Predicting Human Mobility Based on Location Data Modeled by Markov Chains

Junjie Jiang[1], Changchun Pan[1][2][3], Haichun Liu[1][3], Genke Yang[1][3]
[1]Department of Automation of Shanghai Jiao Tong University and Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai, 200240;
[2]Shanghai Key Laboratory of Navigation and Location based Services, Shanghai,200240, P.R. China.
[3]Collaborative Innovation Center for Advanced Ship and Deep-Sea Exploration(CISSE), Shanghai 200240, P.R. China.
{jiangjun, pan_cc, haichunliu, gkyang}@sjtu.edu.cn

***Abstract*—Recently, location-based services have attracted significant attention. Against this background, one pivotal and challenging problem is predicting the future location of a user given his or her current location and associated historical mobility data. Predicting human mobility enables many interesting applications such as navigation services, traffic management and location-based advertisements. In this paper, we first extract the region-of-interest (ROI) from the historical location data. With plenty of statistics the original trajectory is represented by Markov chains composed by many ROIs. To improve the performance of the prediction, we extend 1st-order Markov chains to Kth-order Markov chains by reconstituting the structure of priori knowledge, which is intended to take more significant historical information into consideration. We evaluate the certainty of the prediction outcomes in terms of information entropy. We demonstrate that the prediction using a higher-order Markov chain can be more accurate compared with a 1st-order Markov chain.**

Keywords—stay points detection; region-of-interest extraction; hierarchical clustering; Kth-order markov chains; mobility prediction.

I. INTRODUCTION

Predicting human mobility becomes viable with increasingly available human mobility data, which can be obtained through various channels, such as the Global Positioning System (GPS) and social networks. Mobility prediction is important for the market of location-based services, including navigation services, traffic management and location-based advertisement.

Numerous methods have been dedicated to the identification of human mobility patterns and prediction of human movement from different fields based on human mobility data [1, 2]. To work out the problem of location recognition and prediction, several approaches have been explored, such as hidden Markov model (HMM) [3, 4, 5] and hierarchical dynamic bayesian network (DBN) et al [6, 7, 8]. As the groundwork of LBS, different models have been proposed to extract users' movement patterns [9, 10, 11]. There are some work on discovering ROIs from trajectory databases [12, 13, 14]. In this paper, we apply the concept of region-of-interest (ROI) to the field of mobility prediction by combining hierarchical clustering algorithm and Markov chains. Furthermore, we extend 1st-order Markov chains to Kth-order Markov chains by reconstituting the structure of priori knowledge, considering that the current state and the last (K-1) state affect the choice of the next event. As a result, we acquire a more precise prediction of human mobility.

This paper is organized as follows. In section 2 we describe the concept of stay points, region-of-interest and Markov chains, simultaneously we propose stay points detection algorithm, region-of-interest extraction algorithm and the procedure to construct Markov chains. Afterwards, in section 3 we implement the whole procedure for prediction based on realistic mobility data. We conclude this paper in section 4 finally.

II. METHODOLOGY

A. Architecture

Fig. 1 shows the architecture of the whole process for mobility prediction, which is comprised of the following steps. At the first step, we detect stay points from the user's raw trajectories, which indicate the centroids of small areas, such as dormitories and canteens, where a user stays for a certain amount of time and performs certain activities. Based on these stay points, at the second step, we calculate users' ROIs by implementing a hierarchical clustering algorithm to identify the clusters of stay points which are close to each other. At the third step, we build the Markov chains from the ROIs extracted from the second step to predict the user's mobility. Finally, we apply information entropy to evaluate the certainty of the prediction outcomes. As a result, a higher-order Markov chain reduces the uncertainty of mobility prediction.

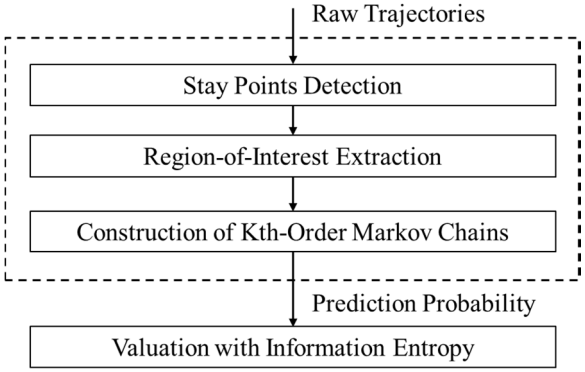

Fig. 1. The architecture of the experiment

978-1-5090-2879-5/16/$31.00 ©2016 IEEE

B. Stay Points Detection

GPS Trajectory: A GPS trajectory T is a sequence of time-stamped points, $T = P_0 \rightarrow P_1 \rightarrow \cdots \rightarrow P_k$, where $P_i = (x_i, y_i, t_i)$, $(i = 0, 1, ..., k)$. Specifically (x_i, y_i) is the coordinate of the point, and t_i is the timestamp.

Stay Point: A stay point *SP* stands for a geographic region where a user stays over a certain time interval. The detection of a stay point depends on two scale parameters, a time threshold (θt) and a distance threshold (θd). Formally, given a trajectory $T: P_0 \rightarrow P_1 \rightarrow \bullet\bullet\bullet \rightarrow P_z$, a single stay point *SP* can be regarded as a virtual location characterized by a sub-trajectory $P_m \rightarrow \bullet\bullet\bullet \rightarrow P_n$, which satisfies the conditions below.

$$Dist(P_m, P_k) < \theta_d, Int(P_m, P_n) > \theta_t \quad (1)$$

for ∀kϵ(m+1,n), Where $Dist(P_m, P_k)$ denotes the geospatial distance between the first point P_k and any other point P_k, and $Int(Pm, Pn)=|P_{m.t} - P_{n.t}|$ is the time interval between the arrival time of the first point and the departure time of the last point. Therefore, we get *SP* = (x, y), where *SP* is the stay point detected.

$$SP.x = \sum_{k=m}^{n} SP_k.x / |SP| \quad (2)$$

$$SP.y = \sum_{k=m}^{n} SP_k.y / |SP| \quad (3)$$

Equation (2) and equation (3) stand for the average x and y coordinates of the stay point *SP*, where |*SP*| stands for the number of points that *SP* contains. Briefly, θt is the minimum time length that a continuous set of GPS points lasts to be a stay point. And θd is the maximum distance between the first point and the last point of a continuous set of GPS points for them to be a stay point.

Algorithm 1:
Stay Point Detection

Input:
A sequence of raw points
A time threshold θt
A distance threshold θd
Output: A set of stay points {SP}

```
Set i=0
Count the number of the raw points N
while i < N do
  j=i+1
  while j <N do
  dist = Dist (Pi, Pj)
  if dist >θd then
    ΔT=Pj.t - Pi.t
    if  ΔT>θt then
    S.coordinate=AverageCoordinate({Pk| i<=k<=j})
    SP.insert(S)
  i=j; break
  j=j+1
return SP
```

C. Region-of-Interest(ROI) Extraction

Given a set of N clusters to be clustered further, the general principle of hierarchical clustering is that the two closest clusters should be combined first.

- Step 1: Start by assigning each point to its own cluster, so that if you have N points, you now have N clusters, each containing just one item. Let the distances between the clusters equal the distances between the points they contain.
- Step 2: Find the closest pair of clusters and merge them into a single cluster, so now you have one less cluster.
- Step 3: Compute distances between the new cluster and each of the old clusters.
- Step 4: Repeat steps 2 and 3 until the distance between the closest pair of clusters is above a threshold.

Before any clustering is performed, it is required to determine the distance between each cluster using a distance function. In hierarchical clustering, the distance between two clusters is defined as the average distance between all the points in different clusters as showed in equation (4). For example, the distance between clusters "r" and "s" in *Fig. 2* is equal to the average length of all the arrows connecting the points in different clusters. The procedure of ROI extraction is showed in algorithm 2.

$$Dist(r,s) = \frac{1}{n_r n_s} \sum_{i=1}^{n_r} \sum_{j=1}^{n_s} Dist(SP_{r.i}, SP_{s.j}) \quad (4)$$

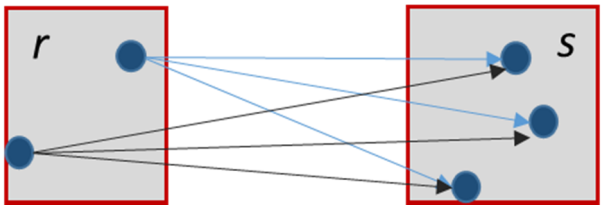

Fig. 2. Measure of the distance between clusters "r" and "s"

Algorithm 2:
Region-of-Interest Extraction

Input:
A set of stay points $\{SP_1, SP_2, \cdots, SP_n\}$
A threshold ΔD
A distance function $Dist(C_i, C_j)$
Output: A set of Region-of-Interest indicated by rectangle

```
for i = 1 to n
  Ci=SPi
end for
C={C1,⋯,Ci}
while Dist(Cmin1,Cmin2)< threshold ΔD do
    Cmin1, Cmin2 = minimum Dist(Ci, Cj) for all Ci, Cj in C
    Remove Cmin1 and Cmin2 from C
    Add {Cmin1, Cmin2} to C
end while
```

D. Markov Chains

Markov Chains describe a state sequence, and the value of each state depends on the prior limited states. It's a system with a set of possible states. At each of a sequence of discrete points in time t >= 0, the system is in exactly one of those states, and the state at time t >= 0 is designated by X_t. The movement from X_t to X_{t+1} is probabilistic, and the probability depends only on the states of the system at or prior to t.

1st Markov Chain: Typically dealing with 1st-order Markov chain, only X_n itself affects the transition probabilities. For each integer n, a 1st Markov chain assigns probability to sequences ($X_1 \ldots X_n$) as follows:

$$p((x_1, x_2, \ldots, x_n)) = p(X_1 = x_1)\prod_{i=2}^{n} p(X_i = x_i \mid X_{i-1} = x_{i-1}) \quad (5)$$

Kth-order Markov chain: The transition probability out of state X_n depend on the values of X_n, $X_{n-1}, \ldots, X_{n-(k-1)}$.

$$p(x_1 \ldots x_n) = p(X_1 = x_1, \ldots, X_k = x_k) \cdot \prod_{i=k}^{n} p(X_i = x_i \mid X_{i-1} = x_{i-1}, X_{i-2} = x_{i-2}, \ldots, X_{i-k} = x_{i-k}) \quad (6)$$

As showed in *Fig. 3*, 1st-order Markov chain means that only the current state affects the choice of the next state. A second-order Markov chain would mean that the current state and the last state affect the choice of the next event. A kth-order Markov chain would indicate that the current state and the last (k-1) states in the sequence affect the choice of the next state.

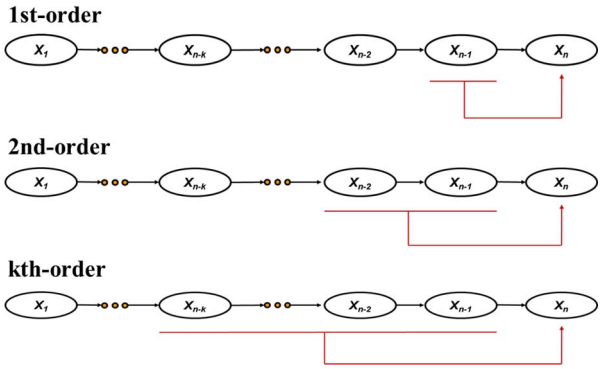

Fig. 3. Markov chains with different order

A mobility Kth-order Markov chain models the mobility behavior of an individual as a discrete stochastic process in which the probability of moving to a state (i.e., ROI) depends on the K previous visited states and the probability distribution of the transitions between states. More precisely, a mobility Kth-order Markov chain is composed of:

- A set of states $X = \{X_1, \ldots, X_k\}$, in which each state corresponds to a ROI. These states generally have intrinsic semantic meanings.
- A set of transitions, such as $P_{i,j}$, which represents the probability of moving from state X_i to state X_j. A transition from one state to itself can occur if the individual has a probability of moving from one state to an occasional location before coming back to this state.

Mobility Markov chains can be represented as a directed graph. The rectangular nodes on behalf of ROIs correspond to different states and there is a directed weighted arrow between two nodes if and only if the transition probability between these two nodes is non-null. The sum of all the weights attached to the arrows which leave from the same node is equal to 1. For example, $P_{AA}+P_{AB}+P_{AC}=1$, as showed in *Fig. 4*.

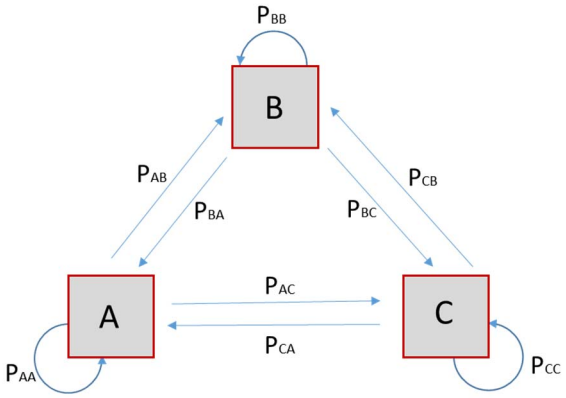

Fig. 4. Markov chains with transition probability

Algorithm 3:
Construction of a Kth-Order Markov Chain

Input:
SPs: the stay points detected
ROIs: the ROI extracted
Output:
The Kth-Order Markov Chain

for each region of interest ROI in ROIs do
Create the corresponding state X_i in the mobility Markov chain
end for
for each mobility trace *SP* in *SP*s do
  if the *SP* is located in the rectangular on behalf of ROI **then**
    Update and label the current stay point with ROI
  end if
end for
Squash all the successive mobility traces sharing the same label into a single occurrence
Compute all the transition probabilities between each pair of states of the Markov chain
return the mobility Markov chain computed

III. EXPERIMENT

In this Section, first we introduce the experimental dataset. Second, we implement the procedure of stay points detection. Third, we conduct the ROI extraction algorithm based on the detected stay points, and in result we get some ROIs represented by rectangles. Fourth, each rectangle is regarded as a state of the Markov chain, and we build the Markov chains from the ROI sequence.

A. Experimental Data Set

This GPS trajectory dataset was collected in (Microsoft Research Asia) Geolife project by 182 users in a period of over five years (from April 2007 to August 2012) [15, 16]. In the GeoLife dataset, a GPS trajectory is represented by a sequence of time-stamped points, each of which contains the information of latitude, longitude and altitude. This dataset contains 17,621 trajectories with a total distance of 1.3 million kilometers and a total duration of 50,176 hours. These trajectories were recorded by different GPS loggers and GPS-phones, and have a variety of sampling rates. 91.5 percent of the trajectories are logged in a dense representation, e.g. every 1~5 seconds or every 5~10 meters per point. The authors claim that the GeoLife dataset recorded a broad range of user's outdoor movements, including not only life routines like going home and going to work, but also some entertainment and sports activities, such as shopping, sightseeing, dining, hiking and cycling. This trajectory dataset can be used in many research fields, such as mobility pattern mining, user activity recognition, location-based social networks, location privacy, location recommendation and location prediction.

Table I shows the raw data format, which consists of three parts. The longitude and latitude of the sampled points tell where the person is, and the timestamp of the sampled points tells temporal information. Fig. 5 is the visualization of the person's raw trajectory whose ID in the dataset is 000.

TABLE I. THE FORMAT OF RAW DATA

Longitude	Latitude	Time Stamp
116.296427	40.011189	2008-11-11 02:01:59
116.296374	40.011168	2008-11-11 02:02:04
116.296339	40.011148	2008-11-11 02:02:09

B. Stay Points Detection

In this section, two parameters are necessary before the detecting algorithm. We set the time threshold θt to be 20 minutes and the distance threshold θd to be 100 meters for stay points detection. A stay point stands for a geographic place, where the user stays over a time interval threshold which is set as 20 minutes and within a distance threshold θd which is set as 100 meters. The first point and the last point of a raw trajectory are directly regarded as stay points. If the distance between any two points matchs the above condition, the points will be merged into one stay point by replacing them with the middle point of the line segment connecting them.

Fig. 5. The visualization of raw data

These stay points enable us to find out some significant places, whose pattern of manifestation is that the user stays there for a while. In other words, the user stops and stays, because he is taking some activity. Furthermore, what we do every day is likely to be repetitive, as a result the places where we take activities are likely to have repeated emergence in the trajectory, which is significant for the prediction. As Fig. 6 showed, we detect many stay points from the raw dataset, and the stay points have a few dense clusters. The clusters are regarded to have some semantic meanings, such as dormitory, restaurant and laboratory. In the next section, the clusters are going to be extracted.

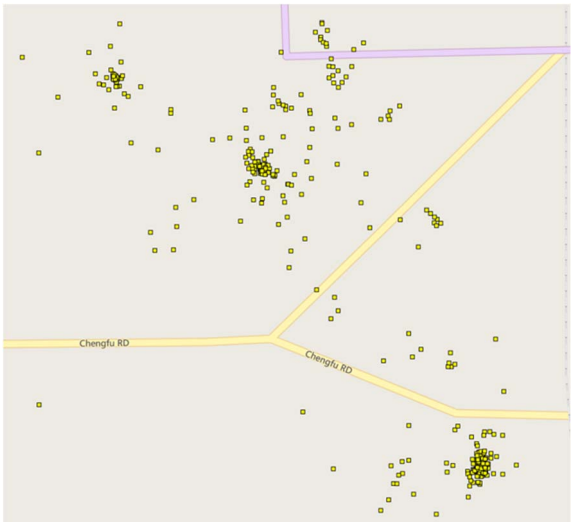

Fig. 6. The visualization of detected stay points

C. Region-of-Interest(ROI) Extraction

In this section, we implement a hierarchical clustering algorithm to calculate the clusters of stay points which are close to each other. As the input of the latter step, the extracted ROIs have a significant impact on the quality of trajectory prediction. Intuitively, a ROI should be as accurate as possible. It means that the place which the users only visit occasionally or rarely has bad influence on the prediction. We make use of local outlier

factor to measure the extent of isolation of a stay point and remove a certain percentage 20% of stay points that are most isolated to achieve a balance between the size of ROIs and the frequency of visits to them.

In this article, we use the minimum rectangular area which covers a cluster of stay points to represent a ROI. Before the implementation of the algorithm, a threshold ΔD is necessary as the minimum distance between each cluster, meanwhile the distance function is needed to measure how far the distance is. We set the threshold ΔD as 150 meters. After implementing the extraction, we get 18 ROIs, the majority of which is showed in Fig. 7.

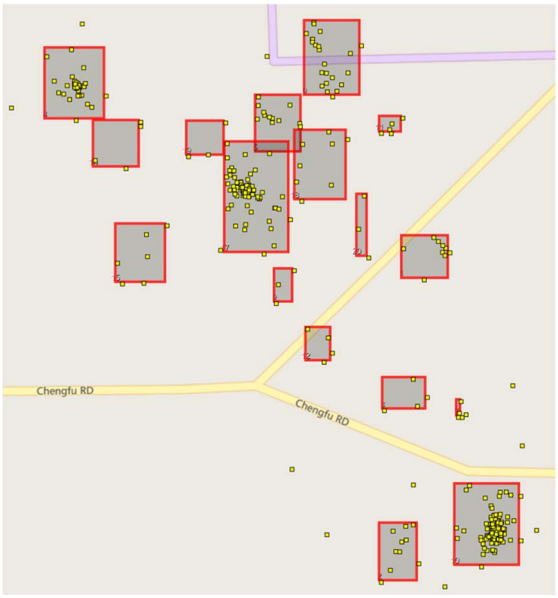

Fig. 7. The visualization of extracted ROIs

D. Mobility Prediction with 1st-order Markov Chains

We select five most frequently visited ROIs denoted by A/B/C/D/E in Fig. 8 so as to demonstrate the results intuitively, and then a Markov model is created for each ROI with transitions to every other ROI. If the user never traveled between two ROIs, the transition probability is set to zero. As showed in Table Ⅱ, if the user arrives in A previously, he leaves for A/B/C/D/E with the probability of 0/0.071429/0/0.714286 /0.214286 which is displayed intuitively in Fig. 8. In other words, the user is most likely go to D afterwards.

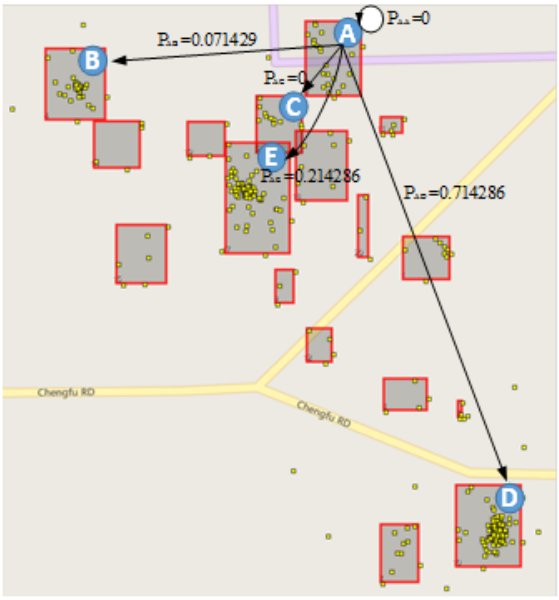

Fig. 8. The five most frequently visited ROIs

TABLE II. TRANSITION PROBABILITY BETWEEN CLUSTERS OF ABCDE

	A	B	C	D	E
A	0	0.071429	0	0.714286	0.214286
B	0	0.315789	0.052632	0.526316	0.105263
C	0.125	0.125	0	0.5	0.25
D	0.125	0.107143	0.035714	0.419643	0.3125
E	0.0408	0.020408	0.020408	0.693878	0.22449

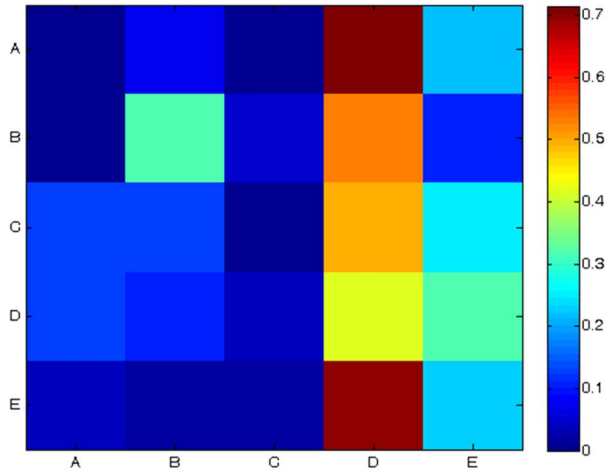

Fig. 9. The visualization of the tansition matrix with 1st-order markov chains

E. Mobility Prediction with Kth-order Markov Chains

In order to predict the next ROI based on the n last positions in the Kth-order Markov chains model, we compute a modified form of the transition matrix whose rows represent the n last visited ROIs. Table Ⅲ shows a 2nd-order Markov chain. The rows of the transition matrix denote all possible combinations of two previous ROIs (AA, AB, AC ,…, EE) while each column represents the next ROI in the 2nd-order Markov chains. For

example, if the previous ROI is A and the current ROI is D, the prediction on the next location may be A/B/D with the probability of 0.111111/0.111111/0.777778 respectively.

Different with 1st Markov chains, in this section we consider the previous 2 ROIs when predicting where the user is likely to go in the future. It is applicative here that the more information we know, the more concrete and precise the prediction will be. Specifically speaking, the probability that the next ROI be E is 0.214286 when the current state is A with 1st Markov Chains as showed in Table Ⅱ, which means that there is a large degree of uncertainty about whether the user will go to E in the future. However, it has great improvement when we take the previous 2 ROIs into consideration. As showed in Table Ⅲ, we're pretty sure that the user will go to E in the future with a probability of 1 when the previous 2 states are E and A respectively.

TABLE III. TRANSITION PROBABILITY WITH 2ND-ORDER MARKOV CHAINS

	A	B	C	D	E
AA	NULL	NULL	NULL	NULL	NULL
AB	0	0	0	1	0
AC	NULL	NULL	NULL	NULL	NULL
AD	0.111111	0.111111	0	0.777778	0
AE	0	0	0	1	0
BA	NULL	NULL	NULL	NULL	NULL
BB	0	0.5	0.25	0.25	0
BC	0	1	0	0	0
BD	0.111111	0.111111	0	0.444444	0.333333
BE	0	1	0	0	0
CA	NULL	NULL	NULL	NULL	NULL
CB	NULL	NULL	NULL	NULL	NULL
CC	NULL	NULL	NULL	NULL	NULL
CD	0	0.333333	0	0.333333	0.333333
CE	0	0	0	0	1
DA	0	0.142857	0	0.714286	0.142857
DB	0	0	0	0.666667	0.333333
DC	0	0	0	1	0
DD	0.171429	0.114286	0	0.228571	0.485714
DE	0	0	0	0.666667	0.333333
EA	0	0	0	0	1
EB	0	1	0	0	0
EC	NULL	NULL	NULL	NULL	NULL
ED	0	0.038462	0.115385	0.576923	0.269231
EE	0	0	0	0.75	0.25

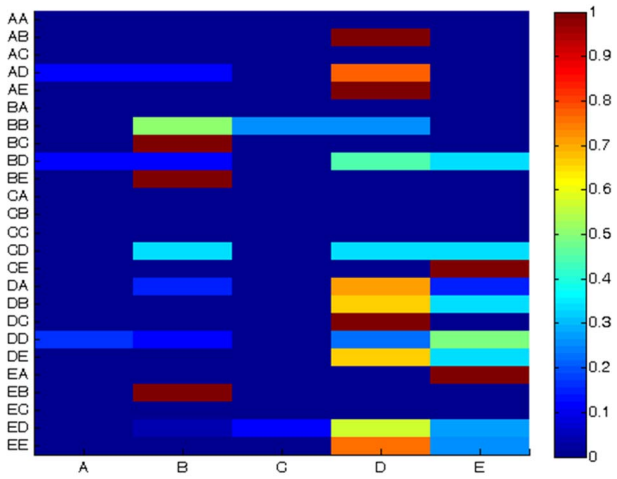

Fig. 10. The visualization of the tansition matrix with 2nd-order markov chain

F. Valuation with Information Entropy

In information theory, information entropy is a mathematical measure of the degree of randomness in a set of data, with greater randomness implying higher entropy and greater predictability implying lower entropy, also called Shannon entropy. Named after Boltzmann's H-theorem, Shannon defined the entropy H of a discrete random variable X with possible values $\{X_1, \ldots, X_n\}$ and probability mass function P(X). The entropy can explicitly be written as:

$$H(X) = -\sum_{i=1}^{n} p_{x_i} \log_2 p_{x_i} \tag{7}$$

In the case of $p(x_i) = 0$ for some i, the value of the corresponding summand $0*\log_2(0)$ is taken to be 0, which is consistent with the limit:

$$\lim_{p \to 0+} p \log(p) = 0 \tag{8}$$

Information entropy is often used as a preliminary test for randomness. Generally speaking, random data will have a high level of information entropy, and a low level of information entropy is a good indicator that the data isn't random. Here in our evaluation, a low level of information entropy stands for that the mobility prediction is more precise and certain. We calculate the average information entropy for Table Ⅱ and Table Ⅲ. As showed in Fig. 11, the column chart is plotted by the order of Markov chains on the horizontal axis and the information entropy on the vertical axis. We can conclude that a higher-order Markov chain reduces the uncertainty of mobility prediction.

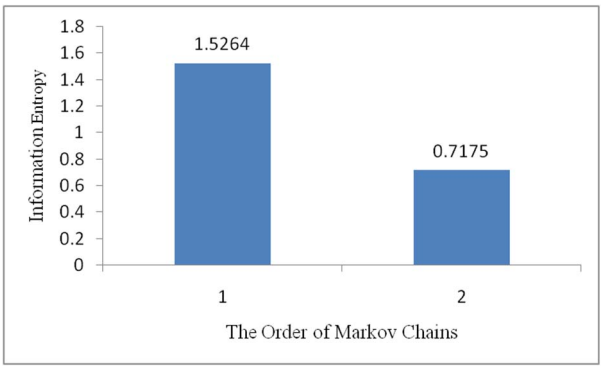

Fig. 11. Valuation with Information Entropy

IV. CONCLUSION

In this paper, we have presented a method consisting of some specific processes to predict human mobility. The extraction of region-of-interest is the premise and foundation, and the Markov chains are the core of prediction. Furthermore, we constructed Kth-order Markov chains which make the mobility prediction more precise. In the future, we plan to investigate in a more systematic method to understand and predict the human mobility by taking the semantic information and the time dimension into consideration in the design of the mobility Markov chains.

V. ACKNOWLEGEMENTS

The research is supported by a joint foundation launched by China Satellite Navigation System Management Office, Shanghai Science and Technology Committee, and Collaborative Innovation Center for Advanced Ship and Deep-Sea Exploration(CISSE) with grant No. BDZX0005 and 13511501302.

REFERENCES

[1] F.Giannotti, M. Nanni, D. Pedreschi, F. Pinelli, and M. Axiak. Trajectory pattern mining. In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM Press, 330–339. 2007.

[2] M. Reaz Uddin, C. V. Ravishankar, and V. J. Tsotras. Finding regions of interest from trajectory data.In Proceedings of the 12th IEEE International Conference on Mobile Data Management. IEEE, 39–48. 2011.

[3] W. Mathew, R. Raposo, B. Martins, Predicting future locations with hidden Markov models, in: Proceedings of the 2012 ACM Conference on Ubiquitous Computing, pp. 911–918, 2012.

[4] R. Simmons, B. Browning, Y. Zhang , V. Sadekar, Learning to predict driver route and destination intent, in: Proceedings of the IEEE Intelligent Transportation Systems Conference 2006, pp. 127–132, 2006.

[5] A. Asahara, A. Sato, K. Maruyama, and K. Seto. Pedestrianmovement prediction based on mixed Markov-chain model. In Proceedings of the 19th International Conference on Advances in Geographic Information Systems, pages 25-33, IL, USA,2011.

[6] M.O. Heo, M. Kang, B.K. Lim, K.B. Hwang, Y.T. Park, B.T. Zhang, Real-time route inference and learning for smartphone users using probabilistic graphical models, J. KIISE: Softw. Appl. 39 (6) (2012) 425–435.

[7] I. Burbey, T.L. Martin, Predicting future locations using prediction-by-partialmatch, in: Proceedings of the First ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments, pp. 1–6, 2008.

[8] L. Liao, D. Fox, H. Kautz, Learning and inferring transportation routines, in: Proceedings of the National Conference on Artificial Intelligence, pp. 348–353, 2004.

[9] A. Monreale, F. Pinelli, R. Trasarti, and F. Giannotti. WhereNext: A location predictor on trajectory pattern mining. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM Press, 637–646. 2009.

[10] I. Rhee, M. Shin, S. Hong, K. Lee, S. J. Kim, and S. Chong. On the levy-walk nature of human mobility. IEEE/ACM Trans. Networking 19, 3, 630–643. 2011.

[11] C. Song, T. Koren, P. Wang, and A.-L. Barabasi. Modelling the scaling properties of human mobility.´Nature Physics 6, 818–823. 2010.

[12] Q. Li, Y. Zheng, X. Xie, Y. Chen, W. Liu, and W.-Y. Ma, "Mining user similarity based on location history," in ACM GIS, pp. 1–10. 2008,

[13] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma, "Mining interesting locations and travel sequences from gps trajectories," in WWW, pp. 791–800. 2009.

[14] X. Cao, G. Cong, and C. S. Jensen, "Mining significant semantic locations from gps trajectory," in VLDB, pp. 1009–1020. 2010.

[15] Yu Zheng, Quannan Li, Yukun Chen, Xing Xie, Wei-Ying Ma. Understanding Mobility Based on GPS Data. In Proceedings of ACM conference on Ubiquitous Computing (UbiComp 2008), Seoul, Korea. ACM Press: 312-321.

[16] Yu Zheng, Xing Xie, Wei-Ying Ma, GeoLife: A Collaborative Social Networking Service among User, location and trajectory.Invited paper, in IEEE Data Engineering Bulletin. 33, 2, 2010, pp. 32-40.