Uncovering Individual’s Mobility Patterns from GPS Dataset

LIN Miao

Thesis Supervisor:  
Assoc. Prof. Hsu Wen-Jing

School of Computer Engineering

A thesis submitted to Nanyang Technological University in partial fulfilment of the requirements for the degree of Doctor of Philosophy

2014
Acknowledgments

Firstly, I would like to thank my supervisor, Prof. Wen-Jing Hsu. During the past four years, his devotions in leading the researches and providing useful guidance have made this thesis possible. I really appreciate the insightful and fruitful discussions with Prof. Hsu, which are the sources for me to make contributions in the area that I studied. His consistent requirements on the rigor of conducting researches and writing papers have greatly challenged me. Besides his research guidance, I have to emphasize that Prof. Hsu has influenced me in every aspect of being a good researcher, such as how to send emails in a polite way, and how to make connections with other researchers, etc. Last but not least, he is a generous and kind person, and has offered many specific helps when I was in Singapore for the first time.

I would like to thank my seniors, Dr. Hui Fang and Dr. Hongyang Sun. I appreciate their helps during the initial stage of my Ph.D. studies. They have always been supportive and kind-hearted in both providing some suggestions in my researches and also offering some helps in my daily life.

I would also like to thank my junior, Zhuo Qi Lee. He is a friendly person for consistently providing some useful suggestions and comments in my researches, and also for borrowing the servers in his lab for me.

The support of my family is the source for me to overcome all the difficulties met both in study and life in Singapore. To all the friends I associated with in China, Singapore, Malaysia and many other nations, namely, Qian Liu, Yulong Li, Jiaying Liu, Xiunian Zhang, Shimin Gong, Yuming Fang, Zhouye Gu, Xing Liu, Chenliang Li, Kerp, etc., I hereby appreciate all the pleasures you bring to me.

Lastly but not least, I would also like to sincerely acknowledge Dr. Yu Zheng and Prof. Larry Rudolph for generously providing their precious GPS datasets, which have made this study possible. Also I appreciate the rigorous comments and suggestions from the examiners and reviewers.
Abstract

Human mobility patterns, including issues such as locations of significance, modes of transport, trajectory patterns, location-based activities, are of great importance to a wide range of research areas and location-related applications. Based on the patterns uncovered, various mobility models may be proposed to predict individual’s future whereabouts, or to evaluate the protocols for wireless communications, among other applications. In this thesis, I present a study of individual’s mobility patterns based on GPS records. The study in this thesis includes inferring the modes of transport, analyzing the predictability of individual’s mobility, constructing mobility model, and predicting future locations.

Modes of transport, such as walking, biking, driving, or taking a bus, are a basic pattern of individual’s mobility. Current studies on inferring the modes of transport apply supervised methods, which include a tedious training process. In this thesis, I present an unsupervised method for inferring the modes of transport, which eliminates the need of manual labeling and training while attaining equal or greater accuracy compared to the best known supervised method. The unsupervised method relies on Kolmogorov-Smirnov Test which offers a theoretical assurance when comparing segments of records.

Various probabilistic models and algorithms, such as Markov models, Bayes models, pattern mining methods, have been proposed to predict individual’s next moves. The predicting accuracy has been greatly improved because of these efforts. However, little is known whether the predicting accuracy is already approaching the limit and hence further research efforts may yield diminishing returns. Moreover, the predicting accuracy is apparently affected by the scale of the places visited and the time interval concerned. In this thesis, I present a study of the predictability of individual’s mobility sequences. The predictability quantifies the potential to foresee the next moves of an individual based on his/her historical records. Using high-resolution GPS data, the scaling effects on predictability are investigated. Given specified spatio-temporal scales, recorded trajectories are encoded into long strings of distinct locations, and several information-theoretic measures of predictability are derived. I show that high predictability is still present at rather high spatial/temporal resolutions. The predictability is found to be independent of the overall mobility area covered, which suggests highly regular mobility behaviors. Moreover, by varying the scales over a wide range, an invariance between the predicting accuracy and spatial resolution is observed which suggests that certain trade-offs between these two are unavoidable.
Most known mobility models are incomplete for describing individual’s mobility because they fail to reproduce essential characteristics of individual’s mobility behaviors. In this thesis, I propose a new Markovian mobility model that fits well the high predictability of individual’s mobility and a few other known statistical properties presented in recent studies. The mobility sequences generated from the new model are verified both theoretically and experimentally. I also present several results that validate popular assumptions about mobility sequences.

The high predictability of individual’s mobility reported in existing literature is based on relatively coarse spatio-temporal resolutions, which is not suitable for the new generation of location-based applications. Also, the high predictability is often dominated by the cases where the individuals stay at a place during most of the time. Therefore, I further investigate methods for predicting individual’s locations at high spatio-temporal resolutions. To achieve this purpose, I present a new method to exploit historical traces that exhibit similarity with the current trace. The similarity of the traces is evaluated based on the notion of edit distance. The predicted location at the future point in time is a weighted mean of the results obtained from modified Brownian Bridge models and linear extrapolation. Compared to using either historical records or linear extrapolation method alone, the proposed location prediction method shows lower mean prediction errors.
# Contents

Acknowledgments ................................................. i
Abstract .......................................................... iii
List of Figures ..................................................... ix
List of Tables ........................................................ xi

1 Introduction ...................................................... 1
  1.1 Background and Motivation .................................. 1
  1.2 Research Issues in Mobility ................................. 3
  1.3 Statement of Results ......................................... 4
  1.4 Synopsis ..................................................... 6

2 Literature Review ............................................... 7
  2.1 Location Inferences and Prediction ......................... 7
  2.2 Modes of Transport .......................................... 12
  2.3 Trajectory Mining ............................................ 13
    2.3.1 Trajectory clustering .................................. 14
    2.3.2 Graph-based trajectory mining ......................... 18
    2.3.3 Trajectory mining as spatio-temporal itemsets .......... 21
  2.4 Location-based Activity Recognition ....................... 25
  2.5 Mobility Models ............................................. 29
  2.6 Chapter Summary ............................................ 32

3 Modes of Transport ............................................. 33
  3.1 Modes of Transport: An Introduction ...................... 33
  3.2 Related Work ............................................... 34
3.3 An Unsupervised Method Based on Kolmogorov-Smirnov Test ................................... 34
  3.3.1 Key ideas of MoDetect ................................................. 35
  3.3.2 Preprocessing: segmentation and noise filtering ................................. 36
  3.3.3 Clustering the high speed mode segments ......................................... 38
  3.3.4 Postprocessing: classifying based on significant locations ..................... 43
3.4 Experimental Results ................................................................. 44
  3.4.1 Parameter setting ........................................................................ 44
  3.4.2 Effectiveness of the preprocessing step .............................................. 44
  3.4.3 Effectiveness of the clustering step .................................................... 45
  3.4.4 Effectiveness of the post-processing step ............................................. 47
  3.4.5 Comparisons with decision tree and EM algorithm ................................. 47
3.5 Discussions ......................................................................................... 49
3.6 Chapter Summary ................................................................................. 50

4 Predictability of Individual’s Mobility ................................................. 51
  4.1 Predictability: An Introduction ............................................................. 51
  4.2 Related Work ...................................................................................... 53
  4.3 Mobility Data ...................................................................................... 54
  4.4 Methodology ....................................................................................... 54
    4.4.1 Grid cells and locations ................................................................ 54
    4.4.2 Entropy measurements .................................................................. 56
    4.4.3 Redundancy and predictability ....................................................... 57
  4.5 Experimental Results ......................................................................... 59
    4.5.1 Estimating entropy from incomplete mobility strings ....................... 59
    4.5.2 Predictability vs spatio-temporal resolutions .................................... 62
    4.5.3 Entropy and predictability .............................................................. 62
    4.5.4 Predictability and redundancy ........................................................ 65
    4.5.5 Impacts of spatial scale on predictability ........................................ 67
    4.5.6 Predictability vs radius of gyration ............................................... 68
    4.5.7 Sensitivity test on the choices of origins ......................................... 69
  4.6 Chapter Summary .............................................................................. 71
5 A Markovian Mobility Model

5.1 Mobility Model: An Introduction ................................. 73
5.2 Related Work ......................................................... 74
5.3 Preliminary ........................................................... 75
  5.3.1 Predictability ..................................................... 75
  5.3.2 The E&P model ................................................... 76
5.4 Predictability of the E&P Model .................................. 77
5.5 A New Markovian Model ............................................. 80
  5.5.1 Properties of the new Markovian model ...................... 80
  5.5.2 Predictability of the new Markovian model ................ 82
5.6 Experimental Results ............................................... 84
  5.6.1 The bound of the predictability ............................ 84
  5.6.2 The statistical properties in the new model ................. 86
  5.6.3 The comparison of predictability in the two models ....... 90
5.7 Chapter Summary .................................................... 91

6 Location Prediction at High Spatio-Temporal Resolution

6.1 Location Prediction: An Introduction ............................ 93
6.2 Related Work ........................................................ 95
  6.2.1 Measuring trace similarities .................................. 95
  6.2.2 Location prediction ............................................. 95
6.3 Edit Distance ........................................................ 96
  6.3.1 The general edit distance ................................... 96
  6.3.2 Three specific edit distances ................................. 97
6.4 Trace Distance ....................................................... 99
6.5 Location Prediction Methods ...................................... 103
  6.5.1 Method A: Linear extrapolation ............................... 103
  6.5.2 Method B: Estimation according to the distance .......... 104
  6.5.3 Method C: Estimation by Brownian Bridge .................. 104
6.6 Experimental Results ............................................... 110
  6.6.1 Mobility data .................................................. 110
# List of Figures

2.1 An overview of one individual’s trace. ........................................... 8
2.2 Three distance components. ...................................................... 14
2.3 A simple example of the Pairwise Curve-Merging algorithm. .......... 19
2.4 Three different tree-structured patterns. ................................... 19
2.5 A sample of simplified trajectory pattern. ................................... 24
2.6 General full conditional model and the factorized conditional model. 25
2.7 Two typical dynamic Bayesian networks. .................................... 27
3.1 (a) An example of the speed distribution from four different modes. (b) The comparison of the speed distribution for the walk mode. .... 36
3.2 The comparison of the original traces and the traces after smoothing. 37
3.3 The speed distributions of the walk mode before and after smoothing. 38
3.4 Comparison of two different speed distribution estimation methods. ... 40
3.5 The plot of precision and recall according to different distance thresholds. 45
3.6 The mean p-value between different modes, and the overall clustering results. 46
4.1 An overview of one individual’s trajectory mapped on a grid map. .... 55
4.2 The distribution of the fraction of unknown locations. ..................... 60
4.3 The plot of the entropy error against the fraction of unknown locations. 61
4.4 Two individuals’ entropy measurements in different spatio-temporal scales. 63
4.5 The distributions of different entropy measures. .......................... 64
4.6 The histogram of the predictability of all the individuals. ............... 65
4.7 The correlation between the redundancy and the predictability. ......... 66
4.8 (a) The plot of the normalized number of locations versus spatial scale. (b) The plot of the predictability versus spatial scale. .............. 67
4.9 (a) The distribution of the radius of gyration. (b) The plot of the predictability versus radius of gyration. 69
4.10 The box plot of the number of distinct locations and the estimated entropy. 70
5.1 The bounds for the predictability in the three cases. 85
5.2 The plot of the visitation frequency of each location against the rank of being explored in the new model. 86
5.3 The distribution of $k(s_i)$. 87
5.4 The comparison of the conditional frequency based on the first two locations. 89
5.5 (a) A comparison of the predictability of the Markovian model and the model with the rule of return akin to the second-order Markov chain. (b) The visitation distribution of the second-order Markovian model. 90
6.1 The boxplot of different edit distance methods between each pair of dataset. 111
6.2 The distribution of the standard deviation estimated in each trace based on the Brownian Bridge model. 112
6.3 The mean prediction errors in one step location prediction based on three different methods. 114
6.4 Two examples of the predicted results. 116
6.5 (a) The distribution of the prediction errors according to linear extrapolation method $e_l$ and Method $C$ $e_b$ in the following 60 seconds, 120 seconds and 180 seconds. 117
A.1 The distribution of the sampling rate of the mobility dataset. 121
A.2 Three individuals’ autocorrelation values in different schemes. 128
A.3 The remaining four individuals’ autocorrelation values in different schemes. 129
A.4 The testing results of Hypothesis One and Hypothesis Two. 132
A.5 The fraction of the predictions that apply the revisiting rules. 134
A.6 The results of average log-loss on Zheng-GeoLife. 136
A.7 Comparisons of the values of $c$ for different values of $\gamma$. 139
List of Tables

1.1 A comparison of different types of data repositories for the study of individual’s mobility. ................................................. 2

2.1 A sample of continuous and discontinuous trajectories. ...................... 21
2.2 A comparison of trace-based mobility models. .............................. 30

3.1 The exact mean p-values between different modes. .......................... 45
3.2 The results of postprocessing step for 11 individuals. ...................... 48
3.3 The overall accuracies in terms of duration and distance. ................. 49
3.4 Matching matrix of the detection results in terms of distance. .......... 49

5.1 Frequently used symbols ............................................................ 75

6.1 The base condition of each edit distance. .................................... 98
6.2 Frequently used symbols ............................................................ 101
6.3 A summary of the one step estimation according to different distance
    threshold $\delta_{thd}$. ................................................................. 114

A.1 The results of average log-loss on Chon-LifeMap. ........................ 135
Chapter 1

Introduction

This chapter introduces the background and motivation of my study on individual’s mobility. Research issues are then identified for focus. My results and contributions on individual’s mobility are also presented. The last part is an outline of this thesis.

1.1 Background and Motivation

Uncovering individual’s mobility patterns is an emergent research area. Owing to the prevalence of mobile phones, a wide variety of mobility-related data are now being captured, including, e.g., the usage of WiFi, and communications between individuals via messages or phone calls. The availability of such data on personal devices will inevitably engender the study of individual’s mobility patterns at an unprecedented scale both in terms of the areas covered by the trajectories and also the number of researchers interested in the study. A few interesting results have been obtained [SQBB10, GHB08, SKWB10, EP06]. For instance, despite great dissimilarities in individual’s mobility areas, which vary from a few square kilometers to more than a few thousands square kilometers, individual’s mobility is found to be highly regular [SQBB10]. Most individuals return to a handful of locations and share a rather universal probability distribution for places visited [GHB08]. After removing individual’s idiosyncrasies, the rules that govern the individuals in exploring new locations or returning to the previously visited locations are found to be similar [SKWB10]. Also, members of a same social group, such as researchers from the same labs, present similar mobility behaviors [EP06].

The study on individual’s mobility patterns have found applications in a wide range of researches, such as personal positioning [FHR09a, FHR09b], complex system [WGHB09, SQBB10], computational sociology [EP06, BDE10], wireless network analysis [RSH+08, LHK+09, HLR10], etc. Both individuals and the society as a whole can benefit from the study of individual’s mobility. For instance, understanding individual’s mobility can help recognize regular activities or abnormal events [PLFK03, LFK07, CMC05, LFK05],...
Chapter 1. Introduction

Table 1.1: A comparison of different types of data repositories for the study of individual’s mobility. Velocity, pause time, inter-contact time (ict for short) are the basic features needed in describing individuals’ mobility. Resolution indicates the degree of spatial accuracy of the data. Scale indicates the area covered by an individual’s data. GSM cell-tower data contain the ict information, whereas both the wireless network traces and GPS data need manual labels to indicate the ict.

<table>
<thead>
<tr>
<th>data type</th>
<th>GSM cell – tower data</th>
<th>wireless network traces</th>
<th>GPS data</th>
</tr>
</thead>
<tbody>
<tr>
<td>resolution</td>
<td>km</td>
<td>m</td>
<td>m</td>
</tr>
<tr>
<td>scale</td>
<td>metropolitan area</td>
<td>campus/workplace</td>
<td>global</td>
</tr>
<tr>
<td>velocity</td>
<td>no</td>
<td>no</td>
<td>yes</td>
</tr>
<tr>
<td>pause time</td>
<td>approximate</td>
<td>approximate</td>
<td>exact</td>
</tr>
<tr>
<td>ict</td>
<td>exact</td>
<td>labels needed</td>
<td>labels needed</td>
</tr>
<tr>
<td>path info</td>
<td>rough</td>
<td>rough</td>
<td>exact</td>
</tr>
</tbody>
</table>

predict future movements or destinations [AS03, FHR09a, GHB08, DGP12], and identify similarities among individuals [LZX+08, ZZXM09]. The study may also be applied in traffic forecast [KCR+09], city planning [MHS95, HO01], mobile virus control [WGHB09], epidemic prevention [CBB+07, Kle07, HBG04], evaluating mobile network protocols [RSH+08, LHK+09, HLR10], logistics management [tL], energy consumption and carbon footprint [Sch11, SJHW09].

Due to the coarse description of individual’s locations provided by the GSM cell-tower data, however, many existing studies of mobility pattern, are incomplete and less than convincing. In GSM cell-tower data, for instance, a location is indicated by the cell tower that provides the connecting services. Hence the resolution for the locations is constrained by the service area of a cell tower, whose size is in the order of hundreds of square meters to several square kilometers. The paths, often indicated by a series of discontinuous sudden jumps, are hardly observable in fine details between the destinations.

Besides the GSM cell-tower data [MIT12], some other data repositories, such as wireless network traces [KH05, MV05, HKA04] and GPS data [Zhe10, RSH+09], have recently become available\(^1\). A comparison among the data repositories in presenting the basic mobility features is summarized in Table 1.1. Specifically, GPS data allow to trace the movements of individuals in fine details. GSM cell-tower data are constrained by the service area of cell towers, whereas wireless network is mostly available within even smaller regions, such as campuses or work places, although their accessibility is being improved gradually. By comparison, GPS devices are able to provide positioning data globally. GPS data, however, suffer from three major limitations: 1) GPS signals are

---

\(^1\) A spatio-temporal datasets generator is given in [GMPR05]. Since this thesis targets on mining of real-world mobility patterns, the issue of the generation of synthetic dataset will not be addressed here. Nevertheless, one of the studies based on synthetic data is included in Section 2.3.1 [NP06].
usually blocked indoor or underground places, 2) GPS devices may get interferences near tall buildings, and 3) continuously collecting GPS data may consume devices’ energy quickly. These limitations have prevented collection of continuous, high quality GPS records over long durations.

Thanks to conscientious efforts by volunteers, however, a few GPS datasets have been made available [Zhe10, RSH+09]. The GPS dataset used in this thesis is a subset from [Zhe10], and the details of this dataset are given in Appendix A.1.

1.2 Research Issues in Mobility

The study of individual’s mobility can be broadly classified into five respects [LH13], namely, location inferences and prediction, inferring the modes of transport, mining trajectory patterns, recognizing location-based activities and constructing mobility models.

In the study of location inferences and prediction, the locations of interest, or the significant locations, are places where an individual visits frequently or dwells for a long time. Examples include, home, work place, friend’s residence, shopping mall, etc. Most researchers identify significant locations through either density-based clustering algorithms [AS03, ZFL+04, CCJ10], or time-based clustering algorithms [CLC10]. Based on the inferred significant locations, various probabilistic models along with efficient algorithms are proposed to predict individual’s next moves with various levels of success. However, none of these results indicate whether the predicting accuracy is already approaching the limit or whether further research efforts will yield diminishing returns.

Inferring the modes of transport [ZLWX08, ZCL+10, ZLC+08, RMB+10, PLFK03, LHL13a], such as walk, bike, bus and car, relies on both the velocity information and some additional features, such as the presence of a car park or bus stop, head change rate, stop rate, etc. Supervised methods, such as decision tree [Qui86], support vector machine [CV95], Bayesian net [Nea03], etc., are the popular methods for inferring the modes of transport. All these existing approaches for inferring the modes of transport use supervised methods, which require laborious manual labeling and training process.

In trajectory mining, two issues are of main focus, namely, trajectory clustering [LH07, LHLG08, LHLC11, NP06, PKK+09] and trajectory pattern mining [FHR09b, FHR09a, LZX+08, ZZXMO9, CMC10, CMC05, GNPP07, CMC07]. The main difference between these two topics lies in their objectives and results. The first topic aims to cluster or classify the trajectories generated by the same group of objects, e.g., animals of the same kind. The second topic aims to present a variety of methods for modeling common mobility behaviors and for predicting the future move. For instance, in [FHR09b, FHR09a, LZX+08, ZZXMO9, CMC10] the graph-based trajectory mining methods are applied, where the trajectories are represented as a list of representative points.
with certain relations. In [CMC05, GNPP07, CMC07], the trajectories are converted into sets of transitions among locations, and methods for mining frequent items are then applied. Their location prediction methods can function only if the set of locations remains fixed.

A few location-based activities can be recognized through GPS data, such as visiting places, changing modes of transport, dining, shopping, etc. Probabilistic models, such as factorized conditional model [DGP12], hierarchical Markov model [LPFK06], conditional random fields [LFK07], Bayesian model [PLFK03], etc., are often applied to recognize these activities using features such as velocity, time-stamped positions, the presence of a bus stop or car park, the distance to a semantically meaningful location, etc. The additional information requires tedious manual labeling, and hence hinders wider applications of normally unlabeled GPS data.

The mobility models can be classified into four categories, namely, random mobility models [SM01, JM96, RMssM01, HGPC99, HLV06, LH03], geographic mobility models [BSH03, JBRAS03], social mobility models [LYD06, GPQ07, MM06], and trace-based mobility models [RSH+08, LHK+09, HLR10, SKWB10]. The mobility models in the first three categories presume the randomness without empirical support, whereas the trace-based mobility models emphasize on the statistical patterns of the individuals learned from trajectories. Moreover, the first few trace-based mobility models [RSH+08, LHK+09, HLR10, SKWB10] are only applicable to simple cases since the mobility data they used have only one mode of transport and are within a small mobility area. The mobility model in [SKWB10] does not consider the fundamental high predictability of individual’s mobility [SQBB10, LHL12].

1.3 Statement of Results

In this thesis, I present my studies of individual’s mobility on the following topics, namely, inferring the modes of transport, analyzing the predictability of individual’s mobility, constructing mobility model, and predicting individual’s next locations.

The first topic of my study is inferring the modes of transport, such as walking, driving, or taking a bus. Recall that current studies in inferring the modes of transport use supervised methods, which require a training process. I propose an unsupervised method for inferring the modes of transport. This unsupervised scheme eliminates the need of manual labeling and training. I use Kolmogorov-Smirnov Test to offer a theoretical assurance when computing similarity between segments of records. My analysis shows that the higher speed modes can be better differentiated through a weighted bootstrap procedure. I also augment the decisions with reference to the transfer probabilities between different modes at locations identified from the GPS records. This leads to a
new unsupervised method that attains equal or greater accuracy compared to the best known supervised method.

The second topic of my study is predictability of individual’s mobility. Predictability is the information-theoretic upper bound that fundamentally limits any mobility prediction algorithm in predicting the next locations based on historic records. Song et al. [SQBB10] study the predictability issue based on the analysis of mobile phone data over a large population. Disregarding the apparent differences in individual’s daily routines, their analysis revealed a 93% predictability in individual’s mobility. However, the mobile phone data exhibit low spatio-temporal resolution. Also, the locations are generally of heterogeneous shape and differ vastly in size. Therefore, it is unclear whether the predictability findings remain valid with high-resolution data.

I present a study of the predictability using high-resolution GPS data. For a given specified spatio-temporal scale, recorded trajectories are encoded into long strings of distinct locations, and several information-theoretic measures of predictability are derived. I show that high predictability is still present at rather high spatial/temporal resolutions. Also, the predictability is independent of the overall mobility area covered. This suggests highly regular mobility behaviors. Moreover, by varying the scales over a wide range, an invariance is observed which suggests that certain trade-offs between the predicting accuracy and spatio-temporal resolution are unavoidable.

The third topic of my study is constructing mobility model. I present a new mobility model that fits well the high predictability and the scaling statistical properties reported in [SKWB10]. The predictability of individual’s mobility sequences generated from the well-known model [SKWB10] is studied both theoretically and experimentally. I show that there is an incorrigible gap between the predictability of the mobility model and the properties uncovered in my study and [SQBB10]. I then present an improved Markovian model which exhibits high predictability while preserving the desirable scaling properties of the original model in [SKWB10].

The fourth topic of my study is predicting individual’s next locations at high spatio-temporal resolution. Despite the high predictability of individual’s mobility sequences, the predicted results are still too coarse in terms of both spatial and temporal resolutions for recent location-based services [GG03]. Specifically, the temporal resolution in the earlier context is one hour and the spatial resolution is around 500 meters [LHL12]. Moreover, the high predictability over the entire duration may be dominated by the cases where the individuals stay at a place most of the time. When making moves, it may be difficult to predict individual’s next location at high spatio-temporal resolutions.

To remedy these deficiencies, I present a new approach to predict a person’s future locations at high spatio-temporal resolution. The proposed location prediction method uses the individual’s historical records to find traces that are similar to the current trace. The similarity of the traces is evaluated based on the notion of edit distance.
Chapter 1. Introduction

The predicted location at the future time is a weighted result obtained from a modified Brownian Bridge model that incorporates linear extrapolation. Compared to using either historical records or linear extrapolation alone, the proposed location prediction method shows lower mean prediction errors.

1.4 Synopsis

The remainder of this thesis is organized as follows. Chapter 2 reviews the existing work of a few research topics in individual’s mobility, including their findings and deficiencies. My studies are presented from Chapter 3 to Chapter 6, namely, inferring modes of transport, analyzing the predictability of individual’s mobility, constructing mobility models, and predicting individual’s next move. A brief summary of this thesis along with a few issues worthy of further study are given in Chapter 7.
Chapter 2

Literature Review

In this chapter, I review several topics that are most relevant to my research in individual’s mobility patterns. The contents of this chapter are published in [LH13]. These topics include location inferences and prediction, modes of transport, trajectory patterns, location-based activities and mobility models.

2.1 Location Inferences and Prediction

In the context of mobility studies, significant locations are the places where an individual or a group visits frequently or dwells for a long time. These locations may be different from locations that have geographic or social meanings, such as the parks, coffee shops, etc. Figure 2.1 highlights a few significant locations learned from one individual’s trajectory. Some of the locations may be more meaningful, such as home or office buildings, while others may be merely the intersection points on the road, where the individuals happen to stay for a while. Another example may be given by an individual who regularly parks his car in a specific parking lot. Thus a large number of samples may be collected around this area.

In a typical GPS dataset, a trace is a time-stamped sequence of pairs of latitude and longitude. Because of measurement errors, GPS coordinates of a same location may vary from time to time. Inferring the significant locations from GPS traces is the process of mapping the raw readings to discrete location symbols. In [AAH+11], Andrienko et al. summarize four steps for extracting the significant places from mobility data. Firstly, a subset of points indicating certain events, e.g., traffic jam, is extracted from raw data. Secondly, meaningful locations are generated by clustering the points based on the spatiotemporal attributes. The last two steps analyze the aggregated trajectories that fall with the clusters. In the following, I discuss a few recent papers according to the spatiotemporal clustering methods they used, e.g., the density-based clustering algorithm or its variants [AS03, ZFL+04, CCJ10], or time-based clustering algorithms [CLC10].
Chapter 2. Literature Review

Figure 2.1: (a) One individual’s GPS traces mapped on Google Earth. (b) The red stars are the detected significant locations for this individual. These two figures are from [FHR09b].

Ashbrook et al. [AS03] propose a two-step method to infer the significant locations, and further construct a Markov model to predict future locations. In the first step, the significant places are inferred by the points where the GPS devices lose the signals from the satellites. These places are clustered into locations using a variant of the K-Means algorithm in the second step. In the clustering procedure, the clusters are initially centered at $K$ chosen points with a given radius, and iteratively move to a denser area, until no further changes arise in the clusters. When given a large radius, the significant locations may correspond to the city level, or the campus-level when given a small radius. A second-order Markov model is constructed based on the transition probabilities between locations. According to the high relative frequency of movement patterns between locations, the future location is predicted. Since the loss of GPS signals serves as the main clue to identify significant locations, mostly buildings are found. Other types of significant locations where the signals are continuously collected, such as certain open-air canteens, may hardly be detected.

Zhou et al. [ZFL+04] propose a density- and join-based clustering algorithm called DJ-Cluster to infer significant locations. The dense points are those with at least certain number of other points lying within a distance of their neighborhood. The clusters are formed from a set of dense points, which are density-joinable when the neighborhood of the dense points share a common point. A two-step preprocessing procedure is introduced to improve the performance of the algorithm, which eliminates a GPS reading if either its speed is greater than zero or if it is within a small distance of the preceding reading. The experimental results indicate great improvements in terms of both recall and precision over those obtained from the K-Means algorithm. Moreover, DJ-Cluster
overcomes the drawbacks of DBSCAN\(^1\) [EpKSX96] by reducing the memory and time requirements. However, some useful information in the original data may be lost during the preprocessing step. For instance, when a stay is incurred at a location and a large amount of readings arising from the adjacent locations are generated, then removing the series of reading within a few meters may conceal the fact that a stay is incurred.

Cao et al. [CCJ10] propose a semantics-enhanced clustering algorithm (SEM-CLS) to extract semantically meaningful locations and further rank the locations through a unified probabilistic model. SEM-CLS splits the locations of different semantics in a cluster and merges the locations sharing the same semantics from adjacent clusters, which enhanced the results from OPTICS\(^2\) [ABKS99] and K-Means. A two-layered graph is constructed to model the trajectory, and each layer represents the relations between the locations and the individuals as well as the relations among the locations. When ranking the significance of the locations, a unified probabilistic model, based on random walks with the Markov Chains built from each layer, is proposed along with an extended model, which is able to incorporate stay durations and distances between locations. Locations of various semantics can be extracted more accurately than those by using a single factor, such as rank-by-visit, rank-by-duration, Hyperlink-Induced Topic Search (HITS) [Kle99], etc. Since the model is constructed based on the trajectory of a group of individuals, the detected locations are semantically meaningful for a group and individual’s bias is removed.

Hariharan et al. [HT04] construct a probabilistic model to condense, understand and predict individual’s location based on a GPS log. A stay is a single instance for an individual to have spent sufficiently long duration in one place. A destination is generated from merging the stays according to the geographic scale of interest. Based on the time-dependent probability involved with the destinations, such as the probability of an individual starting from a destination at a given time interval, a first-order Hidden Markov Model is constructed to represent the probability for a transition between two destinations at the given time interval. After estimating the parameters through either a Markovian or non-Markovian solution, the relative likelihood of a new location is estimated from this model. Based on the data of two individuals over one year, the experimental results show that a non-Markovian approach is suitable for identifying typical activity patterns, while the Markovian approach is better in revealing atypical behaviors. However, whether the first-order Markov model is able to capture all of the individual’s activity patterns has not been validated.

\(^1\)In DBSCAN, it clusters the points that have a high density within a given area.

\(^2\)It adopts the theory in DBSCAN, but the difference is that in OPTICS the clusters of points are not generated, whereas all the points are sorted according to the defined radius, within which a high density is achieved.
Krumm et al. [KH06] propose a closed-world model and an open-world model to predict individual’s future locations. A closed-world model merely considers the probability of where a driver has been to, and the destination refers to a rectangular region of the map. The open-world model enriches the closed-world model by introducing 1) the probability of a certain type of area being a destination, 2) the probability of each destination according to the GPS data, and 3) the driving efficiency and trip time. Both closed-world model and open-world model are trained based on the historical data, but a complete data model, sharing the theory with the open-world model, is trained by the data both before and after the test day. The experimental results show that in both the complete data model and the open-world model, the prediction errors decrease when the trip fraction increases, whereas the prediction errors merely fluctuate a little bit in the simple closed-world model. When associating the locations to a grid map, some large places may be separated while some irrelevant places are included, therefore the probability of each small grid of being a destination may not be directly relevant.

Gao et al. [GTL12b] use the spatio-temporal information in location prediction. They formalize the location prediction problem as finding the probability distribution \( p(v_i = l | t_i = t, v_{i-1} = l_k) \), where \( l \) and \( l_k \) denote specific locations and \( t \) denotes the prediction time. Based on the Bayes’ rule, Gao et al. show that

\[
p(v_i = l | t_i = t, v_{i-1} = l_k) \propto p(t_i = t | v_i = l) p(v_i = l | v_{i-1} = l_k)
\] (2.1)

In Eq. (2.1) \( p(v_i = l | v_{i-1} = l_k) \) denotes the probability of observing the location \( l \) at the next time step given the current location \( l_k \). By smoothing the trajectories of different length, Gao et al. apply the Hierarchical Pitman-Yor (HPY) language model [Teh06] to estimate the probability distribution \( p(v_i = l | v_{i-1} = l) \), which yields better performance than estimating by using either the most frequent location or the order-K Markov model [GTL12a]. The associations of the two sources of temporal information, namely, day of the week and time of the day (hours), with locations are considered in estimating the probability \( p(t_i = t | v_i = l) \) in Eq. (2.1). Since the training data do not provide the complete information of \( p(t_i = t | v_i = l) \), Gaussian distribution is assumed in order to estimate the probability \( p(t_i = t | v_i = l) \). Using the mobile data set from Nokia Mobile Data Challenge, Gao et al. show that the model combining the spatio-temporal information presents higher prediction accuracy than the model using only either spatial or temporal information alone. However, this study is based on two assumptions. Firstly, the temporal information, namely, day of the week and time of the day (hours), independently affects individuals’ mobility. Secondly, the probability \( p(t_i = t | v_i = l) \) is a Gaussian distribution. More empirical study should be conducted to validate these two assumptions.

Wang et al. [WP12] consider the problem of predicting the next location in two different ways, namely, applying the periodicity of mobility in location prediction and
considering the location prediction as a multi-class classification problem. To make use of the periodicity of mobility in prediction, Wang et al. overlay the trace of each period according to the temporal alignment. The trace in each period is of one week length with granularity of one hour. The location \( p_{\text{next}} \) at future time \( t_{\text{test}} \) given the current location \( p_{\text{cur}} \) is the one that maximizes the dot product \( \text{dist}(t_{\text{test}}) \cdot \text{dist}(p_{\text{cur}} \rightarrow p_{\text{next}}) \), where \( \text{dist}(x) \) is the distribution of \( x \). Both \( \text{dist}(t_{\text{test}}) \) and \( \text{dist}(p_{\text{cur}} \rightarrow p_{\text{next}}) \) are learned from the traces of each period. For the locations that are not frequently visited, they apply the 1st-Order Markov model. By formulating the location prediction as a multi-class classification problem, Wang et al. extract features from each trace. The features they considered are spatial features, such as the current location and the sequence structure, and temporal features, such as the starting time of the trace, the day of week, the hour of day, etc. The experimental results show that by using LIBSVM [CL11] the classification models present high prediction accuracy than the periodicity methods. When applying the classification methods in location prediction, one important issue is the size of the training samples. One possible solution is to generate more training samples by dividing the traces into subtraces, and the prediction accuracy may be increased.

Etter et al. [EKK12] develop several predictors based on Dynamical Bayesian Network (DBN), Artificial Neural Networks (ANN), and Gradient Boosted Decision Tree (GBDT) [HTF09], for location prediction. In DBN, the distribution of the next location depends on the previous location and the temporal information, and it is given as

\[
\pi p(X(n+1)|X(n)) + (1 - \pi)p(X(n+1)|H_e(n), W_e(n), I_{\text{trust}}(n))
\] (2.2)

where \( \pi \) is the parameter that weights the two distributions. \( p(X(n+1)|X(n)) \) is the location-dependent distribution, i.e., the first-order Markov Chain, where \( X(n) \) indicates the location at time \( n \). \( p(X(n+1)|H_e(n), W_e(n), I_{\text{trust}}(n)) \) is the time-dependent distribution, and \( H_e(n), W_e(n), I_{\text{trust}}(n) \) are the discretized ending hour, the day of week, and the belief of the transition from step \( n \) to step \( n + 1 \), respectively.

In DBN, the parameters are learned in two manners: 1) empirically according to the frequency in the data then \( \pi \) is chosen to minimize the prediction errors in the training data; 2) first introducing a latent variable, then the parameters are learned by using Expectation-Maximization algorithm [DLR77].

A 2-layer ANN is applied to predict the next location according to a few features, namely, the categorical data representing each location, and the temporal information used in DBN. The soft-max transfer function is applied to estimate the distribution of the predicted location, and the optimal parameters are obtained by minimizing the negative log-likelihood function. The GBDT is applied to combine a few weak classifiers to obtain a classifier that greatly improves the performance of these weak classifiers.

\[ ^3 \text{The method of detecting the period} \text{ is given in [LWH12].} \]
Experimental results show that the three methods, namely, DBN, ANN, and GBDT, greatly improve the prediction accuracy from around 40% with the baseline methods to about 60%. In DBN, the authors mention that the sequence is non-stationary and it may affect the estimation of the distribution. Therefore, an aging parameter is introduced to weight the temporally-distant samples. However, the authors did not report whether the performance of the DBN has improved by introducing this parameter.

I have reviewed the approaches for the inference of significant locations and various probabilistic models along with efficient algorithms proposed to predict individual’s next moves with various levels of success. Yet little has been discussed whether the predicting accuracy is already close to the limit. In Chapter 4, some theoretical results [SQBB10, JLJ+10] on the predictability of individual’s mobility sequences along with my study on this issue are both presented.

Significant locations can forward the study of individual’s mobility on the remaining issues. In particular, typical locations, such as bus stop and parking lot, are the indications of changing modes of transport as well as activities. The locations are useful in representing the path patterns graphically.

2.2 Modes of Transport

Mode of transport is another fundamental property of individual’s mobility. Individuals may apply a combination of modes of transport, such as walk, bike, bus and car, etc., in the daily activities. Basically, each mode of transport may present different speed values, but recognizing a mode is not easy, especially when multiple modes are utilized during a trip or when traffic congestion is involved. Besides speed values, some additional features, such as the presence of a car park or bus stop, heading change rate, stop rate, etc., are found to be effective in detecting modes of transport based on supervised methods [ZLWX08, ZCL+10, ZLC+08, RMB+10], such as decision tree, support vector machine, Markov Models, etc. Various supervised methods are presented by Zheng et al. [ZLWX08, ZCL+10, ZLC+08] and Reddy et al. [RMB+10]. Lin et al. [LHL13a] demonstrate that the high speed distributions are apparently differentiable among different modes, thus an unsupervised method is proposed.

Among all the approaches [ZLWX08, ZCL+10, ZLC+08], the preprocessing step is similar, which divides the GPS log into segments based on criteria such as changing points, uniformity in duration and distance. The changing points are the locations where changes of transportation modes from walking to non-walking modes or vice versa are likely to have occurred. The differences between these approaches lie in the various features selected and the post-processing step applied. Basic features such as the distance covered, mean velocity, etc., are used in [ZLWX08], while three additional features,
namely, heading change rate, stop rate and velocity change rate, are further introduced in \[ZLC^+08, ZCL^+10\]. Several instance-based classifiers are developed to infer the transportation modes based on the corresponding features through standard training and predicting steps. In the post-processing step, the result is adjusted according to the transition probability between modes [ZLWX08], and the probability between locations of each mode is used in \[ZLC^+08, ZCL^+10\]. Experimental results indicate that changing points based segmentation method combined with a Decision Tree algorithm shows the highest inference accuracy. Since manually labeled data are used in both the training phase and the post-processing phase, these methods suffer from laborious labeling as well as tedious training and predicting process.

Reddy et al. \[RMB^+10\] detect the modes of transport based on the information from various sensors. Features such as the mean, variance, energy, and the DFT energy coefficients between 1-10 Hz, are calculated based on the data obtained from accelerometer, GPS, WiFi, and GSM. In addition to typical instance classifiers, the authors evaluated continuous Hidden Markov Model and Decision Tree combined with a Discrete Hidden Markov Model (DHMM). Among all these classifiers, Decision Tree combined with DHMM achieves the highest accuracy (about 93.6%) to date for detecting the modes of transport.

Different from the supervised methods mentioned before, I propose an unsupervised method to infer the modes of transport from unlabeled GPS logs \[LHL13a\], which is further discussed in Chapter 3.

### 2.3 Trajectory Mining

Trajectory mining is the most popular topic in the study of individual’s mobility patterns. A host of studies has been conducted on various mobility datasets, e.g., individual human mobility data \[FHR09b, FHR09a, LZX^+08, ZZZM09, CLC10\], animal movement data \[LH07, LHLG08, LDH^+10, LDHK10\], hurricane data \[LH07, LHLG08\], vehicle movement data \[PKK^+09, LHLCl11, CMC05, GNPP07, CMC07\], and synthetic mobility data \[NP06\], etc. I group these studies of trajectory mining into two sub-topics, namely, 1) trajectory clustering, and 2) trajectory patterns extraction. The main difference between the two topics lies in their objectives and results. The first topic aims to cluster or classify the trajectories generated by the same group of objects, e.g., animals of the same kind. The second topic aims to present a variety of methods for modeling common mobility behaviors and for predicting the future move. In the following, the studies in trajectory patterns extraction are discussed according to two subtopics, 1) graph-based trajectory mining and 2) trajectory mining as spatio-temporal itemset.
2.3.1 Trajectory clustering

In this context, a cluster is usually a group of trajectories that share common sub-traces or are relatively close to each other. For extraction of sub-trajectories, some of the studies are conducted with the assistance of visualization [RPN+08, AAR+09]. To measure the proximity or closeness of the traces, a distance measure between trajectories is usually defined. Some variants of the density-based clustering methods are then introduced in order to cluster the trajectories [LH07, LHLG08, LHLC11, NP06, PKK+09]. Specifically, both [LH07, LHLG08] apply the idea of DBSCAN, and [NP06] uses OPTICS, while [PKK+09] is based on fuzzy C-means, and [LHLC11] applies a sequential pattern mining method.

![Figure 2.2: The three components, namely, perpendicular distance $d_\perp$, parallel distance $d_\parallel$ and angle distance $d_\theta$, in measuring the distance between trajectories in [LH07, LHLG08].](image)

In DBSCAN, the core points that have at least $MinPts$ points within its $\epsilon$ distance form the center of the clusters. Similarly, in [LH07, LHLG08], the distance measurement between two line segments is defined as the mean of perpendicular distance $d_\perp$, parallel distance $d_\parallel$ and angle distance $d_\theta$, as shown in Figure 2.2. Rather than clustering the whole trajectories, Lee et al. [LH07] consider it to be more meaningful to cluster the common sub-trajectories. Therefore, by using the principle of minimum description length [GMP05], each trajectory is partitioned into a list of line segments, simulating the rapid changes of the directions. By making an analogy to the idea of DBSCAN, the clusters are the group of lines where each line is within $\epsilon$ distance to the core line, which has sufficient number of lines $MinLns$ with that distance. The representative trajectory of a cluster is a sequence of line segments that show the general trend of the trajectories within the cluster. Further, motivated by the observations that discriminative features mostly appear in sub-trajectories and regions, Lee et al. [LHLG08] classify the trajectory based on the features extracted from region-based clusters and trajectory-based clusters. The target regions are of variable rectangular size and they contain the sub-trajectories.
of a cluster. These regions are extracted by iteratively partitioning the space following the principle of minimum description length, which is similar to [LH07] but with a different objective function. The trajectory-based clustering aims to find the homogeneous $\varepsilon$-neighborhood, indicating the trajectories within $\varepsilon$ distance form the same clusters. The clusters of homogeneous $\varepsilon$-neighborhood is detected by the trajectory-version DBSCAN [LH07], and for each cluster the representative trajectory is similarly generated. The classification of the trajectories is based on the features generated from both the region-based and trajectory-based clusters. Both approaches effectively detect the discriminative features in each class’s trajectory, which demonstrates high classification accuracy in animal movement data as well as hurricane data. However, a major issue with these two methods is that the temporal information is ignored in measuring the distance.

Nanni et al. [NP06] cluster the trajectories by incorporating the temporal information into OPTICS [ABKS99]. Recall in OPTICS the points are reordered by a characteristic distance, namely, the reachability-distance [ABKS99]. For clustering the trajectories, the distance between two trajectories $\tau_1$ and $\tau_2$ in time interval $T$ is given by

$$D(\tau_1, \tau_2)|_T = \frac{\int_T d(\tau_1(t), \tau_2(t))}{|T|} dt$$

(2.3)

where $d(x, y)$ is the 2D Euclidean distance and $\tau_i(t)$ is the position of the trajectory $\tau_i$ at time $t$.

The problem of clustering the trajectories is formalized to find the best temporal interval within which the minimum mean reachability-distance of all the non-noise points are achieved, namely, TF-OPTICS. The clustering result in each time interval is evaluated by considering both the negative mean reachability-distance, and the case in which long time interval has a lower quality value than that achieved by the subset of the long time interval. Thus, the quality function is given by

$$Q_2(D, I, \varepsilon') = -\frac{R_c}{\log_{10}(10 + |I|)}$$

(2.4)

where $R_c$ is the mean reachability-distance within the time interval $I$, $D$ is the trajectory set, and $\varepsilon'$ is the density threshold parameter in OPTICS.

The algorithm for searching the maximum time interval is also given. Specifically, an initial time interval is randomly chosen from the subset of the largest time interval. The iteration is to continually find a better time interval that achieves larger $Q_2(D, I, \varepsilon')$ in the set that extends the previous set by one unit time both backward and forward. The effectiveness of the TF-OPTICS is evaluated by C4C, a synthetic datasets generator [GMPR05]. It shows that, 1) the TF-OPTICS is effective since the computation time grows linearly with the number of trajectories, and 2) with respect to the clustering
results, TF-OPTICS always shows better clusters than K-means, average-based hierarchical approach, or OPTICS, while it shows low coverage than the other methods. The main problem of this method is that, in the synthetic mobility traces, the mobility traces are normally generated at a regular time. In the real-world movement data, however, it is common to have some traces collected irregularly in time, therefore these trajectories may mislead the clustering results.

Pelekis et al. [PKK+09] propose a variant of Fuzzy C-Means (FCM) to cluster the trajectories with the discovered Centroid Trajectory (CenTra). Basically, each trajectory is divided into $p$ sub-trajectories, with each of equal temporal intervals, and each sub-trajectory is mapped to the regular grids with user defined size. By extending the sub-trajectory $\varepsilon$-buffer along the directions that is vertical to its moving direction, the feature of each sub-trajectory is represented as a fuzzy set specifying the extent of the sub-trajectory occupies a list of cells, called region. The distance between two trajectories is measured by, 1) the distance between corresponding regions that contain the two sub-trajectories, and 2) the distance between fuzzy sets representing the features of the sub-trajectories. The clustering of the trajectory is initially to find the Centroid Trajectory that exceeds a region density threshold, and the clusters are generated by evaluating the distance to the Centroid Trajectory. The evaluations of this approach based on the trajectories of GPS-tracked trucks show that, the cluster representative fits the real movements better than an earlier method called TRACLUS [LH07]. However, the authors did not elaborate on the suitable choice of the number of sub-trajectories, which is a crucial factor in determining the length of the sub-trajectory and also the size of the region.

Lee et al. [LHLC11] demonstrate that the sequential patterns are useful for classifying the trajectories on road networks. The sequential patterns are a list of ordered road segments that exceed the minimum support threshold, which is determined by the information gain. The discriminative features are selected according to the F-score by comparing the negative set and positive set, and Support Vector Machine is used to classify the trajectories according to the selected features. The effectiveness of the methods is verified by using both synthetic data and real taxi trajectories. However, in both dataset the number of classes is small, specifically two, thus the classification effectiveness on more classes is unclear, especially when the trajectories share some common roads.

Li et al. [LDHK10] detect moving objects by clustering their trajectories among any shape of locations of non-consecutive timestamps. The method proposed in [LDHK10] can detect a group of moving objects that may be geographically close to each other most of the time but they may also be far apart for a period of time. In the preprocessing step, the clusters of locations are detected using DBSCAN [EpKSX96]. A swarm is defined

\[16\] Since inferring the clusters of locations from the raw trajectory is not the main concern of this paper, Li et al. [LDHK10] choose the popular density-based method, but discuss a few other methods as possible alternatives.
by a pair \((O, T)\), where \(O\) denotes a set of objects and \(T\) denotes a set of timestamps. The pair satisfies the following three conditions: 1) the number of objects should exceed a minimum threshold, 2) the objects are in the same cluster for a minimum number of timestamps, and 3) at least one cluster of locations contains all the objects over certain timestamps. A closed-swarm is a maximal swarm for which neither the set of objects nor the set of timestamps can be enlarged. Li et al. [LDHK10] propose an algorithm to find the closed-swarms as follows. Given the list of objects, a depth-first search algorithm with two pruning rules, namely, apriori pruning and backward pruning, is performed on the complete sets of objects to find the sets that satisfy the closed-swarm conditions. A brute-force solution is performed to check whether every pair of the objects indeed satisfies the definition. The swarm mining algorithm is implemented in a demonstration system [LJL+10], and the algorithm is validated on real animal movement trajectories. The correctness of the algorithm can be proved theoretically. However, practically there is no effective method to verify that the generated closed-swarms are indeed a group of objects with similar mobility. Although the swarm mining is useful for analyzing the mobility patterns of a group of animals, it may not be adequate for the study of individual human mobility since human trajectories on a given route may be of high similarity and individual human seldom wander far away from some timestamp.

By using the temporal information in the trajectories, trajectory clustering can be further extended to uncover the periodicity of the movements [LDH+10, LWH12].

Li et al. [LDH+10] formulate the problem of detecting the periodicity of the trajectories, and an algorithm called Periodica is proposed to uncover the periodic mobility behaviors. The periodic mobility behaviors are detected based on reference spots. These reference spots are the regular grids that exceed certain density threshold, and the density of each grid is computed based on raw trajectories by using the kernel method. The periods are detected on each reference spot individually using Discrete Fourier Transform [JHK10] and autocorrelation [JHK10]. When each reference spot is considered individually, the mobility sequence is converted to a binary sequence, where “1” indicates that the mobility entity is in the reference spot and “0” indicates otherwise. The periodic behaviors are then detected among all the reference spots on each period. Given a specific period, the mobility sequence is converted to a symbol sequence according to all the reference spots, and the sequence is further divided into segments, where each lasts for the length of the period. The periodic movements among the segments are measured by the Kullback-Leibler divergence. The number of periodic behaviors is determined by a hierarchical clustering method. Initially, each segment represents a different periodic movement, and then in the following step the segments with lowest Kullback-Leibler divergence are merged. The algorithm continues until the pre-determined number of periodic movements is achieved. The Periodica algorithm is evaluated on both synthetic data and also the real movement data of bold eagles.
Li et al. [LWH12] propose a probabilistic model for period detection on binary sequence. A probabilistic measure of periodicity, namely, discrepancy score, is firstly defined, and Li et al. show that the true periodicity maximizes the discrepancy score. Based on the designed probability measure, a random observation model is proposed to detect the periodicity for the case of incomplete observations in the sequence. The proposed model is thoroughly evaluated both theoretically and experimentally on synthetic data.

Both the period pattern mining methods [LDH+10, LWH12] could be applied to detect patterns from individual’s mobility sequences. The mobility sequence can be first converted into a binary sequence with regard to a specific location, i.e., “1” indicates in this location, and “0” indicates not in this location. Li et al. [LWH12] also introduce the symbol “-1” to indicate an unknown location at a given timestamp. However, in both methods the characteristics of individual’s mobility behaviors are not considered, e.g., the order of visiting the locations, the spatial and temporal relevance of the locations, etc.

### 2.3.2 Graph-based trajectory mining

There are two types of approaches for describing trajectory patterns from raw traces. In the first type of approaches, the trajectories are described as transitions among places of significance to the given scenarios. Therefore, individual’s mobility is described as a graph, where each node indicates the places the individuals have passed by and the edges are enriched with other features, such as the duration of transitions and the frequencies [FHR09b, FHR09a, LZX+08, ZZXM09, CLC10]. In the second type of approaches, the trajectories are converted to spatio-temporal items [CMC05, GNPP07, CMC07], in which the frequent patterns and periodic patterns are discovered.

Fang et al. [FHR09b, FHR09a] construct a personalized activity map from raw trajectories and further model individual dynamics. Significant locations are detected based on both dwelling time and visiting frequency. Representative paths are extracted by Pairwise Curve-Merging (PCM) algorithm. At each time, PCM algorithm iteratively compares two traces, e.g., trace $P$ and $Q$, under a given tolerance $\epsilon$, such that each point $p$ from $P$ is compared with its nearest neighbor $q$ in $Q$. If $p$ is within the tolerance distance of $q$, and if $p$ is closer to $q$ than any adjacent point of $p$ in $P$, then $p$ is replaced by $q$, where an example is given in Figure 2.3. The personalized activity map, including significant locations, representative paths, and some additional information such as edge width, speed and number of traversals on the edges, is useful for modeling individual dynamics [FHR09a]. Individual’s current location is mapped to the nearest edge on the map. When reaching a crossroad, an outgoing edge is chosen by applying a particle filtering-based inference along with the historical probability of turning at each outgoing
Figure 2.3: A simple example of the Pairwise Curve-Merging algorithm given two traces $P$ and $Q$ according to tolerance $\epsilon$ in [FHR09b]. The distance between $p_2$ and $q_2$, denoted by $dist(p_2, q_2)$, is less than $\epsilon$, and also $dist(p_2, q_2)$ is less than the previous jump distance on the corresponding traces, e.g., $dist(p_1, p_2), dist(q_1, q_2)$, then either $p_2$ is replaced by $q_2$ or vice versa. Fang et al. prove that independent of the choice of replacing either $p_2$ by $q_2$ or vice versa, the Hausdorff distance [AGK+00] between two traces is always less than $\epsilon$.

edge. However, using particle filtering-based inference to predict the location at a cross-road requires sufficient number of historical trajectories because sparse data may bias the probability of choosing a road.

In [LZX+08, ZZXM09, CLC10], the path patterns are presented in a graph-theoretic manner, where the definitions for the nodes in the graph are different.

Figure 2.4: (a) A framework for demonstrating the arrangement of the stay points, represented by the red nodes, in the tree-based hierarchical graph. In each level of the tree, the stay points are grouped into different clusters, based on which each individual’s sequence is constructed according to the different tree level. (b) A sample for demonstrating the connections between individuals and locations at the third level of the tree on the left. Both figures are from [LZX+08]. (c) The adjacency matrix showing the relation between individuals (hubs) and locations (authorities) according to (b).

A tree-based hierarchical graph (TBHG) is proposed for modelling individual’s loca-
tion histories, based on which some useful information is extracted [LZX+08, ZZXM09]. A location history is a sequence of stay points, where an individual stays over a minimum period. The TBHG, consisting of a tree-like hierarchy and a graph associated with each level of the tree, is constructed from the location histories of all individuals. The hierarchy is obtained from all the individual’s stay points in a divisive manner. The graph associated with each level of the tree is obtained by sequentially connecting the clusters in a same journey. For instance, a three-level hierarchy is presented in Figure 2.4(a), where smaller clusters are formed on the lower level of the tree, and larger clusters are on the higher level of the tree.

Locations of interest and individual travel experiences are inferred based on a HITS-based inference model [ZZXM09]. In this model, each individual is considered as a hub and the clustered locations at a specific level are taken as authorities, as shown in Figure 2.4(b). This method is similar to HITS, in that the good hubs and good authorities are always connected. Let $a$ be the authority scores and $h$ be hub scores, e.g., according to the example in Figure 2.4(b), $a = (a_{31}, a_{32}, a_{33})^T$, $h = (h_1, h_2, h_3, h_4)^T$, where each $a_{3i}$ is the score of the cluster $c_{3i}$ and $h_i$ is the score of $u_i$. The relation between authority scores and hub scores are calculated as $a = M^T \cdot h$, $h = M \cdot a$, where $M$ denotes the adjacency matrix which is derived from the individual’s travel histories, e.g., the adjacency matrix in Figure 2.4(c) is derived from Figure 2.4(b). This method outperforms some baseline algorithms such as rank-by-count and rank-by-frequency either separately or jointly, in ranking the interesting locations and inferring travel sequences. The similarity between individuals is quantified through a similarity metric considering both the sequence of the trajectory and the hierarchical properties of the locations in the graph [LZX+08]. Although individuals sharing similar travel sequences can be effectively measured in [LZX+08], a group of colleagues or friends who do not share similar travel sequences may not be differentiable merely through similar sequences.

Chen et al. [CLC10] mine the movement patterns from individual GPS trajectories, and further predict both the destination and route through a pattern matching process. Forward-Backward Matching, a variant of time-based clustering algorithm, is proposed to cluster the origins and destinations from the trajectories. It initially generates the candidate clusters both in a forward and backward manner, and merges the candidate clusters that are at a small distance. Forward-backward Matching avoids the case when the trajectories are not continuously collected. Table 2.1 illustrates several such instances. The remaining points, except for the origins and destinations, are labeled according to a space-partitioning method. Chen et al. avoid the answer-loss problem [JLCZ06] in the following method. Firstly, each cell is labeled by the origin-destination pairs of the trajectories that pass it along with the counts. The cell $c_i$ is merged into its adjacent cell $c_j$, under the case that each origin-destination pair with a path passing through cell $c_i$ has a count less than that in cell $c_j$. Therefore, the trajectories are simplified into, 1)
Table 2.1: A sample of continuous trajectories and discontinuous trajectories with origins and destinations, and it is from [CLC10]. In the discontinuous trajectories row, a × indicates the loss of current trace. If the trajectories are continuously collected the time-based clustering algorithm can successfully detect the significant locations a, b, and c under a density threshold of 2. In the discontinuous trajectories, however, none of these significant locations are observed since all the locations bear a density less than 2 within a temporal neighborhood. The underlying assumption is that only the locations in the adjacent traces are within the same temporal neighborhood. For instance, in the discontinuous case the location a in trace 1 is not temporally adjacent to the same location in trace 3, thus the requirement of minimum density is not met.

<table>
<thead>
<tr>
<th></th>
<th>trace 1</th>
<th>trace 2</th>
<th>trace 3</th>
<th>trace 4</th>
<th>trace 5</th>
<th>trace 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>continuous</td>
<td>a → b</td>
<td>b → c</td>
<td>c → a</td>
<td>a → b</td>
<td>b → c</td>
<td>c → a</td>
</tr>
<tr>
<td>discontinuous</td>
<td>a → b</td>
<td>×</td>
<td>c → a</td>
<td>×</td>
<td>b → c</td>
<td>×</td>
</tr>
</tbody>
</table>

origin-destination pair, and 2) a cell sequence along with the pairs of origin-destination with a path passing through it and the corresponding counts. The proposed algorithm is superior to the 1st- and 2nd-order Markov model with higher accuracy in 1-step prediction, and resulting in smaller Hausdorff distance [HKR93] in continuous route prediction. Considering that mere geographic information, such as the location at each time step, is applied in the proposed algorithm, incorporating additional information, such as time, date, modes of transport, etc., may increase the performance of the algorithm.

2.3.3 Trajectory mining as spatio-temporal itemsets

By describing the trajectories as a list of regions-of-interest or rectangular regions, the path patterns are considered as the frequent spatio-temporal itemsets in [GNPP07, CMC05, CMC07]. Note that, [GNPP07] is the only study considering the temporal information in mining path patterns.

The objective of the study in [GNPP07] is to find the trajectory patterns that exceed the minimum support. A trajectory pattern is defined as the trajectories sharing both similar sequence of locations and the travel times. A spatio-temporal sequence is a location sequence with travel times. The containment relation among the sequences is defined based on the neighborhood function and the time tolerance.

An example of the containment relation between two spatio-temporal sequences $T$ and $I$ is given in Eq. (2.5) and Eq. (2.6). Each of the sequence is indicated by the transitions between location along with time. Given the time tolerance $\tau$, $T$ is contained in $I$ if the corresponding travel time satisfies, $|t_1 - t_1'| \leq \tau$ and $|t_2 - (t_2' + t_3')| \leq \tau$. 

21
Chapter 2. Literature Review

\[ T : \{a\} \xrightarrow{t_1} \{b\} \xrightarrow{t_2} \{c\} \quad (2.5) \]

\[ I : \{a\} \xrightarrow{t_1'} \{b; d\} \xrightarrow{t_2'} \{f\} \xrightarrow{t_3'} \{c\} \quad (2.6) \]

where in sequence \( I \), the locations \( b \) and \( d \) are considered as synonymous since they are close to each other.

The neighborhood function is used to cluster the locations into meaningful areas, i.e., static or dynamic Regions-of-Interest. The static Regions-of-Interest are generated before the mining process, while the dynamic Regions-of-Interest are generated along with the process of mining the sequences. The experimental results on real and synthetic data indicate that the running time of both algorithms are almost linear to the length of input sequences. The main problem in defining the trajectory pattern in [GNPP07] is that, the transition time is not well defined, which involves both the travel time between two locations as well as the dwelling time at the destination.

The mined pattern or the so-called \( T \)-Pattern is useful in predicting the next location [MPTG09]. Firstly, the extracted \( T \)-patterns are ordered in the \( T \)-Pattern tree, where each node contains the entries of region ID, support, and the pointers to the children. The region in this case is referring to the regular grid of high visiting frequency. One node points to another node only when they are observed sequentially in the trajectories, and the connection is enriched by the minimum and maximum travel time between the corresponding regions in the nodes. When predicting the next location, the given trajectory history is matched with the tree according to the specified spatial and temporal tolerance. By tuning the spatial tolerance, indicating the maximum distance of the current trajectory to the matched \( T \)-pattern tree, the balance between the predicting rate and accuracy is adjustable. The authors suggest that the major contribution is that the prediction based on \( T \)-pattern tree allows the end users to specifically choose between predicting rate and the accuracy by tuning the spatial tolerance. A similar study about balancing between the predictability and the accuracy of the predictions is also reported in Chapter 4. However, when the spatial tolerance is small, e.g., 10 meters, the predicting rate is less than 40%, and the corresponding predicting accuracy is around 60%. When the predicting rate is low, they did not propose any method to handle the case. Some possible solutions regarding the spatial and temporal relevance of the locations can be applied in these cases [LLH13].

Cao et al. [CMC05, CMC07] aim to mine frequent patterns and periodic patterns from a spatio-temporal sequence extracted from a GPS trajectory. A series of rectangular regions are used to represent the trajectory into a spatio-temporal sequence. Frequent singular patterns are discovered by using a heuristic called growing [CMC05]. A segment with median length is first chosen. In the filtering and verification steps, the chosen
segment is merged with other segments that share similar characteristics. This process is repeated until either all the segments have been assigned to a region or no region is found for the segments. In the latter case, the segments are outliers. To mine the frequent substrings, a stack is employed in a substring tree, which is a rooted directed tree with the root linking to multiple substring sub-trees. According to the experimental results on verified bus trajectories, the proposed method is more effective than the grid-based method, since some of the sequential patterns within a grid cannot be captured in the grid-based method. Also, the substring tree method is more efficient, in terms of execution time, than the level-wise method and grid-based method.

The objective in [CMC07] is to discover all the valid, frequent and nonredundant patterns with respect to a minimum support. A valid pattern is a sequence of valid regions, with each representing a dense cluster following the definition of DBSCAN [EpKSX96]. A pattern is redundant if it can be implied by the other patterns. The frequent 1-patterns are initially generated from the dense clusters. An example of valid pattern and redundant pattern is given in Figure 2.5 (b) and (c). Three methods are proposed to find longer patterns, which are a variant of level-wise Apriori-TID algorithm [AS94] (STPMine1), a faster top-down approach (STPMine2), and a simplified version of STPMine2 (STPMine2-V2). In STPMine1, the candidates for the frequent patterns with length $k$ are generated from those of length $k-1$. The ones with support exceeding a threshold and an additionally valid pattern are chosen to be the frequent patterns of length $k$. In STPMine2, the trajectory is converted to a time series sequence, where periodic patterns are mined according to a modified algorithm in [HDY99]. STPMine2-V2 mines the patterns approximately in an efficient manner. The experimental results show that both STPMine1 and STPMine2 identify the same longest patterns, but STPMine2-V2 is able to find similar longest patterns. Both STPMine2 and STPMine2-V2 are more efficient than STPMine1, since the former two scan the data less number of times than the latter. However, in all the cases, the length of periodic patterns has to be specified, and the frequent periodic patterns of different lengths cannot be automatically identified.

In summary, the key point in the study of trajectory mining is to find certain representative locations\footnote{In the study of trajectory mining, not all the representative locations in describing the trajectories share the idea of significant locations that I have discussed in Section 2.1 since in some cases all the points in the raw trace are retained.} in describing the raw trajectory. By converting the raw trajectory into a list of representative points, the topics such as trajectory clustering and trajectory pattern extracting can be conducted in a similar manner to the study of clustering, frequent item mining or graph studies.
Figure 2.5: (a) A sample of simplified trajectory pattern in [CMC05, CMC07]. The trajectory for \((x_i, y_i)\), where \(1 \leq i \leq 6\), is simplified by its representative line \(l_{ij}\) according to the distance threshold \(r\). The path patterns of two trajectories are considered to be similar if the corresponding representative lines satisfy two conditions, namely, 1) the angle between two lines is less than a threshold, 2) the distance between the two lines is less than a threshold. (b) and (c) A sample of valid and redundant patterns defined in [CMC07], respectively. In Figure (b), each point falls to the corresponding cluster indicated by the ellipse, and the valid pattern is \(P = C_1 - C_2 - C_3\), where each \(C_i\) indicates a cluster. Figure (c) indicates a pattern \(P' = R_1 - R_2 - R_3\), which can be implied from \(P\), thus it is a redundant pattern. Figures (a) is from [CMC05], and Figures (b) and (c) are from [CMC07].
2.4 Location-based Activity Recognition

Inferring location-based activities is a procedure of learning behaviors from movement or positioning related data. The data can be, for instance, the reading from the GPS devices, the road that the individuals currently on, the information from a digital map, etc. The behaviors can be the destination for the trip, the purpose of this trip, or the kind of activities that will take place at the destination, etc. A few location-based activities can be recognized through GPS data, such as working, visiting friends, on or off a car, dining, shopping, etc. Generally, the individual’s mobility is modelled by types of conditional models, where a few examples are shown in Figure 2.6 and Figure 2.7.

![Figure 2.6: (a) The general full conditional model. (b) The factorized conditional model given in [DGP12]. The number of parameters of the full conditional model increases exponentially with the increasing size of the feature space. Therefore, when the feature space is large, the required training dataset becomes very large. However, by handling the features in a group rather than individually in the factorized conditional models the parameter space may be greatly reduced, which is especially useful in modeling individual’s mobility.](image)

Do et al. [DGP12] infer individual’s next place and the dwelling time at the place by using a factorized conditional model according to the extracted features. The factorized conditional models can greatly reduce the size of parameter space than that from the full conditional model given in Figure 2.6. Specifically, the location information they applied is from the source of GPS sensors and WiFi data. The individual’s trace is converted to a sequence of meaningful locations indicated by grid cells, which are obtained by merging the locations recorded inside. The tasks of predicting the next location and the duration of stay are reduced to finding the conditional relations between the feature space $X$ to the behavior space $Y$ in the factorized conditional model

$$P(Y|X) = \frac{\prod_{k=1}^{K} P_k(Y|C_k)^w_k}{Z(X)}$$

(2.7)

25
where $C_k \subset X$ indicates a subset of features, $w_k$ is the weight for each distribution $P_k$, and $Z(X)$ is the normalizing constant.

The weight of each distribution is learnt based on the training data collected from all the individuals, whereas the parameters for each distribution is exclusively learnt according to the individual’s data. The experimental results show that the accuracy in predicting the next location by using a combination of the subset of the features in the factorized conditional model is more than 10% higher than that based on any single feature or the combination of a few, and a similar result is reported for the prediction for the duration of stay. However, one issue regarding applying the factorized conditional model is the choice of appropriate subset of features. In the original study, the authors obtained the combination by a greedy search process, which always chooses the features that achieve the highest predicting accuracy. However, in general, the best overall results may not be obtained by choosing the list of the features that show the best performance individually.

In the Bayes inference, the posterior probability density for the variable of concern, e.g. individual’s state at current time $x_k$, conditioned on the data so far $z_1, \ldots, z_k$ can be recursively estimated,

$$ p(x_k|z_{1:k}) \propto p(z_k|x_k) \int p(x_k|x_{k-1})p(x_{k-1}|z_{1:k-1})dx_{k-1} \quad (2.8) $$

where the state $x_k$ may contain different factors. In the following I review a few approaches in details.

Patterson et al. [PLFK03] construct a dynamic Bayes net model to infer high level behaviors, such as modes of transport and transportation routes, from noisy GPS data. The dependencies, observable nodes and hidden nodes are presented in Figure 2.7(a).

The state $x_k$, defined as $x_k = (e, d, v)$, indicates that the individual is on edge $e$, at a distance $d$ to the start vertex of $e$ and is traveling at a speed $v$ on the edge. The current location is $l_k = (e, d)$, and $\sigma_k$ is the expected sensor error. The transportation mode $m_k$ is chosen from $\{BUS, FOOT, CAR\}$, and the velocity $v_k$ from each mode is generated according to the Gaussian distribution, where the parameters for the Gaussian distributions of three different modes are learned by a Gaussian mixture model. Since the mode of $FOOT$ is unlikely to generate a high speed value, the velocity from this mode is generated by the Gaussian distribution with smallest mean. The mode of $CAR$, however, is still possible to generate a low speed value, therefore the speed from $CAR$ is assumed to be evenly chosen from the three Gaussian distributions, which is an intuitive choice but it has not been validated.

The posterior distribution of the state $x_k$ is represented in a particle filter method, where $S_k = \{(x^i_k, w^i_k)|i = 1, \ldots, n\}$, and the weight $w^i_k = p(z_k|x^i_k)$ is assigned according to the likelihood of the observation. The parameters, such as the edge transitions, and
Figure 2.7: Two typical dynamic Bayesian networks [PLFK03, LPFK07], including the observable nodes, hidden nodes and the dependencies between them, are presented in modeling individual’s mobility. In the dynamic Bayesian network, each node represents the variable at a given time, and the arrows indicate the dependencies between different variables. In both cases, only the GPS reading are the observable nodes, and the remaining ones have to be learned from the mode. The main difference between (a) and (b) is that, the model in (b) is in a fine-grained level, where the switching between modes, trips and goals are considered. In (a), the solid lines indicate the intra-temporal causal links and the dashed lines indicate the inter-temporal links.
mode transitions, etc., for this model are learned by iteratively maximizing the posterior probability in the EM algorithm [DLR77].

Liao et al. [LPFK07] model an individual’s daily movements by a three-level hierarchical activity model, where the dependencies and variables are shown in Figure 2.7(b). In this model, the lowest level estimates individual’s location and velocity from GPS readings. The middle level represents the modes of transport and segments of a trip. The highest level represents the individual’s next locations. The posterior distribution of the parameters based on the observations $z_{1:k}$ is separated into continuous and discrete parts,

$$p(x_k, m_k, f^m_k, f^t_k, \theta_k, \tau_k | z_{1:k}) = p(x_k | m_k, f^t_k, \theta_k, \tau_k, z_{1:k}) p(m_k, f^m_k, \theta_k, \tau_k | z_{1:k})$$ (2.9)

where the continuous parts are the locations and velocity in the state $x_k$, the discrete parts are transportation mode $m_k$, edge transition $\tau_k$, mode switching $f^m_k$ and trip switching $f^t_k$.

Similar to [PLFK03], Rao-Blackwellised particle filters [DdFMR00] are used for inference. In detail, the discrete states are sampled by a particle filter and the continuous states are sampled by a Kalman filter conditional on the discrete samples. The next locations, transfer locations, and transition matrices are obtained from the model given the structure and parameters estimated from the EM algorithm [DLR77]. Moreover, the errors are detected through two trackers. One tracker uses the learned model to predict individual’s ordinary routine, and the second tracker uses a flat model accounting for general physical constraints but it is not adjusted to the individual’s routine to detect something unexpected.

A three-level hierarchical model is applied in [LPFK06]. However, the inference algorithms are different from [LPFK07]. In [LPFK06], a discriminative relational Markov network (RMN) is used to identify significant locations, meanwhile a generative dynamic Bayesian network is used to learn transportation routines, infer goals and errors.

In [LMP01], the hierarchically structured conditional random fields are used to extract an individual’s activities and significant places [LFK07], which are a simplified version compared to those in [LPFK07, LPFK06]. The lowest level is the associations between each GPS reading and the relevant patch of street map, in which three feature functions are defined to indicate their relationships. The middle level consists of the activity objects, each of which is featured by temporal information, average speed, geographic information, and the compatibility between types of activities. The top level includes place objects, which are featured by the frequency of activities occurred, and cliques that count the number of different homes and work places. A conditional random field [LMP01] is constructed for labeling activities and places based on the features extracted.
All these hierarchical probabilistic models are not naturally inheritable. Given additional training data, the structure of the model must be learned again, as the parameters of the old model become inapplicable with the new data.

There are several other studies [EP06, FM13, JFG12] that target on recognizing individual’s daily activity patterns from the GSM-based localization, bluetooth communications [EP06, FM13], and even the activity survey data [JFG12]. These studies apply formal approaches such as, Latent Dirichlet Allocation [BNJ03], or Hidden Markov Models [Rab89], etc. The interested readers should consult these papers directly.

2.5 Mobility Models

In this section, I review major existing results of mobility models. Roughly, the mobility models are classified into four categories, namely random mobility models [SM01, JM96, RMsM01, HGPC99, HLV06, LH03], geographic mobility models [BSH03, JBRAS03], social mobility models [LYD06, GPQ07, MM06], and trace-based mobility models [RSH+08, LHK+09, HLR10, SKWB10]. In random mobility models, the mobility information, such as direction, speed, dwelling time, etc., are random variables, and a certain degree of dependency may exist among the consecutive states. In geographic mobility models, the obstacles are described as arbitrary polygons, and the objects’ mobility are simulated by considering both the effects of the obstacles and the effects from the mobility behaviors of the objects in various circumstances, such as moving on a given highway. Social mobility models mainly focus on similar mobility behaviors of individuals sharing a community and the effect of the communities on the individuals. Trace-based mobility models target on simulating some statistical properties, such as the power-law distributions for the flight length, pause time or inter-contact time, etc., observed from real trajectories. Since my study is based on the individual’s mobility patterns, the trace-based mobility models will be the major focus in this section. A comparison of the properties of individual’s mobility each trace-based mobility model targeted on is summarized in Table 2.2.

Rhee et al. [RSH+08] propose a human walk model (Truncated Levy Walk) based on the statistical properties observed from GPS data. They reveal two statistical properties: 1) the distributions of the flight lengths and pause times are well-fitted by truncated power-law distributions, 2) the mean square displacement is in accord with that of truncated Lévy walks. Thus, human walk is modeled similarly to the Lévy walks constituting a series of steps, each including the flight length ($l$), direction ($\theta$), flight time ($\Delta t_f$) and pause time ($\Delta t_p$). Both flight length and pause time are randomly generated from a specified Lévy distribution under the conditions that $l \leq \tau_l$ and $\Delta t_p \leq \tau_p$. Flight time is determined by $\Delta t_f = kl^{1-\rho}$, where both $k$ and $\rho$ are constants. Direction is uniformly chosen. This model shows a similar power-law distribution for inter-contact time with
Table 2.2: A comparison of trace-based mobility models. Although Song’s model [SKWB10] is based on mobile phone data, they report similar patterns of human mobility, e.g., fat-tailed distributions of pause time and flight length. Meanwhile two additional statistical properties are revealed, which are exploration and preferential return.

<table>
<thead>
<tr>
<th>dataset</th>
<th>TLW</th>
<th>SLAW</th>
<th>STEP</th>
<th>Song</th>
</tr>
</thead>
<tbody>
<tr>
<td>area of movements</td>
<td>GPS</td>
<td>GPS</td>
<td>same as TLW</td>
<td>GSM cell-tower data</td>
</tr>
<tr>
<td>hotSpot</td>
<td>mobility within a given area</td>
<td>constraint each object in a predefined subarea</td>
<td>rectangular region of waypoints</td>
<td>locations of high frequencies of visits</td>
</tr>
<tr>
<td>pause time</td>
<td>power-law distribution</td>
<td></td>
<td>fat-tailed distribution</td>
<td></td>
</tr>
<tr>
<td>flight length</td>
<td>power-law distribution</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICT</td>
<td></td>
<td>power-law distribution</td>
<td></td>
<td></td>
</tr>
<tr>
<td>path</td>
<td></td>
<td>Least Action Trip Planning (LATP) algorithm</td>
<td></td>
<td></td>
</tr>
<tr>
<td>additional statistical properties</td>
<td></td>
<td>present better expected contact time than TLW, Orbit, CMM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>remarks</td>
<td>These studies are based on the same GPS data, which collected under only walk mode</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
other human mobility studies. Also, when applying this model to simulate the mobile objects, the network based on two-hop relay routing algorithm [GT02] shows a performance somewhere in-between the overly optimistic performance based on random way points and overly pessimistic performance based on Brownian motions [SKW82]. The main drawback of this model is that, the locations are visited in a random manner rather than relative to their importance.

Lee et al. [LHK+09] aim to incorporate four statistical features of human mobility in their mobility model called Self-similar Least Action Walk (SLAW). The statistical features are summarized from existing studies, namely, (F1) truncated power-law flights and pause times, (F2) heterogeneously bounded mobility area, (F3) truncated power-law inter-contact times, and (F4) fractal waypoints. In SLAW, the fractal waypoints are firstly generated, and the flights between waypoints are shown to follow a power-law distribution. The traces are a series of sequentially connected fractal waypoints according to the Least Action Trip Planning (LATP) algorithm. Constraining each walker in a predefined subsection of the whole area, the model satisfies F2. Their simulation indicates that the traces generated by SLAW are effective in 1) exhibiting power-law distribution of inter-contact time, and 2) performing better in expected contact time than other mobility models, such as CMM [LYD06], Orbit [GPQ07], TLW [RSH+08], etc.

Hong et al. [HLR10] build the spatio-temporal mobility model (STEP) to simulate the spatio-temporal correlation observed from human traces. Life cycle function (LCF), which represents the changes of the number of visitors to a hotspot during an interval, is the spatio-temporal correlation considered in STEP. A map of waypoints is initially generated, and then the waypoints are split into rectangular regions, each indicating a hotspot. The average population of a hotspot \( h \) is proportional to \( W^h \), and the number of primary periods for the hotspot is inversely proportional to \( W^h \), where \( W^h \) is the number of waypoints in \( h \). An individual walker is constrained in a bounded area. The STC-aware LATP algorithm, which is rooted in LATP [LHK+09] and introduced the spatio-temporal attractions to a waypoint according to LCF, is to specify the order of visiting the subset of waypoints in the area.

In STEP, the distributions for both flight and inter contact learned from real traces are better fitted by the generated traces than other existing models. Also, the cumulative distribution function of speed of diffusion (SOD), represented as the average temporal distance over all object pairs, shows step-wise increase, which coincides with what are observed from real traces. Although STEP is the first mobility model incorporating the spatio-temporal correlations (LCF) and meanwhile matching the temporal characteristics (SOD) to the real traces, the evaluations are confined within two campuses, which may not be representative for describing human mobility in general.

Song et al. [SKWB10] target on two properties of human mobility, 1) the fat-tailed distribution of waiting time, and 2) the statistical principles, which are exploration and
preferential return that govern human mobility. Although this study is based on GSM cell-tower data, this study is important for two main reasons. 1) The reported fat-tail distribution of waiting time coincides with the findings in GPS data, which further validates this property. 2) The two statistical properties, namely exploration and preferential return, are observed in GSM cell-tower data. Exploration is the tendency to explore new locations, and preferential return indicates a significant probability of returning to the locations visited before. In this model, an individual can either explore new locations according to a probability inverse to the number of distinct locations visited so far, or with a complementary probability of returning to a location visited before. Also, a waiting time following a fat-tailed distribution is incurred at each location. This model simulates the basic statistical principles of human mobility, but the fat-tailed distribution of the waiting time may not be accurate with the GSM cell-tower data. Since the waiting time is defined as the time staying in the consecutive same cell tower service region and also each cell tower covers an area as large as 3 km$^2$, thus some daily activities within each cell tower area may be concealed.

The major differences between the first three mobility models [RSH+08, LHK+09, HLR10] and the last one [SKWB10] lie in the extent of descriptions of individual’s mobility. The first three models are constructed based on an individual’s walk data, therefore the mobility behaviors are described only for simple cases, where the mobility area is small and only one mode of transport is involved. Comparatively, the individual’s mobility behaviors are mostly captured in [SKWB10] since the statistical properties are extracted from a large population and over a rather long period.

2.6 Chapter Summary

I have reviewed many recent results on mining mobility patterns from GPS data. This review serves as a general guide for exploring the uncovered issues in individual’s mobility, which are discussed in the next chapters.
Chapter 3
Modes of Transport

In this chapter, I present my unsupervised method for inferring the modes of transport from unlabeled GPS logs. The modes include common means of transport, such as walking, riding a bike, driving or taking a bus.

3.1 Modes of Transport: An Introduction

Knowing the modes of transport can facilitate location-based services [RGKR07]. For instance, the ability to infer transportation modes can better understand individual’s mobility patterns [FHR09b, SVL+06], assist in the predictions of locations or routes [FHR09a, AEM+07], de-congest transportation system and further enrich the context-aware applications [ZZXM09]. Also, when armed with information about the current transportation mode, a location-based service [Yin02] can recommend routes in a timely fashion, making reference to the current mode and possibly the user’s preferences. Similarly, in a real-time tracking system, e.g., one for the shipping of the goods via multiple modes of transport, the travel time can be accurately estimated according to the specific modes at different mobility stages. Moreover, the uncovered knowledge about the individual’s specific routes and the corresponding modes of transport may provide an opportunity for those individuals to make certain social interactions, e.g., sharing same transport to work.

Most existing approaches [ZLWX08, ZLC+08, ZCL+10, RMB+10] for inferring transportation modes make use of supervised methods and hence they require reliably-labeled GPS data, which may be difficult to obtain. Moreover, the trained results based on one person’s data may not be transferrable to another person due to differences of individual behaviors. Patterson et al. [PLFK03] provide an unsupervised method under the assumption that the underlying distribution of the speed values is Gaussian. The Expectation-Maximization (EM) algorithm is applied to estimate the parameters of each Gaussian distribution. This method exhibits low accuracy, since the low speed values
can be generated from any mode of transport. Thus, it would be desirable to have an unsupervised method that is applicable to different people without requiring tediously labeled data and meanwhile achieve high accuracy as well.

My proposed method, namely MoDetect, is an unsupervised method for detecting the modes of transport. Instead of using individual speed instances to judge the transportation mode, which suffers from the fact that any spurious changes in speeds could affect the inferred modes, I consider segments of GPS trace as a more stable and robust source of information. For each segment, the high speed distribution is taken as the representative feature, which is learned either empirically or by using a kernel method. The similarity between segments is measured using the Kolmogorov-Smirnov Test on the cumulative distributions, which provides statistical assurance. MoDetect offers the same level of detection accuracy while completely removing the need of tedious manual labeling. My experimental results show a slightly higher detection accuracy compared to that of the best known method using a trained Decision Tree [HK06].

The remainder of this chapter is organized as follows. Section 3.2 briefly recalls the major approaches for inferring the modes of transport. Section 3.3 describes MoDetect in detail. Section 3.4 presents experimental results. Section 3.5 discusses about possible further improvements to the proposed approach. Section 3.6 summaries this chapter.

### 3.2 Related Work

As discussed in Section 2.2, most of the existing approaches [ZLWX08, ZLC+08, ZCL+10, RMB+10] for inferring the modes of transport use supervised methods, which require a training stage. Although Patterson et al. [PLFK03] use an unsupervised method to learn the modes of transport by Expectation-Maximization (EM) [DLR77], their studies are based on the assumption that the underlying distribution of the speed values generated from each mode is Gaussian. Their approach also requires supplementary information, such as bus stops and parking lots. In the experiments section, therefore, my method is compared with both the baseline unsupervised method by Patterson et al. and also the best supervised approach from the existing studies [ZLWX08, ZLC+08, ZCL+10, RMB+10].

### 3.3 An Unsupervised Method Based on Kolmogorov-Smirnov Test

Before describing my unsupervised method in detail, I clarify some terminologies first.
Chapter 3. Modes of Transport

- **Trip.** \( T = \{ p_1, p_2, ..., p_n \} \) denotes a trip, which consists of a sequence of time-stamped GPS data points. Each point \( p_i = (x_i, t_i) \) denotes a pair of latitude and longitude, and an exact timestamp, respectively.

- **Segment.** \( S_i = \{ p_{k_i}, p_{k_i+1}, ..., p_{k_i+m_i-1} \} \) is a segment of a trip. \( k_i \) is the initial index of the first record of the segment in the trip, and \( m_i \) is the length of the segment which corresponds to the number of GPS data points in \( S_i \). Our objective is to identify segments that are recorded under one single transportation mode.

- **Segment speed list.** \( SV_i = \{ v_{k_i}, v_{k_i+1}, ..., v_{k_i+m_i-1} \} \) is the sequence of speed values derived from the segment \( S_i \).

- **Low speed mode.** This term presently refers to the mode of walking\(^1\).

- **High speed modes.** This term refers to all the modes that exclude a low speed mode. Typical high speed modes include biking, taking a bus, and driving a car.

### 3.3.1 Key ideas of MoDetect

I assume that different modes of transport exhibit different patterns in the positioning data. Moreover, traces generated from the same transportation mode would exhibit similar patterns. Thus, inferring the modes of transport may be reduced to finding the right patterns and identifying an appropriate similarity measure. Figure 3.1(a) clearly indicates that the high speed distributions are quite distinguishable among different modes. Therefore, I define the pattern under each mode as the distribution for high speed values. Because every mode can generate low speed values and only a few modes, e.g., bus, car, etc., can generate high speed values, I consider the high speed values to be more informative than the low speed values. Therefore, a weighted bootstrap technique is introduced to emphasize on the high speed values from a segment since the high speed values are given larger weights in the resampling process. Also, generating more speed sets from a segment may reduce the impacts of complex traffic condition on each mode. The similarity between different segments is quantitatively measured by p-value, which is obtained from a two-sample Kolmogorov-Smirnov Test on the estimated underlying speed distributions empirically or by a kernel method. Based on the p-values between the segments, I apply a linkage clustering algorithm to hierarchically cluster all the segments to a specified number of modes. Some segments lack matches in terms of high speed distribution with other segments. These segments are labeled according to the transfer probability for different transit modes at locations of frequent visits. The details of each part are discussed in the following subsections.

\(^1\)This definition appears suitable for all the data sets currently available to us. A revision/extension of the definition may be required when special needs arise, e.g., if data are available from the elderly or mobility-challenged people.
3.3.2 Preprocessing: segmentation and noise filtering

Zheng et al. [ZLWX08] proposed a method to generate the segments based on changing points, which signify changes in the transportation modes. I adopt the concept and propose a segmentation method along with a noise filtering process motivated by the following observations.

First, the speed distributions are similar under the walk mode among the individuals. Figure 3.1(b) shows the speed distributions of the walk mode of two individuals and aggregated data from all individuals. This observation suggests that the walk mode could be identified using the statistical information of the speed from walk mode.

Second, two types of abnormal records may occur in a GPS log. The first type of abnormal records are considered as error readings. Positions captured in such records are usually far away from adjacent record positions. Without the assistance of a geographic map, I can still exclude these records through setting a speed threshold, e.g., 40 m/s. The second type of abnormality occurs with sharp turnings, which generate a zigzag trace. This happens especially when the GPS records are accumulated frequently, e.g., every 2 seconds, thus speeds calculated from the consecutive records may be unreliable. In such situation, I apply Kalman Filter [WB01] to smooth the traces. Figure 3.2 shows these two types of abnormalities.

For Kalman filter, the positioning information is modeled as:
Chapter 3. Modes of Transport

Figure 3.2: In the figure on the left, the blue zigzag lines are the original traces, in which some points suddenly jump far away. The black trace is the connection of the remaining points after removing the error reading through a speed threshold. The figure on the right shows the smoothing effect of the Kalman filter on the traces.

\[
x_k = Ax_{k-1} + w
\]  \hspace{1cm} (3.1)

where \( x_k \) is the positioning information at step \( k \), \( A \) is the process transition matrix, \( w \) is the process noise vector.

I assume that the noise is mainly from the speed and is identically distributed in both latitudinal and longitudinal directions. Therefore in my case, the model is:

\[
\begin{bmatrix}
    \text{lat} \\
    \text{lon} \\
    \text{lat}\_\text{speed} \\
    \text{lon}\_\text{speed}
\end{bmatrix}_k =
\begin{bmatrix}
    1 & 0 & t & 0 \\
    0 & 1 & 0 & t \\
    0 & 0 & 1 & 0 \\
    0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
    \text{lat} \\
    \text{lon} \\
    \text{lat}\_\text{speed} \\
    \text{lon}\_\text{speed}
\end{bmatrix}_{k-1} +
\begin{bmatrix}
    0 \\
    0 \\
    w_1 \\
    w_1
\end{bmatrix}
\]

where \( w_1 \sim N(0, \sigma^2_1) \) and \( t \) denotes the time gap in seconds. Both \( \text{lat}\_\text{speed} \) and \( \text{lon}\_\text{speed} \) are calculated in degree.

The measurement is modeled by

\[
z_k = Hx_k + v
\]  \hspace{1cm} (3.2)

where \( z_k \) denotes the measurement at step \( k \), \( H \) denotes the measurement matrix, and \( v \) denotes the measurement noise.

In my case, the measurement model is

\[
\begin{bmatrix}
    \text{lat} \\
    \text{lon}
\end{bmatrix}_k =
\begin{bmatrix}
    1 & 0 & 0 & 0 \\
    0 & 1 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
    \text{lat} \\
    \text{lon} \\
    \text{lat}\_\text{speed} \\
    \text{lon}\_\text{speed}
\end{bmatrix}_k +
\begin{bmatrix}
    v_1 \\
    v_1
\end{bmatrix}
\]
Figure 3.3: The speed distribution for a walk segment before and after smoothing. The process of smoothing involves the steps from Line 12 to Line 14 in Algorithm 1. After smoothing, almost all the speed values are less than 4 m/s, which seem more reasonable for the walk mode. Also, after smoothing, the number of speed values is small, therefore the curve appears in a stepwise shape.

where \( v_1 \sim N(0, \sigma_2^2) \).

The ratio between \( \sigma_1 \) and \( \sigma_2 \) affects the performance results. Comparing the smoothed trajectory with a digital map, setting \( \sigma_2/\sigma_1 = 5 \) gives a good matching.

Algorithm 1 gives a detailed description of the segmentation and noise filtering algorithm. In each individual’s log, the sampling rate is uneven, ranging from a few seconds to more than 10 minutes. Therefore, the purpose of Line 14 is to regenerate the speed values at a more uniform rate and meanwhile moderate the high speed values. Figure 3.3 shows that with this moderation the resulting cdf of the walk mode appears to be more reasonable. After the preprocessing step, the low speed segments are labeled as walk mode and the high speed segments are further differentiated in the following steps.

### 3.3.3 Clustering the high speed mode segments

#### 3.3.3.1 Choosing high speed values

A weighted bootstrap technique [Wu86] is firstly introduced to choose the high speed values from each segment. Given a segment \( S_i \) and its speed values list \( SV_i = \{v_{k_i}, v_{k_i+1}, \ldots, v_{k_i+m_i-1}\} \), each speed value \( v_j \) is resampled given probability \( p_j = v_j^2 / \sum_j v_j^2 \), where \( k_i \leq j \leq k_i + m_i - 1 \).
Algorithm 1 Segmentation and noise filtering algorithm

Input: GPS logs $T$, time interval $\delta_t$, speed threshold $v_{thd}$, top speed of walking $v_{walk}$, distance threshold $D_{thd}$

Output: a set of high speed segments and low speed segments

1: for each $p_i$ do
2:  if $t_i - t_{i-1} > \delta_t$ then
3:    generate a new trip starting from $p_i$, end the current trip at $p_{i-1}$;
4:  end if
5: end for
6: for each trip do
7:  for each $p_i$, calculate the speed $v_i$ do
8:  if $v_i > v_{thd}$ then
9:    remove current point from the trip;
10:  end if
11: end for
12: Apply Kalman Filter to get the smoothed points;
13: Recalculate the speed value for each point;
14: Moderate the speed values at a more uniform rate;
15: for each $v_i$ from the speed list do
16:  if $v_i > v_{walk}$ then
17:    label $v_i$ as high speed mode;
18:  else
19:    label $v_i$ as low speed mode;
20:  end if
21: end for
22: merge the distance covered by consecutive points of same mode;
23: if current distance > $D_{thd}$ then
24:  generate a segment and assign its type according to the speed over the merged distance;
25: else
26:  merge the current distance to nearby segment;
27: end if
28: end for
3.3.3.2 Estimating the underlying distribution

After resampling the speed values from each segment, I introduce two nonparametric methods, namely empirical estimation and kernel density estimation, to estimate the underlying speed distribution based on the sampled speed values from each segment. Figure 3.4 shows the cumulative distribution functions estimated by the two methods, where the kernel density estimator generates a smoother distribution.

**Empirical Estimation** calculates the cumulative distribution function from the observed speed values. Let \( F_i^{\text{emp}}(x) \) denote the cumulative step-function

\[
F_i^{\text{emp}}(x) = \frac{k(x)}{n_i}
\]

where \( k(x) \) denotes the number of speed values that are no larger than \( x \). When comparing between the cumulative step-functions between two segments \( S_i \) and \( S_j \), \( F_i^{\text{emp}}(x) \) and \( F_j^{\text{emp}}(y) \) are calculated at every speed value from both segments, each generating \( n = n_i + n_j \) probability values.

**Kernel Density Estimation** [Sil86] is a nonparametric method for estimating the probability density function through smoothing sample values. A list of speed values \( (v_1, v_2, ..., v_n) \) from a segment, are assumed to be generated from an unknown continuous probability density function \( f \). A Gaussian kernel density estimator \( \hat{f}(x; \sigma) \) is

\[
\hat{f}(x; \sigma) = \frac{1}{N} \sum_{i=1}^{N} \phi(x, v_i; \sigma)
\]

where \( \phi(x, v_i; \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(x-v_i)^2/(2\sigma)} \).

From the above estimated density function, the following equation can be derived

\[
\frac{\partial}{\partial \sigma} \hat{f}(x; \sigma) = \frac{\partial^2}{\partial x^2} \hat{f}(x; \sigma)
\]
Based on this equation, Botev et al. [BGK10] proposed an adaptive kernel density estimation method. The kernel is given by

\[ \kappa(x, v_i; \sigma) = \sum_{k=-\infty}^{\infty} \phi(x, 2k + v_i; \sigma) + \phi(x, 2k - v_i; \sigma) \]  

(3.6)

where \( \phi \) is the Gaussian kernel function mentioned before.

This method overcomes the shortcoming of sensitivity to outliers in Gaussian kernel estimator. Thus, I use the adaptive kernel method [BGK10] to estimate the underlying distribution.

For calculating the cumulative distribution function, denoted by \( F_{\text{ker}}^i(x) \), from the probability density function, I specify the number of steps, which are assumed to be equally divided between \( \min(v_1, ..., v_n) \) and \( \max(v_1, ..., v_n) \). According to [BGK10], the step number denoted by \( \text{cdfnum} \) is assigned to be 32, 64, 128, etc. Section 3.4 examines the effect of the choice of \( \text{cdfnum} \) in more detail.

### 3.3.3.3 Two-sample Kolmogorov-Smirnov test

The similarity between segments is measured by a nonparametric statistical test, namely the Kolmogorov-Smirnov Test, based on the estimated cumulative distribution function from the preceding step.

Two-sample Kolmogorov-Smirnov test [PTVF93] is used to quantitatively determine whether two samples are from the same distribution at a specified level of significance. The test measures the difference based on maximum distance between the cumulative distribution functions of two samples. Let \( F_i^\beta(x) \) and \( F_j^\beta(x) \) denote the cumulative distribution functions learned either empirically or by using a kernel method, where \( \beta \in \{\text{emp}, \text{ker}\} \). Let \( D_{ij} \) be the difference between two segments

\[ D_{ij} = \max_{x} |F_i^\beta(x) - F_j^\beta(x)| \]  

(3.7)

The null hypothesis test is rejected at level \( \alpha \) if the p-value is less than the threshold, i.e.,

\[ p_{ij} = f(\lambda) = 2 \sum_{i=1}^{\infty} (-1)^{i-1} e^{-2i^2\lambda^2} < \alpha, \]  

(3.8)

where \( \lambda = \sqrt{n}D_{ij} \), and \( n = n_in_j/(n_i + n_j) \). Stephens [Ste70] gives a modified version of the KS test, where \( \lambda = D_{ij}(\sqrt{n} + 0.12 + 0.11/\sqrt{n}) \).

A large p-value suggests that two segments are more likely to have been generated from the same distribution. Therefore, I choose the linkage clustering algorithm [HK06] to hierarchically cluster the high speed segments into a user-specified number of modes.
Algorithm 2 Segments clustering algorithm

**Input:** the set of high speed segments, target number of clusters

**Output:** segments in each cluster

1: for segment $S_i$ do
2:  calculate the segment speed list $SV_i$;
3:  calculate the weight $p_j$ for each speed value $v_j$ in $SV_i$, where $k_i \leq j \leq k_i + m_i - 1$;
4:  resample speed list according to the weights $(p_{k_i}, ..., p_{k_i + m_i - 1})$;
5:  end for
6: for each pair of segments $S_i$ and $S_j$ do
7:  estimate distributions for the resampled speed sets from $SV_i$ and $SV_j$, respectively;
8:  apply KS-test on each pair of the resampled speed sets;
9:  calculate mean p-value;
10:  end for
11: for segment $S_i$ do
12:  get the number of segments with p-value less than $\alpha$ with segment $S_i$;
13:  if the matched number < threshold then
14:    exclude segment $S_i$ from clustering procedure;
15:  end if
16:  end for
17: initialization, each segment as a cluster;
18: while current cluster number > input cluster number do
19:  find two pairs of segments with the largest p-value;
20:  merge the original speeds from two segments;
21:  update the mean p-value for the merged segments with other segments by applying KS-test on the resampling speed sets;
22:  end while
Algorithm 2 gives a high-level description of the proposed clustering algorithm. The segments with few matchings are filtered in Line 10 to Line 14. The threshold is set to be $N_s \times 1\%$, where $N_s$ is the number of segments. The noisy segments are handled in the postprocessing step. During the clustering procedure, I merge all the speed values of the segments in the same cluster so far. The rationale is that, with more samples I could more reliably infer the underlying distribution based on the speed values.

### 3.3.4 Postprocessing: classifying based on significant locations

In the postprocessing step, the segments that exhibit weak similarity in terms of the underlying speed distribution will be classified based on the transfer probabilities of different modes between transition places.

**Definition of transition places.** Significant locations are based on visit frequency or the dwelling time during a visit [NB08, KH06, AS03, HT04], as well as correlation among users [ZZXM09, CCJ10]. In the context of transportation modes, I identify places that may involve changing of transportation modes, such as home or a parking lot. Lacking the ground truth about such places (or time) a person changes his/her transportation modes, the starting and ending locations of each *segment* are taken as the possible locations where a person may initiate a journey with a new transportation mode. Such places are referred to as the transition places.

Density-based clustering algorithm, such as DBSCAN [EpKSX96], can be applied to identify a group of transition places. The parameters $Eps$ and $MinPts$ are used to define the density points, which should have at least $MinPts$ within its $Eps$ radius. The $MinPts$ is chosen to be 1, which means if the user revisits a place, this place will be included in the transition places. Otherwise, it is considered as a noise data point. Each transition place is a circular area, where the center is the mean position of the points and the radius is the largest distance from the center.

The transfer probability between transition place for each mode is learned according to the grouped segments from previous steps. For the un-grouped segments if the ending points are the identified transition places, the modes are designated by the highest probability between two transition places. Otherwise, the mode for this segment will be grouped according to the most temporally adjacent grouped segments.

There are other alternatives for classifying the remaining segments. For instance, Hidden Markov Model takes into account observation probabilities and transition probabilities in a unified way. One shortcoming of these sophisticated methods is that training these models requires sufficiently long labeled data. However, the mode of transport for each segment upon the post-processing step may not be temporally connected, since the unlabeled segments may not contain the information of the modes of transport in a trip. Therefore, these methods are not good choices for the post-processing step to classify the remaining segments in my case.
3.4 Experimental Results

MoDetect is evaluated with regard to two objectives based on the dataset GPS-Modes (see Appendix A.1 for details about this dataset). The first objective is to verify the effectiveness of each step of MoDetect. The second objective is to compare the performance of the MoDetect against that of the EM algorithm and Decision Tree.

3.4.1 Parameter setting

In the preprocessing step, the consecutive GPS points are divided into different trips if the time gap is larger than 20 minutes. When generating the segments, the top speed is set to be 2 m/s, which is a rather typical top speed for walking. The distance threshold to determine a segment is tested through the experiments, chosen from 200 meters to 400 meters at an increment of 50 meters each time. In the clustering procedure, the number of high speed modes is meant to be the input specified by the user for the clustering procedure, which, in the experiments, is obtained from labeled traces. The number of high speed modes depends on the individuals, but it is mostly 3 in the datasets I tested, namely, bike, bus and car. When applying kernel method to estimate the underlying distribution, the choice of number of steps, denoted by cdffnum, is tested, ranging from 16 to 256. In the post-processing step, there are two parameters in DBSCAN. The value of radius for each point used to link with the other points during clustering, denoted by Eps, is chosen empirically by varying between 20 meters and 60 meters at increments of 20 meters and the density threshold MinPts is set to be 1.

3.4.2 Effectiveness of the preprocessing step

The preprocessing step differentiates the GPS logs into low speed segments and high speed segments, where the low speed segments are directly labeled as the walk mode. Therefore, the effectiveness of the preprocessing step is measured in terms of the recall and precision for the walk mode. Figure 3.5 shows the precision and recall of both the distance and duration under the walk mode with respect to different distance thresholds. The precision of both distance and duration increases with greater threshold values. Since a higher distance threshold gets more accurate segments under the walk mode, the short segments generated from a high speed mode are removed. Meanwhile, the recall decreases with the increase of the distance threshold. Therefore, the choice of the distance threshold requires a trade-off between precision and recall of the detected segments generated by the walk mode. In the following experiments, I choose the distance threshold of 250 meters, which has a reasonable high precision and recall.
Chapter 3. Modes of Transport

Figure 3.5: Both the precision of duration and distance increase along with the distance threshold, while the recall of both decrease with the increase of distance threshold.

3.4.3 Effectiveness of the clustering step

The effectiveness of the clustering procedure is analyzed through comparing 1) the criteria used in measuring the distance, in terms of p-value, during the clustering procedure, and 2) the overall clustering accuracy of both kernel density estimator and empirical estimator.

Table 3.1: The mean p-values within the same modes or between different modes. The parameter $cdf$ is equivalent to $cdf\text{num}$ in Figure 3.6(a). The values that are larger than $\alpha = 0.05$ are highlighted. In most cases, the p-values within the same modes, e.g., bike vs bike, bus vs bus, etc., are larger than 0.05. In some cases, however, the p-values between different modes, e.g., bike vs car, bus vs car, are also larger than 0.05. For such cases, high clustering accuracy can still be achieved by using the fact that the p-values within the same modes are much larger than those between the different modes.

<table>
<thead>
<tr>
<th></th>
<th>$cdf=0$</th>
<th>$cdf=16$</th>
<th>$cdf=32$</th>
<th>$cdf=64$</th>
<th>$cdf=128$</th>
<th>$cdf=256$</th>
</tr>
</thead>
<tbody>
<tr>
<td>bike vs bike</td>
<td>.053</td>
<td>.518</td>
<td>.393</td>
<td>.270</td>
<td>.191</td>
<td>.126</td>
</tr>
<tr>
<td>bike vs bus</td>
<td>.002</td>
<td>.039</td>
<td>.018</td>
<td>.007</td>
<td>.004</td>
<td>.001</td>
</tr>
<tr>
<td>bike vs car</td>
<td>.009</td>
<td>.154</td>
<td>.098</td>
<td>.065</td>
<td>.041</td>
<td>.030</td>
</tr>
<tr>
<td>bus vs bus</td>
<td>.021</td>
<td>.446</td>
<td>.311</td>
<td>.184</td>
<td>.100</td>
<td>.050</td>
</tr>
<tr>
<td>bus vs car</td>
<td>.013</td>
<td>.217</td>
<td>.134</td>
<td>.078</td>
<td>.042</td>
<td>.024</td>
</tr>
<tr>
<td>car vs car</td>
<td>.021</td>
<td>.349</td>
<td>.220</td>
<td>.158</td>
<td>.103</td>
<td>.075</td>
</tr>
</tbody>
</table>

Figure 3.6(a) shows the p-values between the segments of same high speed modes or between segments of different high speed modes, with Table 3.1 listing the detailed
Figure 3.6: (a) The mean p-values within the same modes or between different modes. The green horizontal line indicates the level of significance at $\alpha = 0.05$. (b) The overall clustering accuracy in terms of segments. In both figures, $cdfnum = 0$ represents the results by empirical estimator and the others are obtained by kernel methods according to different $cdfnum$. In both figures, for the speed set from each segment, 10 different resampled speed sets are generated for computing the mean p-values. The clustering result remains invariant with respect to the number of resampled speed set.

values. In two-sample KS-test, the p-value is the higher the better within the same high speed mode, such as bike vs bike, and the lower the better between different high speed modes, such as bike vs bus. Generally, in all the cases, the p-values between different high speed modes are much less than those with the same mode, indicating that choosing p-value as the metric to differentiate high speed modes is feasible and effective.

A few observations can be made between the kernel estimator and empirical estimator.

- Kernel estimator performs better than empirical estimator in distinguishing between different high speed modes, where the p-values within the same high speed modes are much higher than those between different high speed modes.

- In the kernel density estimator, the p-values of all the pairs of high speed modes decrease with the increase of $cdfnum$. This fact reveals that increasing $cdfnum$ can better differentiate between different high speed modes, but the p-values also decrease between the segments from the same high speed mode.

Also, the following facts are useful for the comparison between different high speed modes.

- Within the same high speed modes, e.g. bike vs bike, bus vs bus, etc., the segments generated under the mode of bike show a higher similarity than those from the
modes of bus and car. All the mean p-values for bike vs bike are larger than 0.05, which enhances the result that they are generated from the same distribution.

- Among the pairs of different high speed modes, the segments generated from the mode of bike can be easily differentiated from the segments from the modes of bus and car, but the segments between bus and car are much more difficult to distinguish.

The comparison between different high speed modes are in accordance with the fact that taking a bus or driving a car may be affected by external factors, such as traffic flow, road condition, etc., whereas riding a bike is rather consistent in terms of speed distribution.

Figure 3.6(b) shows the overall clustering accuracy in terms of segments when estimating the underlying speed distribution empirically or by using a kernel method. Note that, the clustering result based on empirical speed distribution is indicated by the first bar where the parameter $cdfnum = 0$, while the clustering results based on speed distribution by using a kernel method are indicated by the remaining bars, where the parameter $cdfnum$ is chosen between 16 and 256. Kernel methods, attaining accuracies of more than 60%, are better than the results obtained empirically, which are about 40%. Also, different $cdfnum$ may slightly affect the overall accuracy. In particular, when $cdfnum = 64$, the clustering accuracy is higher than 70%.

### 3.4.4 Effectiveness of the post-processing step

The post-processing step handles the segments that lack similarity in speed distribution with the other segments. As it turns out, very few segments are processed in this step. Table 3.2 shows that the number of segments involved in the postprocessing is generally small, accounting for 1% or less of all the segments. Also, when tuning the $Eps$ from 20 meters to 60 meters, the detection accuracy remains high and changes slightly. Therefore, I set $Eps = 40$ meters to get the final results.

### 3.4.5 Comparisons with decision tree and EM algorithm

Here I compare the MoDetect against the EM algorithm and Decision Tree. The EM algorithm is effective for estimating the parameters of each Gaussian distribution from a mixed dataset. However, when applying the EM algorithm for detecting the transportation modes, it merely employs the instantaneous speed values, which are fundamentally different from MoDetect that considers the speed values in the wider scope of segments.

Besides comparing MoDetect with the EM algorithm, I also compare with the supervised method mentioned in [ZLWX08, ZLC+08, ZCL+10], which shows the best performance so far. Referring to the results reported in [ZLWX08, ZLC+08, ZCL+10], I slightly
adjust the values of the parameters in order to get a good detection accuracy\textsuperscript{2}. When generating the segments, the time interval is set to 20 minutes, and the velocity threshold, the acceleration threshold and the distance threshold are 2 m/s, 0.8 m/s\textsuperscript{2} and 200 meters, respectively. The features extracted from each segment include heading change rate, velocity change rate, stop rate, average velocity, expectation of the velocity, variance of velocity, the first maximum velocity and acceleration, the second maximum velocity and acceleration. The threshold for calculating the advanced features, for instance, heading change rate, velocity change rate, and stop rate, are 10°, 0.35 and 3.0, respectively. When choosing between a graph-based postprocessing and a normal postprocessing, the probability thresholds $T\textsubscript{1}$ and $T\textsubscript{2}$ are 0.6 and 0.36, respectively. Due to the lack of the geographic information around Beijing area, I arbitrarily choose $\text{MinPts} = 3$ and $Eps = 50m$ for generating significant locations. Although carefully chosen $\text{MinPts}$ and $Eps$ may increase the accuracy for postprocessing step, the increment is minor, about 4% from the report of Zheng [ZCL\textsuperscript{*}10]. These parameters are thus taken to be adequate for achieving a high accuracy for Decision Tree. About 70% of all the segments are trained, and the remaining are used for testing.

Table 3.3 shows the overall detection accuracies among different methods, where the results by MoDetect are obtained by assigning $cdf\text{num} = 32$. MoDetect achieves a slightly better detection accuracy, 4% higher in terms of duration and 3% higher in terms of distance, compared to Decision Tree which is the best known supervised method on unconditioned data reported to date. Meanwhile, MoDetect greatly improves the accuracies compared to the EM algorithm, by about 30% in terms of duration and 20% in terms of distance. The detailed information of the detection accuracy among different modes for MoDetect is presented as a matching matrix shown in Table 3.4 in terms of distance.

\textsuperscript{2}For the exact meaning of the following parameters, please refer to the papers by Zheng [ZLWX08, ZLC\textsuperscript{*}08, ZCL\textsuperscript{*}10]
Table 3.3: The overall accuracies in terms of duration and distance of detecting the transportation modes according to the EM algorithm, Decision Tree and MoDetect (MD). Accuracy by duration/distance is the duration/distance of correctly detected trajectory divided by the overall duration/distance of the trajectory.

<table>
<thead>
<tr>
<th></th>
<th>EM</th>
<th>DT</th>
<th>MD</th>
</tr>
</thead>
<tbody>
<tr>
<td>accuracy by duration</td>
<td>44.0%</td>
<td>69.2%</td>
<td>73.7%</td>
</tr>
<tr>
<td>accuracy by distance</td>
<td>57.1%</td>
<td>73.7%</td>
<td>76.3%</td>
</tr>
</tbody>
</table>

Table 3.4: Matching matrix of the detection results in terms of distance. The overall accuracy is 76.3% from the matching matrix.

<table>
<thead>
<tr>
<th>Ground truth</th>
<th>Detected results (KM)</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>mode</td>
<td>walk</td>
<td>bike</td>
</tr>
<tr>
<td>walk</td>
<td>1858</td>
<td>336</td>
</tr>
<tr>
<td>bike</td>
<td>119</td>
<td>3202</td>
</tr>
<tr>
<td>bus</td>
<td>195</td>
<td>919</td>
</tr>
<tr>
<td>car</td>
<td>272</td>
<td>1654</td>
</tr>
<tr>
<td>Precision</td>
<td>76.0%</td>
<td>52.4%</td>
</tr>
</tbody>
</table>

3.5 Discussions

Here I examine the time complexity of the algorithms needed in MoDetect. Two major steps appear to be the most time-consuming, i.e., 1) applying DBSCAN to generate transition places, 2) the calculating of p-values of all pairs of segments. However, this is not the case, since in both parts I handle each individual’s segments separately. As a whole the GPS dataset is very large for 11 individuals. It contains a few million GPS points. For each individual, however, the number of segments is very small as shown in Table 3.2. Among the individuals studied, the largest number of segments is less than 900. Since DBSCAN is only applied on the set of the starting and ending points of the segments, the time complexity incurred in identifying the transition places is approximately $O(n_s^2)$, where $n_s$ is the number of segments for a given individual. As for the calculation of p-values, due to the small number of segments, the maintenance of the p-values is not a problem even with the bootstrap procedure. Therefore, the overall time complexity for these two parts is approximately $O(n_s^2)$, which is acceptable in view of the small value of $n_s$.

Besides the KS test, I also considered another apparent choice of nonparametric statistical test, namely Chi-Square test [GN96] in the clustering procedure. Since Chi-Square test compares the distributions based on binned data, therefore in order to guarantee the testing results it has certain requirements on the sample size in each bin or the whole
sample size [GN96]. A commonly used condition is \( \min_i n \cdot p_i \geq 5 \), and a less restrictive one is \( n/r \geq 5 \), where \( n \) denotes the sample size, \( r \) denotes the number of bins, and \( p_i \) denotes the percentage of samples in each bin. In contrast, the KS test is not sensitive to the sample size [Mas51]. Moreover, Chi-Square test is sensitive to the number of bins, whereas KS test is a parameter-free test.

The KS test, however, is known to be a biased test [Kim69]. The testing results are accurate when the sample sizes are not too large. The statistical tool R cautions that the p-value is accurate\(^3\) only when the product of size of the two samples is less than 10000. In my case, when the two samples are of the same size of 128, the product equals to \( 128 \times 128 > 10000 \). Therefore, the p-values calculated under \( \text{cdfnum} = 256 \) may be inadequate under this situation.

Wrongly assigning the number of modes of transport in the clustering procedure may affect the overall results, especially when the assigned number of modes is different from the actual number. Even with a correct choice of the number of modes, the clustering accuracy may be biased when the number of segments collected from each mode is unbalanced. These shortcomings may be overcome by automatically learning the number of modes from the data, a challenging issue worthy of further study.

### 3.6 Chapter Summary

In this chapter, I have presented MoDetect for inferring the modes of transport without relying on large set of labeled training data. Certain contextual information, such as Significant Locations, are automatically extracted from raw GPS log and are utilized in MoDetect. The detection accuracy of MoDetect is comparable to (in some cases even slightly better than) that of the best known result achieved by using supervised methods, while avoiding the tedious and laborious labeling process.

Many issues remain open and they are worthy of further study. For instance, in MoDetect, the acceleration information is not used, which may be an informative source for differentiating high speed modes. The challenge is how to fuse the acceleration information with the high speed distribution in unsupervised method. Finally, as with the other known schemes, MoDetect is an off-line method. Hence, developing an equally accurate online method should find even broader applications in the mobile environments.

---

\(^3\)http://www.stat.psu.edu/~dhunter/R/html/stats/html/ks.test.html
Chapter 4

Predictability of Individual’s Mobility

In this chapter, I present the study of the predictability of individual’s mobility. The predictability measures the ability to foresee individual’s next move based on historical records.

4.1 Predictability: An Introduction

In mobile and pervasive computing, probabilistic models, such as Markov models [AS03, JSZ07, BD02], Bayes models [KH06, HT04], pattern mining method [JLSZ08] have been proposed to continuously predict the next move of individual’s mobility. Although the prediction accuracy has been greatly improved because of these efforts, little is known whether the prediction accuracy is already approaching the limit or whether further research efforts yield diminishing returns. Moreover, the prediction accuracy is apparently affected by the scale of the locations and the time interval concerned. For instance, while it is relatively easy to predict if the required spatio-temporal resolutions of the prediction are low, e.g., the person will probably be in the same country tomorrow, it is challenging to predict exactly where a person will be located between, say, 1045-1100am. How does the prediction accuracy vary with respect to the spatio-temporal scale? Again, little is known beyond intuition level, and it is difficult (or even meaningless) to compare the existing results as such.

In their breakthrough research on individual’s mobility, Song et al. [SQBB10] study the predictability issue based on the analysis of mobile phone data over a large population. Predictability is defined as the information-theoretic upper bound that fundamentally limits any mobility prediction algorithm in predicting the next locations based on historic records. Disregarding the apparent differences in individual’s daily routines, their analysis revealed a 93% predictability in individual’s mobility. This finding is fundamental to
many related fields [SQBB10], such as evaluating the performance of certain location prediction methods, and guiding the construction of individual’s mobility models.

By nature, however, the mobile phone data exhibit low resolution in both spatial and temporal dimensions, and they may be inadequate for further mobility research. In research that uses mobile phone data [SQBB10, GHB08], a user’s location is identified with the service coverage area of the base station, and the user’s locations are captured when he/she initiates or receives a call. Since it is extremely difficult to delineate the exact coverage area of a base station, it is approximated by using a Voronoi diagram. This may give a wrong indication of the user’s actual location. Moreover, using this scheme, the locations are generally of arbitrary shape, and they differ vastly in size, with the larger locations one or two orders of magnitude larger than the smaller ones. At an average size of 3km$^2$, a base-station imposed location may contain several distinct locations of interest. Furthermore, the number of instances collected from mobile phones is limited by the calls generated/received by the users, which could be as low as 3-4 per day. Hence, the hourly location string often contains a significant fraction of unknown locations [SQBB10]. Therefore, it is inherently difficult to further the mobility study based on the phone data. Presently, it is unclear whether the predictability findings remain valid with high-resolution data, and how the predictability varies under different spatio-temporal scales, e.g., with smaller locations or over shorter time intervals.

Comparatively, GPS data offers much higher resolutions, where the temporal resolution ranges between a few seconds to less than a minute and the spatial resolution is in the order of 10 meters. Thus GPS data enable us to study how the predictability varies at different scales of the size of locations and the length of time intervals.

My contributions lie in a few aspects.

- Firstly, my research refines the current knowledge about predictability, and it removes the difficulties arising from assigning a user’s location with the mobile phone data. For instance, my analysis reveals that the predictability can be as high as 90% at an hourly temporal resolution and a spatial resolution where the size of each location is roughly that of a large building–or about 1/20 of the average coverage area of cell towers. Moreover, the predictability is found to be independent of the size of the mobility area covered by each individual. Although the mobility area varies greatly among the individuals, a consistent predictability is obtained.

- Secondly, by varying the spatial scales of the grid cells over a wide range, I observe an invariance between the predictability and spatial resolution (or spatial uncertainty), namely they can’t both achieve high accuracy simultaneously. The invariance suggests that trade-offs may be needed between these two measures when designing algorithms for mobility predictions.
I also confirm both theoretically and empirically that redundancy and predictability are effectively the same statistical quantity in the context of mobility.

The remainder of this chapter is organized as follows. Section 4.2 reviews the existing work on predictability of mobility. Section 4.3 describes the mobility dataset used in this study. Section 4.4 presents the methodology. Section 4.5 presents the detailed experimental results. Section 4.6 briefly concludes this chapter.

4.2 Related Work

Two papers have pioneered the study of the predictability of individuals mobility [SQBB10, JLJ+10]. Song et al. [SQBB10] first defined the predictability as the limit of any algorithm for predicting individual’s future locations. The predictability is derived from the estimated entropies of the mobile phone data. The predictability is centered around 93% over a large population, independent of the size of the area covered by individual’s mobility or other demographic factors. As noted earlier, the high predictability is obtained based on low resolution positioning data. The service area of each cell tower represents the locations discussed in the chapter. The average size of a location is roughly $3 \text{ km}^2$.

For higher resolution positioning data, it is unknown whether the individuals mobility is still highly predictable, which motivates the current study. More importantly, the size and shape of the area covered by each cell tower is totally irregular.

Jensen et al. [JLJ+10] apply the same theory to analyze the predictability based on various types of mobile sensor data of 48 days’ records on average for 14 individuals. A similar high predictability is reported based on the discrete time series constructed from GSM, WLAN, blue tooth and acceleration data at a given window length independently. They further introduce the predictive information, representing the mutual information between the entropies with or without knowing the past history. By varying the time scales from a few minutes to a few hours, the highest predictive information is obtained when the time scale is 4 to 5 minutes. In Lempel-Ziv data compression algorithm, a comparison between the states in the discrete time series is needed, but the measurements that quantify the similarity between the states are not presented in this chapter. Also, their findings may suffer from the limit of each type of data. For instance, both WLAN and blue tooth data are capable of revealing the mobility of an individual in a small region but not between the regions, thus the high predictability based on these data does not generalize to wide-area mobility. For GSM data, the mobility behaviors over a large region can be observed, but they are of low resolution, and suffer from similar disadvantages with mobile phone data where the locations are of irregular shape and arbitrary size.
Chapter 4. Predictability of Individual’s Mobility

4.3 Mobility Data

In this study, the mobility dataset Zheng-GeoLife is used (see Appendix A.1 for details). To categorize different mobility behaviors, I define the radius of gyration $r_g$ as in [SQBB10]

$$r_g = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\text{surface-distance}(p_i, p_c))^2}$$  \hspace{1cm} (4.1)

where $p_i$ represents the location at time $i$, and $p_c = \frac{1}{n} \sum_{i=1}^{n} p_i$ is the center of the trajectory. $\text{surface-distance}$ denotes a function calculating the minimum distance over the earth’s surface between the locations given in longitude and latitude. The radius of gyration of the individuals in the dataset ranges from a few kilometers to more than one thousand kilometers.

The location string for each individual is encoded according to the spatial and temporal resolution into a string of location symbols. At an hourly resolution, for instance, the length of each location string for each individual contains $16 \times 7 \times 24 = 2688$ location symbols.

4.4 Methodology

Here I present the framework for analyzing the predictability of individuals mobility.

4.4.1 Grid cells and locations

**Definition 4.1** A grid cell is a rectangular region characterized by two parameters, namely the origin $=(\text{long}_i, \text{lat}_i)$ and spatial scale, denoted by $s$. The origin is a given point of reference. The parameter $s$ is a numeric value used to specify the size of the grid cell. A grid map is a collection of disjoint grid cells that collectively cover the mobility area in a GPS data set.

Strictly speaking, a region on Earth that is bounded by pairs of longitudinal and latitudinal lines is not rectangular. However, this simplified description may fit well when the scale is small.

Suppose the grid cell of a given grid map has a scale $s$. The increment of adjacent grid points differ by $0.001^\circ s$ in longitude or latitude. Let $p_0 = (\text{long}_0, \text{lat}_0)$ denote the origin of the grid map. The grid cell on the $i$-th row and $j$-th column of the map, labeled as $\text{Cell}(i,j)$ is bounded by its four corners, $p_{ijk}$, $k = 1, 2, 3, 4$. In terms of longitude and latitude, $p_{ij1} = (\text{long}_0 + 0.001 \times (i-1) \times s, \text{lat}_0 + 0.001 \times (j-1) \times s)$, $p_{ij2} = (\text{long}_0 + 0.001 \times i \times s, \text{lat}_0 + 0.001 \times (j-1) \times s)$, $p_{ij3} = (\text{long}_0 + 0.001 \times (i-1) \times s, \text{lat}_0 + 0.001 \times j \times s)$,
Chapter 4. Predictability of Individual’s Mobility

Figure 4.1: (a) An overview of one individual’s partial trajectory on a grid map. (b) The trajectory is converted into a series of transitions between grid cells.

\[ p_{ij} = (\text{long}_0 + 0.001 \times i \times s, \text{lat}_0 + 0.001 \times j \times s) \]. Figure 4.1(a) shows the mapping of one individual’s partial trajectory to a grid map. Figure 4.1(b) shows the trajectory in terms of the transitions between the grid cells.

The distance between the locations indicated by pairs of longitude and latitude are expressed in kilometers. One degree of longitude or latitude covers different length. For instance, around Beijing, China, 0.001° corresponds to 0.087 km in longitude and 0.111 km in latitude respectively. Thus, the area of each grid cell is controlled according to the scale factor \( n \), and the position of each grid cell can be varied according to the choice of the origin.

When mapping each individual’s trajectory to a grid map, the trajectory is converted to a location string. At a given temporal resolution, there may be multiple distinct locations recorded in a time interval, or there may be no records at all. For multiple locations, as with [SQBB10], the location for such a duration is chosen randomly according to the probabilities of visiting these locations recorded during that time interval.

For missing records, I apply the following rule originally from [AS03]. The GPS signals from the satellites are often disrupted inside a building. Thus, when the ending and beginning locations of a gap in the GPS records are the same, the user is taken as dwelling at the same location during that time. This rule also caters for the situation where the individual enters a building and exits later, or where the individual turns off the GPS devices in an indoor place. Otherwise, the user’s location during the time is considered as unknown and is represented by a “?” symbol. New method for estimating the entropy of a location string that contains the unknown locations is discussed in Section 4.5.
4.4.2 Entropy measurements

Let $X = (x_1, x_2, ..., x_n)$ be a discrete time location string converted from a raw trajectory, where $n$ denotes the time instance, and each $x_i$ represents a location encoded in terms of the grid cells. Suppose the location string $X$ is generated from a distribution $p(x) = Pr(X = x)$, where $x$ is from a finite set $A = \{C_i\}$. The following information-theoretic definitions of entropy may be applied to $X$.

**Definition 4.2** The random entropy $H^{\text{rand}}$ of $X$ is defined as

$$H^{\text{rand}} = \log_2 N,$$

where $N$ denotes the number of distinct locations in the location string.

**Definition 4.3** The temporal-uncorrelated entropy of $X$ is defined as

$$H(X)^{\text{unc}} = -\sum_{x \in A} p(x) \log_2 p(x) \quad (4.2)$$

The random entropy entirely ignores the frequencies of the locations. Hence it is an over-estimate of the actual randomness of mobility. The second entropy measure takes the frequencies into consideration. The distribution of $p(x)$ can be estimated through a maximum likelihood estimator, where $p(x)_{x \in A}$ is estimated from the current location string. This measure of randomness is temporal-uncorrelated, as it totally ignores the order of location symbols in the string.

**Definition 4.4** The entropy rate $H = H(X)$, also called per-symbol entropy, of $X$ is defined as

$$H = H(X) = \lim_{n \to \infty} \frac{1}{n} H(X_1, X_2, ..., X_n) \quad (4.3)$$

where $H(X_1, X_2, ..., X_n)$ is the entropy of the joint distribution of random variables $X^n_1 = (X_1, X_2, ..., X_n)$.

The entropy rate takes the order of locations into considerations, which can better measure the predicability of mobility. This measure can be estimated through Lempel-Ziv data compression algorithm [KASW98].

Given a location string $x = (..., x_{-1}, x_0, x_1, ...)$, let $x_i^j$ represent a substring beginning from index $i$ and ending at $j$. $L^n_i$ denotes the shortest substring starting at index $i$ that does not appear in the window $x_{i-n}^{i-1}$ of length $n$.

$$L^n_i = L^n_i(x_{i-n}^{i-1}) = 1 + \max\{0 \leq l \leq n : x_{i-l-1}^{i-l-1} = x_{j-l-1}^{j-l-1}, i - n \leq j \leq i - 1\} \quad (4.4)$$

$L^n_i$ is said to be a sliding window of length $n$. Similarly, $L^n_i$ represents a matching, where each time a match is found by searching through the entire history. Since the
entropy estimator based on a sliding window fluctuates with respect to the window size, I choose an increasing window matching method for estimating the entropy. The result is denoted as $\hat{H}_n$,

$$\hat{H}_n = \left( \frac{1}{n} \sum_{i=2}^{n} \frac{L_i}{\log_2 i} \right)^{-1}$$

(4.5)

### 4.4.3 Redundancy and predictability

**Definition 4.5** The redundancy [FMG92] of a string $X$, denoted by $\rho(X)$, is defined by

$$\rho(X) = \frac{(H^{\text{max}}(X) - H^{\text{true}}(X))}{H^{\text{max}}(X)},$$

(4.6)

where $H^{\text{max}}$ denotes the maximum entropy and $H^{\text{true}}$ denotes the true entropy of the string.

In my case, $H^{\text{max}} = H^{\text{rand}}$ and $H^{\text{true}} = \hat{H}_n$.

The redundancy is equivalent to the compression rate of a string, which is not the only factor that affects the predictability of the string [FMG92]. In the following, I show that the redundancy and predictability are approximately equal quantities in the context of mobility strings.

**Definition 4.6** Let $\Pi^{\text{max}}$ denote the maximum probability based on a given location string with the entropy $\hat{H}_n$, and $N$ the number of distinct locations presented in the location string. $\Pi^{\text{max}}(\hat{H}_n, N)$ is the solution of the following equation

$$\hat{H}_n = -[\Pi^{\text{max}} \log_2 \Pi^{\text{max}}(1 - \Pi^{\text{max}}) \log_2 (1 - \Pi^{\text{max}})] + (1 - \Pi^{\text{max}}) \log_2 (N - 1)$$

(4.7)

It is known that [SQBB10], $\Pi^{\text{max}}(\hat{H}_n, N)$ is an upper bound of the predictability, which is achievable by always predicting the historically most probable location as the next location.

In the following lemma (Lemma 4.1), we have the relation between the redundancy and the predictability of individuals’ mobility strings. Before showing this lemma, recall the definition of the binary entropy of a random variable.

**Definition 4.7** Let $Y$ denote a binary variable whose value is chosen from two values with probability $p_Y(i)$, where $i = 1, 2$ and $p_Y(1) + p_Y(2) = 1$. The binary entropy of $Y$, denoted by $H_b(Y)$, is given by

$$H_b(Y) = -[p_Y(1) \log_2 p_Y(1) + p_Y(2) \log_2 p_Y(2)].$$

(4.8)
Therefore $H_b(\Pi_{\text{max}}) = -[\Pi_{\text{max}} \log_2 \Pi_{\text{max}} + (1 - \Pi_{\text{max}}) \log_2 (1 - \Pi_{\text{max}})]$ is the binary entropy. Since the binary variable is chosen from two values, with probability $\Pi_{\text{max}}$ the algorithm makes a correct prediction and with the complimentary probability $1 - \Pi_{\text{max}}$ the algorithm fails to predict the next move.

**Lemma 4.1** The redundancy $\rho$ and predictability $\Pi_{\text{max}}$ are equivalent statistical quantities on measuring mobility strings.

**Proof:**

According to Definition 4.7, Eq. (4.7) is rewritten as

$$\hat{H}_n = H_b(\Pi_{\text{max}}) + (1 - \Pi_{\text{max}}) \log_2 (N - 1)$$

(4.9)

By approximating $\log_2 (N - 1)$ with $\log_2 N$

$$\hat{H}_n \approx H_b(\Pi_{\text{max}}) + (1 - \Pi_{\text{max}}) \log_2 N$$

(4.10)

Dividing both sides in Eq. (4.10) by $\log_2 N$ and replacing $\log_2 N$ with $H^{\text{rand}}$

$$\frac{\hat{H}_n}{H^{\text{rand}}} \approx \frac{H_b(\Pi_{\text{max}})}{H^{\text{rand}}} + (1 - \Pi_{\text{max}})$$

(4.11)

Rearranging the terms in Ineq. (4.11)

$$1 - \frac{\hat{H}_n}{H^{\text{rand}}} \approx \Pi_{\text{max}} - \frac{H_b(\Pi_{\text{max}})}{H^{\text{rand}}}$$

(4.12)

According to the definition of $\rho$, Ineq. (4.12) is changed to

$$\rho \approx \Pi_{\text{max}} - \frac{H_b(\Pi_{\text{max}})}{H^{\text{rand}}}$$

(4.13)

From Eq. (4.13), redundancy and predictability differ approximately by the ratio the two measures of entropy, i.e., $H_b(\Pi_{\text{max}})/H^{\text{rand}}$. Since the binary entropy $H_b(\Pi_{\text{max}})$ is within the range of 0 to 1, $H_b(\Pi_{\text{max}})$ is small, especially when $\Pi_{\text{max}}$ is close to 1. The random entropy $H^{\text{rand}}$ depends on the number of distinct locations, which is more than 64 when the spatial scale is 18, and more than 128 when the scale is 4. Thus the entropy ratio is minor in the context of mobility data.

Consequently, the redundancy and predictability may be considered as the same quantity when measuring the statistical property of mobility strings, which implies that high redundancy equals to high predictability and vice versa.

In the next section, experimental results are presented to verify the relation between predictability and the redundancy with the mobility strings.

58
4.5 Experimental Results

In this section, the following sets of experiments are carried out:

- Refining the method for estimating the entropy from incomplete mobility strings.
- Evaluating the impacts of varying the spatial scale and the temporal scale on the entropy measures.
- The entropy measures and predictability for 40 individuals at two spatial scales.
- The relationship between the predictability and spatial uncertainty (to be explained later) by varying the scales of the locations.
- How the predictability varies with respect to the radius of gyration.
- The effect of choosing different origins of the grid map on predictability.

From the experiments, I found that the entropy measures are more influenced by the spatial scales, rather than the temporal scales. In the subsequent experiments, I choose to vary only the spatial scale. When the spatial scale is 18, the size of each grid cell is around 3 km$^2$, which is similar to the service area of a cell tower in [SQBB10]. When the scale is 4, the size of each grid cell is approximately 0.15km$^2$, or that of a large building.

4.5.1 Estimating entropy from incomplete mobility strings

In the datasets, none of the individuals has complete records. Figure 4.2 shows the fraction of unknown locations when the spatial scale is 4 and 18 respectively. When the scale is large (or, alternatively, when the grid cells are large), the fraction of unknown locations is smaller since more data collected in a larger area are mapped into a grid cell, making it more likely to encounter the same grid cell label. For the majority of the individuals, the fraction of unknown location is between 0.2 to 0.9, which is much lower than the fraction of unknown locations between 0.7 to 0.9 in the mobile phone data [SQBB10].

In [SQBB10], after analyzing 8-day complete data, it was concluded that the estimated entropy may be considered as the true entropy when the fraction of unknown locations is less than 25%. They also show that the errors from the estimated entropy decreases when the length of the location string increases. In my case, location strings are 16-week long and the fraction of unknown location in them is less than 25%. As such, the estimated entropy may be considered as reflecting the true entropy.

Using mobile phone data, Song et al. [SQBB10] discovered a linear relation between the fraction of unknown locations, denoted as $q$, and the ratio of entropy calculated from
Chapter 4. Predictability of Individual’s Mobility

Figure 4.2: The distribution of the fraction of unknown locations.

the partial string and a randomly shuffled string, denoted as \( \ln(H(q)/H^r(q)) \). Therefore, the estimated entropy is multiplied by a ratio inferred at \( q = 0 \). With GPS data, although a similar linear relationship can be observed for a few individuals, the ratio \( \ln(H(q)/H^r(q)) \) varies greatly when \( q = 0 \). Thus, I am forced to look for a new method for estimating the entropy from a partial location string. The procedure is as follows.

Suppose the true entropy is \( H \) for a location string. For a string with \( q \) fraction of unknown locations, where \( q < 0.25 \), an estimated entropy \( \hat{H}(q) \) is calculated by using the Lempel-Ziv data compression algorithm given by Equation (A.2). The following steps are carried out based on these individual’s data to explore the relation between the estimated entropy and the fraction of unknown locations.

- Assign \( H = \hat{H}(q) \) (since \( q < 0.25 \))
- Randomly replace \( \Delta q \) fraction of locations by “?”, in order to mimic the situation where \( q + \Delta q \) fraction of locations are unknown.
- The estimated entropy, denoted as \( \hat{H}(q + \Delta q) \), is calculated from the substring by removing all the “?”.
- Let \( \Delta e \) be the error of the estimated entropy, calculated by \( \Delta e = \frac{\hat{H}(q+\Delta q) - \hat{H}(q)}{\hat{H}(q)} \% \).
- By gradually increasing \( \Delta q \), I can generate a plot for the error \( \Delta e \) versus the fraction of unknown locations \( (q + \Delta q) \).

By using the preceding procedure, I am able to analyze how the error of the estimated entropy varies with respect to the fraction of unknown locations. Figure 4.3(a) and Figure 4.3(c) show rather consistent error curves independent of the individuals and the scales. Based on this, I calculate the mean errors across different individuals at a given spatial...
Figure 4.3: (a) At the spatial scale 18, there are 4 individuals whose fraction of unknown locations is less than 0.25. (b) When applying the linear curve $y = p_1 \times x + p_2$ to fit the mean error curve, the parameters are $(p_1, p_2) = (0.8932, -0.1878)$. For each individual, two error curves are obtained by changing the origins twice for the grid map. The first origin is chosen from the minimum value of longitude and latitude in the person’s historical records, and the second origin is randomly chosen from his trajectory. (c) At scale 4, there are 2 individuals whose fraction of unknown locations is less than 0.25. (d) A linear curve is used to fit the error plot under scale 4, which has the parameters: $(p_1, p_2) = (0.6655, -0.158)$. 
scale, and a linear curve is fitted, as shown in Figure 4.3(b) and Figure 4.3(d). For the individuals with a greater fraction of unknown locations (than 0.25), the error for the estimated entropy is inferred from the fitted curve.

4.5.2 Predictability vs spatio-temporal resolutions

To evaluate the effects of spatial and temporal scales (sampling rate) on predictability, two individuals with more complete records are chosen. I define the blackout periods, between 8 pm and 6 am of the following day. During the blackout periods, the missing locations are treated the same as the previous locations recorded. The two individuals exhibit significantly different mobility behaviors, with their radius of gyration being 325 kilometers and 15 kilometers, respectively. Figure 4.4 shows the effects of spatial and temporal scales on $H^{\text{rand}}$, $\hat{H}$, and $\Pi_{\text{max}}$.

- Figure 4.4 shows that the higher temporal precision is required, the lower predictability can be achieved; moreover, the effect of the temporal scales on $H^{\text{rand}}$, $\hat{H}$, and $\Pi_{\text{max}}$ is clearly linear.

- For the first individual, the spatial scale has more effects since the values at different spatial scales are well separated from each other. By comparison, for the second individual, the changes in $\hat{H}$ and $\Pi_{\text{max}}$ are minor at different spatial scales. With a radius of gyration of more than 300 kilometers, the first individual exhibits a rather diversified mobility behavior, while the second individual has a radius of gyration of around 15 kilometers.

Thus the experiments suggest that changing temporal scales has similar effects on the predictability of different individuals, while changing the spatial scale has different effects, depending on the mobility characteristics of each individual.

Also, larger temporal scale leads to higher predictability, independent of the spatial scale chosen, as it is more likely to sample the same locations, which increases the redundancy of the mobility string and hence the predictability.

4.5.3 Entropy and predictability

Figure 4.5(a) and Figure 4.5(b) show the three entropy measures for the spatial scales 18 and 4 respectively. For both spatial scales, the values of the three entropy measures are well-separated, with the random entropy covering the widest range and the entropy rate narrowly distributed. For the larger grid cells (scale 18), the random entropy varies between 4 and 8, suggesting that the number of distinct locations visited by each individual varies between $2^4 = 16$ and $2^8 = 256$. Similarly, for the smaller grid cells (scale 4), the number of distinct locations varies between $2^5 = 32$ and $2^9 = 512$. 

62
Figure 4.4: Two individual’s three different measurements under different spatio-temporal scales. The temporal scales range from 5 minutes to 90 minutes, and the spatial scale ranges from 6 to 18. (a) to (c) show the first individual’s random entropy, estimated entropy rate and predictability, respectively. (d) to (f) show the second individual’s random entropy, estimated entropy rate and predictability, respectively. Two individuals exhibit different radius of gyration, e.g., the value for the first individual is 325 km, and the value for the second individual is 15 km.
Chapter 4. Predictability of Individual’s Mobility

Figure 4.5: (a) and (b) show the distributions of the three entropy measures when the spatial scale is 18 and 4, respectively.

In comparison, the entropy rate ranges between 0.5 and 2 in both cases, suggesting that the effective number of locations for each individual is rather consistent, varying from $2^{0.5} = 1.4$ to $2^2 = 4$. My study based on the high spatial resolution GPS data confirms the conjecture that although occasionally the individuals may visit a large number of distinct locations, most of the time they merely revisit a few locations, such as home and work place.

Interestingly, for the larger grid cells (scale 18 – where each grid cell is about the average size of the locations in mobile phone data), the range of entropy rate and random entropy are similar to that reported in [SQBB10]. When I further shrink the size of each grid cell to roughly $1/20$ of the size, I observe an increase in the entropy value.

Figure 4.6 shows the histograms of predictability, with a fitted curve for each plot. When the scale is 18, the predictability is peaked at 93% for all the individuals. The strong law of large number guarantees that when the sample is large enough the distribution for the value of predictability follows a normal distribution. Thus, a normal distribution is used to fit each plot. A Kolmogorov-Smirnov Test favors the hypothesis that the distribution of the predictability follows a normal distribution. Moreover, I observe that when the scale is 4, where each grid covers an area of 0.15 km², the overall predictability decreases to 90%. In other words, the number of effective locations increases by $2^{0.5} = 1.4$. In words, the predictability only decreases slightly even when requiring a much higher precision in the spatial dimension. It should be interesting that individuals mobility is highly regular even when the size of the locations are similar to that of a large building. This result is an important refinement to the findings reported in [SQBB10].

Although the overall trait for the predictability has minor variations (3%) when shrinking the size of the grid cells, for two particular individuals, the predictability decreases by about 7%. In detail, one decreases from 94% to 87%, the other decreases from
4.5.4 Predictability and redundancy

I verify the correlations between predictability and redundancy for all the individuals. Figure 4.7(a) and Figure 4.7(b) show that for both scales 18 and 4, the redundancy varies consistently with predictability, which verifies the theoretical derivation, and confirms that the redundancy of each individual’s mobility string leads to high predictability. Figure 4.7(c) indicates that when greatly decreasing the size of locations the redundancy in the individual’s mobility string also decreases. While reducing the spatial scale can offer better spatial resolution in predictions, it also lowers the redundancy in the mobility string, making the prediction of the next symbol more difficult. Also, the residual between the predictability and redundancy $R_z = \frac{H_b(\Pi_{\text{max}})}{H_{\text{rand}}}$ is plotted in Figure 4.7(d), where the individuals are sorted according to the radius of gyration which ranges between a few kilometers to more than 1000 kilometer. The residual is found to be independent of the radius of gyration.
Figure 4.7: The correlation between the redundancy and predictability of individuals mobility sequence. (a) and (b) show the redundancy and predictability for all the individuals when the scale is 18 and 4, respectively. (c) shows the redundancy for all the individuals for the scales 18 and 4 respectively. (d) The residual among all the individuals (sorted according to the radius of gyration).
4.5.5 Impacts of spatial scale on predictability

This section analyzes how the number of locations and predictability change with respect to the spatial scale, which is varied between 0.1 and 3000. 8 individuals with relatively more complete data are selected. The fraction of unknown locations is less than 0.4 even when the scale is 0.1. The 8 individuals exhibit rather different mobility behaviors. For 4 individuals, the radius of gyration $r_g$ is more than 200 km, whereas for the rest, $r_g$ is less than 50 km. Under each spatial scale, both the number of locations and predictability are averaged based on 100 randomly chosen origins for the grid maps in order to remove the effects for the choice of origins.

Figure 4.8: (a) The normalized number of location ratio versus spatial scale. $N_s$ refers to the number of distinct locations obtained under spatial scale $s$. $N_0$ refers to the initial number of GPS readings from each individual’s trajectory. The parameters for the fitted line $y = \beta x^\alpha$ is $\alpha = -0.54 \pm 0.02$, $\beta = e^{-2.49\pm0.11}$. (b) The predictability versus spatial scale. A linear curve is applied to fit the first part of the curve for each individual when the scale is between 0.1 to 100 (the fitted curve is not presented in this figure). The linear curve indicates a power law relationship between $\Pi_{max}$ and $s$, where $\Pi_{max} = cs^\alpha$ and $c$ is a constant. Among all the individuals, $\alpha \in [0.02, 0.05]$, and $c \in [e^{-0.28}, e^{-0.17}]$. Note that this is a log-log plot, where the Y axis is equally spaced according to the exponent (of 10).

Figure 4.8(a) indicates a power-law relation for the normalized location ratios when the scale increases for all the individuals. Thus,

$$\frac{N_s}{N_0} \approx \beta s^\alpha$$

Therefore $N_s \sim s^\alpha$, where $\alpha = -0.54\pm0.02$. This observation reveals certain interesting properties of individual mobility behaviors. An individual’s mobility merely covers a small portion of the grid map covering his trajectories. If most of the grid cells had been covered by the trajectory, the rate of decrease would have been close to $s^{-2}$.
Chapter 4. Predictability of Individual’s Mobility

Figure 4.8(b) shows an important trait for the predictability when increasing the spatial scale.

- Firstly, when $s = 0.1$, which corresponds to about $10^{-4} \text{ km}^2$ for each location, the predictability for each individual is around 70%, which is much lower than 93% predictability from mobile phone data.

- Secondly, a rather linear relation between the scale and predictability is observed from the log-log plot when $s \in [0.1, 100]$.

- The predictability stays largely constant beyond $s = 100$. In my method, the number of distinct locations is constrained by both the scale and origin. Thus even when the size of a location can cover the whole trajectory of one individual, e.g. when $s = 3000$, the number of distinct location may be more than 1, therefore the predictability can’t reach exactly 100%.

From the first part of the plot ($s < 100$)

$$\Pi_{\text{max}} \approx cs^\alpha \quad (4.15)$$

where $s \in [0.1, 100]$ and $c$ is a constant.

The parameter $s$ corresponds to the spatial resolution the algorithm achieves at each prediction step. Let $\mu = \frac{1}{2}$ be the level of spatial uncertainty of the prediction results for the algorithm, where larger $\mu$ means greater uncertainty. Therefore,

$$\mu^\alpha \Pi_{\text{max}} \approx c \quad (4.16)$$

This implies that the ability to foresee the future location of an individual (predictability) does not commute with the spatial uncertainty associated with each prediction. This phenomenon is reminiscent of the uncertainty principle in quantum mechanics where one can’t simultaneously determine the exact value of the position and momentum of a particle. In the context of individuals mobility, the invariance implies a trade-off between the predictability and spatial uncertainty, since it is impossible to design a prediction algorithm using GPS data to achieve both simultaneously with high accuracy.

4.5.6 Predictability vs radius of gyration

Figure 4.9(a) shows the distribution of the radius of gyration for the 40 individuals, varying between a few kilometers to more than 1000 kilometers. Approximately, 50% of the individuals exhibit a radius of gyration less than 100 kilometers, and the remaining are more than 100 kilometers. The wide range of radius of gyration indicates a great
deterministic location, to ensure that my experiments are conducted under a general setting. When plotting the predictability against $\log_{10}(r_g)$, Figure 4.9(b) shows that the predictability to be independent of the radius of gyration under both scales. It suggests that a large area of mobility does not necessarily lead to a low predictability. When the radius of gyration increases, the predictability does not converge as reported in [SQBB10]. It fluctuates at both spatial scales.

### 4.5.7 Sensitivity test on the choices of origins

In order to check the effect of the origin of the grid map on the entropy measures, a sensitivity test is conducted. Figure 4.10(a) and Figure 4.10(b) show that, the number of distinct locations remains rather invariant when shifting the origin. In rare (2 out of 100) cases, changing the origin does generate a drastically different number of distinct locations. These are considered outliers. For the majority, (6 out of 8 when the spatial scale is 4, and 7 out of 8 when spatial scale is 18), changing the origin leads to rather constant values for the estimated entropy rate, and no outlier values are observed. For the cases where abnormalities arise, the final results can be smoothed by randomizing the position of the origin for sufficiently large number of times, e.g. 100 times. Thus the shifting of the origins of the grid map has little effect on the results for the number of distinct locations and the estimated entropy rate. Since the predictability is calculated based on the number of distinct locations and the estimated entropy rate, it is also considered to be consistent with regard to different origins.
Chapter 4. Predictability of Individual’s Mobility

Figure 4.10: The box plot indicates the median (mark with the box), 25th, and 75th percentiles (the edges of the box), the extremal data points (the whiskers), and outliers (the + surrounding each box), for both the number of distinct locations and estimated entropy from 8 randomly chosen individuals. (a) and (b) show how the number of distinct locations vary with respect to different origins when spatial scale is 4 and 18 respectively. Similarly, (c) and (d) show how the estimated entropy vary with respect to the different origins for the two spatial scales. For each individual, 100 different origins are generated when encoding the trajectories by a grid map. For the first experiment, the origin is chosen as the minimum value of longitude and latitude from the trajectories, and the origins of the 99 remaining experiments are randomly chosen from the trajectory of each individual.
Chapter 4. Predictability of Individual’s Mobility

4.6 Chapter Summary

By using public GPS datasets, my study shows that the predictability can be as high as 90% when a location is about the size of a large building. Also, the study reveals an invariant relation between the predictability and uncertainty which suggests that certain trade-offs are needed when designing prediction algorithms. Since the main purpose of this chapter is to offer a general upper bound on predicting the next moves of an individual, no algorithm is explicitly given in this chapter.

Because of its fundamental importance, issues related to the predictability of individuals mobility deserve further study. Due to a lack of sufficiently long and detailed data (except for 2 individuals), this study evaluated the mobility strings mainly at an hourly sampling rate. How would predictability change at fine-grained temporal scale? Answering this question will require more detailed data to become available. Also, when applying LZ algorithm to estimate the entropy of mobility string, the stationarity of the string should be satisfied, otherwise the estimated result would have a low convergency rate. Given this, how can we model mobility for prediction at given spatio-temporal scales? Answering these basic questions will be most useful to the mobility research.
Chapter 5

A Markovian Mobility Model

In this chapter, I present a study of a Markovian mobility model that fits both the high predictability of individual’s mobility sequences and also several other important statistical properties of individual’s mobility.

5.1 Mobility Model: An Introduction

The study of mobility encompasses the modeling of individuals behaviors in traveling around [GHB08, BHG06, SQBB10, LHL12]. Mobility models have evolved from early models [RSH+08, LHK+09] making random transitions to recent ones [BHG06, SKWB10] matching well with aspects of reality. Realistic mobility models can help in epidemic prevention [BGB11, BV11, BPR+11], mobile virus control [WGHB09], and information spreading [WPS+11, MML11], etc.

Recall that in Section 2.5, I reviewed a few trace-based mobility models since they better reflect individual’s mobility behaviors. The differences among these trace-based mobility models [RSH+08, LHK+09, HLR10, SKWB10] lie in the extent of descriptions of individual’s mobility. The first three models [RSH+08, LHK+09, HLR10] are constructed based on data collected over a single mode of transport, i.e., walking. Therefore, the mobility behaviors are described only for the relatively simple cases, where the mobility area is small and only one mode of transport is involved. In comparison, the individual’s mobility behaviors are more comprehensively captured in the model described in [SKWB10] since the statistical properties are extracted from a large population and over a relatively long duration.

Specifically, the study in [SKWB10] based on extensive mobile phone data revealed certain statistical characteristics which they called the scaling properties of human mobility: a) the number of distinct locations \( S(t) \) follows \( S(t) \sim t^\mu \) with time \( t \), b) the visit frequency of the \( k^{th} \) location may be approximated by \( f_k \sim k^{-\zeta} \), and c) the mean square displacement follows an ultra-slow diffusive process, where the parameters \( \mu \) and
\( \zeta \) vary according to the individual. All these properties are not captured by either the Continuous Time Random Walks (CTRWs) model in [BHG06], which is based on the traces of bank notes, or by the theoretical models in [KKK06, LHK09]. Song et al. [SKWB10] proposed an exploration and preferential return model (the E&P model for short) that mimics the mechanism of exploration, e.g., occasionally visits new locations, and preferential return to frequently visited locations. The E&P model fits well the three aspects mentioned above. It resolves the anomalies arising from the CTRW model.

Meanwhile, my study in Chapter 4 shows that individual’s mobility sequences are highly predictable given historical records. For instance, in mobile phone data the predictability is 93\%, and decreases only slightly to around 85\% with high resolution data [LHL12]. It is therefore highly relevant to see whether the current best known model [SKWB10] can reflect the high predictability. This motivates the first part of this chapter. However, my analysis and empirical study show that the mobility sequences generated from the E&P model exhibit much lower predictability than expected.

To better capture the essence of individual’s mobility, namely, the high predictability given in Chapter 4 and the statistical properties given in [SKWB10], I propose a Markovian mobility model that satisfies these properties. I begin the study by deriving an upper bound on the predictability of the E&P model by analyzing the mobility sequences generated from this model. The general upper bound reveals that there is a significant gap between the predictability of this model and the empirical results from both mobile phone data [SQBB10] and GPS data [LHL12]. In order to bridge the gap, I propose a Markovian model which requires the preferential returns to be conditional on the current location [LHL13b]. I further show that the new model embodies high predictability and preserves the desirable scaling properties of the original model in [SKWB10].

### 5.2 Related Work

The comparisons of major existing trace-based mobility models are presented in Section 2.5. The mobility models described in [RSH+08, LHK+09, HLR10] are incomplete for describing individual’s mobility for the following reason. These three models are constructed based on data collected over walking. Therefore, these models are applicable only to simple cases which cover a small mobility area using only one mode of transport. In reality, individuals employ a variety of modes of transport, and the radius of gyration, which quantifies individual’s mobility area, can be more than 100 km. The mobility model given in [SKWB10] is of great interest here due to two reasons: 1) individual’s general mobility behaviors, namely exploration and preferential return, are extracted from the data, and 2) some statistical properties of individual’s mobility patterns are consistent with the findings in [RSH+08, LHK+09, HLR10].
Chapter 5. A Markovian Mobility Model

Table 5.1: Frequently used symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S$</td>
<td>The number of distinct locations in a mobility sequence</td>
</tr>
<tr>
<td>$s, s_i$</td>
<td>A particular location; the $i^{th}$ distinct location</td>
</tr>
<tr>
<td>$n$</td>
<td>The number of steps in a mobility sequence</td>
</tr>
<tr>
<td>$n_s$</td>
<td>The step that location $s$ is visited for the first time</td>
</tr>
<tr>
<td>$f_i$</td>
<td>The visitation frequency of location $s_i$</td>
</tr>
<tr>
<td>$m(s_i)$</td>
<td>The number of visits to $s_i$</td>
</tr>
<tr>
<td>$\Pi$</td>
<td>The predictability of a mobility sequence</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>Mobility parameter in the E&amp;P model</td>
</tr>
<tr>
<td>$\rho$</td>
<td>Normalizing factor in the E&amp;P model</td>
</tr>
<tr>
<td>$f(s_i</td>
<td>s_j)$</td>
</tr>
<tr>
<td>$m(s_i</td>
<td>s_j)$</td>
</tr>
<tr>
<td>$k(s_i)$</td>
<td>The number of the new locations that are explored for the first time after visiting $s_i$</td>
</tr>
<tr>
<td>$L(s_i</td>
<td>s_j)$</td>
</tr>
</tbody>
</table>

5.3 Preliminary

This study is based on two previous studies, 1) the predictability of human mobility behaviors [SQBB10, LHL12], and 2) the modeling of human mobility [SKWB10]. In the following two subsections, I briefly describe the main ideas of these two studies. Also, the notations used in this chapter are summarized in Table 5.1.

5.3.1 Predictability

Song et al. [SQBB10] defined the predictability of individual human mobility sequence as an upper bound of making a correct prediction by any algorithm in predicting the next move based on historical records. The theoretical study uses the fact that an optimal algorithm always predicts the most probable location according to the historical records. The predictability $\Pi(i)$ of a sequence of length $i$ given historical records $h_{i-1}$ is defined as

$$\Pi(i) \equiv \sum_{h_{i-1}} P(h_{i-1})\pi(h_{i-1})$$ (5.1)

where $P(h_{i-1})$ denotes the probability of observing the historical records $h_{i-1}$, and $\pi(h_{i-1}) = \underset{x}{\text{sup}} P_r(X_i = x|h_{i-1})$ denotes the probability of the most probable location based on the historical records $h_{i-1}$.  

75
Chapter 5. A Markovian Mobility Model

The overall predictability (\( \Pi \)) of a sequence is given by

\[
\Pi = \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} \Pi(i)
\]  
(5.2)

5.3.2 The E&P model

In another study, Song et al. [SKWB10] proposed an exploration and preferential return model (or E&P model for short), which targets on the statistical properties observed from extensive human mobility traces.

Specifically, the E&P model embodies a critical statistical property that allows exploration of new locations as well as returning to familiar locations. In the E&P model, the mobility parameter \( \gamma \), where \( \gamma \in (0, 2] \), controls individuals’ mobility information, e.g., the visitation frequency of each location, and the parameter \( \rho \), where \( \rho \in (0, 1] \), is a normalizing factor. At each step, an individual either

- explores a new location with probability \( P_{\text{new}} = \rho S - \gamma \), where \( S \) denotes the number of distinct locations visited by the individual so far; or
- returns to one of the previously visited locations with complementary probability \( P_{\text{ret}} = 1 - P_{\text{new}} \), where the probability of returning to the \( i^{th} \) location is proportional to its frequency \( f_i \).

Two properties of the E&P model are given

- The number of distinct locations visited during \( n \) steps is \(^2\)
  \[
  S = (1 + \gamma)^{1/(1+\gamma)} (\rho n)^{1/(1+\gamma)}
  \]  
(5.3)

- The visitation frequency of each location is
  \[
  f_i \approx c i^{-(1+\gamma)}
  \]  
(5.4)

where the value of \( c \) is given by \( \gamma/(1 - S^{-\gamma}) \).

In the original paper, the visitation frequency of the \( i^{th} \)-ranked location is also considered for the case when \( \gamma = 0 \). Empirical studies show that the mobility parameter \( \gamma = 0.2 \pm 0.02 \). Thus in this chapter, I will only consider the case of \( \gamma > 0 \). However, when \( i = 1 \) and \( \gamma > 1 \), \( f_1 = \gamma/(1 - S^{-\gamma}) > 1 \). As such, this approximation is inadequate. Considering the importance of the value of \( c \) in deriving the predictability of the E&P model, I offer a better approximation of the value of \( c \) in the following lemma.

---

\(^1\)When \( \gamma \) is larger, more distinct locations are visited during the same number of steps, resulting in a wider spread of visiting frequencies over these locations [SKWB10]. This conflicts with the mobility information obtained from individual's mobility data. Therefore, in this study, I restrict \( \gamma \in (0, 2] \).

\(^2\)In the Supplementary Information [SKWB10], the incorrect solution of \( S \) is \( S = (1 + \gamma)(\rho n)^{1/(1+\gamma)} \).
**Lemma 5.2** For a total of $S$ locations, the frequency of each location is $f_i = ci^{-(1+\gamma)}$, where $1 \leq i \leq S$. A good approximation of $c$ is $q(S)$

$$q(S) = \frac{1}{2} \left[ \frac{\gamma}{1 + \gamma - S^{-\gamma}} + \frac{\gamma}{\gamma + 2^{-\gamma} - (S + 1)^{-\gamma}} \right]$$  \hspace{1cm} (5.5)

The proof and the validation of the approximation are given in Appendix A.3.1.

### 5.4 Predictability of the E&P Model

In this section, the predictability of the E&P model is studied. Let $n_s$ denote the step at which a particular location $s$ is explored for the first time, and $n$ denote the total number of steps. Suppose $n$ is sufficiently large such that the frequency of the $i^{th}$ location is approximated by Eq. (5.4). Let $\Pi(s)$ be the predictability of the model from step $n_s$ to step $n_{s+1} - 1$ based on previous steps. According to Eq. (5.1)

$$\Pi(s) = \frac{1}{n_{s+1} - n_s} \sum_{k=n_s}^{n_{s+1}-1} P(h_{k-1})\pi(h_{k-1})$$  \hspace{1cm} (5.6)

where $h_{k-1}$ denotes the location history at step $k$, and $\pi(h_{k-1})$ is the prediction accuracy of the optimal algorithm that can be achieved by making use of the historical records.

The overall predictability of the model is

$$\Pi = \sum_{i=2}^{S} \frac{n_i - n_{i-1}}{n} \Pi(i)$$  \hspace{1cm} (5.7)

**Theorem 5.1** Given a mobility sequence of length $n$ generated by an E&P model with parameters $\gamma$ and $\rho$, the predictability of the mobility sequence is

$$\Pi \approx \sum_{i=2}^{S} \frac{t^{1+\gamma} - (i - 1)^{1+\gamma}}{S^{1+\gamma}}c(1 - \rho i^{-\gamma})$$  \hspace{1cm} (5.8)

where $S$ is the number of distinct locations observed from the sequence given in Eq. (5.3), and $c$ is given in Eq. (5.5).

**Proof:** The optimal algorithm for predicting the next move according to the previously generated mobility sequences from the E&P model is considered. In the E&P model, individuals would prefer to return to previous locations rather than explore new locations. According to the theory in [SQBB10], an optimal algorithm $A_{opt}$ should always predict...
the most probable location. By the definition of \( n_s \), because it is impossible for \( A_{\text{opt}} \) to correctly predict the newly visited location, the prediction accuracy at the step \( n_s \) is \( \pi(h_{n_s}) = 0 \). Between step \( n_s + 1 \) and step \( n_{s+1} - 1 \), since the location with the highest frequency is always \( s_1 \), i.e., the first location, according to Eq. (5.4)

\[
\pi(h_{k-1}) = \sup_x \Pr(X_k = x|h_{k-1}) = f_1 = c
\]

where \( k \in [n_s + 1, n_{s+1} - 1] \).

The predictability of the sequence from step \( n_s \) to \( n_{s+1} - 1 \) based on previously generated sequence from step 1 to step \( n_s - 1 \) is

\[
\Pi(s) \equiv \frac{1}{n_{s+1} - n_s} \sum_{k=n_s}^{n_{s+1}-1} P(h_{k-1})\pi(h_{k-1})
\]

\[
= \frac{1}{n_{s+1} - n_s} \left[ \sum_{h_{n_s-1}}^{n_{s+1}-1} P(h_{n_s-1})\pi(h_{n_s-1}) + \sum_{k=n_s+1}^{n_{s+1}-1} \sum_{h_{k-1}}^{n_{s+1}-1} P(h_{k-1})\pi(h_{k-1}) \right]
\]

\[
= \frac{1}{n_{s+1} - n_s} \left[ 0 + \sum_{k=n_s+1}^{n_{s+1}-1} c \sum_{h_{k-1}}^{n_{s+1}-1} P(h_{k-1}) \right]
\]

\[
= \frac{1}{n_{s+1} - n_s} \left[ 0 + \sum_{k=n_s+1}^{n_{s+1}-1} cP_{\text{ret}} \right]
\]

\[
= \frac{1}{n_{s+1} - n_s} (n_{s+1} - n_s - 1)P_{\text{ret}}c
\]

\[
\approx c(1 - \rho S^{-\gamma})
\]

Eq. (5.12) follows from Eq. (5.11) by considering the following two cases 1) at step \( n_s \) and 2) the steps between \( n_s + 1 \) and \( n_{s+1} - 1 \). Eq. (5.13) follows from Eq. (5.12) due to the prediction accuracy \( \pi(h_{n_s-1}) = 0 \), and for the remaining cases \( \pi(h_{k-1}) = c \) according to Eq. (5.10). Eq. (5.14) follows from Eq. (5.13) due to the observing probability \( \sum_{h_{k-1}} P(h_{k-1}) = P_{\text{ret}} \). Eq. (5.15) follows from Eq. (5.14) because the value \( cP_{\text{ret}} \) is independent of each step for \( k \in [n_s + 1, n_{s+1} - 1] \). Eq. (5.16) is based on the assumption that \( n_{s+1} - n_s \) is very large.

Since the difference \( n_{s+1} - n_s \) varies with each new location, the overall predictability
is

$$\Pi = \sum_{i=2}^{S} \frac{n_{i} - n_{i-1}}{n} \Pi(i)$$  \hspace{1cm} (5.17)

$$= \sum_{i=2}^{S} \frac{i^{1+\gamma} - (i-1)^{1+\gamma}}{i^{1+\gamma}} \Pi(i)$$  \hspace{1cm} (5.18)

$$\approx \sum_{i=2}^{S} \frac{i^{1+\gamma} - (i-1)^{1+\gamma}}{i^{1+\gamma}} c(1 - \rho^{i^{1+\gamma}})$$  \hspace{1cm} (5.19)

Eq. (5.18) follows from Eq. (5.17) by using the result in [SKWB10] (Supplementary Information p.11) that $n_{i} \sim i^{1+\gamma}$. □

By considering the different cases based on the values of the mobility parameter $\gamma$ and the normalizing factor $\rho$, the following lemma is to bound the predictability of the E&P model given by Eq. (5.19).

**Lemma 5.3** The predictability of the E&P model is bounded in the following cases:

\[
\Pi < \begin{cases} 
1 - \rho^{2-\gamma}, & \text{Case 1: } \rho \in (0, \frac{1}{1+\gamma}] \\
\frac{2}{\gamma+2-\gamma} + \frac{1}{2} \left[ 1 - \frac{\gamma+2-\gamma}{i+\gamma} \right], & \text{Case 2: } \rho \in \left( \frac{1}{1+\gamma}, \frac{1}{\gamma+2-\gamma} \right) \\
\min \left\{ \frac{2}{\gamma+2-\gamma}, \frac{1}{2}, \frac{\gamma}{2(\gamma+2-\gamma)} \right\}, & \text{Case 3: } \rho \in \left[ \frac{1}{\gamma+2-\gamma}, 1 \right] 
\end{cases}
\]  \hspace{1cm} (5.20)

where $\gamma \in (0, 2]$.

The proof is given in Appendix A.3.2. Two remarks of this lemma are made here.

**Remark 1.** Only in Case 1, the upper bound is related to $\rho$. For the remaining cases, the bounds only depend on the mobility parameter $\gamma$. When $\gamma$ is chosen from a range that corresponds to the possible number of distinct locations\(^3\), the normalizing factor $\rho$ mostly lies in the range of Case 1. According to the empirical study from both mobile phone data and GPS data, the number of distinct locations varies between 30 and 1000, indicating that for most individuals the mobility parameter $\gamma \in [0.2, 0.5]$. For instance, when $\gamma = 0.2$, in Case 1 the range of $\rho$ is $(0, 0.83]$, and in Case 2 and Case 3 are $(0.83, 0.93)$ and $[0.93, 1]$, respectively. Therefore, the predictability remains largely constant for the relatively narrow range of $\rho$ in Case 2 and Case 3, and the corresponding upper bound is constant as well for a given mobility parameter $\gamma$.

**Remark 2.** The bound in Case 3 is chosen from the minimum value of the two terms. When the mobility parameter $\gamma < 1.7$, which corresponds to the vast majority of all the cases, the bound can be simplified to $\Pi < \frac{2}{\gamma+2-\gamma}$, as determined by the first term.

---

\(^3\)The number of distinct locations can be estimated from the individual’s historical data.
I compare the derived upper bound of the predictability of the E&P model with the empirical results. By fitting curves to the data obtained from real human trajectories, Song et al. [SKWB10] show that the mobility parameter $\gamma = 0.21 \pm 0.02$ and $\langle \rho \rangle = 0.6$, respectively. Using this range of values for the parameters of $\gamma$ and $\rho$, the predictability of the E&P model is bounded in Case 1 by $\Pi < 1 - \rho 2^{-\gamma}$, or $\Pi < 1 - 0.6 \times 2^{-0.21} = 48.1\%$. However, the empirical studies based on both mobile phone data [SQBB10] and GPS data [LHL12] indicate that the actual, achievable predictability of human mobility is around 90%. Even when shrinking the size of each location, the predictability can still be around 85% in GPS data [LHL12]. Therefore, the E&P model fails to capture the high predictability of human mobility sequence.

5.5 A New Markovian Model

Here I propose a Markovian model that aims to resolve the predictability issue while preserving the desirable properties of the E&P model in [SKWB10]. As observed in [SKWB10], there may exist certain spatial correlation in human mobility leading to the high predictability of human mobility sequences. Moreover, when revisiting each location, the individuals may make a choice according to the current context including the present location. Therefore, individuals may choose the next location conditionally according to the current location rather than choosing from the globally most frequent locations. In order to preserve the original scaling properties, I will retain the rule of exploration.

Specifically, in the Markovian model, the individual may explore new locations with probability $P_{\text{new}}$ subject to the number of distinct locations visited so far,

$$P_{\text{new}} = \rho S^{-\gamma}$$

as with the E&P model.

However, when the individual is at the current location $s_i$, the probability of returning to a location $s_j$ shall be governed by

$$P(s_j|s_i) = P_{\text{ret}} f(s_j|s_i)$$

where the return probability $P_{\text{ret}} = 1 - P_{\text{new}}$. $f(s_j|s_i)$ denotes the conditional probability of observing $s_j$ given the current location $s_i$, where $1 \leq j \leq S$.

5.5.1 Properties of the new Markovian model

In this section, the properties of the new model are analyzed. Let $k(s_i)$ be the number of the new locations that are explored for the first time after visiting location $s_i$. Suppose the current location is $s_i$. Let $L(s_j|s_i)$ be the rank of the location $s_j$ observed immediately
after location $s_i$. The number of visits to location $s_j$ immediately following location $s_i$ is denoted by $m(s_j|s_i)$, and the corresponding probability is $f(s_j|s_i)$. Therefore

$$m(s_i) = \sum_j m(s_j|s_i)$$ (5.23)

$$1 = \sum_j f(s_j|s_i)$$ (5.24)

**Lemma 5.4** For the same parameters in the new Markovian model as that in the E&P model, the number of distinct locations generated in the two models is the same.

Since the manner of exploring new locations is the same in the two models, the number of distinct locations in both models is given by Eq. (5.3).

**Lemma 5.5** The visitation frequency of each location in the new model shares the same Zipf-law distribution of the E&P model, i.e., $f_i \sim i^{-(1+\gamma)}$.

**Proof:** Note that the number of visits of $s_i$, denoted by $m(s_i)$, satisfies

$$\frac{dm(s_i)}{dn} = (1 - P_{new}) \sum_{j=1}^{S} f_j \frac{m(s_i|s_j)}{m(s_j)}$$ (5.25)

Since $f_j = \frac{m(s_j)}{n}$, $f_j \frac{m(s_i|s_j)}{m(s_j)} = \frac{m(s_i|s_j)}{n}$. Also, $\sum_{j=1}^{S} m(s_i|s_j)$ counts all the observations of $s_j$, we have

$$\frac{dm(s_i)}{dn} = (1 - P_{new}) \frac{m(s_i)}{n}$$ (5.26)

which is exactly the differential equation of $m(s_i)$ of the E&P model. □

The preceding two lemmas indicate that the two fundamental properties, i.e., the number of distinct locations and the visitation frequency, are preserved in the new model. In addition, the following two properties, presented in Lemma 5.6 and 5.7, are further observed in the new model.

**Lemma 5.6** For the new model,

$$k(s_i) \sim i^{-(1+\gamma)}$$ (5.27)
Chapter 5. A Markovian Mobility Model

Proof: Recall that, \( k(s_i) \) denote the number of the new locations that are explored for the first time after visiting \( s_i \)

\[
\frac{dk(s_i)}{dn} = f_i P_{\text{new}} \tag{5.28}
\]

Following the chain rule of derivatives and \( \frac{dS}{dn} = P_{\text{new}} \) [SKWB10],

\[
\frac{dk(s_i)}{dn} = \frac{dk(s_i)}{dS} \frac{dS}{dn} \tag{5.29}
\]

Combining Eq. (5.28) and Eq. (5.29), \( \frac{dk(s_i)}{dn} = f_i \). Thus in the new model, \( k(s_i) \sim i^{-(1+\gamma)} \). □

Eq (5.27) shows that \( k(s_i) \) follows a Zipf-law distribution, indicating that the sooner a location is explored the more number of new locations are subsequently visited after this location.

Lemma 5.7

\[
f(s_j|s_i) \sim L(s_j|s_i)^{-(1+\gamma)} \tag{5.30}
\]

The derivation of the conditional frequency of each location after a certain location is similar to the derivation of the overall frequency of each location in the E&P model. Specifically, the conditional frequency of each location after \( s_i \) is \( f(s_j|s_i) \approx c'_i L(s_j|s_i)^{-(1+\gamma)} \), where \( c'_i = q(k(s_i) + 1) \) for \( i > 1 \), and \( c'_1 = q(k(s_1)) \). The function \( q(x) \) is given in Eq. (5.5).

This lemma indicates that the conditional frequency also follows a Zipf-law distribution. Therefore, a similar approximation for the conditional frequency is

\[
f(s_j|s_i) \approx c'_i L(s_j|s_i)^{-(1+\gamma)} \tag{5.31}
\]

where \( c'_i \) is

\[
c'_i \approx \frac{1}{2} \left[ \frac{\gamma}{1 + \gamma - k(s_i)^{-\gamma}} + \frac{\gamma}{\gamma + 2^{-\gamma} - (k(s_i) + 1)^{-\gamma}} \right] \tag{5.32}
\]

5.5.2 Predictability of the new Markovian model

Now, I show that the predictability of the Markovian model is strictly higher than that in the E&P model. For the Markovian model, I consider the optimal prediction algorithm by making use of the historical records. The historical records are applied to make a prediction in two ways: 1) by using the global frequency as in the E&P model, and 2) by using the conditional frequency as in the new model. In the first case, the prediction
accuracy is $\pi(h_k) = c$ when $k \in [n_s + 1, n_{s+1} - 1]$, and $\pi(h_{ns}) = 0$, as shown in the E&P model. In the second case, when applying the local, conditional information at current location $s_i$, the prediction accuracy of an optimal algorithm is

$$\pi(X|h_k = s_i) = \max_j f(s_j|s_i)$$

$$= \max_j c'_i L(s_j|s_i)^{-1+\gamma}$$

$$= c'_i$$

where $L(s_j|s_i) = 1$ and $k \in [n_s + 1, n_{s+1} - 1]$.

Note that $c'_i = q(k(s_i) + 1)$ and $c = q(S)$ given in Eq. (5.5). Also $q(x)$ decreases when $x$ increases, and even $(\max_j k(s_i) + 1) < S$, thus $c'_i > c$. This indicates that using the local, conditional information can achieve higher prediction accuracy than using the global, unconditional information. Based on this observation, the predictability of the Markovian model is shown to be strictly higher than that of the E&P model. Specifically, in the Markovian model, the overall predictability attributed to the subsequence between step $n_s$ and $n_{s+1} - 1$ is

$$\Pi(s) = \frac{1}{n_{s+1} - n_s} \sum_{k=n_s}^{n_{s+1} - 1} \sum_{h=k-1}^{n_{s+1} - 1} P(h_{k-1}) \pi(h_{k-1})$$

$$= \frac{1}{n_{s+1} - n_s} \sum_{k=n_s+1}^{n_{s+1} - 1} \sum_{i=1}^{S} P_{ret} f_i \pi(X|h_{k-1} = s_i)$$

$$= \frac{1}{n_{s+1} - n_s} \sum_{k=n_s+1}^{n_{s+1} - 1} P_{ret} \sum_{i=1}^{S} f_i c'_i$$

$$> \frac{1}{n_{s+1} - n_s} \sum_{k=n_s+1}^{n_{s+1} - 1} P_{ret} \sum_{i=1}^{S} f_i c$$

$$= \frac{1}{n_{s+1} - n_s} \sum_{k=n_s+1}^{n_{s+1} - 1} c P_{ret}$$

$$= \pi(s)$$

Eq. (5.37) follows from Eq. (5.36) since from step $n_s + 1$ to step $n_{s+1} - 1$ the location history $h_{k-1}$ at each step is the current location for the Markovian model, and the probability of observing each location history or current location is the product of the frequency of current location and the return probability. Also at step $n_s$, $\pi(h_{k-1}) = 0$, therefore the index in the following equation is from $n_s + 1$ to $n_{s+1} - 1$. Eq. (5.38) follows from Eq. (5.37) since the algorithm predicts by using local frequency. Ineq. (5.39) follows from Eq. (5.38) since $c'_i > c$. Eq. (5.40) follows from Ineq. (5.39) since the sum of the visitation frequency of all the locations equals to 1. The last step is the definition of the predictability in the E&P model.
5.6 Experimental Results

In this section, I present the experimental results validating the theoretical study on mobility models in the following respects: 1) the upper bound of the predictability of the E&P model, 2) the statistical properties, e.g., the visitation frequency, the distribution of the conditionally visited new locations, and conditional frequency, etc., in the new Markovian model, and 3) the comparative results of the predictability in the two models.

5.6.1 The bound of the predictability

In this subsection, the experiments are conducted in the following manner.

- \( \gamma \) is chosen from 0.2 to 1.0 at increments of 0.1.

- For a given value of \( \gamma \), the range of \( \rho \) is chosen according to the three different cases in Lemma 5.3.

- Within the range of \( \rho \) in the three different cases, 9 different values of \( \rho \) are tested, which are evenly spaced within the range.

- For a given pair of \( \gamma \) and \( \rho \), a sequence of 2500 steps is randomly generated 1000 times. For each mobility sequence, the number of distinct locations, entropy rate, and the predictability are calculated [SQBB10, LHL12]. The predictability of the model under the given pair of \( \gamma \) and \( \rho \) is approximated by the mean predictability from the randomly generated sequences.

Note that, experimental results in [SQBB10] show in mobile phone data, the number of distinct locations is mostly ranging from \( 2^5 = 32 \) to \( 2^8 = 256 \) given a trajectory of length 2352, or in other words an hourly sampled 14-weeks’ trajectory. Therefore, for the ease of comparison I choose the number of steps to be 2500, and the results are plotted only when the number of distinct locations\(^4\) are within the range of \([30, 1000]\).

In Figure 5.1, the plots of the bounds are clearly showed in the three cases according to the different value of \( \rho \). In Case 1, both the bound of predictability and the predictability decreases when increasing \( \rho \). While, in the remaining two cases, the bounds are a straight line, which is independent of \( \rho \). In all the cases, the derived bounds of predictability is strictly higher than the predictability in the E&P model, indicating a general upper bound is achieved.

\(^4\)I relax the upper bound of the number of distinct locations. The reason is that in mobile phone data using in [SKWB10], the average size of each location is roughly 3 km\(^2\), which is relatively large in reality. Therefore, extending the number of distinct locations to 1000 will enclose more cases, e.g., the trajectory recorded by GPS data [LHL12].
Figure 5.1: The bounds for predictability are combined in the three cases. Under each $\gamma$, the value of $\rho$ is chosen from $(0, 1]$. In each figure, “predOrg” denotes the predictability of the mobility sequences based on the E&P model. “bound” denotes the corresponding upper bound of the predictability of the E&P model. “predNew” denotes the predictability of the mobility sequences based on the new model.
5.6.2 The statistical properties in the new model

In this subsection, I validate the statistical properties, given from Lemma 5.5 to Lemma 5.7, in the Markovian model. In these cases, the number of steps is much longer than the previous experiments, since all the lemmas refer to the asymptotic properties of the new model. Therefore, the experiments are conducted in the following manners.

- The number of steps is \( n = 50000 \).
- \( \gamma \) is chosen from 0.2 to 0.5 at increments of 0.1.
- For a given value of \( \gamma \), the maximum value of \( \rho \) is \( 1/(1 + \gamma) \) according to the functional form \( m(s_i) = cnL(s_i)^{1+\gamma} \).

Figure 5.2: The plot of the visitation frequency of each location against the rank of being explored in the new model. The function \( m(s_i) = cnL(s_i)^{1+\gamma} \) shows a good fit of the plot, where \( L(s_i) = i \).

- The number of steps is \( n = 50000 \).
- \( \gamma \) is chosen from 0.2 to 0.5 at increments of 0.1.
- For a given value of \( \gamma \), the maximum value of \( \rho \) is \( 1/(1 + \gamma) \) according to the
Figure 5.3: The distribution of $k(s_i)$.
Chapter 5. A Markovian Mobility Model

range\(^5\) in Case 1. The 4 different values of \(\rho\) are evenly spaced in the interval of \((0, 1/(1 + \gamma))\).

- For a given pair of \(\gamma\) and \(\rho\), the sequence is randomly generated 100 times, where the statistical properties validated in this section are the mean results obtained from these sequences.

The overall visitation frequency of the locations plotted according to their rank of being explored is shown in Figure 5.2. In the case of each \(\gamma\), the plot of the frequency in each chosen \(\rho\) is almost parallel to the fitted curve, indicating that the frequency of each location is best fitted by \(m(s_i) = cn^{(1+\gamma)}\), which is independent of \(\rho\). This property matches well the Zipf-law visitation frequency reported in the E&P model.

Figure 5.3 shows the distributions of \(k(s_i)\) in different cases, where the fitted curve is \(k(s_i) \sim L(s_i)^{-(1+\gamma)}\). Two observations can be made from this figure. In the cases of \(\gamma = 0.2\) and 0.3, the scatter plots are dense and best fitted by \(k(s_i) \sim L(s_i)^{-(1+\gamma)}\). For different \(\rho\), the trend of the scatter plots are almost parallel, indicating similar slope for each plot. However, in the cases of \(\gamma = 0.4\) and 0.5, the scatter plots are sparse and slight deviate from the straight line with slope \(-(1 + \gamma)\). This is due to the fact that when increasing \(\gamma\), the number of distinct locations decreases sharply, leading to biased results. For instance, when \(\gamma = 0.2\) or 0.3, and \(\rho = 0.15\), \(n = 50000\), the number of distinct locations is 1973 and 1171 according to Eq. (5.3), respectively. While increasing \(\gamma\) to 0.4 or 0.5, and keeping the remaining parameters unchanged, the number of distinct locations is 745 and 502, respectively. Also in the new model, except for the first few cases, \(k(s_i)\) is very small, e.g., less than 100, indicating \(k(s_i) \ll S\) and further serves as an evidence to support that \(c'_i > c\).

Figure 5.4 compares the conditional frequency based on the first location \(s_1\) and the second location \(s_2\) in the two models with \(\gamma = 0.2\) and 0.3. In the new model, the conditional frequency based on \(s_1\), in Figure 5.4 (a) and Figure 5.4 (c), follows the Zipf-law distribution and the slope is \(-(1 + \gamma)\) in all these cases. The plots are parallel for different \(\rho\) with the same \(\gamma\). Also, the number of locations observed conditionally on the first location is normally around 200, indicated by the maximum value of \(L(s_i | s_1)\). In comparison, from Figure 5.4 (b) and Figure 5.4 (d), the conditional frequency in the E&P model follows a Zipf-law distribution. However, it deviates from the straight line with a slope of \(-(1 + \gamma)\). More importantly, there are two major differences between the conditional frequency in the E&P model and the Markovian model. Firstly, the number of observed locations in the E&P model is much larger than that in the Markovian model, since the maximum value of \(L(s_i | s_1)\) is close to 1000. Secondly, the conditional

\(^5\)Note that when \(\gamma\) is very small, e.g. from 0.2 to 0.5, \(\rho\) is most probable within the range in Case 1 according to the real cases.
Figure 5.4: The comparison of the conditional frequency based on the first location $s_1$ and second location $s_2$ for different values of $\gamma$ and $\rho$. Figure (a) and (b) show the distributions of the conditional frequency based on $s_1$ given $\gamma = 0.2$, while (c) and (d) show the distributions of the conditional frequency based on $s_1$ given $\gamma = 0.3$. Similarly, (e) and (f) show the distributions of the conditional frequency based on $s_1$ given $\gamma = 0.2$, while (g) and (h) show the distributions of the conditional frequency based on $s_2$ given $\gamma = 0.3$. 

\[ \rho \in [0.15, 0.17, 0.3, 0.5, 0.6] \]
frequency in the E&P model also exhibits a fat-tailed shape, indicating a large amount of conditionally visited new locations are of low frequency. These two observations further demonstrate that the new model preserves the Zifp-law frequency in a local fashion making the generated sequences to exhibit high predictability. Similar observations of the conditional frequency based on the second location $s_2$ can be found in Figure 5.4 (e), (f), (g), (h).

5.6.3 The comparison of predictability in the two models

In this subsection, I compare the predictability of the mobility sequences generated in the two models in a general setting. In this part, the number of steps is 2500. The choices of the parameters $\rho$ and $\gamma$ follow the details in Section 5.6.1. Figure 5.1 verifies that the predictability of the Markovian model is strictly higher than that of the E&P model. In all the cases the predictability obtained from the new model is much higher than the corresponding value from the E&P model. In the E&P model, the predictability drastically decreases when increasing $\rho$ for a given $\gamma$. However, in the Markovian model the predictability changes slightly when increasing $\rho$, even when $\rho$ is very large. Also, when $\gamma = 0.2$ and $\rho$ is between 0.2 and 0.8, the predictability of the Markovian model ranges between 65% and 80%, which greatly increases the corresponding predictability values of the E&P model.

Figure 5.5: (a) A comparison of the predictability of the mobility sequences generated from the Markovian model and the model with the rule of return akin to the second-order Markov chain. (b) A Zipf’s law distribution with an exponent of 1.2, indicated by the solid line, is fitted according to the data generated with $\rho = 0.13$. The details of the visits of the first 7 locations of the case $\rho = 0.13$ and the fitted curve are shown in the inset.

Note that there is still a small gap between the predictability of the Markovian model, which ranges between 65% and 80%, and the empirical results, which are mostly around
90%. The question is whether it is possible to further increase the predictability, while preserving the properties given in Lemma 5.5, Lemma 5.6, and Lemma 5.7. Theoretically, high predictability corresponds to the high redundancy of the mobility sequences [LHL12], which may be achieved by revisiting the sequence of locations in the same order. By requiring the return probability to be conditional on the current location, the Markovian model presented in this chapter is akin to the first-order Markov model.

In this subsection, I also evaluate a new second-order mobility model. This new model retains the characteristic of exploring new locations in both the E&P model and the Markovian model, and the manner of return is akin to the second-order Markov model. Two key statistical properties of the second-order Markovian model are evaluated, namely predictability and the visitation frequency of the locations. Figure 5.5 (a) shows the predictability of the mobility sequences generated by the second-order Markovian model, with the mobility parameter $\gamma$ values of 0.2 and 0.5, and the choices of the remaining parameters are the same as those in Figure 5.1. When comparing the predictability generated by the first-order Markovian model and that of the second-order Markovian model, Figure 5.5 (a) shows that the predictability of the mobility sequences generated by the second-order Markovian model is much higher in certain cases, e.g., 90% when $\gamma = 0.5$. Meanwhile, Figure 5.5 (b) further shows the visitation frequency of each location given in the second-order Markovian model, where the setup of the parameters is the same as that in Figure 5.2. In such cases, however, the frequencies do not seem to fit the Zipf’s law distribution. This is because the first few locations accumulate a large number of visits (above the solid line), leading to much lower number of visits to the other locations than predicted by the Zipf’s law distribution (see the inset of Figure 5.5 (b)). This validates that the proposed new Markovian model (return follows by the first-order Markov model) is actually more appropriate because it embodies high predictability while preserving the desirable scaling properties of the E&P model.

## 5.7 Chapter Summary

In this chapter, I first studied the predictability of the mobility sequences generated from the well-known E&P human mobility model. I showed that the predictability of the E&P model is less than 50% in most cases, while the empirical results of the predictability of individual’s mobility sequences is around 90%. Thus, I had shown that the E&P model is inadequate for describing individual’s mobility. Therefore, I presented a new Markovian mobility model which preserves the desirable scaling properties uncovered in [SKWB10]. The new model better reflects the high predictability exhibited in real mobility data. Additional properties of individual’s mobility sequences are validated in Appendix A.2.
Chapter 6

Location Prediction at High Spatio-Temporal Resolution

In this chapter, I present a study of predicting individual’s next locations at high spatio-temporal resolution.

6.1 Location Prediction: An Introduction

With the wide availability of GPS devices, current location-based services and applications already have high spatial- and temporal-resolution requirements for predicting individual’s future locations [GG03]. These requirements pose a nontrivial challenge to location predictions. For instance, in pervasive or mobile computing, the spatial resolution expected of a prediction is in the order of 10 meters and the temporal resolution can be as high as a few seconds.

In Chapter 4, both theoretical and experimental studies show that individual’s next locations are highly predictable, which is around 90%, in GPS data. However, this study is incomplete in two aspects. Firstly, the predicted results are still too coarse in terms of both spatial and temporal resolutions for the location-based services. Specifically, the temporal resolution in the earlier context is one hour and the spatial resolution is around 500 meters [LHL12]. Secondly, the high predictability over the entire 16 weeks may be dominated by the cases where the individuals stay at home or in the office for most of the time. However, in certain cases, such as, when making moves, it may be difficult to predict individual’s next location at high spatio-temporal resolutions.

In this chapter, I resolve this issue by presenting a location prediction method at high spatio-temporal resolution [LH14]. Given a set of historical positioning records of an individual and a sequence of samples in the current trace, where all positioning records are GPS readings with a fixed sampling rate, e.g., 60 seconds, the problem is to predict this individual’s location in the near future, e.g., 60 seconds, 120 seconds, or...
180 seconds later. To achieve such purpose, a location prediction method makes use of historical records while observing current mobility behaviors. Since previous studies suggest that there is a high probability that the individuals will follow previous traces [SQBB10, LHL12], the current trace may be similar to some traces in the past. Therefore, the edit distance is applied to measure the similarity between traces. The similarity is evaluated by the sum of the cost of applying three types of editing operations, i.e., deleting, inserting, changing, on the two traces. If the edit distance is within a given threshold, whose value is chosen empirically, the two traces are considered similar.

However, when using similar traces to estimate the future location, there may not be a record that corresponds exactly to the required point of time, and hence there is a need to reflect the uncertainties associated with the individual’s movements in between two sampled positions. Therefore, a Brownian Bridge model [Shr04] is proposed to help model the predicted location in between two existing sample points. The parameter of the Brownian Bridge model is estimated from the historical traces. The location at any time is modeled as a Gaussian distribution to cater for the measurement error associated with GPS readings. A linear extrapolation method is also used to model the individual’s current mobility behaviors. The predicted location is a combination of the results from the Brownian Bridge models and the result from the linear extrapolation method.

In order to evaluate the location prediction method, the method is compared to two baseline methods, namely, the linear extrapolation method and a method that only uses similar traces, in predicting the location at 60 seconds later. In terms of average prediction errors, the experimental results show that the prediction method by using Brownian Bridge model is much better than either method that uses only linear extrapolation or similar traces alone. Also, in predicting the location for a further point in future, e.g., 120 seconds or 180 seconds into the future, the mean prediction errors in my method are much lower than the linear extrapolation method. Moreover, the low standard deviations of the predicted locations from my method indicate the reliable prediction using the Brownian Bridge model.

My contributions consist of in a few parts. Firstly, I propose and validate the edit distance for measuring the similarity between mobility traces. Secondly, I present and evaluate the Brownian Bridge model for modeling a person’s movements in between any two sample points. Thirdly, I present a prediction method that is capable of high spatio-temporal resolution prediction by exploiting the individual’s mobility behaviors in the current trace and historical records.

The remainder of this chapter is organized as follows. Section 6.2 reviews existing method for measuring trace similarity and also the location prediction. Section 6.3 compares three different edit distances. Section 6.4 presents my method for measuring trace similarity. Section 6.5 presents the proposed location prediction methods. Section 6.6 shows the experimental results. Section 6.7 concludes this chapter.
6.2 Related Work

In this section, I review existing results on measuring trace similarity and location prediction.

6.2.1 Measuring trace similarities

The existing methods for measuring similarity between two sequences of positioning samples can be classified into two groups. One group measures the similarity by directly calculating the distance between pairs of points from the given sequences. For instance, the Hausdorff distance [Hen99] is the maximum of the nearest distance between any pair of points formed by taking one point from each trace. Longest Common Subsequences [VKG02] measures the similarity by observing the maximal overlap of the two sequences. However, these two methods are not suitable for the GPS traces due to the following reasons. Although, the Hausdorff distance is suitable for measuring the distance in between two sets of points, it does not consider the order of these points, which is fundamental for GPS traces. The Longest Common Subsequences may be greatly affected by the sampled data, e.g., two traces collected on the same route may show no positional similarity due to a systematic shift in the sampled positions. Therefore, I do not consider these two methods for measuring the distance of the traces in my study. Another group of the methods for measuring similarity is based on the notion of edit distance, e.g., dynamic time warping (DTW) [KPC01], edit distance with real penalty (ERP) [CN04], and edit distance on real-valued sequences (EDR) [COO05]. Methods in this group apply a series of editing operations (to be explained shortly) in order to convert one sequence to another, and the distance of the two sequences is the sum of the cost of applying these operations. One major advantage of edit distance is that, they can tolerate a degree of inexactness in the sequence and are able to handle the cases of partial recording and shifted samples. I will compare these edit distance measures in Section 6.3.

6.2.2 Location prediction

The existing approaches for location prediction can be classified into two groups. In the first group, individual’s mobility history is constructed based on either significant locations [AS03, KH06] or region-of-interest [GNPP07], both of which are generated by applying a clustering method, such as DBSCAN [EpKSX96] or K-Means [HK06], on historical records from either the given person or a large population. Then, according to the transition records between the significant locations or region-of-interest, a probabilistic model based on either Bayesian theory or Markovian theory is constructed to infer the next location. A major issue here is that, the constructed probabilistic models can
be applied to infer the next locations only when individuals visit the significant location; otherwise the model fails. Also, in these studies, the transition time between two locations is generally disregarded.

In the second group of approaches, the next move is predicted entirely based on trajectory patterns [CLC10, MPTG09]. In such cases, the trajectories are represented as spatio-temporal items, e.g., a list of locations and the corresponding transition times. The current trace is matched to the historical records by searching through the pattern tree. Two methods [CLC10, MPTG09] that make use of the trajectory patterns alone typically suffer from the major limitation in the timing allowed for making predictions, because of the tight coupling between the prediction time and the sampling rate. Similarly, the trajectory pattern mining method is only applicable to inferring the location at a point in time for which the corresponding sample is available in the matched traces; predicting the location at any other time in between the sampling method is impossible.

6.3 Edit Distance

In this section, I present a general description of the edit distance, and also compare a few popular edit distance measures. Before discussing the edit distance in details, I introduce some notations below.

Let \( S^n = s_1s_2, ..., s_n \) be a sequence of length \( n \), where each symbol \( s_i \) indicates a location with exact x and y coordinates, which are converted from the latitude and longitude readings with respect to the origin. The subsequence from index \( i \) to index \( j \) is denoted by \( S^{ij} \), where \( 1 \leq i < j \leq n \). Similarly, let \( R^m = r_1r_2, ..., r_m \) be a sequence of length \( m \), where each symbol \( r_i \) indicates a location with exact x and y coordinates.

6.3.1 The general edit distance

Generally, there are three operations in edit distance [LBC98].

**Definition 6.1** Given a sequence \( S^n = s_1s_2, ..., s_n \), three operations are defined as follows:

- **Inserting** a location symbol \( s_c \) at position \( i \): \( I(S^n, s_c, i) \), i.e., \( I(S^n, s_c, i) \equiv s_1, ..., s_{i-1}s_cs_i, ..., s_n \).

- **Deleting** a location symbol at position \( i \): \( D(S^n, i) \), i.e., \( D(S^n, i) \equiv s_1, ..., s_{i-1}s_{i+1}, ..., s_n \).

- **Changing** a location symbol at index \( i \) to \( s_c \): \( C(S^n, i, s_c) \), i.e., \( C(S^n, i, s_c) \equiv s_1, ..., s_{i-1}s_cs_{i+1}, ..., s_n \).
Generally, the cost of each operation is constant.

- The cost of insertion is: \( w_I(S^n, s, i) = c_1 \)
- The cost of deletion is: \( w_D(S^n, i) = c_2 \)
- The cost of changing a symbol is: \( w_C(S^n, i, s_c) = c_3 \)

where \( c_1, c_2, \) and \( c_3 \) denote constant real numbers.

**Remark.** Generally, insertion and deletion have the same cost, i.e., \( c_1 = c_2 \). The cost of changing a symbol is normally larger than insertion or deletion, such that \( c_3 > c_1 = c_2 \).

**Definition 6.2** The distance between two sequences \( S^n \) and \( R^m \) is the minimum cost for applying a sequence of operations in Definition 6.1 to convert one of the sequence to another sequence.

**Remark.** When the cost of insertion and that of deletion are the same, the cost of converting \( S^n \) to \( R^m \) is the same as converting \( R^m \) to \( S^n \) since inserting a location symbol in \( S^n \) corresponds to deleting a location symbol at certain position in \( R^m \). However, when the cost of deletion and that of insertion are not the same, the distance between two sequences is the cost of converting \( S^n \) to \( R^m \) or vice versa, whichever is lower. A few examples are given in the following subsection.

### 6.3.2 Three specific edit distances

In this subsection, I discuss three well known edit distances, namely, dynamic time warping (DTW) [KPC01], edit distance with real penalty (ERP) [CN04], and edit distance on real-valued sequences (EDR) [COO05]. These schemes are highlighted in two respects, 1) the base condition, i.e., when either sequence is empty, 2) the routine for calculating the distance. Note that in these edit distances, applying each operation corresponds to introducing a gap location. I will explain this notion in the context of edit distance.

Firstly, depending on specific scenarios, the three edit distances may differ greatly in measuring the cost according to each operation. In DTW, the cost between two locations is

\[
d_{dtw}(r_i, s_j) = \begin{cases} 
 d_e(r_i, s_j), & \text{if } r_i \text{ and } s_j \text{ are not gaps}, \\
 d_e(r_i, s_{j-1}), & \text{if } s_j \text{ is a gap}, \\
 d_e(r_{i-1}, s_j), & \text{if } r_i \text{ is a gap}, 
\end{cases}
\]

where \( d_e(r_i, s_j) \) denotes the Euclidean distance between two locations \( r_i \) and \( s_j \).

(6.1)
Table 6.1: The base condition of each edit distance. Note that in the original EDR method, the base cost given either \( m = 0 \) or \( n = 0 \) is \( n \) and \( m \), respectively. In my case, where the cost is measured in terms of Euclidean distance, the base cost in EDR is multiplied by the distance threshold \( \delta \).

<table>
<thead>
<tr>
<th>Edit Distance</th>
<th>( m=0 )</th>
<th>( n=0 )</th>
<th>( m=0, n=0 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTW ((R^m, S^n))</td>
<td>( \infty )</td>
<td>( \infty )</td>
<td>0</td>
</tr>
<tr>
<td>ERP ((R^m, S^n))</td>
<td>( \sum_{i=1}^{m} d_e(s_i, g) )</td>
<td>( \sum_{i=1}^{n} d_e(r_i, g) )</td>
<td>N.A.</td>
</tr>
<tr>
<td>EDR ((R^m, S^n))</td>
<td>( m\delta )</td>
<td>( n\delta )</td>
<td>0</td>
</tr>
</tbody>
</table>

In ERP, the cost between two locations is measured by either directly calculating the Euclidean distance between two locations or inserting a new location for comparison

\[
d_{erp}(r_i, s_j) = \begin{cases} 
  d_e(r_i, s_j), & \text{if } r_i \text{ and } s_j \text{ are not gaps, it corresponds to } C(R^m, i, s_j) \\
  d_e(r_i, g), & \text{if } s_j \text{ is a gap, it corresponds to } I(S^n, j, g) \\
  d_e(s_j, g), & \text{if } r_i \text{ is a gap, it corresponds to } I(R^m, i, g)
\end{cases}
\]

(6.2)

where \( g \) is a new location.

Chen et al. [CN04] show that the choice of \( g \) does not affect the triangular property of ERP unless it is a fixed value, and they assign \( g = 0 \) in their study of measuring the distance between time series sequences. In my case, the x and y coordinates of a location are converted from its latitude and longitude with respect to zero degree in both directions.

In EDR, there is a distance threshold \( \delta \) to control the cost between two locations

\[
d_{edr}(r_i, s_j) = \begin{cases} 
  0, & \text{if } d_e(r_i, s_j) < \delta \\
  d_e(r_i, s_j), & \text{if } d_e(r_i, s_j) \geq \delta \\
  1, & \text{if } s_j \text{ or } r_i \text{ is a gap}
\end{cases}
\]

(6.3)

when either \( s_j \) or \( r_i \) is a gap, it corresponds to deleting the location, and the cost is the same.

This equation means that when the distance between \( r_i \) and \( s_j \) is small, the two locations are treated as the same and there is no cost associated with the overall edit distance, otherwise the cost is \( d_e(r_i, s_j) \). However, when applying EDR in measuring the distance between mobility sequences, it is not consistent to assign the cost value 1 with the penalty \( d_e(r_i, s_j) \). Therefore, in my case, I assign the cost \( \delta \) when either \( s_j \) or \( r_i \) is a gap.

In the following, I show the routine for calculating the distance between two sequences. The base condition in each edit distance is given in Table 6.1.
The routine for DTW is

$$DTW(R^m, S^n) = d_{dtw}(r_m, s_n) + \min\{DTW(R^{m-1}, S^{n-1}), DTW(R^{m-1}, S^n), DTW(R^m, S^{n-1})\}$$ (6.4)

which combines the current choice $d_{dtw}(r_m, s_n)$, i.e., either directly comparing current two locations or shifting one of them, with the previous optimal cost associated with the shortened sequences.

Similarity, Eq. (6.5) shows the routine of ERP by choosing the minimum cost from the three cases, which are directly compares the current two locations and insert a gap location in either sequence.

$$ERP(R^m, S^n) = \min\{ERP(R^{m-1}, S^{n-1}) + d_e(r_m, s_n), ERP(R^{m-1}, S^n) + d_e(r_m, g), ERP(R^m, S^{n-1}) + d_e(s_n, g)\}$$ (6.5)

For EDR, the cost is determined similarly according to the case that induces the lowest cost whether directly comparing current two locations or inserting a gap in each sequence.

$$EDR(R^m, S^n) = \min\{EDR(R^{m-1}, S^{n-1}) + d_e(r_m, s_n), EDR(R^{m-1}, S^n) + d_e(r_m, gap), EDR(R^m, S^{n-1}) + d_e(s_n, gap)\}$$ (6.6)

I will evaluate the performance of the three edit distances based on my mobility dataset in Section 6.6.2, and the EDR method is tested by choosing a value for $\delta$ from a range between 0.05 km and 0.15 km.

### 6.4 Trace Distance

In this section, I present a formal approach for measuring the distance between traces. The meanings of the notations used are summarized in Table 6.2. Alg. 3 and Alg. 4 present the detailed computation of the distance. The algorithms embody two observations. First, there is a high probability for the individuals to follow the same routes [SQBB10, LHL12]; however, the recorded traces may appear to be greatly different due to: 1) different sampling rate, 2) different initial recording time, or 3) the errors in GPS readings. Therefore, it should be useful to find the most similar subtraces while tolerating certain degree of inexactness. Second, individuals often traverse a route in both directions, e.g., going from home to workplace in the morning, then returning from workplace to home in the evening. Thus, I compare the traces in both directions and record the results that satisfy the distance threshold.
Algorithm 3 Trace-matching($T_p$, $T_{tr}$, $\delta_{init}$, $\delta_{thd}$)

1: for each $p_i \in T_p$, $i \geq 2$ do
2:     initial trace starts at index $j = 1$;
3: while $j \leq i$ do
4:     for each trace $T_i \in T_{tr}$ do
5:         for each point $p_k \in T_i$ do
6:             calculate $dist = \text{distance}(p_i, p_k)$;
7:             if $dist \leq \delta_{init}$ then
8:                 save the trace index and starting index to $M$;
9:         end if
10:     end for
11: end for
12: if $M$ is empty then
13:     return
14: end if
15: $k = j + 1$;
16: for $k \leq i$ do
17:     $M = \text{follow-up-matching}(T_p(j : k), M, T_{tr}, \delta_{thd})$;
18:     if $M$ is empty then
19:         $j = j + 1$; break;
20: end if
21: end for
22: end while
23: end for
24: return $M$
Table 6.2: Frequently used symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_i = {p_1, ... p_n}$</td>
<td>Trace $T_i$ contains $t_n$ sample points</td>
</tr>
<tr>
<td>$p_i$</td>
<td>The coordinate of each point</td>
</tr>
<tr>
<td>$t_i$</td>
<td>The time of $p_i$</td>
</tr>
<tr>
<td>$T_{tr}$</td>
<td>The collection of the training traces</td>
</tr>
<tr>
<td>$T_{tr, 60}$</td>
<td>The resampled traces according to 60 seconds</td>
</tr>
<tr>
<td>$T_i(j : k)$</td>
<td>A sub-trace from index $j$ to index $k$</td>
</tr>
<tr>
<td>$M = {m_1, ... m_k}$</td>
<td>The collection of portions of historical traces that match the current trace</td>
</tr>
<tr>
<td>$m_i.tid$</td>
<td>Trace index of a matched result</td>
</tr>
<tr>
<td>$m_i.sid$</td>
<td>The starting index in the matched trace</td>
</tr>
<tr>
<td>$m_i.eid$</td>
<td>The ending index in the matched trace</td>
</tr>
<tr>
<td>$m_i.\delta(tid, p)$</td>
<td>The distance of the trace with the target trace</td>
</tr>
<tr>
<td>$\delta_{init}$</td>
<td>Initial distance threshold</td>
</tr>
<tr>
<td>$\delta_{thd}$</td>
<td>Maximum distance threshold</td>
</tr>
</tbody>
</table>

In Alg. 3, I present a trace matching algorithm Trace-matching($T_p, T_{tr}, \delta_{init}, \delta_{thd}$). The input of the algorithm consists of the current trace $T_p = \{p_1, ... p_n\}$, the list of existing traces $T_{tr}$, the initial distance threshold $\delta_{init}$ for a matching between two locations and the maximum distance threshold $\delta_{thd}$. The output is the list of traces that maximally match with the current trace. Note that a matching need not be starting from the first index, but it should always end with the last index $|t_p|$. This ensures using the last location in finding a match, which is further elaborated in Section 6.5.3. The matching procedure starts by finding a sequence of locations in each trace in $T_{tr}$, such that each location is within $\delta_{init}$ distance to the initial index in $T_p$ (Line 3 to Line 11). Given the traces with the points within $\delta_{init}$ distance to the initial index of the target trace, the matching is started from the second location of the target trace to the following locations given in Alg. 4. If there is a matched candidate within a given distance threshold $\delta_{thd}$, the matched results are generated. However, if none of the existing traces can match the current trace, the current trace is shorten by removing the most temporally-distant one (Line 19), and the matching process repeats.

Alg. 4 describes the follow-up matching steps given the matched traces from the initial phase. The direction of the matching is determined when calling the follow-up-matching($T_p, M, T_{tr}, \delta_{thd}$) for the first time. Specifically, if the starting index $sid$ is less than the ending index $eid$, the current candidate trace is from index $sid$ to $eid + 1$ (Line 3). Otherwise, the current candidate trace is the reverse of the given trace from index $sid$ to $eid − 1$, i.e., the match is in a reverse order. However, when calling follow-up-matching($T_p, M, T_{tr}, \delta_{thd}$) for the first time, (in which case $|T_p| == 2$, Line 5), an
Algorithm 4 follow-up-matching($T_p, M, T_{tr}, \delta_{thd}$)

1: for each matching information $m_i$ do
2:   Get candidate trace id $j = m_i.tid$, $r = m_i.sid$, $s = m_i.eid$;
3:   if $r < s$ then
4:     $T_j' = T_j(r : s + 1)$;
5:     if $|T_p| == 2$ then
6:         Add a new match $m_k = \{j, s, r\}$;
7:     end if
8:   else
9:     Get the inverse partial trace $T_j' = T_j(r : s - 1)$;
10:   end if
11:   Calculate $dist = D(T_p, T_j')$;
12:   if $dist < \delta_{thd}$ then
13:     if $r < s$ then
14:         update $m_i = \{tid, sid, eid + 1\}$;
15:     else
16:         update $m_i = \{tid, sid, eid - 1\}$;
17:     end if
18:   else
19:     remove $m_i$ from $M$;
20:   end if
21: end for
22: return $M$
additional piece of information is added by reversing the previous matching. The reason
is that when comparing two line segments, the direction of each line is not considered,
therefore the subsequent matching may be conducted in either direction. The distance
of the two traces, given as $D(T_p, T'_j)$, is measured by one of the edit distances discussed
in Section 6.3.2. A comparison of the results from applying these edit distance measures
will be shown in Section 6.6.2. If the distance is less than the threshold $\delta_{thd}$, the matching
information is updated according to the current direction, otherwise the current matching
is removed.

Here the complexity of the two algorithms is analyzed. In Alg. 3, the most computa-
tionally expensive part is the initial matching (Line 3 to Line 11). During the initial
matching step, each location in the existing traces is compared with current initial index
$j$ in the target trace, therefore the complexity of finding the matching in each trace is
$O(N_p)$, where $N_p$ denotes the total number of sample points in existing traces. The space
requirement is $O(N_p)$. Alg. 4 iteratively computes the distance between two sequences
according to the information of initial matching. The complexity of the matching proce-
dure is $O(|T_p|^2)$, where $|T_p|$ is the length of the target trace. However, the computation
of this part can be greatly speeded up by comparing the distance threshold $\delta_{thd}$ with the
distance matrix (c.f. Eq. (6.4) to Eq. (6.6)). Once all the current distance values in
the matrix are greater than the threshold, the calculation can be terminated and current
matching can be removed. The space requirement is upper-bounded by the distance
matrix of size $|T_p|^2$. Therefore, the overall time complexity is $O(|T_p|^2)$. In practice, the
time complexity is determined by the longest trace in each individual’s data. The overall
space requirement is $O(|T_p|^2)$.

### 6.5 Location Prediction Methods

In this section, I consider the issue of predicting the next location at a specific point
in time, given a few previous locations and historical trajectories that match the cur-
rent trace. Three different location prediction methods are discussed in the following
subsections.

#### 6.5.1 Method A: Linear extrapolation

The first location prediction method is based on linear extrapolation without using his-
torical records. Let $p_i$ be the location at time $t_i$. I predict the location at time $t_x$ based
on the last two locations on the current trace. Let the two locations arise at $t_{i-1}$ and
$t_i$. Then there is no record in current trace in between time $t_i$ to $t_x$. In the linear
extrapolation, the location at time $t_x$ is
\[ \hat{p}_i = p_i + (p_i - p_{i-1}) \frac{t_x - t_i}{t_i - t_{i-1}} \] (6.7)

which basically assumes that the individual retains the previous direction and the average speed from location \( x_{i-1} \) to \( x_i \) until time \( t_x \).

The following two methods make use of historical records, i.e., they both use the trace matching results as described in Section 6.4.

### 6.5.2 Method B: Estimation according to the distance

The predicted location is the weighted mean location from the results of extending the matched traces one step further following the matched sequence, where the weight of each location is proportional to the inverse of the corresponding edit distance to the current trace. Note that this prediction method is only suitable for the case in which the current trace has the same sampling rate as that of the historical records. Otherwise, this method may lead to a large error. Therefore, in the following subsection, a mobility model is proposed to predict individual’s next location at any given time.

### 6.5.3 Method C: Estimation by Brownian Bridge

This location prediction method, referred to as Method C for convenience, combines the inferred location according to the Brownian Bridge model [Shr04] of each matched trace. In this subsection, the Brownian Bridge model is briefly described. The location prediction method is presented subsequently.

#### 6.5.3.1 Preliminary

Brownian Bridge is a random process following a specific rule to generate a path between two given locations. This movement model has been applied in estimating animals’ home range, migrate routes, as well as resource selection [HGKL07].

**Definition 6.3** A one-dimensional Brownian motion [Shr04] \( W(t) \in R \) is a continuous-time stochastic process satisfying the following properties:

- \( W_0 = 0 \)
- \( W_t \) is almost surely continuous
- For \( 0 \leq s \leq t \), \( W_t \) has independent increment, and \( W_t - W_s \sim \mathcal{N}(0, t - s) \),

where \( \mathcal{N}(\mu, \sigma^2) \) denotes the normal distribution with mean \( \mu \) and standard deviation \( \sigma \).
Definition 6.4 Let $W(t)$ be a one-dimensional Brownian motion. Given $T > 0$, the Brownian bridge \cite{Shr04} from $W_0 = 0$ to $W_T = 0$ on $[0, T]$ is the process

$$X(t) = W(t) - \frac{t}{T} W(T)$$

where $t \in [0, T]$.

Definition 6.5 Let $W(t)$ be a Brownian motion. Fix $T > 0$, and $a, b \in \mathbb{R}$. The Brownian bridge from $W_0 = a$ to $W_T = b$ on $[0, T]$ is the process \cite{Shr04}

$$X^{a \rightarrow b}(t) = a + \frac{(b - a)t}{T} + X(t)$$

where $X(t) = X^{0 \rightarrow 0}$ is the Brownian Bridge given in Def. 6.4.

According to this definition

$$EX^{a \rightarrow b}(t) = a + \frac{(b - a)t}{T}$$

and the variance is

$$\text{var}(X^{a \rightarrow b}(t)) = EX^2(t) = t - \frac{t^2}{T} = \frac{t(T - t)}{T}$$

Generally the Brownian Bridge from $a$ to $b$ is described in the following way. At time $t$, the estimated location is chosen from a Gaussian distribution, where the mean of the Gaussian distribution is the linear estimation according to Eq. (6.10), and the variance varies with respect to time. Specifically, according to Eq. (6.11) the variance increases at time $T/2$ to the maximum, and decreases till time $T$. The property of the variance is very suitable for modeling the uncertainties about a person’s locations between two known locations. When the predicted time is very close to the time obtained from the existing position samples, the mobility may not change greatly and the prediction can be of high accuracy. When the time of interest lies in between the two actual samples, the accuracy of the prediction result decreases.

In my case, the one-dimensional Brownian Bridge is extended to the two-dimensional version. An additional parameter, the variance of each trace $\sigma^2_m$, is introduced to model the mobility behaviors in each trace. Therefore, the variance of the estimated location is adjusted to

$$\text{var}(X^{a \rightarrow b}(t)) = \frac{t(T - t)}{T} \sigma^2_m$$

This equation can be extended to two-dimensional case, where the x and y coordinates vary independently. The estimation of the parameter $\sigma_m$ is discussed in the following subsection.
6.5.3.2 Variance estimation

In this subsection, I discuss how to estimate the variance $\sigma^2_m$ of a trace modeled as a Brownian Bridge. The estimation also incorporates the error of each GPS reading, which is around 10 meters when the signal is good.

Assuming that each GPS reading has a normally distributed error, which is assumed to be 0.01 km. Given a trace $T_i = \{p_1, p_2, \ldots, p_t\}$, each reading $p_i$ is a Gaussian distribution $\mathcal{N}(\mu_i, \sigma^2_g)$, where $\mu_i$ is the reading and $\sigma_g$ denotes the standard deviation associated with the GPS device used in the measurement. For the sample points in each trace, I group them into sets of three points, e.g., $\{p_1, p_2, p_3\}$, $\{p_3, p_4, p_5\}$, etc. In each group, I use the first and last locations to estimate the middle location. Besides using the odd-indexed points to estimate the even-indexed points, two other methods are considered. For instance, using the even-indexed points to estimate the odd-indexed points and also using the even- and odd-indexed points interchangeably. In the latter case, the points are grouped into $\{p_1, p_2, p_3\}$, $\{p_2, p_3, p_4\}$, etc. The comparison results are presented in Section 6.6.3. In the following the discussions are based on the first method.

In each group $\{p_{2k-1}, p_{2k}, p_{2k+1}\}$, the locations are generated from

\[
p_{2k-1} \sim \mathcal{N}(\mu_{2k-1}, \sigma^2_g) \tag{6.13}
\]
\[
p_{2k+1} \sim \mathcal{N}(\mu_{2k+1}, \sigma^2_g) \tag{6.14}
\]

When applying the Brownian bridge to estimate the location at time $t_{2k}$, I combine the locations given as the Gaussian distributions at time $t_{2k-1}$ and $t_{2k+1}$, respectively. Since the weighted mean of two Gaussian distributions is still a Gaussian distribution, the location at time $t_{2k}$ is

\[
\tilde{p}_{2k} \sim \mathcal{N}(\tilde{\mu}_{2k}, \tilde{\sigma}^2_{2k}) \tag{6.15}
\]

where the parameters are

\[
\alpha_k = \frac{t_{2k} - t_{2k-1}}{t_{2k+1} - t_{2k-1}} \tag{6.16}
\]
\[
t'_{2k} = t_{2k+1} - t_{2k-1} \tag{6.17}
\]
\[
\tilde{\mu}_{2k} = \alpha_k \mu_{2k-1} + (1 - \alpha_k) \mu_{2k+1} \tag{6.18}
\]
\[
\tilde{\sigma}^2_{2k} = t'_{2k} \alpha_k (1 - \alpha_k) \sigma^2_m + \alpha_k^2 \sigma^2_g + (1 - \alpha_k)^2 \sigma^2_g \tag{6.19}
\]

The likelihood of $p_{2k}$ given location $p_{2k-1}$ and $p_{2k+1}$ is

\[
\mathcal{L}(p_{2k}|p_{2k-1},p_{2k+1}) = f(p_{2k}|\mathcal{N}(\tilde{\mu}_{2k}, \tilde{\sigma}^2_{2k})) \tag{6.20}
\]
where \( f(a|N(\mu, \sigma^2)) \) denotes the probability at value \( a \) according to the normal distribution with mean \( \mu \) and variance \( \sigma^2 \).

Without loss of generality, I assume that the number of points in trace \( T_i \) is \( 2n + 1 \). Therefore, the likelihood of all the even-indexed points given all the remaining odd-indexed points is

\[
L(p_{2k} \in T_i | p_{2k-1} \in T_i) = \prod_{k=1}^{n} L(p_{2k}|p_{2k-1}p_{2k+1})
\]

\[
= \prod_{k=1}^{n} f(p_{2k}|N(\tilde{\mu}_{2k}, \tilde{\sigma}_{2k}^2))
\]

\[
= \prod_{k=1}^{n} \frac{1}{\sqrt{2\pi\tilde{\sigma}_{2k}^2}} \exp\left(-\frac{(p_{2k} - \tilde{\mu}_{2k})^2}{2\tilde{\sigma}_{2k}^2}\right)
\]

The corresponding log likelihood function is

\[
\log L(p_{2k} \in T_i | p_{2k-1} \in T_i) = -\frac{n}{2} \log 2\pi - \sum_{k=1}^{n} \log \tilde{\sigma}_{2k} - \sum_{k=1}^{n} \frac{(p_{2k} - \tilde{\mu}_{2k})^2}{2\tilde{\sigma}_{2k}^2}
\]

By dropping the constant term \(-\frac{n}{2} \log 2\pi\), and replacing \( \tilde{\sigma}_{2k} \) according to Eq. (6.19), the standard deviation parameter \( \sigma_m \) (c.f. Eq (6.12)) can be obtained by maximizing the function

\[
\log L' = -\frac{1}{2} \sum_{k=1}^{n} \log \left(t_{2k}'\alpha_k (1 - \alpha_k) \sigma_m^2 + \alpha_k^2 \sigma_g^2 + (1 - \alpha_k)^2 \sigma_g^2\right)
\]

\[
- \frac{1}{2} \sum_{k=1}^{n} \frac{(p_{2k} - \tilde{\mu}_{2k})^2}{t_{2k}'\alpha_k (1 - \alpha_k) \sigma_m^2 + \alpha_k^2 \sigma_g^2 + (1 - \alpha_k)^2 \sigma_g^2}
\]

where \( \sigma_g \) is a constant, \( p_{2k} \) is the reading at time \( t_{2k} \). The remaining parameters are calculated according to Eq. (6.16) to Eq. (6.19).

For notational simplicity, I write Eq. (6.25) in the following form

\[
\log L' = -\frac{1}{2} \sum_{k=1}^{n} \log \left(a_k \sigma_m^2 + b_k^2\right) - \frac{1}{2} \sum_{k=1}^{n} \frac{c_k}{(a_k \sigma_m^2 + b_k^2)}
\]

where

\[
a_k = t_{2k}'\alpha_k (1 - \alpha_k)
\]

\[
b_k^2 = \alpha_k^2 \sigma_g^2 + (1 - \alpha_k)^2 \sigma_g^2
\]

\[
c_k = (p_{2k} - \tilde{\mu}_{2k})^2
\]
Maximizing \( \log \mathcal{L}' \), in Eq. (6.26), is equivalent to minimizing \( -\log \mathcal{L}' \). In the following, I consider the convexity of the two functions: \( f(x) = \log(a_k x^2 + b_k^2) \) and \( g(x) = c_k / (a_k x^2 + b_k^2) \).

For the function \( f(x) \)

\[
\frac{d^2 f(x)}{dx^2} = \frac{-2a_k(b_k - a_k x^2)}{(a_k x^2 + b_k^2)^2}
\] (6.30)

According to the theory of convexity [BV04], the function \( f(x) \) is convex when \( b_k - a_k x^2 < 0 \), and it is concave when \( b_k - a_k x^2 > 0 \).

For the function \( g(x) \), it is strictly convex in my case according to the following lemma.

**Lemma 6.8** Given the values of \( a_k, b_k, \) and \( c_k \) from Eq. (6.27) to Eq. (6.29), and \( \sigma_m \geq \sigma_g \), the function \( g(x) = c_k / (a_k x^2 + b_k^2) \) is convex.

The proof of this lemma is in the Appendix A.3.3.

Minimizing the function \( -\log \mathcal{L}' \) is equivalent to minimizing a list of a mixture of convex and concave functions, or a list of convex functions. A solution to minimize \( -\log \mathcal{L}' \) can be obtained by firstly finding a solution to the first order derivative of \( -\log \mathcal{L}' \), which can be done by using Newton’s method [BV04], and then I have to check whether this solution indeed achieves the minimum. In my dataset, approximately 80% of the traces the constructed likelihood functions can be maximized, indicating the fitness of applying the Brownian Bridge in the current trace, and the estimated standard deviation varies from 0.01 km to 0.05 km. The details are discussed in Section 6.6.

### 6.5.3.3 Location prediction

Given individual’s current records in the given trace \( T_p = \{ p_1, p_2, ..., p_t \} \), similar historical traces and a time value \( t_x \), where \( t_x > t_i \), I focus on predicting the location at the time \( t_x \). The general rule is to combine individual’s current mobility behaviors, described by the linear extrapolation (given in Section 6.5.1), and his/her own historical records, i.e., the list of similar traces that are within the distance threshold \( \delta_{thd} \).

Alg. 5 presents the details of the location prediction method. As the input of this algorithm, I have the set of matched traces according to the set of matched information of each trace in \( M \) and the trace variance \( \sigma_m^2 \), and for those traces without the estimated variance the estimated result is given by linear extrapolation. I first normalize the time in both predicted trace and the matched trace according to the initial matched index (Line 1 and Line 4). This is because the estimated results for the Brownian Bridge model is highly time dependant. For the matched trace, I find the records that are temporally right before and after the normalized predicting time \( t'_x \) (Line 5). Then, the estimated location \( \tilde{p}_i \) based on current matched trace is a Gaussian distribution \( \mathcal{N}(\tilde{\mu}_i, \tilde{\sigma}_i^2) \),
Algorithm 5 location-prediction\((M, t_x, T_p, i_p)\)

1: Normalize the time in trace \(T_p\) and the predicted time \(t_x\) to \(t'_x\) according to the initial matched index \(i_p\);

2: for each matched trace do

3: Get the matched information: trace index \(j = m_i.tid\), initial index \(r = m_i.sid\), ending index \(s = m_i.eid\), and the edit distance to the current trace \(\delta(p, j)\);

4: Normalize the time in the matched trace based on index \(r\);

5: Get the location with index \(i_s\) and \(i_e\) in \(T_j\), such that \(t_{i_s} < t'_x < t_{i_e}\);

6: Estimate the location \(\tilde{p}_i\) at time \(t'_x\) according to the current trace and the locations at index \(i_s\) and \(i_e\);

7: end for

8: Get the mean location from the list of estimated location \(\tilde{p}_i \sim \mathcal{N}(\tilde{\mu}_i, \tilde{\sigma}_i^2)\), and the one \(\tilde{p}_l\) estimated by the linear model given in Eq. (6.7), where the weight for each \(\tilde{p}_i\) is the inverse of the corresponding distance value, and the weight for \(\tilde{p}_l\) is the inverse of the initial distance \(\delta_{init}\);

where the parameters are inferred according to Eq. (6.18) and Eq. (6.19), respectively.

Lastly, I combine the estimated results from each Brownian Bridge and also the linear extrapolation result from the current trace. The estimated location \(\tilde{p}_b\) is

\[
\tilde{p}_b \sim \mathcal{N}(\tilde{\mu}_b, \tilde{\sigma}_b^2)
\]

where \(\tilde{\mu}_b\) and \(\tilde{\sigma}_b\) are

\[
\tilde{\mu}_b = \frac{\sum_{k=1}^{\left| M \right| + 1} w'_k \tilde{\mu}_k}{\sum w'_k} \quad (6.32)
\]

\[
\tilde{\sigma}_b^2 = \sum_{k=1}^{\left| M \right|} \left( \frac{w'_k}{\sum w'_k} \right)^2 \tilde{\sigma}_k^2 \quad (6.33)
\]

where \(\tilde{\mu}_k = \tilde{\mu}_i\), \(\tilde{\sigma}_k = \tilde{\sigma}_i\) and \(w'_i = 1/\delta(p, j)\), for \(1 \leq k \leq |M|\), and \(j\) is the index of the matched trace. For \(k = |M| + 1\), \(\tilde{\mu}_{|M|+1} = \tilde{p}_l\) and \(w'_{|M|+1} = 1/\delta_{init}\), which is the result from linear extrapolation.

The reason for choosing the weight for the result from linear extrapolation as \(1/\delta_{init}\) is as follows. Note that for all the matched traces, I have \(\delta_{init} < \delta(p, j) < \delta_{thd}\). Also in my experiments, I set \(\delta_{thd} \in [\delta_{init}, 6\delta_{init}]\). For instance, when \(\delta_{init} = 0.05\) and \(\delta_{thd} = 0.3\), the weight for the linear extrapolation is \(1/\delta_{init} = 20\), and the sum of the weights for the remaining estimated location is approximately \(|M|/\delta_{thd}\). This means that when the number of matched trace is not too large, e.g., \(4 \leq |M| \leq 6\), the weight for the estimated result from the linear extrapolation is roughly the same as the sum of the weights to the results from the Brownian Bridge model. However, when the matched results are too
small, e.g., $|M| < 4$, high weight will be assigned to the results from linear extrapolation, and similarly more matched traces would lead to the high weights on the results learned from these records. Therefore, the choice of the weight $\delta_{\text{init}}$ for the linear extrapolation method leads to a balance between the current mobility behaviors and historical records.

### 6.6 Experimental Results

In this section, the experimental results are presented in three parts. Firstly, the three edit distance methods are compared by measuring the similarity on individual’s mobility traces. The most suitable one is then applied to the task of location prediction. Secondly, the results of applying the proposed variance estimation method (given in Section 6.5.3.2) are presented and the fitness of the Brownian Bridge model is evaluated. Thirdly, by using the most effective edit distance method, which is obtained from Section 6.6.2, three location prediction methods are compared at different time horizons.

#### 6.6.1 Mobility data

I use the dataset Zheng-GeoLife (see Appendix A.1 for details) in this study, and the following preprocessing is conducted. Firstly, each individual’s trajectory is divided into a series of traces based on the time gap between two GPS points. If the gap is longer than 300 seconds, the sequence will be divided into different traces. In order to test the performance of the edit distance, each trace is resampled at the rate of 10 seconds, 20 seconds, 30 seconds, and 60 seconds, respectively, and the corresponding datasets are denoted as $T_{tr,10}$, $T_{tr,20}$, $T_{tr,30}$, and $T_{tr,60}$. In the following two subsections, the Brownian Bridge model and location prediction method are validated using the sparse dataset $T_{tr,60}$. Predictions on $T_{tr,60}$ are relatively more challenging than using $T_{tr,10}$, $T_{tr,20}$ and $T_{tr,30}$, because individual’s mobility may change more during a longer period. Each individual’s traces are randomly separately into a training set and a predicting set, where the training set contains 70% of all the traces and the remaining traces are used to evaluate the location prediction methods. To cater for the characteristics of individual’s mobility behavior, the training and prediction steps are separately evaluated on the individual’s own data. The location of each point indicated by a pair of latitude and longitude is converted to x and y coordinates according to $0^\circ$ in both latitudinal and longitudinal direction [ZM11].

#### 6.6.2 Comparing different edit distance methods

In this subsection, I test the performance of the three edit distances on the GPS dataset, and I particularly focus on the performance on similar traces that involve either shifted
The edit distances between the traces are calculated in two different cases. In the first case, I calculate the distance of the traces from dataset $T_{tr,10}$ to its counterpart in the dataset $T_{tr,20}$, $T_{tr,30}$, $T_{tr,60}$, which is used to evaluate the edit distance in a shift of sampling. In the second case, I calculate the distance of the traces in dataset $T_{tr,20}$ to $T_{tr,30}$, which is used to evaluate the edit distance in the traces of totally different sampled positions. In both cases, the edit distance is compared to the three methods, namely, 1) dynamic time warping (DTW) [KPC01], edit distance with real penalty (ERP) [CN04], and edit distance on real-valued sequences (EDR)[COO05]. The collections of the distance are denoted as $d(T_{tr,10}, T_{tr,20})$, $d(T_{tr,10}, T_{tr,30})$, $d(T_{tr,10}, T_{tr,60})$ and $d(T_{tr,20}, T_{tr,30})$, respectively. For the EDR method, the distance threshold is set to be 0.05 km, 0.1 km and 0.15 km, respectively. Since each method measures the distance in a different way, the distance is normalized to the range between 0 to 1. To avoid influences from the extremes, the 10 maximum and 10 minimum distances are removed before the normalization.

Figure 6.1: The boxplot for different edit distance methods between each pair of dataset, $d(T_{tr,10}, T_{tr,20})$ in (a), $d(T_{tr,10}, T_{tr,30})$ in (b), $d(T_{tr,10}, T_{tr,60})$ in (c) and $d(T_{tr,20}, T_{tr,30})$ in (d). For the EDR method, the plot in each figure from left to right uses the distance threshold of 0.05 km, 0.10 km, 0.15 km, respectively.
Figure 6.1 shows the distribution of each edit distance for the different cases. Firstly, in these cases, $d(T_{tr,10}, T_{tr,20})$ in (a), $d(T_{tr,10}, T_{tr,30})$ in (b) and $d(T_{tr,10}, T_{tr,60})$ in (c), the DTW method shows the best performance by presenting much lower distance values in the 25%, 50% and 75% percentiles, than the other two methods. The remaining two methods, ERP and EDR, show roughly the same performance, even when the distance threshold in EDR is adjusted from 0.05 km to 0.15 km. This result shows that two similar traces sampled at different rates can’t be matched by just tuning the distance threshold. Finding appropriate matching in the corresponding sequence is more important in measuring the similarity. Moreover, Figure 6.1 (d) shows that in the case of sampling at different positions, e.g., $d(T_{tr,20}, T_{tr,30})$, the performance of DTW is much better than the remaining two methods, which presents a lower 75% percentiles than the 25% percentiles of the two other methods. This shows that DTW is more suitable for the case of sampled locations at different positions since it optimally finds the nearest points in two traces by maintaining the order. Based on these set of experiments, in the task of location prediction, the DTW method is applied to measure the distance between mobility sequences.

### 6.6.3 Estimating the parameter of the Brownian Bridge model

![Graph showing the distribution of the standard deviation estimated in each trace based on the Brownian Bridge model. Note that there is one trace with variance of 0.069 km. The mean standard deviation of these traces is 0.021 km.](image)

In this subsection, I show the results of the parameter $\sigma_m$ of each trace fitted by the Brownian Bridge model according to the method presented in Section 6.5.3.2. The standard deviation of the original GPS reading is set to be 0.01 km. The method of constructing the likelihood function is using the even- and odd-indexed points interchangeably.

Among the 3225 training traces in my dataset, it is possible to maximize the likelihood function given in Eq. (6.25) for 82.5% of the traces, and the distribution of the obtained
standard deviation is in Figure 6.2. Almost all these traces show a standard deviation of less than 0.05 km, with only one trace presents the standard deviation of 0.069 km, and the mean standard deviation is 0.021 km, indicating that the Brownian Bridge model is a suitable choice for predicting the location given the locations before and after the predicting time. When applying the Brownian Bridge model in location prediction, I need to search for the locations in the given trace that are similar to the current situation. In order to have the same timing references in the search, I normalize the time in the matched traces with respect to the initial index of the partially matched trace (c.f. Line 1 and Line 4 in Alg. 5, respectively).

Also, I compare the results of the estimated $\sigma_m$ in different cases mentioned in Section 6.5.3.2, i.e., the likelihood of even-indexed points given odd-indexed points or vice versa. Among 81.8% and 82.2% of the traces, the likelihood functions can be maximized and the estimated mean standard deviation is 0.0188 km and 0.0190 km, respectively. This result shows that the way of constructing the likelihood function does not affect the estimated $\sigma_m$.

### 6.6.4 Location prediction

In this subsection, I evaluate the location prediction methods in two respects. As mentioned in the preceding subsection, the training and prediction are based on the dataset $T_{tr,60}$. Here, firstly the three location prediction methods (given in Section 6.5) are compared. Specifically, using 70% of an individual’s traces as the training data, the location prediction methods are applied to make prediction of the next location on the remaining 30% traces. For each prediction, all the previous locations in the current trace and the time for prediction are given. The performance is evaluated by the distance of the predicted location to the actual location in the current trace. Note that when predicting using the Brownian Bridge model, the estimated mean location is the result. Also the parameter $\delta_{thd}$ is tested with the values of 0.3 km, 0.2 km, 0.1 km and 0.05 km, with the initial threshold set to be $\delta_{init} = 0.05$ km in all these cases. Secondly, the performance of the Brownian Bridge method is evaluated for longer time horizons, i.e., predicting the location after 120 and 180 seconds, respectively.

Figure 6.3 shows a comparison of the mean prediction errors given by the three methods at different values of the distance threshold $\delta_{thd}$. Note that as shown in Table 6.3, although the linear extrapolation method is not affected by the parameter $\delta_{thd}$, for ease of comparisons I plot its results against that of the Brownian Bridge method which is greatly affected by $\delta_{thd}$.

Firstly, in all the cases tested, the prediction errors in Method C, the combination of linear extrapolation and Brownian Bridge model, is always lower than either Method A or Method B that uses linear extrapolation or similar traces alone, respectively. Secondly,
Figure 6.3: The mean prediction errors in one step location prediction based on three different methods, where $e_b$, $e_w$ and $e_l$ corresponds, respectively, with respect to the mean prediction error given by the Brownian Bridge method, prediction results using similar traces, and the linear extrapolation method. $e_w^s$ and $e_b^s$ are the results only using the most similar trace according to the matched results.

Table 6.3: A summary of the one step estimation according to different distance threshold $\delta_{thd}$. $N_p$ is the number of instances where a prediction is made possible from at least one partial match with the target trace, such that the Brownian Bridge model can be applied to predict the future location. $|M|$ indicates the average number of matched traces given the current predicting trace. $\sigma(|M|)$ is the standard deviation of the matched traces.

<table>
<thead>
<tr>
<th>$\delta_{thd}$ (KM)</th>
<th>0.3</th>
<th>0.2</th>
<th>0.1</th>
<th>0.05</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N_p \cdot 10^4$</td>
<td>3.99</td>
<td>3.66</td>
<td>3.14</td>
<td>2.51</td>
</tr>
<tr>
<td>$</td>
<td>M</td>
<td>$</td>
<td>15.6</td>
<td>14.7</td>
</tr>
<tr>
<td>$\sigma(</td>
<td>M</td>
<td>)$</td>
<td>153</td>
<td>149</td>
</tr>
</tbody>
</table>
in Method B the choice of using either the most similar trace or the list of similar traces depends on the distance threshold. Specifically, in Method B, when the similarity threshold is large, e.g., $\delta_{thd} = 0.3$ or 0.2, the prediction result is adversely influenced by the most similar trace, since using only the most similar trace results in a larger mean prediction error than that by using all the matched traces. However, when the distance threshold is small, e.g., $\delta_{thd} = 0.1$ or 0.05, using the most similar trace is more effective than using all the matched traces in Method B, which may be due to the stricter distance criterion in measuring similarity. However, for the prediction results by Method C, in all the cases mentioned before, making use of the most similar trace is slightly worse than that obtained by using all the similar traces. This is because the Brownian Bridge model is able to adjust the prediction with respect to the given point of time. Thirdly, given $\delta_{thd} = 0.1$, Method C is much more discriminative than both the linear extrapolation method (Method A) and the prediction using only similar traces (Method B). Specifically, the mean prediction error for Method C is 0.084 km, while that of Method A and Method B is 0.099 km and 0.105 km, respectively.

Table 6.3 shows the results of trace matching with respect to different $\delta_{thd}$. When $\delta_{thd}$ decreases from 0.3 to 0.05, the number of predictable cases decreases substantially, from $3.99 \times 10^4$ to $2.51 \times 10^4$, reflecting a high spatial resolution is not applicable for the majority cases since the predictable cases decrease to 63% of that achieved when $\delta_{thd} = 0.3$. Also, both the mean numbers of matched traces $|M|$ and $\sigma(|M|)$ decrease along with the decrease of $\delta_{thd}$, which is reasonable since lower similarity threshold leads to fewer matching. However, the relatively large $\sigma(|M|)$ compared to the value of $|M|$ indicates that for the majority of the traces, the number of matched traces are small, while for the minority, there is a large number of matchings. These results also confirm that individuals are indeed highly likely to follow similar routes, and they are less likely to explore new paths [LHL12].

Comparing the results from Brownian Bridge with those from linear extrapolation, my experiments show that using historical results can improve the prediction accuracy. Also, comparing the results from the Brownian Bridge model with those from using similar traces alone, merely using the historical results cannot guarantee the prediction results since individual’s current mobility behavior is ignored. Figure 6.4 illustrates how the historical records and linear extrapolation method mutually enhance each other.

Figure 6.4 (a) shows that the predicted result by a combination of two Brownian Bridge models is closer to the actual location than the result obtained by linear extrapolation method. This is due to the high similarity of the matched traces with the current trace, with a distance of 0.060 km and 0.077 km, respectively. According to the estimation rule given by Eq. (6.32), there is a high weight for the results from the Brownian Bridge model, which is $w_{bb} = 1/0.06 + 1/0.77 = 29.65$, while the weight for the linear extrapolation method is $w_t = 1/\delta_{init} = 20$. As a contrast, Figure 6.4 (b) shows that the
Figure 6.4: In both figures, $p_1 - p_2 - p_3$ indicates the current trace, and $p_l$ is the result of linear extrapolation according to preceding records $p_1 - p_2$. The blue dashed line indicates the starting location and the ending location of the matched trace to the current trace, which is not shown in either figure. The prediction results by using the Brownian Bridge model alone is the weighted mean of the Gaussian distributions. For instance, in (a) the result is the mixture of two Gaussian distributions derived from the two similar traces, and in (b) it is from one similar trace. The corresponding probabilities generated by the Gaussian distributions are indicated by the contours in each figure.

linear extrapolation method is closer to the actual location than the result from a single Brownian Bridge model. Therefore the predicting result is tuned by assigning relatively larger weight on the result from linear extrapolation method since only one similar trace is found.

I evaluate my method for predicting location with time horizons of 120 seconds and 180 seconds. Note that the location prediction by using similar traces is only suitable for the case to predict the location at a point in the similar traces. This is because the longer the prediction time is, the less likely to find a location at roughly the same time in the similar traces. Therefore, the comparison is only made between the linear extrapolation method and Method C. The prediction results, which are in terms of the distribution of errors shown in Figure 6.5 (a). The mean errors for the consecutive steps are 0.099 km, 0.214 km, and 0.345 km for the linear extrapolation method, while the corresponding results for the Brownian Bridge model are 0.084 km, 0.173 km, 0.281 km, respectively. In all these cases, the Brownian Bridge model shows lower mean errors than the linear extrapolation, especially for predicting the location at longer time horizons.

Moreover, Figure 6.5 (b) shows the distributions of the estimated standard deviation of the predictions in Method C in the case of predicting the future location at $t_x = 60$, $t_x = 120$, and $t_x = 180$. A few observations can be made from this figure. Firstly, in each of these cases, the estimated standard deviation is very small, which is less than 0.03 in
Chapter 6. Location Prediction at High Spatio-Temporal Resolution

Figure 6.5: (a) The distribution of the prediction errors according to linear extrapolation method $e_1$ and Method C $e_6$ in the following 60 seconds, 120 seconds and 180 seconds. The two boxplots on the left show the distribution of the one step location prediction results, i.e., predicting the location 60 seconds into the future. The middle two plots are the corresponding results given by predicting the location 120 seconds into the future. The last two are the prediction results for 180 seconds into the future. (b) The distributions of the standard deviation of the results given by Method C in predicting the location at 60 seconds, 120 seconds and 180 seconds later.

According to the results given in Section 6.6.3, the mean standard deviation $\sigma_m$ for the traces is 0.021 km, and also the error of GPS measurement is $\sigma_g = 0.01$ km. Therefore, the maximum standard deviation given by Eq. (6.34) is when $\alpha_k = \frac{1}{2}$, indicating inferring a location in between two samples, which is roughly 0.082 when $t'_{2k} = 60$ in my case. However, a much smaller standard deviation given in Figure 6.5 (b) indicates that the predictions are mostly near either the starting location or the ending location in the similar traces. This is because when constructing the Brownian Bridge model in the similar traces, the predicted locations with low standard deviations are mostly near either the starting point or the ending point (c.f. Eq. (6.19)). Secondly, the estimated standard deviation does not increase significantly when predicting the location at longer time horizons, e.g., when $t_x = 120$ or 180. Recall that in each prediction the time is always normalized according to the corresponding initial index in both the current trace and a matched trace. The locations used in constructing the Brownian Bridge model are always the two locations that correspond to right before and right after the normalized time of prediction. The only difference between the predictions at short time horizon (e.g., $t_x = 60$) and those at longer time horizons (e.g., $t_x = 120$ or 180) is that in the
latter case it is more difficult to find a good matching between the time of the sampled points with the time of prediction, which results in a large variance.

According to the comparisons based on the mean prediction errors and the estimated standard deviations, combining the linear extrapolation method and individual’s historical records is a suitable choice for location prediction.

### 6.6.5 Discussions

For the few location prediction methods discussed before, the Brownian Bridge model fails to predict if there is no similar trace within a given distance threshold to the current trace. Because the linear extrapolation method shows decent results while using only two location samples. A possible remedy to the Brownian Bridge method is to apply the linear extrapolation in the case of failing to find a match in the records.

One useful method for location prediction is the Kalman filter. I do not choose Kalman filter for the following reasons. Firstly, a Kalman filter needs to be specifically designed for each trace since the mobility behaviors may change greatly even for the same individual. Secondly, learning the parameters in each trace needs sufficient number of sample points, implying that the method is unable to make any prediction for traces with few records. In contrast, since the matching can be based on a subsequence of the current trace with a few existing samples for Method C, it will be able to predict the future location even with a few records.

### 6.7 Chapter Summary

In this chapter, I have presented a new algorithm for location prediction by making use of similar traces and individual’s current mobility information. The similar traces with current target trace are found on the basis of the edit distance. In order to predict individual’s location at any given point of time, I use a Brownian Bridge to model the uncertainties about person’s movements in between any two locations. The final prediction combine both the current mobility behaviors described by the linear extrapolation method and also the estimated results from the individual’s Brownian Bridge models from similar traces. Experimental results show that my location prediction method by using Brownian Bridge model is better than that using only the historical records or the linear extrapolation method.

There are a few issues unresolved in this study. For instance, I have shown in some cases, the linear extrapolation method is better than using the historical records or vice versa. Therefore, how to choose the right method at each predicting time is a relevant question for future research.
Chapter 7

Discussions and Conclusion

I have presented my studies in uncovering individual’s mobility patterns based on GPS data. I reviewed a large number of existing studies on individual’s mobility, which covered location inferences and prediction, inferring the modes of transport, extracting trajectory patterns, recognizing location-based activities and constructing mobility models. I also presented my studies on four major topics in individual’s mobility patterns. I presented an unsupervised method for inferring the modes of transport which eliminates the tedious data labeling and training work while achieving competitive level of accuracy with the best known existing supervised method. I also presented a study of the predictability of individual’s mobility at various spatio-temporal resolutions. Based on clearly defined notion of locations, I showed that the individuals’ mobility sequences exhibit high predictability. To reflect the high predictability as well as the scaling properties of individual’s mobility uncovered in recent studies, I presented a Markovian mobility model. I showed both theoretically and experimentally that the proposed Markovian mobility model indeed captured the essential characteristics of individual’s mobility behaviors. I lastly presented a new method to predict a person’s future locations at high spatio-temporal resolution by using the individual’s historical traces that exhibit similarity with the current trace. Compared to using either historical records or linear extrapolation method alone, the proposed location prediction method shows lower mean prediction errors.

7.1 Future Work

Although some studies in individual’s mobility have been presented in this thesis, many interesting issues remain unresolved.
7.1.1 Individual’s mobility and social networks

Recent studies show that there is a high association of individual’s mobility with their social connections [WPS+11, CML11]. Specifically, 10% to 30% of individual’s mobility is determined by their social relationship, and the commuting behaviors account for 50% to 70% of individual’s mobility [CML11]. Moreover, the high predictability of individual’s mobility shown in Chapter 4 indicates that it is highly predictable when individual visits previous locations. However, it is difficult to predict when individual visits a new location based on current context. For instance, in the second-order Markov model the current context is the previous two locations in the sequence. Since individuals often visit new locations because of social connections, an interesting issue here is whether it is possible to combine the information of individual’s relationship in predicting future locations.

7.1.2 Flexible mobility models

The spread of diseases is influenced by individuals’ mobility and their social network structures [EGK+04, BCG+09, NCFGJ12]. Such models generally cover a large area over a relatively long duration, e.g., in a rural area over a few weeks, and the spatio-temporal resolution required may be low under this situation, compared to when the person is in a densely populated area and interacting with other individuals regularly. Such applications require mobility information at different spatio-temporal resolutions. This special requirement challenges the existing mobility models. Is it possible to have a single tunable mobility model that realistically exhibits different mobility behaviors at various resolutions?

7.2 Concluding Remarks

The research in individual’s mobility remains wide open. I hope that the findings of this thesis can help deepen our understanding of individual’s mobility behaviors and also assist future development of applications and services that lead to individual’s well-being and conveniences.
Appendix A

Supplementary Information

A.1 Mobility Dataset

This part provides a general overview of the GPS datasets used in this thesis.

A.1.1 Raw GPS Dataset: Zheng-GeoLife

In the project of GeoLife [Zhe10], Zheng [ZLC⁺08, ZZXM09] releases a collection of more than 100 individual’s GPS traces around China between 2008 and 2009. Each individual’s trajectory consists of a series of time-stamped positioning readings in pairs of longitude and latitude. For example, the following GPS readings show a partial trajectory of one individual

\[ x_1 = \{(116.3329^\circ, 39.9687^\circ), 2008-01-24 13:16:21\}, \]
\[ x_2 = \{(116.3331^\circ, 39.9674^\circ), 2008-01-24 13:19:18\}, \]
\[ x_3 = \{(116.3331^\circ, 39.9667^\circ), 2008-01-24 13:20:54\}, \]

![Figure A.1](image.png)

Figure A.1: The distribution of the sampling rate of the mobility dataset.
The time interval between two consecutive readings within one trace\(^1\) or the called sampling rate in this dataset ranges from a few seconds to a minute, and the distribution of the sampling rate is given in Figure A.1. Two types of sparsity, however, are observed from the raw data: 1) the overall duration of one individual’s data is too short, e.g., less than a month, 2) the large gap between two GPS readings, e.g., a week or more. Therefore, a subset of the GPS data are chosen, which includes 40 individual’s mobility data and each lasts for about 16 weeks. The overall length of the trajectories is more than 300,000 kilometers and the accumulated duration of the trajectories is about 5,200 hours. This dataset is named as Zheng-GeoLife.

### A.1.2 Labled GPS Dataset: GPS-Modes

A subset from the raw GPS dataset [Zhe10] is labeled with the information of modes of transport [ZLWX08, ZLC+08, ZCL+10]. This subset contains about 30 individuals’ labeled trajectories, however, some of the trajectories are sparse and lasted for a short duration while other trajectories contain a few modes of transport, e.g., the number of modes is no more than 2. Therefore, only ten individuals’ trajectories are chosen in the study of Chapter 3. Another dataset with one individual’s trace, with the sampling rate of roughly 30 seconds and the duration of more than 10 months, is also tested in Chapter 3.

In total, there are more than two million GPS data points in the two datasets combined. These trajectories are recorded from April 2007 to September 2009, and the overall durations are more than 2000 hours and covered the distance of more than 30,000 kilometers. A variety of modes of transport are observed from these dataset. 9 individuals travel around with four modes of transport, such as walking, biking, taking a bus, and driving a car, although certain types of the modes are only used occasionally. The remaining 2 individuals apply three types of modes of transport. These two datasets are named as GPS-Modes.

### A.1.3 Supplementary Dataset: Chon-LifeMap

A supplementary dataset, named as Chon-LifeMap, is introduced in the study of individual’s general mobility behaviors. This dataset is collected by a mobility monitoring system, namely ”LifeMap”, developed by Yohan et.al. [CTSC11]. This system monitors the individual’s mobility during most time of the day by using both GPS and Wireless Positioning System (WPS). In particular, the WPS can capture the mobility information, e.g., the location, when GPS fails, such as indoor places. However, the accuracy of the

\(^1\)Normally when two consecutive readings have an exceptional large gap, e.g., half an hour or more, the two sample points are considered in different traces.
two positioning methods is quite different. For instance, the positioning accuracy provided by WPS is normally around 500 meters [CTSC11]. The accuracy of GPS reading is around 10 meters, but it may vary greatly with regard to the location of the device. Therefore, when both GPS readings and the locations of the cell towers are available, the former ones are chosen. This dataset is useful for uncovering the statistical properties of individual’s mobility given in Appendix A.2. In this dataset, 7 individual’s traces that are sufficiently long are chosen.
A.2 Additional Properties of Mobility

Existing studies in predicting individual’s future whereabouts [AS03, SMM11, SKJH03, LHL12] and modeling individuals’ mobility [LH06, GC10, YZC09] share common theoretical background, where Markov models of different orders or hidden Markov models are applied. However, when using Markov models, there are two prerequisites. Firstly, the mobility sequences should be stationary, i.e., the probability distribution of visiting each location is stabilized or slightly changed over the sequence. This condition should be considered especially when a high order Markov model is chosen. Secondly, based on the study of extensive WiFi data [SKJH03], the second-order Markov model presents the highest predicting accuracy among all the models. However, two issues are ignored in the previous studies. 1) They failed to validate the choice of various Markov models rather than other alternatives, such as i.i.d. sources. 2) Even if the Markov property of individuals’ mobility is satisfied, these models seldom, if at all, try to handle the case when the “observed state” is ineffective, i.e., when the next location is novel given current historical records.

In this section, I aim to uncover key statistical properties, e.g., stationarity and Markov property, of individuals’ mobility. The stationarity is inferred based on the analysis of the autocorrelation of the sequences recorded under three popular encoding schemes for representing individuals’ mobility sequences, namely time-based scheme, movement-based scheme and a hybrid of time- and movement-based scheme (see the definition in Section A.2.2.2). Given individuals’ mobility sequences, both i.i.d. model and first-order Markov model are applied to infer the parameters of the corresponding models. The most suitable model is inferred through hypothesis testing, which compares the entropies generated from the corresponding models with the estimated entropy from the mobility sequences. Then, based on the two statistical properties, I model individuals’ mobility by using both the first- and the second-order Markov models along with three revisiting rules to handle the unobserved locations given the previous observations.

My contributions lie in a few aspects.

• The individuals’ mobility sequences are stationary with weekly or daily periodicity.

• The hypothesis testing shows that certain dependency exists in individuals’ mobility sequences, which is not captured by the i.i.d. model.

• By comparing the average log-loss values given by both the variants of the first- and second-order Markov models, I show that the rule that combines both the temporal locality, e.g., the visitation frequency, and the spatial locality, e.g., the distance to the current location, can best estimate the individuals’ next whereabouts.

In particular the last finding is in direct contrast to the recent results in [SKWB10, LHK09], which supposes the individuals’ mobility only affected by either factor alone.
A.2.1 Related Work

Some researchers have found that the Markov model is useful in predicting individuals’ next whereabouts [AS03, SMM+11, SKJH03, LH06, GC10]. Ashbrook et al. [AS03] apply a second-order Markov model to predict individuals’ movement based on the inferred significant locations. By using extensive WiFi data collected, Song et al. [SKJH03] compare various predictors, e.g., Markov predictors of different orders and LZ predictors, on the mobility sequences. They show that the lower order Markov model, i.e., the second-order Markov model, presents the highest predicting accuracy among all the models. However, the theoretical basis of such phenomenon is unclear. Semi-Markov model [LH06] and Hidden Markov model [GC10] are also introduced to model individuals’ spatio-temporal mobility behaviors, e.g., steady state and transient state. The major advantage of semi-Markov model over a Markov model of a given order is that, the transition dependency between the locations can only be satisfied when the time gaps between transitions are within a threshold. Therefore, the temporal information, e.g., the distribution of transition time, can be inferred based on each transitions. In [GC10], the individuals mobility stage and the state of hidden Markov model are associated by the directions of series of sample point’s velocity. Except that the weights are assigned to each training set according to its temporal orders in the sequences, the remaining training and learning procedures are similar to the general hidden Markov model. Although both the semi-Markov model and the hidden Markov model can better infer the spatio-temporal mobility behaviors than the general Markov models, the assumption that individuals’ mobility present high spatial dependency is not validated.

A.2.2 Stationarity and Markovian Properties

Two datasets are used in this study. The first dataset, named as Chon-LifeMap, is applied in validating the general properties of individuals’ mobility, e.g., stationarity and Markovian. The reason is that, in Chon-LifeMap, the individuals’ mobility is captured during most time of the day, e.g., from the time individual uses his/her phone when he/she gets up till the time he/she goes to bed. Both datasets, namely, Chon-LifeMap and Zheng-GeoLife, are used in validating the two Markov models along with three different revisiting rules given in Section A.2.3.

A.2.2.1 The definition of locations

The grid map method is applied to define each location as a rectangular area of same size [LHL12, GC10]. There are two parameters, namely spatial scale \( s \) and origin, in the grid map method. In Chapter 4, I have shown that the choice of origin has minor effects on the information measurement extracted from individuals’ mobility sequences, e.g., the
number of distinct locations and the entropy rate of the sequence. Therefore, I choose the origin to be the minimum value of the latitude and longitude from all the readings in each individuals’ dataset. Comparatively, spatial scale has more impact on describing the mobility, and it is thoroughly considered here. When \( s = 1 \), the difference in the adjacent grid lines along both latitudinal direction and longitudinal direction is 0.001°, which corresponds to 111 meters and 87 meters respectively around the Beijing area. These values may vary in different areas, but as far as individual’s mobility is concerned they are roughly the same. Two choices of \( s \) are tested in this chapter, i.e., \( s = 1 \) or \( s = 6 \). When \( s = 6 \), the size of each location is roughly 600m × 600m, which is slightly larger than the accuracy of the cell towers. While \( s = 1 \) indicates a high level of mobility resolution, and it is used for comparison.

A.2.2.2 Encoding Mobility Sequences

Three schemes [BD02] are introduced to encode individuals’ mobility sequences based on the discrete location symbols converted from the raw readings.

- **Time-based scheme (T-Sch):** the locations are sampled at a uniform temporal scale \( \tau \) (in terms of minutes). When multiple locations are encountered within a sampling period, an arbitrary location is chosen.

- **Movement-based scheme (M-Sch):** the locations are sampled only when a minimum number of distinct locations, denoted by \( m \), is traversed. For instance, if \( m = 1 \), only when there is a change of locations, the location symbol is added to the sequence.

- **Hybrid of time- and movement-based scheme (TM-Sch):** the locations are sampled according to both \( \tau \) and \( m \). Generally, the sequence is constructed by the T-Sch according to \( \tau \). Meanwhile, within the sampling period, M-Sch is applied.

The comparisons of the three schemes are made according to the mobility information, namely, transitions between locations and the dwelling occurs at each location, that each scheme captured. T-Sch performs well in preserving the information of dwelling at each location, since a long dwelling period is modeled by a list of identical location symbols in the sequence. While the transition between locations are mostly missed due to arbitrary choice of a location during a sampling period. M-Sch mainly captures the transitions between locations. Especially when \( m = 1 \), all the transitions are preserved in the sequence. While the situation of long dwelling is not captured since only one single location symbol is added to the sequence during the long dwelling period. Comparatively, both types of mobility information are perfectly preserved in TM-Sch.
When applying the T-Sch or TM-Sch, there will be no record in certain periods of time. I assume that during the period of no record the individuals stay in the previous location until the next location is recorded. Under this assumption, there is no unknown locations in the M-Sch, since there is no changing of locations during the period with no records. Also, considering the privacy issue, I define the blackout periods, from 8:00pm to 6:00am of the following day. During the blackout periods, I ignore any mobility information. Also, the ping-pong transitions are similarly detected among the transitions [LH06] to avoid the fake frequent transitions among the adjacent locations, where they are defined as the eight adjacent grids in my case. However, I do not merge the distinct locations that are involved in the ping-pong effects. Instead, I consider these locations to be the same as the first one in the ping-pong transitions.

A.2.2.3 Stationarity and ergodicity of mobility

The notations used in the following subsection are as follows. Let \( X = (x_1, x_2, ..., x_n) \) be the mobility sequence converted from a raw trajectory. Each \( x_i \) represents a location ID and is from a finite location set \( \Omega = \{a_i|1 \leq i \leq n_A\} \), where \( a_i \) is the location ID and \( n_A \) is the total number of distinct locations in the sequence. Also, \( x_i^j = (x_i, x_{i+1}, ..., x_j) \) indicates a subsequence of \( X \).

A stationary process has a probability distribution that is invariant over time, such that for any \( i \geq 1 \) and \( l \geq 1 \) the probability distribution of a subsequence of length \( l \) in \( x_i^\infty \) is the same. To validate that a sequence satisfies stationarity for every value of \( l \geq 1 \), i.e., strong stationarity, is both impossible and impractical. Hence weak stationarity or local stationarity is usually used. Weak stationarity holds when the stationarity is satisfied only for a certain \( l \), and local stationary only considers a subsequence. Both properties can be verified by analyzing of the autocorrelation of the sequence. At a given lag \( \lambda \) or shift length, the autocorrelation \( r(x_1^n, x_1^n, \lambda) \) of a discrete symbol sequence [SBT06] is given by

\[
r(x_1^n, x_1^n, \lambda) = \frac{|\{i : x_i = x_{i+\lambda}, \max(1, 1 - \lambda) \leq i \leq \min(n, n - \lambda)\}|}{\min(n, n + \lambda, n - \lambda)} \quad (A.1)
\]

where \( \lambda \) may be shifted both forward and backward. The denominator is the length of the overlapped sequence for the given \( \lambda \), and the numerator is the number of identical symbols within the overlapped sequence.

When the mobility sequence is encoded according to T-Sch, the length of the sequence is fixed, and the original method for calculating the autocorrelation can be applied. However, when the mobility sequence is encoded in either M-Sch or TM-Sch, the length of the sequence may vary slightly in different days or weeks. Therefore, in such cases, I modify the meaning of \( \lambda \) from shifting length to shifting period, and every \( \lambda \) corresponds
to the shifting period of 6 hours. Although recent studies [GHB08, SQBB10, LHL12] show that individual may present regular mobility behaviors, it will be imprudent to suppose that \( x_{i+\lambda} = x_i \) only when shifting an exact number of hours. Therefore, I relax the matching condition by allowing an identical location to arise within one hour of the given shifting period.

If this short term autocorrelation is independent of the index \( n \), allowing slight fluctuations, this process is considered to be stationary. If the autocorrelation value is similar only when \( 0 \leq \lambda \leq v \) but different for \( \lambda > v \), then the sequence is considered to be locally stationary of window length \( v \) or less.

![Figure A.2: The plot of autocorrelation values for individual GS1, GS12 and GS2 according to the sequences generated from three different schemes. For each individual the sequence is encoded based on T-Sch, M-Sch, and TM-Sch from top to bottom.](image)

Since Chon-LifeMap captures individuals’ mobility over most time of the day, I choose this dataset to evaluate the fundamental statistical properties of individuals’ mobility. Figure A.2 and Figure A.3 show the analysis of each individual’s mobility sequence in Chon-LifeMap encoded in the three different schemes, where the spatial scale \( s = 6 \) and temporal scale \( \tau = 60 \). Note that every \( \lambda \) corresponds to 6 hours. Therefore the shifting of every 4 lags corresponds to a day, and every 28 lags corresponds to a week, and so on.

Firstly, the mobility sequences encoded in the three different schemes present a great different trend of the autocorrelation values. The sequences encoded under T-Sch or TM-Sch are verified to be at least locally stationary, i.e., the autocorrelation value is similar for a given length. However, the sequence encoded under M-Sch presents a non-stationary property, since the autocorrelation value varies greatly even when \( \lambda \) is shifted slightly. This comparison indicates that ignoring the dwelling at certain locations may result in difficulties in analyzing the sequences with most statistical tools, which presume the sequence to be at least locally stationary.

Secondly, excluding the plots in M-Sch, the other plots present two types of periodicity, daily or weekly. All the subfigures in third row in Figure A.2 present a high
Figure A.3: The plot of autocorrelation values for remaining four individuals according to the sequences from the three different schemes.
autocorrelation value roughly around $\lambda = 28$, $56$ or $84$, indicating that shifting the period of a week leads to the highest autocorrelation value. In other words, a weekly pattern is present. Similarly in Figure A.3, the daily periodicity can be observed for most of the individuals.

For the ergodicity property, which indicates that the statistical properties of the sequence can be learned from a sufficiently long sequence, it is hard to analyze without any prior knowledge of the sequence. Thus, I assume that the mobility sequence for each individual is long enough to estimate the statistical property of the source.

A.2.2.4 Markovian or i.i.d.

The issue of whether there exists a certain dependency in visiting the locations is resolved by analyzing the entropies. If the sequence is indeed generated from a particular type of source, the entropy of the sequence must have an entropy similar to that of the sequence randomly generated from that type of source.

Without knowing which type of the source the mobility sequence is generated from, it is a better practice to estimate the entropy rate of the sequence by using a model free method, e.g., LZ entropy estimator.

Let $\hat{H}_n(w)$ be the estimated entropy rate according to the window length $w$, within which the matching process is performed [GKB08]. The entropy is given by

$$\hat{H}_n(w) = \left( \frac{1}{n} \sum_{i=3}^{n} \frac{L_i^w}{\log n_i'} \right)^{-1} \quad (A.2)$$

where $n_i'$ is the effective window length. $L_i^w$ is the shortest substring starting at index $i$ that does not appear in the subsequence $x_{i-w}^{i-1}$ [GKB08]. The entropy of the sliding window estimator, $\hat{H}_{sw}$, is chosen from the minimum value according to the different choices of $w$, which at least satisfies the local stationarity of the sequence.

To determine whether the mobility sequence is generated from i.i.d. source or Markov source, I compare $\hat{H}_{sw}$ with the range of the entropies generated from the corresponding sources, where the parameters of the source are estimated according to the sequence.

Specifically, this validation takes the following steps. Firstly, estimate the parameters of the i.i.d. source and Markov source. Secondly, randomly generate sequences of the same length as the mobility sequence. Thirdly, get the range of the entropies of the source by estimating the entropy of the random sequences according to Eq (A.2). Lastly, compare the entropy rate of the mobility sequence with the range of the entropies of the source.

The i.i.d. source, indicating no dependency between consecutive locations, has the parameters $\{p(x)|x \in \Omega\}$ to generate each symbol. The maximum likelihood estimation of each parameter is
\[ \hat{p}(a_k) = \frac{|\{i : x_i = a_k, i \in [1, n]\}|}{n} \quad (A.3) \]

where \( k \in [1, n_A] \).

The Markov source, indicating a certain degree of dependency between the consecutive locations, has the parameters \( \{p(a_i|s_j)|a_i \in \Omega, s_j \in \Omega^k\} \), where \( k \) is the order of the Markov source. Specifically, in the first-order Markov model, given current context \( s_j \) and the next symbol \( a_i \), the parameter \( \hat{p}(a_i|s_j) \) is estimated by

\[ \hat{p}(a_i|s_j) = \frac{|\{t : a_{t-1} = a_j, a_t = a_i, 2 \leq t \leq n\}|}{|\{t : a_{t-1} = a_j, 2 \leq t \leq n\}|} \quad (A.4) \]

where \( s_j = a_j \) in the first-order Markov model.

Now I construct two null hypotheses to validate whether i.i.d. source or Markov source is more suitable for modeling the sequence.

- **Hypothesis One**: the sequence is generated from an i.i.d. source. The null hypothesis is \( \hat{H}_0 < \hat{H}_{sw} \), and the alternative hypothesis is \( \hat{H}_0 \geq \hat{H}_{sw} \).

- **Hypothesis Two**: the sequence is generated from a first-order Markov model. The null hypothesis is \( \hat{H}_1 < \hat{H}_{sw} \), and the alternative hypothesis is \( \hat{H}_1 \geq \hat{H}_{sw} \).

In the two null hypotheses, \( \hat{H}_0 \) and \( \hat{H}_1 \) are the corresponding lower 5% entropy value estimated from the randomly generated sequence from each source. The null hypothesis is rejected if \( \hat{H}_0 \) or \( \hat{H}_1 \) is larger than \( \hat{H}_{sw} \). Since the mobility sequences encoded in TM-Sch can best present stationarity, I validate the two null hypotheses on the sequence encoded from TM-Sch.

The results of the two hypotheses based on *Chon-LifeMap* are shown in Figure A.4. Among all the individuals, **Hypothesis One** is rejected, as \( \hat{H}_{sw} \) is much lower than the possible range of the entropy generated from the i.i.d. model. This result shows that there exist certain dependency among individuals’ mobility sequences. As for the **Hypothesis Two**, the 5 individuals show that \( \hat{H}_{sw} \) is slightly larger than the 5% tail of the random generated entropy from the assumed first-order Markov model. This is consistent with the theory that \( \hat{H}_{sw} \) is an upper bound of the entropy of the model [SBT06].

Although these two tests indicate that there exists certain dependency of visiting the locations, I have no evidence to further validate such dependency or the order of Markov model with high level of confidence. This is due to the lack of training data for estimating the parameters in the Markov model, which increases exponentially with the increasing of the order. Given a \( k^{th} \)-order Markov model with the size of alphabet \( n_A \), the possible number of free parameters is \( |n_A|^k/(|n_A| - 1) \). For instance, when \( n_A = 100 \) and \( k = 2 \), the corresponding number is \( 9.9 \times 10^5 \), which is much greater than the length
Figure A.4: (a) The testing results of Hypothesis One. The solid line is the range of the entropy randomly generated from the i.i.d. model. And the circle is the lower 5% tail. The diamond is the entropy rate of the sequence estimated by sliding window method. (b) The testing results of Hypothesis Two. The meaning of each symbol follows (a), and the difference is that the blue line is the range of the entropy generated from first-order Markov model.

of the sequences available (only a few thousands in my case). Nevertheless, even when \( k = 2 \), most of the transitions are not observed in actual mobility sequences. Therefore, in the following, I will not further consider the higher order \((k > 2)\) Markov models in the content of individuals’ mobility.

A.2.3 Modeling Individual’s Mobility

In this section, I consider the issue of modeling individual’s mobility by estimating the probability of the next symbol given the current context. Given the two candidate models, e.g., the first- and second-order Markov models, the corresponding context is either the current location or the current location and the location immediately preceding it. However, there is a case when the ”Markov state” is ineffective, which will be explained in the following subsection, thus three different revisiting rules are embedded to assign the probability of unvisited locations given current context.

Following the definition in [BEYY04], an accurate evaluation of a model is measured by the average log-loss function \( \ell(\hat{P}, x^n_1) = -\frac{1}{n} \sum_{i=1}^{n} \log \hat{P}(x_i|x_1 \cdots x_{i-1}) \), where the logarithm has a base of 2. Minimizing \( \ell(\hat{P}, x^n_1) \) is equivalent to maximizing the probability of observing the sequence. Therefore, the most suitable model achieves the lowest average log-loss value.

Some notations used in the following subsections are described here. Let \( c_d = x_{i-d+1}^i \) be the observed context of length \( d \) in the sequence. I do not differentiate the initial index \( i \) in the context. Let \( \Omega_{c_d} = \{ \sigma : N(c_d\sigma) > 0 \} \) denote the set of observed symbols after context \( c_d \), and \( \Omega'_{c_d} \) denote the remaining symbols.
A.2.3.1 The escape probability

Generally, in the Markov model, given current context \( c_d \), the probability of the next symbol \( \sigma \) is given by Eq (A.4) (for the second-order Markov model, the context is longer). If \( \sigma \notin \Omega_{c_d} \), \( P(\sigma|c_d) = 0 \). Since the symbol \( \sigma \) is novel for the context \( c_d \), and it is impossible to give an accurate estimation of \( P(\sigma|c_d) \). However, two recent studies offer some rules for assigning the probability of unobserved symbols. Firstly, Song et. al. [SKWB10] show that the individuals prefer to return to the previously visited locations. Therefore, it is suitable to assign the probability of unobserved symbols according to their frequencies. Secondly, Lee et. al. [LHK+09] show that when the individuals plan their trips, they tend to minimize the travel distance. Therefore, it is reasonable to assign higher probabilities to nearby locations. Also, the probability for the unobserved symbols may change at different stages. For instance, at the initial stage, there is a high probability of observing unobserved symbols, but when the training sequence is long, there is a low probability to observe the novel symbols. Therefore, I choose the probability estimation method in PPM-C [Mof90].

Specifically, given context \( c_d \) the probability of the observed symbol is given by

\[
\hat{P}(\sigma|c_d) = \frac{N(c_d\sigma)}{|\Omega_{c_d}| + \sum_{\sigma'\in\Omega_{c_d}} N(c_d\sigma')}
\] (A.5)

where \( \sigma \in \Omega_{c_d} \).

The escape probability for the novel symbols is given by

\[
\hat{P}(e|c_d) = 1 - \sum_{\sigma\in\Omega_{c_d}} \hat{P}(\sigma|c_d) = \frac{|\Omega_{c_d}|}{|\Omega_{c_d}| + \sum_{\sigma'\in\Omega_{c_d}} N(c_d\sigma')}
\] (A.6)

When \( |\Omega_{c_d}| \) is comparable to the count of the context \( c_d \), \( \hat{P}(e|c_d) \) is high. With the count of the context \( c_d \) greatly exceeding \( |\Omega_{c_d}| \), \( \hat{P}(e|c_d) \) becomes smaller. In the following, I develop the three rules for assigning the probability for each \( \sigma \in \Omega'_{c_d} \).

A.2.3.2 The revisiting rules

Three rules are proposed to measure the probability of revisiting the novel symbols based on either temporal or spatial attribute, or a combination of the two.

Rule One: revisit according to visitation frequency. In this rule, given the visitation frequency of each location \( f_i \), where \( i \in [1, n_A] \), the probability of the unobserved symbol \( x_k \) with context \( c_d \) is given by

\[
\hat{P}(x_k|c_d) = \hat{P}(e|c_d) \frac{f_k}{\sum_{x_i\in\Omega_{c_d}} f_i}
\] (A.7)
**Rule Two:** revisit according to the distance to the current location. In this rule, given the distance of location $x_k$ to the current location $d(c_d, k)$, then

$$\hat{P}(x_k|c_d) = \hat{P}(e|c_d) \frac{d(c_d, k)^\alpha}{\sum_{x_i \in \Omega'_{c_d}} d(c_d, i)^\alpha} \quad (A.8)$$

where the distance $d(c_d, k)$ is calculated according to the Euclidean distance between two grids, and the location of each grid is given by its index in both latitudinal and longitudinal direction. In my case, I test $\alpha \in \{0, -1, -2\}$.

**Rule Three:** revisit according to both visitation frequency and distance to current location. In other words,

$$\hat{P}(x_k|c_d) = \hat{P}(e|c_d) \frac{f_kd(c_d, k)^\alpha}{\sum_{x_i \in \Omega'_{c_d}} f_id(c_d, i)^\alpha} \quad (A.9)$$

In the experimental part, I will evaluate the log-loss value $\ell(\hat{P}, x^n_i)$ according to each rule. Comparing $\ell(\hat{P}, x^n_i)$ will reveal the characteristics of individuals’ mobility behaviors.

### A.2.4 Model Evaluation

In this subsection, I evaluate the first- and second-order Markov models, along with three revisiting rules in modeling individuals’ mobility sequences. The evaluation uses both Chon-LifeMap and Zheng-GeoLife datasets (see Appendix A.1 for details).

![Figure A.5: The fraction of the predictions that apply the revisiting rules. The center of each vertical line is the mean of the fraction over all the individuals, and the half length of the line segments indicates the standard deviation.](image)

Figure A.5: The fraction of the predictions that apply the revisiting rules. The center of each vertical line is the mean of the fraction over all the individuals, and the half length of the line segments indicates the standard deviation.

Figure A.5 firstly shows the fraction of the predictions that apply the revisiting rules, where this fraction is defined as, $p_e(i) = |\{x_i | x_i \in \Omega'_{c_d}, i \leq n(i)\}|/n(i)$, where $n(i) = np(i)$. And the context $c_d$ varies according to the index $i$. 

134
Initially, in Chon-LifeMap, the mean fraction of the predictions applies the revisiting rule is about 45% in the first-order Markov model and 60% in the second-order Markov model, and in Zheng-GeoLife, the mean fraction is as high as 60% in the first-order Markov model, and 70% for the second-order Markov model. The mean fraction will decrease with the increasing length of the sequence. This fact shows that it is necessary to introduce the revisiting rules in modeling individuals’ mobility to handle the case where a novel symbol is observed after the current context. Also it is reasonable to vary the escape probability, given in Eq (A.6), according to the number of observed symbols, which may gradually increase with the length of the sequence.

Table A.1: The results of average log-loss on Chon-LifeMap. Note that in Rule Three when \( \alpha = 0 \), it is equivalent to Rule One. And in the rule that achieves the minimum average log-loss value is highlighted.

<table>
<thead>
<tr>
<th>R1</th>
<th>O1</th>
<th>O2</th>
<th>R2</th>
<th>O1</th>
<th>O2</th>
<th>R3</th>
<th>O1</th>
<th>O2</th>
</tr>
</thead>
<tbody>
<tr>
<td>n.a.</td>
<td>4.61</td>
<td>4.95</td>
<td>3.33</td>
<td>3.65</td>
<td>2.57</td>
<td>2.73</td>
<td>2.83</td>
<td>2.99</td>
</tr>
<tr>
<td>R2</td>
<td>4.76</td>
<td>5.59</td>
<td>4.06</td>
<td>4.86</td>
<td>2.59</td>
<td>3.07</td>
<td>2.82</td>
<td>3.25</td>
</tr>
<tr>
<td>-1</td>
<td>4.34</td>
<td>4.74</td>
<td>3.02</td>
<td>3.36</td>
<td>2.30</td>
<td>2.50</td>
<td>2.48</td>
<td>2.67</td>
</tr>
<tr>
<td>-2</td>
<td>4.55</td>
<td>4.91</td>
<td>3.01</td>
<td>3.26</td>
<td>2.44</td>
<td>2.60</td>
<td>2.45</td>
<td>2.55</td>
</tr>
<tr>
<td>R3</td>
<td>4.20</td>
<td>4.32</td>
<td>3.02</td>
<td>3.11</td>
<td>2.37</td>
<td>2.33</td>
<td>2.53</td>
<td>2.56</td>
</tr>
<tr>
<td>-1</td>
<td>4.41</td>
<td>4.65</td>
<td>2.98</td>
<td>3.09</td>
<td>2.42</td>
<td>2.47</td>
<td>2.45</td>
<td>2.46</td>
</tr>
<tr>
<td>-2</td>
<td>4.20</td>
<td>4.32</td>
<td>3.02</td>
<td>3.11</td>
<td>2.37</td>
<td>2.33</td>
<td>2.53</td>
<td>2.56</td>
</tr>
</tbody>
</table>

Table A.1 evaluates the adequacy of the models for individuals’ mobility in terms of the average log-loss value using Chon-LifeMap, and Figure A.6 for Zheng-GeoLife.

Firstly, a consistent trend is observed from both Table A.1 and Figure A.6. The top two highest values of \( \ell(\hat{P}, x^n) \) among all the individuals are achieved in both Markov models combining with either Rule One or Rule Two given \( \alpha = 0 \). Note that in Rule Two given \( \alpha = 0 \), it assigns the unobserved locations with equal probability. This fact shows that excluding the spatial locality i.e., the distance of each novel location to the current location, will lead to a large bias in estimating the probability of the next whereabouts.

Secondly, the minimum value of \( \ell(\hat{P}, x^n) \) is achieved in different cases in the two datasets. In Chon-LifeMap, in most cases, the minimum value of \( \ell(\hat{P}, x^n) \) falls into Rule Three with the same parameter setting, i.e., same value of \( \alpha \), in both first- and second-order Markov models except for individual GS2 and GS9. However, for individual GS2 and GS9, the minimum \( \ell(\hat{P}, x^n) \) achieved in the first-order Markov model under Rule Three is slightly higher, e.g., 0.01, and 0.02, respectively, than the corresponding minimum value. For individual GS7, the value achieved given by Rule Three is slightly higher, about 0.07, than the corresponding minimum value. Thus, it is generally consistent to conclude that the Markov model embedded with Rule Three is the best model for describing individuals’ mobility. However, the best choice of the parameter \( \alpha \) is different among the individuals. It is justifiable to conclude that combining the information of
Figure A.6: The results of average log-loss on Zheng-GeoLife. Both (a) and (b) plot the results under all the cases in two models, respectively. While (c) is the plot in both model under Rule Two and Rule Three where $\alpha = -2$. 
visiting frequency and the distance to the current location is able to accurately model individuals’ mobility and to predict the next whereabouts.

In Zheng-GeoLife, the minimum value is achieved under the Rule Two given $\alpha = -2$ among all the individuals. This shows that merely applying the information about the distance of each location to the current location is the most effective rule in modeling individuals’ mobility, and higher weights should be assigned to the locations nearby.

Moreover, Figure A.6(c) further compares the performance of the first- and the second-order Markov model under Rule Two and Rule Three given $\alpha = -2$. As expected, the higher-order Markov model achieves lower average log-loss value in all the cases, and thus it is more effective in predicting the next whereabouts. However, Figure A.5 reminds us that increasing the order of Markov model will increase the fraction of the predictions of novel symbols. The proposed Rule Two and Rule Three are shown to be effective in predicting the novel symbols given current context. However, I still prefer to reduce the fraction of such predictions which has lower predicting accuracy than the predictions that are based on previous observations. Thus the choice of the order of Markov model in predictions should be a balance between the prediction accuracy of the observed symbols and the fraction of the prediction of the novel symbols.

The two datasets capture individuals’ mobility in different respects. The Chon-LifeMap records both the daily routes and long dwelling at certain places, while the Zheng-GeoLife merely contains the daily routes, the optimal revisiting rule generated from Chon-LifeMap is more applicable to the general scenarios.

A.2.5 Summary

In this section, I study the two statistical properties, i.e., stationarity and Markovian, of individuals’ mobility. By comparing three different schemes in encoding individuals’ mobility sequence, I show that under appropriate choice of spatial scale and temporal scale, the stationarity and the periodicity of individuals’ mobility can be best presented. Also, by testing two hypotheses, I show that a certain dependency exists between the visitations to the locations. Based on these two statistical properties, I present two Markov models with three different revisiting rules, aiming to handle the issue of the unobserved symbols given current context, to model individuals’ mobility. Experimental results show that the rule that combines both the information of visiting frequency and the distance of location relative to the current location can best model individuals’ mobility. This finding should be a useful refinement to the result in [SKWB10], which reports that individuals’ return preference is merely based on visit frequencies.
A.3 Proof of Lemmas

A.3.1 Proof of Lemma 5.2

Note that the visitation frequency of the $i^{th}$ location is proportional to $i^{-\zeta}$, and $c \sum_{i=1}^{S} \frac{1}{i^\zeta} = 1$.

The sum of all the frequencies is given by

$$\sum_{i=1}^{S} \frac{1}{i^\zeta} = 1 + \sum_{i=2}^{S} \frac{1}{i^\zeta}$$  \hspace{1cm} (A.10)

$$> 1 + \sum_{i=2}^{S} \int_{i}^{i+1} \frac{1}{x^\zeta} dx$$  \hspace{1cm} (A.11)

$$= 1 + \int_{2}^{S+1} \frac{1}{x^\zeta} dx$$  \hspace{1cm} (A.12)

$$= 1 + \frac{1}{1-\zeta} [(S+1)^{1-\zeta} - 2^{1-\zeta}]$$  \hspace{1cm} (A.13)

Similarly,

$$\sum_{i=1}^{S} \frac{1}{i^\zeta} < 1 + \sum_{i=2}^{S} \int_{i-1}^{i} \frac{1}{x^\zeta} dx$$  \hspace{1cm} (A.14)

$$= \frac{S^{1-\zeta} - \zeta}{1-\zeta}$$  \hspace{1cm} (A.15)

Thus

$$\frac{1-\zeta}{1-\zeta} < c < \frac{1-\zeta}{1-\zeta + (S+1)^{1-\zeta} - 2^{1-\zeta}}$$

Replacing $\zeta$ by $1 + \gamma$ according to the case of $\gamma > 0$, an approximation of $c$ is given by the mean value of the upper bound and the lower bound.

Figure A.7 shows that the value estimated using Eq. (5.5) is much closer to the actual value of $c$ than either the lower bound or the upper bound. Note that when calculating the bounds of $c$, the number of distinct locations visited within $n$ steps is given by $S = (1 + \gamma)^{1/(1+\gamma)} (\rho n)^{1/(1+\gamma)}$.

A.3.2 Proof of Lemma 5.3

Let $\delta(x) = c(1 - \rho x)$. Then, $\Pi(S) = \delta(S^{-\gamma})$, where $0 < S^{-\gamma} < 1$. When replacing the value of $c$ from Eq. (2)

$$\delta(x) \approx \frac{\gamma}{2} (1 - \rho x) \left[ \frac{1}{1 + \gamma - x} + \frac{1}{\gamma + 2^{-\gamma} - x} \right]$$  \hspace{1cm} (A.16)

where $S^{-\gamma} \approx (S + 1)^{-\gamma}$ when $S$ is large (e.g. $S > 10$).
Figure A.7: Comparisons of the values of $c$ for different values of $\gamma$. The figure shows the actual value, estimated value, upper bound and lower bound, where $\rho = 0.1$ and $n = 10000$.

$$\delta(x) \approx \frac{\gamma}{2} (1 - \rho x) \left[ \frac{1}{1 + \gamma - x} + \frac{1}{\gamma + 2^{-\gamma} - x} \right]$$  
(A.17)

$$= \frac{\gamma}{2} \left[ 2\rho + \frac{1 - \rho - \rho \gamma}{1 + \gamma - x} + \frac{1 - \rho \gamma - \rho 2^{-\gamma}}{\gamma + 2^{-\gamma} - x} \right]$$  
(A.18)

Since $1 + \gamma > \gamma + 2^{-\gamma} > 1 > S^{-\gamma}$ and $1 > x > 0$, the denominators in Eq. (A.18) are both greater than 0. Similarly, $1 - \rho - \rho \gamma < 1 - \rho \gamma - \rho 2^{-\gamma}$. The three cases in Eq. (5.20) are considered when bounding the predictability.

$$\begin{cases} 
0 \leq 1 - \rho - \rho \gamma < 1 - \rho \gamma - \rho 2^{-\gamma}, & \text{Case 1} \\
1 - \rho - \rho \gamma < 0 < 1 - \rho \gamma - \rho 2^{-\gamma}, & \text{Case 2} \\
1 - \rho - \rho \gamma < 1 - \rho \gamma - \rho 2^{-\gamma} \leq 0, & \text{Case 3} 
\end{cases}$$  
(A.19)

In Case 1, both the numerators in Eq. (A.18) are greater than 0. When $S$ increases, $x = S^{-\gamma}$ decreases, thus $\delta(x)$ decreases. Thus

$$\forall i \geq 2 \Pi(i) \leq \delta(2^{-\gamma})$$

$$= \frac{\gamma}{2} (1 - \rho 2^{-\gamma}) \left[ \frac{1}{1 + \gamma - 2^{-\gamma}} + \frac{1}{\gamma + 2^{-\gamma} - 3^{-\gamma}} \right]$$

$$< \frac{\gamma}{2} (1 - \rho 2^{-\gamma}) \left[ \frac{1}{\gamma} + \frac{1}{\gamma} \right]$$

$$= 1 - \rho 2^{-\gamma}$$  
(A.20)

where in this case $S$ is very small, I do not treat $S^{-\gamma}$ and $(S + 1)^{-\gamma}$ as being equal.
The predictability of the model is bounded by

$$\Pi = \sum_{i=2}^{S} \frac{i^{1+\gamma} - (i-1)^{1+\gamma}}{S^{1+\gamma}} \Pi(i)$$  \hspace{1cm} (A.21)

$$< \sum_{i=2}^{S} \frac{i^{1+\gamma} - (i-1)^{1+\gamma}}{S^{1+\gamma}} \Pi(2)$$  \hspace{1cm} (A.22)

$$= \frac{\Pi(2)}{1} < 1 - \rho 2^{-\gamma}$$  \hspace{1cm} (A.23)

In Case 2

$$\frac{1}{1+\gamma} < \rho < \frac{1}{\gamma + 2^{-\gamma}}$$  \hspace{1cm} (A.24)

Let

$$g(\rho) = \frac{1 - \rho - \rho \gamma}{1+\gamma - i^{-\gamma}} + \frac{1 - \rho \gamma - \rho 2^{-\gamma}}{\gamma + 2^{-\gamma} - i^{-\gamma}}$$  \hspace{1cm} (A.25)

which decreases when $\rho$ increases.

Then

$$g(\rho) < g\left(\frac{1}{1+\gamma}\right)$$  \hspace{1cm} (A.26)

$$= \frac{1}{1+\gamma} \left[\frac{(1+\gamma) - (\gamma + 2^{-\gamma})}{\gamma + 2^{-\gamma} - i^{-\gamma}}\right]$$  \hspace{1cm} (A.27)

Therefore, the predictability is bounded by

$$\forall_i \Pi(i) = \frac{\delta(i^{-\gamma})}{2}\left[2\rho + g(\rho)\right]$$  \hspace{1cm} (A.28)

$$< \rho \gamma + \frac{1}{2} \frac{\gamma}{1+\gamma} \left[\frac{(1+\gamma) - (\gamma + 2^{-\gamma})}{\gamma + 2^{-\gamma} - i^{-\gamma}}\right]$$  \hspace{1cm} (A.29)

$$< \rho \gamma + \frac{1}{2} \frac{\gamma}{1+\gamma} \left[\frac{(1+\gamma) - (\gamma + 2^{-\gamma})}{\gamma + 2^{-\gamma} - \gamma^{-\gamma}}\right]$$  \hspace{1cm} (A.30)

$$= \rho \gamma + \frac{1}{2} \frac{(1+\gamma) - (\gamma + 2^{-\gamma})}{1+\gamma}$$  \hspace{1cm} (A.31)

$$< \frac{\gamma}{\gamma + 2^{-\gamma}} + \frac{1}{2} \frac{(1+\gamma) - (\gamma + 2^{-\gamma})}{1+\gamma}$$  \hspace{1cm} (A.32)

$$= \frac{\gamma}{\gamma + 2^{-\gamma}} + \frac{1}{2} \left[1 - \frac{\gamma + 2^{-\gamma}}{1+\gamma}\right]$$  \hspace{1cm} (A.33)

Similarly, the overall predictability is bounded by

$$\Pi < \frac{\gamma}{\gamma + 2^{-\gamma}} + \frac{1}{2} \left[1 - \frac{\gamma + 2^{-\gamma}}{1+\gamma}\right]$$  \hspace{1cm} (A.34)
In Case 3
\[ \rho \geq \frac{1}{\gamma + 2^{-\gamma}} \] (A.36)

Thus both numerators in Eq. (A.25) are less than 0. By dropping the negative part, \( \delta(x) < \rho \gamma \). Therefore, for all \( i > 1 \), \( \Pi(i) = \delta(i^{-\gamma}) < \rho \gamma \). A general upper bound is given by \( \Pi(\rho = \rho_i) < \rho_i \gamma \), where \( \Pi(\rho = \rho_i) \) indicates the predictability achieved when the parameter \( \rho = \rho_i \). According to Eq. (A.22), the predictability will decrease when increasing \( \rho \). Therefore, subject to Ineq. (A.36)

\[ \Pi(\rho = \rho_i) < \Pi(\rho = \min \{ \rho_i \mid \forall \rho_i \geq \frac{1}{\gamma + 2^{-\gamma}} \}) = \frac{\gamma}{\gamma + 2^{-\gamma}} \] (A.37)

which is a general bound in Case 3.

This bound is tight when \( \gamma \) is very small, e.g., \( \gamma < 1 \). However, when \( \gamma \) is very large, e.g., \( \gamma > 1 \), the bound is close to 1. Therefore, in the following, a tighter bound is explored in the case when \( \gamma \) is large.

Since \( \delta(x) \) increases with \( S \), then

\[ \forall i \Pi(i) \leq \delta(S^{-\gamma}) \]
\[ = \frac{1}{2} \gamma (1 - \rho S^{-\gamma}) \left[ \frac{1}{1 + \gamma - S^{-\gamma}} + \frac{1}{\gamma + 2^{-\gamma} - S^{-\gamma}} \right] \] (A.39)
\[ < \frac{1}{2} \gamma \left[ \frac{1}{1 + \gamma - 2^{-\gamma}} + \frac{1}{\gamma + 2^{-\gamma} - 2^{-\gamma}} \right] \] (A.40)
\[ = \frac{1}{2} + \frac{1}{2 \gamma} \left[ \frac{\gamma}{1 + \gamma - 2^{-\gamma}} \right] \] (A.41)

Thus, generally I have

\[ \Pi < \frac{1}{2} + \frac{1}{2 \gamma} \left[ \frac{\gamma}{1 + \gamma - 2^{-\gamma}} \right] \] (A.42)

Combining Ineq. (A.37) and Ineq. (A.42)

\[ \Pi < \min \left\{ \frac{\gamma}{\gamma + 2^{-\gamma}}, \frac{1}{2} + \frac{1}{2 \gamma} \left[ \frac{\gamma}{1 + \gamma - 2^{-\gamma}} \right] \right\} \] (A.43)

A.3.3 Proof of Lemma 6.8

Recall that

\[ g(x) = \frac{c_k}{a_k x^2 + b_k^2} \] (A.44)

The second order derivative of \( g(x) \) is given by
\[ \frac{d^2 g(x)}{dx^2} = 2a_k c_k (a_k x^2 + b_k^2)(3a_k x^2 - b_k^2) \] (A.45)

The values of the constant \( a_k, b_k, \) and \( c_k \) are given by

\[
\begin{align*}
  a_k &= t'_{2k} \alpha_k (1 - \alpha_k) \quad \text{(A.46)} \\
  b_k^2 &= \alpha_k^2 \sigma_g^2 + (1 - \alpha_k)^2 \sigma_g^2 \quad \text{(A.47)} \\
  c_k &= (p_{2k} - \tilde{\mu}_{2k})^2 \quad \text{(A.48)}
\end{align*}
\]

Therefore, we have to check whether \( \frac{d^2 g(x)}{dx^2} > 0 \) for all \( x \geq \sigma_g \). And \( \frac{d^2 g(x)}{dx^2} > 0 \) is equivalent to \( (3a_k x^2 - b_k^2) > 0 \), which is given by

\[
\begin{align*}
  3a_k x^2 - b_k^2 &= 3t'_{2k} \alpha_k (1 - \alpha_k) x^2 - (\alpha_k^2 \sigma_g^2 + (1 - \alpha_k)^2 \sigma_g^2) \\
  &\geq 3t'_{2k} \alpha_k (1 - \alpha_k) \sigma_g^2 - (\alpha_k^2 \sigma_g^2 + (1 - \alpha_k)^2 \sigma_g^2) \\
  &= \sigma_g^2 (3t'_{2k} \alpha_k (1 - \alpha_k) - (\alpha_k^2 + (1 - \alpha_k)^2)) \quad \text{(A.49)}
\end{align*}
\]

Note that \( \alpha_k = \frac{t_{2k} - t_{2k-1}}{t_{2k+1} - t_{2k-1}} \) and \( t'_{2k} = t_{2k+1} - t_{2k-1} \), where all the time \( t_x \) is in terms of seconds. The minimum value of \( 3t'_{2k} \alpha_k (1 - \alpha_k) \) is

\[
3t'_{2k} \alpha_k (1 - \alpha_k) > 3t'_{2k} \frac{1}{t'_{2k}} \frac{t'_{2k} - 1}{t'_{2k}} = 3(t'_{2k} - 1) \quad \text{(A.52)}
\]

The maximum value of \( (\alpha_k^2 + (1 - \alpha_k)^2) \) is

\[
\alpha_k^2 + (1 - \alpha_k)^2 < \left( \frac{1}{2} \right)^2 + \left( \frac{1}{2} \right)^2 = \frac{1}{2} \quad \text{(A.53)}
\]

Thus \( 3a_k x^2 - b_k^2 > 0 \) and the function \( g(x) \) is convex in my case according to the theory of convexity [BV04].
Appendix B

Publication List

- **Miao Lin** and Wen-Jing Hsu, “Brownian Bridge Model for High Resolution Location Predictions”, *18th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, 2014.


- **Miao Lin**, Wen-Jing Hsu, and Zhuo Qi Lee, “Modeling the High Predictability and Scaling Laws of Human Mobility”, *The First International Workshop on Human Mobility Computing (HuMoComp) held in conjunction with the 14th IEEE MDM*, 2013.

- **Miao Lin**, Wen-Jing Hsu, and Zhuo Qi Lee, “Predictability of Individuals’ Mobility with High-Resolution Positioning Data”, *14th ACM International Conference on Ubiquitous Computing (UbiComp)*, Pittsburgh, 2012.

CHAPTER B. PUBLICATION LIST
References


REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


[HDY99] Jiawei Han, Guozhu Dong, and Yiwen Yin. Efficient Mining of Partial Periodic Patterns in Time Series Database. In Proc. of the 15th Int. Conf. on Data Engineering, ICDE ’99, pages 106–115, 1999.
REFERENCES


[HK06] Jiawei Han and Micheline Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 2006.


REFERENCES


REFERENCES

[LDH+10] Zhenhui Li, Bolin Ding, Jiawei Han, Roland Kays, and Peter Nye. Mining Periodic Behaviors for Moving Objects. In Proc. of the 16th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, KDD ’10, pages 1099–1108, 2010.

[LDHK10] Zhenhui Li, Bolin Ding, Jiawei Han, and Roland Kays. Swarm: Mining Relaxed Temporal Moving Object Clusters. Proc. VLDB Endow., 3(1-2):723–734, 2010.


REFERENCES


[LHLC11] Jae-Gil Lee, Jiawei Han, Xiaolei Li, and Hong Cheng. Mining Discriminative Patterns for Classifying Trajectories on Road Networks. IEEE Transactions on Knowledge and Data Engineering, 23(5):713–726, 2011.

[LHLG08] Jae-Gil Lee, Jiawei Han, Xiaolei Li, and Hector Gonzalez. TraClass: trajectory classification using hierarchical region-based and trajectory-based clustering. Proc. VLDB Endow., 1(1):1081–1094, 2008.

[LJL+10] Zhenhui Li, Ming Ji, Jae-Gil Lee, Lu-An Tang, Yintao Yu, Jiawei Han, and Roland Kays. MoveMine: Mining Moving Object Databases. In Proc. of the 2010 ACM SIGMOD Int. Conf. on Management of Data, SIGMOD ’10, pages 1203–1206, 2010.


REFERENCES


[LWH12] Zhenhui Li, Jingjing Wang, and Jiawei Han. Mining Event Periodicity from Incomplete Observations. In *Proc. of the 18th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, KDD ’12, pages 444–452, 2012.


REFERENCES


REFERENCES


REFERENCES


REFERENCES


