Next Place Prediction using Mobility Markov Chains

Sébastien Gambs
Université de Rennes 1 - INRIA / IRISA
Campus Universitaire de Beaulieu
35042 Rennes, France
sgambs@irisa.fr

Marc-Olivier Killijian[1]
Miguel Núñez del Prado Cortez[1,2]
[1]CNRS ; LAAS 7 avenue du Colonel Roche
F-31400 Toulouse, France
[2]Université de Toulouse, INSA,
F-31077, Toulouse, France
{marco.killijian,mnunezde}@laas.fr

Abstract

In this paper, we address the issue of predicting the next location of an individual based on the observations of his mobility behavior over some period of time and the recent locations that he has visited. This work has several potential applications such as the evaluation of geo-privacy mechanisms, the development of location-based services anticipating the next movement of a user and the design of location-aware proactive resource migration. In a nutshell, we extend a mobility model called Mobility Markov Chain (MMC) in order to incorporate the n previous visited locations and we develop a novel algorithm for next location prediction based on this mobility model that we coined as n-MMC. The evaluation of the efficiency of our algorithm on three different datasets demonstrates an accuracy for the prediction of the next location in the range of 70% to 95% as soon as $n = 2$.

Categories and Subject Descriptors K.4 COMPUTERS AND SOCIETY [*K.4.1 Public Policy Issues*]: Privacy

Keywords Next location prediction, Mobility model, Markov chain, Clustering.

1. Introduction

The collection of the locations visited by individuals through mobile devices equipped with GPS capacities, cell towers or Wi-Fi positioning has attracted a lot of the attention, both from the industry and the research community. In this paper, we address the issue of predicting the next location of an individual based on the observations of his mobility behavior over some period of time and the recent locations that he has visited. More precisely, we extend a mobility model called Mobility Markov Chain (MMC) developed in a previous work [5] in order to incorporate the n previous visited locations. The extended model is coined as a n-MMC and we design a novel algorithm for next location prediction based on this mobility model. This work has several potential applications such as the evaluation of geo-privacy mechanisms, the development of location-based services anticipating the next movement of a user and the design of location-aware proactive resource migration. The evaluation of the efficiency of our algorithm on three different datasets demonstrates an accuracy for the prediction of the next location in the range of 70% to 95% as soon as $n = 2$.

The remainder of this paper is organized as follows. First, we review relevant related work in Section 2 before describing the concept of Mobility Markov Chain (MMC) and its extension the n-MMC model in Section 3. Afterwards, we present the algorithm for next place prediction based on n-MMC in Section 4 before reporting on the experimental results in Section 5 and finally concluding in Section 6.

2. Related Work

In this section, we describe related work on location prediction based on Markov models [2, 3], raw trajectories [7, 8] and semantic trajectories [13]. We also discuss the results of studies comparing location predictors [9, 11] and review the literature on the modeling of human mobility.

Markov model. This type of predictors represents the mobility behavior of an individual as a Markov model and predicts the next location based on the previously visited locations [2, 3, 12]. For instance, Ashbrook and Starner [3] have built a method for predicting future movements that first extract the Points Of Interests (POIs) frequently visited by an individual before building a mobility model. POIs are discovered using a variant of the k-means clustering algorithm on the individual's mobility traces. Finally, a Markov model is computed in which each node is a POI and the transition between two nodes corresponds to the probability of moving from one POI to another. This work is very similar in spirit

MPM'12, April 10, 2012, Bern, Switzerland.

to the Mobility Markov chains (MMC) [5], however the major difference between this previous work and our work lies in the clustering algorithm used to discover the POIs. Indeed, Ashbrook and Starner have used the standard k-means algorithm while we rely on a clustering algorithm tailored for geolocated data called DJ-cluster [16]. This algorithm is adaptive in the sense that the number of POIs extracted depends on the mobility behavior of the individual studied, while in k-means the number of clusters to be discovered has to be fixed in advance. More precisely, DJ-Cluster takes only as input parameters an upper bound on the radius of clusters and the minimal number of mobility traces that should be contained in a cluster. Furthermore, in the context of MMCs, we perform an additional step in which neighboring and overlapping clusters are aggregated thus leading the generation of clusters of potentially different size and shape. Finally, while Ashbrook and Starner proposed a method for learning a model for next place prediction, they did not actually assess its accuracy on a real dataset.

A variant of Markov model called the *Mixed Markov-chain Model* (MMM) [2] has recently been proposed for next place prediction. This approach considers that standard Markov Models (MM) and Hidden Markov Models (HMM) are not generic enough to encompass all types of mobility. Therefore, the concept of MMM was proposed as an intermediate model between individual and generic models. The prediction of the next location is based on a Markov model belonging to a group of individuals with similar mobility behavior. This approach clusters individuals into groups based on their mobility traces and then generates a specific Markov model for each group. The prediction of the next location works by first identifying the group a particular individual belongs to and then inferring the next location based on this group model. This approach was tested experimentally on artificial and real datasets and shows an accuracy for the prediction task (as measured by the ratio between the correct and total number of predictions) of 74,1% for MMM, 16,9%– 45,6% for MM and 2,41%–4,2% for HMM.

Semantic and raw trajectories. Ying *et al.* have proposed to integrate semantic information about the places visited by an individual in addition to its location data in order to enhance the accuracy of the prediction about his future location [13]. The proposed approach relies on the notion of *semantic trajectories*, which represents the mobility of an individual as a sequence of visited places tagged with semantic information. For instance, the semantic tags can be "home", 'favorite restaurant" or "sport center visited on Wednesday and Friday". To support the prediction of next location based on semantic trajectories, the authors have developed a framework called SemanPredict, which is composed of two modules. The *offline mining module* extracts the semantic trajectories from raw data by first computing the stop points of a trajectory [1], which corresponds to locations in which the user has stayed more than a certain amount of time. Afterwards, the offline module queries a Geographic Semantic Information Database (GSID) storing information of landmarks collected via Google Maps in order to attached semantic information to the stop points. The stop points and semantic trajectories are then stored in a tree-like structure and then the semantic trajectories are clustered using an algorithm based on the MSTP similarity [14]. The *online prediction module* is responsible for matching the current trajectory of a user with the closest trajectory in the database by relying on the geographical and semantic features. This module computes a similarity measure combining a geographical and semantic score quantifying the closeness between two trajectories. A partial matching strategy is applied on the tree-like structure in order to identify the closest trajectory. Finally, the prediction of the next location is simply the child node of the candidate trajectory with the highest similarity.

Some other works also rely on the notion of trajectories to predict the next location. For instance, Krumm and Horvitz have developed a tool called *Predestination* [7], which aims at predicting the destination of a trip based on the trajectory information gathered so far. In a nutshell, this method divides the spatial area into a grid in which each cell has a surface of 1 km^2 and then counts the number of times an individual has visited each cell. From this information, *Predestination* computes a probability distribution representing the likelihood of visiting a particular cell. This probability distribution is then used later to predict the most likely destination of a trip. Following the same line of research, Froehlich and Krumm have designed an approach predicting the next location based on the *trip similarity* [4], which displays an accuracy of 85%.

Modeling human mobility. Song *et al.* have made a study comparing four different families of location predictors [11] that have been tested on a dataset gathered by the Dartmouth College from Wi-Fi users between April 2001 to March 2003. From the results obtained, the authors conclude that more complex predictors are not necessarily much more accurate than Markov predictors. They also establish that Markov predictors beyond the second order (*i.e.*, basing their predictions on the $n \geq 3$ previous locations) are less precise. On the theoretical side, Barabasi et *al.* [10] have analyzed the predictability of the human mobility using three different entropy measures. Their approach first constructs a graph in which each node is associated with the percentage of time spent in a cell. Afterwards, the probability distributions of the three proposed entropy measures are computed in order to characterize the predictability of the population. Finally, a predictability score is computed, which represents the accuracy of the prediction of future whereabouts. This predictability score is derived from the Fano's inequality and quantifies the entropy of a particular user moving between n locations. In their experiments, the authors have used a sample of 45 000 users of mobile phones registered during a period of 3 months. This dataset was collected by an American phone company for billing purpose, recording the user in a cell when he uses his phone. From the combination of the empirically measured entropy and the Fano's inequality, the authors conclude the human mobility can be predicted with a probability of success of 93% on average.

3. Extended Mobility Markov Chain

In this section, we briefly review the concept of mobility Markov chains that consider only the current location to predict the next one before extending this concept to take into account the n previous locations visited (for $n \geq 1$). We coin this extended mobility Markov chain as a n-MMC. We also describe an algorithm for learning a n-MMC, which is a variant of the algorithm described in [5], to which some memory has been added to learn the previous n locations visited.

3.1 Mobility Markov Chain

A Mobility Markov Chain (MMC) [5] models the mobility behavior of an individual as a discrete stochastic process in which the probability of moving to a state (*i.e*, POI) depends only on the previous visited state and the probability distribution of the transitions between states. More precisely, a MMC is composed of:

- A set of *states* $P = \{p_1, \ldots p_k\}$, in which each state corresponds to a frequent POI (ranked by decreasing order of importance). These states generally have an intrinsic semantic meaning and therefore semantic labels such as "home" or "work" can often be attached to them. The semantics of some states can sometimes be deduced automatically from the structure of the MMC.
- A set of *transitions*, such as $t_{i,j}$, which represents the probability of moving from state p_i to state p_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. For instance, an individual can leave his "home" to go to the pharmacy before coming back to "home".

A MMC can be represented either as graph (see Figure 1) or a transition matrix. In the graph representation, nodes represent POIs while arrows symbolize the transitions between POIs along with the associated probability of performing this transition. In the matrix representation, the row corresponds to the POI of origin and the column the destination POI. The value stored in the cell is the probability of the associated transition.

Standard MMCs are memoryless in the sense that the prediction of the future location depends only on the current location. However, this limitation in which the MMC "forgets" the previous locations visited before reaching the current state can impact negatively the accuracy of the prediction. To address this issue, we introduce the concept of a n-MMC, which is a MMC in which the states do not correspond only to a single POI, but rather represent the sequence of the n previous visited POIs.

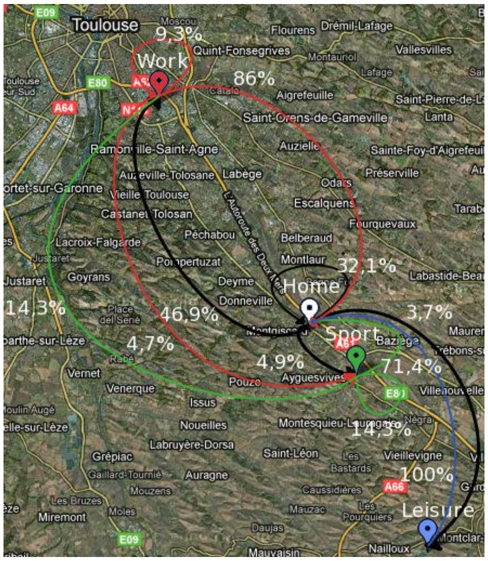

Figure 1. Example of a n-MMC with $n = 1$.

3.2 Learning a n-MMC

Thereafter, we describe an algorithm for learning a n-MMC out of the trail of mobility traces of an individual, which is decomposed in two steps. During the first step, a clustering algorithm called *Density-Joinable cluster* (DJ-Cluster) [16] is used to discover the POIs (Algorithm 1). Afterwards, during the second step, the transitions between those POIs are computed. DJ-Cluster takes as input three parameters: $MinPts$ the minimal number of points necessary to form a cluster, ϵ the maximum radius of the cluster and d_{mer} a merging distance for clusters.

DJ-Cluster is itself decomposed into three phases. During the first phase (*preprocessing*) only the "static" mobility traces (*i.e.*, traces whose $speed \leq \delta$, for δ a small predefined positive constant) are kept by deleting all traces in movement (*i.e.*, with $speed > \delta$), and then consequently redundant traces are also removed. The second phase (*clustering*) consists in processing all remaining traces in order to construct clusters containing at least $MinPts$ points within a radius ϵ of their centers. Once these clusters are computed, during the third and last phase (*merging*), the algorithm merges the clusters sharing at least a common trace, which can lead to the creation of clusters of arbitrary shape. For example, given two clusters $C_1 = \{m_1, m_3, m_7, m_9\}$ and $C_2 = \{m_9, m_{11}, m_{12}\}$, we first verify that their intersection is not null, which is the case here as $C_1 \cap C_2 = \{m_9\}$. These clusters are then merged into a single cluster composed of their union $C_1 \cup C_2 = \{m_1, m_3, m_7, m_9, m_{11}, m_{12}\}$. Finally, the resulting clusters whose centroids are within d_{mer} distance are also merged.

Once the clustering is performed, the *radius*, *time interval* and *density* of each cluster are computed. In a nutshell, the radius is the distance between the center of a cluster and the farthest mobility trace, the time interval is the difference in days between the oldest and the most recent mobility trace and the density is the number of traces in the cluster. Each cluster (*i.e.*, POI) corresponds to a state in the Markov model. Once the POIs are formed, the transitions and their associated probabilities are computed. This process is done by considering the trail of mobility traces in chronological order and labeling each trace with the identifier of the POI to which it belongs (*i.e.*, the mobility trace is located inside the radius of this cluster). If the trace does not belong to any cluster, it is labelled as "unknown". Afterwards during a second pass on the trail of mobility traces, all "unknown" traces are removed and successive traces sharing the same label are squashed in a single occurrence. For instance, a succession of 10 mobility traces sharing the same label will be squashed into a single trace with this label. Finally, from these labeled traces, the transition between states taking into account the n last visited states are computed.

4. Next Place Prediction

In order to predict the next location based on the n last positions in the n-MMC model, we compute a modified form of the transition matrix whose rows represent the n last visited positions. To illustrate the concept of prediction based on a n-MMC, Table 1 and Figure 2 respectively show the transition matrix and graphical representation of a 2-MMC learnt on the trail of mobility traces taken from a user of the Phonetic dataset [6] that we simply name as Bob to preserve his anonymity. This 2-MMC consists of three different states: "Home" (H), "Work" (W) and "Others" (O) and the goal is to predict the next location based on the two previous locations (*i.e.*, $n = 2$). Thus, the rows of the transition matrix denote all possible combinations of pair of previous locations (HH,HW,HO,WH,WW,WO,OH,OW,OO) while a column represents the next position in the n-MMC. For instance, if the previous position was H and the current position is W, the prediction on the next location will be home H and a transition will occur from state HW to state WH, thus updating the previous location to W and the current one to H.

Source/Dest.	H	W	L	O
H W	1,00	0,00	0,00	0,00
H L	1,00	0,00	0,00	0,00
H O	0,64	0,34	0,00	0,00
W H	0,00	0,84	0,08	0,08
L H	0,00	0,50	0,00	0,50
O H	0,00	1,00	0,00	0,00
O W	1,00	0,00	0,00	0,00

Table 1. Transition matrix of Bob.

Algorithm 1 Construction of a n-MMC

```
Require: $D$ a trail of (mobility) traces $D$, $n$ the number of
    previous location kept, $MinPts$ the minimum number of
    traces in a cluster, $\epsilon$ the maximum radius of a cluster and
    $d_{mer}$ the merging distance for clusters.
  Preprocess the trail of mobility traces $D$ by deleting moving and redundant traces thus producing $D'$
  Run a clustering algorithm on $D'$ to discover the most significant clusters
  Merge the clusters that share at least a common point
  Merge the clusters that are within $d_{mer}$ distance of each other
  Let $listPOIs$ be the list of all constructed clusters
  for each cluster $C$ in $listPOIs$ do
    Compute the $time\_interval$, $radius$ and $density$ of $C$
  end for
  Sort the clusters in $listPOIs$ by decreasing order according to their densities
  for each cluster $C_i$ in $listPOIs$ do
    Create the corresponding state $p_i$ in the mobility Markov chain
  end for
  for each mobility trace $m$ in $D'$ do
    if the distance between the trace $m$ and the center of the cluster $C_i$ is less than the $radius_i$ then
      Update the $n - 1$ previous locations (FIFO) and the current position with $C_i$
      Label the trace $m$ with the $n - 1$ previous locations and $C_i$
    else
      Label the trace $m$ with the value "unknown"
    end if
  end for
  Delete all traces that are "unknown"
  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
```

The prediction algorithm (Algorithm 2) requires as input the n previous visited locations and a n-MMC and works in the straightforward following manner. For instance, the input could be a transition matrix such as Table 1 and the two previous locations HO. The algorithm finds the row corresponding to these n previous locations and searches the most probable transition (ties are broken arbitrarily). In our example, as the previous locations are HO, the prediction is H with a probability of 64%.

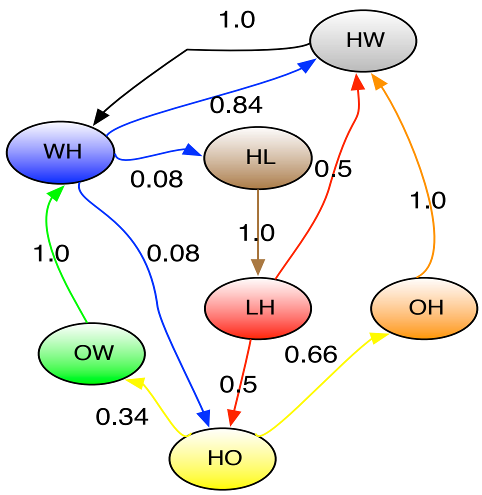

Figure 2. A graphical representation of the 2-MMC from Bob.

```
Algorithm 2 Prediction using a $n$-MMC
Require: Transition matrix $M$ of a $n$-MMC, $n$ previous visited locations
  Search the row $r$ in $M$ corresponding to the $n$ previous visited locations
  Find the column corresponding to the maximum probability of transition $p_{max}$ for the row $r$ (ties are broken arbitrarily)
  return the POI corresponding to the column with $p_{max}$
```

5. Experimental Evaluation

In this section, we report on experiments conducted to evaluate the accuracy of our prediction algorithm and the theoretical predictability of users. In these experiments, we used three different datasets, whose characteristics are summarized in Table 2. The *Phonetic dataset* [6] is composed of the mobility traces from 6 researchers sampled at a rate of 1 to 5 minutes from October 2009 to January 2011. The *Geolife dataset* [15] has been gathered by researchers from Microsoft Asia and consists of mobility traces collected from April 2007 to October 2011 using GPS-enabled devices, mostly in the area of Shanghai. The *synthetic dataset* has been generated out of the first user of the Geolife dataset and we use it mainly as a sanity check to verify the behavior of the prediction algorithm. For instance, the user of this dataset corresponds to the first user of Geolife obtained by duplicating the mobility traces of the first user and then applying a translation in the time domain.

In order to assess the efficiency of our location predictors, we compute two metrics: the *accuracy* and the *predictability*. The accuracy Acc is the ratio between the number of correct predictions $p_{correct}$ over the total number of predictions p_{total}:

$$Acc = p_{correct}/p_{total}. \tag{1}$$

Char. (average)	Phonetic	Geolife	Synthetic
#Users	6	175	1
#Traces per user	16363	126970	694136
Duration of capture	255	146	511
Frequency (#trace/day)	67,53	1263,5	1 358,38
#POI per user	5	8	9

Table 2. Characteristics of datasets used.

The predictability $Pred$ is a theoretical measure representing the degree to which the mobility of an individual is predictable based on his n-MMC (in the same spirit as the work of Barabasi and co-authors [10]). For instance, if the location predictor knows that Bob was previously at work (W) and he is currently at home (H), the probability of making a successful guess is theoretically equal to the maximal outgoing probability transition, which is 84% for this particular example (see Figure 2 and Table 1). More formally, the predictability $Pred$ of a particular n-MMC (and thus a particular individual) is computed as the sum of the product between each element of the stationary vector π of the n-MMC model, which corresponds to the probability of being in a particular state (for l, the total number of states of this n-MMC) and the maximum outgoing probability (P_{max_out}) of the k^{th} state:

$$Pred = \sum_{k=1}^{l}(\pi(k) \times P_{max\_out}(k,*)). \tag{2}$$

In our experiments, we split each trail of mobility traces into two sets of same size: the *training set*, which is used to build the n-MMC, and the *testing set*, which is used to evaluate the accuracy of the predictor. Finally, we also compute the average predictability score for each user based on the n-MMC -learnt from his training dataset. Figure 3 shows the results obtained for a user from the Geolife dataset with n ranging from 1 to 4. As expected, the accuracy first improves as n increases but then seems to stabilize or even decrease slightly as soon as $n > 2$. Moreover, while unsurprisingly the prediction accuracy is usually better on the training set that on the testing set, this difference is not significative. This seems to indicate that the mobility behavior of an individual is similar in the second part of his trail of traces (the testing set) to the first part of the traces (the training set), which may not necessarily be the case if the mobility behavior of a user naturally drift due to an important change in his life. Finally, Figure 4 displays the results obtained for all users of the three different datasets. To summarize, the results consistently show that the accuracy and predictability are optimal (or almost optimal) when $n = 2$, with an accuracy and predictability ranging from 70% to 95%.

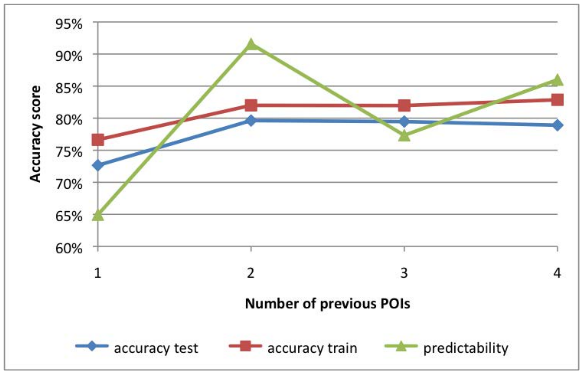

Figure 3. Accuracy and predictability measured for a single user of Geolife.

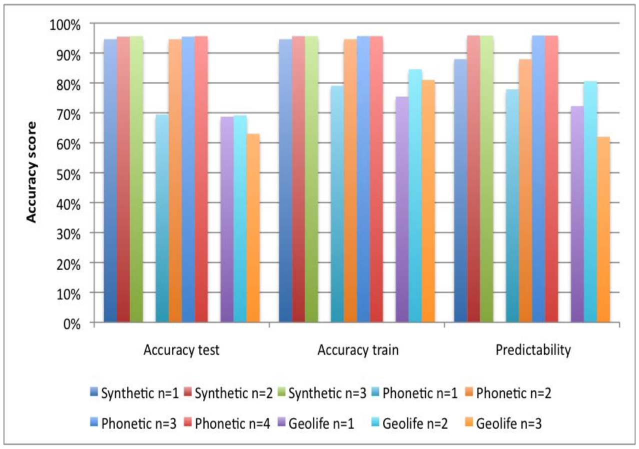

Figure 4. Accuracy and predictability on the three datasets.

6. Conclusion

In this work, we have presented an algorithm for next place prediction based on a mobility model of an individual called a n-MMC that keeps track of the n previous locations visited. Experiments on three different datasets show that the accuracy of this prediction algorithm ranges from 70% to 95%. Moreover, while the accuracy of the prediction grows with n, choosing $n > 2$ does not seem to bring an important improvement at the cost of a significant overhead in terms of computation and space for the learning and storing of the mobility model. In order to further improve the accuracy of the prediction, we are planning as future work to introduce more explicitly the notion of time in the constructed MMC.

References

[1] L. O. Alvares, V. Bogorny, B. Kuijpers, B. Moelans, J. A. Fern, E. D. Macedo, and A. T. Palma. Towards semantic trajectory knowledge discovery. Technical Report, Hasselt University, Limbourg, Belgium, 2007.

[2] A. Asahara, A. Sato, K. Maruyama, and K. Seto. Pedestrian-movement 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.

[3] D. Ashbrook and T. Starner. Learning significant locations and predicting user movement with GPS. In Proceedings of the 6th International Symposium on Wearable Computers, pages 275-286, Sardina, Italy, 2003.

[4] J. Froehlich and J. Krumm. Route prediction from trip observations. In Proceedings of the Society of Automotive Engineers World Congress, MI, USA, 2008.

[5] S. Gambs, M.-O. Killijian, and M. Nuñez del Prado C. Show me how you move and I will tell you who you are. Transactions on Data Privacy, volume 2:103-126, Catalonia, Spain, 2011.

[6] M. Killijian, M. Roy, and G. Tredan. Beyond San Francisco cabs: Building a *-lity mining dataset. In Proceedings of the Workshop on the Analysis of Mobile Phone Networks, pages 75-78, Cambridge, MA, USA, 2010.

[7] J. Krumm and E. Horvitz. Predestination: Inferring destinations from partial trajectories. In Proceedings of the 8th International Conference on Ubiquitous Computing, pages 243-260,CA, USA, 2006.

[8] A. Meyerson and R. Williams. On the complexity of optimal k-anonymity. In Proceedings of the 23rd Symposium on Principles of Database Systems, pages 223-228, NY, USA, 2004.

[9] J. Petzold, F. Bagci, W. Trumler, and T. Ungerer. Comparison of different methods for next location prediction. In Proceedings of the 12th International Euro-Par Conference, pages 909-918, Berlin, Germany, 2006.

[10] C. Song, Z. Qu, N. Blumm, and A.-L. Barabasi. Limits of predictability in human mobility. Science, volume 327:1018-1021, LA, USA, 2010.

[11] L. Song, D. Kotz, R. Jain, and X. He. Evaluating next-cell predictors with extensive wi-fi mobility data. IEEE Transactions on Mobile Computing, volume 5:1633-1649, CA, USA, 2006.

[12] J. S. Vitter and P. Krishnan. Optimal prefetching via data compression. Journal of the ACM, volume 43:771-793, NY, USA, 1996.

[13] J. J.-C. Ying, W.-C. Lee, and T.-C. Weng. Semantic trajectory mining for location prediction. In Proceedings of the 19th International Conference on Advances in Geographic Information Systems, pages 34-43, NY, USA, 2011.

[14] J. J.-C. Ying, E. H.-C. Lu, W.-C. Lee, T.-C. Weng, and V. S. Tseng. Mining user similarity from semantic trajectories. In Proceedings of the 2nd International Workshop on Location Based Social Networks, pages 19-26, NY, USA, 2010.

[15] Y Zheng, Q. Li, Y. Chen, X. Xie, and Wei-Ying Ma. Understanding mobility based on GPS data. In Proceedings of the 10th International Conference on Ubiquitous Computing, pages 312-321, Seoul, Korea, 2008.

[16] C. Zhou, D. Frankowski, P.J. Ludford, S. Shekhar, and L.G. Terveen. Discovering personal gazetters: an interactive clustering approach. In Proceedings of the 12th ACM International Workshop on Geographic Information Systems, pages 266-273, DC, USA, 2004.