PLOS ONE

RESEARCH ARTICLE

A geographical location prediction method based on continuous time series Markov model

Yongping Du, Chencheng Wang*, Yanlei Qiao, Dongyue Zhao, Wenyang Guo

Faculty of Information Technology, Beijing University of Technology, Beijing, China

* 18511515058@126.com

Citation: Du Y, Wang C, Qiao Y, Zhao D, Guo W (2018) A geographical location prediction method based on continuous time series Markov model. PLoS ONE 13(11): e0207063. https://doi.org/10.1371/journal.pone.0207063

Editor: Ivan Olier, Liverpool John Moores University, UNITED KINGDOM

Received: October 24, 2017

Accepted: October 25, 2018

Published: November 19, 2018

Data Availability Statement: Data are available from https://www.microsoft.com/en-us/download/details.aspx?id=52367.

Funding: This study was supported by National Science and Technology Support Plan under grant NO. 2013BAH21B02-01, http://www.most.gov.cn/; and Beijing Natural Science Foundation under grant NO.4153058, http://www.bjnsf.org/.

Competing interests: The authors have declared that no competing interests exist.

Abstract

Trajectory data uploaded by mobile devices is growing quickly. It represents the movement of an individual or a device based on the longitude and latitude coordinates collected by GPS. The location based service has a broad application prospect in the real world. As the traditional location prediction models which are based on the discrete state sequence cannot predict the locations in real time, we propose a Continuous Time Series Markov Model (CTS-MM) to solve this problem. The method takes the Gaussian Mixed Model (GMM) to simulate the posterior probability of a location in the continuous time series. The probability calculation method and state transition model of the Hidden Markov Model (HMM) are improved to get the precise location prediction. The experimental results on GeoLife data show that CTS-MM performs better for location prediction in exact minute than traditional location prediction models.

Introduction

Location-based Service (LBS) is a kind of information service that provides users geographical positions located by mobile devices and wireless network. There exists wealth information in the location data, such as user's interests, user's hobbies and user's behavior pattern. LBS may be employed in a number of applications, including: location-based advertising [1], personalized weather services, entertainment [2], personal life and so on. An effective location prediction or recommendation can make the users have good experience.

The advances in location-acquisition and mobile communication technologies empower people to use location data with existing online social networks in a variety of ways. People can share their present location, record travel routes with GPS to share travel experiences in GeoLife [3]. Zheng [4] gives the overview of trajectory data mining, including the trajectory data preprocessing, pattern mining and classification, it explores the connections, correlations and differences among these existing techniques and also some public trajectory datasets are presented. Zheng [5] puts forward the approach to find the top-k candidate trips within the uncertain trajectory data. The historical data is used to inference the travel trip and it reduces the uncertainty of the user's trajectory.

In order to improve the location service experience, it is needed to know the user's location in advance. For example, if it can be predicted that the user will appear location B at 6:00pm based on the previous locations visited, the LBS provider can send the recommendation information or advertisements for restaurants in location B to the user in advance. Xue [6] puts forward the SubSyn algorithm for location prediction. The user's historical trajectory data is decomposed into the set of sub trajectory, which increases the number of tracks and the training data size, and the prediction performance is improved. As for the track prediction, the method of Markov Model [7,8] is widely used and its central idea is to build the Markov chain for speculation. The algorithm of SMLP (Social-aware Mobile user Location Prediction algorithm) is put forward by Yu [9]. It integrates the user's correlation with Markov Model for location prediction. Although the algorithm requires less space than the Markov model, the prediction results are heavily affected by the region division. Lian [10] proposes the CEPR(Collaborative Exploration and Periodically Returning model) algorithm which adopts collaborative filtering technique and the user historical behavior is used for location prediction and recommendation. They also give the correlation analysis [11] between the user statistical information and location predictability on the data of Gowalla(https://snap.stanford.edu/data/loc-gowalla.html). The prefix tree and heuristic search strategy are adopted to implement the personalized trip recommendation by Zhang [12]. Wu [13] uses Markov Random Field to predict the annotation of location records and the user's destiny, the better performance is achieved when there exists more user's records. Nghia [14] uses matrix factorization to select features and predicts the user's location. Although the algorithm can predict geographic location on real time, their dataset is composed of tweets containing lots of semantic information. The results are influenced by people's subjective emotions and expressions.

Gambs [15] extends a mobility model called Mobility Markov Chain (MMC) to incorporate the n previous visited locations for next location prediction. However, it cannot predict location within any time interval. Mathew [16]presents a hybrid method for predicting human mobility on the basis of Hidden Markov Models (HMMs). They use forward algorithm to compute the probability of possible sequences and return the next place from the sequence with the highest probability. But the experimental results on GeoLife is not satisfied with the highest Precision@5 of 26.40. Qiao [17] proposes a hybrid Markov based prediction model that contains three stages: mobility pattern discovery, variable-order Markov predictor and mobility pattern based users similarity calculation. The human trajectory data is extracted from data traffic of an LTE(Long Term Evolution) network. The extensive experimental evaluations should be conducted to compare with other related work on different datasets. Huang [18] proposes a predictive model taking into account activity changes. It is implemented for two users selected from the GeoLife dataset and gets performance improvement. The study results are limited by the spatial and temporal coverage of the dataset used and it should be applied to predict human movement by different days of the week with better quality data.

In addition, many other approaches are also used to build the prediction model, such as the association rules based method [19] and so on. However, all of these existing strategies cannot give the prediction based on the real time.

GeoLife(https://www.microsoft.com/en-us/download/details.aspx?id=52367) is the commonly used data for location based service, which records a broad range of users' outdoor movements, including not only life routines but also some entertainments and sports activities. This trajectory dataset can be used in many research fields, such as mobility pattern mining, user activity recognition, location-based social networks and location recommendation.

In this paper, we address the issue of predicting the user's location on the continuous time series based on the historical trajectory data and give the improvement to the original Markov model. The discrete time sequence is simulated to the continuous sequence by Gaussian Mixture Model.

Potential location discovery method

Location prediction structure

The potential location is discovered by filtering and clustering technique on large scale tracing point data. The structure of location prediction based on the real time series is shown in Fig 1.

The user's trajectory data, which is represented by <Time, Location> series, is filtered and clustered to produce a series of candidate location. Gaussian mixture modeling is implemented on the serial density of each location. Combined with the transition probability matrix, the original markov model is improved to the new model which is based on the continuous time series. The symbols used in this paper are shown in Table 1.

Tracing data filtering

The noise data should be filtered to reduce the interference. The filtering algorithm retains the tracing points which maybe selected as the candidate location and discards the irrelevant tracing points. Three kinds of filtering strategies are shown in the following.

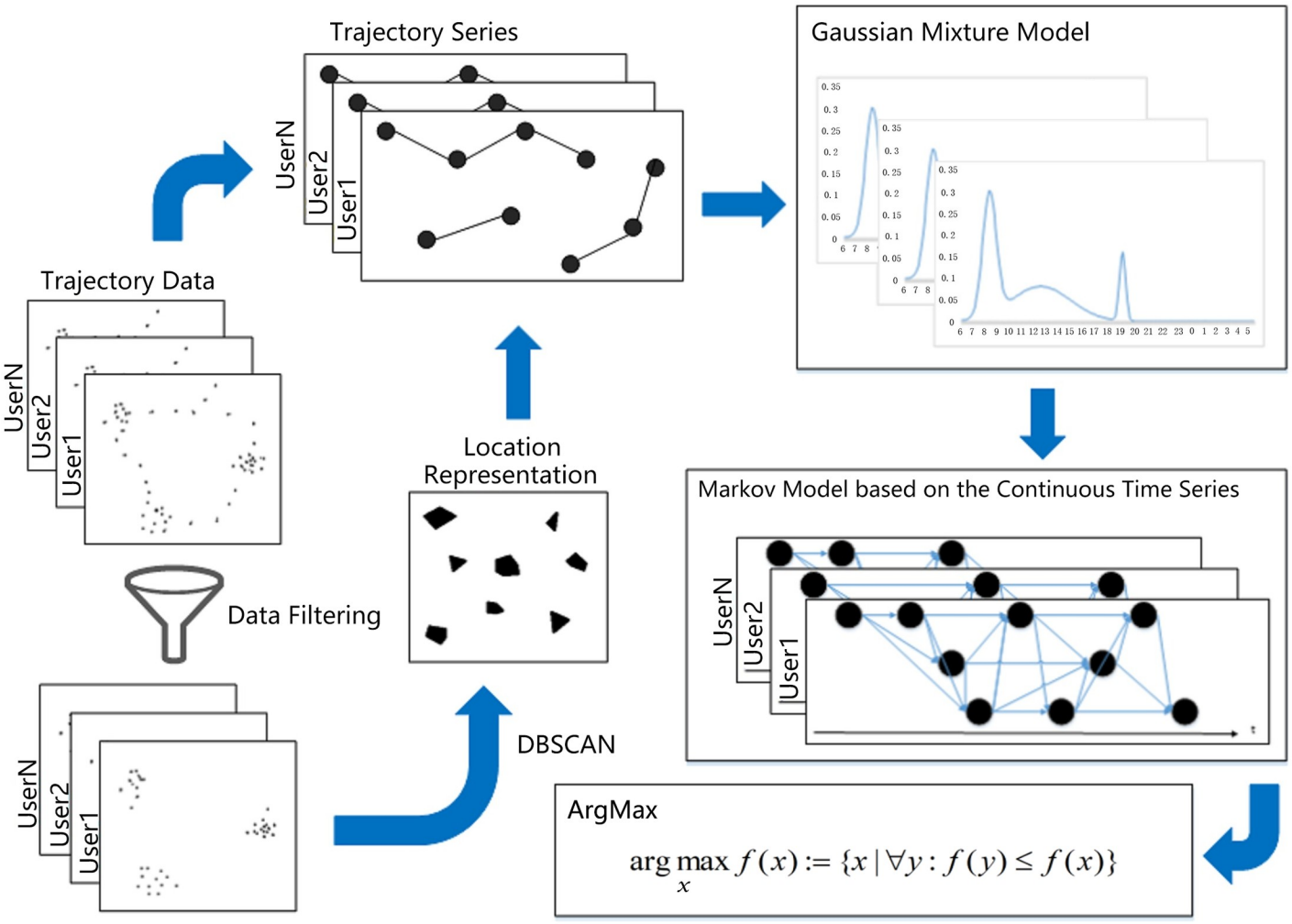


Fig 1. Location prediction based on the real time series.

https://doi.org/10.1371/journal.pone.0207063.g001

Table 1. The symbol definition.

Symbol	Definition
$\mathcal{T}$	The continuous trajectory data series with a starting tracing point and an ending tracing point. $\mathcal{T}_j^i$ denotes the *jth* tracing point within the *ith* trajectory.
L	The location area with a large number of tracing points by clustering, such as business area and park. L_i denotes the *ith* Location.
T	The set of user's sign-in time.
Υ	The set of time for the possible location transition.
∂	Threshold for the walking time.
$\mathcal{I}_k^{i,j}$	The *kth* trajectory from location L_i to L_j.
M	Probability matrix for location transition.
ξ	Threshold for the location transition time.

https://doi.org/10.1371/journal.pone.0207063.t001

1. Filtering the drift tracing point and rebuild the trajectory. The sample is shown in Fig 2. The drifting phenomenon is caused by the strength of GPS signal and the satellite switching. The filtering rule is shown in the following:
   Rule 1. For $\mathcal{T}_j^i \in \mathcal{T}^i, (j = 1, 2 \ldots N)$, if $|\mathcal{T}_j^i \mathcal{T}_{j-1}^i| > \xi$, then delete $\mathcal{T}_j^i$ in $\mathcal{T}^i$.
   Here, $|\mathcal{T}_j^i \mathcal{T}_{j-1}^i|$ denotes the distance between tracing point $\mathcal{T}_j^i$ and $\mathcal{T}_{j-1}^i$. ξ is the threshold.
2. Filtering the tracing point with a higher speed than the average walking speed.
   The pedestrian speed is relatively slow usually in the candidate location where the traffic is

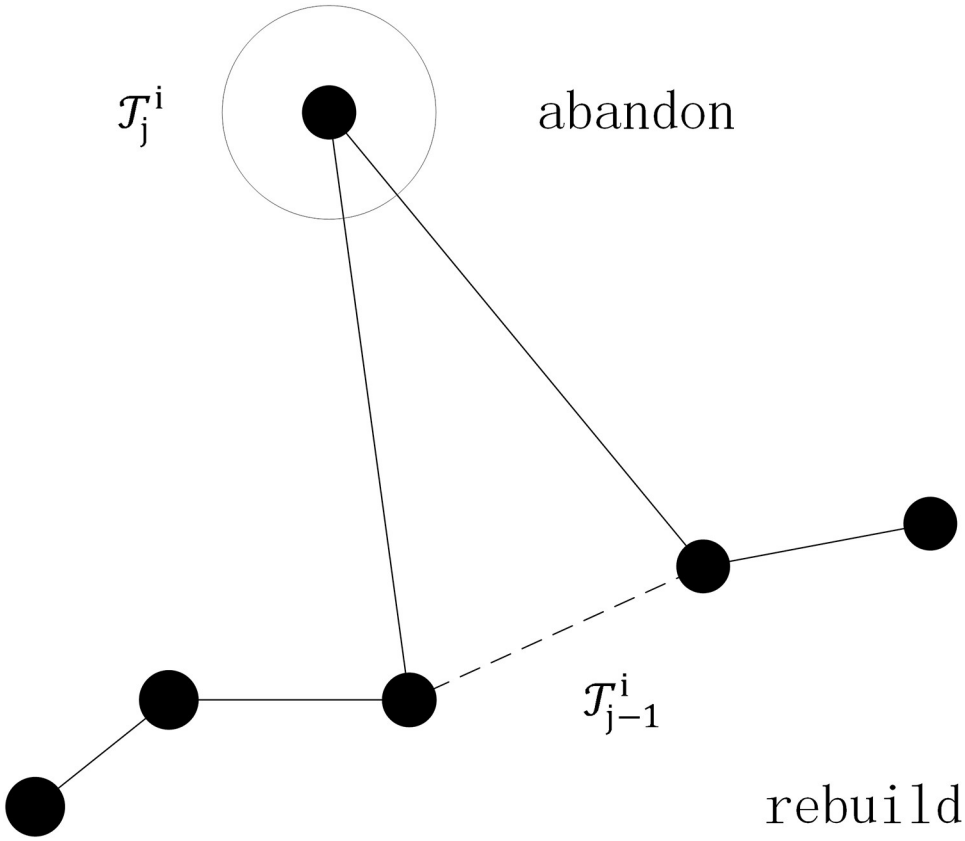


Fig 2. The drift tracing point filtering.

https://doi.org/10.1371/journal.pone.0207063.g002

greater than the regional carrying capacity. On the other hand, there exists the activity which can only be carried at lower speed, such as playing and watching.

For $\mathcal{T}_j^i \in \mathcal{T}^i, (j = 1, 2 \ldots N-1)$, the average speed v_j is computed by formula 1.

$$v_j = \frac{\frac{|\mathcal{T}_{j-1}^i \mathcal{T}_j^i|^2}{\Delta t_{j-1,j}} + \frac{|\mathcal{T}_j^i \mathcal{T}_{j+1}^i|^2}{\Delta t_{j,j+1}}}{|\mathcal{T}_{j-1}^i \mathcal{T}_j^i| + |\mathcal{T}_j^i \mathcal{T}_{j+1}^i|} \tag{1}$$

The filtering rule is:

Rule 2. if $(v_j > \eta)$, then discard the tracing points $\mathcal{T}_{j-1}^i, \mathcal{T}_j^i, \mathcal{T}_{j+1}^i$ in $\mathcal{T}^i$.

Here, η value is set to 1.5m/s which is faster than the average walking speed.

3. Filtering the trajectory with shorter residence time.
   The trajectory with shorter residence time has a little chance to be the part of the candidate location. The filtering rule:
   Rule 3. if $\Delta t_{0,N}(\mathcal{T}^i) < \partial$, then delete $\mathcal{T}^i$.
   Here, N denotes the tracing point number in trajectory $\mathcal{T}^i$ and $\Delta t_{0,N}(\mathcal{T}^i)$ represents the residence time for trajectory $\mathcal{T}^i$. The threshold ∂ value is set to 20 minutes which is also used by Huang [18] and Zheng [20] to extract meaningful human activities.

After filtering the noisy tracing point, we adopt the DBSCAN clustering algorithm to discover the candidate location. K-means and DBSCAN are the most commonly used algorithms for location clustering. However, K-means algorithm needs to set the number of clusters in advance, and it is not suitable for the experimental dataset GeoLife with the larger size and dispersion. We also try the Birch algorithm which cannot distinguish the tracing points by walking. DBSCAN is a density-based spatial clustering algorithm. It groups points together that are closely packed. The clusters with different shapes can be discovered and it is not needed to determine the cluster number in advance, and so DBSCAN algorithm is better to mine the potential locations.

Location prediction method based on the Gaussian Mixture Model (GMM)

GMM

The prediction model of a user's location L_r in the time t is shown in formula 2.

$$L_r = arg\ max_{L_k} P(L_k|t)\ (k = 1 \ldots n) \tag{2}$$

Here, L_r represents the location with maximum probability value at time t.

It is difficult to compute $P(L_k|t)$ because of the continuity of time t, and we transform it to formula 3.

$$P(L_k|t) = \frac{P(t|L_k)P(L_k)}{P(t)} \tag{3}$$

Here, $P(t)$ denotes the sign-in probability for the time t and $P(t|L_k)$ denotes the sign-in distribution for location L_k within a day. $P(L_k)$ denotes the sign-in probability of location L_k.

Therefore, the location prediction model is shown in formula 4.

$$L_r = arg\ max_{L_k} \frac{P(t|L_k)P(L_k)}{P(t)} \tag{4}$$

The sign-in time $\hat{t}$ in the dataset is discrete, the GMM is used for modeling $P(t)$ on the continuous time t by formula 5. The Gaussian Mixture Model is a combination of multiple Gaussian distributions. Here, α_i denotes the weight of each Gaussian distribution in GMM.

$$P(t) = \Sigma_{i=1}^{N} \alpha_i N(t; \mu_i, \sigma_i^2) \ (\alpha_i \geq 0, \Sigma_{i=1}^{N} \alpha_i = 1) \tag{5}$$

Here, $N(t; \mu_i, \sigma_i^2)$ is the Gaussian distribution in the time t and it is computed by formula 6.

$$N(t; \mu_i, \sigma_i^2) = \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{(t-\mu_i)^2}{2\sigma_i^2}} \tag{6}$$

It is similar to compute $P(t|L_k)$ by the modeling with GMM shown in formula 7.

$$P(t^{(k)}|L_k) = \Sigma_{j=1}^{M_k} \beta_j N(t^{(k)}; \mu_j, \sigma_j^2) \ (\beta_j \geq 0, \Sigma_{j=1}^{M_k} \beta_j = 1) \tag{7}$$

Here, $t^{(k)}$ represents the sign-in time for location L_k. $P(L_k)$ is computed by formula 8.

$$P(L_k) = \frac{|L_k|}{\sum_i |L_i|} \tag{8}$$

Here, $|L_k|$ denotes the tracing point number for location L_k and $\Sigma_i|L_i|$ denotes the total tracing point number for all locations.

Finally, the location prediction model is shown in formula 9.

$$\begin{aligned} L_r &= \underset{L_k}{arg\,max} \frac{\sum_{j=1}^{M_k} \beta_j N(t^{(k)}; \mu_j, \sigma_j^2) \frac{|L_k|}{\sum_i |L_i|}}{\sum_{i=1}^{N} \alpha_i N(t; \mu_i, \sigma_i^2)} \\ &= \frac{arg\,max_{L_k} |L_k| \sum_{j=1}^{M_k} \beta_j N(t^{(k)}; \mu_j, \sigma_j^2)}{\sum_i |L_i| \sum_{i=1}^{N} \alpha_i N(t; \mu_i, \sigma_i^2)} \end{aligned} \tag{9}$$

GMM training algorithm

The sign-in frequency for different location L_k is various and so the Gaussian distribution in formula 6 is different. It is needed to adjust dynamically. The algorithm is shown in Table 2.

Here, EM($\overrightarrow{T_i}$,d,e) represents that the GMM parameters are achieved by EM algorithm [21]. The Expectation-Maximization (EM) algorithm is an iterative method to find maximum likelihood of parameters in statistical models, where the model depends on unobserved latent variables. In our paper, EM algorithm is used to solve the parameters of Gaussian Mixture Model. If the Gaussian Mixture coefficient is lower than the threshold, the number of Gaussian distribution will be decreased and the Maximum error value will be increased for retraining. Here, we set the p_threshold to 0.01. Although GMM can predict the location by the use of time information, it can't take into account the context of the trajectory data which limits the prediction performance.

Location prediction by Markov Model based on the continuous time series(CTS-MM)

Markov model is a stochastic model which assumes that future states depend only on the current state, not on the events that occurred before it. The standard Markov Model cannot give the location prediction based on continuous time series. It only takes into account the context of the trajectory data without the time information. And furthermore, the discrete time series is needed to transform to the continuous time which is simulated by the Gaussian Mixture Distribution with EM algorithm. We put forward the Markov Model based on the continuous

Table 2. GMM training algorithm for different location.

Algorithm 2. GMM Training Algorithm for Different Location

Input:

$$\overrightarrow{T} = \begin{pmatrix} \{t_0^{(0)}, t_1^{(0)}, t_2^{(0)} \cdots t_{N_0}^{(0)}\} \\ \{t_0^{(1)}, t_1^{(1)}, t_2^{(1)} \cdots t_{N_1}^{(1)}\} \\ \cdots \\ \{t_0^{(M)}, t_1^{(M)}, t_2^{(M)} \cdots t_{N_M}^{(M)}\} \end{pmatrix}$$ The time t of different tracing point for different M locations

D: Initial number of Gaussian Model in GMM
E: Maximum error value
p_threhold: Threshold for Gaussian Mixture Coefficient
fix_rate: Ratio for error modification
Output:

$$\overrightarrow{\lambda} = \begin{pmatrix} \overrightarrow{\beta_0} & \overrightarrow{\mu_0} & \overrightarrow{\sigma_0} \\ \overrightarrow{\beta_1} & \overrightarrow{\mu_1} & \overrightarrow{\sigma_1} \\ \vdots & \vdots & \vdots \\ \overrightarrow{\beta_M} & \overrightarrow{\mu_M} & \overrightarrow{\sigma_M} \end{pmatrix}$$ Parameter for M Gaussian Mixture Models

Begin
  For $\overrightarrow{T_i}$ *in* $\overrightarrow{T}$:
    Begin
      d = D //Parameter Initialization
      e = E // Parameter Initialization
      do
        $\left(\overrightarrow{\beta_i} \quad \overrightarrow{\mu_i} \quad \overrightarrow{\sigma_i}\right)$ = EM($\overrightarrow{T_i}$, d, e)// Solve the *lth* Gaussian Mixture Model by EM Algorithm
        d = d-1
        e = e + *fix_rate*// Update the parameter iteratively
      while(d >1&& min($\overrightarrow{\beta_i}$) < *p_threhold*)
      // Gaussian Model number is more than 1 and min($\overrightarrow{\beta_i}$) is smaller than the threshold
$\overrightarrow{\lambda_i} = \left(\overrightarrow{\beta_i} \quad \overrightarrow{\mu_i} \quad \overrightarrow{\sigma_i}\right)$
    End
End

https://doi.org/10.1371/journal.pone.0207063.t002

time series (CTS-MM), which considers not only the geographic feature in trajectory data but also the time feature.

CTS-MM

It is known that the user visits location L_i in the time t and we want to predict the user's next location aftertime interval Δt. The model is shown in formula 10.

$$L_r = arg\ max_{L_k} P(L_k | L_i, t, \Delta t) \tag{10}$$

The CTS-MM is used to modeling the user's visiting sequences, which is shown in Fig 3.

There are three locations L_1, L_2 and L_3 in Fig 3. Each node represents the possible transition time. For example, node A denotes a transition time point for location L_2 and it can be transferred to location L_1 or L_3 after time interval ξ. For the location L_2 at the time t, the black arrow and black node in Fig 3 represent the status transition process during the time interval Δt.

The value of $P(L_k \mid L_i, t, \Delta t)$ can be calculated with HMM shown in formulas 11 and 12.

$$P(L_k | L_i, t, \Delta t) = \Sigma_l P(L_k, \mathcal{I}_l^{i,k} | L_i, t, \Delta t) \tag{11}$$

$$P(L_k, \mathcal{I}_l^{i,k} | L_i, t, \Delta t) = \\ P(L_i \rightarrow L_a) \times P(L_a | \Upsilon_\alpha^{(a)}) \times P(L_k, \mathcal{I}_l^{a,k} | L_a, \Upsilon_\alpha^{(a)}, \Delta t - \Upsilon_\beta^{(k)} + \Upsilon_\alpha^{(a)}) \quad (12)$$

Here, $\mathcal{I}_l^{i,k}$ denotes the *lth* trajectory from location L_i to L_k and L_a is the first transferred location after L_i in $\mathcal{I}_l^{i,k}$. The first item of formula 12 is the transition probability from location L_i to L_a. The second item is the conditional probability with formula 3. The third item is a recursion item which represents the transition probability from L_a to L_k. The variable $(L_k, \mathcal{I}_l^{a,k} | L_a)$ denotes the transferring status from L_a to L_k. The variable $(\Delta t - \Upsilon_\beta^{(k)} + \Upsilon_\alpha^{(a)})$ denotes that the user will change the location from L_a to L_k aftertime interval $(\Delta t - (\Upsilon_\beta^{(k)} - \Upsilon_\alpha^{(a)}))$. $\Upsilon_\beta^{(k)}$ represents the βth transition time for L_k and $\Upsilon_\alpha^{(a)}$ represents the αth transition time for L_a.

It is needed to get the transition time for each location. $\Upsilon^{(i)} = \{t_1, t_2, \ldots, t_n\}$ represents thetransition time series for location L_i and they are labeled as the node in Fig 3. The possible transition time is extracted from the training data. Firstly, the location for every sign-in time is recognized. And then select the marginal sign-in time when the location transition occurs. Finally, all of the marginal sign-in time is clustered and the center of each cluster $t_1, t_2, \ldots, t_n$ is selected as the transition time series. In addition, a random bias ξ for t_i is used to simulate the interval of status transition.

Location prediction algorithm

The location prediction algorithm based on the time series is shown in Table 3, which is implemented by the recursion strategy. For start location L_i, predict the next location with the maximum probability after time interval Δt. The array P records the prediction probability for each location.

The probability distribution for each location is calculated recursively. The transition time for location L_a is recorded in the vector $\Upsilon^{(a)}$, and if there is more time to transfer to next location, the algorithm ***TDLP*** will be implemented recursively. Otherwise, the recursion will be stopped.

Here, ξ is a random value used to simulate the transition time interval. It means that user may change location after time interval ξ. We give the experiments to set different ξ value, and it denotes that the better performance is achieved with $\xi = 5$ minutes, which is shown in the experimental section.

Experimental analysis

DataSet

GeoLife, developed by Microsoft, is a location-based social-network project. It enables users to share life experience and build connections among each other by using location history. Furthermore, it contains 182 users' travel records and total of 17621 trajectories.

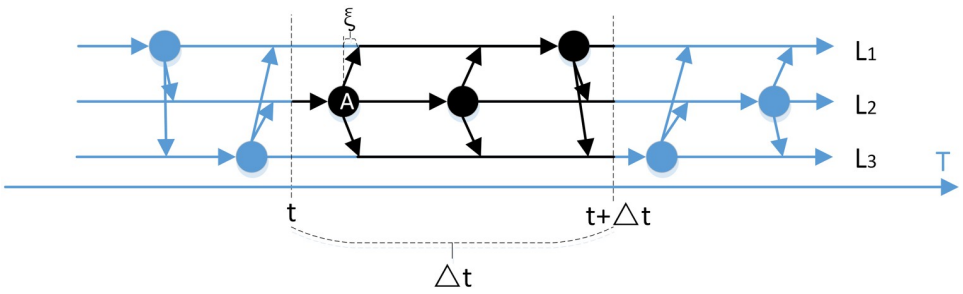


Fig 3. Status transition sample by CTS-MM.

https://doi.org/10.1371/journal.pone.0207063.g003

Table 3. Time-Dependent Location Prediction Algorithm (TDLP).

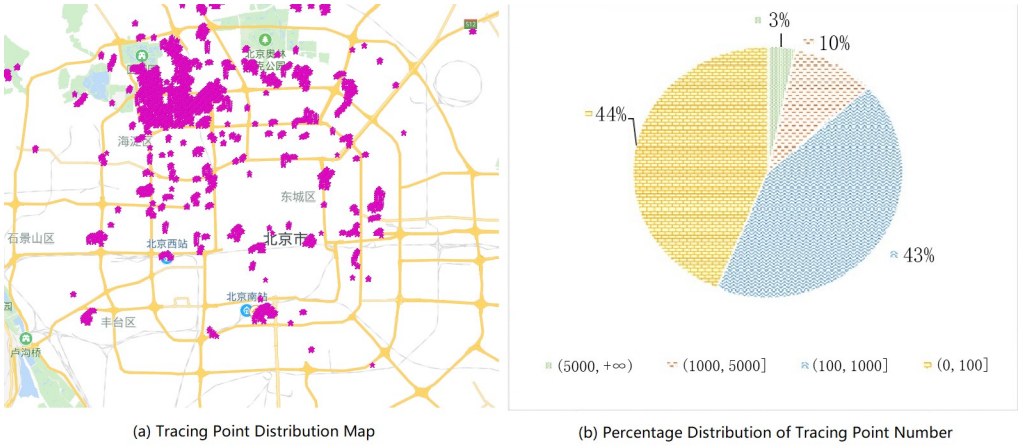
Algorithm 3. TDLP(Time-Dependent Location Prediction Algorithm)

Global Variable:$\mathcal{P} = (P(L_0L_i, t, \Delta t), P(L_1L_i, t, \Delta t), \ldots, P(L_mL_i, t, \Delta t))$
$\mathcal{P}_{L_a} = P(L_aL_i, t, \Delta t)$
Input:
L_i: Start Location
t_{now}: Start Time
Δt: Transition Time Interval
$\Upsilon = (\Upsilon^{(0)}, \Upsilon^{(1)}, \ldots, \Upsilon^{(m)})$: Transition Time Series for Each Location
$M = \begin{pmatrix} P(L_0L_0) & \cdots & P(L_mL_0) \\ \vdots & \ddots & \vdots \\ P(L_mL_0) & \cdots & P(L_mL_m) \end{pmatrix}$: Location Transition Probability Matrix
P_{cur}: The Current Probability with the Initial Value of 1
Output: argmax $\mathcal{P}$

```
Begin:
  For 0≤ a≤ m
    Begin
      Get the next transition time Υ^(a)_next in Υ^(a) relative to t_now
      if (Υ^(a)_next − t_now ≥ Δt)
        Begin
          P(L_a|L_i, t, Δt) = P(L_a|L_i, t, Δt) + P_cur
          Continue;
        End
      else
        Begin
          P_cur = P(L_a, T^{i,a}L_i, t_now, Δt)// Compute by formula 12.
          //Recursion
          TDLP(La, Υ^(a)_next + ξ, Δt − (Υ^(a)_next − t_now − ξ), Υ, M, P_cur)
        End
  End
End
```

https://doi.org/10.1371/journal.pone.0207063.t003

Most of the tracing points in the dataset are located in Beijing and our experiment also gives the prediction and analysis for location here. The statistical data distribution after filtering and clustering is shown in Fig 4. The parameter of maximum density for DBSCAN clustering algorithm is set to 0.0005 and the minimum size of a cluster is set to 20.



(a) Tracing Point Distribution Map

(b) Percentage Distribution of Tracing Point Number

Fig 4. Tracing point distribution on GeoLife.

https://doi.org/10.1371/journal.pone.0207063.g004

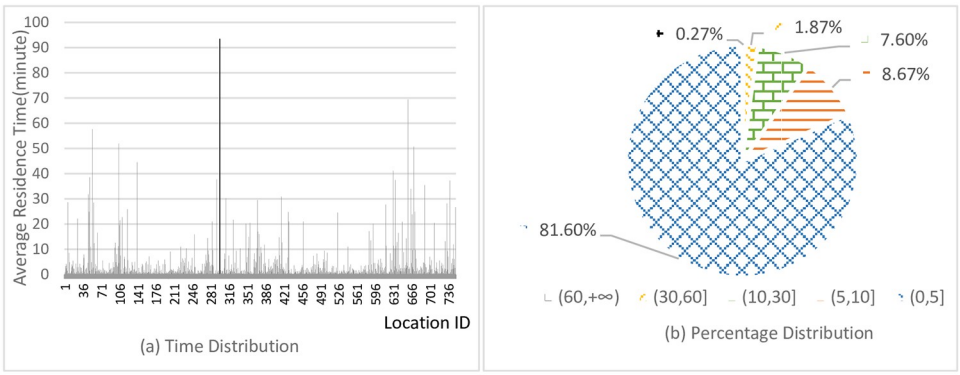


Fig 5. Average residence time distribution.

https://doi.org/10.1371/journal.pone.0207063.g005

Most locations, around 87%, have only a little tracing points(less than 1000). For total 182 users, average sign-in frequency in these locations is less than 6.

The average residence time distribution for different locations is shown in Fig 5. It is also found that average residence time is less than 5 minutes for more than 80% locations.

It can be concluded that the distribution of the tracing points is dispersive and user's moving frequency is relatively fast, which will play an important role in experiment performance.

Experimental result

Among the total 182 users, there are 22 users who have trajectory less than 4 and the distribution is shown in Fig 6. They are filtered from the dataset. For the remaining 160 users, we select their trajectory data in adjacent 20 days as training set and the data of next 5 days is used as test set. For users who have trajectory data less than 25 days, we choose 80% of it for training and the rest for testing. The evaluation metric is precision and itis computed by $Precision = A/B$. Here, A denotes the number of correct samples by prediction and B denotes the total number of samples by prediction.

Clustering result for location discovery. The trajectory distribution has a big difference between weekdays and weekends. We divide the data into two groups for clustering respectively. The sample of clustering results on weekdays and weekends data are shown in Fig 7(a)–7(d).

As it can be seen from Fig 7, there are more tracing points on weekdays than weekends because more activities happened within five weekdays. The DBSCAN algorithm is effective for location clustering, and the boundary of different clusters are clearly in line with reality. Here, the clusters with different color denote different discovered locations.

Prediction performance impact by different parameter

- **Impact by different time interval Δt**

The evaluation result of location prediction after different time interval Δt is shown in Fig 8.

The prediction precision of GMM is shown in Fig 8(a). For different time interval Δt, the performance of CTS-MM are shown in Fig 8(b)–8(f) separately. It is noticed that the start time of Fig 8(f) is set to 1 o'clock because sign-in behavior across different day is not considered.

We find that the performance varies with time t on these two models: GMM and CTS-MM. The precision is higher in early of the morning and lower in other time on GMM, which is shown in Fig 8(a). On the other hand, the prediction performance of CTS-MM shown in

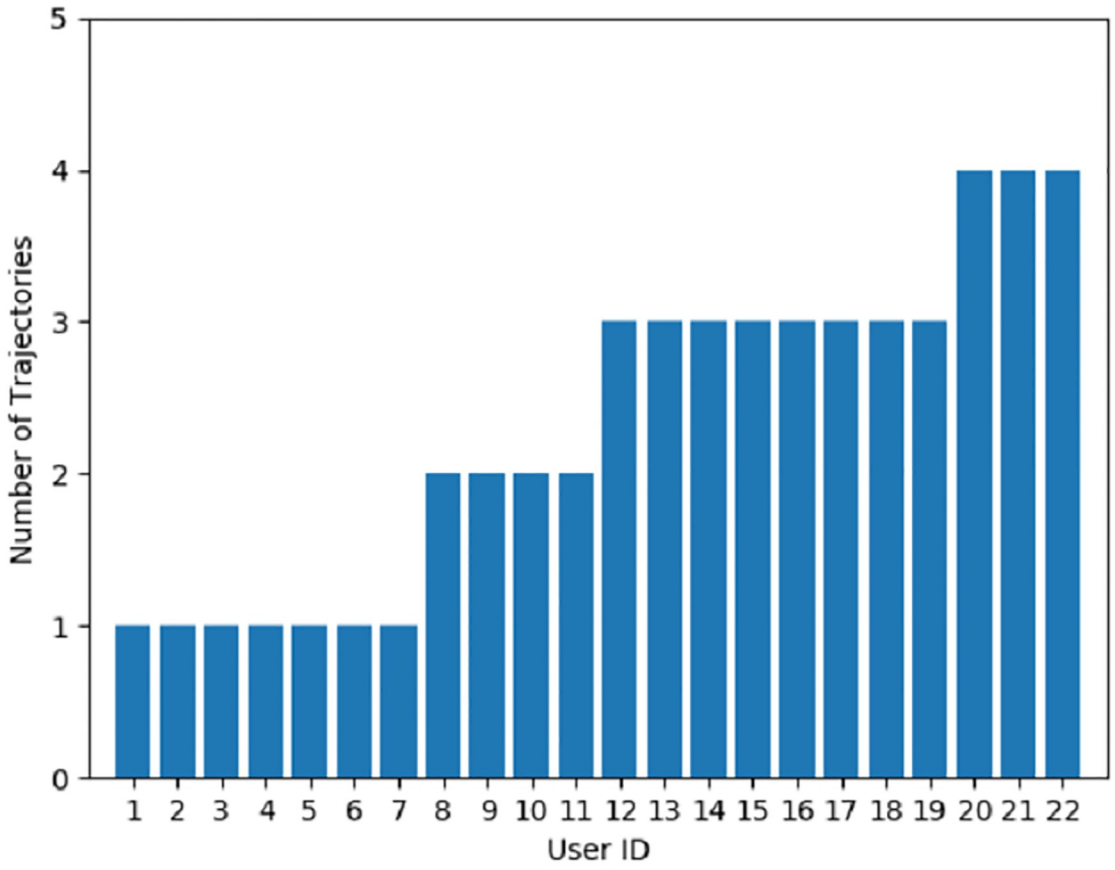


Fig 6. Trajectory distribution of 22 users filtered.

https://doi.org/10.1371/journal.pone.0207063.g006

Fig 8(b)–8(f) is varied with the increasing of time interval Δt which brings too much travel uncertainty. Especially, the precision is higher around 18 o'clock than other time mostly.

- **Impact by different ξ value**

The parameter ξ mentioned in Table 3 is used to simulate the transition time interval between different locations. We give the experimental results with different ξ value in Table 4.

With different prediction time interval Δt, the prediction performance is better when ξ is set to 5 minutes. And the precision gets 0.451 when time interval Δt is set between 10 minutes to 30 minutes.

Prediction performance by different user. We give experiment on different users and the comparison results on both GMM and CTS-MM with different time intervals Δt are shown in Fig 9(a)–9(f).

Compared with the average precision, each user's prediction performance deviated largely (no more than ±15%). The performance of CTS-MM is improved about 12% with the comparison of GMM when time interval Δt is shorter than one hour. It declines significantly after one hour, which is shown in Fig 9(f). The user's travel uncertainty increases with longer duration time.

Discussion. The prediction performance is evaluated on GeoLife dataset and the average precision is about 43% by CTS-MM. Compared with other related algorithms shown in Table 5, fewer features are used and the precision is higher by CTS-MM.

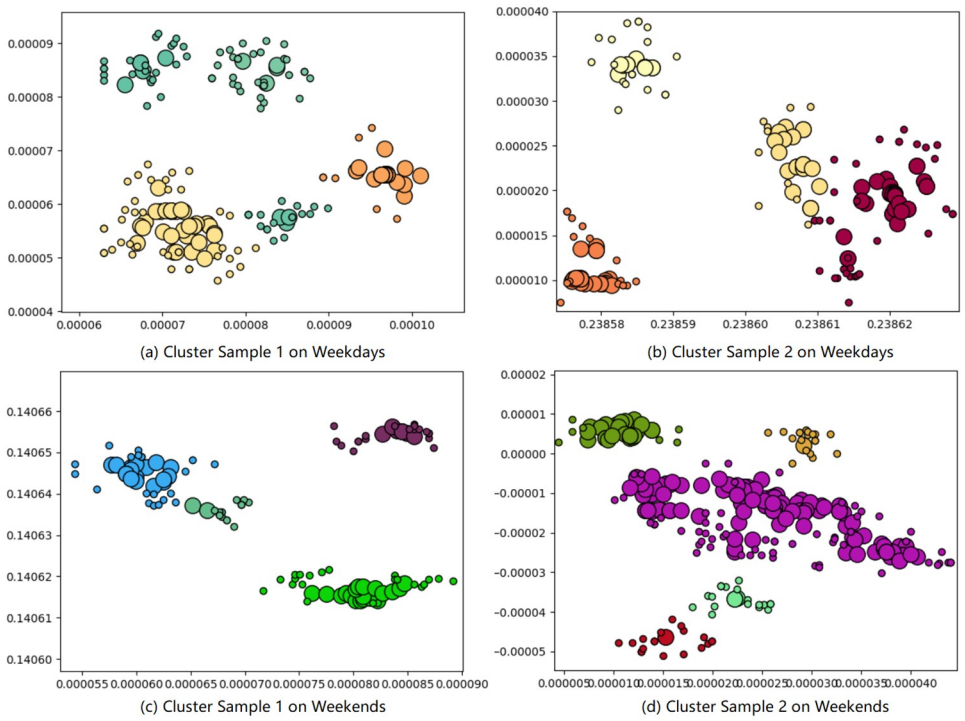


Fig 7. Clustering result sample on weekdays and weekends.

https://doi.org/10.1371/journal.pone.0207063.g007

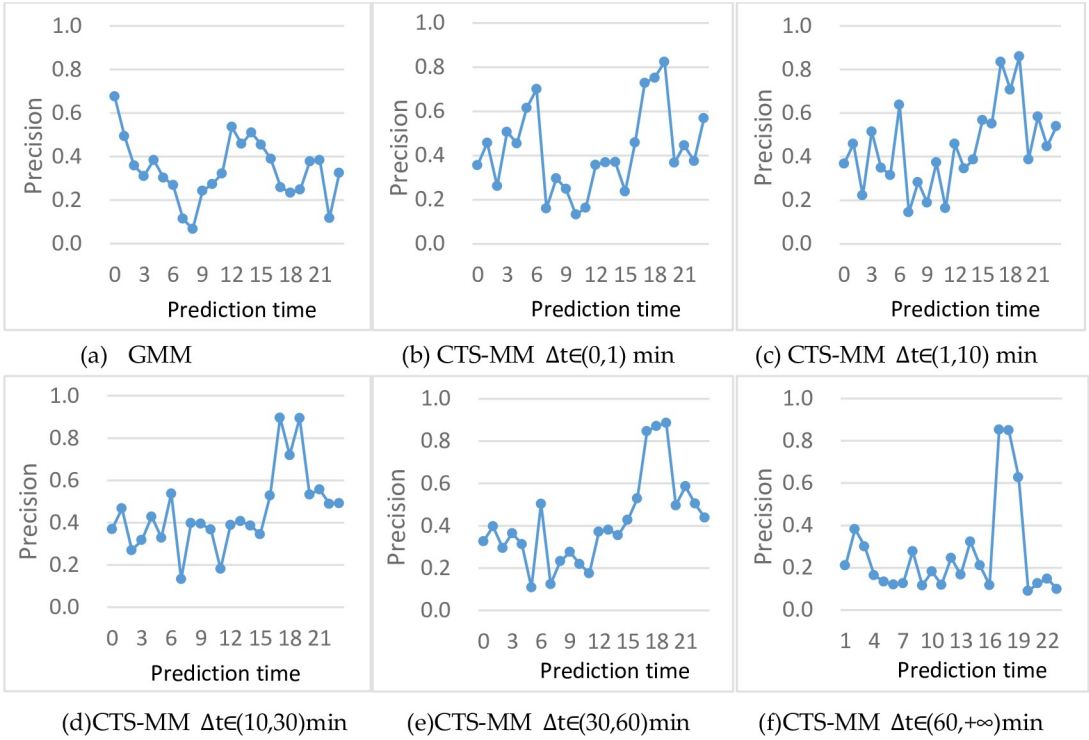


Fig 8. Prediction performance by different time interval Δ*t*.

https://doi.org/10.1371/journal.pone.0207063.g008

Table 4. Prediction precision by different ξ.

ξ(min)	$\Delta t < 1$	$1 < \Delta t < 10$	$10 < \Delta t < 30$	$30 < \Delta t < 60$	$\Delta t > 60$	Average
0	0.416	0.437	0.436	0.394	0.241	0.3848
3	0.421	**0.450**	0.439	0.402	0.248	0.3922
5	**0.425**	0.445	**0.451**	**0.418**	**0.260**	**0.3998**
7	0.425	0.421	0.426	0.394	0.239	0.3810

https://doi.org/10.1371/journal.pone.0207063.t004

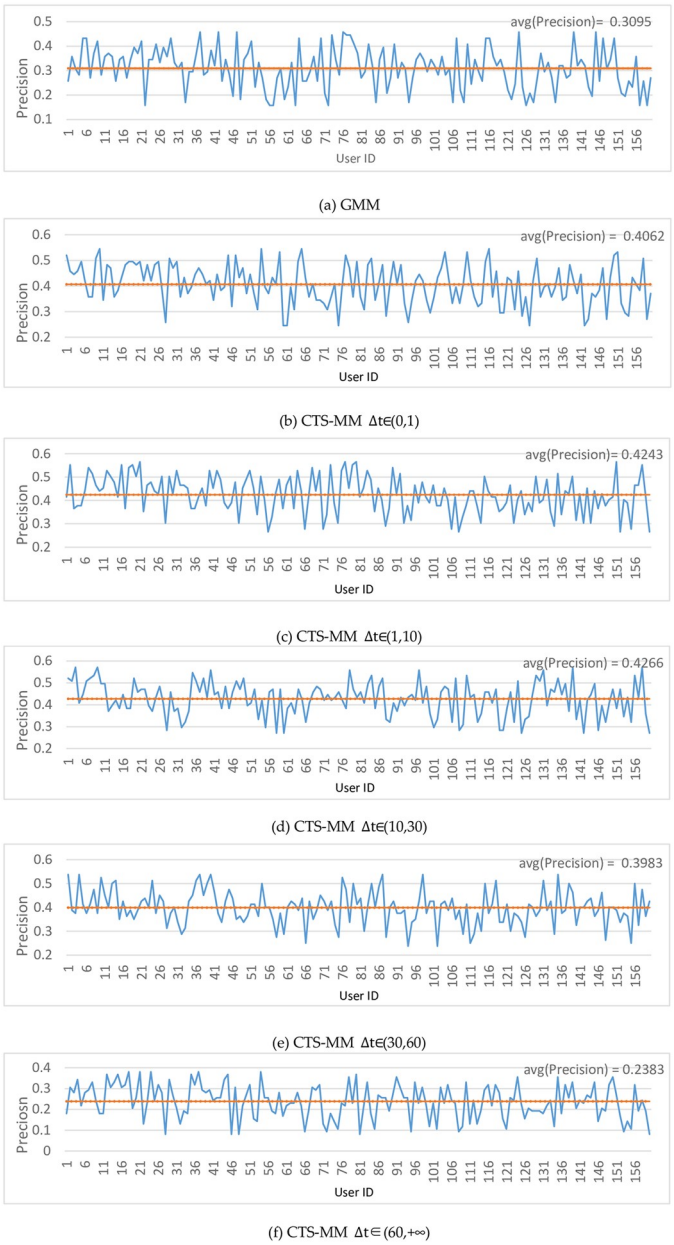


Fig 9. The average precision distribution on different user by different time interval.

https://doi.org/10.1371/journal.pone.0207063.g009

Table 5. Precision of different methods.

	Predict on real time	The best Precision	Features		
			Social Information	Trajectory Information	Sign-in Information
CTS-MM	√	0.4266		√	
GMM	√	0.3095		√	
PST [22] (L = 1)		0.40 ~ 0.41		√	
RCH [23](P = 10)		0.35 ~ 0.40	√	√	√

https://doi.org/10.1371/journal.pone.0207063.t005

Our proposed model CTS-MM and GMM can give prediction on continuous time and furthermore CTS-MM performs better than GMM improved by approximately 12% on precision. The trajectory information feature is used for training by these two models. A common problem with related work is the inability to make location predictions on real continuous time. In addition, Song [24]also gives the location prediction by part of the better trajectory with ID number 0 and 17 on GeoLife and the precision varies from 20% to 80%. But the result on all of the trajectory data is not given.

Location prediction on continuous real-time data is a challenging task and our CTS-MM model makes it possible to predict for any time parameter t. The real time information is used for training and the model is applied on real time series for prediction, which makes result more reasonable. Our work is implemented on the total dataset of GeoLife and the average precision on different time interval varies from 10% to 90%. Within the time interval of 1 hour, the average prediction precision of all the users varies from 20% to 60%.

We have filtered 22 users who have fewer trajectory from the GeoLife data, and the robustness of our model is not checked on the sparse trajectory data. It may make a difference in prediction performance and the further analysis will be our next step. An effective location prediction can bring a better user experience, such as advertising directed at customers based on their current location.

Conclusion

With the increasing of user's location data, the application of location based service becomes more and more popular and it is important to provide superior service for the user. The most common method for location prediction is Markov Model. But it only considers the user's location movement sequence and it is impossible to give a prediction related to the time information. We put forward a new method by Markov Model based on the Continuous Time Series(CTS-MM). The Bayes model and GMM are used for modeling the posterior probability of the location with continuous time series. The discrete status sequence of the HMM is changed to continuous sequence, which enables the model to predict the location in different real time. The experimental results on GeoLife data denote that the proposed model achieves a higher precision than traditional methods. The distribution of the tracing points on GeoLife is dispersive and the user's moving frequency is relatively fast, which makes the prediction task more challenging. Specially, with the increasing of time interval Δt, the precision will decline because it brings much travel uncertainty.

In future work, other models will be considered to improve prediction performance. At the same time, some other effective information, such as sign-in data and user data, will be used for assistance.

Supporting information

S1 File. Geolife Trajectories 1.3.part01.rar. This file is part of the Microsoft dataset.
(RAR)

S2 File. Geolife Trajectories 1.3.part02.rar. This file is part of the Microsoft dataset.
(RAR)

Author Contributions

Conceptualization: Wenyang Guo.

Data curation: Dongyue Zhao.

Methodology: Yongping Du.

Writing – original draft: Yanlei Qiao.

Writing – review & editing: Chencheng Wang.

References

1. Li Y, Guo A, Liu S, Gao Y, Zheng YT. A location based reminder system for advertisement. International Conference on Multimedea. 2010;1501–1502.
2. Guo B, Fujimura R, Zhang DQ, Imai M. Design-in-play: improving the variability of indoor pervasive games. Multimedia Tools & Applications, 2012; 59(1):259–277.
3. Zheng Y, Xie X, Ma WY. GeoLife: A Collaborative Social Networking Service among User, Location and Trajectory. Bulletin of the Technical Committee on Data Engineering. 2011; 33(2):32–39.
4. Zheng Y. Trajectory Data Mining: An Overview. ACM Transaction on Intellignet Systems and Technology. 2015; 6(3):41.
5. Zheng K, Zheng Y, Xie X, Zhou XF. Reducing Uncertainty of Low-Sampling-Rate Trajectories. IEEE, International Conference on Data Engineering. 2012;1144–1155.
6. Xue AY, Zhang R, Zheng Y, Xie X, Huang J, Xu ZH. Destination prediction by sub-trajectory synthesis and privacy protection against such prediction. IEEE, International Conference on Data Engineering. 2013;254–265.
7. Peng Q, Ding ZM, Guo LM. Prediction of trajectory based on Markov chains. Computer Science. 2010; 37(8):189–193.
8. Song C, Qu Z, Blumm N, Barabási A-L. Limits of predictability in human mobility. Science. 2010; 327 (5968):1018–1021. https://doi.org/10.1126/science.1177170 PMID: 20167789
9. Yu RY, Xia XY, Li J, Zhou Y, Wang XW. Social-Aware Mobile User Location Prediction Algorithm in Participatory Sensing Systems. Chinese Journal of Computers. 2015; 38(2):374–385.
10. Lian DF, Xie X, Zheng VW, Yuan NJ, Zhang FZ, Chen EH. CEPR: A Collaborative Exploration and Periodically Returning Model for Location Prediction. Acm Transactions on Intelligent Systems and Technology. 2015; 6(1):8.
11. Lian DF, Zhu Y, Xing X, Chen EH. Analyzing Location Predictability on Location-Based Social Networks.Advances in Knowledge Discovery and Data Mining. 2014;102–113.
12. Zhang C, Liang H, Wang K, Sun JL. Personalized Trip Recommendation with POI Availability and Uncertain Traveling Time. Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. 2015;911–920.
13. Wu F, Li ZH. Where Did You Go: Personalized Annotation of Mobility Records. Proceedings of the 25th ACM International on Conference on Information and Knowledge Management. 2016;589–598.
14. Nghia DT, Schilling N, Schmidt-Thieme L. Near Real-time Geolocation Prediction in Twitter Streams via Matrix Factorization Based Regression. Proceedings of the 25th ACM International on Conference on Information and Knowledge Management. 2016;1973–1976.
15. Gambs S, Killijian M-O, Cortez MNDP. Next place prediction using mobility Markov chains. Proceedings of the First Workshop on Measurement, Privacy, and Mobility. 2012;Article No.3.
16. Mathew W, Raposo R, Martins B. Predicting future locations with hidden Markov models. ACM Conference on Ubiquitous Computing. 2012;911–918.
17. Qiao Y, Si Z, Zhang Y, Abdesslem FB, Zhang X, Yang J. A hybrid Markov-based model for human mobility prediction. Neurocomputing, 2018; Vol.278:99–109.
18. Huang W, Li S, Liu X, Ban Y. Predicting human mobility with activity changes. International Journal of Geographical Information Science. 2015; 29(9):1569–1587

19. Morzy M. Mining Frequent Trajectories of Moving Objects for Location Prediction. International Conference on Machine Learning and Data Mining in Pattern Recognition. 2007;667–680.
20. Zheng Y, Zhang L, Ma Z, Xie X, Ma WY. Recommending friends and locations based on individual location history. ACM Transactions on the Web. 2011; 5(1): Article No.5.
21. Dempster AP, Laird NM, Rubin DB. Maximum Likelihood from Incomplete Data via EM Algorithm. Journal of Royal Statistical Society. 1977; 39(1):1–38.
22. Wang X, Jiang XH, Lin J, Xiong JB. Prediction of moving object trajectory based on probabilistic suffix tree. Journal of Computer Applications. 2013; 33(11):3119–3122.
23. Wang Y, Yuan NJ, Lian DF, Xu LL, Xie X, Chen EH, et al. Regularity and Conformity: Location Prediction Using Heterogeneous Mobility Data. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2015;1275–1284.
24. Song LJ, Meng FR, Yuan G. Moving object location prediction algorithm based on Markov model and trajectory similarity. Journal of Computer Applications. 2016; 36(1):39–43.