Hindawi
Mobile Information Systems
Volume 2017, Article ID 7871502, 9 pages
https://doi.org/10.1155/2017/7871502

Research Article

Recommending Locations Based on Users' Periodic Behaviors

Bing Xu, Zhijun Ding, and Hongzhong Chen

Department of Computer Science and Technology, Tongji University, Shanghai, China

Correspondence should be addressed to Zhijun Ding; dingzj@tongji.edu.cn

Received 23 September 2016; Revised 11 December 2016; Accepted 22 December 2016; Published 21 February 2017

Academic Editor: Qingchen Zhang

Copyright © 2017 Bing Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The research of location recommendation system is an important topic in the field of LBSN (Location-Based Social Network). Recently, more and more researchers began focusing on researching how to recommend locations based on user's life behavior. In this paper, we proposed a new model recommending locations based on user's periodic behaviors. In view of multiple periodic behaviors existing in time series, an algorithm which can mine all periods in time series is proposed in this paper. Based on the periodic behaviors, we recommend locations using item-based collaborative filtering algorithm. In this paper, we will also introduce our recommendation system which can collect users' GPS trajectory, mine user's multiple periods, and recommend locations based user's periodic behavior.

1. Introduction

LBSN (Location-Based Social Network) is a new kind of social network which combines the user's friendship and the user's position. There are two hot research hotspots in the filed of LBSN: recommendation system and mining user's life pattern. For recommending system, the user can get more suggestion for deciding where he can choose to go to. For mining life pattern, the recommendation system can better understand the use's preference. Periodic behavior is a kind of life pattern. In this paper, we proposed a new kind of recommendation model based on the user's periodic behaviors. For solving the problem of mining multiple periods, we proposed a new algorithm which can mine multiple periods in the time sequence.

The research of recommendation systems in the field of LBSN has attracted many researchers' attention and produced a lot of research results. Currently, the recommendation system of LBSN is divided into good friends recommended [1], location recommended [2], activities recommended [3], and event recommended [4]. In the field of friend recommendation, Papadimitriou et al. proposed a friend recommendation algorithm based on user social network [1]. For location recommendation, Bellotti et al. proposed a personalized location recommendation algorithm based on social network and location influence [3]. The activities recommendation is recommending a place for users' preferred activities. Bellotti et al. proposed a system which can recommend the preferred location of activities. The event recommendation is a special case of activities recommending; the main research directions in the field is event detection. Papadimitriou et al. proposed a [2] position information label incident detection algorithm. Furthermore, Gao et al. [5] proposed a location recommendation model which can solve the cold start problem. Zhang et al. [6] proposed a personalized and efficient geographical location recommendation framework called iGeoRec. Huang [7] proposed a novel context similarity measure to quantify the similarity between any two contexts and develop three context-aware collaborative filtering methods for recommending locations. Yuan and Li [8] proposed a location recommendation algorithm based on temporal and geographical similarity in location-based social networks.

With the gradual accumulation of user' location data, the research of users' life behavior has aroused the interest of some researchers and has produced the relevant research results. Rekimoto et al. proposed a user interest point travel sequence mining algorithm based on GPS trajectory and proposed an algorithm which can predict the next destination based on the user's current location [9]. Ye et al. proposed a user behavior pattern mining algorithm based on WIFI's location monitoring technology [10]. Lahiri and Berger-Wolf proposed a LP-Mine algorithm which can mine user's behavior patterns from the original GPS trajectory data [11].

Periodic behavior is a kind of life behavior. The user arriving in a certain area periodically is called a periodic behavior. For example, a user goes shopping every weekend and a user reaches the company for working every Friday to Monday. In order to obtain periods, we should use some period mining algorithm. In this paper, we proposed a new algorithm for mining periods and proposed a new model for recommending locations based on users' periodic behaviors.

Recently, some researchers have proposed some period mining algorithms. Li et al. used the periodogram and self-correlation method for mining the user periodic behaviors [12, 13], and to combine the periodic behavior, the authors proposed a kind of location predicting model based on the thought of probability. Wang proposed the basic thought of periodogram [14]. After several years of development, this algorithm has been successfully applied in different areas for mining periods, for example, hydrological time series analysis [15] and medical time series analysis [16]. The basic idea of the period diagram is based on the power density spectrum estimation which is based on Fourier transform. Because of the leakage of the spectrum, this algorithm is not accurate for mining all periods. In order to avoid this problem [17], windowing technique is proposed by the researcher, but this technology is not accurate enough. Self-correlation method has been proposed. Compared with the periodogram, this algorithm can solve the problem of mining periods from arbitrary length time sequence. In addition, Parthasarathy et al. proposed an algorithm for obtaining [18] time series based on cross entropy; the algorithm can get into one of the most significant periods of time series. By using this algorithm, it is possible to find out the periodic behavior of the users in the purchase of electronic goods. Wang et al. also proposed a time series period mining algorithm [19], which is designed based on the frequent features of periodic behavior and the basic idea of the structure. Xu et al. proposed an [20] algorithm model of hydrologic sequence period based on Mexhat wavelet; this model can be used to dig out periodic behavior from the monthly rainfall in Shanghai.

In real life, some users go to a certain area with multiple periods. For example, a user goes to an area not only every Friday but also every two weeks on Monday. In this paper, we proposed a period acquisition algorithm which can mine multiple periods in time sequence correctly. In our research, we found that there are multiple periods existing in some time sequence generated by users. For example, a teacher not only comes to a campus every Monday but also comes to this campus on Tuesday every two weeks. For this teacher, there are two periodic behavior in his daily work behavior. The classical algorithm, periodogram and self-correlation function, cannot mine all correct periods because multiple periods affect each other while mining periods. In our experience, they cannot mine multiple periods in the time sequence generated by restrict multiple periodic behaviors. The main thought of periodogram is DFT (Discrete Fourier Transform), and an explanation of DFT is that a Fourier coefficient is a relativity between its sine curve and original time sequence. So it cannot avoid the interference of other periodic behaviors. Self-correlation function calculates correlation of time sequence with different time delay and cannot avoid the interference of other events happening in other time stamps. These two algorithms cannot get the start time stamp and all time stamps at which a periodic behavior happened, so we cannot know the time stamp at which the periodic behavior happened. In this paper, we proposed a multiple period acquisition algorithm which can not only mine all periodic behaviors but also get the time stamp at which a periodic behavior happened. So it can help us to predict a user's future behavior based on these periodic behaviors.

After mining the user's periodic behaviors, we have done some researches on how to recommend location based on the user's periodic behaviors. In the field of location recommendation system, some researchers have proposed some models based on the user's preference and friendships. But we found few location recommendation models based on the user's periodic behaviors during our search work. Collaborative filtering is a famous recommendation algorithm. Item CF and User CF are the basic algorithms of collaborative filtering. In this paper, because there are no friendships in our data, we use Item CF for recommending location based on the users' preference and the periodic behaviors. We designed and developed a recommendation system. This system includes an android application for showing recommending locations and collecting GPS trajectory and the server for mining periodic behaviors and recommending locations for the users based on the users' periodic behaviors.

The rest of the paper is organized as follows. Section 2 introduces the multiple periods mining algorithm which is proposed in this paper. Section 3 introduces the location recommendation model based on the periodic behaviors proposed in this paper. Section 4 introduces the thought and the architecture of our location recommendation system and we report our experimental results in Section 4. Section 5 introduces the conclusion and our future works.

2. Mining Multiple Periods in Time Sequence

In users' daily life, the user comes to a certain area not only with single period but also with multiple periods. The search of periodic behavior has attracted many researchers' attention. Recently, some algorithms proposed for mining periods are not accurate for mining multiple periods. In this section, we will introduce the proposed algorithm for mining multiple periods. For solving the problem of mining multiple periods, we proposed a new algorithm based on the thought of matrix, the thought of item hitting, and the thought of changing the matrix dynamically for reducing the execution time. Using this algorithm, we can mine all periods from time sequence.

```
Input: S (time sequence)
st (support threshold)
ts (target state)
Out put: period (all periods in the time sequence)
begin
(1) matrix,map, T = GeneratingMatrix (S, ts) // creating the suspected periods matrix based on the
time-stamps the target state happened
(2) periods = MiningPeriods (matrix, st,map,T)
(3) return periods
end
```

Algorithm 1: Mining multiple periods.

This algorithm [17] uses the matrix for storing all suspected periods in the time sequence. In order to access elements in the matrix according to the time stamp, this algorithm constructs a hash mapping relationship between the time stamps and the number of the rows and columns of the matrix and all the items in the matrix are suspected periods. For mining true periods from the matrix, this algorithm judges all the suspected periods by counting the number of successful hitting times. If the count of hitting is more than the support threshold, the value of the current suspected period is a true period of this time sequence and if not this algorithm will judge the next suspected period by the same way. For reducing the execution time of mining all periods, this algorithm will change the value of hitting items in the matrix to be zero dynamically. This algorithm can be shown as in Algorithm 1.

In Algorithm 1, GeneratingMatrix(S, ts) is the progress of creating all suspected periods' matrix based on the time stamps at which the target state happened which will be introduced in Section 2.1. MiningPeriods(matrix, st,map,T) is the progress of mining all true periods stored in the matrix which will be introduced in Section 2.2.

2.1. Creating the Suspected Periods Stored Matrix.

In the time sequence, the period of a target sate is greater than 1 and less than len(S)/2. len(S) is the length of the time sequence. If we judge whether the value from 1 to len(S)/2 is a true period, the algorithm may require a higher time complexity. Based on the characteristic that the periodic behavior always occurs at the same time interval, the period of the time series can only be the time difference between any two time stamps at which the target state happened. The algorithm of generating all suspected periods can be shown in formula (1). t_i and t_j are the time stamp at which the target state happened. sp is the suspected period. T is the time stamps at which the target state happened and L is the set of all suspected periods. This progress is shown in the following:

$$L = \left\{ \mathrm{sp} \mid \mathrm{sp} = t_i - t_j \left(t_i \in T \text{ and } t_j \in T \text{ and } t_i \geq t_j \right) \right\}. \quad (1)$$

Example 1. The time sequence is "00011001010100011." The target state is "1." The time stamps at which the target state happened are $\{3, 4, 7, 9, 11, 14, 15\}$. The all suspected periods can be gotten according to formula (1) as shown as $\{1, 4, 6, 8, 11, 12, 3, 5, 7, \ldots\}$. And the true periods $\{4, 5\}$ are stored in this suspected periods stored set.

In the above analysis, the time interval between any time stamps can be calculated as the suspected periods of this algorithm. While storing all suspected periods, if we choose a set or a list for storing the suspected time, the complexity of accessing the target item is $o(n)$. Therefore, this algorithm uses hash mapping algorithm and matrix to store the suspected periods and the number of the rows and the columns of the matrix mapping to the time stamps at which the target state happened. So this algorithm can access the item in the matrix according to the time stamps. And the complexity of accessing the item is $o(1)$. Another advantage of this method is that it can execute the hitting method according to the time stamps which simplify the progress of judging the suspected periods.

The progress of establishing suspected periods stored matrix can be shown in Algorithm 2. From lines (5) to (13), the algorithm acquires all time stamps at which the target state happened and creates the mapping relation to the number of rows and columns of the matrix. The hash mapping algorithm is shown in formula (2). In this formula, h is a hash value. t is the time stamp at which the target state happened. S is the time sequence. len(S) is the length of the time sequence.

$$h = t\% \mathrm{len}(S). \quad (2)$$

From lines (15) to (22), the algorithm calculates the suspected periods according to the time stamps at which the target state happened and changes the value of the period which is bigger than len(S)/2 or less than zero to be zero for reducing the count of judging steps. In Example 2, we introduce an example for explaining the progress of Algorithm 2.

Example 2. According to the time sequence shown in Example 1, the suspected periods stored matrix can be created as

```
Input: S(time sequence)
       ts (target state)
Output: matrix (the suspected periods stored matrix)
       map (hash mapping relation)
       T (the time stamps the target state happened)
begin:
(1) map = new HashMap (key, value) // hash mapping relation between time stamps and the
number of the row and column in the matrix
(2) T = [] // the time stamps the target state happened
(3) r = 0 // the number of the row in the matrix
(4) i = 0
(5) While i < len(S) do
(6)        if S[i] == ts do
(7)            map.put (i, r)
(8)            i + +
(9)            r + +
(10)           T.append(i)
(11)       end
(12)       i + +
(13) end
(14) matrix = int[r + 1][r + 1]
(15) for t1 in T do
(16)     for t2 in T do
(17)         sp = t2–t1//suspected periods
(18)         if sp > 0 and sp < len(S)/2 do
(19)         matrix[map.get(t1)][map.get(t2)] = sp
(20)         end
(21)     end
(22) end
(23) return matrix, map, T
end
```

Algorithm 2: Generating matrix.

shown in the following based on the algorithm shown in Algorithm 2:

$$\begin{array}{cccccccc} & 3 & 4 & 7 & 9 & 11 & 14 & 15 \\ 3 & 0 & 1 & 4 & 6 & 0 & 0 & 0 \\ 4 & 0 & 0 & 3 & 5 & 7 & 0 & 0 \\ 7 & 0 & 0 & 0 & 2 & 4 & 7 & 0 \\ 9 & 0 & 0 & 0 & 0 & 2 & 5 & 6 \\ 11 & 0 & 0 & 0 & 0 & 0 & 3 & 4 \\ 14 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}. \tag{3}$$

2.2. Acquiring True Periods and Changing the Matrix Dynamically. In this step, the multiple periods mining algorithm will mine all true periods stored in the matrix step by step. The algorithm proposed in this paper will count the times the suspected period happened actually by using item hitting method. Then, the algorithm calculates the period of the target period. If the support is more than the threshold, the target period is a true period and the matrix is changed dynamically for reducing execution time. If not, the algorithm will judge the next suspected period.

During mining all true periods, the values stored in the matrix are called suspected periods. The row number of the suspected period is the first time the target state happened in this period and the column number is the second time. After mining a true period, the algorithm then changes the hitting item in the matrix to be zero. Then, the next steps will not judge the same periodic behavior. If the current item is not a true period, the algorithm will judge the next item. After judging all suspected periods stored in the matrix, the algorithm will stop and generate all mined true periods of this time sequence. The main thought of mining periods from the matrix can be shown in Algorithm 3.

In Algorithm 3, the input of the algorithm includes the storage matrix of the suspected period, the occurrence time stamp collection of the target state, the support threshold value, the hash mapping between time stamps, and the number of rows and columns of the matrix. In line (1), the algorithm initializes the period object set. Each period object contains two fields, that is, the period and the first time the periodic behavior happened. From lines (2) to (21), the algorithm mines all true periods from the matrix. Lines (9) to (15) are the progress of item hitting method. The hitting count is the times the target suspected periodic behavior happened. From lines (13) to (14), the algorithm generates the next hitting item's row and column number. If the column number

```
Input: matrix
       st
       map
       T
       S
Output: periods
Begin
(1) periods = []
(2) for t1 in T do
(3)   for t2 in T do
(4)       p = matrix[t1][t2]
(5)       If p! = 0 and p ≤ len(S)/2 do
(6)           r = t1
(7)           c = t2
(8)           hitting_count = 0
(9)           while c ≤ len(S) do
(10)                  If r in T and c in T do
(11)              hitting_count++
(12)            end
(13)            r = c
(14)            c = c + p
(15)          end
(16)      If support_sp(hitting_count, t1, p, len(S)) ≥ st do
(17)          periods.append(new PeriodObject(p, t1))
(18)          DynamicChangeMatrix(matrix, t1, t2,T)
(19)      end
(20)   end
(21) end
end
```

Algorithm 3: Mining periods.

is more than the length of time sequence, the hitting progress of the suspected periods will stop.

After executing the hitting method, the algorithm will execute the code from lines (16) to (19). Line (16) is calculating the support of the suspected period. The method of calculating support is shown in formula (4). In formula (4), SP is the support, h is the hitting count, l is the length of the time sequence, t is the time stamp at which the period happened firstly, and p is the value of the period. If the support is more than the support threshold, the algorithm will execute lines (17) to (18). In line (17), the algorithm will change the hitting items' value to be zero. Line (18) will store the mined period in the set. The method of change the matrix dynamically is shown in Algorithm 4.

$$SP = \frac{h}{(l-t)/p}. \tag{4}$$

In Algorithm 4, the method will execute the item hitting method one more time. But the difference is that in this time the algorithm will change the hitting item's value to be zero for avoiding judging the same periodic behavior more times. The full progress is shown in Algorithm 4.

In Algorithm 4, we describe the progress of the proposed multiple periods mining algorithm in detail. Example 3 will introduce the full progress according to a detailed example. In this example, we will give the situation of matrix after judging two items.

Example 3. In this example, we will continue to use the time sequence of Example 1 and the suspected periodic matrix of Example 2, while the support threshold value is 80%. (5) is the algorithm after determining whether the M[3][7] and M[3][4] are true periods. Among them, because M[3][7] is a true period, the value of the hitting element in the suspected periodic matrix is changed to 0.

The state of the matrix after judging M[3][4] and M[3][7] is as follows:

$$\begin{array}{cccccccc} & 3 & 4 & 7 & 9 & 11 & 14 & 15 \\ 3 & 0 & 1 & 0 & 6 & 0 & 0 & 0 \\ 4 & 0 & 0 & 3 & 5 & 7 & 0 & 0 \\ 7 & 0 & 0 & 0 & 2 & 0 & 7 & 0 \\ 9 & 0 & 0 & 0 & 0 & 2 & 5 & 6 \\ 11 & 0 & 0 & 0 & 0 & 0 & 3 & 0 \\ 14 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}. \tag{5}$$

The process of this example is shown as follows.

(1) For the value of M[3][4] is 1, the algorithm executes the hitting method based on the period of 1 and this periodic behavior starts from the third locations of time series. We can get that the support value of M[3][4] is 7/13. And the support

```
Input: matrix
t1
t2
p
T
Output: None
Begin
While $t2 \leq \max(T)$ do
        If $t1$ in $T$ and $t2$ in $T$ do
                $matrix[t1][t2] = 0$
                $t1 = t2$
                $t2 = t2 + p$
        end
end
End
```

Algorithm 4: Dynamic change matrix.

value is less than 80%. So M[3][4] is not a true period. Then, the algorithm will judge the item M[3][7].

(2) The algorithm will execute hitting method based on the period of 4 and this periodic behavior starts from the third locations of time series. We can get that the support value is 1 which is more than the support threshold. So M[3][7] is a true period. The algorithm then changes the matrix dynamically. And the state of the matrix is shown in (5). From (5), we can find that all values of the hitting items are changed to be zero. So the algorithm will not judge the same periodic behaviors. The hitting items are M[3][7], M[7][11], and M[11][15]. After this judging, the algorithm will judge the next item by the same method.

3. Location Recommendation Based on Users' Periodic Behaviors

In Section 2, we proposed a new algorithm which can mine all periods in time sequence. By using this algorithm, we can mine the user's period of arriving at a certain area. In this section, we will introduce how to recommend shops based on the user's periodic behaviors.

In the field of recommendation system, Item Collaboration Filter is a hot recommendation model. During the past several years, many companies recommended items to the users by using this algorithm. The thought of this model is that the users may buy the similar items. In this paper, we think that the users may go to the shops which are near the areas the users arrive at periodically. After mining users' periodic behaviors, we can recommend locations to the target user according the period and the latitude and longitude of the periodic arrival area. We choose the Item CF algorithm for calculating the user's preference of a certain location. For recommending locations to the user personally, in this paper, we choose the user's taste and the distance between the shop and the center of the periodic arrival area as this model's characteristics.

According to the thought of item-based collaborative filtering algorithm, in this paper, we use cosine-based similarity algorithm to calculate the similarity between two items. The formula is shown in formula (6). i and j are two shops' feature vectors. In this paper the feature vector is (*similarity_of_user's_taste, the_distace*). The first element of this vector is the similarity between the shop's tag and the user's taste. The second element of this vector is the distance of this shop to the center of this user's periodic arrival area:

$$\operatorname{sim}(i, j) = \cos(\vec{i}, \vec{j}) = \frac{\vec{i}.\vec{j}}{\|\vec{i}\|_2 * \|\vec{j}\|_2}. \tag{6}$$

After calculating the similarity of the shops, in this paper, we calculate the user's preference of a shop by using formula (7). In this formula, $S_{i,N}$ is the similarity between i and N. And $R_{u,N}$ is the user's preference of the shop N that this user has been there.

$$P_{u,i} = \frac{\sum_{\text{all\_similar\_items},N} (s_{i,N} * R_{u,N})}{\sum_{\text{all\_similar\_items},N} (|s_{i,N}|)}. \tag{7}$$

After calculating the user's preference of these shops, we can sort these shops according to preference. And then we can give this user some suggestions for deciding which shop this user can go to. In real life, we may go to the same shop for shopping or having a dinner. In our project, we do not filter out the shops this user has been to. In Section 4, we will introduce the system we developed based on this recommendation model.

4. Experiments and System

In this section, we will use the algorithm proposed in this chapter to conduct multiple periodic behavior mining experiment. We use the check-in data collected in a company. In this section, we give the comparative experiment analysis of the multiple periods mining algorithm proposed in this paper. Then, we will introduce the shops recommendation systems based on the user's periodic behaviors.

4.1. Comparative Experiment Analysis of the Multiple Periods Mining Algorithm. In this section, we mined periodic behaviors by using periodogram and the algorithm proposed in this paper. The time sequence is shown in (8). The result of periodogram algorithm is shown in Figure 1. In this experiment, we set the support threshold to be 80%. From the original time sequence, we can find that this user has two periodic behaviors directly. The employees come to this company every week on Friday and every two weeks on Tuesday.

The time sequence is as follows:

$$\begin{gathered} 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, \\ 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0. \end{gathered} \tag{8}$$

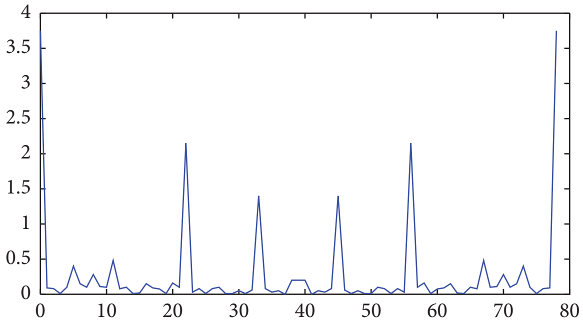


Figure 1: The power spectrum of periodogram.

Table 1: The result of the proposed algorithm.

Periods	Time stamps
7	2,16,30,44,72,
14	5,12,26,33,40,47,54,61,68,75,

Firstly, we mine multiple periodic behaviors by using the proposed algorithm in this paper. The result can be seen in Table 1. We can find that the result of this experiment is correct.

Then, we mine the periodic behaviors by using the algorithm of periodogram. The result of density of power spectrum is shown in Figure 1. The x-axis is the frequency and the y-axis is the density of power spectrum. We take the frequency of the power spectrum in the top as the true frequency. According to formula (9), we can calculate the period. Because the period is an integer, we get the period of 7. In formula (9), len(s) means the length of the time sequence.

$$\text{period} = \frac{\text{len}\,(S)}{\text{frequency}}. \tag{9}$$

In this experiment, we found that the periodogram cannot mine all true periods because of spectral leakage. The period of 14 cannot be divided by 77 (the length of this time sequence). So when we mine periodic behavior by periodogram, we cannot get the period of 14 directly. And the periodogram cannot tell us when the behavior happens in a period. Based on this period, we cannot predict the user's behavior in future. The proposed algorithm in this paper solved this problem.

4.2. Location Recommendation Experiment. In our location recommendation experiment, we use the GPS trajectory collected in our system as our experiment's data. We mine the user's periodic behavior by using the method proposed in Section 2. Based on the periodic behaviors, we generate the recommending location list by using the method proposed in Section 3. In this section, we will use the accuracy to evaluate the recommendation method proposed in this paper. The accuracy of our recommendation method is shown in Figure 2. The x-axis of the figure is the count of recommending locations. And the y-axis of this figure is the accuracy. The red line is the result of the algorithm proposed in this paper. The black line is the result of item-based CF algorithm and the blue line is the result of the user-based CF algorithm. In Figure 2, we evaluate the item-based CF algorithm without using the characteristic of periodic behaviors. We can see that the recommendation method proposed in this paper is higher than the raw item-based CF algorithm. We also recommend the location to the user by using user-based CF algorithm and it is less accurate than our model. From this result, we can know that mining users' behavior is important for recommending the users locations, because the user may always go a location which is near his frequent or periodic arrival area. For example, we may go shopping near our home, go to a restaurant for a dinner near our working place, and so on. We give an example of the result of location recommendation as follows.

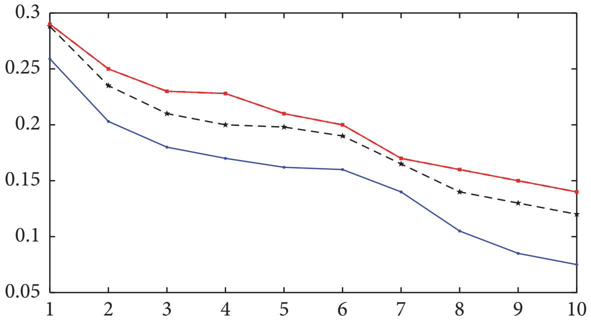


Figure 2: The result of recommendation experiment.

The list of recommending shops is as follows:

Locations List

(1) KFC

(2) Green Tea

(3) Northeast Restaurant

(4) Boiled dumplings

⋮

4.3. The Location Recommendation System Based on the User's Periodic Behaviors. In our system, we developed the mobile application and the server-side application. In the mobile side, the application includes GPS trajectory collection module and the location recommendation module. In the server side, the application includes the GPS trajectory storage module, the stay-points mining module, the frequent arrival area mining module, and the periodic behavior mining module. For recommending location to the users, we also developed the location recommendation module based on the user's periodic behaviors in the server side. In Figure 3, the architecture of our system can be seen. We divided the system into three layers. They are the mobile application layer, the service providing layer, and the data storage layer. The mobile application layer can access the service providing layer through HTTP and the service providing layer can access the data storage layer through JDBC technology. In this section, we will introduce the data processing flow of this system in Section 4.3.1.

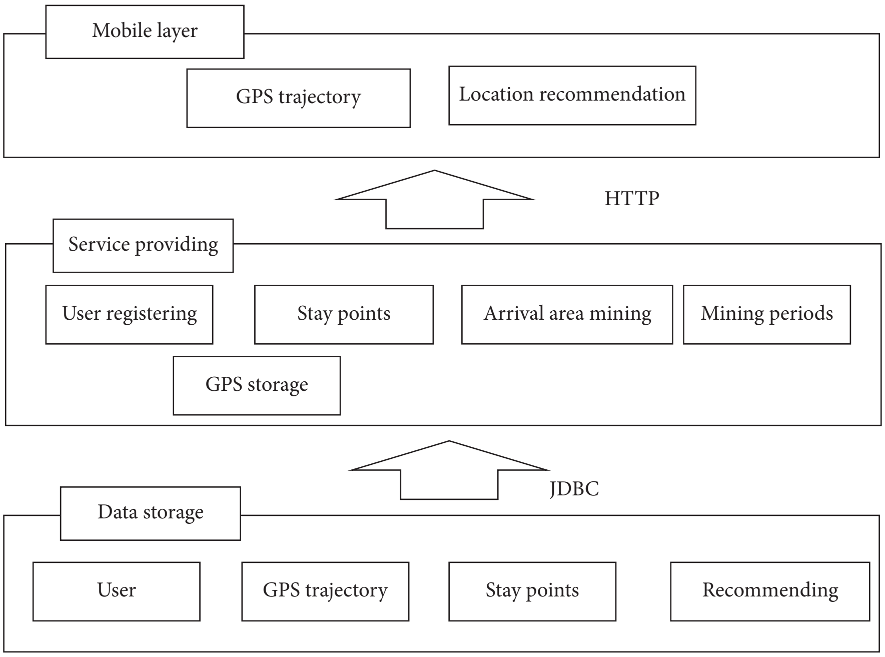


Figure 3: The architecture of our system.

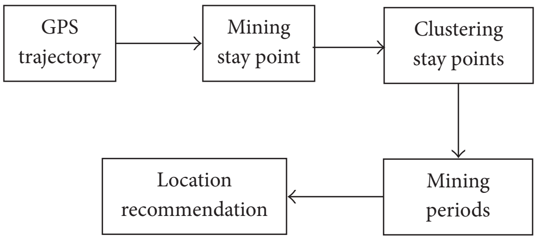


Figure 4: The data processing flow of our system.

4.3.1. The Data Processing Flow of This System. In our system, we design the data processing flow as shown in Figure 4. The server-side application receives the GPS data from the mobile client and stores the data in database. Then, it mines the stay points from the raw GPS data of each user, because a stay point means a staying behavior of a user and the GPS positions of all stay points in a frequent arrival area are not the same. The server then mines semantic area by using OPTICS [21] clustering algorithm. The OPTICS algorithm is a kind of the clustering algorithms based on the density of data. Because it does not need to set the number of classes and we cannot understand the number of the user's frequent arrival area, we choose it for mining the user's frequent arrival area. Such as the K-Means clustering algorithm, we should give the number of classes firstly and then the algorithm can mine all classes effectively; it is not suitable for our application. The OPTICS clustering algorithm is used in many areas widely such as web clustering. Based on the frequent arrival areas mined by the OPTICS algorithm, the server-side application mines the periods of each frequent arrival area by using the periods mining algorithm proposed in this paper.

After mining the user's periodic arrival areas and the periods, we designed the location recommendation module based on the user's periodic behaviors. For each periodic arrival area, we generated the recommending location list by using item-based CF algorithm. In the server side, the system can predict the user's future arrival area based on the periodic behavior. After predicting the future arrival area, this system pushes the recommending location list to the mobile application. The user can consider these locations as a future choice. As introduced above, the architecture of the data processing flow can be seen in Figure 4.

5. Conclusion and Future Works

In this paper, we proposed a new periods mining algorithm which can mine all periods in the time sequence and proposed a periodic behaviors based recommendation method for recommending the locations to the users. From this paper, we can see that our periods mining algorithm is more accurate than some other algorithms and the recommending model is more accurate than the raw item-based CF algorithm. But in this paper we do not consider the effect of the user's friendship in our recommending methods. And we found that the group's periodic behavior is also a true phenomenon in our daily life. So in the future, we will go on doing some research in a location recommending methods based on the friendship and the user's periodic behaviors. And we will do some research on how to mine the group's periodic behaviors.

Disclosure

This paper is extended from the paper named "Mining Multiple Periods in Event Time Sequence" in APSCC 2015.

Competing Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is partially supported by the National Natural Science Funds of China under Grants no. 61173042 and no. 61472004, Hong Kong, Macao, and Taiwan Science and Technology Cooperation Program of China under Grant no. 2013DFM10100, and Special Fund Project of Shanghai Economic and Information Committee under Grant no. CXY-2013-40.

References

[1] M. Ye, P. Yin, W.-C. Lee, and D.-L. Lee, "Exploiting geographical influence for collaborative point-of-interest recommendation," in *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '11)*, pp. 325–334, ACM, Beijing, China, July 2011.

[2] A. Papadimitriou, P. Symeonidis, and Y. Manolopoulos, "Friendlink: link prediction in social networks via bounded local path traversal," in *Proceedings of the International Conference on Computational Aspects of Social Networks (CASoN '11)*, pp. 66–71, Salamanca, Spain, October 2011.

[3] V. Bellotti, B. Begole, E. H. Chi et al., "Activity-based serendipitous recommendations with the magitti mobile leisure guide," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*, pp. 1157–1166, ACM, Florence, Italy, April 2008.

[4] R. Lee, S. Wakamiya, and K. Sumiya, "Discovery of unusual regional social activities using geo-tagged microblogs," *World Wide Web*, vol. 14, no. 4, pp. 321–349, 2011.

[5] H. Gao, J. Tang, and H. Liu, "Addressing the cold-start problem in location recommendation using geo-social correlations," *Data Mining and Knowledge Discovery*, vol. 29, no. 2, pp. 299–323, 2015.

[6] J.-D. Zhang, C.-Y. Chow, and Y. Li, "IGeoRec: a personalized and efficient geographical location recommendation framework," *IEEE Transactions on Services Computing*, vol. 8, no. 5, pp. 701–714, 2015.

[7] H. Huang, "Context-aware location recommendation using geotagged photos in social media," *ISPRS International Journal of Geo-Information*, vol. 5, no. 12, p. 195, 2016.

[8] Z. Yuan and H. Li, "Location recommendation algorithm based on temporal and geographical similarity in location-based social networks," in *Proceedings of the 12th World Congress on Intelligent Control and Automation (WCICA '12)*, pp. 1697–1702, Guilin, China, June 2016.

[9] J. Rekimoto, T. Miyaki, and T. Ishizawa, "LifeTag: WiFi-based continuous location logging for life pattern analysis," in *Location- and Context-Awareness: Third International Symposium, LoCA 2007, Oberpfaffenhofen, Germany, September 20-21, 2007. Proceedings*, Lecture Notes in Computer Science, pp. 35–49, Springer, Berlin, Germany, 2007.

[10] Y. Ye, Y. Zheng, Y. Chen, J. Feng, and X. Xie, "Mining individual life pattern based on location history," in *Proceedings of the 10th International Conference on Mobile Data Management: Systems, Services and Middleware (MDM '09)*, pp. 1–10, May 2009.

[11] M. Lahiri and T. Y. Berger-Wolf, "Mining periodic behavior in dynamic social networks," in *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM '08)*, pp. 373–382, December 2008.

[12] Z. Li, B. Ding, J. Han, R. Kays, and P. Nye, "Mining periodic behaviors for moving objects," in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '10)*, pp. 1099–1108, ACM, Washington, DC, USA, July 2010.

[13] M. S. Bartlett, "Periodogram analysis and continuous spectra," *Biometrika*, vol. 37, pp. 1–16, 1950.

[14] Z. Wang, "Analysis of the hydrological cycle in the middle and upper reaches of the Yellow River," *Northwest Hydropower*, no. 2, pp. 1–5, 1998.

[15] X. Tao, "Preliminary study of periodic regularity of scarlet fever by periodogram method," *Journal of Preventive Medicine*, vol. 14, no. 3, pp. 146–148, 1998.

[16] X. Wei, "Window function analysis of power spectrum estimation by periodic graph method," *Modern Electronic Technology*, vol. 28, no. 3, pp. 14–15, 2005.

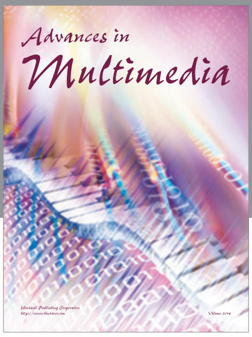
[17] Y. Hua, R. Yang, and Y. Qian, "On detecting customer behavior periodicity with cross entropy," in *Proceedings of the ICIS SIGBPS Workshop on Business Processes and Services (BPS '12)*, pp. 3–7, 2012.

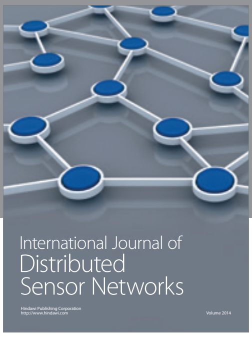
[18] S. Parthasarathy, S. Mehta, and S. Srinivasan, "Robust periodicity detection algorithms," in *Proceedings of the 15th ACM Conference on Information and Knowledge Management (CIKM '06)*, pp. 874–875, Arlington, Va, USA, November 2006.

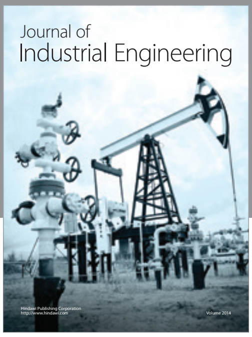
[19] H. Wang, D. Liu, C. Wang et al., "Wavelet periodic analysis model of hydrological series based on seasonal adjustment and trend decomposition and its application," *Journal of Applied Science and Engineering*, vol. 21, no. 5, pp. 823–836, 2013.

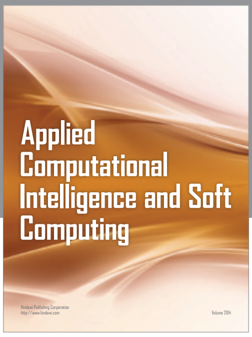
[20] B. Xu, Z. Ding, and H. Chen, "Mining multiple periods in event time sequence," in *Advances in Services Computing*, vol. 9464 of *Lecture Notes in Computer Science*, pp. 278–288, Springer, 2015.

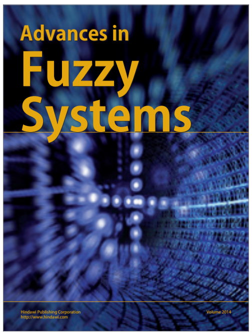
[21] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "OPTICS: ordering points to identify the clustering structure," in *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD '99)*, pp. 49–60, ACM Press, Philadelphia, Pa, USA, June 1999.











Hindawi

Submit your manuscripts at
https://www.hindawi.com