Location Prediction With Sparse GPS Data

Ayush Jaiswal[1], Yao-Yi Chiang[2], Craig A. Knoblock[3], Liang Lan[4]

[1]National Institute of Technology Calicut, India
Email: ayush_bcs10@nitc.ac.in

[2]Spatial Sciences Institute, University of Southern California, USA
Email: yaoyic@usc.edu

[3]Department of Computer Science and Information Sciences Institute, University of Southern California, USA
Email: knoblock@isi.edu

[4]Noah's Ark Lab, Huawei Technologies
Email: lan.liang@huawei.com

1. Introduction

Predicting the next location of a user from their movement history is useful in building intelligent applications that can continuously assist users without explicit user-input. Data collected by applications on consumer-grade mobile devices, such as GPS data, can have missing records (e.g., due to the application crashing) and the sensor sampling frequency needs to be kept low so that it does not drain out the mobile battery. Thus, there can be a significant time gap between each pair of recordings. Our work in this paper focuses on predicting the next location of a mobile user using such sparse GPS data, collected at a very low frequency of once in every 10min. To give an example of dense data, Krumm and Horvitz (2005, 2006) use data collected once in every six seconds.

Figure 1: Movement patterns might be disjoint. The blue and the red points were recorded on two different days.

Sparseness in GPS data makes finding patterns in a user's movement history difficult. Moreover, the low sampling rate might capture movement patterns that are along the same path but are disjoint (Figure 1). Losses in GPS connection and imperfect behavior of the data collection application further increase the sparseness of the data. We tackle the problem of sparseness by interpolating user movements using a routing service.

Location prediction can be viewed as a classification problem, in which the possible next locations are discrete classes, but GPS data is continuous in nature. Hence, we use a grid over the region where the GPS data is centered, and map the points to grid-blocks. We discuss the results of using four different Markov models for the prediction task on the sparse and the processed data.

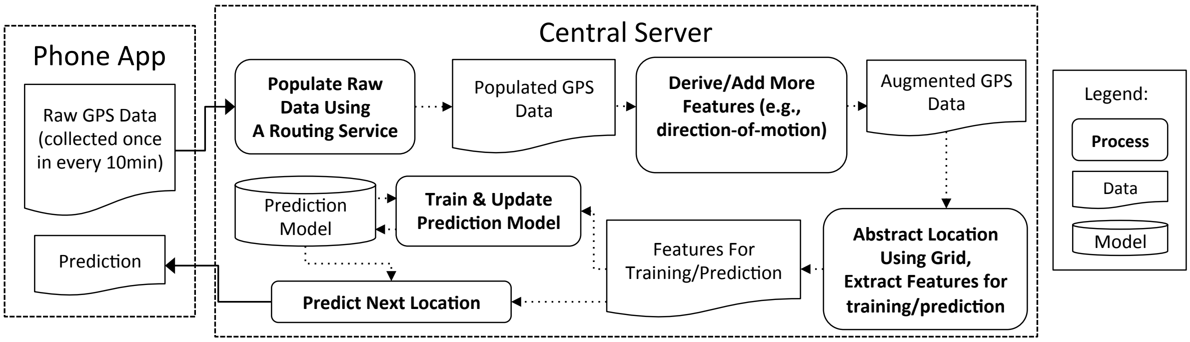

Figure 2: Location Prediction System for Sparse GPS Data.

2. Next Location Prediction

Figure 2 shows the overall workflow of our approach. The sparse GPS data is populated using a routing service to produce a dense set of user movement history, additional features (such as direction-of-motion, described later) are added, and the points are abstracted to locations using a grid. The resulting features are given as inputs to the prediction model.

2.1 Dealing with Sparseness

Our approach uses a routing service to find the shortest path between every consecutive pair of points and uses the route returned to fill up the gap between the pair with dummy points. The underlying assumption is that people tend to take the shortest path between any two places that are near one another, especially when they are separated by just 10 minutes in time. For example, Figure 3 shows how our system populated some of the sparse GPS data that we work on. The dummy points filled up using the routing service complete the original path very elegantly. We use the Google Directions API[1] to get the shortest driving route between every consecutive pair of points that are separated in time by not more than 2 hours (to keep the interpolation reasonable).

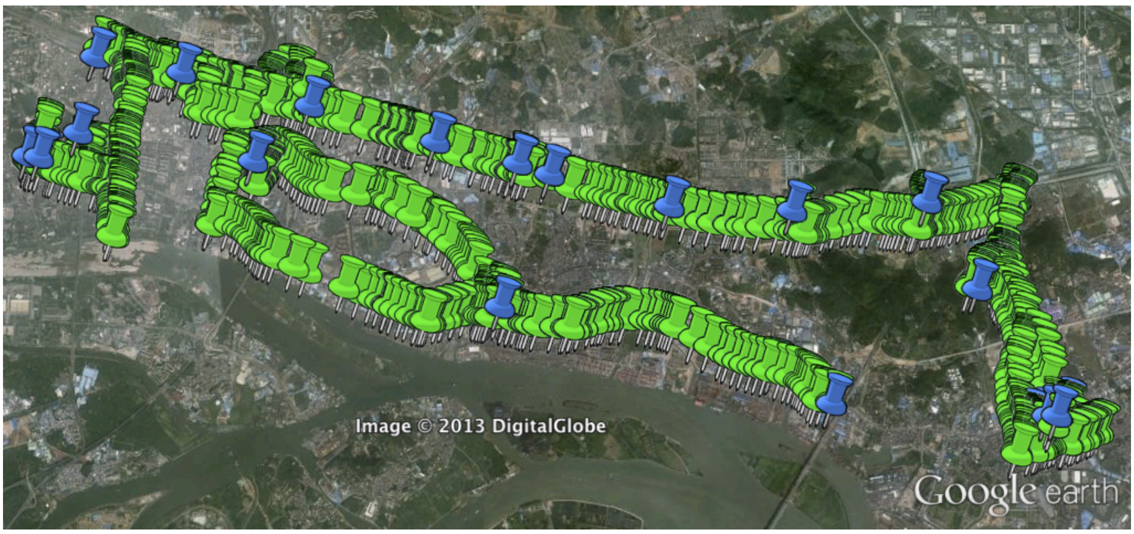

Figure 3: The blue points are original points in the data while the green ones were added using the routing service.

[1] http://developers.google.com/maps/documentation/directions

2.2 Features and Prediction Models

We use Markov models to predict the next grid-block the user will be in, as illustrated in Figure 4. Markov models help in describing sequences of events and their associated probabilities. Cheng et al. (2003) explain how Markov models can be used for location prediction. We employ four different Markov models to test four hypotheses for location prediction from sparse GPS data:

- order-1 Markov model (O1MM): predict the next location of the user based on *their last known location*
- order-2 Markov model (O2MM): predict the next location based on *their two last known locations*
- order-2 Markov model with fallback on order-1 Markov model (FMM): try predicting with O2MM, and when it is unable to make a prediction, use O1MM
- order-1 Markov model with direction-of-motion feature (O1MMD): we use the direction-of-motion between every consecutive pair of points. The directions that we employ are: North, North-East, East, South-East, South, South-West, West, North-West, and *stationary*. This feature removes the need of keeping track of multiple previous locations as it captures the information contained in them.

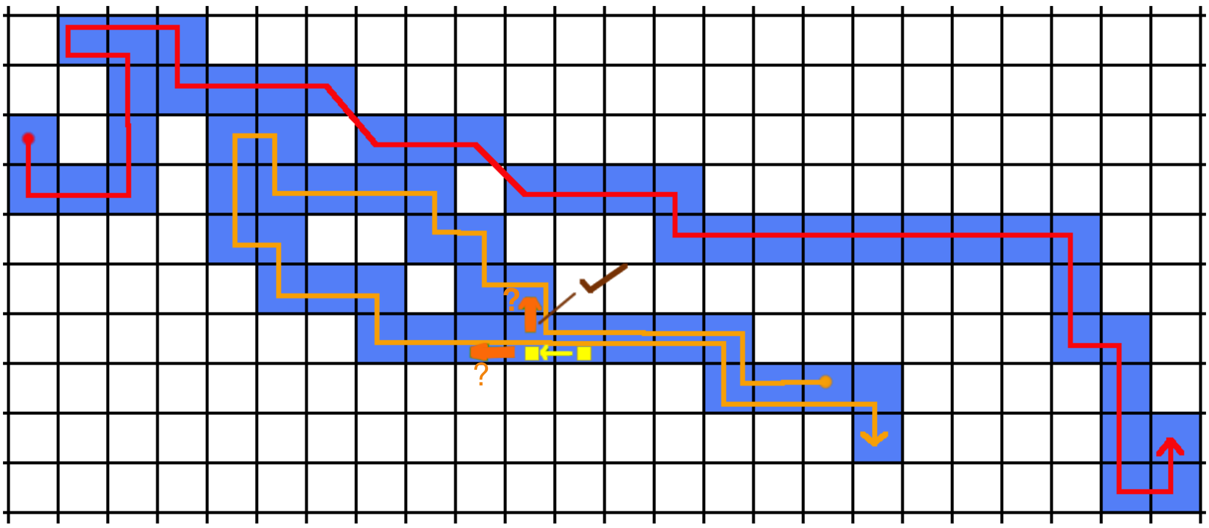

Figure 4: Predicting the next grid-block the user will be in. The model has learnt user movement patterns from day-1 (red line) and day-2 (orange line). On day-3 (yellow squares), it tries to predict the next location of the user.

3. Experiments and Results

Our data were collected by a user in Shenzhen, China over a 24 days period. On an average, it has 14 GPS points in a day. We used the aforementioned Markov models for the task of location prediction on both the original data and the data resulting from the application of our processing steps. We calculated the average prediction accuracies using two experiment settings: leave-one-day-out cross-validation, L1CV, that uses the data from a particular day as test data and data from all other days as training data, and SEQ that uses data from a particular day as test data and data from only the days in the movement history before that day as training data. While cross-validation is a general approach to comparing the accuracies of machine learning models, SEQ is closer to how we would want the prediction to work in real world settings. A correct prediction is one that matches the next observed grid-block of the user. Our accuracy measure is the fraction of predictions that are correct.

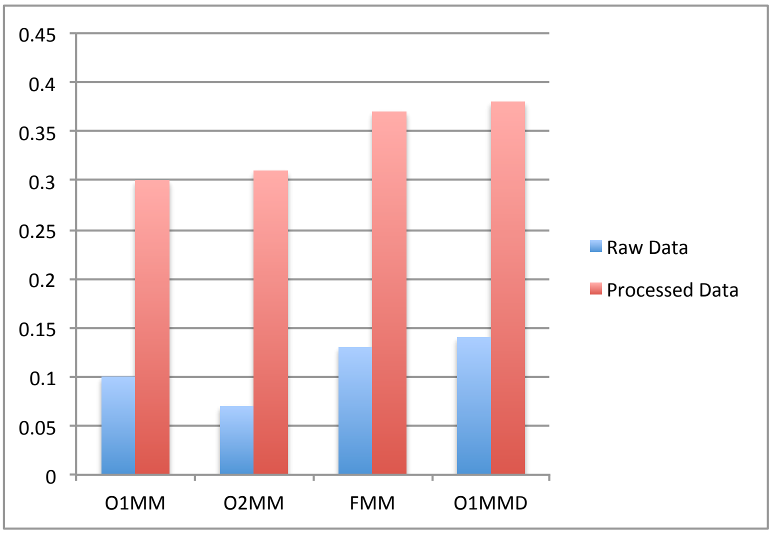

Figure 5: Average SEQ accuracies.

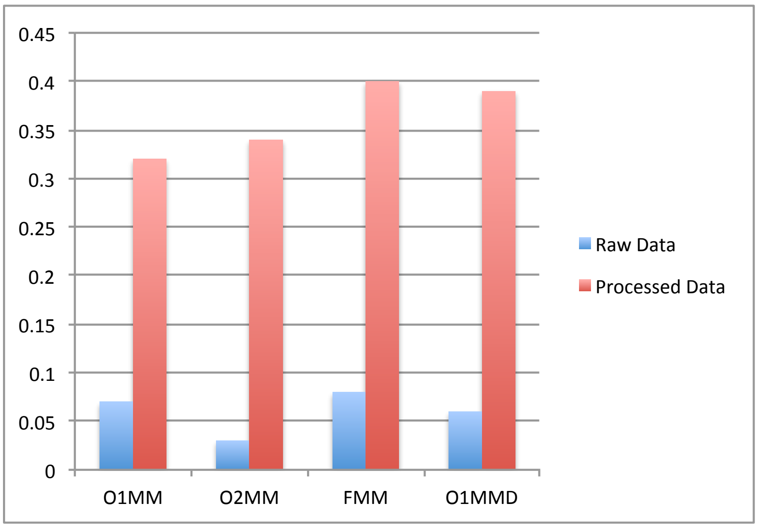

Figure 6: Average L1CV accuracies.

Figures 5 & 6 summarize our results. O1MMD and FMM performed almost equally well and better than the other models on the processed data. The desired order of accuracies should be O1MM ≤ O2MM ≤ FMM as the ones to the right make use of more information about the user's history, but we do not find this order in case of sparse data as O2MM could not learn many patterns because of the sparseness. In general, the prediction models were unable to learn patterns in the user's movements from the sparse GPS data. Solving the problem of sparseness improves their prediction accuracies. The overall accuracies appear low because of significant randomness in the movement patterns of the user whose data we used. It has been found that randomness in a user's movement patterns reduces the accuracy of prediction models (Anagnostopoulos et al. 2009). Such randomness is inevitable in the movements of real users.

4. Related Work

Krumm and Horvitz (2006) use grid-based location abstraction to predict the destination of the user from partial trajectories. Our work is different from theirs as we predict the user's next location, and our data is much more sparse than theirs. While their data is collected once in every 6 seconds, ours is collected once in every 10 minutes. Gao et al. (2012) report that

Hierarchical Pitman-Yor language gives a higher accuracy as compared to Markov models. Anagnostopoulos et al. (2009) work on location prediction using decision trees, k-nearest neighbors, and ensemble learning algorithms, finding ensemble learning algorithms to perform the best among them. The methods proposed in these works cannot be applied directly to sparse data, such as ours, as the machine learning algorithms used in them will not be able to learn patterns effectively. Our processing steps interpolate the sparse data and improve location prediction on such data.

5. Discussion and Future Work

This paper presents an approach for location prediction using sparse user movement history. We showed that by exploiting an online routing service, we made location prediction possible on sparse movement data. We plan to build an intelligent method for automatically generating the dynamic grid size specific to a dataset. We plan to incorporate other sensor data on mobile phones into the location prediction framework.

Acknowledgements

We thank USC Viterbi School of Engineering for providing summer fellowship to Ayush Jaiswal, and Huawei Technologies Co. Ltd. for supporting the project and providing the data.

References

Christine Cheng, Ravi Jain and Eric van den Berg, 2003, Location prediction algorithms for mobile wireless systems. *Wireless internet handbook, 245-263*.

John Krumm and Eric Horvitz, 2006, Predestination: Inferring Destinations from Partial Trajectories. *8th International Conference, UbiComp 2006 Orange County, CA, USA, September 17-21, 2006 Proceedings, 243-260*.

Huiji Gao, Jiliang Tang and Huan Liu, 2012, Mobile location prediction in spatio-temporal context. *Proceedings of the Mobile Data Challenge by Nokia Workshop in conjunction with International Conference on Pervasive Computing, Newcastle, U. K.*

Theodoros Anagnostopoulos et al., 2009, Predicting the Location of Mobile Users: A Machine Learning Approach. *ICPS '09 Proceedings of the 2009 international conference on Pervasive services, 65-72*.

John Krumm and Eric Horvitz, 2005, The Microsoft Multiperson Location Survey. *Microsoft Research (MSR-TR-2005-103): Redmond, WA USA*.