. Han. "Discovery of Spatial Association Rules in Geographical Information Databases." In *Proc of 4th Int Symp. on Advances in Spatial Databases*, **951**, pp. 47–66, 1995.

17. L. Forlizzi, R. H. Guting, E. Nardelli, and M. Schneider. "A Data Model and Data Structures for Moving Objects Databases." *ACM SIGMOD Record*, **260**, pp. 319–330, 2000.
18. M-B Song, S-W Kang, and K-J Park. "On the design of energy-efficient location tracking mechanism in location-aware computing," *Mobile Information Systems: An International Journal*, IOS Press, **1**, 2, pp. 109–127, 2005.
19. P. K. C. Tse, W. K. Lam, K. W. Ng, and C. Chan. "An Implementation of Location-Aware Multimedia Information Download to Mobile System," *Journal of Mobile Multimedia*, **1**, 1, pp. 33–46, 2005.
20. P. Zarchan. *Global Positioning System: Theory and Applications*, vol **1**. American Institute of Aeronautics and Astronautics, 1996.
21. R. Agrawal and R. Srikat. "Fast Algorithms for Mining Association Rules." In *Proc. of the 20th VLDB*, pp. 487–499, 1994.
22. R. Agrawal and R. Srikat. "Mining Sequential Patterns." In *Proc. of 11th ICDE*, pp. 3–14, 1995.
23. S. Chakrabarti, S. Sarawagi, and B. Dom. "Mining Surprising Patterns using Temporal Description Length." In *Proc. of 24th VLDB*, pp. 606–617, 1998.
24. S. Y. Chen, and X. Loi. "Data mining from 1994 to 2004: an application-oriented review", *International Journal of Business Intelligence and Data Mining*, **1**, 1, pp. 4–21, 2005.
25. S-Y Hwang, Y-H Loi, J-K Chiu, and E-P Lim. "Mining Mobile Group Patterns: A Trajectory-Based Approach," In *Proc. of the 9th Pacific Asia Conference of Knowledge Discovery and Data Mining PAKDD 2005, Lecture Notes in Computer Science*, Springer-Verlag, **3518**, pp. 713–718, 2005.
26. U. Varshney, R. Vetter, and R. Kalakota. "Mobile Commerce: A New Frontier." *IEEE Computer: Special Issue on E-commerce*, pp. 32–38, October 2000.
27. W. Wang, J. Yang, and P. S. Yu. "InfoMiner+: Mining Partial Periodic Patterns in Time Series Data." 2nd *IEEE International Conference on Data Mining ICDM 2002*, pp. 725, 2002.
28. Y. Wang, E-P Lim, and S-Y Hwang. "Efficient Group Pattern Mining Using Data Summarization". In *Proc. of the 15th International Conference on Database and Expert Systems Applications DEXA 2004, Lecture Notes in Computer Science*, Springer-Verlag, **2973**, pp. 895–907, 2004.
29. Y. Wang, E-P Lim, and S-Y Hwang. "On Mining Group Patterns from Mobile Users". In *Proc. of the 14th International Conference on Database and Expert Systems Applications DEXA 2003, Lecture Notes in Computer Science*, Springer-Verlag, **2736**, pp. 287–296, 2003.
30. M. Cho, J. Pei, H. Wang, W. Wang. "Preference-Based Frequent Pattern Mining", *International Journal of Data Warehousing and Mining*, **1**, 4, pp. 56–77, 2005.
31. S. Bagui, "An Approach to Mining Crime Patterns", *International Journal of Data Warehousing and Mining*, **2**, 1, pp. 50–80, 2006.
32. S. W. Kang, M. Song, K. Park, C. Hwang. "Semantic Prefetching Strategy to Manage Location Dependent Data in Mobile Information Systems," *Mobile Information Systems*, **1**, 3, pp. 149–166, 2005.
33. M. Safar, "K-Nearest Neighbor Search in Navigation Systems," *Mobile Information Systems*, **1**, 3, pp. 207–224, 2005.
34. B. Zhou, S. Cheung, A. Fong. "A Web Usage Lattice Based Mining Approach for Intelligent Web Personalization," *International Journal of Web Information Systems*, **1**, 3, 2005.