Universidad
Carlos III de Madrid

EPS POLYTECHNIC SCHOOL

Department of Telematic Engineering

Ph.D. Dissertation

Contributions to the Understanding of Human Mobility and Improvement of Lightweight Mobility Prediction Algorithms

Author: Alicia Rodríguez Carrión
Co-Advisor: Dr. María Celeste Campo Vázquez
Co-Advisor: Dr. Carlos García Rubio

November, 2015

Contributions to the Understanding of Human Mobility and Improvement of Lightweight Mobility Prediction Algorithms

By

Alicia Rodríguez Carrión

Directed By

Dr. María Celeste Campo Vázquez
Dr. Carlos García Rubio

A Dissertation Submitted to the
Department of Telematic Engineering
and the Committee on Graduate Studies
in Partial Fulfillment of the Requirements for the Degree of
DOCTOR OF PHILOSOPHY

Approved by the
Supervisory Committee:

General Chair

_____

Chair

_____

Secretary

_____

Grade:

_____

Leganés, ___ , _____ _____

Acknowledgments

Han sido muchas las personas que, por fortuna, me han acompañado (y empujado a seguir adelante) durante esta andadura. Todas ellas se merecen mi más sincero agradecimiento, que quiero expresar en estas líneas.

En primer lugar, me gustaría agradecer especialmente a mis tutores, Celeste y Carlos, vuestra enorme involucración, no sólo profesional, sino sobre todo personal. Gracias por todo el tiempo, esfuerzo, apoyo, confianza, consejos y ayuda que me habéis dedicado incondicionalmente, por enseñarme tanto, y porque no creo que se pueda pedir más que todo lo que me habéis aportado. Gracias también a Florina, Andrés y Dani, por vuestro apoyo constante. A mis chicas pervasivas, Estrella y Patri, por tantos ratos de reflexión, ánimo o, sencillamente, de ponerle buena cara al día. A Alberto, por ser tan estupendo compañero de despacho y por todo lo que he aprendido estos años trabajando contigo.

También me gustaría dar las gracias a los compañeros del departamento de Ingeniería Telemática. A Carlos Delgado, por darme la oportunidad de trabajar en el grupo GAST. A los profesores con los que he tenido la suerte de compartir docencia, por enseñarme año tras año a defenderme en el peculiar mundo de la enseñanza. Gracias a M.Carmen Fernández Panadero, por la "hiper-empatía" y otras valiosas reflexiones. A mis compañeros de despacho (y de tutorías, y de alegrías y penas doctorales), Isra, Derick y Damaris, porque hicimos del apoyo y el buen humor la ley de "Siberia". Gracias también a los innumerables compañeros que han formado parte de nuestras filas a lo largo de estos años. Por los cafés y tantos otros buenos ratos que he tenido la grandísima suerte de compartir y disfrutar con todos vosotros.

Quería agradecer también la colaboración de todos aquellos que os prestasteis tan amablemente a recolectar los datos que forman parte de esta tesis.

I am very thankful to the international experts that reviewed the dissertation, Dr. Adnan Khan, Dr. Qizhi Zhang, and Dr. Andrea Saracino, for your time and valuable comments. I would like to thank also the committee members, for kindly accepting to be part of the defense process and for devoting time to it.

I would like to express my most sincere gratitude to Dr. Sajal K. Das, for making my research visit possible, and for devoting time to discuss ideas and help to improve my work. I wish also to give thanks to the colleagues I met at Missouri University of Science and Technology, and specially to Andrea and Armita, who made my visit much more valuable, since I gained two friends.

Gracias a Dim, por su apoyo y paciencia, sobre todo durante la última etapa, que se ha hecho tan cuesta arriba. Por animarme a seguir y por mostrarme siempre puntos de vista

tan diferentes.

Gracias a mis queridas hermanas, Lau y Vir, por estar siempre ahí (incluso después de tantos años), con una sonrisa por bandera sin importar la situación. A María, por las incontables conversaciones y reflexiones, en vivo y en la distancia, sobre tan diversos temas (y los que nos quedan). A Rubén, por nuestro realismo compartido, y el apoyo a base de música, recetas y humor. A Patri Uriol, por ser paciente con mis ausencias y seguir ahí a pesar de ellas.

Gracias a mi familia, por sus ánimos constantes. Pero sobre todo, mi mayor agradecimiento está dedicado, sin duda, a mis padres, por inculcarme desde siempre el afán trabajador, la bondad, la reflexión y el ingenio. Y sobre todo, por poner todo lo que ha estado en vuestra mano para que tomara y siguiera el camino que en cada momento decidí emprender.

Abstract

Human mobility is key in fields like urban planning, protocols for mobile networks, or service personalization, among others. Besides, a large number of studies emerged in the last years thank to more complete mobility data sets, coming from the use of mobile phones as mobility proxies that continuously record their owners' locations. Traditionally, many of the applications of human mobility, like service personalization, were based on the current location of the user. However, this focus has recently started to shift to the user's mobility habits and her future location. This change allows having time in advance to provide services related to the usual habits of the user.

This thesis focuses on broadening the understanding of human mobility through the analysis of the location data recorded by mobile devices, and finding ways to increment the probability of making right predictions about their future locations. To confront this challenge, it is divided into three stages: the mobility data collection, the extraction and analysis of the mobility features reflected into the recorded data, and the analysis of a set of prediction algorithms to propose some improvements. The intrinsic privacy risks associated to the disclosure of the location and mobility data of the user are also considered.

In the first stage, the analysis of the sensors available in mobile devices and the requirements of the thesis lead to choose the cellular network as the source of mobility data. After analyzing the existing data sets containing this kind of data, it is decided to carry out a new mobility data collection campaign to obtain a more complete data set.

The second stage is focused on extracting mobility features from the data chosen in the previous step, and spot the biases introduced by the data collection scheme. In order to eliminate these biases, several filtering techniques are proposed to delete the maximum number of events not representing the movement of the user.

For the next stage, the specific family of LZ-based prediction algorithms is chosen to analyze their results when using mobility data obtained using different schemes and then filtered. By leveraging the mobility features studied in the previous stage, and based on their relationship with the prediction results, several modifications of the original algorithms are proposed to increase the fraction of right predictions.

Finally, in the privacy preservation plane, the shift from disclosing static location profiles to mobility profiles leads to the proposal of a new privacy metric, based on the concept of entropy rate. The goal is to consider both the spatial as well as temporal information in a mobility profile. Some privacy-enhancing perturbation techniques are tested with both location and mobility profiles using the new privacy metric, which unveils the noticeable amount of information stored in the temporal correlations.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Contents

1.1 Motivation

The human need for mobility has driven the development of communication technologies such as wireless networks, which fast growth and evolution during the last decades demonstrates the need for keeping on seamlessly communicating in any possible way (calls, messaging, Internet access, e-mail, etc.) while moving. As an example, cellular telephony networks have grown until reaching 7.53 billion mobile connections, including machine-to-machine (M2M) connections[1], which exceeds the worldwide population, which reaches around 7.27 billion of inhabitants[2]. Focusing on just the number of unique mobile subscribers, the numbers drop down to 3.74 billion unique mobile subscribers[3], which means more than half of the planet population carrying a mobile device with them. The penetration of mobile devices is so high, and they have evolved so quickly in terms of computational capabilities, as well as in the number and variety of integrated sensors, that they have become a powerful tool to monitor their original driving force: human mobility.

Mobile devices have demonstrated to be key mobility sensors that allow to investigate global mobility-related topics such as the spread of infectious biological [27, 57] and mobile viruses [149], the habits of people around a city [9], or the impact of mobility on wireless communications [113, 80], among many others. Mobility at individual level also unveils interesting information, since the subsequent locations a person visits define her in many ways. For instance, the location data of a user provide many clues about her usual activities (dining out, going to theaters, sport centers, church, hospitals, etc.), if she moves to

[1] According to the statistics published by https://gsmaintelligence.com as of October, 2015.
[2] According to the statistics published by http://www.census.gov/popclock/ as of October, 2015.
[3] According to the statistics published by https://gsmaintelligence.com as of October, 2015.

many places or mainly stays home, if she travels, etc. For this reason, location is usually one of the main elements of the context used to tailor the behavior of applications to each specific user. Personalization is key for applications and services to be attractive for users, and location is a multifaceted reference to personalize applications: from the most typical example of searching restaurants or other venues close to the current user's location, or finding routes from that current location to a destination, to different configurations dependent on the location (e.g., silent mode at work). This type of location-based services (LBSs) focused in the current location, which is considered independently to any other previous or future location, have been very popular so far. However, LBSs are already evolving: the focus has started to shift from the current location of the user to the sequence of visited locations and mobility patterns in general, considered as the current location together with the whole implicit and explicit details enclosed in the mobility tracking data. The current location provides just a small piece of information, but continuously tracking the user and appropriately mining the resulting sequence of locations disclose much more information than just the visited, isolated, locations: how much or far away the user moves (to manage battery consumption); the places the user visits consistently (such as the work place, children's schools, hospitals, religious or political venues), which provides information far beyond the visited locations; the routes to get to those places (which can serve to prevent the user from getting into traffic jams or pubic transport breakdowns, and let choosing a different route in advance); or when the user falls out form her usual habits (meaning she may need some assistance through maps, or warning data about a suspicious behavior). There already exist some applications taking advantage of this type of knowledge, like Google Now[4]. This application performs data mining over the mobility information collected through the mobile phone usage (location, calendar, contacts, maps, etc.), thus being able to inform the user how much time it will take her to get from home to work, which means that the application previously learned which location corresponds to her home and her job, and her usual route between these places. However, the application does not provide this information on weekends because it learnt the days in which the user follows this home-job route, and the days in which she does not. This is clearly an evolutionary step with respect to the classical LBSs. However, in order to intelligently provide this tailored information, an evolution on the underlying foundation technologies, to mine the mobility patterns and predict the most probable next locations, is required as well.

With all the new data captured by mobile phones about user mobility, it became clear that classical mobility models, such as Random Walk [103] or Random Waypoint [64], among others, fall short to capture the real features driving human mobility [14, 113]. Thus, this huge amount of location data captured using mobile devices as monitoring tool needs to be carefully considered to determine which mobility-related information is able provide, and their limitations, so that more accurate conclusions about mobility can be derived from it. Moreover, it is demonstrated that people behave somehow similarly, yet existing noticeable differences between individuals' behavior that must be captured to personalize services for each specific user. This task requires different metrics to faithfully capture the user-specific mobility aspects. In order to provide services like the ones mentioned before,

[4]Accessible at https://www.google.com/landing/now/, as of October, 2015.

it is not enough to consider general features, but also transient ones: where the user is now, and where she is most likely to go to next. This task is performed by what is known as location prediction algorithms, which need to sharpen their prediction accuracy, aiming at getting as close as possible to the maximum accuracy they could achieve, depending on the specific mobility features of the user.

Mobile devices are the proxies that reveal the location of the individual. They offer several ways to obtain the location-related data associated to them, allowing to collect this data by themselves or letting third parties (base station transceivers, BTSs, access points, APs, or external Internet servers) to perform the collection task. However, the location tracking (obtaining the individual's location at certain intervals) needs to be done by the terminals, as they are the ones carried by the user. These devices keep on increasing their computing capabilities, as well as adding more sensors and communication interfaces able to, among other duties, locate the individual. Unfortunately, one of the main components in a mobile device, the battery, cannot follow this development rate, and power consumption is still one of the main concerns. Besides this constraint, in order to be useful for different services and applications, location prediction must work in an online fashion such that the user can use the associated services right on time (e.g., not receiving a warning about a traffic jam in the near future route when the user is already in it, but sometime in advance to explore and choose other alternative routes). This requirement leads to continuous mobility tracking and data mining, where the *continuous* component is critical. Thus, not all tracking and prediction methods are suitable, but their selection is constrained to their resource consumption. It must be taken into account that user mobility produces a considerable amount of data per day, that pile up as time goes by. Either the location data collection and prediction are made by the terminal or by an external entity, the selected approach faces the problem of processing continuous flows of data, searching for patterns in the continuously growing data set, with the requirement of having immediate useful results such that, as soon as the user moves, she can take advantage of the next location prediction. Thus, tracking and prediction methods that require low battery consumption and low computational complexity are key so that, whichever entity is the one in charge of these tasks, it can continuously and appropriately perform them.

The challenges associated to dealing with present and future user locations goes further than this. As mentioned before, either the mobile phone or an external entity can collect the user's visited locations and predict her future whereabouts. Of course, considering third parties to deal with location data automatically triggers an alert on the privacy preservation of the user. However, even if the mobile phone is chosen to collect the data and estimate the predictions, which means that all the sensitive location information is kept in the mobile device, it is not enough to preserve the user's privacy. If these predictions are used to feed some service (e.g., traffic alerts) in order to get the personalized response, there is an unavoidable leakage of location data, since the predicted location needs to be sent to an external service that will have the associated information that wants to be retrieved. Actually, there is no need to address the prediction case. Nowadays, a mobile device gets the user location, but it immediately sends it to a third party in order to obtain the corresponding LBS: personalized news, weather information, directions in a map, etc. Whenever location wants to be used as context data for location or tracking-related services,

it leads to their systematic disclosure. As argued before, a wide range of information can be extracted from user mobility records, beyond the visited locations by themselves: home or job locations, or other venues such as hospitals, which disclose different potential sensitive data about the user. Therefore, a need for having some control over the disclosed data emerges, in order to be aware of the amount of sensitive data revealed and, hopefully, to decrease the potential negative effects this unintentional data disclosure leads to.

1.2 Objectives

As discussed so far, studying and leveraging individuals mobility may serve to provide useful and diverse innovative services that go beyond the limited LBSs users enjoy right now. However, many challenges and constraints must be faced before, which are the driving force of the present dissertation. *This thesis is focused on the study of individuals mobility by means of the location data provided by mobile devices, aiming at extracting conclusions that can be applied to the specific application of future mobility prediction in order to improve its performance.* From this general target and the previous motivation, multiple fronts to tackle emerge. In order to logically organize them, Figure 1.1 can help to understand the stages to address. Each stage is represented by a block, whilst the different challenges associated to each objective are enumerated next to each block. The combination of the stages and their challenges represent the goals to achieve throughout the thesis.

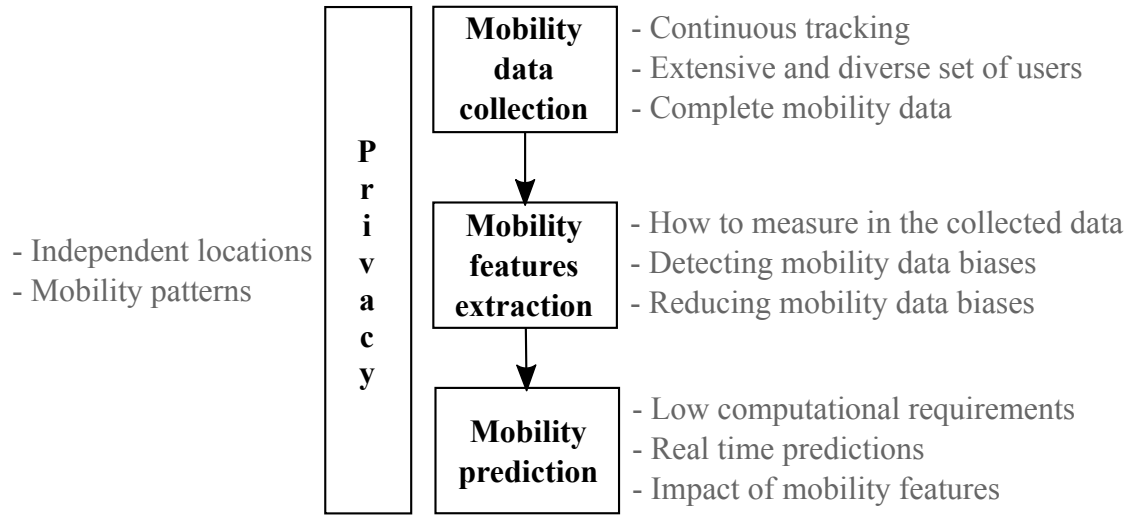


Figure 1.1: Main stages representing the objectives of the thesis, together with their main challenges.

- **Mobility data collection.** In this first objective, the aim is to inspect the available mobility data sources and data sets, and determine their suitability for continuous tracking, and their potential to accurately reflect the real mobility of the user's mobility. The target data sets would be as extensive as possible to be able to obtain

statistically meaningful conclusions, with the most up to date and appropriately sampled data, leading to complete mobility information, and which mobility traces come from people with different mobility habits to avoid potential biases. These properties aim at guaranteeing a good quality set of mobility data that allows to assess the mobility features and the prediction algorithms considered further in.

- **Mobility feature extraction.** With the data sets obtained in the previous stage, the second goal is directed towards the extraction of the mobility information enclosed in the data. To that purpose, the existing indicators about individual mobility will be explored, focusing first on the characteristics of the data used to analyze them, and second, on the potential impact they might have in the prediction process. These two aspects will determine the set of mobility features that will be thoroughly analyzed, using the data sets previously obtained. The main goal is to determine if the conclusions on the mobility features of the users can be extracted independently of the mobility data used, as extensively done in the literature so far. In order to do so, the possible existing biases associated to the data collection schemes will be exposed, together with several proposals to reduce them.

- **Mobility prediction.** The first target at this stage is to acquire a deep understanding of the working principles of a specific family of next location predictors. As discussed in the motivation, the algorithms considered in this dissertation will be characterized by low computational requirements, and the capability of generating accurate next location predictions right in time, as the individual moves. The second and main goal at this stage is the improvement of the fraction of correct predictions the algorithms can currently achieve. In order to do so, several modifications of their working principles are proposed, based on the initial analysis of their performance, and on the relationship between some of the mobility features analyzed in the previous stage and the predictions results.

- **Location data privacy.** Since disclosing location data risks user privacy, and seems to be unavoidable in order to enjoy nearly any LBS that relies on information provided by an external entity, the last objective of the thesis aims at measuring how much of this privacy is compromised by using the information theory concept of entropy. In order to cope with the shift in the working principles of the LBSs devised in the motivation, two approaches will be considered: the privacy loss derived from disclosing independent location samples, representing the threats associated to the current LBSs; and the privacy loss coming form revealing location samples sharing temporal correlations among them, which represents the threats related to the mobility patterns of a user.

1.3 Outline

The previous objectives are addressed along this dissertation following the structure described next.

Chapter 2 provides an overview of the state of the art concerning the topics covered in this thesis. It will also include some background on one of the main building blocks in which the thesis is based on, the estimation of the entropy of finite sequences, key concept for further chapters.

Chapter 3 describes, first, the mobility data source chosen, to focus later on the data sets that will be used. First, the available mobility data set to be used will be described, which will be followed by the reasons that led to run a new data collection campaign carried out for the research, as well as final data captured during the experiment.

Chapter 4 deals with the following step in the process, which is focused on extracting the interesting features out of the data sets described in the previous chapter. A review of the main metrics to be used will be done, from which follows an analysis on how mobility data collected in different ways lead to very different conclusions about mobility features, and a proposal on how to pre-process the data in order to avoid systematic biases in the conclusions.

The next step, location prediction, is covered in Chapter 5. The proposals on improvements and new algorithms will be presented, together with the results obtained after processing the data described in the initial chapters.

The privacy aspects accompanying the use of location data will be presented in Chapter 6. More specifically, this chapter will propose a privacy metric based on different entropy-related concepts, which will allow to analyze the impact that perturbing location data has on the privacy and data utility preservation.

Lastly, Chapter 7 collects the main conclusions and contributions derived from the dissertation, along with some of the many interesting research challenges that remain open for further investigation.

Two additional appendices are also included. Appendix A explains the details of the application developed for the data collection campaign described in Chapter 3, and Appendix B includes the demonstrations of privacy preserving mechanisms described in Chapter 6.

Chapter 2

Related Works and Background

Contents

Before diving deeper into each of the objectives described in Chapter 1, a review of the works related to each of them will be exposed in this chapter. The literature review exposed in the present chapter covers the three first objectives of the thesis. As can be seen in Figure 1.1, privacy is not another step in the flow, but appears as a parallel topic throughout the whole process, in a parallel plane. For the sake of clarity, the state of the art associated to location privacy will be covered in Chapter 6, so that the aspects concerning that objective, which are not directly but tangentially related to the process, do not confuse the reader along the mobility analysis process.

Besides the literature review, the chapter includes a section on a topic that will show to be key along the thesis: the entropy of a finite symbol sequence. This information theory concept will be tightly related to the mobility features extraction, the proposal of mobility prediction improvements, and the privacy metrics considered. For this reason, it seems convenient to understand first what entropy is, as well as the available estimators for finite symbol sequences like the ones representing the mobility traces of individuals.

2.1 Related Works

This section provides a review of the state of the art on each of the different stages described in Figure 1.1, which will be the building blocks of the thesis.

2.1.1 Mobility Data Sources in Mobile Devices

The very first step in understanding human mobility consists in having data representing such mobility in order to extract characteristics from them. Therefore, it is important that the data reflects as faithfully as possible the information related to mobility. It can be inspected from many different perspectives, again, depending on the application.

- The are different **mobility levels**: general (all the movements of the user), medium-small scale (mobility inside a campus), or indoor (inside a building, floor, room, etc.), among others.
- **Accuracy** might be key (for indicating a friend where we are, or to locate ourselves in a foreign city), or can be a relaxed requirement (when looking for restaurants nearby).
- **Locating** the user might be enough (for applications with just the current location attached to each request), or the user might need to be tracked. In this last case, the **tracking** frequency is also another parameter to decide.

Next, an overview of the technologies available in mobile devices to track their owners location is presented. The three features mentioned before will be specially considered. In particular, Global Positioning System (GPS), Wi-Fi networks, and cellular telephony networks are compared, and also a new tracking mechanism that has emerged with the advent of location-based social networks (LBSN).

- **Global Positioning System (GPS).** The great majority of new mobile devices integrate a GPS system. When a user wants to obtain her location using this technology, first she enables the GPS of the terminal, then the device searches for the satellites and synchronizes with them and finally, once the synchronization is set, the user can perform location requests.
  - *Location data accuracy.* This is the main strength of this technology, since the location data accuracy is close to 10 meters [36] [159].
  - *Coverage.* GPS has global coverage, but does not reach indoor environments, thus not being possible to track mobility inside buildings and similar scenarios.
  - *Power consumption.* This is one of the main weaknesses of GPS, since location requests greatly drain the battery of the device [28]. Although there are works that try to minimize the power consumption by proposing different location acquisition schemes [67, 68, 69], the decrease in the battery consumption comes at the cost of reducing the accuracy of the location data obtained.

- **Wi-Fi-based location.** By monitoring the Wi-Fi access point (AP) the user's device is attached to as she moves, her mobility patterns can be indirectly tracked. The mapping between the AP medium access control (MAC) address and its location is needed in order to know the zone where the user is at all times. This indirect mobility tracking method has the following characteristics:

  - *Location data accuracy.* It is related to the AP coverage area, which is usually close to 100 meters radius. Therefore, a device attached to certain AP is located somewhere within a 100 meters radius circle [55], which made the accuracy worst than GPS case, but bounded.
  - *Coverage.* Wi-Fi networks provide indoor coverage, but only on local areas. Therefore, the users' mobility can only be tracked on bounded areas such as buildings, campus, or similar scenarios.
  - *Power consumption.* Since location tracking using Wi-Fi only needs to know the AP MAC address (no need for data transferring), the power consumption of having the Wi-Fi antenna working, scanning the radio environment looking for new Wi-Fi networks, and being attached to some AP is low (the main power consumption of Wi-Fi connections is due to data transferring [45]). However, depending on the method for translating the MAC address of the AP to the corresponding coordinates, there may exist an extra power consumption if an Internet connection is needed.
  - *Additional comments.* The main weakness of this system is the translation from MAC address to coordinates, since Wi-Fi APs may easily change its location. Therefore, the mapping should be updated frequently or it may lead to erroneous locations. On the other hand, the location accuracy may be improved by means of triangulation methods [159, 7, 73].

- **Cellular telephony network-based location.** The working principles of this system are very similar to those of Wi-Fi case. The user mobility is tracked by knowing the network base transceiver station (BTS), also referred to as cell, the device is attached to as the user moves. In this case, a translation from BTS (cell) information to coordinates is also needed. The main features of this location system are the ones below:

  - *Location data accuracy.* This is the worst feature of this system since the accuracy depends on the user's location. A cell from Global System for Mobile Communications (GSM) network ranges from 200 meters radius in urban areas to up to several kilometers in rural scenarios [159], thus the accuracy being much worse than GPS or Wi-Fi systems and unbounded.
  - *Coverage.* This is the main advantage of this technology since it provides global coverage, even in indoor environments.
  - *Power consumption.* The power consumption due to cellular telephony network connection is the lowest one, since it is the basic feature of a mobile phone (both

Wi-Fi and GPS consumptions are added to this basic one). But as in the Wi-Fi case, there is an additional consumption due to the translation step, that will depend on how this translation is done.

- *Additional comments.* There is an advantage of this approach over the Wi-Fi case due to the fact that cellular telephony networks are much more stable in terms of BTS locations with respect to AP locations.

- **Location-Based Social Networks (LBSN).** This new type of social network is based on each of its users indicating (check-in) the place (restaurant, airport, sport center...) where she is at every moment, like in Foursquare[1]. Therefore, the location history can be directly obtained by taking the sequence of check-ins made by the user. The main features of this approach are the following ones:

  - *Location data accuracy.* The accuracy depends on the honesty of the user: if she checks-in where she really is, then she will be located inside the place she says to be, and depending on the size of the place, the accuracy will be higher (if the place is small, like a restaurant), or lower (if the place is big, like a mall). However, if the user lies about the location, the accuracy is totally unbounded.
  - *Coverage.* This is an advantage of this system. It provides global coverage, both in outdoor and indoor environments, since the social network provides the whole map for the user to check-in where she is.
  - *Power consumption.* The power consumption needed for each check-in is associated with the data connection required to connect to the Internet. It will depend on the technology used (Wi-Fi or cellular telephony network), and the duration of the connection.
  - *Additional comments.* Although this emerging type of social network provides global coverage, the location tracking completely depends on the user, since she needs to actively check-in in every place she goes. Unfortunately, supposing that every user will check-in every single place she visits is unrealistic. For the purposes of this thesis, the tracking process needs to be independent of the user's will to collect precise mobility information about herself to precisely capture her mobility patterns at all times of the day and week.

Once the data sources are exposed, the next step is to explore the mobility data sets coming from the different sources exposed above and used in the literature.

2.1.2 Mobility Data Sets Used in the Literature

Human mobility has received much attention during the last decade. The growth of mobility-related studies stems from overcoming the barriers holding the mobility data collection process back. The first data sets were originally collected by surveys. One of the first studies on human mobility not relying on surveys was performed by analyzing the

[1] Accessible at https://foursquare.com, as of October, 2015.

circulation of bank notes in the United States [16], but was demonstrated, years later, that the conclusions could hide a correlation of population-based heterogeneity and individual human trajectories due to the nature of the data collected. Thus, it was obvious that the mobility data that can be obtained was not enough to have statistical significant results, or biased depending on the collection technique used.

This situation was completely turned upside-down with the tremendous growth of the use of mobile phones. Nowadays almost every person carries a mobile device with her, all day long, everywhere she goes. Thus, the mobile device is the perfect proxy to track user's mobility. This fact was rapidly acknowledge by the research community, who took advantage of the increasing number of sensors as well as the available application programming interfaces (APIs) of the different operating systems, which incredibly ease the use of the systems and sensors in the devices, to develop applications capable of tracking the user's movements.

Among the several systems integrated in the mobile devices that can be leveraged as location proxies, the most popular ones are GPS, as it is the only one providing real location coordinates, and also Wi-Fi and the cellular telephony network. There are other systems, like Bluetooth and Radio-frequency IDentification (RFID), which are also used in some specific scenarios to locate people in small size environments. However, since the thesis is focused on the global mobility of users, these systems fall short to capture such global behavior, and thus will not be considered in this literature review, nor in the rest of the dissertation. As explained in next sections, this thesis will be focused on using GSM-based location data due to its global coverage and low battery impact during the collection process. Thus, the next review will be focused on the data sets based on the data coming from this technology. Nonetheless, a great number of studies [113, 87, 84, 85, 48, 160, 161, 79, 95, 70, 83, 121] based on GPS or Wi-Fi can be found in the literature, yielding also interesting conclusions in human mobility-related research.

Focusing on GSM-based data, there is a large number of mobility-related studies handling this type of data sets. However, as described in next sections, there are different ways to capture cellular telephony-based location data. The most used approach is to use the data stored in the network nodes themselves, which record the BTS to which the device is connected when the user is making or receiving a call, or sending or receiving an short message. These records are known as call-detail records (CDR). These data sets are characterized by their high number of users and long duration, since the data is recorded anyways for billing purposes, and thus is widely available (although not easily accessible, since it depends on the network operators permission). Some of the works using these data sets are used to study the trajectory of users in spatial and temporal terms [50], studying the predictability in human mobility [134, 88], running large-scale studies to characterize the behavior of mobility in cities [138, 10, 42, 60, 58, 59, 95, 61, 9, 37], modeling scaling properties of human mobility [133] or improving public transport [11], and even examples from individuals who published their own recorded CDRs [46, 137]. And there are also some works [106] questioning the validity of CDRs to capture human mobility features.

In order to answer such question, other data collections are possible using cellular data. The second methodology to collect this location-related data is recording the cell to which the device is attached every moment. This can be done from the mobile device itself,

Data Set	Technology	GSM sampling	Users	Duration	Date	Available
Nokia Data Challenge	GSM, Wi-Fi, GPS	CDR	200	24 months	2011	By request
MIT	GSM, Wi-Fi	Cell change	95	9 months	2005	Yes
Michael Ficek	GSM	CDR	1	142 days	2012	Yes
Malte Spitz	GSM	CDR	1	6 months	2011	Yes
PlaceLab	GSM, Wi-Fi, GPS	Wardriving	-	2 hours	2004	Yes
Rice Context	GSM, Wi-Fi	Cell change	14	3 weeks	2003	Yes

Table 2.1: Summary of the public available mobility data sets in the literature.

and thus, the data available is much less extensive. This collections method implies the users installing a dedicated application that collects the data continuously and sends it to certain server from time to time. Thus, these data sets are not already available, like in the previous case, but they are much harder to obtain. Among the related works using this type of data set, some are focused on prediction of the next location [41], some others focus on studying predictability of users [62] or context-related studies [105]. As can be checked, the number of this type of data sets is much lower than the CDR-based ones.

Besides these data sets, there are some others which gather location-related information coming from different sources, and are probably the most popular data sets. The MIT Reality Mining data set [43] is one of them, which will be study more in depth in next sections, but there are also others like the data coming from the Nokia Data Challenge [72, 78], the PlaceLab data set [77] and the NetSense data set [140].

Thanks to the generosity of the contributors to the data sets and to communities like CRAWDAD[2], many of these data sets are publicly available. Table 2.1 details the data sets publicly available and their main characteristics.

2.1.3 Study of Human Mobility Features

The resurgence of the interest on human mobility has generated a myriad of works analyzing an extensive set of features, extracted from distinct mobility data sources and data sets, exposed in the previous section. Focusing now on the features extracted from these data sets, this section presents a review of some, for the sake of brevity, of the most significant works in the framework of the objectives of the thesis.

The studies surveyed can be roughly classified, mainly, into two groups, depending on

[2]Accessible at http://crawdad.org/, as of October, 2015.

the perspective form which mobility is analyzed. On the one hand, many works hold a perspective centered on the environment where mobility is studied, generally at city-level. Its main goal is to uncover how mobility shapes the environment or vice versa, but with the central aspect being mobility in the environment. The opposite perspective is user-centric, that is to say, the goal in this case is to characterize the individual or group behavior, but having the person as the center piece instead of the environment.

2.1.3.1 Mobility Features from a Location-Centric Perspective

Considering the case of location-centered mobility, sometimes known as urban dynamics, many works have appeared in the last years in line with the advent of smart cities and the corresponding need to understand the inhabitants flows defining the scenario [139] to help in planning and provision of municipal facilities and services, provide better public transportation [11] and road usage [150]. This type of studies were not possible until traditional approaches, like surveys, were replaced by the data provided by cellular network operators, which disclose more than just snapshots of people movements, but where the spatial extent and temporal correlations are wider than the ones provided by previous studies.

One of the first works using cellular telephony data to characterize urban-related features was carried on by the MIT in collaboration with Telecom Italia [109]. The study divided Rome into pixels and chose six different locations of one pixel each. The Erlang daily traffic distributions were studied, to group them by degree of similarity and map them to hot and cold areas of activity along the day, week and month. This work was taken as reference by Sun et al. [141], who also divided a southern city in China into pixels, and by using the CDRs collected in such city, analyzed the population distribution based on the cellphone usage data from the CDRs in each pixel. Their study, using principal component analysis, revealed the low dimensionality needed to characterize the structure as well as a temporal stability and dominance of periodic trends. Both at pixel and group level, three categories were detected, namely regular or predictable patterns, unusual patterns and insignificant patterns, being the predictable patterns the dominant ones.

More recent works, like [10], use CDRs to capture the city dynamics by determining the residential areas where work people live and the residential areas of late-night people, thus demonstrating that clustering people based on cellphone usage is possible, even without taking into account temporal correlations. Follow up works [60, 58, 59] extended the former study by finding mobility patterns in New York and Los Angeles regions, such as identifying important locations, who travels further, who travels more distances, when people move more and at what season, among others, using a metric called daily range, which corresponds to the maximum distance traveled in a single day. This kind of metrics are only possible when the location of the BTSs recorded in the CDRs are known. Yet another follow up work [61] using the same data set models the movement of large populations within different metropolitan areas, aiming at producing synthetic CDRs taking into account the probabilities of being at home or the work place, and the probabilities of a call being made or received at certain location (cell) or at certain time. Another related work [95], this time using GPS and Foursquare data, uncovers variations in hu-

man mobility in different cities due to different distribution of places. Yet another work comparing different cities [53] perform a clustering analysis and compare human activities in urban environments based on the detection of mobile phone usage patters (number of calls, messages, data traffic events). The analysis leads to identifying locations with similar patterns within a city, where their core financial centers all share similar activity patterns and commercial or residential areas present more city-specific patterns.

In [65] the authors use CDR data from eight cities in Chine to investigate how human mobility patterns inside cities are impacted by the compactness and size of cities, using displacement and radius of gyration. As in previous studies, they found out that the distribution of displacements, in this case in intra-urban scenario, follows exponential laws, whilst the exponents vary from city to city, thus confirming that the city sizes and shapes impact mobility (which follows the intuition that individuals living in large cities have to travel further). The dependency of the parameters of the model on the specific city is again highlighted in [138]. This work uses CDRs of several million users during one month to characterize the footprint of the users in the city, disregarding temporal or spatial features. In order to do so, they use clustering techniques and a geometrical construct called minimum area bounding rectangle (MABR) that delimits the daily movement range of a user. The results indicate that, whilst the area, average trajectory length, and area of clusters follow the same distributions, the distributions of the MABR skew or angle, or the number of clusters depend on the locale.

One of the most common studies regarding cities is to uncover which are the different regions of the city, like the one previously mentioned [58] or in [157]. In this last work, the authors try to complement the knowledge of points of interest (POIs) of a city with the information provided by CDRs to differentiate the intensity of each function in each region or location (e.g., a small restaurant has a different impact than a big attracting one, even when the two of them are considered POIs). Another study featuring different urban areas [158] uses hourly time series representing the dynamic mobility patterns in different urban areas and use dynamic time warping algorithm to measure the similarity between time series to classify different urban areas, which allows to investigate outliers urban areas through abnormal mobility patterns, differentiate weekends from weekdays and to locate commercial zones. Analyzing this same aspect, [37] uses mobile telephony data to visualize the regional flows of people across the Republic of Ireland, using Markov chains to rank significant regions of interest to mobile subscribers.

As can be expected, mobile phones can help in demographic research, as demonstrated in [99], where the authors describe the pilot study "Human Mobility Project (HMP)" that explores the use of mobile phones in demographic research and tests dynamic, location-based surveys. They use GPS and GSM data to determine where people live, where they spend time when not at home and what are their trajectories.

2.1.3.2 Mobility Features from a User-Centric Perspective

The second standpoint of mobility is user-centric, referring to the works aiming at characterize the intrinsic features of human mobility, disregarding the specific scenario where they move.

In the survey elaborated by Lin et al. [84], they review relevant results in some of the main areas studied in human mobility studies: inferring important locations, detecting modes of transport, mining trajectory patterns, and recognizing location-based activities. They also classify mobility analysis into two main areas: mining mobility patterns and constructing mobility models. These two big blocks were also pointed out in [66], inside a bigger framework that also includes data collection and the final applications where the mobility results and models are applied as additional areas conforming the big picture of human mobility study.

Regarding the identification of salient locations, Eagle et al. [41] analyzed CDRs of 215 individuals recorded during 5 months to cluster the most used BTSs (i.e., locations) and validated the results using data coming from Bluetooth beacons placed in the individuals homes, aiming at predicting subject's next movements. In [161], the authors use GPS data from 107 users taken during a year to mine interesting locations, as well as classical travel sequence, for travel recommendation. These salient locations are important, due to the asymptotic characteristic human mobility behavior, studied by Song et al. [133], of the individuals returning systematically to some preferred places, and just sometimes exploring new areas. Boldrini et al. [14] identify also the preference to spend time in a limited number of popular locations, and together with the preference to select short distances over longer ones, and the sociability of users, meaning the larger the social network the higher the mobility, they propose a mobility model based on these three properties for reproducing accurately the behaviors of users in mobile ad-hoc networks (MANETs), opportunistic and delay tolerant networks.

On the motion mode detection, Zheng et al. [160] used GPS data from 65 people recorded during 10 months to detect transportation modes, for which they focus on heading change rate, stop rate or velocity change rate, all of which show to be more robust than velocity and acceleration.

Another of the main features analyzed is the displacements of the users. In the first works studying human mobility using real data, bank notes in this case [16], the authors already found out that the distribution of the traveling distances decays as a power law, and that the probability of remaining in small, spatially confined regions for certain period of time is dominated by long tails, properties that will be found once and again in all the future works. In [50, 144] the authors use the CDRs of 100,000 users collected during 6 months to study the basic laws driving human motion. They found out that human trajectories show a high degree of temporal and spatial regularity, since each individual is characterized by time-independent travel distance and a significant probability to return to a few highly frequented locations. The distribution of the distance covered in the displacements suggests that human motion follows a Levy Walk, and the calculation of the radius of gyration (i.e., the distance traveled by the user when observed up to time t) follows a truncated Levy Flight distribution also. These results were backed up by the work of Rhee et al. [113], where GPS data were used instead of CDRs, but leading to the same heavy-tailed distribution of the distances covered by the individuals in their displacements. These behaviors, widely detected in several different data sets and populations, is in line with the exploration and preferential return model proposed in [133]. Some other works use different data sets, like in [24], where the authors use information from a location-based

social network and cell phone data to conclude the same features seen before with other data sets: human mobility is a combination of periodic movement geographically limited and eventual random jumps, correlated with social network. They go a step further by determining that there is some sort of short-ranged periodic travel, that explains 50-70% of the behavior, which is spatially and temporally not affected by the social network structure, whilst there is also a long-distance travel more influenced by social network ties, which explains 10-30% of the movement. In [162] the authors complement the observation of power-law distributions observed in the displacements with the observation of this same distributions in the inter-arrival and dwell times, using cellular data collected during one month. A novel finding is the two types of individuals found in [100], where the authors split the population into the so called returners (people who only visits a very limited set of locations) and explorers (people who travels to many more different and distant locations than the most usual ones), and explain the role of both types of individuals in the spread of diseases and social networking.

Song et al. went a step further by proposing a new metric for mobility [134]. In their study, they used CDR data of 50,000 individuals recorded during 3 months, and study their entropy and entropy rates (by using Shannon entropy and Grassberger entropy rate estimators, described in Section 2.2), to finally propose a new metric, the predictability, which sets the upper bound on the best accuracy a location prediction algorithm could ever achieve, depending on the specific user entropy rate and different number of locations visited. They show that predictability is largely independent from the radius of gyration, and that the average predictability is centered in the 93% of correct predictions. After its proposal, predictability has been widely studied. In [62], the authors extend Song's work using a more detailed data set including data from multiple sensors (GSM, GPS, Bluetooth, and Wi-Fi) and with higher temporal resolution, comparing how different time scales affect predictability. Lin et al. [85] used high resolution GPS data to measure the predictability of the users' trajectories using different scales, checking that the high predictability is still present at very high resolutions, being independent of the overall mobility area covered, and with an invariance respect spatial resolution, meaning that the predictability decreases if the spacial precision increases. In [86] the authors further extend their analysis by proposing a new Markovian mobility model addressing what they consider the two driving forces modelling human mobility: travels governed by occasional exploration of new places and preferential return to some highly frequented locations (as exposed in [50, 134, 133]), and the high predictability showed by human mobility sequences, which were not addressed before for mobility modeling. This new model yields predictability values much lower than 93%, which can be in line with the results that will be shown further in this chapter. In [88], the authors calculate this metric in a data set made of the CDRs of 500,000 individuals during 5 months, specifically targeting the predictability of population displacement after the 2010 Haiti earthquake, in order to improve the response to disasters and outbreaks under extreme events.

Lin et at. [87] test two widely accepted assumptions: the stationarity and dependency of visited locations and the preference for revisiting locations according to the visitation frequency. The first assumption was studied by comparing the correlations in the sequences, which uncovers daily and weekly patterns. The second assumption was tested by com-

paring the entropy rate of the movement histories with respect to sequences drawn from independent identically distributed and Markov processes. Using a combination of GPS and GSM data, they claim Markov chains of order 1 to be the most similar ones to the actual movement histories in terms of entropy results.

Among the future research lines on human mobility, some studies like [84] mention further study about predictability, since it depends on how the data is collected (e.g., a fixed sampling rate, as the one used in many of the previous studies, may condition the results since many samples are going to be taken at the same location, thus increasing the predictability, but not accounting for the real movement of the user); model the process of mobility and construct more accurate mobility models; and determine which data compression algorithm is more suitable for modeling individuals' mobility sequences and how to tailor the algorithms based on the special characteristics of individuals' mobility behaviors. This last aspect is precisely the main focus of this thesis.

Focusing on the possible biases in the features studied in the previous works, in [106, 125] the authors raise their concern on the potential bias introduced due to the use of CDR-based data to study human mobility. In their reasoning, they claim that the users usually chosen to conduct the studies are those with high voice-call frequency, which may not be representative of the real situation. They conclude that this kind of data sets can infer home and work locations, but provide poor results for overall spatio-temporal properties, such as the set of significant locations (those concentrating 90% of the visits) and the entropy and radius of gyration.

Another shortcoming, treated separately for being a very specific problem of certain data such as GSM or Wi-Fi-based location data sets, is the ping pong effect, which is studied in [79]. The authors detect this effect in Wi-Fi networks, and consider two ping pong scenarios, between two and three coverage areas. When the ping pong is detected, the coverage areas implicated are clustered into a new one, and then compare the number of transitions after and before. In [153] another offline method to filter ping pong is proposed, in which more information is collected in order to check if the sequence labeled as ping pong sequence is in reality a ping pong sequence, and then removes it. In next sections, these works will be extended by considering different detection and filtering approaches, that could be applied online, as well as a more extensive study on the mobility features observed in the traces before and after filtering.

2.1.4 Mobility Prediction Algorithms

Location prediction is a topic largely studied. Plenty of different methods, using varied types data and aiming at predicting distinct aspects of the individuals future whereabouts, have been proposed in the literature. This section contains an overview of the most significant works. The present analysis covers the research carried out with diverse types of data, since the methodologies and conclusions drawn form them can be sometimes applied to location prediction in general, regardless of the nature of the input location data.

The main focus of the review is set in two perspectives: the differences in prediction when different data types are used, and the diverse approaches followed to calculate predictions, including those based or helped by leveraging human mobility features.

2.1.4.1 Prediction using Different Location Data Types

As the review on the previous sections shows, different data sources can be used to describe individuals' movements. These different sources can be, then, used as input data of different location prediction algorithms. The main classification regarding the input data splits these prediction algorithms into two groups: the ones using the location physical coordinates, and the ones using sequences of symbols representing the visited locations.

All the works using GPS data fall in the first group. In [124], Scellato et al. proposed NextPlace, a location predictor that first used the residence time at each input GPS location to identify salient locations. After that, the algorithm assumes that the period of 24 hours strongly determines human behavior, thus basing the prediction of the next movement on what happened in the previous periods of 24 hours. Ashbrook et al. [6] also use the time spent at different GPS locations to group them into significant places, but in this case Markov models of order 1 and 2 were used to perform the prediction. Monreale et al. [93] followed the same strategy of mining trajectories based on time information, to use then a prefix tree of trajectory patterns.

In [71, 25], the authors also use GPS data, but in this case the location recognition is carried out through the G-means clustering technique, to further apply hidden Markov models (HMMs) for path classification. Alvarez-Garcia et al. [3] followed a similar approach, extracting GPS trajectories by clustering destination points and then applying a HMM to detect the invisible process (destination) through visible observations (the sequence of significant points detected). Again, Ying et al. [155] performed a clustering process for extracting significant GPS locations and movement together with a probabilistic model for predicting the time at which the next movement will happen.

In [94], the authors uses the coordinates of the different locations, but the data set is synthetically generated, although the process is the same than in real data studies: mining the location logs to discover frequent trajectories which are further transformed into movement rules by using prefix trees. The authors of [19] also used location coordinates, although in this case the data comes from the AirSage service, which provides physical coordinates from CDR and Wi-Fi data. Besides, they divided the space into a grid and predicted the user location at time k using the information at times $k - T$, $k - 2T$, etc.

In general, GPS data face the problem of location recognition as the first step in the prediction process. This problem comes from the fact that the GPS receiver provides many diverse coordinates, slightly different among them every time a location is visited, when the individual is in one single location. The most used solutions are grids and spatial or temporal clustering approaches, most of them performed in an offline phase.

Another type of location data is that coming from location-based social networks (LBSNs). Despite their drawbacks (as discussed in Section 2.1.1, these data depends on the willingness of the user to indicate her location, which is not equal for all the locations she visits), this kind of studies are becoming hot topics nowadays [96, 82, 151]. Their strongest point is that with this data there is no need to identify the location form the coordinates, since the user already points out the exact locations she is at.

The last location data usually considered for location prediction comes from Wi-Fi and cellular telephony networks. The most current works are already using data from

the Long Term Evolution (LTE) network, collecting events not from the user's phone yet, but from the LTE network infrastructure itself [89, 54], knowing the exact location of the nodes collecting the data. On the opposite side, there are some works using synthetic data, like [154, 4], which do not reflect the ping pong effect that shown to heavily impact the resulting location trace.

In other cases, like in [135], where the authors aimed at predicting the time of the next move of a user, they use Wi-Fi data and leverages also the ping pong sequences since their goal is to predict the next connection.

In [76], the author performs first a location recognition procedure, based on clustering. However, in this case the clustering is made leveraging the ping pong effect usually detected in cellular networks. Once the locations are detected, they are divided into regular locations and bases (i.e., locations where the time spent by the individual is above certain threshold), aiming at predicting which the next base will be. Both the location recognition and base learning processes are carried out offline. The same happens in [1], where the authors filter the ping pong effect by clustering cells and combining all clusters that have shared cells, in an offline fashion.

2.1.4.2 Prediction using Different Approaches

When considering the approaches followed to calculate the predictors, the variety of methods proposed in the literature is even wider than when considering the input data type. The most classical approaches have relied on different machine learning algorithms to use them as predictors. Other widely used approaches are based on Markov models and state-based techniques, pattern matching algorithms, or time-series analysis. On the other hand, the newest works on prediction shifts the attention from the classical approaches to the consideration of human mobility features observed in real users to apply them on the prediction improvement.

Location prediction based on machine learning methods is perhaps the most common approach, and includes clustering techniques, Bayesian models and neural networks, among other alternatives. The general working principle in these cases is that the algorithm trains a system, called classifier, to classify observations in order to predict unknown situations based on a history of patterns. In [63], the authors propose to use predictors based on clustering of the mobility data collected by all the users, and detecting the mobility information of similar users to predict the next location of one of them by using bayesian models. The authors of [4] use also clustering techniques, k-nearest neighbors (K-NN) in their case, together with decision trees to build a trajectory classifier and, further, a location predictor based on short-term historical data. In [25, 71] the authors use a different clustering technique, G-means, applied also to the trajectory clustering, combined this time with a HMM for path classification. In [3], clustering is used once again to extract the destination points, whilst a HMM is applied to detect the destination (invisible process) through the sequence of significant points (visible observations). Clustering is combined with HMM also in [91], but the clusters are, in this case, constructed according to the temporal period in which the visits to the different locations are made. Ying et al. [155] made use of clustering, but combined with semantic data to mine significant locations,

which are stored in a tree that facilitates a similarity calculation of the current pattern with respect to the previous stored ones. In [76], the author uses also clustering to group similar paths and use additional temporal data to make the predictions.

Besides clustering, other machine learning techniques are used as well. In [1] the authors used a hybrid technique, combining bayesian inference in artificial neural networks.

Markov models are also a popular option for many prediction algorithms. Some of these works use the classical Markov models of low order (generally, 1 or 2), like in [6] or even in the most recent works like in [89, 54]. In [21], the authors use Markov models under the observation that movement changes over time and that movements are affected by individual and collective properties, thus needing to consider three different levels—global, personal, and regional—that are adequately modeled by Markov models. Some other algorithms are based on HMMs, where the invisible states (generally, the final destination of the trajectories), are calculated through the visible observations (i.e., the locations that build the trajectory). Some works like [25, 71, 3, 91] follow this approach, combined with clustering methods, as described before. A slightly different approach is the use of state-based predictors, like in [13], where the authors estimate a user's transition probabilities between discrete locations using transition frequencies counts estimated from other similar users. A evolved family of algorithms based on the working principles of Markov models are LZ-based algorithms [163, 12, 51]. In this case, instead of using a model of fixed order, the algorithms dynamically compute the optimal order to minimize the uncertainty about the next location of the user, depending on the input mobility history.

Pattern matching techniques are similar to Markov or state-based models in general, since in most of the cases the sequences of states are concatenated to build a string or pattern representing the trajectory followed by the individual. These strings or patterns are usually stored in trees, sometimes referred to as prefix-trees. The work presented in [104] is an example of the use of Markov models in the form of trees to further apply pattern matching techniques such as prediction by partial matching (PPM) [26].

Lately, the research on movement prediction has shifted the foundation of the newest proposals of the algorithms to the newest mobility features extracted mainly form the CDR data sets, like the ones described in [133, 134, 50]. The most used feature is the similarity of a user's trajectories with respect to her closes acquaintances, or even to a global scale collection of trajectories [63, 121, 93, 19, 151]. Time periodicity is also a clearly characteristic of human movement, in which there exists a strong correlation between events separated one day, or one week. Works like [124] try to leverage this feature over salient locations identified by applying first a 2-D Gaussian distribution weighted by the residence time, whilst in [19], the authors divide the space into a grid and predict the person's location at time k by using information at $k-T$, $k-2T$, etc., being T one day, one week, etc. Finally, in order to tackle the problem of predicting the next location of a user who is going to visit a new place not seen before, the latest works have tried to use the exploration and preferential return model proposed in [133], like in [82], whose authors proposed an exploration predictive model aiming at predicting if the user's next location exists in the location history considering how much she is likely to explore, and considering collaborative social knowledge and geographical influence for seeking candidate locations to explore.

Several comparatives among different models can be found in [136, 8].

2.2 Background

The remaining of this chapter is devoted to a concept that will be key in subsequent chapters: the entropy of a symbol sequence. The next sections provide an overview of the theoretical definitions and the practical meaning of this concept, and will analyze different estimators applied to finite symbol sequences, and more specifically, to location sequences that will be the case treated along the thesis.

Before starting with the formal definitions, it is convenient to set the notation that will be used throughout the thesis. The convention of uppercase letters for random variables and lowercase letters for particular values they take on will be followed. For simplicity, all random variables take on values in a finite alphabet, since in the specific scenario of mobility there is a finite number of locations. Probability mass functions (PMFs) are denoted by p, sub-indexed by the corresponding name of the random variable when not understood from the context. For instance, the PMF of a random variable X at x will be denoted by $p_X(x)$, or simply by $p(x)$.

2.2.1 Entropy and Entropy Rate Basics

Before diving into the applications of calculating the entropy of specific mobility data sequences in the following chapters, this section provides an introduction to the concept of entropy and its practical interpretation. A wider review on this topic can be found in [49, 29]. Starting with the basics, next we introduce the definition of what is known as **Shannon entropy** in the information theory domain, introduced by Claude E. Shannon [128].

Definition 2.1. Let X be a discrete random variable taking values on an alphabet $\mathscr{X}$, being $|\mathscr{X}|$ the cardinality of the alphabet, and with PMF $\Pr(X = x) = p(x), \forall x \in \mathscr{X}$. Then, the entropy can explicitly be written as:

$$\mathrm{H}_S(X) = -\sum_{x \in \mathscr{X}} p(x) \log_2 p(x) \tag{2.1}$$

where the base 2 logarithm denotes that the resulting entropy value is measured in bits.

Paying attention to the practical meaning of entropy, $\mathrm{H}(X)$ measures the expected "surprise" or uncertainty enclosed by the random variable X. For instance, consider the case of tossing a coin with fair probabilities of coming up heads or tails. One cannot guess what it is going to come up next, since both events have the same probability. In this case, the uncertainty is maximum and thus, the entropy of the discrete random variable representing the outcome of tossing the coin is maximum too. On the other hand, if the coin is not fair and one event (e.g., coming up heads) has a higher probability than the opposite one, there exists less uncertainty of what is going to come up next. Thus, the entropy is lower. The extreme case is when the coin has heads in both sides and, therefore, it is always going to come up heads. In this case, there is no uncertainty at all about the outcome and thus, the entropy would be zero. Figure 2.1 illustrates this example. Let X be the random variable representing the outcome of tossing the coin, taking values on the alphabet $\mathscr{X} = \{a, b\}$ where a represents heads and b represents tails. For instance,

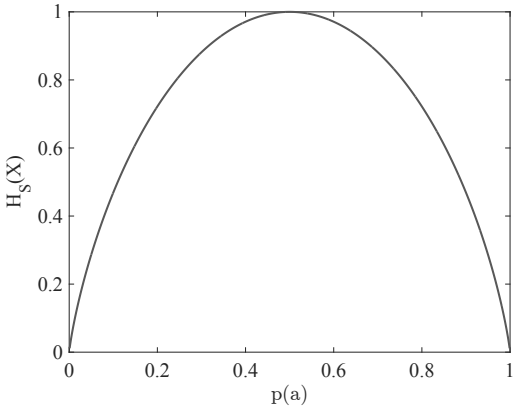


Figure 2.1: Shannon entropy for different probability mass functions of the Bernouilli distribution.

a fair coin corresponds to the case in which $p(a) = p(b) = 0.5$, whilst a coin with heads in both sides corresponds to the case $p(a) = 1, p(b) = 0$. In the figure, the x-axis of the figure shows the probability of the coin coming up heads, $p(a)$, whilst the y-axis shows the entropy associated to each of this probabilities. As can be observed, the entropy is maximum for $p(a) = 0.5$ (heads and tails have the same probability), and it decreases until zero when $p(a) = 1$ or $p(a) = 0$ (when it always or never comes heads, respectively), when there is no uncertainty about the outcome.

The Shannon entropy is a member of a wider family of entropy functions, known as the family of Rényi α-order entropy functions, which general expression is the following one:

$$\mathrm{I}^{\alpha}(X) = \frac{1}{1-\alpha} \log_2 \sum_{x \in \mathscr{X}} p^{\alpha}(x)$$

The Shannon entropy corresponds to the case in which $\alpha \to 1$. Another special case, $\alpha = 0$, will be also considered. This case is known as **Hartley or maximum entropy**, which will be a useful normalization parameter along the thesis, and can be expressed as follows:

$$\mathrm{H}_H(X) = log_2|\mathscr{X}| \tag{2.2}$$

This initial definition can be further extended to the case of the joint entropy of two discrete random variables, and also to the case of conditional entropy.

Definition 2.2. Let X and Y be two discrete random variables, taking values on alphabets $\mathscr{X}$ and $\mathscr{Y}$, respectively, and with a joint probability mass function $\Pr(X = x, Y = y) = p(x, y)$. Then, the joint entropy can be expressed as:

$$\mathrm{H}(X, Y) = -\sum_{x \in \mathscr{X}} \sum_{y \in \mathscr{Y}} p(x, y) \log_2 p(x, y) \tag{2.3}$$

Definition 2.3. Let X and Y be two discrete random variables, taking values on alphabets $\mathscr{X}$ and $\mathscr{Y}$, respectively, with a joint probability mass function $\Pr(X = x, Y = y) = p(x, y)$ and conditional probability mass function $\Pr(X = x|Y = y) = p(x|y)$. Then, the conditional entropy can be expressed as:

$$\mathrm{H}(X|Y) = \sum_{y\in\mathscr{Y}} p(y)\,\mathrm{H}(X|Y = y) = -\sum_{y\in\mathscr{Y}}\sum_{x\in\mathscr{X}} p(x, y)\log_2 p(x|y) \tag{2.4}$$

With the initial concept of entropy it can be hard to measure the real randomness of a sequence of events. To illustrate that fact, the sequence of events $abababababababababab\ldots$ will be considered. Applying expression (2.1), and estimating $P(X = a) = 0.5 = P(X = b)$, then $\mathrm{H}(X) = 1$, since both events have the same probability. However, can the sequence be considered as a completely random process, as the value of $\mathrm{H}(X)$ suggests? The answer is clearly no, since at each step the next event to happen is easily known. Besides the probability of each symbol, there exists information regarding the temporal correlations among one sample and the previous ones: the symbol at position i is the same one than in position $i - 2$. In order to account for these correlations that may significantly reduce the uncertainty of a random process, the concept of stochastic process is introduced. Let:

$$(X_n)_{n\in\mathbb{N}} = X_1, X_2, X_3, X_4, X_5, \ldots$$

be a stochastic random process with samples defined on a common alphabet $\mathscr{X}$. Then, in order to calculate the entropy of the stationary process, instead of considering the outcome of each random variable in the process independently, the average block entropy can be calculated using the joint entropy previously defined, leading to the following sequence:

$$\{\mathrm{H}(X_n)\}_{n=1}^{\infty} = \left\{\frac{1}{n}\,\mathrm{H}(X_1, \ldots, X_n)\right\}_{n=1}^{\infty}$$

Taking this definition of the average block entropy sequence, an evolved concept of entropy can be foreseen, which takes into account the correlations among the samples of the observed stochastic random process: the **entropy rate**.

Definition 2.4. Let $(X_n)_{n=1}^{\infty}$ be an stochastic process taking values on the alphabet $\mathscr{X}$. The entropy rate, H_R, of (X_n) is defined as the limit, if exists, of the sequence of average block entropy when the length of the block tends to infinity:

$$\mathrm{H}_R(X_n) = \lim_{n\to\infty} \frac{1}{n}\,\mathrm{H}(X_1, \ldots, X_n) \tag{2.5}$$

A related entropy measurement is the following one.

Definition 2.5. The conditional entropy rate of a stochastic process (X_n) is defined as:

$$\mathrm{H}'_R(X_n) = \lim_{n\to\infty} \mathrm{H}(X_n|X_1X_2\ldots X_{n-1}) \tag{2.6}$$

if the limit exists.

If (X_n) is also stationary, then the following theorem applies:

Theorem 2.1. *For stationary stochastic processes, both sequences* $\left\{\frac{1}{n}\mathrm{H}(X_1,\ldots,X_n)\right\}_{n=1}^{\infty}$ *and* $\mathrm{H}(X_n|X_1X_2\ldots X_{n-1})$ *are non-increasing and have a common limit, called entropy rate:*

$$\mathrm{H}_R(X_n) = \lim_{n\to\infty}\frac{1}{n}\mathrm{H}(X_1,\ldots,X_n) = \lim_{n\to\infty}\mathrm{H}(X_n|X_1X_2\ldots X_{n-1})$$

For n large, either of these entropy quantities constitutes an arbitrarily accurate approximation to the entropy rate of the process. Stationarity also implies that the samples of the process are identically distributed according to a common PMF. When, in addition, they are statistically independent, the process, or the samples thereof, is then called independent, identically distributed (i.i.d.). More colloquially, a process with independent samples is called memoryless or without memory. For an i.i.d. process, entropy rate and the entropy of individual samples coincide, that is $H(X_n) = H_R(X_n)$. For a general stationary process, $H(X_n) \leqslant H_R(X_n)$, with equality if and only if the process is memoryless. The highest entropy rate is attained by processes with independent, uniformly-distributed samples, that is $H_R(X_n) \leqslant \log|\mathscr{X}|$, with equality if and only if the process is uniformly distributed and memoryless.

These results are important in the context of the thesis, since the mobility model of a user will be defined as an stationary stochastic process, which allows to apply the equations presented above (see Chapter 3).

Since entropy rate emerges from considering the joint probability mass function of the different random variables of the stochastic process, it reflects also the correlations among such random variables, if any. The following chapters will reflect that this evolution with respect to the Shannon entropy allows to accurately estimate the randomness of users mobility by taking into account the temporal correlations (i.e., mobility patterns) in their movement histories.

It should be noticed that for any entropy calculation described so far, the probability mass functions of the random variable or stochastic process are needed. Unfortunately, this is one of the missing pieces in the human mobility puzzle: the accurate statistical description of a user mobility model is still unknown. Thus, in the following section some estimators helping to cope with this problem are presented.

2.2.2 Entropy and Entropy Rate Estimators of Finite Sequences

The previous section exposed the main entropy definitions that will be used in subsequent sections: maximum or Hartley entropy, Shannon entropy, and entropy rate. However, the probability mass function of the random variables or stochastic processes to be analyzed (i.e., the mobility model of the user) are largely unknown. For this reason, the need for using entropy estimators emerges. These estimators are obtained based on the available information: the movement history of the user, i.e., the observed outcomes of a realization of the stochastic process. This section describes some available estimators and justifies the choice of the ones that will be used throughout the following chapters.

Let (X_n) be an stochastic process, taking values on the alphabet $\mathscr{X}$, with cardinality $|\mathscr{X}|$. Let $S = s_1s_2\ldots s_n\ldots s_N$, be the finite sequence of observed outcomes of a realization

of (X_n), of length N. Therefore, the set of different values that s_n can take on, $\mathscr{S}$, is a subset of $\mathscr{X}$, $\mathscr{S} \subseteq \mathscr{X}$, with cardinality $|\mathscr{S}| \leq |\mathscr{X}|$.

The estimation of the Hartley entropy, $\mathrm{H}_H(X_n)$, is straightforward, just considering the cardinality of the alphabet $\mathscr{X}$ as the number of different symbols observed in the available sequence, S:

$$\hat{\mathrm{H}}_H(X_n) = \mathrm{H}_H(S) = \log_2 |\mathscr{S}| \tag{2.7}$$

For calculating the Shannon entropy estimator, the analogous process to the previous one is applied to equation (2.1). Since the probability mass function $p(s_j)$ is not available, it is approximated by a maximum likelihood estimator based on the observable data. Let $N(s_j)$ be the number of elements, s_n, of the observed sequence, S, that are equal to s: $N(s) = |\{n \in \{1, 2, \ldots, N\} : s_n = s\}|$. Then, the estimator of $p(s_j)$, is then calculated as follows:

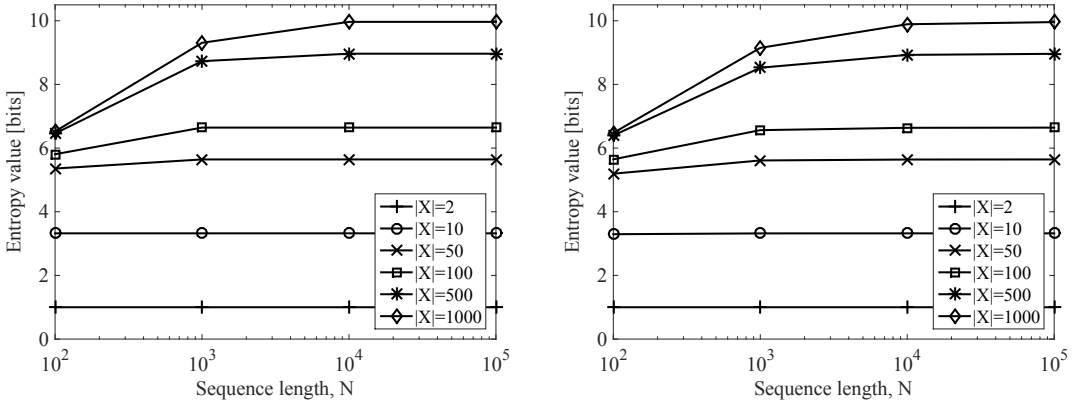
$$\hat{p}(s) = \frac{N(s)}{N}, \forall s \in \mathscr{S} \tag{2.8}$$

Therefore, the estimator for the Shannon entropy can be expressed as:

$$\hat{\mathrm{H}}_S(X_n) = \mathrm{H}_S(S) = -\sum_{s \in \mathscr{S}} \hat{p}(s) \log_2 \hat{p}(s) \tag{2.9}$$

Since the estimations are calculated based on a finite sequence of events, S, of length N, it must be considered the quality of the estimation based on the available data. Figure 2.2 shows the entropy values $\hat{\mathrm{H}}_H(X_n)$ and $\hat{\mathrm{H}}_S(X_n)$, respectively, for sequences of different lengths, drawn from uniform distributions taking values in alphabets with different cardinality. To perform this experiment, six random variables were considered, taking values on alphabets of cardinality, $|\mathscr{X}| = \{2, 10, 50, 100, 500, 1000\}$ each; and for each variable, six different sequence lengths were considered, $N = \{100, 500, 1000, 5000, 50000\}$ symbols. Then, 100 different realizations of the process (observable sequences, S) were generated for each possible combination of alphabet cardinality and sequence length. Applying equations (2.7) and (2.9), the values for $\mathrm{H}_H(X_n)$ and $\mathrm{H}_S(X_n)$ were estimated using the available data, S, to further average the results of the 100 realizations for each case. Since the uniform probability distribution is well-known, the real value of entropy is known and can be compared to the estimations. For the specific case of uniform random variables, $\mathrm{H}_H(X_n) = \mathrm{H}_S(X_n) = \log_2 |\mathscr{X}|$. Thus, the estimations should be as closed as possible to $\{1, 3.32, 5.64, 6.64, 8.97, 9.97\}$, respectively regarding the cardinality considered in each case. From the figures it can be observed that for low cardinality values (2, 10 symbols), the estimators reach always the theoretical values, independently of the sequence length. However, the higher the cardinality, the longer sequences must be in order to correctly estimate the entropy values. Otherwise, there is an underestimation of the real value because not every different symbol may have appeared in the sequence (i.e., the cardinality of the observed S, is lower than the cardinality of the real process, X_n, $|\mathscr{S}| \leq |\mathscr{X}|$), or not every symbol in the alphabet $\mathscr{X}$ has appeared with equal frequency.

Estimating the entropy rate of a finite sequence is more complex. In fact, the complexity stems from the problem of not having enough samples of the sequence so as to completely capture the probability mass function describing the model underneath. A first approach



(a) Hartley entropy estimations.

(b) Shannon entropy estimations.

Figure 2.2: Behavior of Hartley and Shannon entropy estimators for different combinations of alphabet cardinality, $|\mathscr{X}|$, and sequence length, N.

could be using the equation (2.3) and extending it with the well-known property of entropy:

$$\mathrm{H}(X,Y) = \mathrm{H}(X) + \mathrm{H}(Y|X) \tag{2.10}$$

Taking into account the available data, this leads to the estimation of the entropy rate by calculating block entropies of size k:

$$\hat{\mathrm{H}}_R(X_n) = \mathrm{H}_R(S) = \lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} \mathrm{H}(S_k|S_1,\ldots,S_{k-1}) \tag{2.11}$$

Applying the equation (2.4):

$$\mathrm{H}(S_k|S_1,\ldots,S_{k-1}) = -\sum_{s_1\ldots s_{k-1}\in\mathscr{S}^{k-1}} p(s_1\ldots s_{k-1}) \sum_{s_k\in\mathscr{S}} p(s_k|s_1\ldots s_{k-1}) \log_2 p(s_k|s_1\ldots s_{k-1}) \tag{2.12}$$

In order to apply the previous equations, the corresponding probability mass functions need to be estimated using the available data provided by S. Let $N(s_1 s_2 \ldots s_k)$ be the number of blocks $s_1 s_2 \ldots s_k$ of length k of the observed sequence, S, that are equal to some $\{s\}^k$, this is, to a block of k possible values of $\mathscr{S}$. The number of blocks of length k in the observed sequence, S, of length N is $N-k+1$. Then, the estimator of the joint and conditional probability mass functions are expressed as:

$$\hat{p}(s_1\ldots s_{k-1}) = \frac{N(s_1\ldots s_{k-1})}{N-(k-1)+1}, \forall s_1\ldots s_{k-1} \in \mathscr{S}^{k-1} \tag{2.13}$$

$$\hat{p}(s_k|s_1\ldots s_{k-1}) = \frac{N(s_1\ldots s_k)}{N(s_1\ldots s_{k-1})}, \forall s_k \in \mathscr{S}, s_1\ldots s_{k-1} \in \mathscr{S}^{k-1} \tag{2.14}$$

And, considering that S is finite, the H_R, sometimes known as block entropy, needs to be estimated by a finite number of blocks of different sizes:

$$\hat{\mathrm{H}}_R^m(X_n) = \mathrm{H}_R^m(S) = \frac{1}{m}\sum_{k=1}^{m} \mathrm{H}\left(S_k|S_k,\ldots,S_{k-1}\right), m \in [2,N] \tag{2.15}$$

using the probability mass function estimators in equations (2.13) and (2.14). As can be expected, the larger m, the more accurate the estimation, and also the harder the calculation process.

Figure 2.3 shows the $\hat{\mathrm{H}}_R^m(X_n) = \mathrm{H}_R^m(S)$ values calculated using the previous estimator for a uniform process, for different cardinality of the alphabet and sequence length, and choosing $m = 2$ and $m = 3$ for illustrative purposes. The expected results, considering the uniform distribution driving X_n, are the same ones than in the Hartley and Shannon entropies, since for uniform distributions all entropy and entropy rate values are the same. However, as can be seen, the behavior of this entropy rate estimator for the uniform distribution case makes no sense. Since all the possible blocks have the same probability, for each cardinality, there are $|\mathscr{S}|^2$ and $|\mathscr{S}|^3$ possible blocks of length 2 and 3, respectively (the maximum length considered by the entropy estimator). For random variables with alphabet of cardinality 2, the entropy estimations match the real value, but when the cardinality grows, the number of possible blocks (all equally probable) is so high that there are not enough samples in the sequence so that every block can appear. In these cases, $p(s_1 \ldots s_{m-1}) = 0$ for a big portion of the possible blocks, leading to $\hat{\mathrm{H}}_R^m \approx 0$. The figure also shows that for $m = 2$, the results are slightly better than for $m = 3$, due to the rapidly increasing number of possible blocks not appearing in the sequence. Thus, if m keeps on increasing, the quality of the estimations would keep on decreasing.

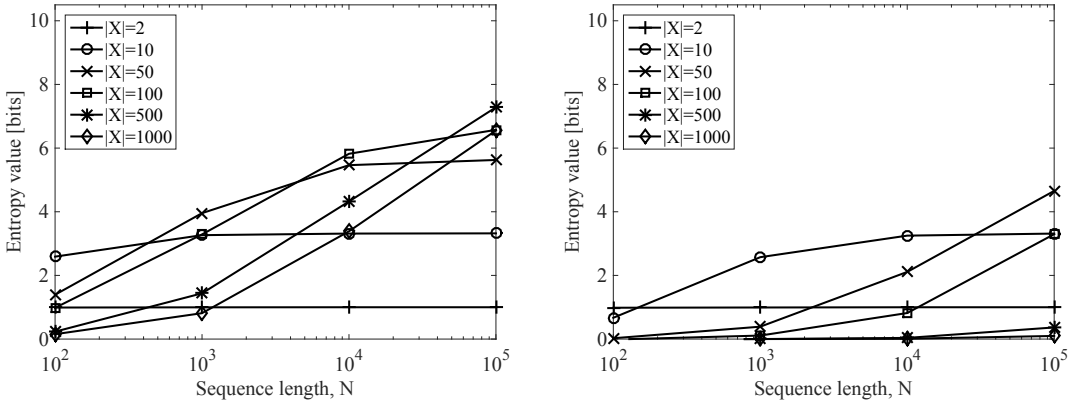
In these examples, different cardinality values up to 1,000 different symbols and sequences up to 10,000 symbols were considered. It should be noted that these calculations will be further applied to the mobility sequences of different individuals. Considering that the cardinality of the alphabet representing the set of visited locations by an individual is, in many cases, even higher than in the values considered in these examples, it can be foreseen that these estimators are unable to calculate accurate approximations to the real entropy rate of the sequences of locations visited by any individual. Besides that, a mobility model of a user has both short and long-range correlations of unknown order. Therefore, the optimal value of m to correctly capture all these correlations is unknown. For these reasons, another alternative estimator for $\hat{\mathrm{H}}_R(X_n)$ needs to be used in order to obtain more accurate estimations despite the finite length of the sequences.

An alternative entropy rate estimator is found in Grassberger *et al.* work [52, 74]. The authors propose an entropy estimator based on block lengths:

Definition 2.6. The Grassberger entropy rate estimator is expressed as:

$$\hat{\mathrm{H}}_R(X_n) = \mathrm{H}_R(S) = \left(\frac{1}{N}\sum_{i=2}^{N}\frac{\Lambda_i}{\log_2 i}\right)^{-1} \tag{2.16}$$

where Λ_i is the length of the shortest substring starting at index i of the sequence, S, that did not appear in the range $[1, i-1]$, and N being the length of the whole sequence.



(a) $m = 2$ (b) $m = 3$

Figure 2.3: Behavior of the entropy rate estimators for $m = 2$ and $m = 3$ for different combinations of alphabet cardinality, $|\mathscr{X}|$, and sequence length, N.

Figure 2.4 shows $\hat{\mathrm{H}}_R(X_n)$ calculated with the Grassberger estimator for the same cases than before. Although the estimations for the process drawn from alphabets of high cardinality are still not accurate, the behavior is more consistent than with the previous estimator for all the cases.

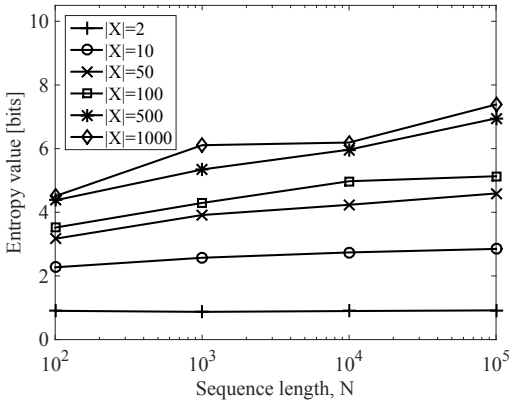


Figure 2.4: Behavior of the Grassberger entropy rate estimator for different combinations of alphabet cardinality, $|\mathscr{X}|$, and sequence length, N.

With these estimators—$\hat{\mathrm{H}}_H$, $\hat{\mathrm{H}}_S$, and $\hat{\mathrm{H}}_R$ given by Grassberger's formula—all the analysis related to entropy that will be presented in the next chapters can be finally performed, and the important differences between entropy and entropy rate, both in calculation and meaning, haven been exposed. In the next chapters, the generic symbol sequences used for the previous illustrative examples and definitions will be replaced by the sequences

representing the movement of the individuals.

2.3 Conclusions

Leaving aside the explanation of the entropy and entropy rate foundation, practical meaning and available estimators, all of them aspects that will become handy in many points of the following chapters, the rest of the chapter offered an overview of the main topics that this dissertation focuses on.

The first step is to choose the most convenient mobility data source. Four options were presented at the beginning of the chapter, namely GPS, Wi-Fi, cellular telephony networks, and LBSNs. The choice will be based on the parameters evaluated for each of them (coverage, location data accuracy, and battery consumption) and the requirements of the study to be performed, which will be presented in the next chapter.

Many different mobility data sets have been used in the literature to carry on different analysis. The ones reviewed in the previous sections included the most extensive and used ones. As can be observed, they come from different data sources and were collected using diverse collection schemes or sampling rates. Along with choosing the most convenient mobility data source, Chapter 3 will consider these available data sets, checking if they can provide the data needed for the analysis that will be carried out throughout the thesis. This evaluation will lead to propose a new data collection campaign to cover the gaps of the current available data sets that can potentially impact further analysis. These gaps are mainly two: first, the available data were taken, in some cases, many years ago and may not reflect the current users' behaviors; and second, the mobility data comes in many cases from participants with potential similar mobility patterns that do not represent the real differences of users' movement behaviors.

The literature review also shown the wide variety of human mobility-related studies, mainly classified into location or user-centric perspectives. These studies propose an extensive set of mobility features to characterize the movement of individuals. However, each study uses a different data set with no mention to how using each specific set of data influences the quantitative or qualitative conclusions drawn from the study. Chapter 4 will study the potential impact that the specific data used might have in the obtained results, and will propose several methods to cope with some of the mentioned biases already spotted by other researchers, such as the ping pong effect in cell-based mobility data.

The review of mobility prediction algorithms provides a similar conclusion than the review of studies on human mobility: there exists an extensive variety of algorithms, but there is a lack of metrics that allow to evaluate if the announced improvements on prediction come from the prediction algorithm itself or from the specific data used for the evaluation, due to their source, collection scheme, or mobility features of the individual generating such data. In Chapter 5, a specific set of prediction algorithms are evaluated with data coming from individuals with different mobility features, collected using different schemes, to show how these factors impact the resulting prediction performance. Besides, the mobility features considered will be used to design different modifications that aim to improve the prediction accuracy.

Chapter 3

Collection and Description of Human Mobility Data based on Cellular Networks

Contents

As argued in Chapter 1, one of the most interesting ways to personalize the behavior of applications is to take advantage of one of the components of user's context: the location. This aspect, both in their spatial and spatio-temporal perspectives, makes it possible to filter searches, locate interesting places around the user, infer her mobility behavior, check if she often visits a hospital (thus being plausible that she has some illness), church (she probably practices the corresponding religion), political building (she supports certain political group), school (she has children), among others. Therefore, an application knowing the mobility data of a user can benefit of a varied data collection that can be used to improve the service provided, by tailoring it to perfectly fit every specific user.

The first step to be able to design such services, is to have mobility data available, in such a way that it can be analyzed to extract the mobility features of the users, which can be further leveraged to fine tune the design of the applications. Collecting mobility data used to be a difficult task, and its was driven mainly by surveys, which involved a non trivial selection of participants that could probably bias the results, since the amount of people surveyed could not be very extensive, and because the detail of the mobility behavior that could be extracted from the survey was not enough to extract robust conclusions about mobility. This scenario has completely changed over the last decade, due to the enormous

growth of the use of mobile phones. They are not only a communication device, but they can also record extensive data collections about many aspects, mobility among them, due to the varied set of sensors integrated on them. Mobile phones overcame many of the shortcomings of previous data collection approaches.

However, the variety of sensors and systems integrated in the devices capable of collecting mobility data poses the question of which one to use. In order to ask this question, it must be taken into account the potential data that each one can provide and also the limitations of each system. This chapter describes the data source that have been used in the research described in this thesis, considering the options described in Chapter 2. Furthermore, three different approaches for data collection will be presented. Finally, the actual data sets to be used later in next chapters are presented. Among the variety of data sets presented in the related works on mobility data, the most complete so far, the MIT Reality Mining Data Set is described. However, after examining this data set, some shortcomings were noticed. Since any of the available data sets in the literature contained all the information required for the analysis done in next chapters, a new data collection campaign was launched, resulting in what was called UC3M data set. This campaign will be described, along with a comparative of the preliminary statistics comparing both the MIT and UC3M data sets.

3.1 Mobility Scenario Definition

Section 2.1.1 started by enumerating the main features to analyze when considering the mobility data source to use for mobility analysis:

- The environment where mobility wants to be considered: general mobility, or mobility restricted to certain bounded area.
- The static or dynamic dimension of mobility data, by considering just independent sparse locations, or a continuous tracking of the individual.
- The accuracy of the locations reported by the user.

Among the available choices described in Section 2.1.1—GPS, Wi-Fi networks, cellular telephony networks, and LBSN,—the cellular telephony network will be used throughout the thesis. Since at the moment of the experiments exposed along this work the LTE network was starting to be deployed, there are no data with this technology, and only GSM or Universal Mobile Telecommunications System (UMTS) will be considered. For simplicity, both cellular or GSM network will be used, interchangeably. Despite being the worst choice regarding location accuracy, the main reasons to choose this option are the following ones:

- The global coverage that allows to collect data both in outdoor and indoor environments (as opposed to GPS), and also in not populated areas (as opposed to Wi-Fi networks) like the commute paths people travel everyday (e.g., highways).

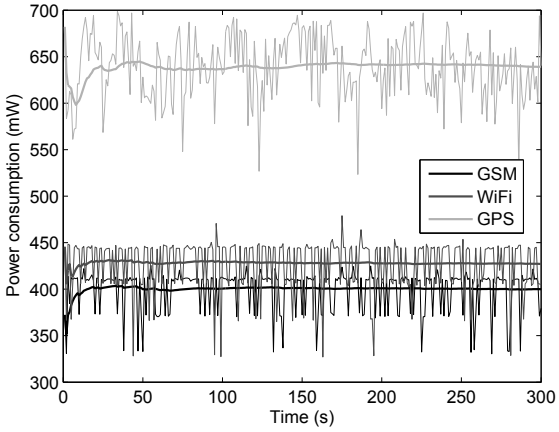


Figure 3.1: Instantaneous and average power consumption of different technologies used for location tracking.

- The lowest power consumption that allows a continuous mobility tracking (as opposed to GPS), thus generating data sets collecting frequent data without missing any visited location. Figure 3.1 shows the instantaneous and average power consumption of having enabled only the cellular telephony network connection, only the Wi-Fi antenna scanning for Wi-Fi networks, and only the GPS system requesting location fixes. The experiment was carried out with an HTC Desire mobile phone, with full battery and screen bright dimmed. Each experiment was run for more than 5 minutes (in order to avoid transitional periods) and repeated 10 times. In the figure it is shown the average of the 10 realizations, both for the instantaneous (thin traces) and average over time (thick traces) power consumption. As shown in the figure, the GPS consumption is much higher than both GSM and Wi-Fi ones, thus leading to a quick battery drain that does not allow to keep the tracking mechanism running for more than 4 hours without completely run out of battery.

- The data collection process does not depend on the user (as opposed to using LBSNs), thus existing no locations filtered out for any other reason than the possible constraints of the technology itself, in other words, the user cannot select which locations to report and which ones to hide.

In the cellular network scenario, the whole space is split into different areas corresponding to the coverage area of each BTS providing the cellular telephony service. Each of these areas, also known as cells, has their own identifier, called Cell Identifier (CellID). For network purposes, the cells are grouped into what is known as location areas, each of them labeled also with different Location Area Code (LAC). Thus, each cell is uniquely identified by its LAC and CellID. For simplicity, the unique pair (LAC, CellID) that represents each cell will be referred to as CellID.

With this scenario, there are several ways to describe the location of a user, based on the BTS her phone is attached to, and depending on the domain:

- Considering the physical domain, the location of the user can be approximated using the coordinates of the BTS her mobile phone is attached to.
- Considering the symbolic domain, the user location can be identified by the CellID of the BTS (or cell) her mobile device is attached to.

Since all the mobility analysis and applications considered throughout this work use symbolic locations as input, we will use the symbolic domain. However, as can be observed, the translation from one domain to the another one can be easily done if the correspondence between CellIDs and BTSs locations is known.

Therefore, the movement history of a user will be represented by a sequence of symbols (CellIDs), each of them representing a different BTS or cell visited by the user. From now on, when referring to the locations visited by the user, the terms cell, BTS, and location will be used interchangeably. The movement history is a deterministic sequence, describing what happened in the past. It is just one possible realization of the statistical process describing the mobility model of the user. However, the goal of mobility research is to construct the mobility model of the user in order to be able to foresee her future movements. In order to do so, the mobility model of a user can be initially defined as follows:

Definition 3.1. The mobility model of a user is a stationary stochastic process

$$(L_n)_{n=1,2,\ldots} = L_1, L_2, \ldots L_n \ldots$$

representing the sequence of locations over discrete time instants $n = 1, 2, \ldots$ The corresponding location L_n at time n is a discrete random variable on the alphabet $\mathscr{L}$, corresponding to the set of different locations (or CellIDs) visited by the user.

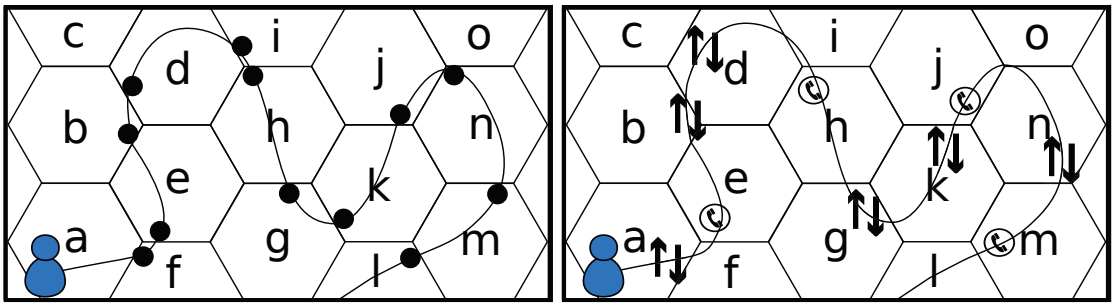
Stationarity means that the process is invariant with respect to time shifts:

$$Pr\{L_1 = l_1, L_2 = l_2, \ldots, L_n = l_n\} = \Pr\{L_{m+1} = l_1, L_{m+2} = l_2, \ldots, L_{m+n} = l_n\}$$

Then, the **location history**, **movement history**, or **trace**, $l = l_1 l_2 l_3 \ldots l_N$, is a finite time series of length N, extracted from one realization of (L_n), representing the locations already visited by the user in time instants 1 to N, $l_n \in \mathscr{L}$.

Finally, having the scenario and mobility model defined, the last step is to decide how to actually collect the sequence of CellIDs that will represent the movement history of the user, l. Both the cellular network and the user's mobile device are aware of the cell the device is attached to, but in different ways. Thus, different data collection schemes are possible, resulting in different types of trace:

- Every mobile device knows the CellID of the cell it is attached to at every moment. Therefore, in this case the location history collects every cell change experienced by



(a) Events recorded by the device. (b) Events recorded by the network.

Figure 3.2: Events recorded by each location data collection scheme.

the device. This approach will be referred to as **baseline data collection scheme**. The cases when the mobile device cannot connect to the network (e.g., no coverage in a specific place, network problems...) are not part of the trace, since these cases do not represent any place, just a particular state of the network itself. Only one symbol per location is recorded, thus no two consecutive equal CellIDs can be captured in the history. In the case represented in Figure 3.2a, the location history of the user would be $l^{baseline} = afebdihgkjnml$, i.e., every cell change detected by the device (represented by black spots).

Put mathematically, let $\mathscr{L}$ be the set of CellIDs corresponding to the different cells visited by the user. The complete movement history or trace, $l^{baseline}$, is defined as the temporal sequence of cells, $l_n \in \mathscr{L}$, the device was attached to as the user moves, $n \in [1, N]$, being N the length of the trace:

$$l^{baseline} = l_1 l_2 l_3 \dots l_N, l_n \in \mathscr{L}, l_i \neq l_{i+1} \forall i \tag{3.1}$$

- The mobile telephony network collects information regarding the cell the user's device is attached to every time the user sends or receives voice calls and text messages. This information is stored in what is known as CDR. Therefore, the **CDR-based data collection scheme** records a different version of the mobility trace of the user, where only the CellIDs of the cells the device was attached to when the user performed some network event (calls, messages). In this case, there could be two (or more) subsequent CellIDs which are equal, since the user could make a call today and the next one tomorrow at the very same place. With this approach, the user in Figure 3.2b would have a location history like $l^{CDR} = ehjm$, which corresponds to the sequence of cells where a voice call (or a message event) took place.

  Put mathematically, let $\mathscr{L}$ be the set of CellIDs corresponding to the different cells visited by the user. The CDR-based movement history or trace, l^{CDR}, is defined as the temporal sequence of cells, $l_n \in \mathscr{L}$, the device was attached to as when the user sent or received a phone call or text message, $n \in [1, C]$, being C the number of CDR-related events (calls or messages):

$$l^{CDR} = l_1 l_2 l_3 \dots l_C, l_n \in \mathscr{L} \tag{3.2}$$

It is worth to notice that l^{CDR} is a sampled version of $l^{baseline}$, which sampling frequency depends on the call frequency of the user: the more calls or messages she sends or receives, the more complete l^{CDR} we would get.

- The mobile telephony network also collects information regarding the cell the user's device is attached to every time the user transfers data (through an Internet connection using the telephony network). This information is also stored in a location history that will be referred to as **Data Detail Record (DDR)-based trace**. It contains a different version of the location history of the user, where only the CellIDs of the cells the device was attached to when the user performed some data transferring event. With this approach, the user in Figure 3.2b would have a location history like $l^{DDR} = abdgkn$, which corresponds to the sequence of cells where the user used her Internet connection through the telephony network.

  Put mathematically, let $\mathscr{L}$ be the set of CellIDs corresponding to the different cells visited by the user. The DDR-based movement history or trace, l^{DDR}, is defined as the temporal sequence of cells, $l_n \in \mathscr{L}$, the device was attached to as when the user used the data connection, $n \in [1, D]$, being D the number of DDR-related events:

  $$l^{DDR} = l_1 l_2 l_3 \dots l_D, l_n \in \mathscr{L} \tag{3.3}$$

  It is worth to notice that l^{DDR} is a also a sampled version of $l^{baseline}$, which sampling frequency depends on the frequency of the data connection usage: the more frequently the data connection is used, the more similar $l^{baseline}$ and l^{DDR} would be.

As mentioned above, the use of each data collection scheme generates traces, l, with different characteristics in terms of the locations recorded in them. As can be expected, these differences will impact the mobility features reflected, which will be described in the next chapter. However, in order to make such analysis, it is not enough with the previous definitions, but some real data is needed. In the next section, the data sets that will be used throughout the thesis will be described.

3.2 Description of the Mobility Data

As previously said, the first step in analyzing mobility and proposing related applications is to collect a data set representing the mobility features of the users involved in the data sample. The goal of this section is to select and describe the two data sets used throughout this work: one taken from the data sets available in the literature, reviewed in Section 2.1.2, and a new one generated in the framework of the thesis, which collection campaign will be also described next.

3.2.1 Mobility Data from the Literature: the MIT Data Set

In Section 2.1.2, many of the data sets used in the literature on human mobility were reviewed. Focusing on the ones collecting GSM data and that are available for their public use, listed in Table 2.1, only two of them use the baseline data collection scheme: the MIT

and Rice Context data sets. They have also available the timestamps of call, message and data events, so that the CDR and DDR-based traces can be inferred. The MIT data set is comprised of 95 traces collected during 9 months, whereas the Rice Context data set gathers 14 traces of 3 weeks of duration. The rest of the data sets capture only CDR-based data. Considering the sample size and duration in each of the two cases, the MIT data set was chosen.

It was collected back in 2004 for the the MIT Reality Mining Project [43]. It includes information about 75 students of faculty in the MIT Media Laboratory, and 25 incoming students at the MIT Sloan business school adjacent to the Media Laboratory. To the best of our knowledge, the Reality Mining experiment was the first to collect tracking information of a significant amount of people and during a long period through their mobile phones.

The experiment consisted on providing a Nokia 6600 mobile phone to each of the 100 participants during 9 months. The mobile phones were equipped with several pieces of software in charge of continuously collecting information related to the mobile phone, including call logs, Bluetooth devices nearby, CellID of the cells to which the phone was attached to, application usage, data traffic events, and phone status.

Although the complete data set is made of 106 users, the location information based on CellIDs is only available for 95 of them. Therefore, the data set finally handled is the one made up of these 95 users. The data was recorded during an academic year, but the length of the traces vary from one user to another. Figure 3.3 shows the duration of the MIT traces in the temporal context.

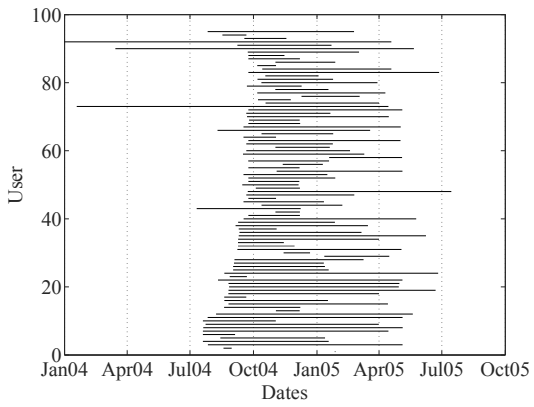


Figure 3.3: Time span of the location histories included in the MIT data set.

3.2.2 New Mobility Data Collection Campaign: the UC3M Data Set

Although the MIT data set was very useful in analyzing mobility, it was collected several years ago when the cell phone usage was different. For instance, at that time, data traffic in mobile phones was very rare, whilst phone calls were probably more frequent. Besides that, the subjects conforming the study were all MIT students or faculty, who may have

followed similar patterns (since they study or work in the same environment, and have similar timetables). There is not enough information in order to confirm these hypothesis nor to discard them. Therefore, aiming at having a different data set, more recent and whose subjects were not related in locations nor habits, a new mobility data collection campaign was launched in the framework of this thesis, which is described in this section.

3.2.2.1 Subject Pool and Campaign Duration

The call for participation in the mobility data collection campaign was launched to approximately 65 people in our closest network, including colleagues and also many people not related to the department nor the university. Fortunately, 25 people out of the 65 initially asked, kindly accepted to participate in the campaign. Among them, there are people living in 5 different countries, and working in completely different and independent places, thus not sharing the same space or specific timetables, like in the case of the MIT data set.

It must be noted that the subjects volunteered to become part of the experiment under no conditions nor rewards. This fact potentially led to two consequences: a lower participation, and a variable duration of the subjects collecting data. Figure 3.4 shows the duration of the collected traces. As can be observed, it ranges from several weeks to approximately a year and a half. All the users who stopped using the application reported the cause as having changed the device, and forgetting to reinstall the application in their new device.

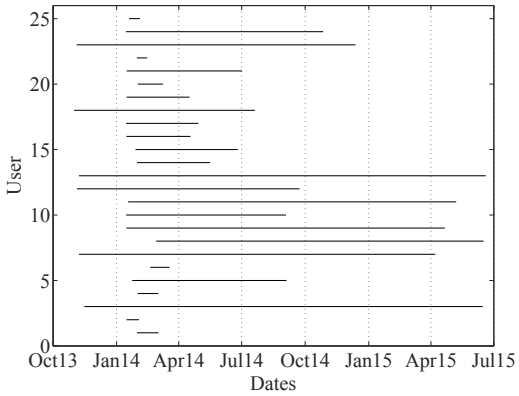


Figure 3.4: Time span of the location histories included in the UC3M data set.

3.2.2.2 Monitoring and Logging Application

As mentioned in Chapter 1, mobile phones are the vehicle that allows a massive data collection of several aspects related to human mobility, given that they go with their owners practically all day long. For this reason, all the data collected during the campaign comes from the mobile device of each participant. They were asked to download and install the monitoring application in their devices, which keeps on continuously running on the

background. This application, described more in detail in Appendix A, monitor all the cell changes, call, and data traffic events, among others, to record them into a file and finally send the file to an e-mail account where all the data is stored for further processing. Every two weeks, the participants were notified to send the new data to the e-mail account, which could be done in two screen touches in order to make it extremely easy and avoid more inconveniences to the user. The final result was positive, since no data was lost and the participants did not complain about the method, nor do they reported stop using the application because of this reason.

3.2.2.3 Data Description

The data collected includes the CellID of the cells to which each subject's device was attached at all times, call events, and data traffic events, which are the main information that will be used in the thesis. However, additional data was collected. Table 3.1 describes all the events registered by the mobile phone application, providing also examples of the collected traces. As can be observed in the examples, JavaScript Object Notation (JSON) was the selected format to encode the information.

Each record has three fields: the type of event, the information about the event, and a timestamp. Depending on the event type, the information varies. For some of the events, it is just an identifier indicating different states, as specified in the table. In other cases, the information is more complex. For instance, for cell change events, the information includes the type of network (GSM or CDMA), and the data about each cell detected by the mobile phone, meaning the one it is connected to and the neighboring ones from which it receives signal. These data include the CellID, LAC, Mobile Country Code (MCC), Mobile Network Code (MNC), the signal strength received, the specific type of network (GSM, UMTS, HSPA...), and a flag indicating if the cell is the one to which the device is registered or not. Another event with more complex information associated is the service state. Unfortunately, by processing the information associated from this event coming from all users, it was found out that each device code the information of this event in completely different ways, thus making it impossible to parse the information due to the lack of fixed structure and content. Regarding the screen events, they were captured to make sure that the devices were still collecting data even when the screen is off. This checking measure was taken after observing problems to keep monitoring events when the device had the screen off in past versions of Android. Luckily, these problems were fixed when the application was developed and launched among the subjects.

3.2.2.4 The UC3M Mobility Data Set

The raw data collected by the mobile devices were processed to obtain the final data set used in the following chapters. For each user, three sequences of pairs {timestamp, CellID} were extracted, corresponding to the baseline, CDR and DDR-based traces. The baseline trace is extracted directly from the cell change events, just by collecting the LAC and CellID from each cell, together with their timestamp. In order to simplify, an auxiliary table mapping each pair {LAC,CellID} to a unique identifier was also stored for each user,

Event Type (Id)	Event Info (Id)	Example
Application (1)	Stop (0) Start (1)	{"timestamp":1383585792577,"eventInfo":1, "eventType":1}
Data Traffic (2)	None (0) In (1) Out (2) Inout (3) Dormant (4)	{"timestamp":1383585793740, "eventInfo":0, "eventType":2}
Cell Change (3)	network type, [{cellId, lac, mcc, mnc, signal strength, network type, registered},...]	{"timestamp":1383585803570, "eventInfo": {"networkType":0, "cellsInfo": [{"mCellSignalStrengthGsm": {"mAsu":18, "mDbm":-77, "mBitErrorRate":0}, "mCellIdentityGsm": {"mMcc":214, "mCid":18062, "mPsc":149, "mLac":1158, "mMnc":3}, "mNetworkType":10, "mRegistered":true}, {...},...]}, "eventType":3}
Screen (4)	Off (0) On (1)	{"timestamp":1383585792577,"eventInfo":1, "eventType":4}
UserMessage (5)	Message written by user	{"timestamp":1383582993740, "eventInfo":"train", "eventType":5}
ServiceState (6)	{operator name, roaming/ home,...}	"0 home simyo 21403 (manual) HSPAP CSS not supported -1 -1 mDataStateÕ03d0 RoamIndÕ03d-1 DefRoamIndÕ03d-1 EmergOnlyÕ03dfalse mIsVoiceSearchingÕ03dfalse mIsDataSearchingÕ03dfalseDual carrier0"
Call Forwarding (7)	Off (0) On (1)	{"timestamp":1383585792577, "eventInfo":1, "eventType":7}
Call State (8)	Idle (0) Ringing (1) Offhook (2)	{"timestamp":1383585793772,"eventInfo":2, "eventType":8}
Data Connection (9)	Disconnected (0) Connecting (1) Connected (2) Suspended (3)	{"timestamp":1383585793739, "eventInfo":{"networkType":15, "state":2}, "eventType":9}

Table 3.1: Set of different event types, their associated data, and examples of each in a trace recorded in the UC3M data collection campaign.

Cell Mapping			Baseline Trace		CDR Trace		DDR Trace	
Id	LAC	CellID	Timestamp	Id	Timestamp	Id	Timestamp	Id
1	3	30646	t1	2	t'1	2	t"1	5
2	3	31116	t2	3	t'2	2	t"2	4
3	8	14953	t3	5	t'3	4	t"3	5
4	59	21582	t4	3	t'4	2	t"4	3
5	70	7006	t5	1	t'5	3	t"5	1
...			...		...		...	

Table 3.2: Structure of the UC3M data set extracted from the collection campaign.

then using the unique identifiers in the traces. The CDR-based traces were generated by collecting the CellID of the cell to which the device was registered during the timestamps associated to call states ringing or off-hook. Finally, the DDR-based traces were extracted by collecting the CellID of the cell to which the device was registered during the timestamps corresponding to data traffic events in, out or inout. From these traces, the consecutive CellID repetitions detected, caused by many data traffic events in the same cell, were filtered out. However, in the CDR-based traces, the repetitions were only filtered if the time difference was lower than a minute (which corresponds to a call retry).

The set of traces and the mapping table are stored in a Matlab structure array, where each structure corresponds to the data of a participant, with a format like the one shown in Table 3.2. In this case, the timestamps were stored in Matlab format in order to make it easier to handle them in further analysis.

3.2.3 Overall Comparison of the Data Sets

In order to initially compare both data sets, Table 3.3 shows the values providing a first glance to the main differences among the traces collected in each data set. First of all, the dates are 10 years apart, which leads to noticeable differences in the communication habits of the users. Although the number of average calls per day is still similar, the number of data traffic events greatly differs. The UC3M data set shows the main shift in the use of mobile phones, given by the massive use of data traffic nowadays, showing an average number of data events of 42, whilst the MIT data set has an average value of 3. Thus, it would be interesting to study if the DDR-based traces can capture individual mobility features better than CDR-based traces, now that they collect seven times as many events as the CDR-based ones.

The number of participants is also different, being the MIT data set the one with a more extensive set of subjects, and thus, registering a higher number of cumulative days among all the users. In average, 153 days per user were collected in the MIT data set, whilst this quantity increases up to 229 days for the users in the UC3M data set. Thus, a lower number of traces have been collected, but with longer lengths.

The cell changes and different cells per day are also quite different. However, since there is no knowledge about the deployment of the BTSs for any trace, it can not be concluded

Feature	MIT data set	UC3M data set
Dates	2004/01 - 2005/07	2013/10 - 2015/07
Total duration (days)	14,487	5,716
Number participants	95	25
Average cell changes per day	238	159
Average different cells per day	31	43
Average call events per day	9	7
Average data events per day	3	42

Table 3.3: Summary of the general features of the MIT and UC3M data sets

if these differences are founded on the density of BTS in the region, or because of different behaviors of the users.

In the following section, more metrics like the randomness of the users in each data set will be analyzed, and thus it could be confirmed if the MIT users share common patterns, as suggested by the subject pool selected in that case.

3.3 Conclusions

The first step when studying mobility is to carefully select the best data source that provides the most faithful and complete mobility data, considering any potential constraint of the application at hand. In the case of mobility prediction in mobile devices, the main constraint, besides the computational capabilities of these devices, is the battery consumption. Thus, continuously tracking the user using the GPS of the device becomes the worst option due to its high consumption. Therefore, other sources that can provide information which can be translated into mobility data are considered. Among these options, the wireless networks are a popular approach. Aiming at being able to continuously track the user locations, cellular telephony network is selected due to its global coverage (as opposed to Wi-Fi, Bluetooth and other small range wireless networks) and the lowest impact on battery consumption.

When using the mobile network, three different data collection schemes can be considered to retrieve the location data: baseline scheme, when the user's device is in charge of recording every single cell change it experiences; CDR-based when the location data is extracted from the operator's CDRs, thus collecting just the cell the device was attached to during calls or messages; and DDR-based, when the mobility data is retrieved from the equivalent to CDRs for data traffic. The main advantage of the last two ones is that network operators store these records containing the mobility-related information for millions of users, which makes further studies based on them statistical significant. However, the quantity of locations based on cell data might be not enough to faithfully reflect mobility features. The baseline movement histories, on the other hand, record all possible mobility-related data, but the available volume of data gathered this way is much smaller, due to the difficulty of getting a wide set of users installing applications to collect these data, even if the application executes seamlessly in their devices. The next chapter will focus on

analyzing how well each of the three approaches reflect the mobility features of the users taking into account the constraints of each one.

In order to conduct such analysis, some data sets are needed. The analysis performed along this dissertation is based on the best data set found so far in the literature, which provides all the data needed for the study—the MIT data set. However, after processing these data from several perspectives, it was found that it presents some shortcomings: it was recorded more than 10 years ago, when the calls and data traffic profiles were quite different than nowadays and, besides, the data set is comprised of traces from users studying or working in the very same institution, which raised the suspicions of potential correlations (same campus, timetables, academic calendar, etc.) that might bias the results of the analysis that will be presented in the next section. For this reason, it was decided to collect new data from users geographically distributed, and with completely diverse occupations. This initiative led to the UC3M data collection campaign, from which a new data set was generated. The UC3M data set includes data about the cell changes, calls, data traffic, among other events which can be interesting for other studies. All these data were combined to obtain the baseline, CDR and DDR-based traces used for all the research carried out along the thesis.

With this diverse and updated data, together with the MIT data set, a mobility study will be presented in the next chapter, from which it is expected to derive useful conclusions helping to understand and improve the prediction algorithms considered in the thesis.

Chapter 4

Extraction and Analysis of Human Mobility Features Reflected in Mobility Data based on Cellular Networks

Contents

Going back to the flow diagram depicted in Figure 1.1, the previous chapter covered its first block, focused on the mobility data collection. The output of this first stage is composed of two mobility data sets, presented in Section 3.2. These mobility data allow to advance towards the next step of the diagram: the extraction and analysis of the mobility features reflected on them. The goal at this stage is to unearth interesting information on how human mobility works through the mobility features reflected, but apparently hidden, in the users' movement histories.

Section 2.1.3 thoroughly reviewed many of the human mobility-related studies existing in the literature. This analysis shown the wide variety of features that can be studied from different perspectives, as well as some of the concerns raised by the research community about the shortcomings and potential biases of the current studies. With these ideas in mind, the next sections focus on selecting the specific features that will be considered in the chapter. Then, the results of analyzing each feature extracted from the available data sets are presented. Recalling Section 3.1, three collection schemes were presented, namely baseline, CDR-based, and DDR-based, thus the comparison of the results extracted from the data collected using each collection scheme will be highlighted. The focus of this comparison will be placed on determining if the conclusions on the users' mobility features remain the same independently of the mobility data used, as some of the existing studies suggest.

By carefully inspecting the traces coming from the baseline collection scheme, and according to previous works found in the literature, the so-called ping pong effect was noticed in the data. In order to prevent the potential bias this effect introduces, different filtering techniques are proposed and evaluated, based on the features reflected in the traces coming from the baseline collection scheme and the ones extracted form the filtered traces. These comparisons raise a discussion on the importance of a wise choice of the mobility data to be analyzed and its potential effects on the prediction phase, exposed in the conclusions of the chapter.

4.1 Human Mobility Features in the Symbolic Domain

As discussed earlier in Chapter 3, the mobility data to be considered along this work is based on the CellID identifiying the BTS the user's mobile device is attached to as she moves. When using the mobile telephony network for mobility-related purposes, mobility is not reflected in a physical domain described by longitude and latitude coordinates, and thus the mobility features to be studied must be translated from the usual physical domain to the symbolic one.

The related works discussed in Section 2.1.3 presented a wide variety of mobility features that can be considered. Since mobility prediction, which will be studied in the next chapter, is focused on the mobility of each individual, the features to be considered are the ones concerning user-centric mobility, thus neglecting other aspects related to city, campus, or building mobility perspectives, among others.

Among the numerous user-centric mobility features considered in the literature (refer to Section 2.1.3.2 for their review), some of them can be translated into the symbolic domain for their study. As mentioned in Section 3.1, the mobility data considered in this thesis is made up of symbolic locations, with no physical coordinates. Therefore, the mobility features to be studied will be those that own a translation to the symbolic domain, as explained next:

- The concept of **amount of movement** is usually measured as the distance travelled by a person during certain reference time period, say an hour or a day. Many of the works in the literature focus on the distribution of the distance covered during

the individuals' displacements. However, since the real coordinates of each BTS (i.e., each cell) of the available data sets are completely unknown, in this thesis the amount of movement will be approximated by the number of cells visited per reference time period.

- A different perspective comes from focusing on the area covered by the person's displacements during certain reference period of time. This measurement allows to distinguish a person who travels certain amount of kilometers per day by going back and forth from home to work several times, from another person traveling the same distance, but visiting five different places. In this last case, the visited places are be more diverse, and thus the next location more difficult to predict. In the reviewed literature, this features is represented by the radius of gyration of the user, that measures the diameter of the area covered by the user. Again, due to the lack of real coordinates, this **diversity of the visited locations** will be represented in the symbolic domain by the number of different cells visited during certain period of time.

  A slightly different standpoint is the rate at which the person visits new places never seen before in her movement history. This feature seems potentially useful, as it points out time periods in which no prediction can be ever trusted, since there is no past information about these new places from which predictions can be built. This concept is directly translated as the number of cells not visited before per amount of time.

- Many works studied the distribution of the time spent at different locations, also described as the detection of the salient locations visited by an individual. Since this is one of the most analyzed aspects of mobility in the literature, it will be also considered in this thesis. In this case, the **salient locations** visited by the user will be measured by the fraction of the total number of visits paid to each of the different visited cells.

- There is even a different perspective about people mobility related to their **randomness**. Two people can travel the same distance per day among the same number of different places. Keeping the snapshot of one day, they both can seem similar in terms of mobility. However, if one of them takes always the same route to visit those places, whilst the other person visits those places in a different order each day, then the behavior of both individuals is very different. The movement of the first one is easy to predict since her patterns are always the same, whereas the movement of the second person encloses a higher uncertainty. There is one well-known concept in information theory used to measure, precisely, this uncertainty: the entropy of a sequence of symbols (refer to Section 2.2 for more details on the entropy concepts used along the dissertation). This concept will be used to characterize the uncertainty of the movement history.

- **Predictability**, already mentioned among the related works in Section 2.1.3, directly relates to the previous concept of movement uncertainty. This measurement

is clearly tied to the movement predictions that will be studied in Chapter 5, since predictability sets the upper bound of the prediction accuracy that any algorithm could ever achieve, depending on the movement history. Song et al. [134] defined this concept with the following expression, based on Fano's inequality:

$$\mathrm{H}_R(L_n) = -P_{max}\log_2(P_{max}) - (1-P_{max})\log_2(1-P_{max}) + (1-P_{max})\log_2(|\mathscr{L}|-1) \tag{4.1}$$

where P_{max} is the maximum predictability of the user, H_R is the entropy rate of her movement history, and $|\mathscr{L}|$ is the number of different locations visited by the user.

In the next section, the data sets presented in Section 3.2 are analyzed under the set of features just described, stressing out the impact of considering the different movement histories ($l^{baseline}$, l^{CDR}, and l^{DDR}) on the features observed.

4.2 Impact of Mobility Data Collection Schemes into Observed Human Mobility Features

The previous section described the set of mobility features with potential impact on the prediction process. But before checking its relationship with prediction, which will be covered in Chapter 5, an analysis of the features enclosed in the mobility data sets presented in Section 3.1 is carried out in this section.

Most of the research carried out so far regarding user-centric mobility gravitates around the topics covered in Section 2.1.3.2: most visited locations, distribution of displacements, etc. However, each of these studies was conducted by using a single type of data, without taking into account the bias introduced by the type of data used. The question arising here is if the mobility features are equally reflected in all location data types, or if the conclusions on human mobility can depend to some extent on the data used. This section is in charge of answering this question by considering some of the most used data sets in the existing literature, CDRs-based data, in contrast with more complete data coming from the baseline traces or from DDR-based data.

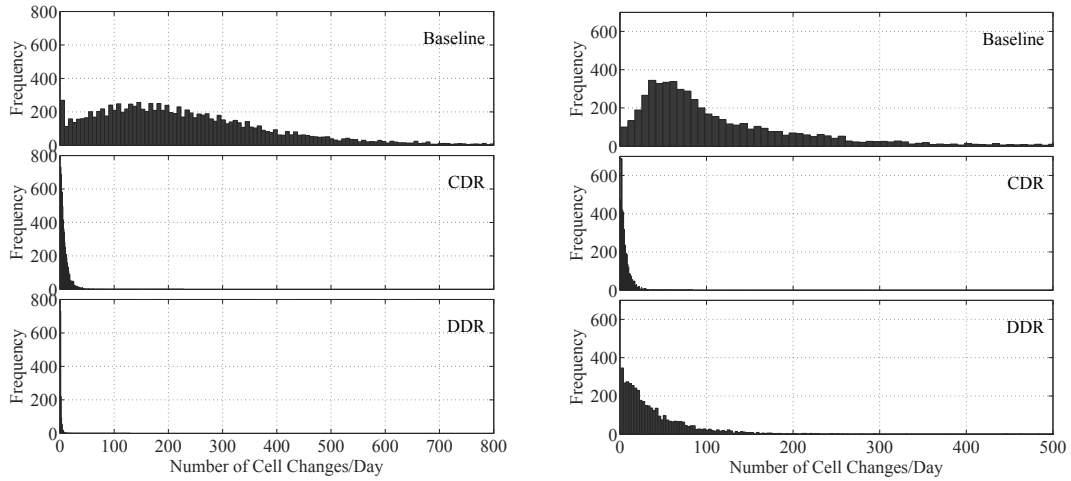
Specifically, the aim is to explore the impact of collecting mobility information using the three different approaches described in Section 3.2. Recall that these approaches generated the movement histories $l^{baseline}$, l^{CDR}, and l^{DDR}, respectively, which will be considered for each user in the MIT and UC3M data sets. The results drawn from the analysis would help to determine the data collection approach that most faithfully reflects the user mobility, since it would be the one providing the best movement history to serve as data source to further predict next movements of the user.

4.2.1 Amount of Movement

To start the analysis, the focus lies on the daily amount of movement performed by the users. In order to have a general idea of the distribution of this metric in the two data sets considered, each movement trace is split into days and the number of cell changes performed during each day is accounted. Days with no cell changes are neglected because the focus is

on the movement features that have potential impact in the prediction process that will be studied in a further chapter. Since the prediction algorithms considered work when each new CellID is recorded into the movement history, when there are no cell changes, there is no useful information for these algorithms and, for this reason, this no cell change situation is not considered in any of the analysis done throughout this section.

Figure 4.1 shows the distributions drawn from the aggregation of all days for all users in each data set, independently, namely MIT data set in subfigure 4.1a and UC3M data set in subfigure 4.1b. For each subfigure, three plots are presented, displaying the results of the metric when calculated over the movement histories resulting from collecting the mobility data using the baseline approach, $l^{baseline}$, the CDR-based approach, l^{CDR}, and the DDR-based approach, l^{DDR}, explained in Section 3.1. First, focusing on the differences among data collection approaches, it becomes clear that they severely impact the mobility features reflected into the resulting movement histories. $l^{baseline}$ shows a much wider distribution ranging from 1 to around 2,000 cell changes per day (refer to Table 4.1 for specific statistical values of the distribution). l^{CDR} and l^{DDR} lead to more than one order of magnitude smaller maximum values. The reason behind these facts roots in the way the mobility data are collected in each case. What l^{CDR} and l^{DDR} are really reflecting is not the amount of movement, but rather the number of calls or data transfer events per day, which is usually smaller compared to the cell change rate due to movement of the user (except for the cases in which the user does not move), reflected by $l^{baseline}$. Taking a closer look at the average values, the median of cell changes per day reflected by l^{CDR} is 6 and 4, depending on the data set, which seems a reasonable number of calls a person can make during one day. Regarding the value corresponding to l^{DDR}, it is worth to recall that when the MIT data set was collected, data traffic in mobile phones was not very spread yet, which leads to the median value of 1 event per day in this data set. For the UC3M data set, this value increases up to 27 events per day. Considering the frequent use of data connections in the mobile phones nowadays, this value can seem low to represent the number of data events per day. It should be noted that data traffic has a bursty behavior, meaning that when a user performs data traffic-related activities with the mobile device, they make an intensive use during certain period of time (say minutes or up to hours), and then stop using it for a while. Thus, during the period of time the user is generating data traffic, there are many data events every few seconds. That leads to having many subsequent events with the same associate CellID, which do not represent the movement of the user. Therefore, in order to make the analysis, when two subsequent events have the same CellID associated, the last one is filtered out to delete this deception coming from the data traffic behavior. Note that when considering call events, there exists a similar side effect, but coming from very different behavior. In this case, people tend to make calls in the same places (e.g., home, work place, etc.). For this reason, when visually inspecting the CDR-based movement traces, they show also a large amount of repetitions. However, the time space between them is of hours and even days. Therefore, some movement happened in between but it was not captured by the movement history, l^{CDR}. In this case, the repetitions were not filtered out since, in many cases, the number of events after this filtering were too small to be meaningful, and because there might been some movement between events with the same CellID with separation of hours or days between them. Even by considering



(a) MIT data set. (b) UC3M data set.

Figure 4.1: Distribution of the cell changes per day reflected in the traces collected with each data collection scheme—(from top to bottom) baseline, CDR, and DDR-based—of the data sets considered.

repetitions, the median is 6 and 4 for the MIT and UC3M data sets, respectively, which provides an idea of the limited power of l^{CDR} to reflect real mobility features of the users.

An interesting aspect reflected by the results of $l^{baseline}$ is the long tail of the distribution. Figure 4.1 does not show the complete long tail distribution for visualization reasons (the tail for the baseline approach is much longer than the one form CDR and DDR-based approaches, thus only the most significative part of the distributions are displayed to make the comparison possible). To provide the whole picture of the distribution drawn from the baseline approach, Figure 4.1 shows it together with the known distributions that best fit the empirical data. This long-tailed distributions are in line with the long-tailed distribu-

Data Set	l	Max	Min	Mean	Median	Mode
MIT	$l^{baseline}$	1853	1	237.92	203	4
	l^{CDR}	228	1	8.27	6	1
	l^{DDR}	129	1	2.14	1	1
UC3M	$l^{baseline}$	2026	1	158.11	88	57
	l^{CDR}	84	1	6.22	4	2
	l^{DDR}	769	1	41.02	27	11

Table 4.1: Summary of the main statistics related to the distribution of cell changes per day of the MIT and UC3M data sets.

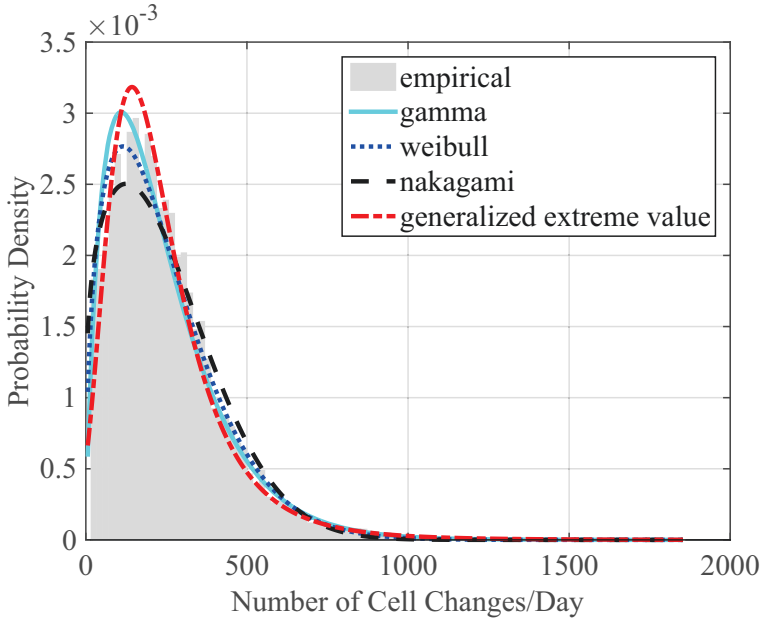
tions observed and described in the literature [113] when measuring the distances covered by people during their displacements. This fact reinforces the selection of the number of cell changes per time period as a way to translate the amount of movement into the symbolic domain. Besides, this long-tailed behavior is the reason that drives the use of the median to compare the results coming form both data sets. Finally, comparing the results for MIT and UC3M users, it can be observed that MIT users seem to move more than UC3M participants. However, this initial observation will be reconsidered later in this chapter under a different perspective that will provide a different conclusion.

4.2.2 Diversity of the Visited Places

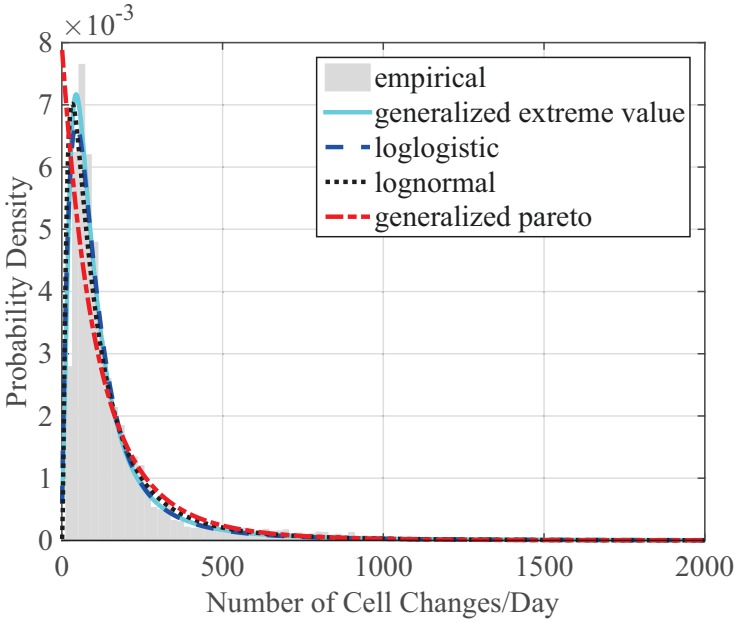
Changing the focus to the next mobility feature, Figure 4.3 shows the distribution of the number of different cells visited per day. Again, the two subfigures 4.3a and 4.3b represent the MIT and UC3M results, respectively, whereas each of the three plots of each subfigure shows the results reflected by each movement history, from top to bottom, $l^{baseline}$, l^{CDR}, and l^{DDR}. Starting again by comparing the results obtained for each data recording approach, the same differences than in the previous case can be identified, even more intensified. The distribution of values reflected by l^{CDR} is notably narrower than that drawn from the baseline and even DDR-based approaches. As commented before, people usually make calls in the same places, which according to the median value for both data sets (see Table 4.1 for the actual statistical values of the distribution), are just 2 different places per day. Again, those data raises the concern on the quality of the mobility data gathered by the CDR-based approach. Even the DDR-based approach seems to better capture the mobility-related data. The median value for the UC3M data (recall that the MIT data set has very few data traffic samples) is 15 out of the 29 for the $l^{baseline}$ case, whereas the maximum value is 267 with respect to the 532 of the $l^{baseline}$ case. The big difference in the maximum value comes from the fact that when a high number of different cells are visited per day it means that the person is making a long trip. Generally, during long trips, and even more if they made by car, data connections are fewer (hopefully people driving should not be constantly consulting the device). However, the median is not that distant, which means that l^{DDR} collects a great deal of the locations visited throughout the day.

Once again, the distribution of the data coming form $l^{baseline}$ shows a long-tailed behavior. Due to visualization purposes, as before, Figure 4.3 just shows the distribution spanned up to 100 different cells visited per day, but in Figure 4.4 the whole distribution for the baseline approach of each data set can be observed. Like in the cell changes per day metric, the different cells visited per day concentrate mainly around low values, not greater than 100, but the distributions have a long tail reaching up to 900 and 550 different cells visited per day in the MIT and UC3M data, respectively. That leads to mean values displaced to higher values. As in the previous metric, this is directly related to the human behavior described in [113] of usual short displacements mixed with occasional long trips that add up having an important weight that make them worth to be taken into account.

Regarding the comparison between data sets, whereas the cell changes per day shown a higher median value for the MIT than for the UC3M users, when considering the number of

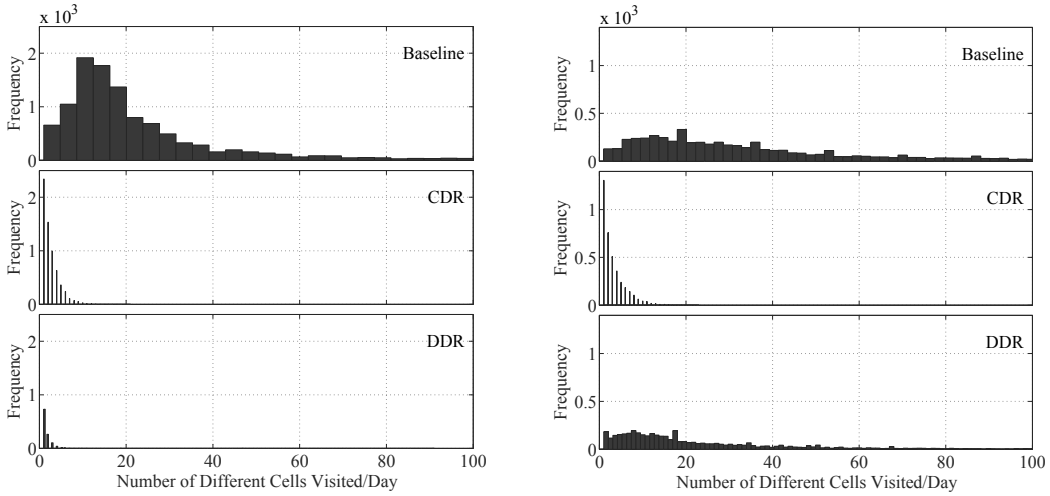


(a) MIT data set.



(b) UC3M data set.

Figure 4.2: Probability distributions best fitting the number of cell changes per day reflected in the baseline traces of the data sets considered.



(a) MIT data set.

(b) UC3M data set.

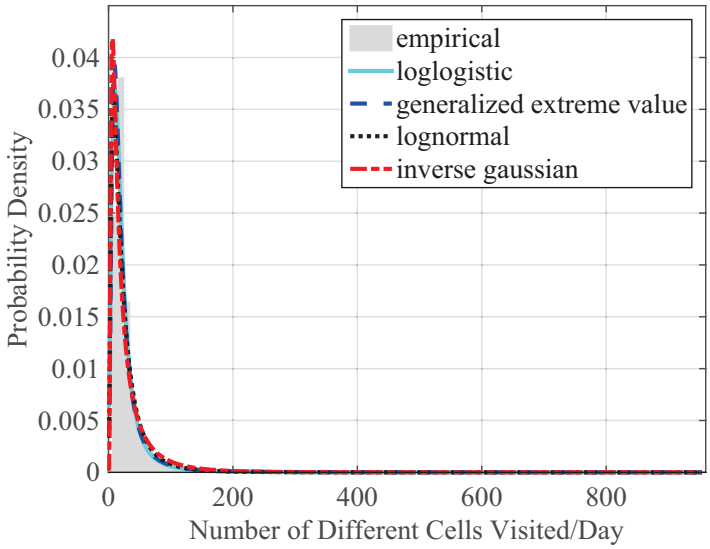
Figure 4.3: Distribution of the number of different cells visited per day reflected in the traces collected with each data collection scheme—(from top to bottom) baseline, CDR, and DDR-based—of the data sets considered.

different cells visited per day the scenario is the opposite one: UC3M users visit a median of 29 different cells per day, compared to the 17 of the MIT users. Considering that the MIT users are all students and stuff of the same university, whilst the UC3M users have different jobs and even reside in different countries, it can be normal that MIT users visit less different locations since all of them share the same closed environment.

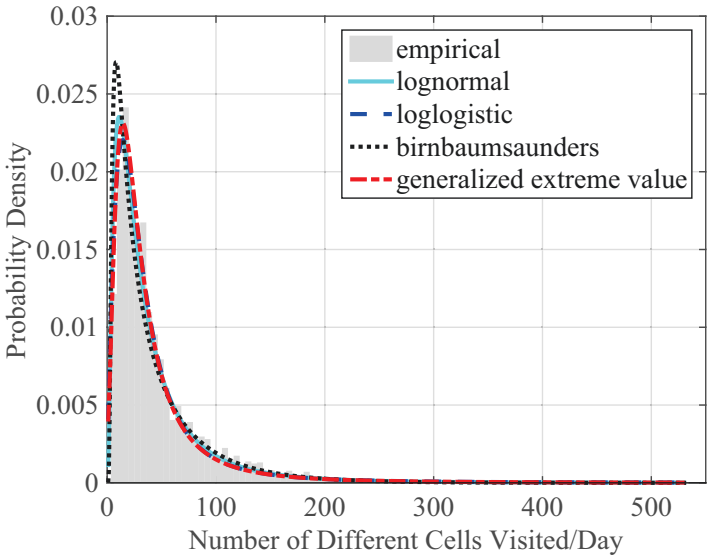
The combination the two metrics analyzed so far provides different perspectives on the users' mobility. The analysis presented is focused on general features, measured by the aggregated data of all users. Instead, Figure 4.5 shows the temporal evolution of different metrics for a user from the MIT data set (subfigure 4.5a) and another user from the UC3M data set (subfigure 4.5b). The upper subplot represents the number of cell changes per day

Data Set	l	Max	Min	Mean	Median	Mode
MIT	$l^{baseline}$	954	1	30.24	17	11
	l^{CDR}	21	1	2.63	2	1
	l^{DDR}	91	1	1.80	1	1
UC3M	$l^{baseline}$	532	1	42.85	29	12
	l^{CDR}	23	1	3.23	2	1
	l^{DDR}	267	1	23.61	15	8

Table 4.2: Summary of the main statistics related to the distribution of the number of different cells visited per day of the MIT and UC3M data sets.



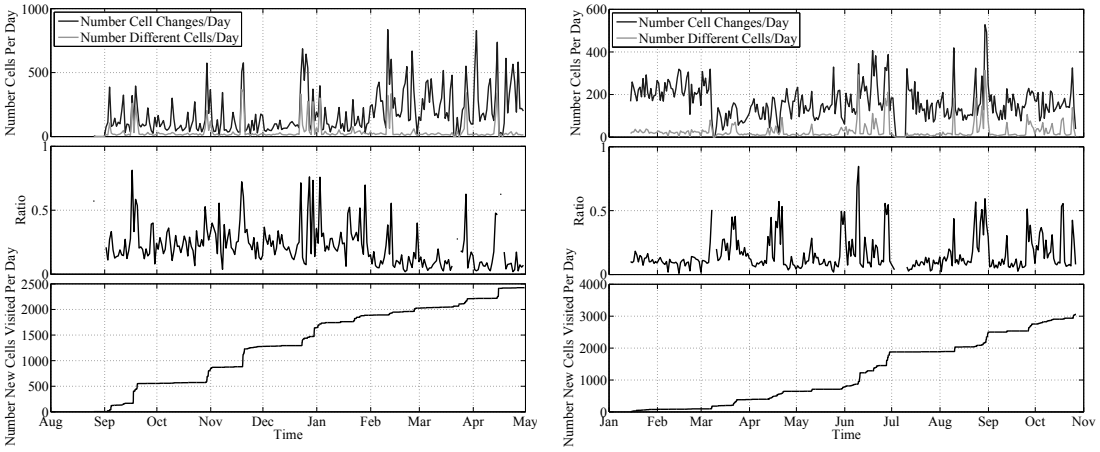
(a) MIT data set.



(b) UC3M data set.

Figure 4.4: Probability distributions best fitting the number of different cells visited per day reflected in the baseline traces of the data sets considered.

and the number of different cells per day. These data give an idea of the temporal evolution day by day, but it seems noisy. However, when dividing the number of different cells of a day by the number of cell changes in that same day (for all the days recorded in the

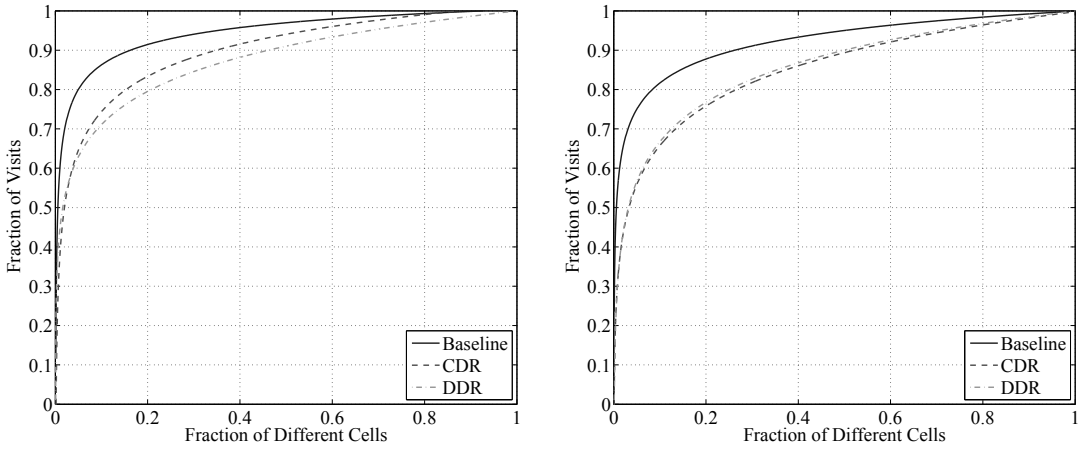


(a) MIT data set user.

(b) UC3M data set user.

Figure 4.5: Temporal evolution of three mobility features of two users (from top to bottom): Number of cell changes per day and number of different cells visited per day; ratio of different cells per cell changes, per day; and cumulative new cells visited per day.

movement history), the results leads to the middle subplot. This ratio will be high when the number of different cells visited gets closer to the number of cell changes. Looking at this subplot, clear peaks are devised along both users' movement histories. Unfortunately for the MIT user, it was not possible to asses the events behind those peaks. However, it was possible to check that for the UC3M every single peak above 0.4 corresponds to a long trip (i.e., occasional trips to further places than usual), and that no long trips made during the whole time span of the movement history were not reflected by some of these peaks. It is probable that something similar happens with the MIT user, since some of the main peaks happen around the end of December (coinciding with Christmas holidays), but it cannot be verified. Therefore, this metric that combines the two previous ones, can be very useful to determine long occasional trips that might potentially affect the prediction accuracy. Another interesting aspect is determining the first time a user visits certain place. These situations can be easily detected by measuring the number of new CellIDs never seen before visited per hour. The bottom subplot shows the temporal evolution of the cumulative CellIDs never seen before per day. The steps that can be observed in these subplots of both users correspond to some of the peaks in the middle subplot, which reinforce the hypothesis of those peaks corresponding to long occasional trips. However, there is a subtle difference between these two metrics. The rate at which CellIDs never seen before are discovered unveils moments when the user is visiting totally new places (thus, impossible to foresee and predict). However, the ratio of different cells by cell changes just uncovers potential long trips, which in many cases headed to new visited places, but in some other cases are just occasional trips to places already visited before. Checking the plots of the UC3M user with the information provided, it can be observed that the last peaks in the middle subplot do not coincide with an increase in the new cell discovery rate shown in the bottom plot, since they correspond to occasional long trips to places already



(a) MIT data set.

(b) UC3M data set.

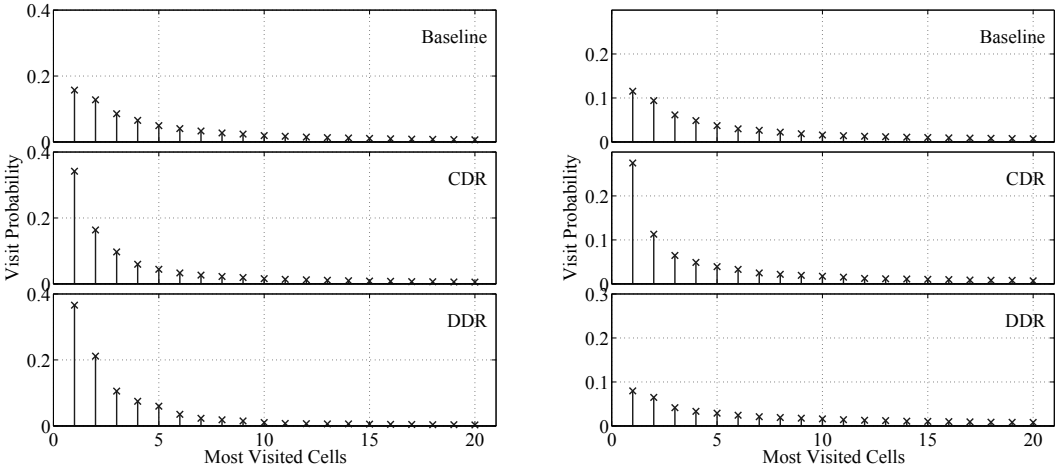
Figure 4.6: Aggregated fraction of visits as a function of the fraction of different visited cells, for the two data sets.

visited before. The same behavior is observed in the trace of the MIT user. The peaks in the middle subplot do not lead to such big steep increases in the new cell discovery rate in the lower plot, probably because they correspond to trips to already known places, although, again, this could not be verified.

4.2.3 Distribution of Visits

The next feature under study is the fraction of the total number of different cells visited by a user that concentrate different fractions of visits. Figure 4.6 shows the cumulative distribution for the aggregated population of the MIT (subfigure 4.6a) and UC3M (subfigure 4.6b) data sets. In order to calculate these curves, all the visited cells were sorted out from the most to the least visited one, for each user. Then, the fraction of cumulative visits (out of the total ones for a user) was calculated for each cell, which represents certain percentage out of the total number of different visited cells by that same user. Finally, all the data from each user were aggregated and the curve fitting tool of Matlab was used to get the parameters defining the power curves with with a 95% confidence. This same process was performed independently for the $l^{baseline}$, l^{CDR}, and l^{DDR}. The results remind of a Pareto distribution, where 20% of the different visited cells concentrate 80% of the total visits in the CDR and DDR-based approaches for both data sets, whereas for the baseline approach this 20% of different visited cells concentrate up to the 90% of the total visits. Thus, some few cells are much more visited, like home or work, whilst the majority of locations just slightly add up to the total number of visits.

Another way to visualize this characteristic is by inspecting the visit probability of the most visited cells, which is shown in Figure 4.7. Although taking a look at the cumulative visits distribution depicted in Figure 4.6 it may seem that both l^{CDR} and l^{DDR} are very



(a) MIT data set.

(b) UC3M data set.

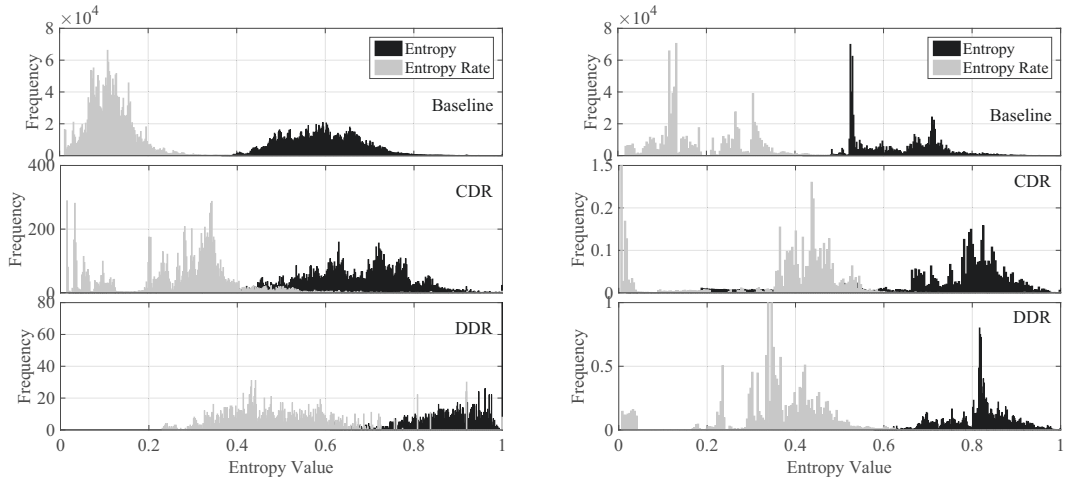
Figure 4.7: Average probability of visiting each of the 20 most visited cells, for each of two data sets.

similar with respect to the visit distribution, Figure 4.7 demonstrates that the cumulative distribution might be misleading. Both l^{CDR} and l^{DDR} shares almost the same curve for the cumulative distribution, but the visit probability to the most visited cell in the l^{CDR} approach is more than twice higher than the corresponding probability for the l^{DDR} case (recall that the DDR data in the MIT data set was scarce and thus not meaningful, so it is not analyzed here). This effect comes from the small number of events in the l^{CDR} approach and the already mentioned fact that people tend to make calls always from the same locations. However, the cumulative distribution masks this fact, showing a behavior similar to the DDR-based approach. In the l^{DDR} case, though, the two most visited locations have a slightly higher probability than the third and remaining cells, so that the probability is more equally distributed. It should be noted that these plots represent the probability of visiting the 20 most visited cells, but each movement history gathers more than 20 different cells (see Table 4.2 to check just the number of different cells visited per day, which is already higher than 20). Therefore, the CDR-based results show a much narrower probability distribution (thus, the high probability of the most visited cell), whilst both baseline and DDR-based histories enclose many more different cells, and thus the probability distribution is wider. Nonetheless, the baseline approach shows a fast decrement of the probability of visit, which drops to half already in the fourth most visited cell in both data sets, which reinforces the concern about possible biased predictions due to this noticeable difference between a small group of cells with respect to the rest.

4.2.4 Mobility Randomness

The next feature to analyze is the uncertainty about user mobility. Section 2.2 already presented the information theory concept of entropy as a way to quantify uncertainty of symbol sequences. Besides, this uncertainty can be quantified in terms of time-uncorrelated samples, through the use of the Shannon entropy, or in terms of stationary processes, thus factoring the time dimension in, which required the use of entropy rates and, in particular, of a specific entropy rate estimator known as Grassberger entropy rate. Figure 4.8 shows the aggregated entropy and entropy rate values calculated at each sample of, from top to bottom, $l^{baseline}$, l^{CDR}, and l^{DDR}, for both the MIT (subfigure 4.8a) and UC3M (subfigure 4.8b) data sets. In order to be able to compare these entropies for different users (who can display very different entropy values), all the entropy and entropy rate absolute values are normalized by the maximum entropy value corresponding to each step of the movement history considered, value obtained by calculating the Hartley entropy (refer to Section 2.2 for more details). The analysis focuses first on the comparison of the three data collection schemes. Regarding the distribution of entropy, the baseline approach leads to a distribution located at slightly lower values than in the CDR or DDR cases. This means that, without taking into account temporal dependencies, the uncertainty enclosed by $l^{baseline}$ about the next movement of the user is slightly lower than in the CDR or DDR cases. But what is really significant is the decrease of this uncertainty as soon as temporal correlations (i.e., movement patterns) are considered, that is to say when entropy rate is used to measure the uncertainty. For all the cases in both MIT and UC3M data, the distribution of entropy rate values is clearly located at much lower values than the distribution of entropy. This is an indicator of the great influence of mobility patterns in the movement histories. If locations are considered as independent events, the uncertainty about what the next location will be is much higher than when locations are considered as a sequence of interrelated events, which occur following certain order and this order is consistently observed along the whole history of the user.

These results are in consonance with the ones presented in [134]. However, in such work the data considered came from l^{CDR} of an extensive population whereas Figure 4.8 shows the results for the three data collection approaches, showing noticeable differences between the baseline and CDR collection schemes. For MIT users, the entropy rate distribution is concentrated in the range between 0 and 0.2, whereas for the CDR case is much more spread, spanning form 0 to 0.5 values with an irregular distribution. It should be also noted that MIT users were all people working or studying in the same university, which can lead to this concentrated distribution in the baseline case due to similar timetables and academic calendars. However, the UC3M users were much diverse and that is reflected into the entropy and entropy rate distributions, which show a much more irregular shape, and in the case of the entropy rate, a wider distribution than in the MIT case. The subplot representing the results drawn from the CDR-based movement histories of the UC3M data set has a very specific distribution concentrated around 0 (people that always make calls from the very same places, thus having no uncertainty on where the next call will be made), and then, larger values than the baseline case (since CDR-based movement history skips all the data between calls, it is much more difficult to record the mobility patterns responsible



(a) MIT data set.

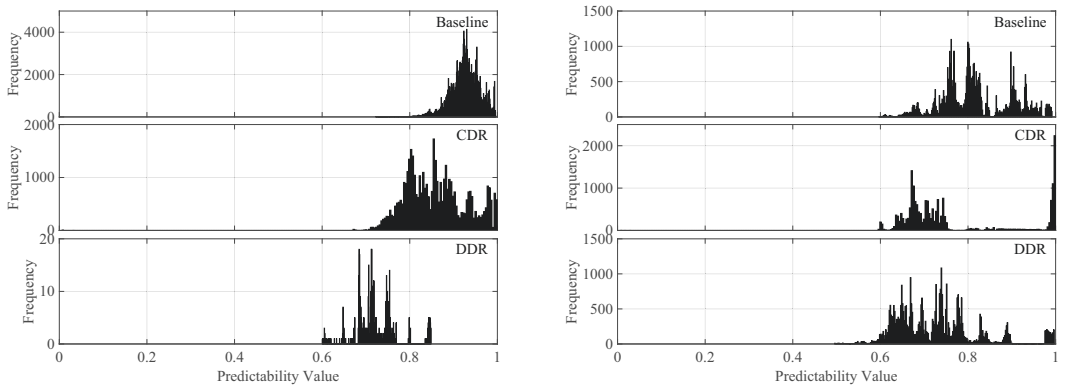
(b) UC3M data set.

Figure 4.8: Distribution of the entropy and entropy rate values at each step of the traces contained in the two data sets considered.

for the low values of entropy rate in the baseline case). The behavior of the DDR-based results might be the most surprising ones. They could be expected to behave similarly to the baseline case but, however, its distribution follows the same two focus shape than the entropy rate distribution of the CDR case. Although surprising, the explanation for it will come in next sections (no analysis on the DDR-based case of the MIT data set is described due to the small number of samples).

4.2.5 Mobility Predictability

The last feature under examination, the predictability, is tightly coupled with the entropy rate. As explained in Section 4.1, predictability measures the maximum percentage of right predictions that the best prediction algorithm would ever attain, taking into account the entropy rate and number of different cells visited by the user. Figure 4.9 shows the distribution of the values of such metric for the MIT (subfigure 4.9a) and UC3M (subfigure 4.9b) data sets and, from top to bottom, when considering $l^{baseline}$, l^{CDR}, and l^{DDR}. In this case, the predictability at every single step of each movement history could not be calculated due to the high computational cost, and thus subsequent time needed to process the whole set of available data. For this reason, 2,000 samples were randomly chosen over the entire movement history of each user. This way, all possible situations (more random ones, like holidays, and less random ones, like labor days) were aimed to be captured, thus depicting a low biased image of the general distribution of predictability values. Starting by comparing the different data collection approaches, in the MIT data set there is a more clear difference among them. The distribution in the baseline case is narrower centered around 93% (as in [134] for the CDR data). However, for CDR-based data, the distribution is wider and more weight is concentrated in lower predictability values. The results for the



(a) MIT data set.

(b) UC3M data set.

Figure 4.9: Distribution of the predictability values at some steps of the traces contained in the two data sets considered.

UC3M data set, however, are more irregular (just like the entropy rate distribution). For the baseline case, there is not a single value clearly concentrating the majority of values. Since the users follow much more different patterns, the predictability of the population varies greatly. In the CDR case, the two sides distribution noticed when analyzing the entropy rate are clearly reflected in the predictability too: one side of the distribution concentrated around high values (people making calls from the same places), and the other side of the distribution located at lower values than the baseline case, due to the lack of information about mobility patterns. The DDR case presents also a wide distribution, even wider than the baseline case, with no clear value concentrating a high frequency of occurrences, again showing the diversity of the users in this data set.

After studying all these data, it seems that the data collection approach that most faithfully reflects the mobility features of the user is the baseline one. However, it should be noted that this approach is not perfect either. It is just an indirect way to collect data representing mobility based on the cellular telephony network. Thus, it suffers from side effects of using such network. Next section focuses on these side effects and proposes different ways to try to diminish their impact on the observed mobility features.

4.3 Impact of Filtering Mobility-Unrelated Data on the Analysis of Human Mobility Features

Using an indirect data source to collect mobility histories has some drawbacks. Recalling Section 3.1, a movement history could be collected from the user's device or from the network. In the first case, every single cell change detected by the mobile phone is recorded into the movement history, $l^{baseline}$, whilst in the second case, only the cells the phone is attached to when making or receiving a call, or during data transfers, are reflected into the movement history, l^{CDR} or l^{DDR}, respectively. But in the first case, is every cell change a

direct reflection of the user mobility? The reality is that there are cell changes due to the user mobility, but there are also many other cell changes due to diverse network-related causes that do not reflect any aspect of the user mobility: received signal degradation and consequent change of BTS to another one from which the received signal has better quality, load balance, etc. These cell changes can be observed in the movement histories in the form of long sequences of repeated changes between two or three cells mainly, as for example, $l^{baseline} = ...fgababababcbabacabcabcba....$, from where it takes the name of ping pong or oscillation effect. These sequences should be filtered out so that they do not interfere or bias the mobility features reflected or the behavior of the applications based on the movement histories to perform their services. This section proposes a two-steps mechanism to detect and filter out these network-related events, that will be further evaluated based on the mobility features previously studied.

4.3.1 Ping Pong Sequence Detection and Filtering Proposals

As explained in Chapter 3, the reference scenario of human mobility corresponds to a GSM network. Attending to a typical hive network cell distribution, the coverage shape of each BTS is not perfectly hexagonal, but it suffers from time-variant coverage area radius. The two most common cases where the ping pong effect takes place are when user is in the intersection of two or three cells [79]. These two ping pong cases are represented by the dotted and dashed circles in Figure 4.10.

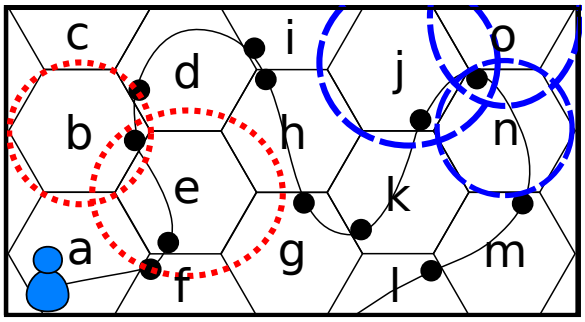


Figure 4.10: Coverage areas provoking ping pong sequences.

If the user is in the intersection of two cells, as in the case when the user is inside the dotted circles in Figure 4.10, her device might be oscillating between cells b and e, even if the user is not moving. This situation leads to movement sequences of the form:

$$l^{baseline} = \ldots l_n l_{n+1} l_n l_{n+1} l_n l_{n+1} l_n l_{n+1} \ldots$$

For example, the user inside both dotted circles could record a trace like $l^{baseline} = \ldots bebebe \ldots$. Therefore, a ping pong effect between two cells is detected when finding a sequence of events where

$$l_n^{baseline} = l_{n-2}^{baseline}$$

until a different symbol from $l_n^{baseline}$ and $l_{n-1}^{baseline}$ appears in $l^{baseline}$. However, this may result in a relaxed requirement that leads to filtering real locations. For instance, if the user records a trace like $l^{baseline} = aebea$, it can mean that the user went from a to b and back, but it can also mean that the user went from a to e and back, but a ping pong event occurred in the meantime. For this reason, a more flexible definition of the detection of 2-cell ping pong effect is proposed. The 2-cells ping pong sequence will be detected if only two different CellIDs are present in, at least, p consecutive symbols of the sequence:

$$|\dot{l}_{i,i+p-1}^{baseline}| = 2$$

where $l_{i,i+p-1}^{baseline}$ is the sequence from step i to step $i+p-1$, and the operator $|\dot{|}$ represents the number of different locations found in the considered sequence.

In the case in which the user is in the intersection of three cells, as shown in Figure 4.10 for cells j, n and o, the oscillation behavior produces fake movement sequences like:

$$l^{baseline} = \ldots l_n l_{n+1} l_{n+2} l_n l_{n+2} l_n l_{n+1} \ldots$$

For example, the user inside both dotted circles could record a trace like $l^{baseline} = \ldots jnojojnojn \ldots$. Therefore, a ping pong sequence between three cells will be detected when encountering a sequence of events where the following three conditions applies:

$$l_n^{baseline} = l_{n+3}^{baseline}, l_n^{baseline} \neq l_{n+1}^{baseline}, l_{n+1}^{baseline} \neq l_{n+2}^{baseline}$$

until a different location from to any of $l_n^{baseline}$, $l_{n+1}^{baseline}$, and $l_{n+2}^{baseline}$ appears in $l^{baseline}$. Notice that the three locations must be neighbors for the ping pong sequence to happen. For instance, the sequence $l^{baseline} = \ldots l_n l_{n+1} l_n l_{n+3} \ldots (l_{n+1} \neq l_{n+3})$, is not considered a ping pong sequence, unless there is some previous information indicating that l_{n+1} and l_{n+3} are neighbors (i.e., there exist previous transitions between them two). Otherwise, not all possible transitions are present in the potential oscillating sequence (the transitions $l_n l_{n+1}$ and $l_n l_{n+3}$ are possible, but $l_{n+1} l_{n+3}$ is not). As in the case of 2-cell oscillations, determining the occurrence of a ping pong effect among three cells as soon as the previous condition is met might result in filtering out data that is actually not a ping pong sequence. For this reason, a more configurable 3-cell ping pong effect detection scheme is proposed, in which the ping pong sequence is detected if only three different neighboring cells are present in, at least, q consecutive symbols of the sequence:

$$|\dot{l}_{i,i+q-1}^{baseline}| = 3$$

The complete ping pong detection scheme combines these two mechanisms with the appropriate parameters, (p, q), to determine all the ping pong sequences in a trace, so they can be filtered out.

In order to have a preliminary idea on the impact of different values of p and q on the ping pong detection process, the movement history of a user, $l^{baseline}$, is processed to detect ping pong sequences using different detection schemes, (p, q). Table 4.3 shows in the second column the percentage of events which were detected as ping pong sequences. With the most restrictive scheme, $(p, q) = (3, 4)$, 86.98% of the CellIDs in $l^{baseline}$ are

(p,q)	Total Ping Pong Events [%]	Ping pong Events during week [%]	Ping pong events matching movement [%]	Events during Movement Matching ping pong [%]	Events during no movement Matching ping pong [%]
(3,4)	86.98	89.71	8.64	86.95	89.98
(4,6)	68.28	74.77	7.82	65.61	75.67
(4,8)	60.41	67.07	7.78	58.50	67.91
(6,8)	53.09	61.45	7.28	50.20	62.55
(8,10)	43.42	53.17	7.99	47.63	53.71

Table 4.3: Summary of the fraction of ping pong events detected in the trace of a user in the UC3M data set during specific week, for different detection schemes, in three situations: in the whole week, matching movement periods, and matching no-movement periods.

considered to belong to mobility unrelated events, which represents an enormous percentage of all the available data recorded in the movement history. By relaxing the scheme to be $(p, q) = (4, 6)$ (i.e., as soon as the ping pong is detected in double number of the ping pong cells, 4 for 2-cells and 6 for 3-cells ping pong), the percentage drops down to 68.28%, which represents more than half of the cells in the history, but it is 20% lower than in the previous setup. From there, by increasing p and q, the percentage keeps on decreasing but in a more gradual way. Increasing both parameters in 2 units, the percentage drops a 15% more, and finally when nearly doubling p and q, the percentage of detected ping pong events decreases around 25%. Therefore, the most dramatic decrease occurs when relaxing p in 1 unit and q in 2 units with respect to the most restrictive scheme.

Still, it is difficult to be able to ascertain what really is a ping pong sequence and what is just a real movement of the user. Thus, a different approach to its determination was carried out. The starting hypothesis is that ping pong sequences happen mainly when the user is not moving, whilst the device is switching from one cell to another one due the user movement, and thus those cell changes are not ping pong sequences but just the result of the location change. To check this hypothesis, the same user, which $l^{baseline}$ was analyzed with different (p, q) schemes, provided data about the specific time periods she spent moving and stationary for a whole week. These data were compared against the ping pong detection results in the following way. Each CellID in $l^{baseline}$ is accompanied by its corresponding timestamp and a flag indicating if it belongs (true) or not (false) to a ping pong sequence (one flag per each different detection scheme was provided). Then, for each CellID recorded in $l^{baseline}$ during the week under study, it is checked if it corresponds to a period of movement or a stationary period. Under the initial assumption, ping pong events should not occur during movement periods, thus all ping pong events should happen during stationary periods. The third column of Table 4.3 shows the percentage out of the total number of ping pong events that occur during a movement period. The values for

Data Set	(p,q)	Max [%]	Min [%]	Mean [%]	Median [%]
MIT	(3,4)	99.63	85.16	93.58	93.71
	(4,6)	98.59	71.18	84.92	84.44
UC3M	(3,4)	91.45	48.89	75.25	78.00
	(4,6)	82.17	25.13	52.73	53.08

Table 4.4: Summary of the statistics of ping pong events in the whole set of MIT and UC3M traces for two different detection schemes: (3,4) and (4,6).

all detection schemes are very similar, around 8%. This may lead to think that, actually, ping pong events happen mainly during stationary periods. However, a deeper look must be taken to assert where this low value comes from. The fourth column of the table shows the percentage of the total number of CellIDs recorded during the movement periods that coincide with ping pong events. The last column shows this same metric applied to the case of stationary periods. The reason why the percentage of ping pong events matching movement periods is so low is that the number of CellIDs recorded during movement periods is much lower. This means that most of the recorded CellIDs correspond to just ping pong effects instead of movement. This fact raises an important concern when making predictions, since it should be carefully considered which predictions are really useful, that is to say which predictions are actually predicting the next movement of the user rather than the next event, which most probably will be related to network issues. The results of this preliminary analysis seem to indicate that the hypothesis set out to determine what really is a ping pong sequence, and thus avoid further filtering out no ping pong events is not valid. Thus, two schemes will be chosen to perform the next analysis, based on the notion of the most restrictive detection scheme, and the one that by slightly relaxing the most constrained parameters, notably reduces the percentage of ping pong detections: $(p, q) = (3, 4)$ and $(p, q) = (4, 6)$, respectively.

Since there are ping pong events both during stationary and moving stages, the detection cannot be made following this criteria. Therefore, the aim would be to detect as many ping pong sequences as possible. Thus, the two first schemes, namely $(3, 4)$ and $(4, 6)$ are selected for further analysis. These two schemes were applied to the data sets considered, obtaining the statistics about the number of ping pong events detected in each case shown in Table 4.4. The statistics show that the MIT traces are made of more ping pong events than the UC3M data: whereas for the MIT data the percentage of ping pong events in average is 93.58% for the $(3, 4)$ detection scheme, and 84.92% for the $(4, 6)$ scheme, these values just reach 75.25% and 52.73% in the UC3M data set, respectively. In the comparison among the different schemes, UC3M still shows the 20% drop from the most constrained to the more relaxed detection scheme, whilst in the MIT case the decrease is only around 10%, probably due to the huge percentage of ping pong events collected.

The next question is how to perform the filtering itself. The three following methods are proposed to that purpose:

- **Representative technique.** Replace the whole ping pong sequence by one of the

symbols of the sequence, the *representative_symbol*. Among the symbols of the sequence, the one that concentrates a higher number of visits is chosen. The reason behind this choice is that, if there is no ping pong effect, the most probable symbol to be observed at that location is the one with the highest probability of being visited, i.e., the one with a highest number of visits along the location history. This way, the real visit distribution is tried to be preserved.

The problem of this approach is that it does not take into account the adjacency of cells. For example, if the individual in Figure 4.10 has a history location $l = \ldots kjnojonjnojonm \ldots$, and the most visited location is o, the filtered sequence would be $l^{representative} = kom$, but there is no real transition between ko or om, since those pairs of cells are not adjacent.

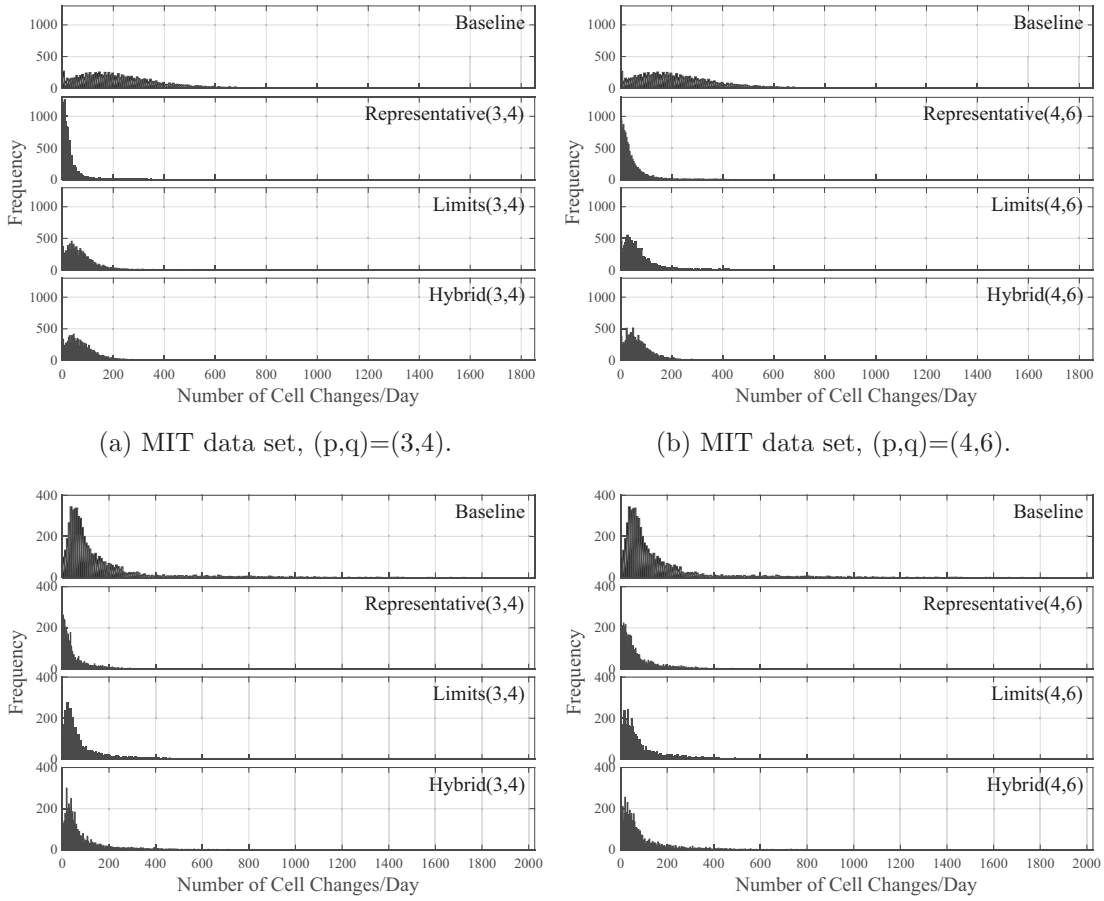
- **Limits technique.** Replace the whole ping pong sequence by $entry\_symbol \rightarrow exit\_symbol$, where *entry_symbol* and *exit_symbol* are the first and last symbols, i.e., the limits of the ping pong sequence. In previous the example, the filtered sequence would be $l^{limits} = kjnm$, where every pair of consecutive cells are adjacent. This case solves the problem of the representative technique.

  However, since ping pong sequences appear very frequently, both delimiting symbols, *entry_symbol* and *exit_symbol*, accumulate a large number of visits, whilst the most visited location of the sequence might be ignored. This may change the real probability of visiting each location, and lead to a situation where the symbols limiting the ping pong sequences have more visits than the originally most visited locations.

- **Hybrid technique.** Replace the whole ping pong sequence by $entry\_symbol \rightarrow representative\_symbol \rightarrow exit\_symbol$. If *entry_symbol* or *exit_symbol* is equal to *representative_symbol*, this last one is neglected. In the example above, the filtered sequence would be $l^{hybrid} = kjonm$, which merges the advantage of the previous techniques.

4.3.2 Analysis of the Mobility Features Reflected in the Filtered Traces

Next, the filtering techniques previously proposed are evaluated by applying them to the data sets explained in Chapter 3, to see how the mobility features extracted from the resulting movement histories are impacted. The analysis of the mobility features reflected in the filtered traces starts again with the comparison of the number of CellIDs recorded in $l^{baseline}$ per day with respect to the number of CellIDs per day remaining in the movement histories after different filtering techniques and detection schemes are applied to $l^{baseline}$. Figure 4.11 shows the distribution of cell changes per day for the MIT data (subfigures 4.11a and 4.11b) and for the UC3M data (subfigures 4.11c and 4.11d), for the two selected detection schemes, namely $(p, q) = (3, 4)$ and $(p, q) = (4, 6)$, respectively, and considering, from top to bottom, the baseline movement history as a reference and then the three filtering techniques: representative, limits, and hybrid. Focusing on the filtering techniques first, the figure shows that the representative technique is the one deleting more mobility-unrelated events, since the final distribution of CellIDs recorded per day is the narrowest



(a) MIT data set, (p,q)=(3,4).

(b) MIT data set, (p,q)=(4,6).

(c) UC3M data set, (p,q)=(3,4).

(d) UC3M data set, (p,q)=(4,6).

Figure 4.11: Distribution of the number of cell changes per day reflected in the baseline trace and the traces filtered using two detection schemes and the three filtering techniques, for the two data sets considered.

and placed at the lowest values. The results for the limits and hybrid techniques are very similar, reducing also the values around which the resulting distribution is centered around, although not as much as the representative technique. This behavior holds for both data sets and detection schemes.

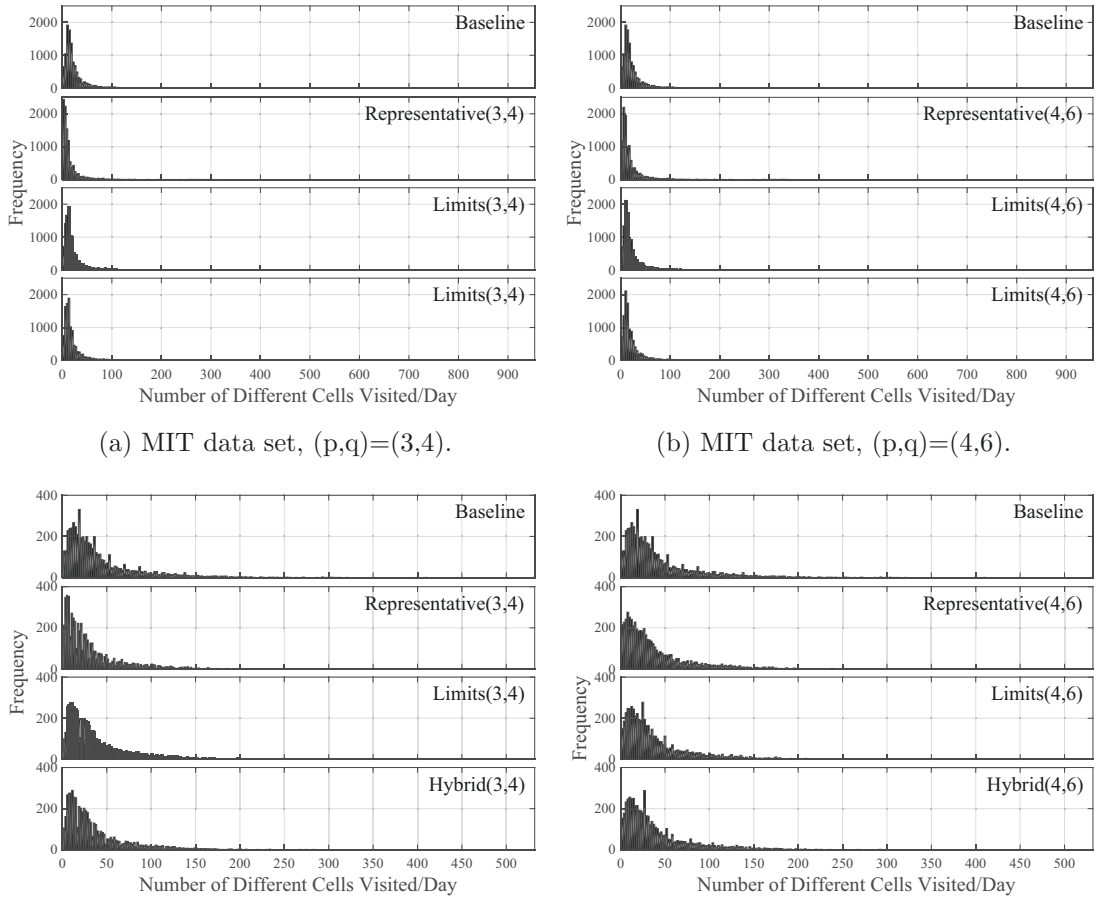
Taking a look at the actual statistical values of the distributions, displayed in Table 4.5, there are some interesting details to notice. Recalling the comparative between the median value drawn from the UC3M and MIT data sets, users from the MIT set seemed to move much more (median of 203 cell changes per day) than the UC3M users (median of 28 cell changes per day). However, after filtering the traces applying the representative technique (i.e., replacing each ping pong sequence by just one CellID, which is the most simplified version of the trace), the resulting median values are just the opposite: UC3M

Data Set	(p,q)	Filter	Max	Min	Mean	Median	Mode
MIT	Baseline		1853	1	237.92	203	4
	(3,4)	Representative	846	1	32.71	18	6
		Limits	940	1	79.45	61	36
		Hybrid	944	1	84.85	66	2
	(4,6)	Representative	1009	1	49.08	29	2
		Limits	1039	1	73.64	54	38
		Hybrid	1046	1	82.32	62	2
UC3M	Baseline		2026	1	158.11	88	57
	(3,4)	Representative	516	1	49.60	29	4
		Limits	825	1	85.05	49	30
		Hybrid	796	1	86.34	50	27
	(4,6)	Representative	679	1	76.38	45	13
		Limits	803	1	89.67	52	20
		Hybrid	841	1	94.01	55	20

Table 4.5: Summary of the main statistics related to the distribution of the number of cell changes per day reflected in the baseline trace and the traces filtered using two detection schemes and the three filtering techniques, for the two data sets considered.

users move almost double (median values of 29 and 45 cell changes per day for the $(3, 4)$ and $(4, 6)$ detection schemes, respectively) than MIT users (median values of just 18 and 29, respectively). However, since the limits and hybrid techniques add more symbols per ping pong sequence detected, and the MIT data set has such a high amount of these sequences, then the median values greatly decrease (from 203 to 61 and 54 for the limits technique and two detection schemes, and to 66 and 62 for the hybrid technique and two detection schemes), but they are still greater than the values for the UC3M data set (29 and 52 for the limits technique, and 50 and 55 for the hybrid technique, and both detection schemes). Once again, this fact highlights the impact of such high number of ping pong events in the movement histories. Another interesting observation is that regardless of the decrease in all the statistics, the long-tailed behavior still remains, as can be checked by the noticeable difference between the mean and median values, and the long tail of the distributions depicted in Figure 4.1, which reinforces the idea of usual short trips with eventual long travels (i.e., the occasional high number of cell changes per day is not due to just ping pong effects, but when these are filtered out, this behavior remains).

Regarding the number of different cells visited per day before and after filtering, Figure 4.12 shows the distribution of the metric for the same cases considered before. In this metric, the differences between the distribution before and after filtering are not very significative. It can be observed a really slight displacement of the distributions to lower values, whilst the shape and width remain practically unchanged. Comparing both data sets, the same properties observed before are present now: UC3M users have a more wide



(a) MIT data set, (p,q)=(3,4).

(b) MIT data set, (p,q)=(4,6).

(c) UC3M data set, (p,q)=(3,4).

(d) UC3M data set, (p,q)=(4,6).

Figure 4.12: Distribution of the number of different cells visited per day reflected in the baseline trace and the traces filtered using two detection schemes and the three filtering techniques, for the two data sets considered.

distribution, before and after filtering, than MIT users, which now seems to be coherent with the number of cell changes after filtering. Looking at the actual statistical values, exposed in Table 4.6, it can be checked that, indeed, the median values for UC3M users for all detection schemes and filtering techniques are higher (almost double) than for MIT users. With respect to the differences before and after the filtering phase, the reduction of the statistical values is much smoother than in the case of the cell changes per day. For the MIT users, the median drops to half of the value in the baseline case when using the representative technique, and the most restrictive detection scheme, whereas for the rest of cases, it just decreases up to 4 different cells less. In the case of UC3M data, the reduction attains up to 8 different cells less in the representative technique and $(3, 4)$ detection scheme, and for the rest up to 4 different cells less. These results seem to indicate that

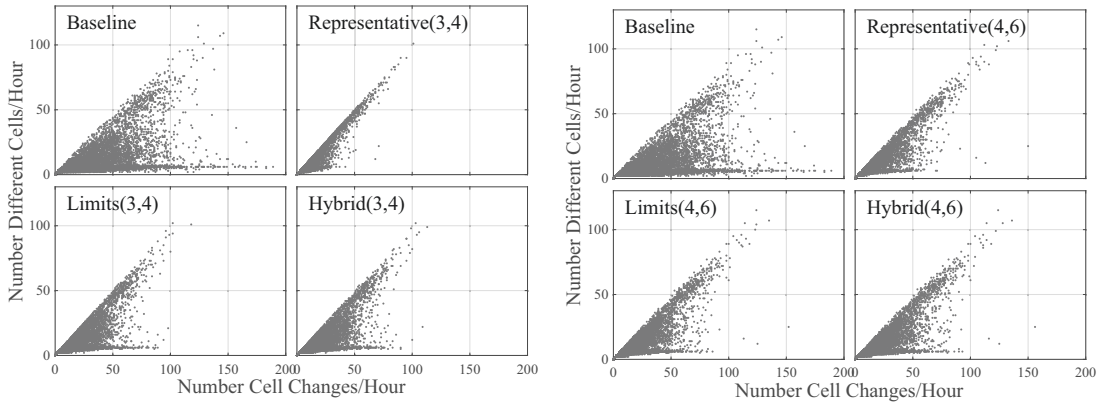
the filtering phase respects the diversity of the users movement, and just filters out events that do not provide any mobility-related useful information.

Data Set	(p,q)	Filter	Max	Min	Mean	Median	Mode
MIT	Baseline		954	1	30.24	17	11
	(3,4)	Representative	832	1	20.76	9	5
		Limits	882	1	27.42	15	9
		Hybrid	878	1	26.67	14	9
	(4,6)	Representative	920	1	26.07	13	8
		Limits	947	1	28.83	15	10
		Hybrid	948	1	28.83	15	9
UC3M	Baseline		532	1	42.85	29	12
	(3,4)	Representative	442	1	32.48	21	8
		Limits	476	1	39.27	26	10
		Hybrid	472	1	38.37	25	12
	(4,6)	Representative	510	1	40.05	26	9
		Limits	522	1	41.81	28	14
		Hybrid	521	1	41.79	28	14

Table 4.6: Summary of the main statistics related to the distribution of the number of different cells visited per day reflected in the baseline trace and the traces filtered using two detection schemes and the three filtering techniques, for the two data sets considered.

In order to provide a complementary perspective on the effect of the different filtering techniques over the two analyzed features, Figure 4.13 shows the number of cell changes per hour versus the number of different cells visited per hour for the baseline case and the three filtering techniques, for the $(3, 4)$ (subfigure 4.13a) and $(4, 6)$ (subfigure 4.13b) detection schemes. The diagonal of these plots represent hours in which all the CellIDs recorded in the movement history are not repeated, so there is continuous movement among different places. Ping pong sequences are characterized then for low number of different cells with respect to the number of cell changes experienced, that is to say values far away from the diagonal. As can be observed, the representative technique is the one erasing or displacing these values far from the diagonal towards it, even more significantly when combined with the $(3, 4)$ detection scheme. The limits and hybrid techniques clean also the zone far away from the diagonal, very similarly both of them, but not as much as the representative technique.

Recalling Figure 4.5 from the initial comparison between baseline and CDR and DDR-based approaches, two potentially interesting metrics regarding further predictions were the different visited cells over cell changes per day ratio, and the rate at which new cells never seen before are discovered per day. What happens with these two metrics after filtering? Figure 4.14 answers this question. Before filtering, the number of cell changes per day is much higher and greatly contrasts with the real number of different visited cells.



(a) (p,q)=(3,4). (b) (p,q)=(4,6).

Figure 4.13: Comparison of the number of cell changes per hour versus the number of different cells visited during the corresponding hour, for one user of the UC3M data set, considering the baseline case and the three different filtering techniques, for two detection schemes.

Therefore, the ratio of these two features instantly highlights the temporal moments in which the number of cell changes gets closer to the number of different cells (there are more changes due to actual movement). After filtering, these moments become much less clear, and the resulting ratio seems quite noisy, as can be observed in the top subplot. Thus, the baseline movement history results more useful for detecting these high mobility moments. Regarding the new cells discovery rate, as previously shown in Figure 4.12 and Table 4.6, the different visited cells metric does not significantly changes after filtering. Therefore, it could be expected that the new cell rate discovery rate remains practically unchanged, as the lower subplot of Figure 4.14 shows. Thus, any of the filtered movement histories could also be used to detect moments in which long trips or visits to unknown locations so far are taking place.

When focusing on the fraction of visits concentrated by different percentages of distinct cells, Figure 4.15 shows tendencies similar to the ones drawn from the original movement histories. Even after filtering out ping pong sequences, which add many virtual visits to certain locations, the 80-20 relation is still present. When using the hybrid or limits filters, the result is 20% of the different visited cells concentrating close to 80% of the visits, both for MIT and UC3M data sets. The representative technique, on the other hand, leads to a slightly different result, depending on the data set. Due to the enormous number of ping pong events in the MIT data set, after filtering with the representative technique, 20% of the different visited cells concentrate around 70% of the visits, and 80% of the visits are attained with the $(3, 4)$ detection scheme, whilst that percentage of different cells climbs up to 25% for the $(4, 6)$ detection scheme. Thus, very few cells concentrating a significative percentage of visits is an inherent property of human mobility, that is only intensified by ping pong, which make the case sharper for the baseline movement history (20% of cells

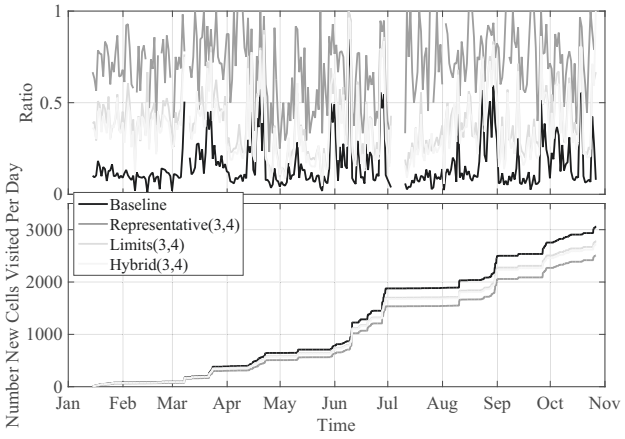
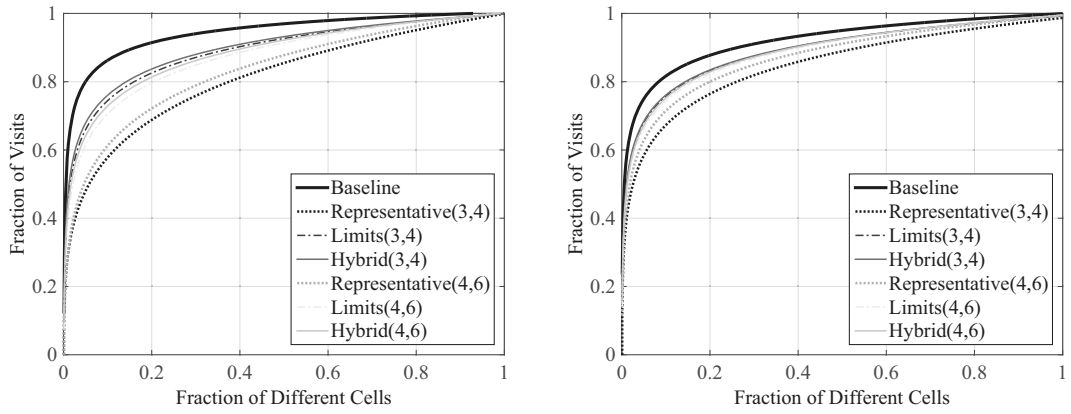


Figure 4.14: Temporal evolution of two mobility features of a user from the UC3M data set (from top to bottom): Ratio of different cells per cell changes per day; Cumulative rate of new cells visited per day.



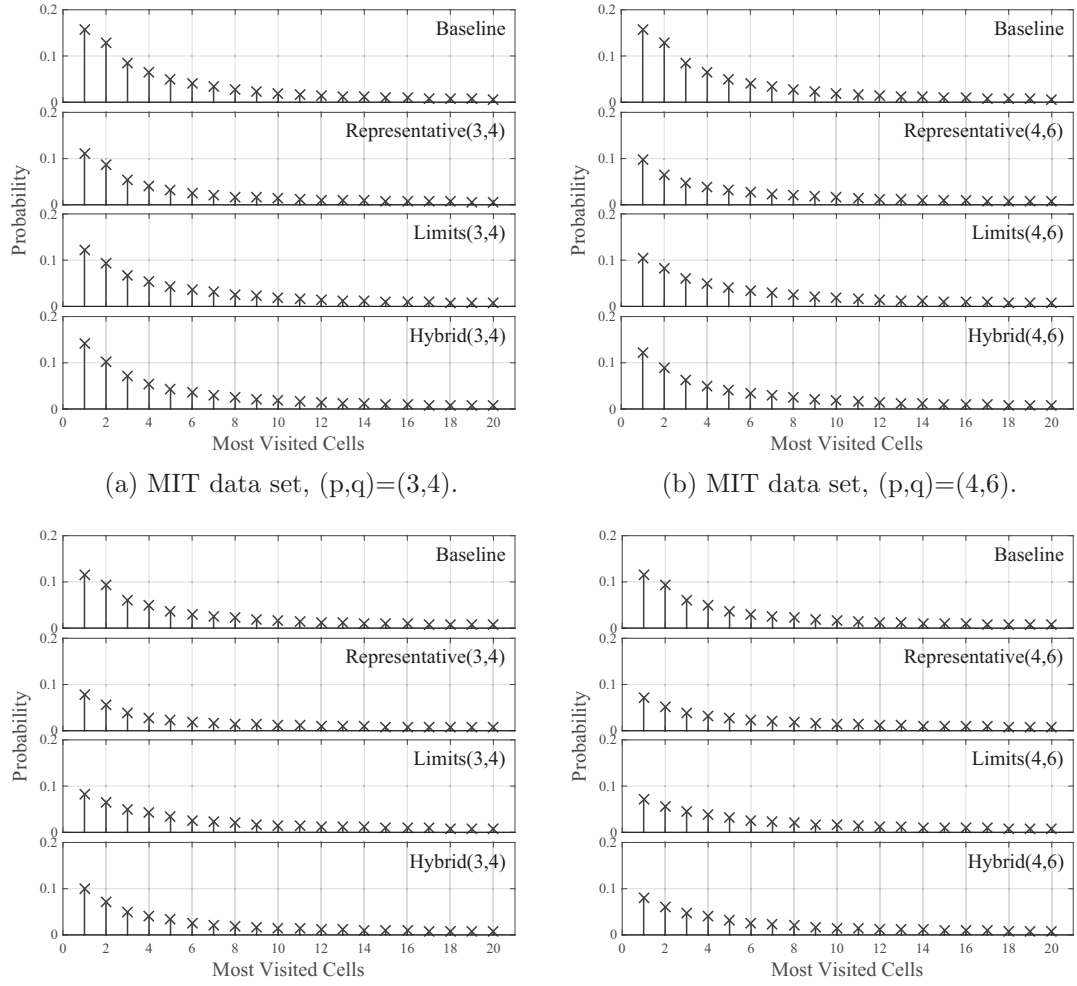
(a) MIT data set.

(b) UC3M data set.

Figure 4.15: Aggregated fraction of visits as a function of the fraction of different visited cells, for the two data sets reflected in the baseline trace and the traces filtered using two detection schemes and the three filtering techniques, for the two data sets considered.

concentrate around 90% of visits for both data sets).

Again, the visit probability of the 20 most visited cells represented in Figure 4.16 serves to reinforce these observations. But this figure also unveils how this difference in the cumulative distributions shown in Figure 4.15 is actually translated into individual cells. As can be seen, the high probability of the most visited cells is clearly diminished in the two detection schemes and for both data sets, which leads to think that a great deal of the ping pong sequences happened in these most visited cells. Still, the 4 most



(a) MIT data set, (p,q)=(3,4).

(b) MIT data set, (p,q)=(4,6).

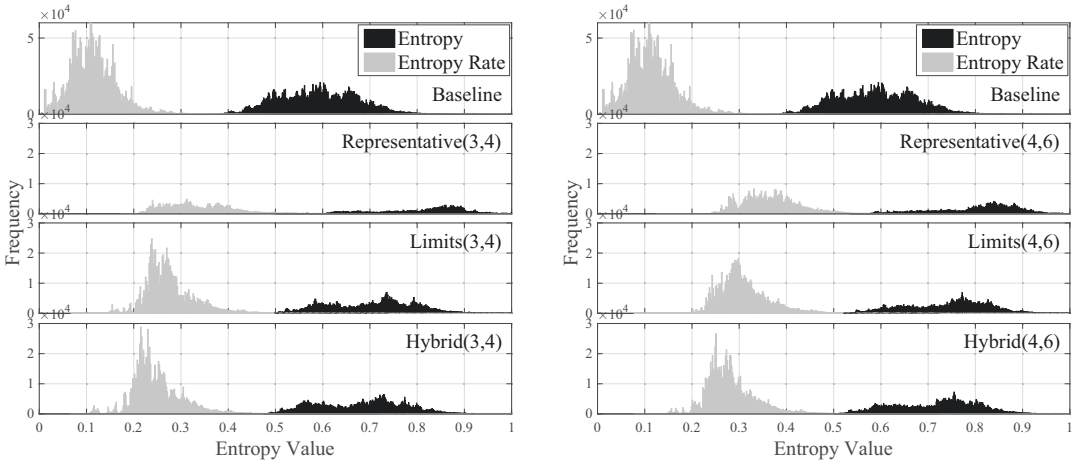
(c) UC3M data set, (p,q)=(3,4).

(d) UC3M data set, (p,q)=(4,6).

Figure 4.16: Average probability of visiting the 20 most visited cells when considering the baseline trace and the traces filtered using two detection schemes and the three filtering techniques, for the two data sets considered.

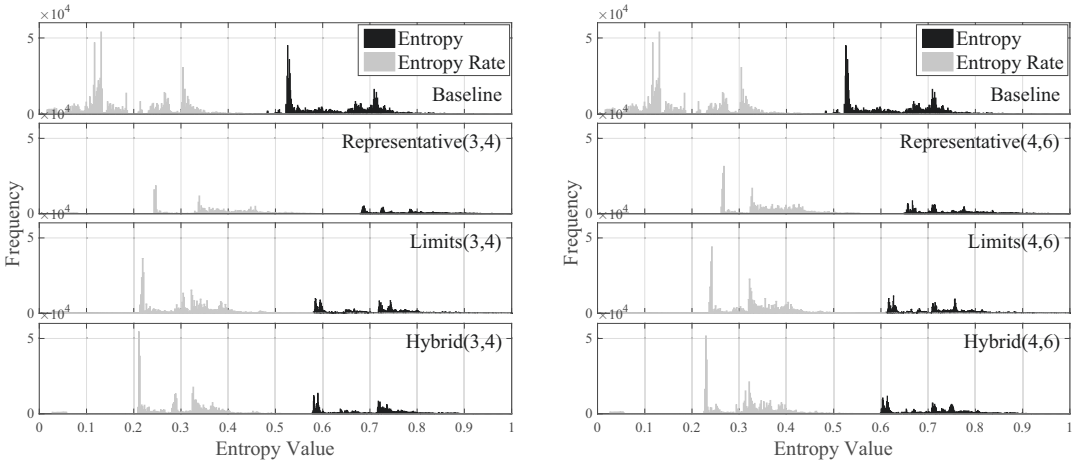
visited cells are dominants in terms of visit probability. Another interesting observation, regarding the comparison among the different filtering techniques is that both the limits and hybrid ones lead to a lower decrement in the probability of the 4 most visited cells. Recall that these two techniques respected the basic structure of the ping pong sequence by keeping the cells at the limits of the sequence, which correspond to added cells recorded in the movement history with respect to the representative case. These added cells affect mainly the visit probability of the most visited cells, thus reinforcing the hypothesis of the ping pong sequences mainly being among these most visited cells.

The next feature to analyze is the uncertainty of next movements before and after the



(a) MIT data set, (p,q)=(3,4).

(b) MIT data set, (p,q)=(4,6).



(c) UC3M data set, (p,q)=(3,4).

(d) UC3M data set, (p,q)=(4,6).

Figure 4.17: Distribution of the entropy and entropy rate values at each step of the baseline trace and traces filtered using two detection schemes and the three filtering techniques, for the two data sets considered.

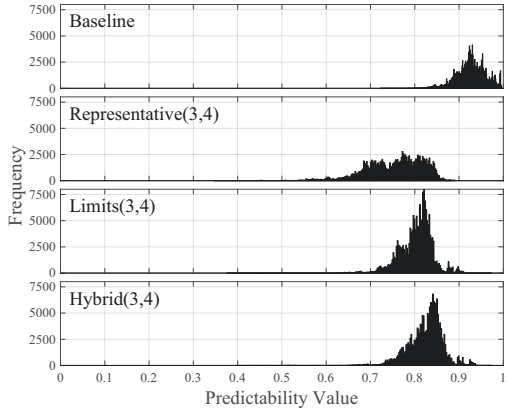
filtering phase. The distributions of the corresponding entropy and entropy rate values are represented in Figure 4.17. The most significative effect of filtering is a clear increase of the uncertainty, that can be observed by the shift of both entropy and entropy rate distributions to higher values. This effect is understandable due to the certainty introduced by ping pong sequences: they are sequences easy to identify and also easy to determine which will be the next CellID when they are happening. Thus, the entropy values decrease, even more taking into account the high percentage of ping pong events. But again, this may lead to deception when the goal is to correctly predict the next CellID to appear that represents movement. Thus, it can be roughly said that the entropy values of the filtered

traces represent the uncertainty of the real movement of the user. Another interesting observation about the MIT data specifically, is how the entropy and entropy rate distributions flatten leading to much more uniform distributions than the resulting distribution of the baseline case. That corresponds to users behaving differently among them, and along their own movement histories. Ping pong sequences might also mislead this observation, since those sequences have the very same repetitive behavior for all users and represent a great percentage of the whole movement history. The uniformity obtained by the representative filtering technique disappears when using the limits or hybrid filters. As pointed out before, these two techniques respect the ping pong sequence structure and just reduce its length to the minimum. However, the basic structure is always the same, which leads to small versions of ping pong sequences, and thus slightly lower values of entropy and less uniform entropy rate distributions. The results drawn form the UC3M data are different, due mainly to the original distributions of entropy and entropy rate for the baseline movement history. Since the original distributions are much spread and irregular due to the very different users participating in this data set, the distributions drawn after filtering are also quite spread, although they present also the shift to higher values than in the baseline case. With these results, it becomes critical to look for useful correct predictions about real movements, since the effect of ping pong sequences can easily lead to deception.

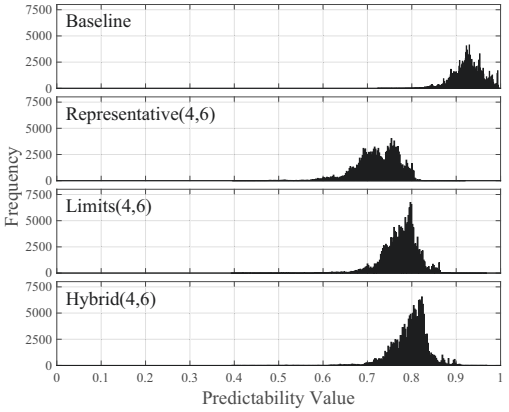
Finally, the comparison among of the distributions of the predictability of the baseline and filtered versions of the traces is presented in Figure 4.18. The differences among the filtered and original are in consonance with the entropy rate results. The distribution of predictability of the filtered traces shifts to lower values, meaning a decrease of the maximum fraction of correct predictions that can be estimated. As explained before, the misleading high predictability of the baseline traces was supported by the striking effect of the ping pong sequences. But once they are filtered out, the predictability does not exceed values of 0.9, contrasting with the 93% value where the maximum of the distribution was centered at in the case of MIT traces. Besides, the representative filtering technique provides a flattening effect over the predictability distribution, widen also the shape. Both the limits and hybrid techniques narrow again the distribution, peaking around 0.8 and 0.85, respectively. Regarding the UC3M data set, the filtering process narrow the distribution, specially when using the $(4, 6)$ detection scheme. But in any case, the most significant effect is the shift to lower values that will be directly translated into the predictions results, as will be shown in the next chapter. However, it should be noted that this does not mean a poorer performance of the prediction algorithms, it just means that predicting the real movement of the user is more difficult because the real movements are much more random than when external effects, like ping pong sequences, which does not incorporate any useful mobility data are included in the traces.

4.4 Conclusions

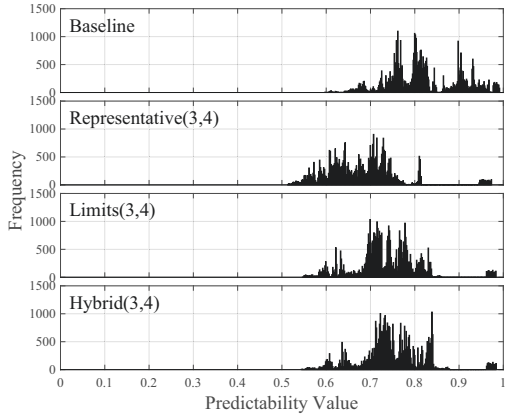
The analysis carried out throughout this chapter has shown the many perspectives from which mobility can be considered. Each of them would be more dominant depending on the application at hand, but none should be disregarded. What is more important is that



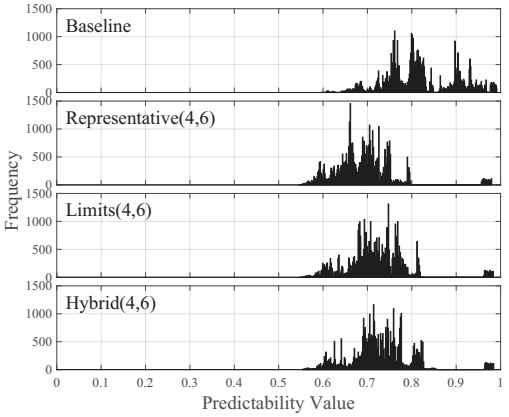
(a) MIT data set, (p,q)=(3,4).



(b) MIT data set, (p,q)=(4,6).



(c) UC3M data set, (p,q)=(3,4).



(d) UC3M data set, (p,q)=(4,6).

Figure 4.18: Distribution of the predictability values at some steps of the baseline trace and traces filtered using two detection schemes and the three filtering techniques, for the two data sets considered.

studying such features is worth it, but it is even more important to think on which mobility data is being considered to foresee the constraints it may have in order to reflect the real mobility features of the user. Chapter 3 introduced three different approached to obtain the movement history of the user based on cellular telephony network, namely baseline, CDR-based and DDR-based approaches, which reflects mobility very differently. CDR-based approach greatly simplifies the real features of mobility due to the general scarce CDR events (calls or messages). Thus, for instance, the number of different locations visited per day extracted from CDRs is reduced to a maximum of 4 different locations per day. This value might be enough to describe the most important places (e.g., home, work), but lacks on reflecting the routes followed by the user to reach them. This fact has direct implications in urban planning, traffic forecasting and, in general, in the understanding of

the routes followed by the users. In a dense populated city, people going to and back from work visit many more locations, but they will rarely appear in CDRs if the user is driving and cellphones are not allowed, or if there is no coverage in the public transportation used. Plenty of works about user mobility [138, 133, 134, 106] are focused mainly on entropy and predictability. Almost all of them use CDR-based traces, since it is easy to obtain them from a large population. However, the above results suggest a re-consideration of the conclusions drawn from those works. This raises a concern about the trade-off between the number of traces available using certain collection scheme, with respect the quality of the data collected. Considering DDRs could improve this trade-off, since the movement histories based on these records are more complete, in terms of mobility, than CDRs.

Nevertheless, it was also demonstrated that the baseline technique alone is not perfect either. However, in this case, the problem is just the opposite one: with the baseline approach, many additional events not reflecting real movements are registered into the movement history. And, as seen in the previous results, the volume of these events, nicknamed as ping pong effect, is extremely high, thus having an enormous impact on the mobility features extracted right directly from the baseline movement histories. Thus, filtering out these ping pong sequences should not be neglected.

In this chapter, the filtering phase has been proposed as a two-stage process: first, detecting the ping pong sequences, and then, replacing those detected sequences by the corresponding CellID, thus filtering them out. The proposal on the detection stage suggests a configurable approach to avoid filtering sequences that might seem at first like ping pong ones, but that are just real movement of the user. This flexibility comes from the difficulties found to determine what is really a movement and because ping pong sequences also happen during movement periods of the user. In fact, by slightly changing the most restrictive detection scheme, the percentage of detected ping pong events decreases notably. Regarding the filtering stage, three different techniques were proposed, namely representative, limits and hybrid one. The representative technique shown to provide the most simplified "movement-pure" histories. However, if the application at hand requires to maintain the real locations structure (e.g., cells adjacency), then the limits or hybrid techniques provides similar results and also a noticeable filtering capacity.

As for the purposes of this dissertation, after studying the biases introduced by the baseline approach when collecting mobility data, it becomes clear that special attention needs to be taken in the next chapter when analyzing the prediction results. At first sight, the results drawn from prediction might seem better than they really are because predicting ping pong sequence is fairly easy. However, it must be noted that those predictions do not foresee movement, but network related issues that are probably worthless for the applications where the mobility predictions are being used.

Chapter 5

Contributions to the Improvement of Lightweight Mobility Prediction Algorithms

Contents

The previous chapter described the main mobility indicators considered in this work to characterize the mobility features of an individual. These features demonstrated to reflect many details of the individual's life, which directly reflects on mobility, such as trips, or changes of the habits. As discussed earlier, all those indicators can help in determining the context of the user and, thus, being able to tailor the behavior of the services or applications to that specific user behavior. The approach followed to capture the location

data, and the noise attached to the useful information, have shown to impact the mobility features encoded by the traces. Therefore, a careful processing of the raw information should be carried out prior to any further analysis or use. The previous chapter shown some proposals on how to detect and filter this noise in an online manner, comparing the noticeable differences between the original and filtered traces.

All these efforts were carried out in order to prepare the path to study the impact of the mobility data used and the observed mobility features on the prediction of future locations. By reviewing the literature and comparisons among different prediction algorithms, exposed in Section 2.1.4, it can be noticed that the results obtained in each study are not contextualized with the type of mobility data or with the mobility features of the individuals in the data set used. However, these factors might have some impact on the prediction results, beyond the effect that using a particular algorithm has in the prediction process itself. In the analysis carried in this chapter, the goal is to analyze both how the prediction algorithms selected—the LZ family—work, and also their relationship with the data itself and the intrinsic mobility features defined in Chapter 4.

First, a proposal to divide and mix the algorithms of the family up is presented and evaluated, aiming at finding the best combination. Next, the central part of the chapter will focus on the prediction results obtained and its relationship with the mobility features studied in Chapter 4. The conclusions extracted from this analysis are next used to propose some improvements in the algorithms, which will be further evaluated with the data sets presented in Chapter 3.

5.1 Background

As described in Section 2.1.4, there exist a myriad of location prediction algorithms, based on many different approaches and data sources. Even narrowing the variety down to those algorithms using location data based on symbolic locations (e.g., Wi-Fi or cellular networks), the range of possible predictors is wide. Therefore, a specific family of prediction algorithms needs to be selected first, in order to concentrate the efforts around a concrete prediction methodology. Specifically, this thesis is focused on a particular family of compression algorithms, known as LZ family, which is comprised by three algorithms: LZ, LeZi Update, and Active LeZi. They are based on Markov models, which will be also considered along the work as reference. The reasons behind this choice are the following ones:

- **The input data of these algorithms must be a sequence of symbols**. This constraint perfectly fits the scenario considered in this thesis, described in Section 3.1, where the space is divided into different regions, each represented by a unique identifier. This option is opposed to the algorithms that focus on direction, velocity and similar physical magnitudes. It should be noted that in order to use such prediction approaches, GPS data would be needed, which implies a continuous tracking of the individual using this technology in her mobile phone, and the subsequent battery drainage.

- The LZ algorithms **detect and continuously integrate changes of the indi-**

vidual's behavior. These algorithms continuously update the individual's mobility model with each new symbol they process. Therefore, if the individual being tracked usually visits certain places in a given order and, at some point in time, she changes the order in which she visits these places, the routes among them, or simply she starts visiting some other different locations, these algorithms realize the changes in the routine, record them into the mobility model, and make the predictions according to the new information. This working principle contrasts with the wide set of algorithms that need an initial offline training stage, after which the parameters of model built by the training phase are set and fixed, thus being not possible to incorporate changes in the individual's mobility habits.

- **Their execution does not imply a high resource consumption**. These algorithms work in a sequential and incremental manner. The mobility model they gradually build is conveniently stored in a tree data structure, so that look-ups are fast and the memory needed for storage as compact as possible. These characteristics are important, since the user is continuously tracked, thus reporting several locations per day. In Chapter 4, Table 4.1 and 4.5 show an average number of location records per day (cell changes per day in the particular case of this thesis) ranging from 30 with the most restrictive filter, up to more than 200 for the baseline case. Considering the increasing volume of data that must be analyzed to make each prediction, the ability of the algorithm to generate predictions in a timely manner becomes critical. Thus, wether the prediction stage is performed by the mobile phone or by a third party, the sequential and incremental behavior, and the compact storage needed is key to be able to continuously generate the predictions during long time periods.

The general working principles of the algorithms considered here are simple. The input data is the movement history of the user, this is the sequence of locations represented by symbols, l, as explained in Section 3.1. This sequence is parsed by the algorithms in order to discover mobility patterns and to estimate, based on the frequency of the locations following each of the detected patterns, which is the most probable next location. The specific details of how each algorithms performs the pattern discovery and next location prediction will be explained next. The description will start with the foundation of the LZ algorithms—the Markov models. Then, each of the algorithms pertaining to the LZ family will be described along with its relationship with Markov models.

5.1.1 k-order Markov Model

An order-k Markov model (or Markov O(k) model) [23] can be used to forecast the next location that will be visited by an individual based on what is called prediction context, c. This prediction context is the sequence of the k most recent symbols (i.e., locations) recorded in her movement history, l.

The Markov O(k) model consists of:

- A finite set of states, which represent each possible prediction context.

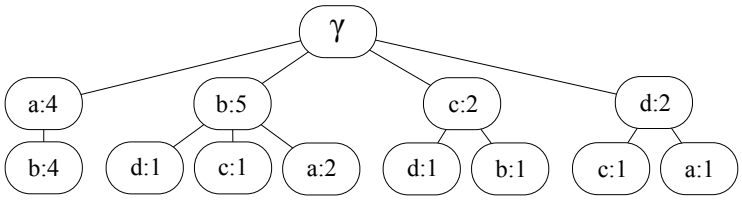


Figure 5.1: Markov O(1) tree after parsing the example movement history.

- The transitions among states and their corresponding probability, representing the possible locations the individual could visit given the current context, meaning the k most recent locations visited by the individual (state).

This information can be stored by means of a tree data structure. This data structure provides a compact storage that allows to speed up the prediction contexts look-ups. For example, the tree corresponding to the movement history $l = ababab cdcbdab$ is the one shown in Figure 5.1. Each level of the tree contains a set of nodes representing all the contexts, c, of order k equal to that level, considering the root as level zero. The symbols making up each context in the tree can be obtained by concatenating the k symbols traversed from the root to the corresponding node. The number accompanying each symbol corresponds to the number of times that the context was recognized in l. For instance, Figure 5.1 shows four contexts of order-1, corresponding to the four nodes a level 1 of the tree. The children of each context represents the events that happened after that context, and with which frequency. For instance, the node $a : 2$ at level 2 means that location a has been visited twice after being at location b, which occurred 5 times in total.

In order to make it easier to understand how each algorithm parses l step by step and how the tree is built, Table 5.1 shows the tree nodes added or which frequency is incremented when each new symbol is recorded in l.

l	**a**	**b**	**a**	**b**	**a**	**b**	**c**	**d**	**c**	**b**	**d**	**a**	**b**
Markov O(1)	a	b	a	b	a	b	c	d	c	b	d	a	b
		ab	ba	ab	ba	ab	bc	cd	dc	cb	bd	da	ab
LZ	a	b	a	ab	a	ab	abc	d	c	b	bd	a	ab
LZU	a	b	a	b	a	b	c	d	c	b	d	a	b
				ab		ab	bc				bd		ab
							abc						
ALZ Window	a	b	a	ab	ba	ab	abc	bcd	cdc	dcb	cbd	bda	dab
ALZ	a	b	a	b	a	b	c	d	c	b	d	a	b
				ab	ba	ab	bc	cd	dc	cb	bd	da	ab
							abc	bcd	cdc	dcb	cbd	bda	dab

Table 5.1: Comparison of the example movement history parsing done by each algorithm.

In order to make predictions, the interesting information in this model is the probability of the next location being each of the ones contained in $\mathscr{L}$, given the current context. Thus, the most probable next location will be the one with the highest transition probability. Transitions among states, or the probability of the next location to be l_{n+1} given the current prediction context, c, is expressed by equation (5.1):

$$P(L_{n+1} = l_{n+1}|c) = \frac{N(cl_{n+1}, l)}{N(c, l)}, \forall l_{n+1} \in \mathscr{L} \tag{5.1}$$

where $N(cl_{n+1}, l)$ is the number of times the context c has been followed by symbol l_{n+1} in the whole movement history l, and $N(c, l)$ is the number of times the context c is contained in l.

An increment in the order of the model may suggest an improvement in the prediction accuracy, since the model deals with more different and concrete contexts, thus adapting better to each particular situation. However, as the order of the model increases, the frequency of the tree leaves decrements significantly. This is due to the fact that the number of possible prediction contexts grows exponentially: with an alphabet $\mathscr{L}$ of cardinality $|\mathscr{L}|$, the number of different contexts of size k grows as $|\mathscr{L}|^k$. Therefore, for a given trace length, N, there are more different contexts as k increases, and thus, less samples of each context and its corresponding next locations. This fact makes the probability estimation worse, since there are less samples with which the calculations can be made. In [12], the authors state that the entropy of a mobility model built by Markov O(k) algorithm decreases as k increases, meaning that the uncertainty the model encloses about the next location of the user decreases, as explained in Section 2.2. However, it does not decrease indefinitely, but there is a value of k above which it is not possible to obtain a lower entropy. Therefore, the question that follows from this fact is: which is the optimal order, k, and how to dinamically figure it out based on each specific movement history? The answer is given by the LZ-based algorithms. They build a similar model to Markov ones (represented by means of a tree like the one depicted in Figure 5.1), but with a variable order, k, that grows or remains constant depending on l. The main advantage of these algorithms is that the k value is granted to be always optimal, meaning that the entropy of the model will always be minimal. This implies that the uncertainty about the next event in the movement history of the user will be minimal too. Next sections will describe each LZ-based algorithm in detail.

5.1.2 LZ Algorithm

This is the basic algorithm of LZ family and works as follows [163]. Let γ be the empty string and l the input movement history. The LZ algorithm takes l and splits it into substrings $s_0 s_1 \dots s_m$ such that $s_0 = \gamma$ and for all $j \geq 1$ the prefix of substring s_j, meaning all but the last character of s_j, is equal to some previous s_i, for all $i < j$. It is important to notice that the division is made sequentially, so when each s_i is determined the algorithm then considers only the remaining trace. Taking the following example history, $l = ababab cdcbdab$, the division will be as follows: γ, a, b, ab, abc, d, c, bd, ab.

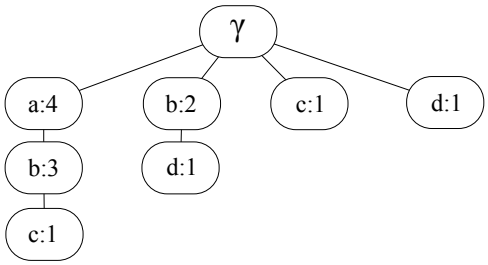


Figure 5.2: LZ tree after parsing the example movement history.

In order to store these substrings (also referred to as patterns), the LZ algorithm builds a tree, called LZ tree, which grows dynamically during the analysis and division of the movement history. Each tree node represents a substring and stores the number of times that substring appears among the patterns parsed by the algorithm. Figure 5.2 depicts the LZ tree resulting from parsing the example movement history. For example, the node $b:3$ corresponds to substring ab, which have appeared 3 times among the substrings parsed by LZ algorithm: ab, abc, ab. Table 5.1 shows how l is parsed by the LZ algorithm and stored in the tree step by step.

Each time a symbol is processed, the first step is to update the tree as explained above. The next step is to calculate the probability for each known symbol to be the corresponding to the next location. In order to do that, the LZ algorithm uses an approach proposed by Vitter [147] that can be expressed as in equation (5.2):

$$P\left(L_{n+1} = l_{n+1}|l\right) = \frac{N^{LZ}(cl_{n+1}, l)}{N^{LZ}(c, l)}, \forall l_{n+1} \in \mathscr{L} \tag{5.2}$$

where c is called prediction context and corresponds to the last substring that has been parsed by LZ algorithm (e.g., looking at Table 5.1, it can be seen that the context, c, in step 6 is ab and in step 12 is a); $N^{LZ}(cl_{n+1}, l)$ represents the frequency of the substring cl_{n+1} (i.e. the prediction context followed by symbol l_{n+1}) in the LZ tree and $N^{LZ}(c, l)$ represents the frequency of the substring c also in the LZ tree. Finally, the LZ algorithm chooses the symbol with the highest probability of being the next location.

Observing behavior of this basic algorithm, three drawbacks can be spotted:

- The patterns between two detected substrings are lost. In the example, dc is followed by b, but cb is not in LZ tree.

- The patterns within substrings parsed by the LZ scheme are also lost. For instance, the abc pattern is in the LZ tree, but bc is not.

- The Vitter calculation method presents problems when a pattern is detected for the first time, since it has not enough information and is not able to make any prediction

The two next algorithms try to overcome these limitations.

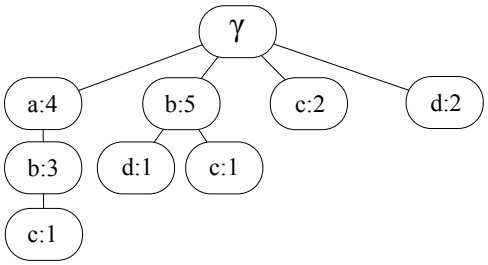


Figure 5.3: LZU tree after parsing the example movement history.

5.1.3 LeZi Update Algorithm

Bhattacharya and Das [12] proposed an heuristic variation in order to include patterns within the substrings parsed by LZ algorithm. The LeZi Update algorithm applies the same parsing made by the LZ algorithm, but instead of adding only the substrings resulting from the LZ parsing, the LeZi Update also adds to the so called LZU tree all the suffixes of each substring. Analyzing the former example, LeZi Update parses l as follows: γ, a, b, $ab\{b\}$, $abc\{bc, c\}$, d, c, $bd\{d\}$, $ab\{b\}$, where the substrings outside the brackets correspond to the output of the LZ parsing process, and the ones inside the brackets are the additional ones added by the modifications introduced by the LeZi Update algorithm. In Table 5.1 it can be seen which LZU tree nodes are added or updated when each new symbol is recorded on l, and the final LZU tree is shown in Figure 5.3.

Regarding the probability calculation method, the LeZi Update algorithm uses Prediction by Partial Matching (PPM) [26]. This approach solves the problems arising from using the Vitter approach and the probability estimation is made up from much more information. PPM works as follows. First, the longest prediction context c, needs to be determined. In this case, c corresponds to the longest substring (starting by the last symbol of l) already included in the LZU tree. In the former example, the longest prediction context is order 2 ($k = 2$), $c_2 = ab$, since there is no substring dba (immediate higher order prediction context) in the LZU tree yet. With the current prediction context, a table like Table 5.2 can be built. It gathers the frequency of each substring that has followed the prediction contexts of order 2 ($c_2 = ab$), order 1 ($c_1 = b$) and order 0 ($c_0 = \gamma$). In this table it is also included what is called escape event, which refers to the number of times a pattern is not followed by any symbol. For example, the substring ab has a frequency equal to 3 but it has a child whose frequency sums 1 event, and thus there are 2 escape events. This happens because the first time ab is parsed, the next symbol is not considered, and because the last substring of the trace is also ab, and the symbol following it is still unknown. Besides, LeZi Update applies what is known as exclusion technique. This means that it only considers one-symbol substrings (as the goal is only to predict the next location), excluding the remaining ones. For instance, for c_0, only the contexts $a : 1$, $b : 3$, $c : 2$, and $d : 2$ would be considered.

Once having the table filled, the probability is calculated as shown in equation ((5.3)):

$c_2 = ab$	$c_1 = b$		$c_0 = \gamma$				
c:1	c:1	esc:3	a:1	abc:1	bc:1	c:2	esc:0
esc:2	d:1		ab:2	b:3	bd:1	d:2	

Table 5.2: Frequency of the substrings following each context, c_k, when the LeZi Update algorithm parses the example movement history.

$$P(L_{n+1} = l_{n+1}) = P_k(l_{n+1}) = P(l_{n+1}|c_k) + P(esc|c_k) \cdot P_{k-1}(l_{n+1}), \forall l_{n+1} \in \mathscr{L} \tag{5.3}$$

which translated to the example, and taking $l_{n+1} = c$ as the target symbol which probability is wanted to be known, results in Eq. (5.4):

$$\begin{aligned} &P(L_{n+1} = c) = P_2(c) = \\ &P(c|ab) + P(esc|ab) \cdot P_1(c) = \\ &P(c|ab) + P(esc|ab) \cdot \{P(c|b) + P(esc|b) \cdot P_0(c)\} = \\ &P(c|ab) + P(esc|ab) \cdot \{P(c|b) + P(esc|b) \cdot P(c|\gamma)\} \end{aligned} \tag{5.4}$$

and applying the data in Table 5.2 results in Eq. (5.5):

$$P(L_{n+1} = c) = \frac{1}{3} + \frac{2}{3} \cdot \{\frac{1}{5} + \frac{3}{5} \cdot \frac{2}{13}\} \tag{5.5}$$

Despite the improvements introduced by this algorithm, the patterns between consecutive parsed substring are still undetected, and thus not included into the corresponding tree. The following algorithm addresses this pending drawback.

5.1.4 Active LeZi Algorithm

The algorithm proposed by Gopalratnam [51] is intended to consider the substrings among consecutive parsed patterns when building the so called ALZ tree, thus solving the remaining problem of the LZ algorithm. In order to achieve this, the Active LeZi algorithm uses a window of variable length, which is determined by the longest pattern parsed by the LZ algorithm at each step. This scheme works as follows. When the algorithm detects a new symbol, it makes the same parsing as the original LZ algorithm. Once the length of the new parsed pattern is known, the window length is updated (if needed) and the new symbol is added to the window. Finally, all the suffixes of the window are added to the tree. Table 5.1 shows the evolution of the window (row labeled as ALZ Window) and the substrings that are added or updated at each step to the ALZ tree represented in Figure 5.4.

The probability calculation process is based on the PPM algorithm as before, so expression (5.3) still applies. However, instead of using the exclusion method, in this case the PPM method takes into account the symbols (not substrings) that are children of a given context, as shown in Table 5.3. The Active LeZi algorithm solves all the initial problems at the expense of increasing the information stored, and therefore memory and time resources required.

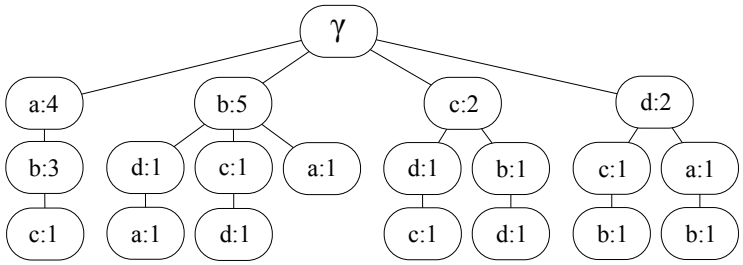


Figure 5.4: ALZ tree after parsing the example movement history.

$c_2 = ab$	$c_1 = b$		$c_0 = \gamma$		
c:1	a:1	d:1	a:4	c:2	esc:0
esc:2	c:1	esc:2	b:5	d:2	

Table 5.3: Frequency of the symbols following each context, c_k, when the Active LeZi algorithm parses the example movement history.

5.2 Combining LZ-based Location Prediction Algorithms

The previous section detailed the working principles of the prediction algorithms under study. In order to provide a first overview of their prediction performance, the MIT and UC3M data sets will be used as input mobility data, to analyze the baseline prediction accuracy of each algorithm.

But first, it is worth to take an even deeper look into the working principles of each algorithm. By examining them carefully, it can be noticed that they share a common structure. Every algorithm takes each new symbol, processes it to update the mobility model stored in corresponding tree, and finally, using the transition probabilities enclosed in the frequencies of each node of the tree, calculates some probabilities to determine the most probable next location. Therefore, two different and independent stages can be distinguished, as shown in Figure 5.5:

1. **Tree updating scheme**. This phase is in charge of learning and building up the mobility model of the user. In order to do that, the algorithm processes each new symbol together with the current context at each step, looking for mobility patterns. The patterns can be sequences of symbols already seen before, and also sequences not seen before but that can potentially be new patterns. In order to save the mobility model of the user in a compact way, so that further pattern look-ups are fast, the parsed location sequences are added to the corresponding tree, which contains the mobility model of the user.

2. **Probability calculation method**. The second step is the one actually providing the location prediction. For this task, each algorithm uses the updated tree resulting from the previous stage (i.e., the mobility model of the user). The model contains the probability transitions between states, and thus by performing certain calculations,

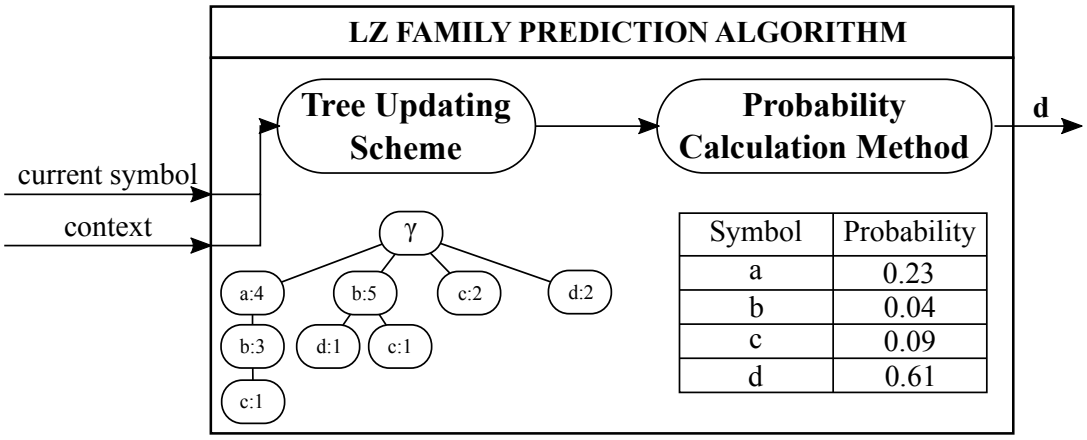


Figure 5.5: Internal division in two stages of the LZ-based prediction algorithms.

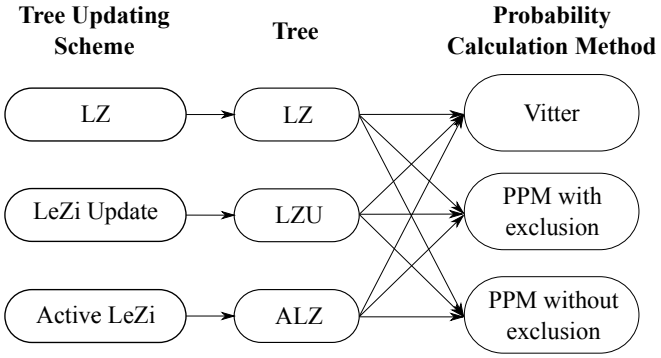


Figure 5.6: Available combinations when splitting the algorithms into two independent stages.

the probability of each known symbol to be the corresponding to the next location can be estimated. The estimation is based on the individual's current context, this is, the most recent locations, together with the current one. Once all the probabilities are calculated, the prediction will be the symbol with the highest probability.

Figure 5.6 shows that this division gives rise to nine combinations. This procedure allows to study which combination shows the best prediction accuracy, and to analyze the impact of each stage in the prediction process. The next section covers the evaluation of the combinations proposed here.

5.2.1 Evaluation of the Basic Combinations

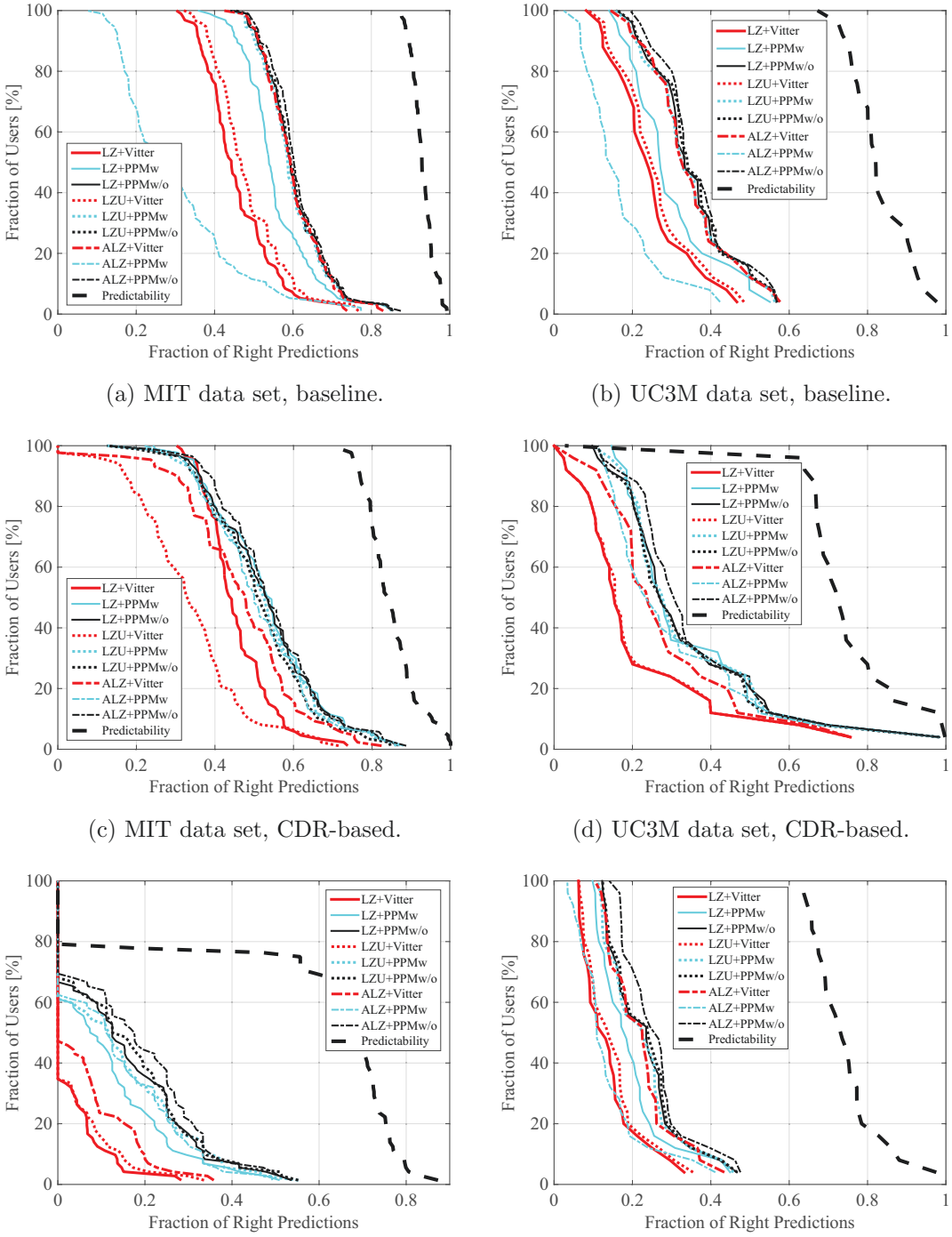
The division of the original LZ-based algorithms into two independent stages generates nine different combinations that will be evaluated in this section. First, focusing on the number of right predictions with respect to the total number of predictions made attained by each combination (which will be referred to as accuracy), Figure 5.7 shows the percentage of

users (y-axis) that attain, at least, the corresponding fraction of correct prediction (x-axis), for the baseline, CDR and DDR-based traces of the MIT and UC3M data sets. Besides, in order to compare the results with the individuals' mobility characteristics considered in these data sets, one of the human mobility features studied in Chapter 4 will be considered: the mobility **predictability**. As described in Section 4.1, the concept of predictability measures the maximum fraction of correct predictions that any prediction algorithm can ever achieve when considering a particular movement history. It is determined by the randomness of the user's mobility, quantified in terms of the entropy rate of the trace (see Section 2.2 for more details about the concept and calculation of the entropy rate of a finite symbol sequence). Thus, the predictability provides an idea of how far the prediction accuracy of each algorithm could reach, given the particular users considered (which will never be a 100%).

Comparing first the probability calculation methods (lines with the same color), Figure 5.7 shows that PPM without exclusion (PPMw/o) works best with any of the tree updating schemes, for the three data sources. Regarding the results derived from using Vitter method, they are very close to those attained by the PPM without exclusion approach when combined with the ALZ tree in both cases, even being a much simpler calculation method, and thus consuming less resources. However, the Vitter calculation method does not work well when using the LZ or LZU trees, since they lack much information about the movement patterns to provide a good prediction if a simple method, like the Vitter approach, is used. Lastly, PPM with exclusion (PPMw) provides poor results when combined with the ALZ tree, even when this tree is the one storing the maximum number of patterns. This fact may be surprising, but taking a deeper look into the working principles of the PPM with exclusion method, the reason of this behavior becomes clear. Let $l = ababab cdcbdab$ be the movement history of a user and the ALZ tree represented in Figure 5.4 be the one corresponding to such movement history built by the Active LeZi algorithm. Table 5.4 is the table that the PPM with exclusion method builds when analyzing that tree, where c_k are the different contexts PPM uses for calculating the next symbol probabilities.

$c_2 = ab$	$c_1 = b$		$c_0 = \gamma$				
c:1	a:1	da:1	a:1	ba:1	bda:1	cd: 1	dab:1
esc:2	c:0	esc:0	ab:2	bc:0	c:0	cdc:1	dc:0
	cd:1		abc:1	bcd:1	cb:0	d:0	dcb:1
	d:0		b:3	bd:0	cbd:1	da:0	esc:0

Table 5.4: Frequency of the substrings following each context, c_k, when the Active LeZi algorithm parses the example movement history.



(a) MIT data set, baseline.

(b) UC3M data set, baseline.

(c) MIT data set, CDR-based.

(d) UC3M data set, CDR-based.

(e) MIT data set, DDR-based.

(f) UC3M data set, DDR-based.

Figure 5.7: Fraction of users in the MIT and UC3M data sets attaining, at least, the corresponding fraction of right predictions (or less), for each algorithm combination, when considering the baseline, CDR, and DDR-based data collection schemes.

By taking the one-symbol substrings (the ones representing the next location), it can be seen that most of them have frequency 0 due to the fact that the Active LeZi algorithm adds all the substrings whose length is equal to the window length at each step to the ALZ tree. Therefore, most of the intermediate tree nodes are only part of longer substrings instead of being a substring parsed by themselves. For example, substring *dcb* has been added in one step, and therefore the PPM with exclusion method only considers it as a complete substring *dcb*, without noticing about intermediate nodes *dc* or *d*. Therefore, as the PPM method considers shorter contexts ($c_1 = b$, $c_0 = \gamma$), the symbol frequencies are lower (being 0 in many cases) because all or most instances of those nodes are probably part of longer substrings that have been added in one step. This phenomenon entails two conclusions:

- PPM with exclusion does not seem to be very appropriate in the scenario covered in this thesis, since the focus is on predicting the next event (instead of the sequence of future events).
- The lowest orders are barely taken into account, since the PPM with exclusion algorithm quantifies these frequencies in such a way that they turn to be very low or even 0. This fact is specially critical since the lowest orders have the highest number of samples, thus potentially providing more accurate predictions.

With respect to the comparison of updating schemes (same line types), thus fixing the probability calculation method and applying different updating schemes, the Active LeZi algorithm is the best choice when working with Vitter and PPM without exclusion, although the differences in the last case are very small. LeZi Update works better with PPM with exclusion because of the reasons previously discussed. Without taking into account the combination of Active LeZi with PPM with exclusion method, the results are coherent with those shown in [136]. This conclusion could be foreseen since the patterns information gathered by the ALZ tree is greater with respect to the LZU tree, and the same applies to the LZU tree with respect to the LZ tree.

Regarding the results with the CDR and DDR-based traces, the results when comparing the algorithms are different. For the case of CDR-based data, the PPM without exclusion approach is still the best probability calculation method, but in this case, the PPM with exclusion method works equally well. The Vitter method is the one performing the worst, for all the tree updating schemes. This fact gives an idea of the different type of information enclosed by the traces collected following the CDR-based approach with respect to the baseline case, which makes it difficult to compare different prediction algorithms if they are evaluated with data coming from different sources, as usual in the literature.

As mentioned in Chapter 3, the DDR-based data in the MIT data set is scarce, so the results shown in Figure 5.7e cannot be taken into account due to the small amount of data from which they come from. However, the DDR-based traces of the UC3M data set complement this information, showing a similar behavior of all the algorithms combinations with respect to the baseline case: PPM without exclusion is again the calculation method providing the best results, whereas ALZ tree is the updating tree scheme that performs best. The main difference stems from the maximum prediction accuracy values, which are

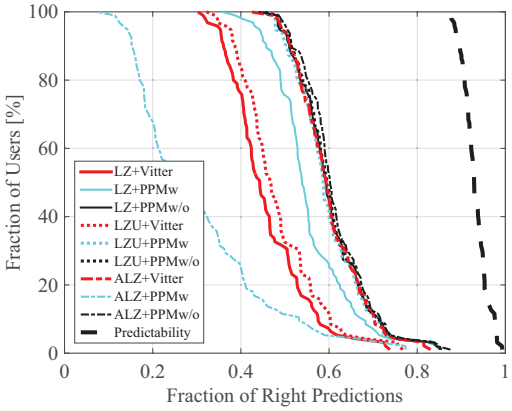
much lower than in the baseline case, as well as the predictability values. Later on this chapter, these results will be re-considered again, unveiling the real similarity between the DDR and baseline schemes in the prediction results.

The comparison between the results obtained with each data set provides also interesting insights. In all cases, the prediction accuracy and the predictability share a very similar shape. However, in the UC3M data set, the prediction accuracy is further away from the upper bound set by the predictability than in the MIT case. Therefore, in order to evaluate predictions algorithms, the data set used need to be evaluated first, before coming to any conclusion. In all plots it can be observed that a decrease in the predictability implies an even lower prediction accuracy. Both lines are not exactly parallel, but their distance shrinks as the value of the predictability increases. Therefore, knowing this parameter of human mobility is key to assess the performance of the algorithm, isolating this performance from the specific data features.

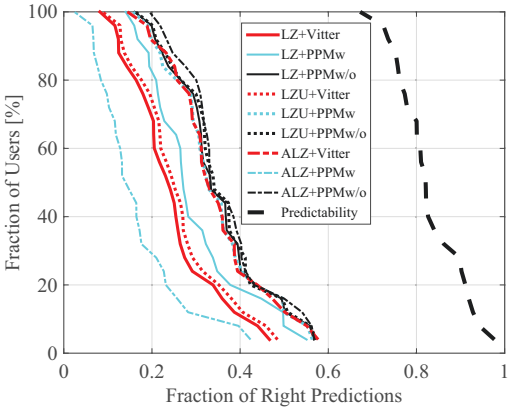
5.2.2 Evaluation of the Useful Predictions

In Section 4.3, it was shown that the traces of both the MIT and UC3M data sets include a high number of ping pong events, especially those of the MIT data set. In that section, different detection and filtering techniques were proposed in order to eliminate these events, which do not represent any real movement of the user, and which shown to bias the real mobility features reflected in the traces. Therefore, if the mobility features are noticeable biased by these ping pong events, the learning and prediction processes can be potentially biased as well. This section analyzes this hypothesis and shows the results of making predictions using the filtered traces obtained in Section 4.3, comparing the results with the ones corresponding to the baseline traces, exposed in the previous section.

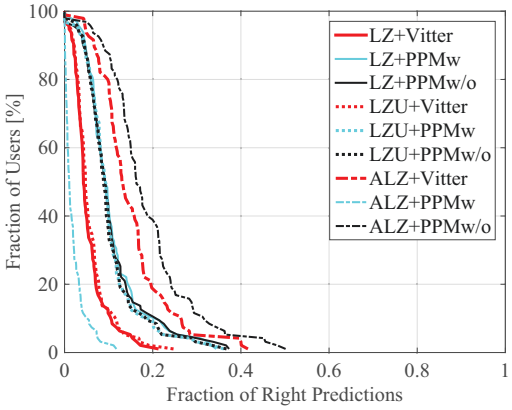
Recalling the ping pong detection procedure described in the previous chapter, it depended on two parameters, p and q, which delimited the number of samples needed to determine if a ping pong sequence among 2 or 3 cells, respectively, was taking place. The output of this procedure was just a sequence indicating if the cell record at each position of the movement history corresponded to a ping pong sequence or to a real movement. For instance, if the movement history $l = abcbcbcdeababa$ is considered, the output of the detection procedure would be 01111110011111, where 1 indicates the existence of a ping pong sequence. Therefore, with this sequence indicating the symbols (locations) recorded into the location history and pertaining to ping pong sequences, it can be determined the number of correct predictions corresponding to ping pong sequences, and the number of right predictions indicating the most probable next real location of the user. This last aspect will be referred to as **useful predictions**, since among all the predictions provided by the algorithms, the actual interest lies in those predicting the real movement of the user. Figure 5.8 shows the fraction of correct predictions attained for the baseline case, as well as the fraction of useful predictions of the MIT and UC3M traces, when using two ping pong detection schemes, (p, q), corresponding to $(3, 4)$ and $(4, 6)$. The fraction of correct useful predictions is calculated considering the number of events in the trace that do not belong to a ping pong sequence.



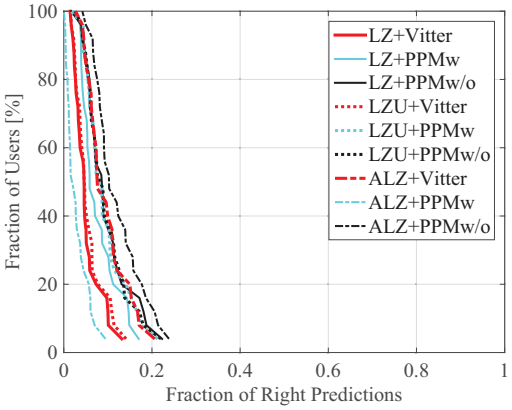
(a) MIT data set, baseline.



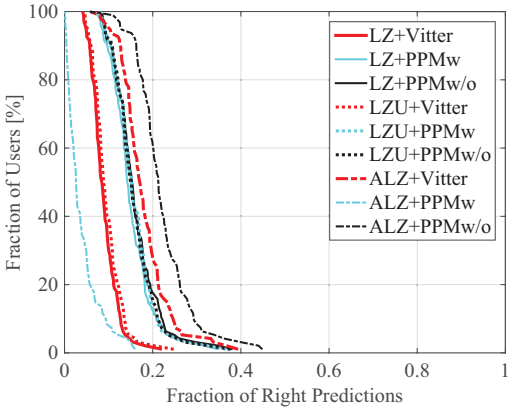
(b) UC3M data set, baseline.



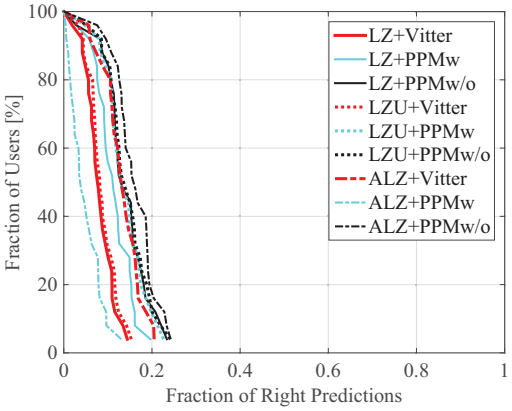
(c) MIT data set, $(p, q) = (3, 4)$.



(d) UC3M data set, $(p, q) = (3, 4)$.



(e) MIT data set, $(p, q) = (4, 6)$.



(f) UC3M data set, b$(p, q) = (4, 6)$.

Figure 5.8: Fraction of users in the MIT and UC3M data sets attaining, at least, the corresponding fraction of right predictions (or less), for each algorithm combination, when considering the baseline case and the useful predictions, for two ping pong detection schemes, (p, q).

Starting with the MIT results, it can be observed that, whereas for the baseline traces 50% of the users can attain 60% of correct predictions, when considering the fraction of right useful predictions for both schemes collapse to values where 50% of the users do not even obtain 20% of right predictions about real movements. This is due to the observed high number of ping pong events, already described in 4.3.2. Therefore, the most part of the correct predictions in the baseline traces correspond to ping pong sequences. In the UC3M case, the fraction of correct predictions drops from around 35% for 50% of the users, to 10% or 15%, for each detection scheme, (3,4) and (4,6), respectively. The decrease is not as big as in the MIT case, corresponding to a lower number of ping pong effects, as already noticed in Section 4.3. In general, for both data sets, the noticeable difference of the prediction accuracy when the ping pong effect is considered or not responds to the fixed structure of those ping pong sequences—two or three symbols, continuously repeated. Thus, it is extremely easy to predict the next symbol of the ping pong sequence, just by knowing the one or two previous symbols. However, although easy to predict, this predictions are useless to foresee the future location the user will visit.

Therefore, the next question to investigate is the real fraction of right predictions achieved when considering the filtered traces coming form the three filtering techniques proposed in Section 4.3—representative, limits, and hybrid. Figure 5.9 shows these results for the case of the detection scheme $(3, 4)$, considering both the MIT and UC3M data sets. The results and conclusions obtained are just the same ones than for the $(4, 6)$ detection scheme, so the plots corresponding to that case are neglected to avoid redundancy. It is worth to emphasize the difference between this case and the results of considering the useful predictions, studied so far. In this later case, each baseline trace is used as the input of the prediction algorithms in order to obtain the predictions of the next location at each step. Then, considering the ping pong events considered in each trace, the predictions corresponding to ping pong events are neglected, since they are not useful for movement prediction purposes. However, the case analyzed next provides a different perspective. The baseline traces are first filtered with the procedures described in Section 4.3, thus generating three different filtered traces for each baseline one. These filtered traces are the ones used now as input of the prediction algorithms, to obtain the corresponding predictions. The goal is to check if the prediction accuracy improves when no ping pong sequences are included as part of the mobility model of the user (stored in the corresponding tree), and thus just movement patterns are detected and learnt. That would mean that the real movement of the user are better predicted by using the filtered traces instead of the baseline ones as input of the prediction algorithms.

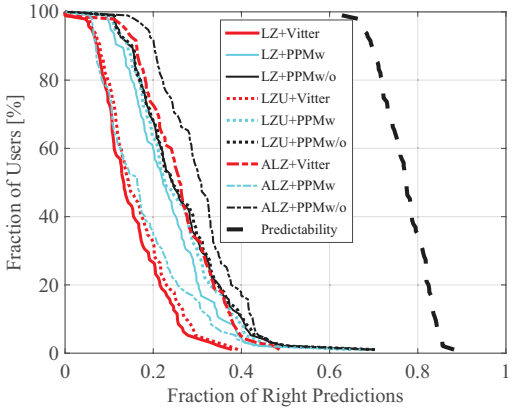
In all subfigures of Figure 5.9, the upper bound determined by the predictability of the resulting traces in each case is also represented. The predictability values in the filtered traces are substantially lower than in the baseline case, shown in Figure 5.8, because the ping pong sequences contribute to decrease the randomness of the trace (i.e., the perceived, but unreal, randomness of the individual's mobility). Thus, by eliminating this effect, the randomness increases and the predictability decreases. However, the prediction results fall down even more than the predictability, and the gap between both of them is wider than in the baseline case. As mentioned before, the lower the predictability, the bigger is the difference between its value and the prediction accuracy. In the case of the representative

filtering technique, the prediction accuracy achieves the lowest value because this technique completely deletes all ping pong sequences, leaving just one symbol as representative of the sequence. Thus, the technique is not adding any fixed structure than can be easily predicted. However, in the limits and hybrid techniques cases, both of them add some more overhead (the limits of the sequence), thus decreasing the impact of the complete ping pong sequences, but leaving some reminiscence of it. This can be observed in the prediction accuracy achieved in both filtered traces sets, where the prediction accuracy is higher than in the representative filtering case.

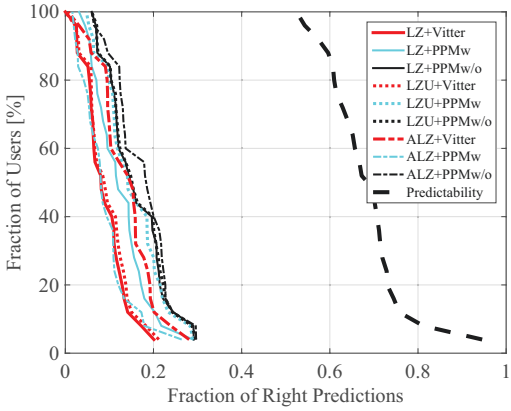
Regarding the prediction algorithms performing best, it can be noticed that, unlike in the baseline case, there is a combination performing clearly better than the rest: the Active LeZi with PPM without exclusion. That means that the amount of information this algorithm stores in its tree is critical in this case, since the results of the rest of updating schemes are further away to the results obtained by when using the ALZ tree, when combined with the PPM without exclusion method. Besides, this probability calculation technique shows to play an important role also, since the combinations closest to this best one are those combined with PPM without exclusion method.

Comparing the prediction results of the filtered traces with respect to the useful predictions, shown in Figure 5.8, it can be noticed that filtering the traces improves the prediction accuracy. In the MIT case, whereas the fraction of right prediction was around 15% for 50% of the users, it increases up to around 25% for the representative case and close to 30% in the limits and hybrid cases. In the UC3M traces, the improvement is less noticeable. Whilst obtaining around 15% of correct useful predictions for 50% of the users, this percentage increases up to 20% when filtering the traces with the representative technique, and to close to 25% when using the other two filtering techniques. It can also be observed that, even after the filtering process, the UC3M traces show a lower predictability than the MIT ones. This leads to think that the set of users considered in the MIT data set are indeed more predictable due to the common working or studying environment. However, the UC3M data set was composed of traces coming from users with varied occupations and frequented places. Thus, their predictability is much lower, which implies also a lower prediction accuracy. These results obtained after filtering the original traces reinforce the importance of a previous analysis of the data, before assessing any prediction algorithm.

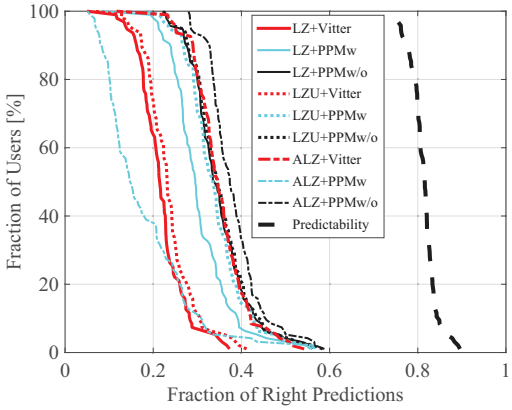
Finally, it is worth to notice the similarity between the prediction results derived from the filtered and the DDR-based traces, depicted in Figure 5.7 (for the UC3M data set). As mentioned in the previous section, the DDR-based traces led to a prediction accuracy much lower than the baseline case. However, as can be seen now that the filtering process was applied, this decrease observed in the DDR case came from neglecting the ping pong effects introduced in the baseline traces. The noticeable similarity between the results of the filtered and DDR-based traces suggests that the DDR-based collection scheme can be also a good candidate to capture the mobility of the user, better than the widely used CDR-based one.



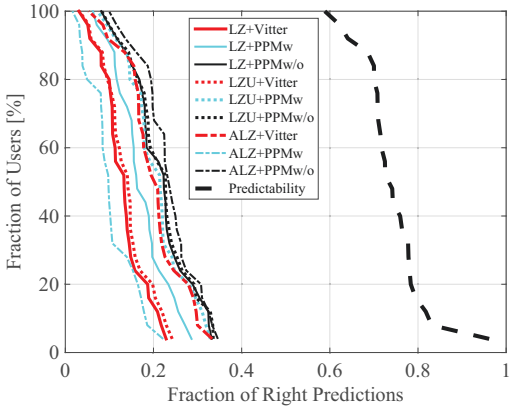
(a) MIT data set, representative (3,4).



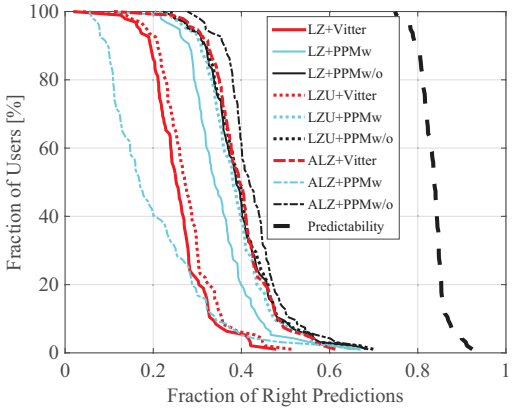
(b) UC3M data set, representative (3,4).



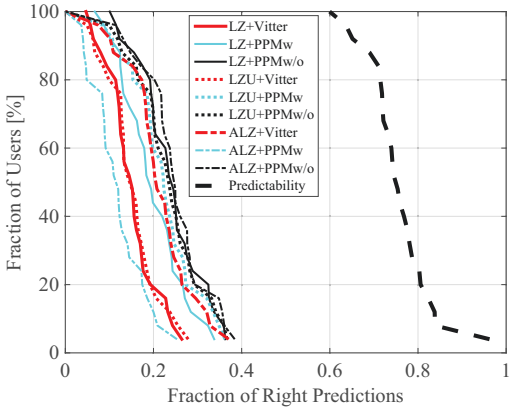
(c) MIT data set, limits (3,4).



(d) UC3M data set, limits (3,4).



(e) MIT data set, hybrid (3,4).



(f) UC3M data set, hybrid (3,4).

Figure 5.9: Fraction of users in the MIT and UC3M data sets attaining, at least, the corresponding fraction of right predictions (or less), for each algorithm combination, when considering the baseline case and the three predictions techniques combined with the detection scheme (3,4).

5.2.3 Comparison with Classical Markov Models

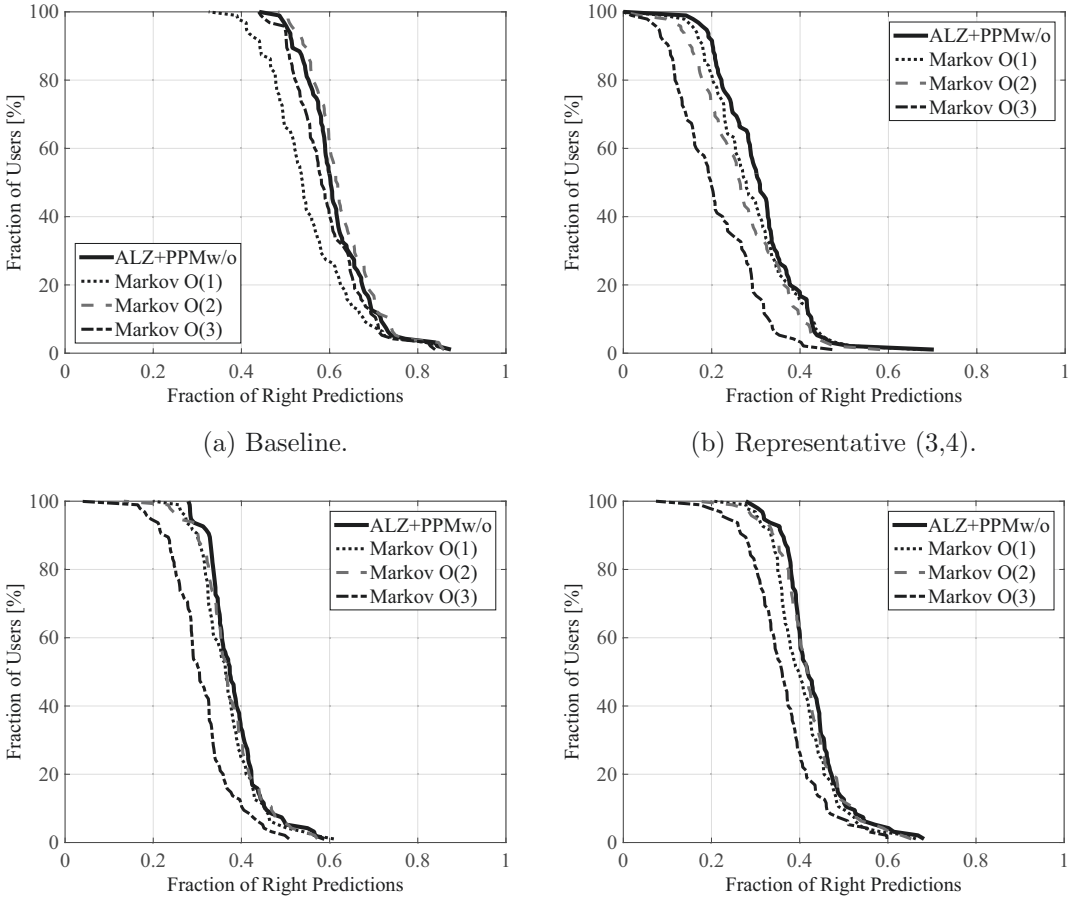
As stated in the literature review in Section 2.1.4, many works use Markov models, mainly of orders 1 or 2, as prediction algorithms of the future movements of the user. Although these models are simpler than the LZ-based ones, as discussed in Section 5.1, the LZ-based models should, theoretically, outperform Markov models because they store more information in their trees. As previously described, the main feature of the LZ family is that the algorithms are able to dynamically compute the optimal order of the model underneath, so that the entropy, and thus uncertainty about the next location, is minimized. However, some works in the literature, like [136], show how an order-2 Markov model achieves better results than the LZ or LeZi Update algorithms when applied to prediction purposes. These results pose the question on why a Markov model of low order can perform better than a LZ-based algorithm that, theoretically, achieves minimum entropy and uncertainty values.

In order to ask this question, the MIT data set has been used to feed three Markov models or orders 1 to 3, in order to check their prediction accuracy. Figure 5.10 shows the comparison of the fraction of right predictions attained by these Markov models, with respect to the ALZ tree combined with the PPM without exclusion method. It can be seen that for the baseline case, the order-2 Markov model provides the best results, slightly outperforming the LZ-based solution. However, when considering the filtered traces, meaning that no ping pong sequences are present, the results are quite different. The Markov models perform worse than the LZ-based solution, whilst the difference of prediction accuracy among them decrements as more repetitions are added to the trace. The traces filtered by the representative technique show a larger gap between the Markov and LZ results, whilst the traces filtered by the limits and hybrid methods, which add some fixed repeated patterns representing the ping pong sequences, show a narrower gap between the results of the different prediction approaches.

It can be deducted that the order-2 Markov model can easily and accurately predict the next symbol in every sample of a ping pong sequence (or any sequence with a fixed structure), since these sequences are composed of two or three symbols continuously repeated. However, as soon as those effects are deleted, Markov models show to fall short to represent the movement patterns of the user. This result contrasts with many works in the literature, where Markov models are used due to the good results they seem to offer, thus inviting to reflect on the data used in such studies and the potential data-related effects that might lead such good results when using Markov models.

5.2.4 Using Several Symbols as Prediction Output

The evaluation done so far considered as prediction the symbol with the highest probability to be the next one (i.e., the symbol with the highest probability in Figure 5.5). However, as mentioned in Chapter 4, a drawback of using the cellular telephony network is that even if the individual is not moving, her mobile phone can be switching the connection among several cells due to reasons others than movement, mainly related to the network initiative (load balance, better signal reception due to weather conditions, etc.). Therefore, sometimes a location can be equally represented by more than one cell, and thus, the



(a) Baseline.

(b) Representative (3,4).

(c) Limits (3,4).

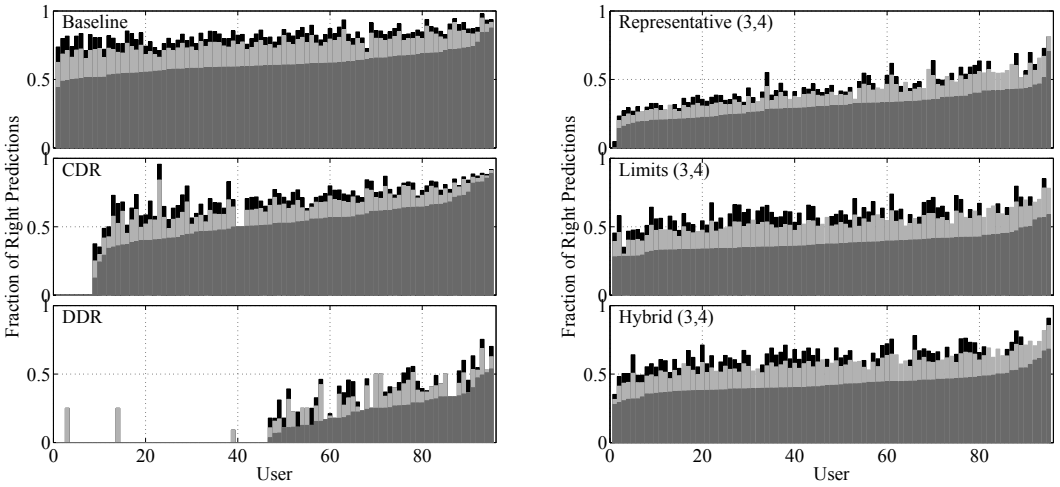
(d) Hybrid (3,4).

Figure 5.10: Fraction of users in the MIT data set attaining, at least, the corresponding fraction of right predictions (or less), when considering the baseline case, and the original Active LeZi algorithm compared to the Markov models of order 1, 2, and 3.

prediction can be equally useful if it is either of the cells representing the location.

Figure 5.11 shows the increment in the percentage of correct predictions when using the two and three symbols with the highest probability of being the next one, with respect to the case of considering only the most probable symbol. It can be seen that using just two symbols improves greatly the prediction accuracy, whilst using three symbols does not provide such a significant improvement.

The increment in the prediction accuracy can be specially observed in the baseline case, as well as the filtered traces using the limits and hybrid techniques. On the other hand, the traces filtered with the representative technique are the ones less improved by the use of two symbols, and using three symbols provide a barely noticeable improvement. Recall that in this case no ping pong sequences are present in the trace, thus the use of two



(a) MIT data set, original traces.

(b) MIT data set, filtered traces.

Figure 5.11: Fraction of right predictions for each of the users in the MIT data set, when using the original Active LeZi algorithm and considering as prediction the one, two or three most probable next symbols, for the baseline, CDR and DDR-based traces, as well as the traces filtered with the three filtering techniques and relying on the (3,4) detection scheme.

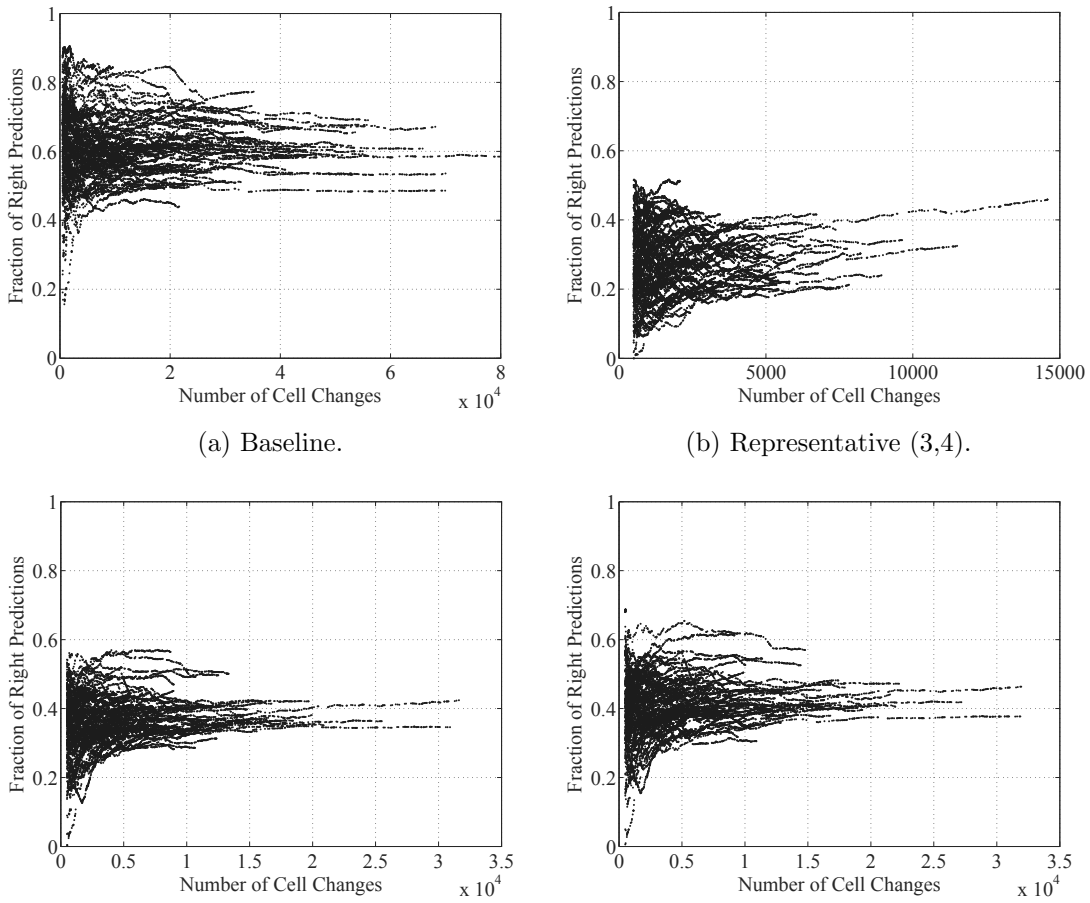
symbols is not that critical as in the other cases.

Still, the fact that the user can connect equally to more than one cell when moving in the same direction due to differences in the signal strength received at every moment or because load balance issues, gives rise to an uncertainty not only about the movement of the user, but also in the network behavior. Thus, using two symbols can help to tackle this problem, inherent to the use of cellular networks as proxies of the user movement, and which can be extended to the use of other wireless networks.

5.3 Relationship between Prediction Accuracy and Mobility Features

Once the prediction performance of the original algorithms has been assessed, the next step is to relate that performance with some of the mobility features studied in Chapter 4. The analysis will focus on potential relationships that can be further leveraged to improve the prediction performance achieved so far by the original LZ-based algorithms. Since the results have shown to be equivalent for both the MIT and UC3M data sets, only the figures corresponding to the first one will be shown, to avoid redundancy.

The first mobility feature to focus on is the amount of movement, which, recalling the definitions in Section 4.1, roughly corresponds to the number of cell changes when translated to the symbolic domain. Figure 5.12 shows the relationship between the fraction



(a) Baseline.

(b) Representative (3,4).

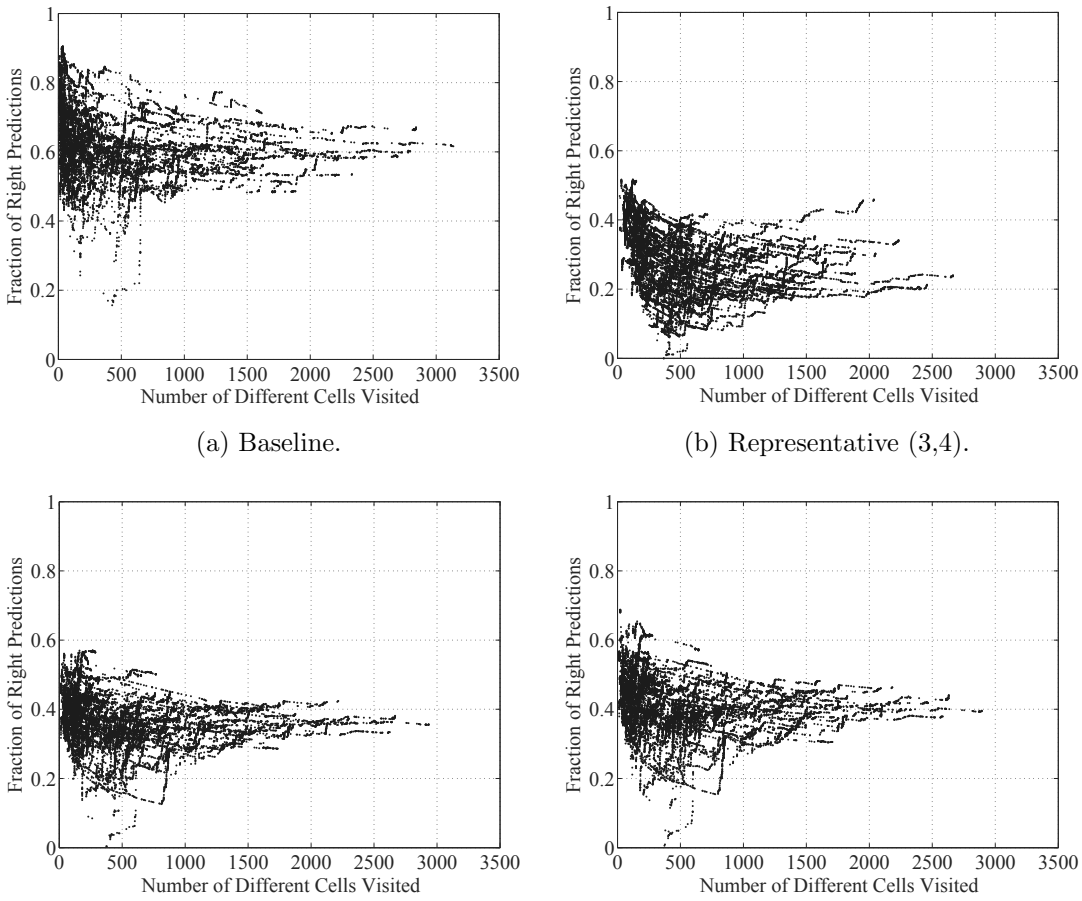
(c) Limits (3,4).

(d) Hybrid (3,4).

Figure 5.12: Fraction of right predictions as a function of the number of cell changes, when using the original Active LeZi algorithm and considering the baseline traces as well as the traces filtered with the three filtering techniques relying on the (3,4) detection scheme.

of correct predictions and the number of cell changes recorded so far when each prediction is calculated. Due to the high number of total cell changes of the movement history of each user and in order to reduce the volume of data shown in the figure, 1,000 samples have been randomly selected for each user, excluding the first 500 ones to avoid the learning phase of the algorithms that may mislead the results. No significant relationship can be seen in any of the cases—baseline, and traces filtered using representative, limits and hybrid techniques with the (3,4) ping pong detection scheme,— where a wide range of possible prediction accuracies are equally likely to happen for all the span of number of cell changes.

The same lack of relationship can be observed when considering the number of different cells visited with respect to the fraction of right predictions. However in this case, this



(a) Baseline.

(b) Representative (3,4).

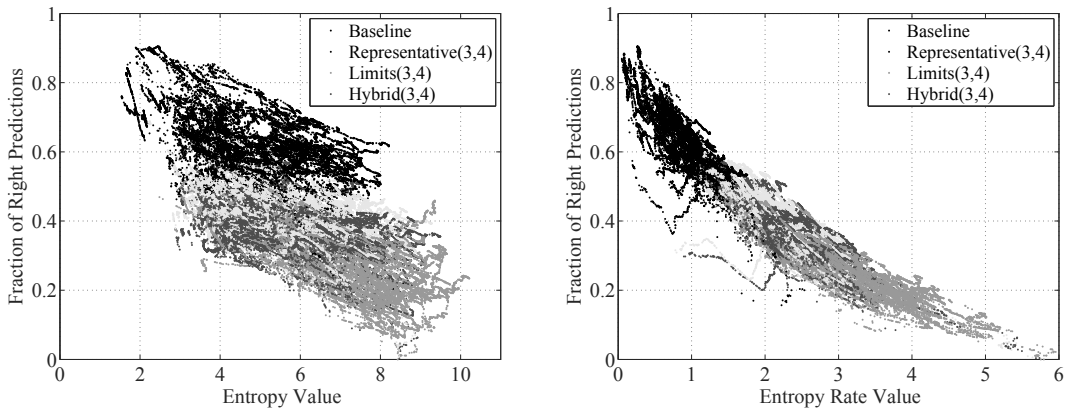
(c) Limits (3,4).

(d) Hybrid (3,4).

Figure 5.13: Fraction of right predictions as a function of the number of different visited cells, when using the original Active LeZi algorithm and considering the baseline traces as well as the traces filtered with the three filtering techniques relying on the (3,4) detection scheme.

result is somehow surprising because the more different cells are known to be places visited by the user, the algorithms need to select the next location among a wider set of possible choices. This could suggest at first a decrease in the prediction accuracy as the number of different visited cells increases. However, taking a look at Figure 5.13 it can be checked that this theoretical effect does not take place. The fact that only a limited number of cells can be visited next considering the current cell the user's device is connected to eradicate this hypothesis, which would apply if all the locations would be equally probable to be visited given any current location, or if the algorithms would have no memory.

Regarding the entropy and entropy rate of the movement histories of the users, and recalling Section 4.2, there was a noticeable difference in their distribution, as well as their



(a) Entropy.

(b) Entropy rate.

Figure 5.14: Fraction of right predictions as a function of the entropy and entropy rate values, when using the original Active LeZi algorithm and considering the baseline traces as well as the traces filtered with the three filtering techniques relying on the (3,4) detection scheme.

value, being the entropy values higher than the entropy rate ones. However, when analyzing their relationship with the fraction of right predictions achieved, it can be seen that the initial differences go beyond the value. Figure 5.14 shows the relationship between the fraction of right predictions and entropy in subfigure 5.14a, and entropy rate in subfigure 5.14b. The results for the baseline and filtered traces are shown together in both cases, in different grey colors. As can be observed, even when both entropy concepts measure in some way the randomness of the user movements, the relationships with the prediction behavior are very different. In the case of entropy, for the same range of values, the fraction of right predictions varies greatly, and seems to depend on the data used: for the same entropy, the fraction of right predictions with the baseline data is higher than with the filtered traces. However, subfigure 5.14b clarifies that the dependency is not on the data itself, but on its entropy rate. Although the four data sets share some span of entropy values, the baseline data reflects a much lower entropy rate than the filtered traces, being almost disjoint sets to that respect, as shown in Figure 4.18. It seems to be precisely the entropy rate the one determining the fraction of right predictions, as can be seen in Figure 5.14b. The tendency is clearly decreasing, meaning that a higher entropy rate value (i.e., higher user mobility randomness) leads to a lower prediction accuracy, as can be expected. Therefore, the results seem to indicate that the key to improve the prediction behavior of the algorithms relies on reducing the entropy rate, being the rest of parameters not that important.

5.4 Prediction Improvement Proposals

After analyzing the interplay between the prediction results and the intrinsic mobility features of the users whose future movements are predicted, some conclusions on how

these mobility features impact prediction can be extracted. In the previous section it was shown that the feature with the deepest impact on prediction accuracy is the randomness of the user's movements, this is, her entropy rate. Thus, by focusing on this feature, it might be possible to improve the prediction results. In this section, some improvements based on the observations done in the previous sections to achieve better prediction results are proposed, to further evaluate.

5.4.1 Extended LeZi Algorithm

In the previous section it was concluded that the randomness of the user's mobility, reflected into the entropy rate of her trace, seems to be the decisive feature impacting the fraction of right predictions attained by any prediction algorithm. Thus, it seems reasonable to focus on this aspect to try to improve the amount of right predictions. In fact, a reasoning similar to this one was already followed in [12]. In this work, the authors discussed that the improvement on prediction accuracy achieved by the LeZi Update algorithm they proposed came from reducing the entropy rate enclosed in the mobility model represented by the LZU tree, with respect to the entropy rate shown by the original LZ tree. Therefore, it seems that by reducing the entropy rate enclosed by the mobility model built by the algorithm (i.e., the entropy rate enclosed in the corresponding tree), the prediction accuracy increases. Therefore, the first improvement proposal is precisely based on this observation: to design a tree updating scheme that allows to reduce the entropy rate of the mobility model represented by the resulting tree.

In order to design the new prediction algorithm, it is required to have a method that allows to quantify such entropy rate. In [12], the authors describe a method that is easy to apply for short traces, and thus small trees. But it becomes too complex when the trace length increases and the tree becomes more complex. Therefore, a new entropy rate estimator, easier to calculate at each step of long traces, and based on Grassberger's estimator described in Section 2.2, is proposed to evaluate the entropy rate enclosed in the trees built by the original algorithms and the newly proposed one, which will be described further on.

5.4.1.1 Quantifying the Entropy Rate of a LZ-based Mobility Model

As mentioned before, in order to reduce the entropy rate of the mobility model enclosed in the corresponding tree, the first problem to tackle is how to quantify such entropy rate. Recalling Section 2.2, the entropy rate of a finite symbol sequence can be estimated using Grassberger's estimator. Applied to the mobility model, L_n, that corresponds to the stationary stochastic process which entropy rate will be estimated, and the trace, l, corresponding to the specific finite time series of length N coming from a realization of such mobility model, expression (2.16) turns into:

$$\hat{\mathrm{H}}_R\left(L_n\right) = \mathrm{H}_R\left(l\right) = \left(\frac{1}{N}\sum_{i=2}^{N}\frac{\Lambda_i}{\log_2 i}\right)^{-1}$$

where Λ_i is the shortest substring starting at position i that has not previously appeared in the trace from position 1 to $i-1$.

Observing the working principles of the LZ-based algorithms, described in Section 5.1, their behavior is similar to Grassberger's estimator. The LZ-based algorithms divide the trace in patterns such that each substring is different in at least one symbol to some previous substring. This is equivalent to searching the shortest substring not seen before. In fact, in Section 5.1 it was already mentioned that the LZ-based algorithms dynamically compute the optimal order, k, (this is, the optimal Λ_i) at each step to minimize the entropy rate of the mobility model they build (represented by the corresponding tree). The question is how to quantitatively estimate this entropy rate.

Considering how Grassberger's estimator and the LZ-based algorithms look for the shortest pattern not seen before, there is one main difference: the sequential behavior. Grassberger's estimator considers all the known trace in order to compute each Λ_i. Therefore, if a new symbol is recorded into the trace, l, the estimator needs to recompute all Λ_i from beginning to end. On the other hand, the LZ-based algorithms compute the optimal order, k—the equivalent to Λ_i—at each step, independently of the next symbol that will be next recorded into l. Thus, when a new location is recorded, the LZ-based evaluate the new symbol (parsing a new subpattern or recognizing an already seen one), without recomputing the previously parsed patterns. Each new pattern is added to the corresponding tree, and depending on how the parsing is done, the tree will contain more or less patterns, as can be observed when comparing the LZ, LeZi Update and Active LeZi versions. The more patterns the tree contains, the more probable the different length substrings formed with the new recorded symbol are already in the three, and therefore, the more probable is to search for a longer substring not stored by the tree yet. The longer the substring finally parsed, the higher the Λ_i value, and thus, the lower the entropy rate. Thus, the LZ-based algorithms are a sequential approximation to Grassberger's approach. Considering this approximation, the challenge falls on finding a way in which the LZ-based algorithms can be used to actually estimate the entropy rate value reflected by the patterns each algorithm is able to parse. In order to do so, the new estimator will be based on Grassberger's formula, but adapting it to work sequentially with the patterns parsed by the LZ-based algorithms at each step, which are contained in the corresponding tree, instead of processing the whole trace at once.

First, in order to compute the entropy rate at each step of the trace, the initial formula is slightly modified. The authors of [92] proposed the concept of instantaneous entropy, which refers to the entropy rate, based on Grassberger's estimator, applied to each of position, i, of the trace. With this proposal, the idea is to calculate $\mathrm{H}_R^i(L_n)$ in an instantaneous way, as new symbols are recorded into l, using the following expression:

$$\mathrm{H}_R^i(L_n) = H_R^i(l) = \left(\frac{1}{i} \sum_{j=2}^{i} \frac{\Lambda_j^I}{\log_2 j} \right)^{-1}, \forall i \in [1, N] \tag{5.6}$$

where Λ_j^I (where I represents instantaneous) corresponds to the length of the shortest substring from $j-\Lambda_j^I+1$ to j that did not appear in the sequence from index 1 to $j-\Lambda_j^I$, for

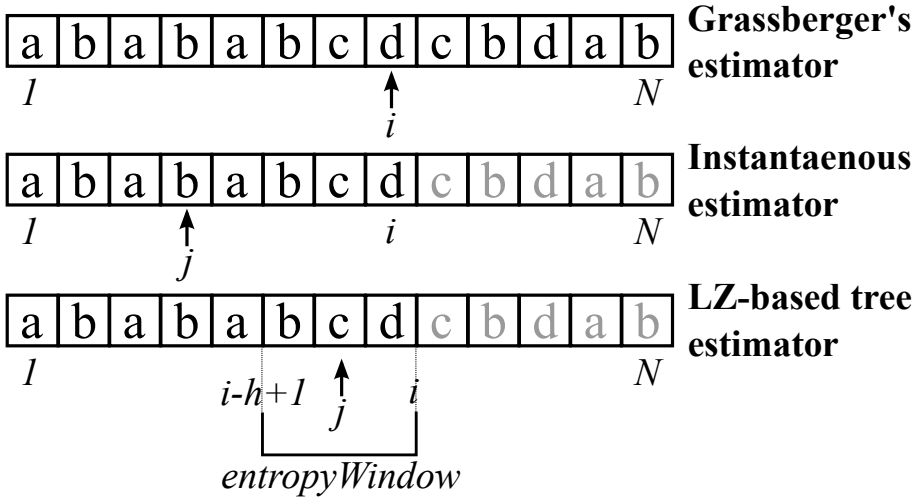


Figure 5.15: Elements of Λ calculation for each entropy estimator.

each sample of the trace. This way, only the samples from 1 to i are analyzed. Figure 5.15 shows the difference between Grassberger's and instantaneous entropy rate estimators.

Now, in order to actually consider entropy rate determined by the content of each tree, the general idea relies on using the tree built by each algorithm to estimate the entropy rate, in such a way that at each step, the elements of the summation of equation (2.16) will be determined by the length of the shortest substring not stored in the corresponding tree so far.

To complete the elements needed to apply the LZ tree entropy estimator, a sliding window, *entropyWindow*, is defined as shown in Figure 5.15. Its length, h, is equal to the longest substring parsed by the prediction algorithm we are considering (i.e., its tree height). Therefore, the $entropyWindow$ slides when the tree height does not change from one step to the next one (e.g., the window is ab at $i = 4$, and it slides to be ba at $i = 5$), whilst it is increased when the tree height changes (e.g., the window is ab at $i = 6$ and it increases to be abc at $i = 7$).

The proposed entropy rate estimator works as follows. At each step, Λ_j^{alg} is calculated, where $alg = \{LZ, LZU, ALZ\}$, for each positions j within the window ($j \in [i - h + 1, i]$). In order to calculate Λ_j^{alg}, the substring starting at position j up to position i, $l_{j,i}$, is looked up in the tree considered. Then, Λ_j^{alg} would be the longest substring matching $l_{j,i}$ plus one (which is the length of the shortest substring not stored yet).

The reason to recalculate Λ_j^{alg} of the positions j within the $entropyWindow$ (instead of just calculating Λ_j^{alg} for the last position, i) is that it might be the case in which Λ_j^{alg} is different before and after adding the next symbol to the trace. Taking as reference the example in Figure 5.15, and applying the Active LeZi algorithm, the ALZ tree at step $i = 13$ would be the one in Figure 5.4. Thus, at the last step, $i = 13$, $entropyWindow = dab$, and $\Lambda_{11}^{ALZ} = 2$, $\Lambda_{12}^{ALZ} = 3$ and $\Lambda_{13}^{ALZ} = 2$. Then, two cases can happen:

- If a new symbol, $l_{14} = c$, is detected, then: $entropyWindow = abc$, $\Lambda_{12}^{ALZ} = 4$,

$\Lambda_{13}^{ALZ} = 3$ and $\Lambda_{14}^{ALZ} = 2$.

- If the new symbol is $l_{14} = e$, then: $entropyWindow = abe$, $\Lambda_{12}^{ALZ} = 3$, $\Lambda_{13}^{ALZ} = 2$ and $\Lambda_{14}^{ALZ} = 1$.

Therefore, if the first estimates of Λ_{12}^{ALZ} or Λ_{13}^{ALZ} are not recomputed, they might be not accurate, depending on the next symbol that will be recorded in the sequence. The most accurate Λ_j considering the corresponding tree at each step is obtained when $j = i - h + 1$, i.e., for the first position of the $entropyWindow$. The estimator could only get the highest possible value of Λ_j in this case, and it is guaranteed that, considering the corresponding tree at that step, it is not possible to find any longer pattern with the information up to that moment. Therefore, Λ_j at the positions j older than the ones within the $entropyWindow$ will not change.

Since the values of Λ_j^{alg} for $j \in [i-h+1, i]$ (i.e., the positions within the $entropyWindow$) might change, the summation in expression 5.6 is split into two parts:

- A $finalSum$, which is the sum of all the terms Λ_j^{alg} outside the $entropyWindow$ (i.e., $j \in [2, i-h]$). This $finalSum$ is updated every time the $windowEntropy$ slides through the trace with the value of Λ_{i-h}^{alg}, which is the last value Λ^{alg} which will not be recalculated anymore.

- A $nonFinalSum$, which sums all the terms Λ_j^{alg} within the $entropyWindow$ (i.e., $j \in [i-h+1, i]$).

Algorithm 1 describes how the LZ tree entropy rate estimator works when each new symbol, l_i, is recorded into the trace, l

Since with this approach is based on the substrings parsed by the LZ-based algorithm chosen, this entropy rate estimator has the same drawback than the algorithms: not all the patterns are detected due to the increasing length of the considered patterns and the sequential behavior. Thus, when the $entropyWindow$ length is increased, the previous patterns of such length are neglected. This means that the trace is not revisited, which contrasts with Grassberger's estimator behavior that continuously revisit the whole trace to calculate each Λ_i. This results in existing patterns not parsed by the LZ-based algorithms that might impact the estimation of Λ_i. The entropy rate estimation would be better as the corresponding LZ-based algorithm is able to detect longer mobility patterns. In next sections, a comparative of the estimation done with each LZ-based algorithm with respect to the results obtained using Grassberger's approach will be presented.

5.4.1.2 Extended LeZi Algorithm Proposal

Once being able to estimate the entropy rate enclosed in the mobility model stored in the tree generated by each LZ-based algorithm, the next step is to design a new tree updating scheme, which will be called Extended LeZi, capable of detecting more existing patterns, thus reducing the entropy rate of the resulting model. As seen in previous sections, this could potentially improve the fraction of correct mobility predictions done by the algorithm.

Algorithm 1 Entropy Rate Estimation of a LZ-based tree when processing trace l

Input: $l = l_1 l_2 \dots l_i \dots l_N$
Input: alg={LZ, LeZi Update, Active LeZi}
Output: $entropyRateValues[]$

1: $i \leftarrow 1$
2: $tree \leftarrow \gamma$
3: $\Lambda[*] \leftarrow 0$
4: $h \leftarrow 0$
5: $entropyRateValues[*] \leftarrow 0$
6: **for** $i = 2$ to N **do**
7: $newSymbol \leftarrow l_i$
{1. UPDATE $entropyWindow$}
8: **if** h==($entropyWindow$ length) **then**
9: $finalSum \leftarrow finalSum + \Lambda[i-h]/\log_2(i-h)$
10: $entropyWindow$=$l_{i-h+1,i}$
11: **else**
12: increase $entropyWindow$ length in 1
13: $entropyWindow$=$l_{i-h,i}$
{2. CALCULATE $\Lambda_j, \forall j \in [i-h+1, i]$}
14: $j \leftarrow i-h+1$
15: **while** $j <= i$ **do**
16: $\Lambda[j] \leftarrow 1+$ length of longest string starting at position j of $entropyWindow$ found in $tree$
17: $nonFinalSum \leftarrow nonFinalSum + \Lambda[j]/\log_2(j)$
18: $j \leftarrow j+1$
{3. UPDATE TREE}
19: $tree \leftarrow$ update $tree$ with $newSymbol$ applying alg
20: $h \leftarrow$ ($tree$ height)
{4. CALCULATE $H_R^i(l)$}
21: $entropyRateValues[i] \leftarrow i/(finalSum + nonFinalSum)$
22: $i \leftarrow i+1$
23: **return** $entropyRateValues[]$

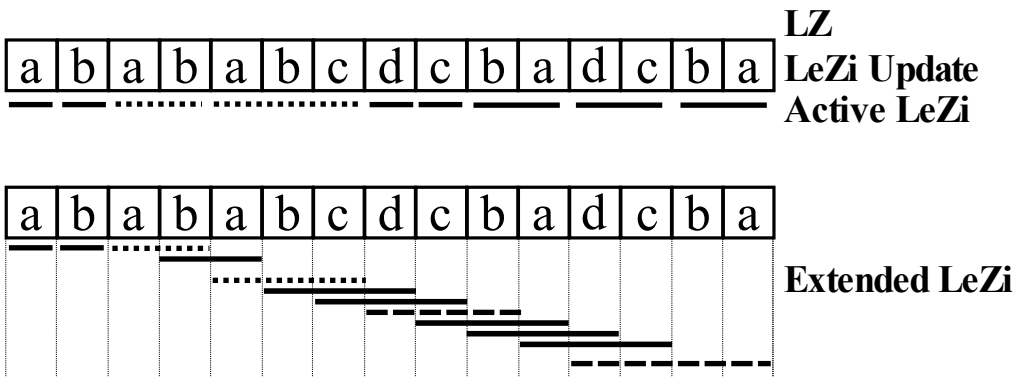


Figure 5.16: Comparison of maximum length calculation, k, for the four LZ-based algorithms.

In order to propose this new scheme, it is precise to carefully examine how the updating tree scheme of the three LZ-based algorithms existing so far—LZ, LeZi Update, and Active LeZi—work. Going back to Section 5.1, although each LZ-based prediction algorithm has its own peculiarities, the maximum length of the parsed substring at each step is determined in the same way: given the sequence of locations, l, the LZ-based algorithms split it into substrings, $s_0 \dots s_m$ such that $s_0 = \gamma$, being γ the empty string, and $s_i \neq s_j$, $\forall j \geq 1$ and $i < j$. In other words, every substring s_j is different in at least one symbol with respect to all the previous parsed s_i.

As explained in the previous section, this behavior is parallel to that of Grassberger's entropy estimator. The main difference stems from the sequential operation of the LZ-based algorithms. That operation mode leads to parse the trace in subsequent, not overlapping, substrings. That means that, whereas Grassberger's approach evaluates the longest substring, Λ_i, for every single position of the trace, $i \in [1, N]$, the LZ-based approaches only evaluates Λ_i at the starting symbol of each parsed pattern, s_i. The rest of the symbols composing every s_i, that could potentially be the starting points of longer substrings to parse, are neglected. Consequently, the entropy rate value might be overestimated, since longer patterns are neglected, which leads to Λ_i values lower than possible. The Extended LeZi algorithms aims at overcoming this limitation of the behavior of the LZ-based prediction algorithms, by releasing the constraint set by the original LZ-based algorithms, as shown in Figure 5.16. As explained, the three original LZ-based algorithms parse the trace in subsequent and not overlapping substrings. Thus, only patterns up to three symbols are detected, as shown in dotted lines. However, the Extended LeZi scheme neglects this limitation, and evaluates the longest substring not seen before at every position of the trace. This way, patterns up to 4 symbols are detected, as the dashed lines show.

The idea is to mimic the behavior shown by Grassberger's approach, without fully loosing the best properties of the LZ-based approaches: their sequential behavior, and the compact storage and fast look-up process provided by the use of tree structures to store the parsed patterns. In order to do so, a sliding window of variable size, w, is used. This window will always have the length of the longest substring parsed so far, k. For each

position delimited by the window, the algorithm will check the longest substring, starting at that position up to the end of the window, already parsed, and thus, stored in the corresponding tree.

Let l be the mobility history or trace, $l = l_1 l_2 \dots l_N$, and w a window of variable length, k, such that $w = l_{j,j+k}$, where $l_{j,j+k}$ represents the subsequence of symbols in l from position j to $j + k$. The initial value of w is the empty string, γ, and thus the initial value of k is 1. The subpatterns enclosed in the window correspond to every substring from position j up to position $j + k$: $\{l_{m,j+k}\}, \forall m \in [j, j + k]$. For instance, if $w = abcd$, then the subpatterns contained in w are $\{abcd, bcd, cd, d\}$

A tree data structure is also defined, that will be referred to as ELZ tree from now on. This tree will hold the patterns parsed by the Extended LeZi scheme, as the equivalent trees do in the LZ, LeZi Update and Active LeZi algorithms.

When a new symbol is recorded into the movement history, l, the window w slides to include the new symbol. Then, for each subpattern enclosed in the window, the algorithm checks if it was already parsed, i.e., if it is in the ELZ tree. If the subpattern is not in the tree, it is added; otherwise, the frequency of the tree node representing such subpattern is increased. If the longest subpattern in the window (i.e., the whole sequence of symbols enclosed by the window) is already in the tree, it means that the length of the longest substring not parsed yet is greater than the window length, meaning that k increases. In such situation, after updating the corresponding tree node, the length of the window is also incremented in one, so that it remains equal to k. Incrementing the window length allows to capture this new longest subpattern just detected. Thus, when the next symbol is recorded, the window will not slide but just enclose also that new symbol.

Following the previous description, it is possible to parse the longest substring starting at position j that did not appear from position 1 to $j + k$, $\forall j \in [i - k + 1, i]$, being i the number of symbols recorded in l so far. However, recalling the original algorithms, the goal is to find the longest substring that did appear form 1 to $j - 1$. For instance, in Figure 5.17, the algorithm is parsing at position $i = 9$. The window at that point is $w = aba$, where $j = 7$ and $k = 3$. In the past, the substring aba was already parsed in step $i = 7$, and is thus stored in the ELZ tree. Therefore, when the algorithm looks-up for the first substring of the current window, aba, it is going to find it in the tree, thus making k to increase. However, it must be noticed that the substring aba already parsed corresponds to positions $[5, 6, 7]$, whereas the substring being currently considered corresponds to position $[7, 8, 9]$. Thus, the symbol at position 7 creates an overlap. Following the principles of the LZ-based algorithms, the new substrings to be parsed, in this case, the ones starting at position $j = 7$, have to be compared with those appearing from position 0 up to $j-1$, which in this case is $j - 1 = 6$. Therefore, in this case k should not increase its value, because the longest substring starting at position 7 which have already appeared from position 0 to 6 is size 2, ab, and thus the shortest substring not seen for this case is $k = 3$.

In order to comply with this constraint, for each position of the window, there is an associated list of substrings. The list, referred to as pending substrings list, corresponding to each position of the window contains all the substrings parsed by the first time (i.e., just added to the tree) which last symbol is the one in the position of the list. In other words, when the substring starting at position j is found in the ELZ tree, it is also searched in

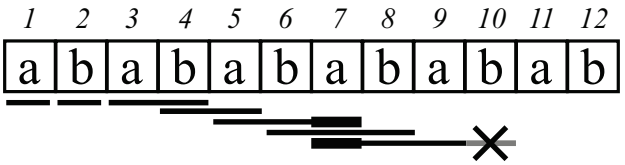


Figure 5.17: Overlap problem of the Extended LeZi algorithm.

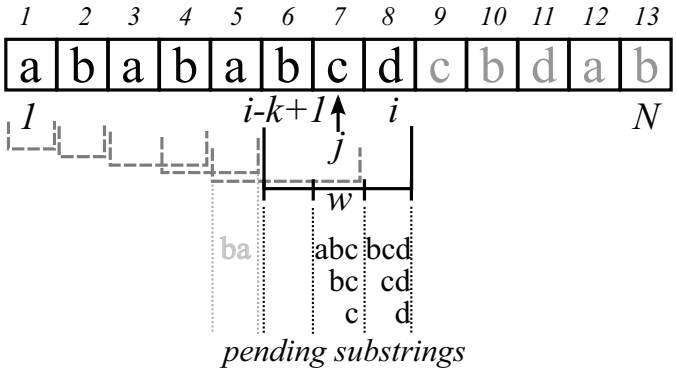


Figure 5.18: Elements involved in the Extended LeZi scheme.

the pending lists of positions j to $j+k$. All the substrings stored in these lists appeared from position 1 to $j+k$, $\forall j \in [i-k+1, i]$, and thus they must be considered as not parsed yet, even if the algorithm finds them in the tree. This way, the final result resembles to comparing the current substring with respect to those appearing from position 1 to $j-1$. As the window slides or increases its size when new symbols are recorded into l, the pending substrings lists update the same way: if the window slides, the list corresponding to the first position of the window is removed and a new one is created, corresponding to the new last position; when the window grows, a new list is added at the end.

Figure 5.18 shows the working principles of the Extended LeZi algorithm. It can be shown how the window, w, which goes from position $i-k+1$ to i, slides or grows depending on the case, as explained before. It has a set of lists storing the pending substrings. At the previous step from the current one $i = 10$, the window, comprising positions from 5 to 7, have one list at position 5 with the string *ba*. Then, when the window slides to positions 6 to 8, that previous list is deleted (the substring *ba* can be considered as stored in the tree when looking for new substrings to parse). Position 7 has an associated list containing substrings *abc*, *bc*, and *c*, which were parsed for the first time in the previous step. Position 8 has also an associated list containing the substrings *bcd*, *cd*, and *d*, which were parsed for the first time in the current step. Therefore, when the window keeps on sliding, if the algorithm finds in the new window any of these substrings, they will not be recognized as parsed before, even if they are stored in the ELZ tree, since they did not appear from position 1 to $j-1$. Algorithm 2 summarizes the behavior of the algorithm.

Algorithm 2 Extended LeZi

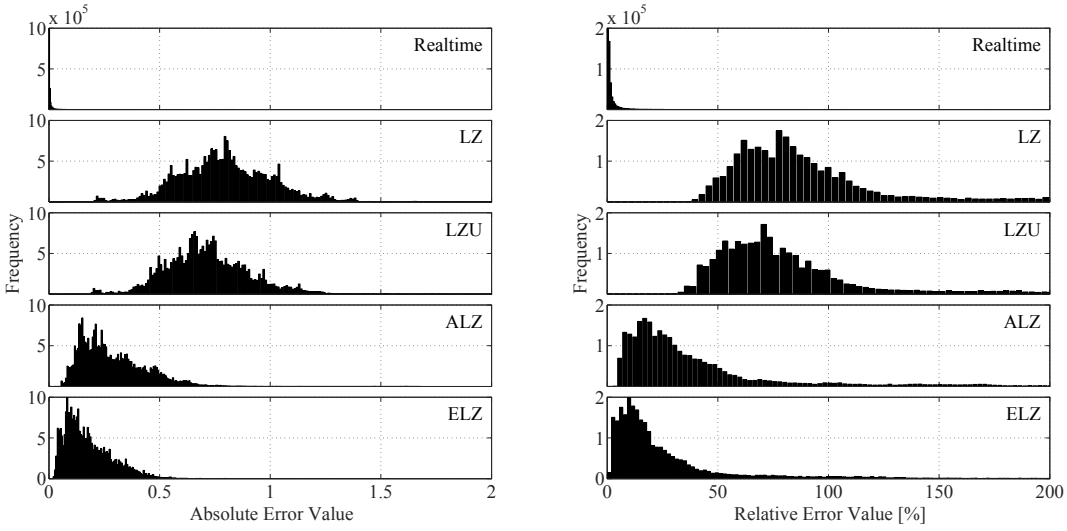
Input: $l = l_1 l_2 \dots l_i \dots l_N$
Output: $ELZtree$

$ELZtree \leftarrow \gamma$
$w \leftarrow \gamma$
$k \leftarrow 1$
$pendingNodesLists[*] \leftarrow \gamma$
for $i = 1$ to N **do**
  $newSymbol \leftarrow l_i$
  {1. UPDATE w and $pendingNodesLists[]$}
  if k==(w length) **then**
    $w \leftarrow l_{i-k+1,i}$
    slide $pendingNodesLists$
  else
    $w \leftarrow l_{i-k,i}$
    increase $pendingNodesLists$ length in 1
  {2. UPDATE ELZ TREE}
  $j \leftarrow 1$
  while $j < k$ **do**
    $startingIndex \leftarrow i - k + j$
    if $l_{startingIndex,i}$ is not in ELZ tree OR is in $pendingNodesLists[j-k]$ **then**
      if $l_{startingIndex,i}$ is not in ELZ tree **then**
        add $l_{startingIndex,i}$ to ELZ tree
        add $l_{startingIndex,i}$ to $pendingNodesLists[j]$
      else
        increase frequency of ELZ tree node corresponding to $l_{startingIndex,i}$
    else
      increase frequency of ELZ tree node corresponding to $l_{startingIndex,i}$
      if $j == 1$ **then**
        $k \leftarrow k + 1$
    $j \leftarrow j + 1$
  $i \leftarrow i + 1$
return $ELZtree$

5.4.1.3 Evaluation of the Extended LeZi Algorithm

In order to assess the Extended LeZi algorithm, the first step is to evaluate the decrease of the entropy rate enclosed by the mobility model represented by the ELZ tree, with respect to the previous proposals, ALZ, LZU and LZ. Using the entropy rate estimation procedure previously described, the entropy rate of the mobility models stored in the trees generated by these four tree updating schemes is calculated for the MIT data set. Then, the difference between the entropy rate value estimated by each of these four LZ-based algorithms and that estimated using Grassberger's approach is shown in Figure 5.19. As reference, it is also shown the error with respect to the instantaneous entropy estimator. Each of the subfigures represents the distribution of the absolute and relative errors, considering the Grassberger estimation as the real value and normalization factor. The estimations are calculated for each sample of all the movement histories.

As can be observed, the distribution of the error in the ELZ case is shifted towards lower values, although still far away from the error values obtained by the instantaneous estimator. Thus, although the ELZ tree captures more information that allows to reduce the entropy rate enclosed by the mobility model built, it is far away from the optimal value. Table 5.5 provides the statistical values of the error distribution for each case. It shows a decrease of a 15% in average of the ELZ approach with respect to the ALZ one, which was the LZ-based approach yielding the lower error up until now.



(a) Absolute error. (b) Relative error.

Figure 5.19: Comparison of the absolute and relative error distribution of the entropy estimation achieved by using each of the LZ-based algorithms, with respect to the Grassberger estimator.

Considering these results in the entropy rate estimation, an improvement in the fraction of right predictions achieved by the ELZ algorithm could be expected, based on the results

Algorithm	Max	Min	Mean	Median	Var
Realtime	0.843	0	0.004 (0.63%)	0.001	0.00019
LZ	2.534	0	0.800 (97.91%)	0.790	0.04788
LZU	2.366	0	0.716 (88.74%)	0.701	0.03855
ALZ	1.679	0	0.290 (43.22%)	0.252	0.02449
ELZ	1.539	0	0.180 (28.30%)	0.150	0.01500

Table 5.5: Summary of the main statistics related to the distribution of the entropy estimation error achieved by using each of the LZ-based algorithms, with respect to the Grassberger estimator.

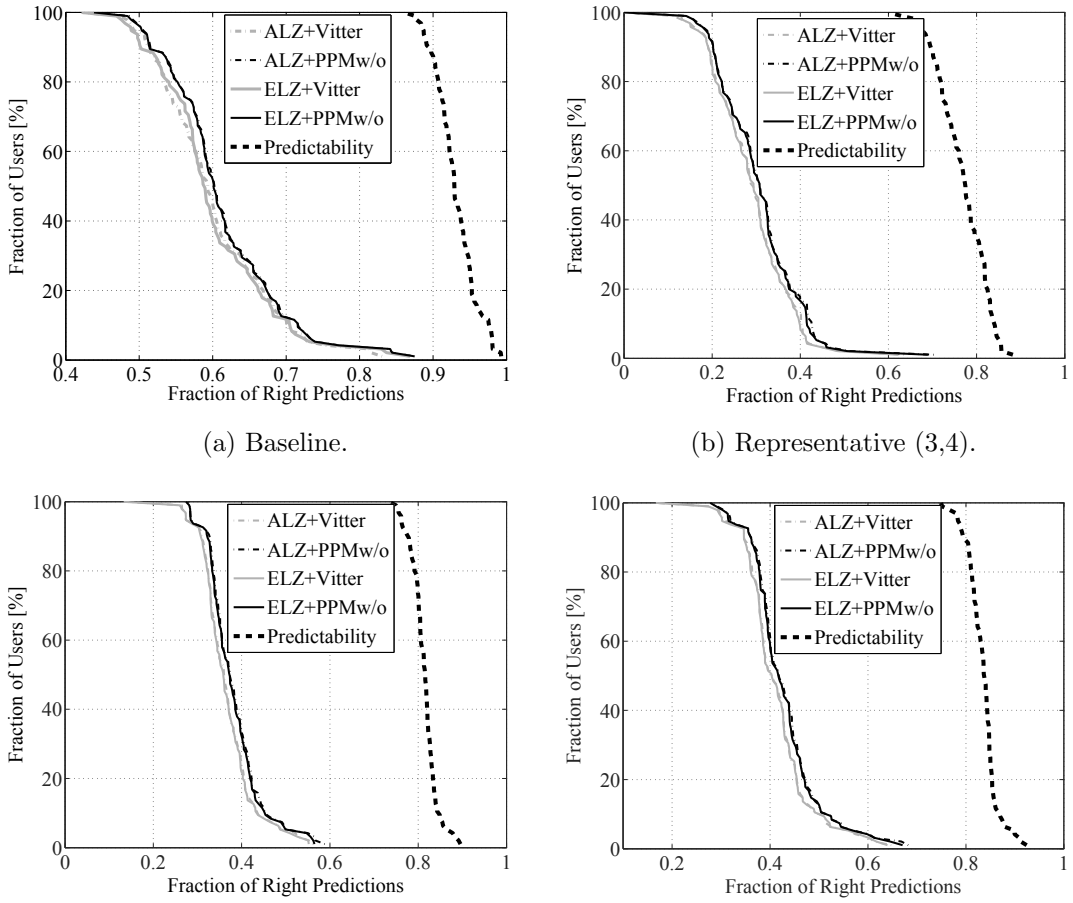
shown in Figure 5.14b. When evaluating the original algorithms it was shown that the LeZi Update algorithm achieved a better prediction accuracy than the LZ approach, matching the decrease in the entropy rate estimation error. The same happens when considering the Active LeZi algorithm, which improvement is even higher with respect to the two other original approaches, as it is also the difference between its entropy rate estimation error with respect to that achieved by the LZ and LZU choices.

Figure 5.20 shows the comparison of the prediction accuracy achieved by the ALZ and ELZ trees combined with the Vitter and PPM without exclusion technique, for the baseline case and the three filtering techniques combined with the $(3, 4)$ ping pong detection scheme. Surprisingly, the plots show that the improvement in the entropy rate estimation, and thus, the increase in the number and length of the patterns stored by the ELZ tree, did not yield a prediction accuracy improvement for any of the for traces considered. This fact suggests that it is not the entropy rate enclosed in the mobility model built by the algorithms which increases the fraction of right predictions, but the entropy rate of the mobility model itself, depending on the user's behavior. The tree updating schemes can increase the number of correct predictions by storing more and longer patterns, but only until certain point in which an improvement on their capacity to better estimate the entropy rate does not yield any further improvement in the prediction task. The Active LeZi scheme already reaches that saturation point, and thus, the better entropy rate estimation done by the Extended LeZi version does not impact the prediction results.

5.4.2 Probability Calculation Improvement Proposals

In the previous improvement proposal it was checked that lowering the entropy rate of the mobility model enclosed in the tree built by the algorithms, does not lead to a higher prediction accuracy indefinitely. This fact seems to indicate that the prediction accuracy improvement is not about detecting and storing longer mobility patterns of the user. Therefore, the next improvement proposal will be focused on how the pattern information already stored can be better used to increment the number of correctly predicted next locations.

To that purpose, tt must be noticed that as the length of the contexts increase, the number of times it can be detected in the movement history, and thus stored in the tree,



(a) Baseline.

(b) Representative (3,4).

(c) Limits (3,4).

(d) Hybrid (3,4).

Figure 5.20: Fraction of users in the MIT data set attaining, at least, the corresponding fraction of right predictions (or less), when considering the baseline case, and the three filtering techniques combined with the ping pong detection scheme (3,4), and applying the original Active LeZi and Extended LeZi schemes, combined with the PPM without exclusion algorithm.

decreases. The first reason is that the longer the context considered, the less number of blocks of the length of the context can be extracted from the trace. Second, given an alphabet with cardinality C, the number of different possible blocks of size n is given by C^n. Thus, as n increases, the number of different possible blocks increases as well, and there would be less samples of each possible block. Third, the LZ-based algorithms start with context length equal to 1, and it increases the movement history is parsed. Then, the shortest contexts have more samples, since the algorithms start to parse longer substrings when many symbols in the trace have been already analyzed.

For these reasons, the number of samples of the longest contexts stored in the corre-

sponding trees are very low. Taking into account that Vitter and PPM without exclusion assign the highest weight to those longest context (or consider only them in the case of Vitter technique), it is possible that the predictions are impacted by the low number of samples available for the probability calculation. Thus, the proposal for the probability calculation methods is to consider the contexts that have a minimum number of samples.

Figure 5.21 shows the results of applying this proposal to the Vitter technique and different values of the minimum number of samples required to take the prediction context into account. Only 3 values are represented since they are the ones providing the most significant results. It can be observed that for the baseline case, there is a noticeable difference when using the original Vitter technique with respect to using it but taking into account contexts with frequency of at least 10 samples. This last case improves the prediction accuracy for the whole population of the data set. When increasing that number to 20, there is still some improvement in the results, but lower than the ones obtained with the value set to 10. For the case of the traces filtered with the representative technique, the differences are very small and differ depending on the specific user, whereas for the traces filtered with the limits or hybrid techniques, the results are also better using the modified version of the Vitter technique.

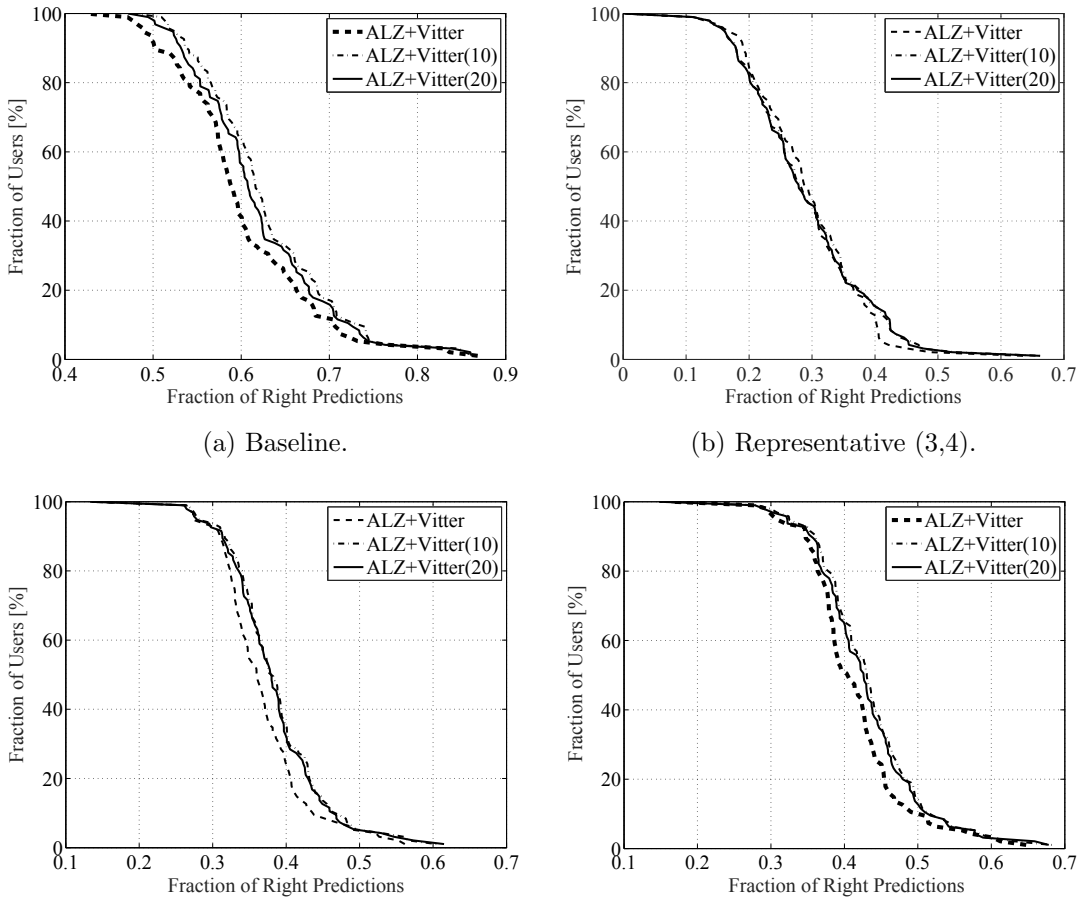
The same procedure is applied to the PPM without exclusion technique. However, in this case the results are quite different. For the baseline traces, the accuracy obtained both with the original and the modified version are very similar, but when the modified version is applied to the filtered traces, the prediction accuracy drops down to a 10%. This effect can be due to the fact that PPM without exclusion considers not only the longest context, but all the contexts from the longest to the shortest one. Therefore, if the longest context has a low number of samples, the potential poor probability estimation is corrected by the estimations coming from the shorter contexts, whereas if the probability estimation is right, it is leveraged to provide more accurate predictions.

Finally, comparing the results of the best Vitter and PPM without exclusion approaches, the prediction results are the ones shown in Figure 5.23. Surprisingly, using the modified version of the Vitter technique, the results outperform those obtained with the PPM without exclusion in the baseline case and also in the traces filtered with the limits and hybrid filtering techniques. In the case of the traces filtered with the representative technique, PPM without exclusion keeps on being the best option, although the results obtained from the Vitter case are very close.

5.5 Conclusions

Despite the wide variety of location prediction algorithms described in Section 2.1.4, the focus on the specific family of LZ-based prediction algorithms allowed to study them from different perspectives, trying to better understand their working principles and improve their results.

The first proposal was to divide the algorithms into two independent phases, namely the tree updating scheme, in charge of detecting and learning the patterns conforming the mobility model of the user, and the probability calculating technique, which uses the



(a) Baseline.

(b) Representative (3,4).

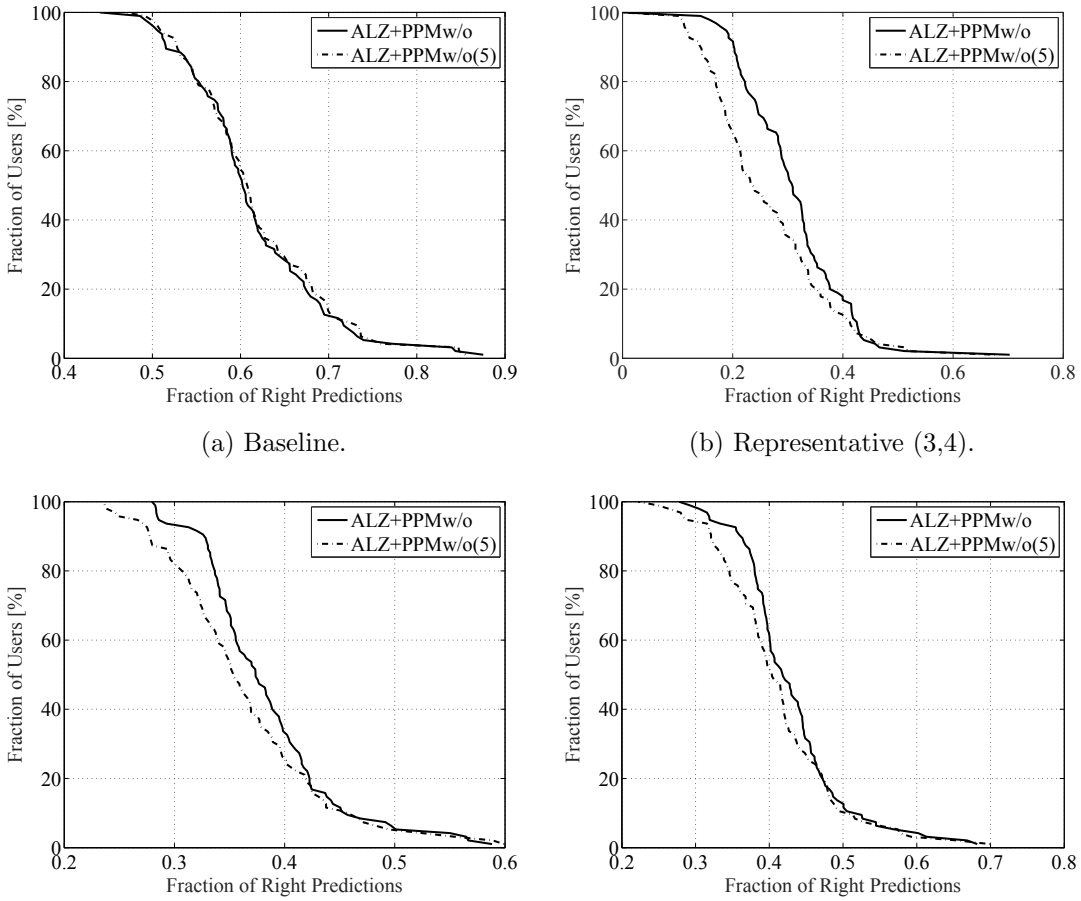
(c) Limits (3,4).

(d) Hybrid (3,4).

Figure 5.21: Fraction of users in the MIT data set attaining, at least, the corresponding fraction of right predictions (or less), when considering the baseline traces as well as the traces filtered by the three filtering techniques combined with the ping pong detection scheme (3,4), using different depths of the Vitter method.

information of the mobility model updated in the previous phase to estimate the most probable next symbol. By evaluating all the possible combinations of these two phases, the results pointed to the probability calculation technique as the main factor in obtaining higher fractions of right predictions.

One of the more striking results was derived from considering the ping pong sequences of the movement histories, already studied in Chapter 4. Evaluating the predictions of real locations (i.e., leaving out the symbols belonging to ping pong sequences), the fraction of right predictions drops from at least 60% for 50% of the users, to a 20% for 50% of the users. The network-related effect disclosed in Chapter 4 has a huge impact on the prediction process. The traces filtered with the three filtering techniques described in such



(a) Baseline.

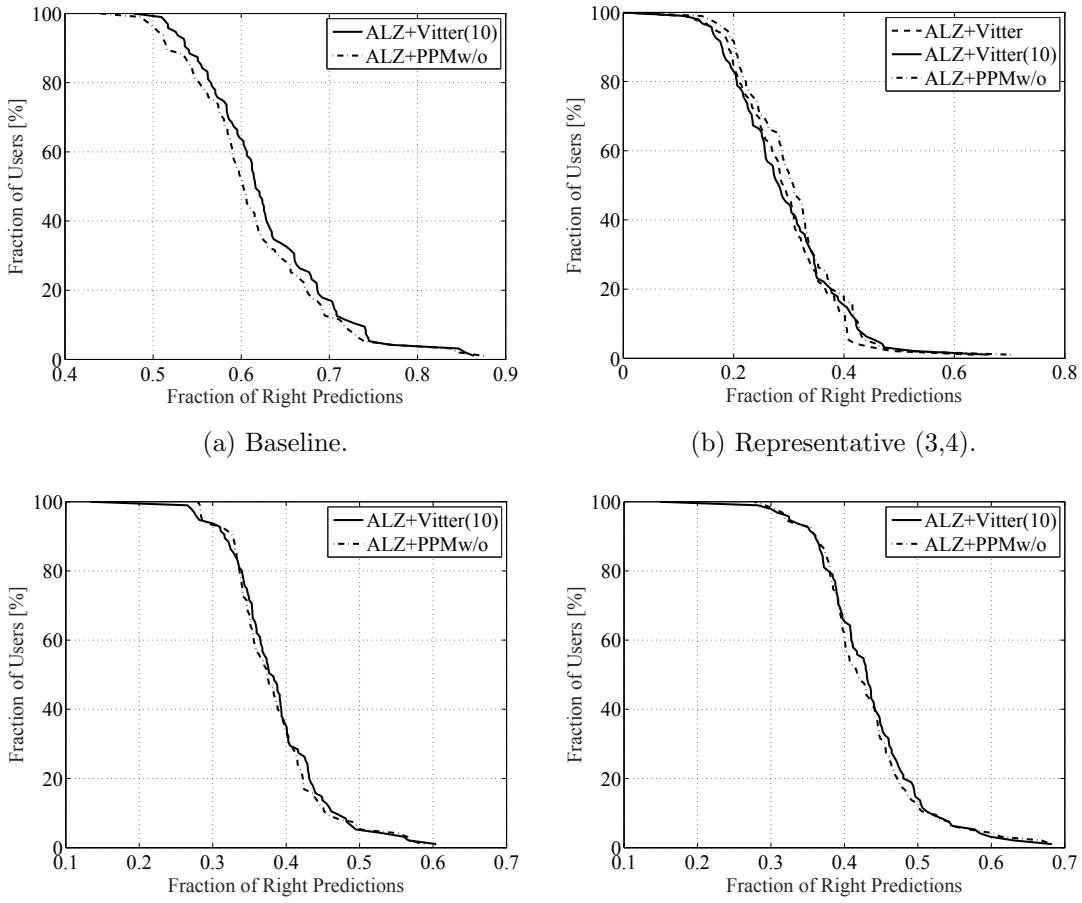
(b) Representative (3,4).

(c) Limits (3,4).

(d) Hybrid (3,4).

Figure 5.22: Fraction of users in the MIT data set attaining, at least, the corresponding fraction of right predictions (or less), when considering the baseline traces and the traces filtered with the three filtering techniques combined with the ping pong detection scheme (3,4), using different depths of the PPM without exclusion method.

chapter were also processed with all the combinations of the prediction phases enumerated above, reaching a percentage of right predictions of at least 30%-40% for 50% of the users, depending on the filtering technique used. The representative technique leads to the worst results, since the resulting filtered traces have no block repetition at all. The limits and hybrid techniques, despite filtering out the great part of the ping pong sequences, introduce blocks with certain fixed structure (the limits of the ping pong sequence, or the limits plus the representative symbol), which leads to an increased prediction accuracy. These ping pong sequences lead also to the delusion of simpler predictive models, like Markov models of order 2, outperforming the results obtained by the LZ algorithms. However, as soon as the traces are filtered, the results coming from the prediction based on Markov models



(a) Baseline.

(b) Representative (3,4).

(c) Limits (3,4).

(d) Hybrid (3,4).

Figure 5.23: Fraction of users in the MIT data set attaining, at least, the corresponding fraction of right predictions (or less), when considering the baseline traces and the traces filtered with the three filtering techniques combined with the ping pong detection scheme (3,4), comparing the best Vitter and PPM without exclusion approaches.

show to achieve a lower fraction of right predictions than the combination of Active LeZi and PPM without exclusion algorithm. Again, due to the fixed structure highly repeated of the ping pong sequences, Markov models learn those sequences and perfectly predict what symbol comes next. However, as soon as those sequences disappear, these models fall short to capture the longer mobility patterns of the users.

Another of the problems derived from using the cellular network as a proxy for the individuals mobility is that their mobile phones can connect to either of two or three BTSs even being at the same concrete location. When the applications where the next location predictions are going to be applied are not critical, using the two most probable next locations as predictions has shown to be an effective way to greatly increase the fraction

of right predictions, whilst the use of the third more probable locations is not noticeable.

The analysis of the relationship between the prediction accuracy of the original algorithms and the mobility features showed that entropy rate (i.e., the randomness of the user mobility behavior) is undoubtedly the feature impacting the most in the prediction success. For this reason, the first prediction improvement was aimed at lowering the entropy rate of the mobility model enclosed in the tree built by the corresponding algorithm. For that purpose, an algorithm to sequentially calculate the entropy rate of the tree as it is updated was proposed. This algorithm helped to estimate the entropy rate of the trees built by the LZ, LeZi Update and Active LeZi, and compare it to that of the tree built by a newly proposed algorithm, called Extended LeZi. The entropy rate of the ELZ tree showed to be a 15% lower than the entropy rate of the ALZ tree. With this result, it was expected an improvement of the prediction results of the Extended LeZi algorithm combined with some of the probability calculation techniques. However, comparing the results of the Extended LeZi algorithm with those of the Active LeZi one, they are practically the same. Therefore, it can be concluded that an increase in the information stored by the tree beyond the patterns stored by the Active LeZi algorithm does not improve the prediction accuracy.

Since the main determining factor to increase the prediction success seemed to be the probability calculation technique, the next proposal was focused on improving the algorithms of this phase. Under the observation of the decreasing number of samples of the longest patters stored in the corresponding trees, the Vitter and PPM without exclusion techniques were modified so that they start taking into account the longest patterns with a frequency higher than certain threshold. Trying different values of the threshold, it was found out that the modified version of Vitter obtained the best results with a value of 10 samples, whereas PPM without exclusion does not provide better results with any value of the threshold. By comparing the best options of both techniques, the modified version of Vitter with threshold equal to 10 showed an improvement in the baseline traces and those filtered with the limits and hybrid techniques, whereas the results for the traces filtered with the representative technique are very close to the original PPM without exclusion, but being a much faster approach.

Chapter 6

Contributions to Privacy Metrics in Human Mobility Scenarios

Contents

The previous chapters analyze human mobility through various stages, aiming at understanding individual dynamics to improve the predictions of future locations the individual will visit. Besides this application, the study of human movement can be applied to several others such as understanding the spread of infectious viruses, the urban dynamics of a city, the behavior of wireless networks, among others, as mentioned in Section 1.1. However, it would be naive to consider just this perspective, without noticing that information about human mobility can lead to not so honest purposes. When using location-based services (LBSs) in mobile phones—such as weather, traffic, or news widgets—the user's phone sends, quite frequently, a service request together with the user location, aiming to obtain the most up to date service information associated to that location. Therefore,

the LBS provider might end up with a location history made up of the sequence of locations attached to the service requests. This location history is not very different from the ones considered in Section 3.1: the sampling rate would be, in general, lower than the baseline case but, for some users, it might be higher than the CDR case. As discussed in Section 2.1.3, an extensive share of the research on human mobility use these CDR-based location histories, since they carry enough mobility data so as to, for instance, infer the user work or home locations. But the implicit information contained in the collected locations may reach beyond, unveiling details such as if the individual has children (the number of visits to a kindergarten or school is high), if she may suffer from some chronic disease (the number of visits to a hospital is high), if she travels much (there are visits to locations located in many different countries), among others. Therefore, if the location history obtained by collecting the locations attached to the service requests might be even more complete, the LBS provider ends up knowing much mobility-related features about the user, without any guarantee on how this information could be used (or to whom it may be disclosed).

In general, many LBSs, and other services using the user location as data source, build upon the creation of user profiles, which combined across several information services pose evident privacy and security risks. On the other hand, as exposed in the previous chapters, it is precisely the availability to a system of such sensitive information what enables such intelligent functionality. Therefore, the need for preserving privacy without compromising the utility of the information emerges naturally. The existence of this inherent compromise is a strong motivation to develop quantifiable metrics of privacy and utility, and to design practical privacy-enhancing, data-perturbative mechanisms achieving serviceable points of operation in this privacy-utility trade-off.

This chapter describes a new privacy metric based on one of the mobility features that have demonstrated to be key in the study of human mobility: the entropy rate. There will follow an study on how to apply data-perturbative methods to the location histories of individuals to preserve the level of privacy, as measured by the new metric, while preserving the utility of the data.

6.1 Privacy-Enhancing Technologies and Metrics for Location Profiling Scenarios

As exposed in [152], the evolution of LBSs and the associated location techniques leads to a privacy degradation. Anonymous location traces can be identified by correlation with publicly-available databases, thus increasing the possibility of disclosing sensitive data, such as home and work locations [32] or specific points of interest of the user [47]. Therefore, users are exposed to different kinds of attacks (e.g., tracking, localization or meeting attacks, among others [130]) with the available information collected by LBS providers, which can disclose a great deal of the mobility profile of the user. For this reason, privacy enhancement is key in order to tackle the increasing new threats that arise from the evolution of LBSs.

The following is a brief overview of the state-of-the-art on privacy-enhancing technolo-

gies and privacy metrics related to LBSs and user mobility profiling.

6.1.1 Privacy-Enhancing Technologies for LBSs

Many different privacy-enhancing techniques focused on LBSs and location profiling can be found in the literature. The statistical disclosure control (SDC) community proposed many of them, aiming to prevent the disclosure of the contribution of specific individuals by inspecting published statistical information. k-anonymity [123, 122] is one of the proposed techniques. A specific piece of data on a particular group of individuals is said to satisfy the k-anonymity requirement if the origin of any of its components cannot be ascertained, beyond a subgroup of at least k individuals. The concept of k-anonymity is a widely popular privacy criterion, partly due to its mathematical tractability. However, this tractability comes at the cost of important limitations, which have motivated a number of refinements [145, 142, 90].

In the context of statistical databases appears also the concept of differential privacy [40, 56, 22]. The idea behind this approach is to guarantee that, after adding random noise to a query, if it is executed on two databases that only differ on one individual, the same answer must be generated with similar probabilities in both databases. Differential privacy is used for LBSs when aggregated location data are published. However, the scenario considered in this thesis is that of a single user sending requests to an LBS provider, which is a slightly different case. In order to cope with this difference, the concept of geo-indistinguishability has emerged recently [15, 20]. It is a variant of differential privacy for the specific case of LBSs based on the principle that, the closer two locations are, the more indistinguishable they should be. In other words, given two close locations, they should generate the same reported location to the LBS provider with similar probabilities. Since the concept of distance is present, this case is not applicable to the case of symbolic locations, since there is no distance information associated to them.

Other widely used alternatives, known as user-centric approaches, rely on perturbation of the location information and user collaboration. In this last context, the authors in [131] propose the collaboration of the users to exchange context information among the interested user and another one who already has that piece of data. This way, many interactions with the LBS provider disappear, thus increasing the location privacy by avoiding as many requests (with the user's location attached to it) to the provider as possible. On the other hand, users' interactions pose in some cases additional privacy risks. That is the case of the effect of co-location in social networks, as demonstrated in [98]. In these situations, even if the user does not disclose her location, she might reveal her friendship and current co-location with a user who does disclose her location. The authors then quantify the impact of these co-location data, deriving an inference algorithm.

Hard privacy [30] is one of the existing privacy-enhancing techniques (PETs) [35] that consists in the preservation of the privacy by the user itself by minimizing, obfuscating or perturbing the information released, without the requirement of trusted intermediaries. In principle, by perturbing the confidential data prior to its disclosure, users attain a certain degree of privacy, at the expense of degrading the system performance (or utility). A wide variety of perturbation methods for LBSs has been proposed [39]. In [38], locations and

the adjacency between them are modeled by means of the vertices and edges of a graph, assumed to be known by users and providers, rather than coordinates in a Cartesian plane or on a spherical surface. Users provide imprecise locations by sending sets of vertices containing the vertex representing the actual user location. Alternatively, [5] proposes sending circular areas of variable centers and radii in lieu of actual coordinates. Finally, we sketch the idea behind [156]. First, users supply a perturbed location, which the LBS provider uses to compose replies sorted by decreasing proximity. The user may stop requesting replies when geometric considerations guarantee that the reply closest to the undisclosed exact location has already been supplied. Besides these approaches, a number of hard-privacy mechanisms relying on data perturbation have been formulated in an application context wider than LBSs, primarily including online search and resource tagging in the semantic web. Indeed, an interesting approach to provide a distorted version of a user's profile of interests consists of query forgery. The underlying principle is to accompany original queries or query keywords with bogus ones, in order to preserve user privacy to a certain extent. The associated cost relates to traffic and processing overhead, but on the other hand, the user does not need to trust the service provider nor the network. Building on this simple principle, several protocols, mainly heuristic, have been proposed and implemented, with various degrees of sophistication [75, 44, 129]. A theoretical study of how to optimize the introduction of bogus queries from an information-theoretic perspective, for a fixed constraint on the traffic overhead, appears in [110]. The perturbation of user profiles for privacy preservation may be carried out not only by means of the insertion of bogus activity, but also by suppression [101]. These approaches constitute the basis of the present chapter.

Finally, going a step further by preserving not only privacy related to locations understood as a set of independent samples, but also the correlations among locations, the most recent works on location privacy, like [143], take into account the sequential correlation between locations, aiming at protecting the present, past and future locations, as well as the transitions between locations. The authors tackle the problem as a Bayesian Stackelberg problem and use the attacker's estimation error as the privacy metric.

6.1.2 Privacy Metrics for Data Perturbation against User Profiling

Quantifiable measures of performance are essential to the evaluation of privacy-enhancing mechanisms described before. As the focus will be placed on those mechanisms relying on data perturbation, the metrics to be reviewed will be focused on these mechanism, in terms of both the privacy attained and any degradation of utility. In a recent study on privacy metrics [112], it is shown that many of them may be understood from a unifying conceptual perspective that identifies the quantification of privacy with that of the error in the estimation of sensitive data by a privacy adversary, this is, privacy is construed as an attacker's estimation error.

Of particular significance is the quantity known as Shannon entropy [29], a measure of the uncertainty of a random event, associated with a probability distribution across the set of possible outcomes, already defined in Section 2.2.

Some studies [126, 34, 33, 97, 148, 2] propose the applicability of the concept of entropy

as a measure of privacy, by proposing to measure the degree of anonymity observable by an attacker as the entropy of the probability distribution of possible senders of a given message in an anonymous communication system. More recent works have taken initial steps in relating privacy to information-theoretic quantities [110, 111, 81].

6.2 Entropic Measures of User Privacy

As discussed at the beginning of the chapter, when an individual uses LBSs through her mobile phone—such as weather, traffic, or news widgets—the device sends, quite frequently, a service request together with the user location, aiming to obtain the most up to date information related to the current location. For this kind of services, it is sufficient to know a coarse precision location, thus cell-based location being accurate enough to obtain a reasonable result. The LBS provider may, then, collect or disclose to third parties sensible data related to the locations visited by the user. The main difference of the contributions of the work described in this chapter with respect to previous works is the distinction of two types of profile that can be built from the collection of locations sent to the LBS provider. The first profile to be defined will be the **location profile**, which consists of the set of locations visited by the user and the visit frequency to each one. This profile may disclose implicit information related to the user: her home and work locations; if she has children (the number of visits to a kindergarten or school is high); if she may suffer from some chronic disease (the number of visits to a hospital is high); if she travels much (there are visits to locations located in many different countries), among others. In these cases, an attacker aims at obtaining the most accurate estimation of the real probability distribution of the visits to each location. Then, it would be easier for the attacker to derive the implicit information enclosed in the location profile if some few locations concentrate many more visits, i.e., if the location profile is as different as possible than an uniform distribution. There exist several metrics to measure privacy in this type of scenarios where a set of labeled data exposes the user profile. Some of them are based on the concept of entropy of a set of independent samples, but as far as the literature reviewed indicates, it has never been applied to the specific case of sequences of cell-based locations.

Furthermore, a second type of profile is defined, the **mobility profile**, which can be built by taking advantage of the frequent and ordered LBS requests mobile phones usually send to obtain the updated information related to their location. It is defined as the temporal sequence of locations visited by the user. Therefore, the stress in this profile lies on the temporal correlations among the visited locations, instead of considering the locations as independent events. In this case, an attacker will aim at correctly predict the next location of the user, given her past history of locations. With this profile the adversary could derive more refined information due to the knowledge of temporal dependencies. An innocent example of personal mobility information disclosure might the following one. If the untrusted LBS provider knows, by inspecting the mobility profile, that the user goes from home (first most visited location) to work (second most visited location) and then to a third location near a supermarket, the provider might infer that the user regularly buy products at that supermarket. Therefore, the LBS provider might leak this data to other

related services which can start sending advertisement or offers of different establishments offering the same products right before the user goes to her usual supermarket. This behavior, which might result very effective for advertisement, is more persuasive when adding the temporal component to the locations profile to transform it into a mobility profile. The problem arising in this situation is that not only the set of visited locations and their visit frequency are target of a privacy attack, but also the correlations among the visits to those locations constitute a privacy threat.

As demonstrated in [134] and in the previous chapters, the correlations among location samples enclose a great deal of information when aiming at predicting the next location of a user. Since this is the target of an adversary, it is necessary to measure privacy taking into account such correlations. However, the classical concept of entropy used for the location profiles does not work on processes with memory, because it is only applicable to sequences of independent samples. Therefore, applying privacy metrics based on entropy to a mobility profile does not reflect the real privacy level, since the temporal correlations among locations visits, which represent the main component of a mobility profile, remain ignored.

Next, the different profiles considered will be formally defined, and then, the use of entropy and entropy rate as privacy measures for each case will be discussed.

6.2.1 User Mobility Profiling and the Adversary Model

Users of a LBS disclose trajectories, i.e., sequences of locations, to a service provider. With a small loss of generality for the purposes of user profiling on the basis of behavior, those positions are assumed not to be treated in the form of space coordinates, but categorized into a predefined, finite set of labeled symbolic locations corresponding to the different BTSs the individual's device connects to. As explained in Section 3.1, as the individual moves, the locations are recorded into what is known as *location history*. Further, the data contained in this location history allows to define two types of user profile, the location profile and the mobility profile. In the following subsections, these two types of profiles are defined, along with their corresponding adversary models and their connection to the concepts of entropy and entropy rate as privacy criteria.

6.2.1.1 Location Profile

The location profile is defined as the probability distribution of the visits to each of the locations in the set of visited locations of the user, i.e., the relative frequency of visits of the user's visited location set. This is analogous to the histogram of the relative frequency of the different search categories, in the case of the web search presented in [110, 102]. This profile reveals information related to different locations, independently of the rest of the visited locations and correlations among them. For instance, an attacker may be interested in knowing the probability distribution of the visits in order to know several pieces of related data, such as: home or work locations, which are demonstrated to be very easy to derive [50, 134], even when the attacker has access to just a few LBS requests [31]; if the user travels to many different countries; if the user usually visits (the relative visit

frequency is high) some hospital, religious or political organization, children school, sports center, among others. The attacker, say the LBS provider or a third party to whom the provider relinquishes the user location profile, might use this information to provide personalized advertisement or vary prices depending on the user's demand (e.g., if the frequency of the cumulative visits to locations in a different country to the one with the highest number of visits is high, it can be derived that the user travels frequently, thus she will be prone to book flights at higher prices, because traveling might be part of her work). A high number of visits to a hospital or a religious or political-related venue can have also an impact when looking for jobs or insurances.

- **Definition (location profile)**: Let L be a random variable representing the location of a given user, taking values from an alphabet of predefined location categories $\mathscr{L}$ (the set of BTSs the user's device has connected to). The time of the location referred to is chosen uniformly at random. The location profile of the user is modeled as the probability mass function, $p(l)$, of the discrete random variable L. Thus, $p(l)$ is the probability that the user is at location $l \in \mathscr{L}$ at any given time. In other words, $p(l)$ represents the relative frequency with which the user visits this location.

- **Adversary model**: The adversary model for the location profile is, in this case, estimating the visit probability distribution as accurately as possible, by inspecting the locations attached to the LBS requests. To this end, the adversary could utilize a maximum likelihood estimate of the distribution, directly as the histogram of relative frequencies, simply by counting observed locations, or any other well-known statistical techniques for the estimation of probability distributions, such as additive or Laplace smoothing.

  An intuitive interpretation of Shannon entropy as privacy metric stems from the observation that the higher the entropy of the distribution, informally speaking, the flatter the distribution and the less information the attacker could derive about predictable locations. In other words, if all of the locations have the same visit frequency, the attacker can know the visited locations, but not which of them are visited more frequently.

6.2.1.2 Mobility Profile

The mobility profile is defined as the joint probability of visited locations over time or, equivalently, as the sequence of conditional probabilities of the current location, given the past history of locations. In this case, locations are not considered independently as in the user's location profile, but the most important component is the correlation among different locations, i.e., the short- and long-range temporal dependencies among them. In this case, an attacker will aim at predicting the next location that the user will visit, given the past history. The predictions about future locations provide a further refinement for advertisement purposes: the advertiser knows not only which product might be most interesting for the user regarding her visited locations, but also when to offer it for maximizing the impact of the ad. For instance, suggesting some entertainment activity

might be more effective if, by inspecting the mobility profile, the attacker finds out that the user did not go from home to work, as usual, which might indicate a weekend or holiday. The adversary's goal is then to be able to predict as accurately as possible the next location of the user, given her past mobility history. There exists many prediction algorithms that can be used to do so, as discussed in Chapter 5, and their success depends on the predictability of the mobility history. As already discussed in Chapters 4 and 5, and demonstrated in [134], the temporal dependencies among the locations visited by the user enclose information that noticeably increases the predictability of the mobility. It must be recalled the concept of predictability, closely linked to the entropy rate of the sequence, that constitutes an upper bound on how much of the time the next location of the user can be correctly predicted, given her past mobility history.

- **Definition (mobility profile)**: More precisely, for each user, we define a stochastic process $(L_n)_{n=1,2,\dots}$ representing the sequence of categorized locations over discrete time instants $n = 1, 2, \dots$ The corresponding location L_n at time n is a discrete random variable on the alphabet of predefined location categories $\mathscr{L}$ introduced earlier. The mobility profile of the user is then defined as the joint probability distribution of locations over time,

  $$p(l_1\, l_2, \dots, l_{n-1},\, l_n,\, l_{n+1}, \dots),$$

  which may be equivalently expressed, by the chain rule of probabilities, as the sequence of conditional probability mass function of the current location, L_n, given the past location history, $L_{n-1}, L_{n-2}, \dots$, *i.e.*,

  $$p(l_n \,|\, l_{n-1}, l_{n-2}, \dots).$$

  To be consistent with the location histories described in Section 3.1, discrete times are defined as times relative to a change in the BTS the user's device is connected to, so that the actual logged data are the order of the given locations in time, but not their duration.

- **Adversary model**: The mobility profile, characterized by the probability distribution of categorized locations across time, serves to effectively model the knowledge of an adversary about the future locations of a user and raises the concern that motivates the contribution of the work described in this chapter. Since predictability is directly linked to the entropy rate of the mobility profile (the higher the entropy rate, the lower the predictability, as shown in [134]), this information theory concept could be used in order to quantify the privacy of the user mobility profile in such a way that the less predictable a user is (the higher her entropy rate is), the higher her mobility profile privacy will be.

6.2.2 Additional Discussion on the use of Entropy and the Entropy Rate as Privacy Measures

Along this chapter, an abstract privacy model is considered, in which individuals send pieces of confidential data, related to each other in a temporal sequence, to an untrusted

recipient. This intended recipient of the data is not fully trusted. In fact, it is regarded as a privacy adversary capable of constructing a profile of sensitive user interests on the basis of the observed activity or prone to leaking such observations to an external party who might carry out the profiling. Disclosure of confidential data to such untrusted recipient poses a privacy risk. However, it is precisely the submission of detailed data on preferences and activity that enables the desired, intelligent functioning of the underlying information system. Although this abstraction is readily applicable to a wide variety of information systems, the exposition done is focused on the important example of LBSs.

More sophisticated user profiling may be carried out if the privacy adversary exploits the statistical dependence among location samples over time, in order to infer temporal behavioral patterns. This responds to the observation that the disclosure of a sequence of user locations poses a clear privacy risk, especially when these locations are viewed in conjunction and time is factored in. Examples include answers to questions, such as: Where does a user commonly go after work, before heading back home? On a typical weekend, what is the user's preferred activity after leaving their house? What route does the user usually follow to get to work or back home?

The natural extension of the measurement of privacy by means of entropy to the case at hand, namely random processes with memory, is the entropy rate, formally defined in Section 2.2. Because the definition of the entropy rate is approximated by the entropy of a large block of consecutive samples (normalized by the number of samples), the very same argument in favor of entropy can be extended to the entropy rate, the latter more suitable to user profiling in terms of trajectory patterns rather than individual locations.

As already mentioned, an intuitive justification in favor of entropy maximization is that it boils down to making the perturbed, observed user profile as uniform as possible, thereby hiding a user's particular bias towards certain visited places. Less informally, the fact that entropy is a lower bound on the optimal (Huffman) code length enables to regard it as a quantifiable measure of the effort of a privacy adversary in obtaining additional bits of information in order to narrow the current uncertainty down to a deterministic outcome. Consistently, Fano's inequality lower bounds the probability of estimation error in terms of a conditional entropy, in the sense of maximum *a posteriori* (MAP) estimation, which can be readily applied to the entropy rate, written as the conditional entropy of a future location of a user given the past history. Here, MAP estimation might be construed as the action taken by a smart privacy attacker.

The arguments above justifies the use of entropy and of the more general information-theoretic quantity known as the entropy rate, as formal, quantitative measures of the effort of a privacy attacker in order to characterize and predict its behavior.

6.3 Data Perturbation Mechanisms

Following the reasons previously stated, particularly motivated by the advantages of hard privacy against the reliance on trusted intermediaries, two data-perturbation strategies prior to the disclosure of trajectories would be investigated theoretically and experimentally, in order to trade-off usability for privacy. In the first strategy, referred to as uniform

replacement from now on, with certain probability, samples are replaced with values drawn according to a uniform distribution over the alphabet of possible categorized locations. In the second mechanism, which will be called improved replacement, the same fraction of samples is replaced, although a more sophisticated policy is employed. Precisely, the replacing samples are drawn from the distribution obtained from the solution to the problem for optimized query forgery developed in [110]. It should be noted that because the optimization carried out was originally intended for memoryless processes and anonymity was measured by means of entropy instead of the entropy rate, the aforementioned improved solution need not be optimal whenever the privacy attacker exploits existing statistical dependencies over time. Consequently, both mechanisms are merely heuristics chosen to evaluate the previous proposed metrics.

The probability of replacement is indicative of the degradation in data utility and the theoretical analysis is equivalent for sample replacement and addition. In this last case, the utility degradation is understood as an increase in the information sent to the LBS provider, thus incrementing the energy consumption of the mobile device and, potentially, the economic cost derived from data traffic. The applications that could benefit from the privacy enhancement coming from sample addition must be able to send location samples to the corresponding LBS more frequently than in a normal situation (i.e., where no privacy-enhancing method is applied) with no impact over the service provided (e.g., sending more requests to a weather or news service do not alter the quality of the service obtained). That would allow the user to send fake locations together with the original ones without degrading the service provided, only increasing the cost associated with a more intensive communication. From now on, the description will focus on sample replacement, but keeping in mind that it could be extended to sample addition, by slightly changing what is understood by utility in that case. Because sample values may occasionally be replaced by themselves, especially if the number of location categories is small, counting the number of effectively perturbed values is a more adequate measure of utility. While there is ample room for the development of more sophisticated metrics of utility reflecting the quality of the LBS response, the necessarily limited scope of this contribution prefers to cover the aspects of privacy and perturbation, as the first insightful step towards the problem of privacy-enhanced perturbation of processes with memory.

Let $(X_n)_{n\in\mathbb{Z}}$ be a *stationary random process* with samples distributed on a common *finite* alphabet $\mathscr{X}$. Two alternative privacy-enhancing *data perturbation mechanisms* are proposed, in which individual samples of the random process X_n are replaced with X'_n, with probability ρ and independently from each other, as follows.

- **Uniform replacement**: X'_n is drawn uniformly from $\mathscr{X}$.
- **Improved replacement**: X'_n is drawn according to the distribution obtained as the solution to the maximum-entropy problem of [110].

Even though [110] was meant for sample addition rather than replacement, the mathematical formulation turns out to be completely equivalent. However, it should be noted that the optimality guarantee of the cited work applies to the entropy of individual samples, but *not* entropy rates in general processes with memory. Consequently, the two alternative

mechanisms described above are merely heuristic in the current context. In both cases, the resulting *perturbed* process $(X'_n)_{n\in\mathbb{Z}}$ is stationary.

Let ρ be the *replacement rate*. Because sample values may be conceivably replaced with themselves, a different *utility measure* will be proposed, which more accurately reflects the actual impact of the data perturbation mechanism. Precisely, the *perturbation rate* $\delta = \Pr\{X_n \neq X'_n\}$ is defined, constant with n on account of the stationarity of the processes involved, and observe that $\delta \leqslant \rho$, as only replaced samples may be effectively perturbed, that is, actually different.

Even in the heuristic called improved replacement, the samples to be replaced are chosen randomly and replaced independently of their original value. A truly optimal strategy, however, should choose which samples to replace, exploit the statistical model of the memory of the process, and be optimized for δ rather than ρ as a measure of utility. The scope of this work is limited to the analysis of the heuristic mechanisms described, as a first step towards shedding some light on the problem of designing perturbative strategies for processes with memory and with a truly optimal privacy-utility trade-off (or privacy-cost qualitatively talking if we would consider sample addition).

6.3.1 Uniform Replacement

Uniform replacement on stationary processes with a strictly positive replacement rate is proved to always increase the entropy rate, unless the original process is already uniformly distributed and memoryless.

Lemma 6.1. *Let S and U be independent random variables, the latter uniformly distributed on the alphabet of the former. Let T be a third random variable, in general statistically dependent on S. Take $S' = U$ with probability ρ, independently from S and T, and $S' = S$ otherwise. Then, $\mathrm{H}(S'|T) \geqslant \mathrm{H}(S|T)$, with equality if and only if either $\rho = 0$, or else S is uniform and independent of T. (Refer to Section B.1 for the demonstration)*

Theorem 6.1. *Let $X = (X_n)_{n\in\mathbb{Z}}$ be a stationary random process with samples distributed on a common finite alphabet $\mathscr{X}$. Although the process X itself need not be independent, each of its samples X_n is altered completely independently as follows. Each sample X_n is replaced by another random variable U_n, uniformly drawn from the alphabet $\mathscr{X}$, with probability ρ, and left intact otherwise. Let $X' = (X'_n)_{n\in\mathbb{Z}}$ be the resulting process, also stationary. Then, for any $m \geqslant 0$,*

$$\mathrm{H}_S(X'_0|X'_{-1},\ldots,X'_{-m}) \geqslant \mathrm{H}(X_0|X_{-1},\ldots,X_{-m}),$$

with equality if and only if either $\rho = 0$, or else X_0 is uniform and independent of $X_{-1},\ldots,X_{-m}$. The same inequality holds in the limit of $m \to \infty$ yielding entropy rates, that is, $\mathrm{H}_S(X') \geqslant \mathrm{H}_R(X)$, with equality if and only if either $\rho = 0$, or else X is uniformly distributed and memoryless. (Refer to Section B.2 for the demonstration)

6.3.2 Improved Replacement

In the case of memoryless processes not originally uniform, it is proved that improved replacement will require a lower replacement rate to achieve maximum entropy than that

demanded by uniform replacement. It will also be shown that when the cardinality of the alphabet is large, the perturbation rate approaches the replacement rate.

In the perturbative mechanisms described earlier, the *critical replacement rate* ρ_{crit} is defined as the replacement rate ρ required for the entropy rate $\mathrm{H}_R(X')$ of the perturbed process $(X'_n)_{n\in\mathbb{Z}}$ to attain its maximum possible value $\log|\mathscr{X}|$, achievable only when X' becomes memoryless and uniformly distributed. Denote by δ_{crit} the corresponding, *critical perturbation rate*. Write

$$p_{\max} = \max_{x\in\mathscr{X}} p(x) \geqslant \tfrac{1}{|\mathscr{X}|},$$

with equality if and only if X is uniformly distributed.

Theorem 6.2. *Assuming the nontrivial case in which the original process X is not already independent, uniformly distributed.*

In uniform replacement,

$$\begin{aligned}\delta &= \rho\left(1-\tfrac{1}{|\mathscr{X}|}\right),\\ \rho_{\text{crit}} &= 1,\\ \delta_{\text{crit}} &= 1-\tfrac{1}{|\mathscr{X}|}.\end{aligned}$$

In improved replacement, for any $\rho \geqslant 1-\frac{1}{|\mathscr{X}|p_{\max}}$,

$$\delta = (1-\rho)\sum_x p(x)^2 + \rho - \tfrac{1}{|\mathscr{X}|}.$$

If the original process is i.i.d.,

$$\begin{aligned}\rho_{\text{crit}} &= 1-\tfrac{1}{|\mathscr{X}|p_{\max}},\\ \delta_{\text{crit}} &= 1-\tfrac{1}{|\mathscr{X}|}-\tfrac{1}{|\mathscr{X}|p_{\max}}\left(1-\sum_x p(x)^2\right).\end{aligned}$$

Otherwise, in the general case of processes with memory,

$$\rho_{\text{crit}} = 1 \text{ and } \delta_{\text{crit}} = 1-\tfrac{1}{|\mathscr{X}|}.$$

(Refer to Section B.3 for the demonstration)

Observe that in the case of uniform replacement, a large alphabet $|\mathscr{X}|$ implies that the perturbation rate will approach the replacement rate, that is, $\delta \simeq \rho$, because of the unlikelihood of replacing a sample by itself. In the case of improved replacement, the approximation requires not only $|\mathscr{X}| \gg 1$, but also $\sum_x p(x)^2 \ll 1$, and only holds for sufficiently large ρ.

6.4 Experimental Study

The previous section formulated the theoretical problem of privacy-enhancing in processes with and without memory and how we tackle it. In [110] the authors show some results when the mechanisms proposed are applied to web queries, memoryless process, and using a small number of categories. In this section, it will be shown what happens when the scenario switches to the use of LBSs, where the number of categories increases, the probability model underneath becomes more complex, and time starts playing an essential role. The first part of the section exposes the privacy gain obtained after applying the privacy enhancing mechanisms to different processes, both synthetic and real, and the last part discusses the differences that using real location data brings to the generic problem.

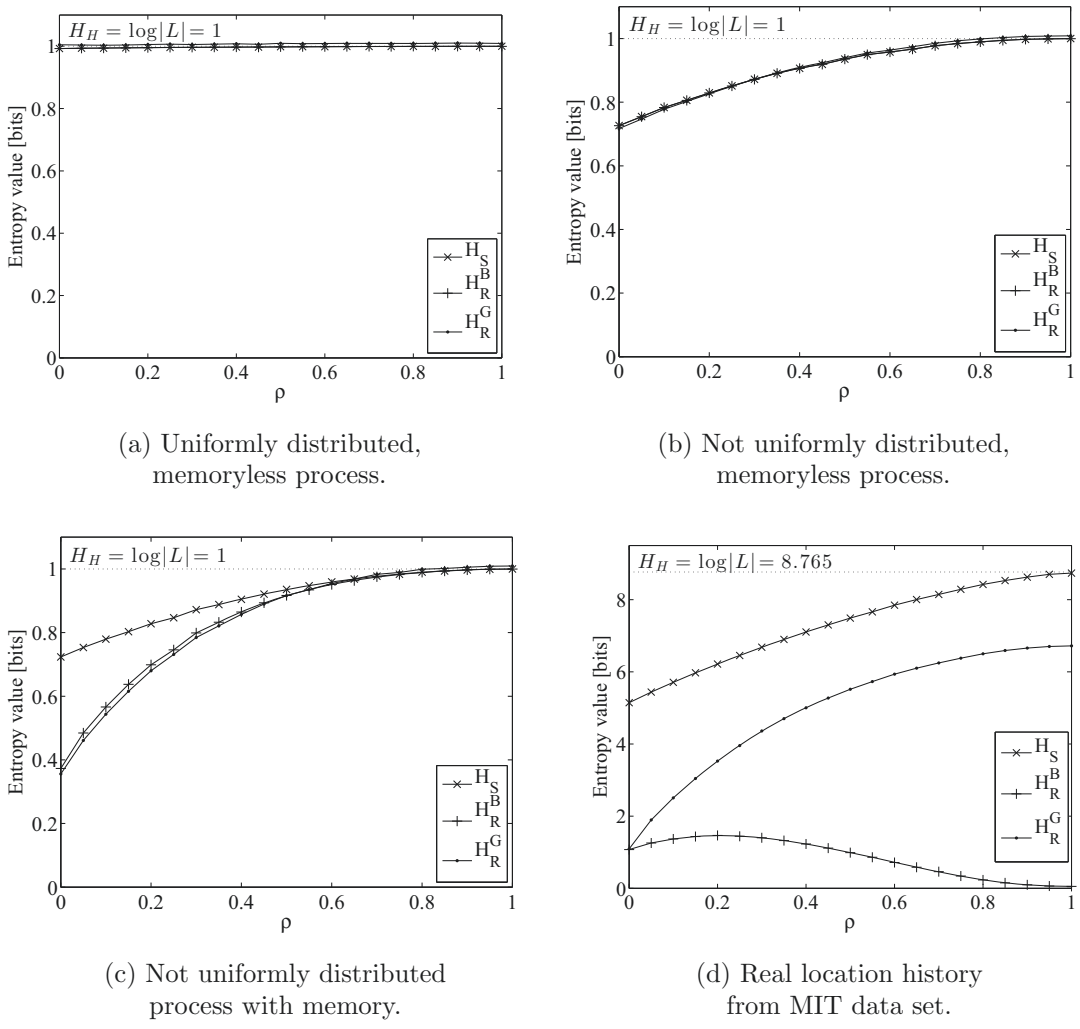
6.4.1 Experimental Results

This section collects the results drawn from applying the perturbation mechanisms described in the previous section to two different data sets. On the one hand, synthetic data coming from several symbol sequences generated by Markov processes and basic alphabets of 2 symbols. These data will allow to check the performance of the perturbation methods in simple ideal conditions, and to observe the influence of an increase of the process memory. On the second hand, real mobility data, taken from the MIT data set, will be processed and compared with the results of the synthetic Markov processes, since the real scenario can be considered as an extrapolation of simple Markov processes in terms of memory and cardinality of the alphabet. More precisely, the location history considered collects the sequence of locations visited by a user during an academic year, whose mobile device was attached to more than 500 different cells (symbols) and performed more than 10,000 cell changes (number of samples of the location history).

In order to show the privacy enhancement evolution, each process is perturbed from 0% of replaced samples (i.e., the original symbol sequence) to 100% of replacements (all samples are replaced), as explained in [110]. For each process and percentage of replacements, 10 realizations are averaged. As a general rule, the original process corresponds to $\rho = 0$, and therefore the original (and minimum) entropy value. As ρ increases, the process starts to become a uniform distribution, situation reached when ρ is maximum, i.e., when all samples are replaced by another one using the perturbation methods previously described, and therefore for the maximum value of ρ, the entropy value should be equal to H_H. It should be noticed that ρ is the percentage of replacements, but the real replacement rate is δ, since the replaced sample is sometimes equal to the original one.

First, the influence of an increase of the process memory in the entropy estimation will be studied, as well as the consequences of the process becoming less uniformly distributed, both in terms of entropy and entropy rate. For this last case, two entropy rate estimators will be compared: the block entropy and the Grassberger estimator (see Section 2.2.2 for the definition of both estimators).

Figure 6.1 shows the privacy enhancement at different values of ρ for the processes described below. Each process, L_n, will be considered as location profile, thus using Shannon's entropy, H_R, to asses the privacy improvement, an as mobility profile. When considering



(a) Uniformly distributed, memoryless process.

(b) Not uniformly distributed, memoryless process.

(c) Not uniformly distributed process with memory.

(d) Real location history from MIT data set.

Figure 6.1: Privacy enhancement at different values of ρ for different processes.

mobility profiles, two entropy rate estimators will used to evaluate the privacy improvement: the block entropy in expression (2.15), H_R^B, and Grassberger's estimator, H_R^G. The different processes considered, composed of 10,000 samples each, comes from the different distributions described below:

- An almost uniform distribution, memoryless, drawn from an order-1 Markov process. A Markov O(1) process has only two states, and thus $|\mathscr{L}| = 2$. This Markov process is determined by the following probabilities: $p(1|0) = 0.45$, $p(0|1) = 0.55$, $p(1) = 0.55$. Since the sequence is drawn from a well-known probability mass function, the real values of the maximum entropy, entropy and entropy rate are known: $\mathrm{H}_H = \log_2|\mathscr{L}| = 1 \approx \mathrm{H}_S = \mathrm{H}_R = 0.993$. This configuration represents the baseline case.

- An independent and identically not uniformly distributed (i.i.d.) process, drawn from an order-1 Markov process with $p(1|0) = 0.8$, $p(0|1) = 0.2$, $p(1) = 0.8$. The real values for the maximum entropy, entropy and entropy rate in this case corresponds to $\mathrm{H}_H = \log_2 |\mathscr{L}| = 1$, $\mathrm{H}_S = \mathrm{H}_R = 0.772$. In this case, the process is still memoryless (which holds true since $p(1) = p(1|0)$ and $p(0) = p(0|1)$), and the probability distribution of the possible symbols is slightly different than in the previous case, such that one of the two symbols of the alphabet is more likely to happen than the other one.

- A not uniformly distributed process with memory, drawn form an order-1 Markov process, in which $p(1|0) = 0.2$, $p(0|1) = 0.05$, $p(1) = 0.8$. The real values of maximum entropy, entropy and entropy rate corresponds to $\mathrm{H}_H = \log_2 |\mathscr{L}| = 1$, $\mathrm{H}_S = 0.772$, $\mathrm{H}_R = 0.374$. In this case, the process shows some memory, keeping the same cardinality and probability distribution with respect to the second case.

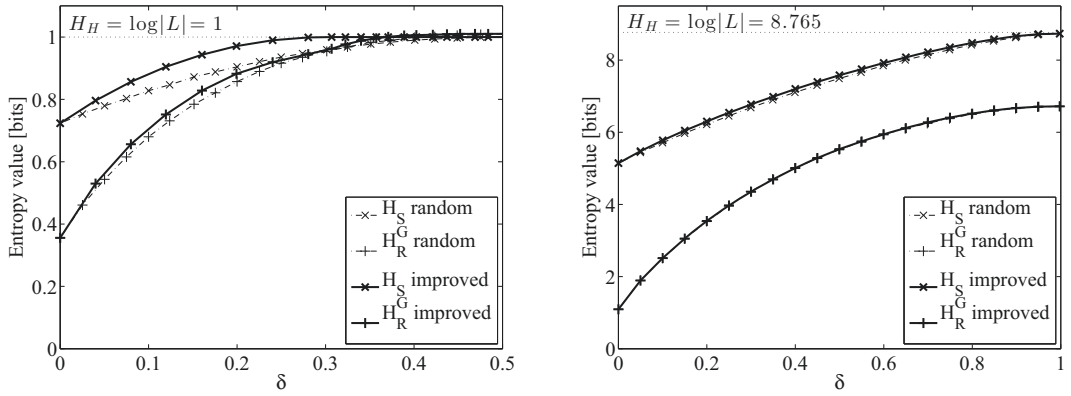
- A real mobility trace taken from the MIT data set. Only an estimation of the maximum entropy can be known, which corresponds to $\mathrm{H}_H = 8.765$ (drawn from the cardinality of the alphabet, i.e., the number of different symbols representing the BTSs the user's device connected to). Since the underlying probability distribution is unknown, neither the entropy nor the entropy rate real values are available. The use of the location history also means an increase both in the cardinality and the memory of the process, due to the long range dependencies of human mobility.

For each process, the entropy and entropy rate evolution with respect to the replacement rates is represented. The samples are replaced using the uniform perturbation method, i.e., choosing the new sample from the original alphabet of the sequence with the symbols uniformly distributed. Each process has been generated 10 times, and the results shown here are the average value of the entropy calculated in each repetition.

In the first case shown in Figure 6.1a, the original process without replacements is already very close to a uniformly distributed one, therefore there is no evolution in none of the entropy estimates. When the process is not uniform but still memoryless., such as the one in Figure 6.1b, H_S and H_R coincide, as there is no temporal information that can be captured by H_R to lower the uncertainty. Besides, both entropy and entropy rate values are lower than H_H for $\rho = 0$, since the original process is not uniformly distributed, and their values increase as the replacements turn the process into a uniform one.

Figure 6.1c shows what happens when the process is not memoryless anymore. In this case, for $\rho = 0$ H_R is lower than H_S, since it the entropy rate leverages the temporal information present now in the original process to lower the uncertainty.

Finally, Figure 6.1d shows what happens when the number of different symbols (i.e., the cardinality of the alphabet) increases, as well as the memory of the process. In this case, $|\mathscr{L}| = 500$ different symbols, what leads to $500^2 = 250,000$ possible blocks of $m = 2$ symbols to compute H_R using block entropy (the blocks are of two symbols to compare with respect to the Markov processes). Since the number of possible blocks is so high and the number of samples is only of 10,000, more different blocks of two symbols come to scene as the process becomes uniform. With this number of samples not every different block can appear in L_n, which probability would be $p(l_1, \ldots, l_m) = \frac{1}{250,000} = 4 * 10^{-6}$.



(a) Not uniformly distributed process with memory.

(b) Real location history.

Figure 6.2: Comparison of perturbative methods for different privacy measures for different processes.

Therefore, when computing $\mathrm{H}_R(L_n)$, the values of the elements of the summation are very small, due to the scarcity of occurrences of each possible block. This scarcity becomes more severe as the process tends to uniformity. Thus, $H_R(L_n)$ decays to near zero as the number of replaced samples increases, as shown in the figure. As explained in Section 2.2.2, this entropy estimation is biased by the small number of samples available in the location history of the user (even when it comes from a year of location tracking). This is the reason behind considering a different estimator like the one proposed by Grassberger et. al and described in Section 2.2.2, represented by H'_R in the figure. Figure 6.1d shows how this estimator obtains more reasonable results. Both $H_R(L_n)$ and $H'_R(L_n)$ are equal for the original sequence ($\rho = 0$). However, in order to analyze the privacy improvement, an estimator that works well for all the replacement rate span is required.

Once the role of memory in the processes and how entropy rate is able to capture it by using an appropriate estimator are understood, the perturbative methods proposed in the previous section will be analyzed under the entropy and entropy rate-based privacy metrics. Figure 6.2 represents the privacy level obtained using the perturbative methods described in Section 6.3, for the synthetic not uniform process with memory described before, and for the location history drawn from the MIT data set. For each case, four plots can be distinguished: the privacy enhancement in terms of entropy and entropy rate values, for the two perturbative methods considered, uniform and improved replacements.

For the case of the synthetic process in Figure 6.2a, it is observed that the privacy enhancement is faster for the improved perturbation method, mainly when no correlations are considered, this is, when the privacy metric is based on entropy measurements. That highlights the fact that preserving the information enclosed in those correlations is more difficult, using any of the replacement methods. This fact is reinforced by observing that the privacy level measured by the entropy rate reaches the maximum value when 35% of

samples are replaced, value that lowers up to 25% when the improved perturbation method is used and the privacy metric is just the entropy of the sequence, thus not taking into account temporal information.

When this same analysis is applied to the mobility trace, the results are quite different, as shown in Figure 6.2b. In this case, since the cardinality of the alphabet is so high, a 100% of replacements are required in order to obtain the highest privacy level, when measuring privacy by means of the entropy of the sequence. Besides, the maximum entropy is only achieved when no temporal correlations are considered. In order to get the maximum value for sequence-based data, this is, using the entropy rate-based privacy metric, many more samples would be needed in order to have a precise entropy estimation. In this case it could be checked that the improved perturbation method does not provide faster privacy enhancement for any case, thus opening up an interesting research line to find perturbative methods that improve this result.

6.4.2 Discussion

In the previous figures it could be observed the great difference between theory, with simple Markov processes, and real scenarios, such as users' mobility profiles. But, where do these differences stem from? Although the high cardinality of the alphabet and the complexity of the short and long term dependencies of location histories play an important role, the probability distribution underneath the mobility trace is also crucial. A great majority of visits are concentrated in two or three locations, corresponding to home, work and the main points of interest of the user, as already discussed in Section 4.2. Therefore, the probability distribution is very biased toward certain locations. The improved perturbation method is based on flattening the underlying distribution with as few replacements as possible in order to get closer to a uniform distribution, and thus to maximum entropy (i.e., privacy). When the number of different locations is not very high and the probability distribution is not very biased to certain few categories, it is easier to flatten it, as in the case of the Markov processes shown. However, in order to flatten the mobility traces, the visits to 2 or 3 locations would be needed to be compensated through the rest of the 500 different locations visited along the year. Although there are more than 10,000 samples, the cardinality is still very high, and would need many more samples to be flatten. This issue is even more critical when considering not the distribution of the visits, but the sequences of locations. By considering just the short-term dependencies (short mobility patterns), longer mobility patterns are being neglected, and even in this case the number of combinations is too high to compensate the number of occurrences of the most repeated sequences. Considering long-term dependencies (long mobility patterns) leads to so many combinations that there are not enough samples to even calculate a good entropy estimate, even worse if the block probability distribution is trying to be flatten.

The bias in the visits probability distribution carries an important consequence: for an attacker, it is quite easy to analyze a set of locations and determine where the main points of interest of the user are. Therefore, these become sensible data that must be masked. The bias can be leveraged in such a way that, instead of trying to flatten all the distribution, it could be enough to focus on the set of the most visited locations and

just flatten their number of visits, leaving the least visited ones as they are. This way the uncertainty of which of the most visited locations is home or work increases with few number of replacements. If the number of replacements is not critical, or the locations to be disclosed can be faked, the approach could be to select some of the least visited locations and increase their number of visits to make it comparable to the most visited places. However, as mentioned before, this strategy will require a great number of replacements or additional fake locations. Again, it should be noted that adding fake locations incurs in a battery and data traffic increase, thus being utility-related factors to be taken into account when deciding which data perturbation approach to follow.

In the case of location sequences, where the focus is on preserving the privacy of the correlations among locations, improving such privacy without compromising the data utility (or avoiding additional cost when adding fake samples instead of replacing the original ones) is more complicated and depends heavily on the application at hand. As can observed in the figures, in order to obtain high privacy levels, the fraction of location samples to change grows fast. Furthermore, the replacements should be done wisely. For example, let be a user walking in Madrid. If during the user's location sampling done by the corresponding mobile application communicating with the LBS provider (done every few minutes) the system replaces a location in Madrid by another one in New York city (or just adds the location in New York city in the mobility profile), an attacker could easily detect that it is not possible for a user to make this large jump in such a short period of time. Therefore this replacement/addition might seem to theoretically improve greatly the privacy level (it is an unexpected movement, thus the entropy rate of the mobility profile would increase) with little disruption of the utility of the result (because just one location was replaced/added, and the system can ignore the result of the associated request, by knowing it is a fake one). However, it would be easy for an attacker to notice the impossibility of the jump, due to the recent past history, and ignore the location in New York. This happens when considering a mobility profile, since location profiles by their own just account for the number of visits to each place, leaving unnoticed this kind of impossible large jumps between locations in a short period of time. It can be devised then a semantics and scale-related problem. What data wants to be preserved? For instance, if only the work/home locations are the ones to be protected, the perturbation methods should focus on replacing or adding samples of the same city repeatedly, so that their frequency is comparable to the one of home/work locations. Since the places could be nearby, it would be more difficult for an attacker to distinguish among the real and fake ones. However, if the target is to preserve the country where the user is, the perturbation mechanism needs to be more sophisticated to make the attacker believe the user might be at any of several countries by creating equally believable location profiles.

6.5 Conclusions

This chapter has analyzed privacy-enhancing mechanisms based on information theory concepts, such as entropy and entropy rate, applied to locations and mobility profiling scenarios. Starting with synthetic and simple processes, it has been shown that the the-

ory applicable to these low alphabet cardinality, memoryless processes cannot be directly applied to more complex cases, such as mobility profiles of users. Therefore, the remarkable results obtained in the simpler case get degraded until little privacy enhancement is observed, unless utility is completely lost.

The main reasons leading to these results are the increase in the alphabet cardinality (from a few categories to hundreds of visited places by a user), and the temporal dependencies introduced by the fact of considering mobility profiles instead of set of independent samples (location profiles). This last reason leads to the need of using general privacy metrics, such as the one proposed in this chapter, based on the information theory concept of entropy rate. This concept allows to to consider the temporal dependencies of the mobility profiles. Moreover, the probability distribution defining the mobility profile of a user is highly biased toward certain frequently visited places, which makes it difficult to hide these locations just by replacing the rest of samples by random locations.

As discussed earlier, careful replacement methods should be studied for these special cases. An interesting future research line might be to investigate how to replace samples taking into account the current and past locations, in order to provide reasonable replacements, and to exploit the biases toward the most visited locations to flatten the probability distribution, since these locations and their visitation profile are the keys to identify the user behind such profiles.

Chapter 7

Conclusions and Future Work

Contents

Along the dissertation, the objectives described in Chapter 1 have been addressed in each of the chapters. This final chapter collects the main conclusions derived from the work carried out on the pursuit of each of them. Besides, some of the work described along this document has been published in different scientific dissemination sources and interacted with different research projects. Thus, the chapter includes also a reference of the publications derived from this thesis, together with the related research projects. Finally, some of the many ideas emerging from the work started with this thesis and that can extend it in the future are depicted in the end of the chapter.

7.1 Conclusions

Recalling the thesis proposal, it focus on studying the individuals' mobility by means of the location data provided by their mobile devices, aiming at extracting conclusions that can be applied to improve mobility prediction. This aim was tackled from the very first step of the process, the mobility data collection, all the way to the prediction process, going through the analysis of the mobility data, and accompanying the process by considering the privacy issues related to the disclosure of a person's mobility data. Each of these steps led to several conclusion, that are summarized next.

The research on the mobility data that best suit the purposes of the thesis resulted in the next conclusions:

- The mobility data source selected among the available ones (GPS, Wi-Fi, cellular telephony network, and LBSN) is the cellular telephony network. The choice is supported by the following reasons: Its global coverage, both in indoor and outdoor environments; its low power consumption, which allows to continuously track the individual without draining her device battery; and the autonomous location tracking, without requiring the individual explicit participation. These characteristics make this data source the best candidate to collect the most complete mobility data history, although location accuracy is sacrificed.

- When using the cellular telephony network, three different ways to capture data that can be directly translated into mobility information are exposed. The baseline approach refers to the mobile device capturing the BTSs to which the user's mobile phone is attaching to as she moves. The CDR-based approach retrieves the BTSs to which the user's mobile phone was attached to when the user made or received a voice call or message. And the DDR-based approach is the equivalent to the CDR one, but based on data traffic events (sending to or receiving data from the Internet through the mobile network). The first scheme generates the most detailed movement history, but it is difficult to collect these histories from an extensive set of users. The two last schemes collect less detailed histories, but the set of users is more extensive because this information is collected by operators for billing purposes, and thus all users from an operator are indirectly tracked using these two schemes.

- An extensive literature review revealed that there exist many works using different sets of mobility data based on cellular telephony network. However, only one of them with a significant amount of users providing data from the three approaches described above is available: the MIT data set. It was collected back in 2005 among a group of 95 persons working or studying in the same campus. Although being a very useful mobility data information, two main factors were spotted, which can lead to future deceiving mobility conclusions: the data set was collected a few years ago, when the data traffic connections were not very popular yet, thus providing poor DDR-based data; and the subjects contributing to the data set share the place where they spend most of their days, thus there is a high probability of the individuals sharing timetables, calendars and spatial mobility patterns.

- In order to have an additional mobility data source, more updated and with mobility data coming from people not sharing temporal nor spatial patterns, a new data set has been collected in the framework of the thesis: the UC3M data set. It collects baseline, CDR and DDR data from 25 users living in different countries and with no relationship among all of them, during more than a year. The data collection campaign shown to be difficult to be extended to more people, and the subjects participating varied in the time they continued with the data collection process, even when the application collecting the data in their mobile phones demonstrated to have a negligible battery consumption and be unobtrusive to the normal usage of the device.

Once having the most suitable mobility data sets, their analysis to characterize the users movements features disclosed the following ideas:

- The literature review shown the wide variety of data used to study human mobility, as well as the variety of features studied to characterize how people move. However, no comparisons among the features reflected by each type of mobility data were found in the research published so far, thus generating an uncertainty on the potential biases the different mobility data might introduce into the conclusions on mobility features.

- To clear up such uncertainty, a comparison among the mobility features enclosed in the baseline, CDR and DDR-based data were carried out, using both the MIT and UC3M data sets. Such comparison unveiled big differences in the number of cell changes experienced per day, the number of different cells visited per day, the entropy rate or randomness of the users, and the resulting predictability of the user's movements. These results invite to a reflection on the generalizations usually made in the mobility studies found in the literature, which should be put into context considering the data used for the study.

- Although the data coming from the baseline approach shown to be the one most faithfully capturing the user's real mobility features (independently of their network usage), it was also shown that it introduces certain bias due to a network-related issue known as ping pong effect. This effect causes the user's mobile phone to switch continually between mainly two or three cells while not moving. This effect heavily impacts the data, as it introduces a very high number of cell changes not related to movements.

- In order to alleviate this problem, an online filtering algorithm was proposed. It is based on a detection scheme and a filtering stage. The detection scheme is in charge of rapidly determining whether the current cell represents a real location or is part of a ping pong sequence. This scheme can be configured to delimit the number of cell changes to be analyzed in order to determine the existence of a ping pong sequence. For the filtering stage, three different techniques were proposed: representative, limits, and hybrid. They differ in how the original ping pong sequence is substituted in the movement history: by the most visited symbol, by the limits of the ping pong sequence, of by the limits and the most visited symbol, respectively.

- The representative technique shown to provide the most simplified histories. However, if the application at hand requires to maintain the real locations structure (e.g., cell adjacency), then the limits or hybrid techniques are better choices that provide similar results and also a noticeable filtering capacity. In any of the three cases, the impact of the ping pong sequences on the mobility features reflected on the movement history is clearly decreased. The mobility feature most affected is the entropy rate (i.e., the randomness of users' movements), which increases in the case of the filtered traces, meaning that the movement of the user is more random that it seemed when the ping pong sequences were taken into account. The reason behind this fact is that ping

pong sequences have a fixed structure continuously repeated, which add a virtually deterministic behavior.

The prediction process used all the previous knowledge to provide the following outputs:

- The review of an extensive set of existing location prediction algorithms revealed many different approaches to perform predictions about the future location of a user. Among the many alternatives, a family of algorithms was found that allow for online execution (i.e., no training phase), low computational requirements that make it possible to be executed with the continuously growing mobility data, and adaptive ability that allows to learn changes in the mobility behavior of the user. This family, known as LZ family, is comprised by the LZ, LeZi Update and Active LeZi algorithms.

- By carefully inspecting the three algorithms, a division into two independent phases is proposed, which allows for combinations among them. The first phase, the tree updating scheme, is in charge of building and storing the mobility model of the user in the form of a tree. The second phase, the probability calculation method, takes care of combining the information of the mobility model to estimate the most probable next location, considering the current context (i.e., last visited locations).

- The evaluation of the different combinations of these two phases revealed that the most impacting phase is the probability calculation technique, since selecting the best instance of this phase, the PPM algorithm without exclusion, the prediction accuracy results are very similar for the three different tree updating schemes. The small differences among these three schemes are owed to the amount of patterns stored on the corresponding trees or models.

- When considering the useful predictions, meaning the predictions not corresponding to ping pong sequences, it can be seen that the prediction accuracy suffers a noticeable decrease. Thus, most of the part of the right predictions are the ones corresponding to the ping pong sequences, since as said before, these symbols are easy to predict due to the fixed structure of the sequence.

- As previously observed in the mobility features reflected in the baseline and filtered traces, the entropy (i.e., randomness) of the filtered ones is higher than in the baseline case. Thus, it directly translates into a decrease of the prediction accuracy obtained when processing the filtered movement histories. The representative case is the one with the lowest fraction of right predictions, whilst the limits and hybrid ones provide similar results, slightly better than the ones coming from the representative filtered traces.

- Markov models of orders 1 or 2 are widely used in the literature. However, the analysis done with the baseline and filtered traces reveals that, whilst Markov models achieve the best results in the baseline case, these results are virtually better due to the ping pong sequences. When Markov models are applied to predicting the next locations of the filtered traces, they show to perform worst than the Active LeZi algorithm.

- In order to tackle the intrinsic problem of the mobile phone connecting equally to either of two or three cells, the two most probable next locations can be used. It has been shown than using two symbols greatly improves the prediction accuracy, whilst the improvement of using three symbols is not that noticeable. This result holds for both the baseline and filtered traces, which means that the effect of the network in the collected data goes beyond the ping pong sequences. There is an additional noise coming from the network behavior because of which when a user follows a fixed route from point A to B, the cells to which her mobile device connects to during the route can be different every time the user follows it. At each location, the device receives the signal coming from several near BTSs, connecting to the one from which it receives better signal strength. This cell is not always the same, but varies mainly between two choices, as suggested by the results obtained. Unfortunately, for this kind of noise, the solution is not as easy as for the ping pong sequences, because there is no fixed structure that can be easily detected.

- The analysis of the prediction accuracy results with respect the mobility features previously studied revealed that the entropy rate is clearly the feature driving the prediction success. Whilst the prediction accuracy and entropy shows a slightly decreasing relationship, the dependence of the accuracy with the entropy rate is also decreasing, but much stronger. Thus, by reducing the entropy rate of the mobility history of a user, the number of correct predictions increases.

- Under the previous conclusion, a new tree updating scheme is proposed: the Extended LeZi algorithm. It is based on the principle of collecting more significant patterns that allows to reduce the entropy rate of the mobility model representing the user behavior. An auxiliary algorithm to calculate the entropy rate enclosed in the mobility model represented by the tree built by these algorithms was also proposed. It shows that the mobility model built by the Extended LeZi algorithms has a lower entropy rate than the model built by any of their ancestors–LZ, LeZi Update, or Active LeZi–. However, when the Extended LeZi tree updating scheme was applied to prediction purposes, it obtained the same results than the Active LeZi algorithm in terms of prediction accuracy. This leads to think that it is not enough to decrease the entropy rate of the built model, but also that of the movement history itself. Thus, the prediction algorithm used is an important choice, but the mobility data itself takes on a critical importance in light of these results.

- In order to improve the probability calculation methods, it was taken into account the low number of samples collected by the longest patterns stored in the corresponding tree. To overcome the poor estimation that would potentially come from considering such subsampled patterns, Vitter and PPM without exclusion were modified in order to take into account patterns with a frequency above different thresholds. This modification did not provide any improvement on the PPM case. However, when applied to the Vitter method, the results were improved up to the ones provided by PPM, and even beyond in some cases. This is a great results considering that Vitter method implies a much lower computational complexity, and thus, a shorter

processing time for each prediction.

Finally, the concern about the preservation of the privacy of the user permeates any use of mobility data due to its sensitive nature. Under the research on the privacy metrics covered in the thesis, the main conclusions can be summarized as follows:

- The review of the state of the art on privacy metrics related to the location profiles of the users revealed a high focus on preserving the locations visited by the user when considered independently from each other. However, little attention has been paid to the effect of disclosing sequences of locations, with not only spatial but also temporal correlations that discloses the mobility patterns of the user.

- To cope with the potential privacy threats coming from this new perspective of mobility profiles, a new privacy metric based on the entropy rate of the user's movement history is proposed. As exposed along the thesis, the entropy rate is tightly coupled with the right predictions fraction that can be attained. Thus, increasing the entropy rate leads to a decrease in the capability that a potential attacker might have to foresee the future movements of the user. Thus, entropy rate is used as privacy measure for location and mobility profiling scenarios.

- Two perturbative techniques were also proposed to be applied in these scenarios, showing that preserving the privacy of mobility profiles (i.e., not independent locations, but location sequences) is way more difficult than preserving the privacy of location profiles, without loosing the utility of the data.

7.2 Contributions

Besides the conclusions discussed in the previous section, four main contributions can be extracted from the work carried out in this thesis:

- A mobility data collection campaign was carried out. It generated an updated mobility data set, comprised by 25 users living in different countries and following completely independent lives, and thus, following completely different mobility patterns. The data collected includes, among other information, the GSM and UMTS cell the user's cellphone is connected to at every time instant and the timestamps of all calls and data traffic events. This data set allows to provide comparisons between the results obtained when using the data collected by the users' mobile devices (more complete, but with smaller data set population) with respect to the results observed when the data is collected by the operator (less complete, but with a more extensive population).

- A thorough analysis of the mobility features reflected in the data collected by the mobile phone with respect to the data collected by operators, pointing out their striking differences. To address the biases detected in the data collected by mobile

phones, several filtering techniques were proposed and evaluated regarding the mobility features reflected before and after the filtering process, unveiling also noticeable differences not mentioned in the literature before.

- A dissection of the prediction algorithms coming from the Markov and LZ families, which shown that Markov models are not always the best choices, as largely stated in the literature. This analysis also unveils the importance of, not just the method, but the quality of the input data. Their specific characteristics must be carefully studied first to avoid delusional results.

- A method for calculating the entropy of the mobility model built by each algorithm was proposed, as well as two modifications of the algorithms: a new LZ-based mobility model construction that reduces the entropy of the resulting model, and a variation in the probability calculation methods. The new mobility model revealed that reducing the entropy rate of the mobility model does not guarantee the improvement of the prediction accuracy, but the reduction of the real entropy rate of the mobility sequence does. The variation in the probability calculation method leads to equal or better results, at a lower computational cost.

- A new privacy metric was proposed to deal with the disclosure of the user location. This metric aims at measuring the privacy loss when disclosing, not just independent locations (as largely studied in the literature), but also the locations as a sequence, which discloses also the mobility patterns of the user. By using two data perturbative methods, it was demonstrated the difficulty to preserve the privacy of this new dimension brought by the concept of pattern, compared to the relatively ease to preserve the privacy of independent locations.

7.3 Impact of the Thesis

7.3.1 Publications and Conferences

Part of the work carried out during this thesis has been published in different scientific dissemination venues. The following list collects those publications:

- The contributions on the analysis of the mobility features reflected in the mobility data records coming from different data sources were published in:

  – Alicia Rodriguez-Carrion, Sajal K. Das, Celeste Campo, and Carlos Garcia-Rubio. Impact of location history collection schemes on observed human mobility features. In Proceedings of the **2014 IEEE International Conference on Pervasive Computing and Communications Workshops** (PERCOM Workshops), 254–259, 2014. [115]

- Different analysis of the prediction performance of the algorithms analyzed in this thesis were published in:

  - Alicia Rodriguez-Carrion, Celeste Campo, and Carlos García-Rubio. Recommendations on the Move. **Book: Recommender Systems for the Social Web**, 23:179–194. Springer Berlin Heidelberg, 2012. [114]
  - Alicia Rodriguez-Carrion, Carlos Garcia-Rubio, Celeste Campo, Alberto Cortés-Martín, Estrella Garcia-Lozano, and Patricia Noriega-Vivas. Study of LZ-based location prediction and its application to transportation recommender systems. **Sensors**, 12(6):7496–7517, 2012. Impact factor as of 2012: 1.953. JCR(8/57), Q1, category: Instruments& Instrumentation. [117]
  - Alicia Rodriguez-Carrion, Celeste Campo, Carlos Garcia-Rubio, Alberto Cortés-Martín, Estrella Garcia-Lozano, Patricia Noriega-Vivas. Analysis of location prediction performance of LZ algorithms using GSM Cell-based location data. In Proceedings of the **5th International Symposium on Ubiquitous Computing and Ambient Intelligence** (UCAmI 2011), Mexico, 2011. [118]
  - Alicia Rodriguez-Carrion, Carlos Garcia-Rubio, and Celeste Campo. Performance evaluation of LZ-based location prediction algorithms in cellular networks. **IEEE Communications Letters**, 14(8):707–709, 2010. Impact factor as of 2010: 1.060. JCR(29/80), Q2, category: Telecommunications. [116]

- The contributions on the proposal of an entropy estimator for mobility models built by LZ-based algorithms were presented in:

  - Alicia Rodriguez-Carrion, Carlos Garcia-Rubio, Celeste Campo, and Sajal K. Das. Analysis of a fast LZ-based entropy estimator for mobility data. In Proceedings of the **2015 IEEE International Conference on Pervasive Computing and Communication Workshops** (PerCom Workshops), 451–456, 2015. [119]

- The contributions on the privacy metrics for location profiles based on entropic measurements were published in:

  - Alicia Rodriguez-Carrion, David Rebollo-Monedero, Jordi Forné, Celeste Campo, Carlos Garcia-Rubio, Javier Parra-Arnau, and Sajal K. Das. Entropy-Based Privacy against Profiling of User Mobility. **Entropy** 17(6): 3913–3946, 2015. Impact factor as of 2014 (the last published JCR): 1.502. JCR(34/78), Q2, category: Physics, multidisciplinary. [120]

7.3.2 Research Projects

This thesis has been carried out in the framework of the research projects described below:

- **INRISCO: INcident monitoRing In Smart COmmunities**
  - Organization: Science and Innovation Spanish Ministry.
  - Duration: January, 2015 - December, 2017.

  - Partners: University Polytechnic of Catalonia, University of Vigo, Galician Research and Development Center in Advanced Telecommunications (Gradiant).
  - Contribution or Influence: In order to monitor incidents by using the data collected by the mobile phones of the citizens, the experience extracted from the data collection campaign carried out during the thesis will be used. Since any incident is an unusual event, the techniques based on measuring the entropy rate of a sequence of events and its increment can be used to rapidly detect this type of situations.

- **EMRISCO: EMergency Response In Smart COmmunities**
  - Organization: Science and Innovation Spanish Ministry.
  - Duration: January, 2014 - June, 2015.
  - Partners: University Polytechnic of Catalonia, University of Vigo, Galician Research and Development Center in Advanced Telecommunications (Gradiant).
  - Contribution or Influence: Considering citizens as mobile sensors, they could collect data that can be used to detect emergency situations. The data collection campaign carried out for this thesis provides knowledge and experience on how to perform the data collection using mobile phones. Besides, the entropy rate estimators studied allow to detect unusual patterns that complement other indicators in the task of early detection of emergency situations.

- **TransITS: Modeling Public Transport Passenger Flows in the Era of ITS**
  - Organization: COST - European Cooperation in Science and Technology. COST Action TU1004
  - Duration: May, 2011 - May, 2015.
  - Website: https://sites.google.com/site/costtransits/
  - Partners: Transport&Telecommunication Institute Latvia, LogistikCentrum AB sweden, The University of Sydney Australia, Royal Institute of Technology Sweden, ENPC France, University of Malta, TU Delft Netherlands, University of Rome "Tor Vergata", University of Cantabria (Spain), PTV Austria, TU Graz (Austria), Napier university (United Kingdom), Universitat Stuttgart (Germany), Leeds University (United Kigndom), University Carlos III of Madrid (Spain), University of Rome "La Sapienza" (Italy), EPFL (Switzerland), Akdeniz Universitesi (Turkey), ICOOR (Italy), University Polytechnic Madrid (Spain), Transport Analysis (Sweden), City University London (United Kingdom), Road and Bridge Research Institute (Poland), Budapest University of Technology and Economics Muegyetem (Sweden), Transport and Logistic Centre (Hungary), Krakow University (Poland), Gifu University (Japan), University of Stavanger (Norway), Molde University College (Norway), Ohio State University (United States), University of Thessaly (Greece), University of Stavanger (Norway), PTV AG (Germany), University of Porto (Portugal), Universita

Mediterranea di Reggio Calabria (Italy), University of Hawaii at Manoa (United States), Autoritat del Transport Metropolità (Spain), TU Dresden (Germany), KU Leuven (Belgium), TfL (United Kingdom), University of Luxembourg, MTU Harjumaa Ühistranspordikeskus (Estonia), MIT (United States).

- Contribution or Influence: Presentation of mobile phones as a key alternative to collect data from public transports, to get over the problems coming from using surveys, video-surveillance or tickets.

- **MONOLOC: Indoor Positioning and Mobile Network Management**
  - Organization: Science and Innovation Spanish Ministry
  - Duration: September, 2011 - December, 2014.
  - Website: http://monoloc.creativit.com/eng/index.html
  - Partners: University Polytechnic Madrid, University of Malaga, Alcatel-Lucent España S.A., Innovati S.L.
  - Contribution or Influence: Apply data collection methods and knowledge obtained along the thesis, to capture individuals mobility in indoor environments, taking into account the limited resources of mobile phones.

- **CONSEQUENCE: Continuity of Service, Security and QoS for Transportation Systems**
  - Organization: Science and Education Spanish Ministry
  - Duration: January, 2011 - April, 2015.
  - Website: http://consequence.it.uc3m.es/index.html
  - Partners: University Polytechnic of Catalonia.
  - Contribution or Influence: Collaboration with the privacy research group at University Polytechnic of Catalonia to design and propose a privacy metric to preserve location-related information of the users.

- **España Virtual**
  - Organization: Science and Innovation Spanish Ministry. CENIT Program
  - Duration: February, 2008 - December, 2011
  - Website: http://www.xn--espaavirtual-dhb.org/
  - Partners: University Polytechnic Madrid, University of Zaragoza, UNED, University Polytechnic of Catalonia, University Jaume I, University of Valladolid, Universitly of Illes Baleares, University of Malaga, University Polytechnic of Valencia, University Pompeu Fabra, Barcelona Media, Vicomtech, Elecnor Deimos, Geographical Information National Center, Indra Espacio S.A., Androme Iberica S.L., GeoSpatiumLab S.L., Designit, Prodevelop, Telefonica I+D.
  - Contribution or Influence: Study of prediction algorithms able to foresee the future whereabouts of the user, executed in mobile phones.

7.4 Future Works

Despite having addressed all the objectives set at the beginning of this thesis, the work carried out during the process has also opened many interesting paths to explore that can be considered as new objectives to focus on.

- The study of mobility features can be applied to many fields. Location prediction is one of them, but another very popular application of the analysis of human mobility is the development of mobility models that can help to provide realistic synthetic traces. The main advantage of these synthetic movement histories is that the model allows for certain personalization. That means that, whilst in the real world people cannot be configured to have certain features, mobility models allow to tune parameters so that it is possible to generate traces of more or less random users, who travel more or less distance, visit more or less different locations a day, etc., always maintaining the main mobility features of real users. This ability to configure the main mobility features can help to better understand the behavior of certain network protocols or similar applications depending on mobility under certain scenarios. One of the main future lines is the proposal of a mobility model able to create movement histories made up of the BTSs the synthetic user connects to as she moves. It can help to improve services based, not on the coordinates of the user, but on her location based on the cellular network. This line has been already started by using a well known mobility model called SLAW [80], which shows to faithfully represents the mobility of users as compared to their real mobility traces, in terms of real coordinates. The current work is focused on tuning this work to be applied in a BTS map, and matching the mobility features reflected in the MIT and UC3M data sets.

- The prediction algorithms exposed along the dissertation can be extended in many directions. One of them is to consider the prediction of further locations beyond the next one. The works in the literature show low prediction accuracy to this respect, that can definitely be very interesting for pervasive applications aiming at knowing what the user wants before she does, and applied to the location part of the user context, knowing where she wants to go before she starts the trip. Another possible direction is related computational cost of the prediction. It must be taken into account that the mobility information keeps on increasing daily and, thus, the entity performing the prediction will at some point run out of memory to keep storing and processing it. Even using LZ-based trees, which store the mobility patterns in a very compact way, some pruning strategies need to be designed. Based on one of the features revealed in the human mobility study done in the thesis, the concentration of the most part of the visits to a reduced number of locations, methods to delete locations or paths rarely transited to bound the data stored to those paths that will concentrate most of the visits (and thus, most of the potential right predictions) can be tested.

- The mobility data collection process described in this work can be extended to other areas, such as transportation, as proposed in the framework of the COST project

mentioned before. When studying transportation models, some real data is needed in order to calibrate and validate the model. However, obtaining such data is usually costly, both economically and in terms of time, and sometimes its availability depends on the decision of third parties. The usual surveys need a previous study of the sample size and take a non-negligible time to perform them. Besides that, the selection of people is an important step, and even after a careful selection, it is possible to obtain biased results due to common features of the sample individuals, which were not supposed to affect the analysis but they actually do. Data obtained from transportation infrastructure (ticketing, video-surveillance) face these problems because it collects the information of all passengers, so that the results are more general and reliable. However, obtaining such data sets requires the permission of the transportation operator, and this is not always an easy task. If we want to cover multi-modal trips, the process becomes even harder.

In the project, the use of some wireless communication technologies (mobile telephony system, Wi-Fi, or Bluetooth) was proposed as a tool to record data related to the usage of public transportation: start and finish stops of bus users, waiting time at bus stops, as well as routes and combinations in subway trips. The use of these technologies potentially provides important benefits: the low cost, since there is no need to install new infrastructure; the sample size, which is remarkable due to the huge penetration of mobile phones providing access to such technologies; and the short time needed to obtain the data, that can even be collected in real time.

- **Mobile telephony network** has been widely used as location and tracking technology for several applications, such as deriving origin-destination (O-D) matrices [17, 132], mapping geographical cell phone usage at different times of the day for urban analysis and planning [107, 127, 108], estimating general traffic data [18] or studying human mobility patterns [50]. It can be also extended for tracking bus journeys. The bus stops sequences corresponding to each line are usually described in terms of the geographical location of each stop. This can be translated to the mobile network domain by mapping each bus stop with the BTSs (cells) from which a user receives signal from that stop, and adding fictitious stops when two consecutive real ones are far apart, so as to ease the tracking. Then, when a user takes a bus, the sequence of BTSs her device is being attached to as the bus moves can be matched with the database to see which bus line the user has taken. Some post-processing is needed in order to detect when the user is actually taking and leaving the bus, or to improve the matching. The main advantage of this method is the complete independence from third parties, both transportation and network operators.
- **Wi-Fi technology** can also be used for data collection purposes, inferring when a user takes or leaves a bus or the trip made by train or subway, taking advantage of the Wi-Fi access provided by some transportation operators in these modes (e.g., Madrid bus lines or New York subway). The raw data obtained (from the Wi-Fi routers or users' phones) should be processed in order to filter events such as users near the bus who are connecting to the bus Wi-Fi network, or

take somehow into account users who do not connect to the AP right when they take the bus, among others. Two main drawbacks derive from this solution. The collaboration of the corresponding transportation operator is required in order to obtain the router logs or, at least, the correspondences between router identifier and the concrete bus (with its timetable and the bus line it corresponds to) carrying that router. The sample size would also be smaller than when using the mobile phone network, as not every mobile phone is Wi-Fi enabled and not every person with a Wi-Fi-enabled phone actually connects to the transportation Internet services.

– The last example of wireless technology that can be leveraged for information collection purposes is **Bluetooth**. Its main feature is the short range, below 10 meters. This allows tracking actions that require more location accuracy. For example, if we would like to know when a user arrives to a bus stop and when she leaves, the coverage area of a BTS or AP is too wide as to be sure of when these events happen. However, with a Bluetooth device installed in each bus stop, as in Madrid bus lines, we could infer that a user arrives at a stop when the stop Bluetooth receiver detects the user's phone and that she leaves when the mobile phone becomes out of range. The main drawback is shared with Wi-Fi technology and related to the sample size, as Bluetooth is not included in every phone, nor used by all people.

The previous examples highlight wireless communication technologies as alternatives to the traditional data collection task. The low cost and effort required, the short time needed to collect the data and the great number of potential passengers that could be involved in the data sets, make of these approaches interesting and feasible techniques that could improve the process and provide data remaining uncovered until today.

Appendix A

Mobility Data Collection Application

Contents

In Chapter 3, the mobility data collection campaign carried out and conducting to the UC3M data set was described. In order to make the data collection possible, an application for mobile devices in charge of collecting the corresponding data was developed. This appendix describes the main design requirements followed, as well as the most interesting details and issues found during its implementation and use.

A.1 Requirements

The main goal of the application is to detect and record all the events concerning changes in the network state, cell changes, received or sent calls, and data traffic events. But the main challenge comes from making this process seamless for the user, both in terms of processing needs and battery drainage. Thus, the most impacting factors to determine the requirements of the application are two: the application should be totally unobtrusive for the user, and it must collect the data without loosing any piece of it.

With these ideas in mind, the resulting requirements of the application are the following ones:

- The application must detect all the events related to the network state, cell changes, incoming and outgoing calls, and incoming and outgoing data traffic.
- The application must record all the data related to each event, together with the timestamp of the moment in which the event happened.

- The application must perform the aforementioned tasks continuously, and it must be executed right after the device in which it is executing is powered on.
- The data collected must be recorded in a compact, yet easy to parse format.
- The data must be frequently sent to a centralized entity that will backup the information of all the users.
- The application must run seamlessly, without disrupting the normal usage of the device.
- All the tasks will be performed consuming as few resources as possible, mainly in terms of battery and processing needs.

A.2 Mobile Phone Platforms

The two main mobile platforms leading the market—Android and iOS—were considered as candidates to develop the application with the requirements previously defined.

The study of Android capabilities revealed the existence of many APIs, some of them used in previously developed applications [117] that allow to detect and record the events under study in background, thus without interfering with the normal use of the device. Since the terminals of the users can be very varied, the minimum version for which the application works if version 8, so that anyone can install it in her device.

An analysis of the iOS platform [146], however, unveiled the impossibility of developing an application like the one required for iPhone devices, except if they are for developer purposes. The APIs related to cell and network information are private, meaning that they are prohibited to be used in applications thought to be available for the general public. Thus, this platform was finally discarded.

A.3 Implementation Details

Once the application requirements and mobile platforms have been analyzed, an Android application was designed. Its main blocks are depicted in Figure A.1, and commented next.

- The **service** is the central part of the application. It is one of components provided by Android, which main characteristic is that it can keep on running on the background, even if the user is not actively interacting with the application. This component allows to continuously run the monitoring and collection process, to capture and store all the events.
- The **event listeners** block comprehend all the elements actively listening for events. They are configured in such a way that when an event takes place, the corresponding listener is triggered and collects the current timestamp and the data associated with the event. The different listeners included in this block are the following ones:

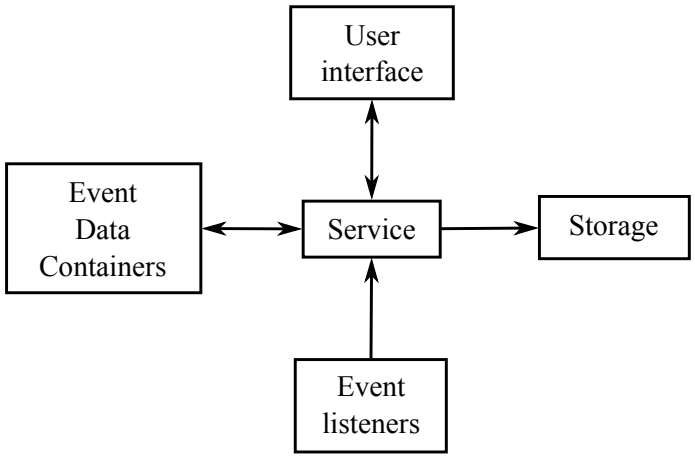


Figure A.1: Block diagram of the data collection application.

  - **Phone state listener**. Actively listens for service state, call, cell changes, signal strength, and data traffic events.
  - **Screen state listener**. It gets notified every time the screen goes on or off. This listener was included to check if the monitoring and recording process kept on executing when the device screen is off.
  - **Power connection listener**. In this case, the function of the listener is not to collect data, but to detect when the suer connects the device to the power line to try to send the recorded data (if at least two weeks have past since the last data upload). This way, the user will not notice the battery drain coming from sending data, aiming to comply with the seamless working requirement.

- The **event data containers** block includes all the containers to which each event data is dumped. These containers hold the data until it is further formatted and recorded into the corresponding file.

- The **storage** block contains all the functionalities needed to write the data regarding each event in a file and transfer it to a centralized entity. These functionalities can be classified as follows:

  - **JSON formatter**. One of the requirements of the application was to record the data in a compact format, easy to be parsed by other applications later on. In order to comply with this, JavaScript Object Notation (JSON) format was selected. Some of the data to be captured, like the data of the different cells from which the device receives signal strength, is variable. Besides, as shown in Table 3.1, different events contain diverse data. for these reasons, JSON seemed a good choice: more compact than eXtensible Markup Language (XML), more readable than Comma Separated Values (CSV), and flexible enough to reflect the different data to be captured. It was also decisive the fact that there exist

JSON parsers for many platforms, like Matlab for instance, which is the one used to further analyze the data collected.

- **File management**. All the functionalities to manage files and dump the collected data into them are included in this block.
- **Data compression**. Considering the increasing size of the file that registers the data associated to the events, these data are compressed before sending them to the centralized entity. In order not to loose data, and relying on the large storage capacity of the current devices, no data is ever deleted. Thus, in case there is an error when sending the data to the centralized entity, no data will be lost because it will be sent again with the next upload. For this reason that complies with the no data loss requirement, and aiming at making the data uploading process as fast as possible to save time and battery, the file is compressed into a zip file using the functionality provided in this block.
- **Data uploading**. In order to obtain the data form the users with their minimal interaction, and guaranteeing that it will not be lost, it was decided to send it to an e-mail account provided by a trusted company. This option guarantees high availability (anyone can send her data at any moment of any day), and no data loss, with a minimum maintenance. In the user's side, once every two weeks, a reminder pops up in the device, inviting the user to send the data. By tapping the notification, the corresponding e-mail application (included by default in all Android devices) opens up with the e-mail and the attached data ready to be sent by just tapping on the send button.

- **User interface**. A simple user interface is provided to manually start or stop the application when needed, and to allow the user to record personal messages into the file.

A.4 Usability and Working Issues Reported by the Users

Before making the application available for all the subjects participating in the collection campaign, a smaller test was performed during two months with a subset of 5 subjects. This initial test led to some modifications:

- In the first version of the application, the data coming form the accelerometer was also recorded every time a new cell changes was detected. This version shown to significantly increase the battery consumption, and thus the accelerometer data collection was discarded.
- The format of the data was modified to obtain a version easier to parse.
- It was checked that the Android issues that did not allow to monitor events when the screen was off had been fixed. Therefore, the continuous data collection was possible.

After this test, the final version was distributed among the participants. There were no reported issues on battery consumption nor usability. The people who stopped using

the application reported to have done so because of a change of device or a reset of the original one, and forgetting to reinstall the application.

Appendix B

Mathematical Demonstrations

Contents

This appendix gathers the mathematical demonstration of the lemmas and theorems formulated in Chapter 6, concerning the privacy preserving mechanisms proposed in a joint work between out research group and the Security Group of the University Polytechnique of Catalonia [120].

B.1 Proof of the Lemma 6.1

Proof. For each t (with $p(t) > 0$) and each s,

$$p_{S'|T}(s|t) = (1-\rho)\, p_{S|T}(s|t) + \rho\, \tfrac{1}{k},$$

where k is the cardinality of the alphabet of S. Due to the concavity of the entropy and the fact that uniform distributions maximize it, for all t,

$$\mathrm{H}(S'|t) \geqslant (1-\rho)\, \mathrm{H}(S|t) + \rho \log k \geqslant \mathrm{H}(S|t),$$

where $\mathrm{H}(S|t)$ denotes the entropy of S given $T = t$, and similarly for S'. Taking expectations on t, $H(S'|T) \geqslant H(S|T)$. Clearly, equality holds only when $\rho = 0$, or else, when S given t is uniformly distributed, regardless of t, i.e., $p(s|t) = \frac{1}{k} = p(s)$. □

B.2 Proof of Theorem 6.1

Proof. We prove the statement for the nontrivial case when $\rho > 0$. In Lemma 6.1, take $S = X_0$, $S' = X'_0$ and $T = (X_{-1}, \ldots, X_{-m})$, thus

$$\mathrm{H}(X'_0|X_{-1}, \ldots, X_{-m}) \geqslant \mathrm{H}(X_0|X_{-1}, \ldots, X_{-m}),$$

with equality if and only if X_0 is uniform and independent of $(X_{-1}, \dots, X_{-m})$. Next, observe that X'_0 and $(X'_{-1}, \dots, X'_{-m})$ are conditionally independent given $(X_{-1}, \dots, X_{-m})$. Apply the conditional-entropy form of the data processing inequality to write

$$\mathrm{H}(X'_0|X'_{-1}, \dots, X'_{-m}) \geqslant \mathrm{H}(X'_0|X_{-1}, \dots, X_{-m}),$$

with equality if and only if X'_0 and $(X_{-1}, \dots, X_{-m})$ are conditionally independent given $(X'_{-1}, \dots, X'_{-m})$. Combine both inequalities to immediately conclude the assertions in the theorem regarding m past samples. The claims on the limit of m for entropy rates follow the same proof, with $S = X_0$, $S' = X'_0$ and $T = (X_{-1}, X_{-2}, \dots)$. □

B.3 Proof of Theorem 6.2

Proof. In uniform replacement, a sample X_n will be effectively perturbed when replacement occurs, with probability ρ, and the replacement sample U_n does not match the original one. Precisely,

$$\delta = \Pr\{X_n \neq X'_n\} = \rho(1 - \Pr\{X_n = U_n\}).$$

Because X_n and U_n are independent and U_n is uniform,

$$\Pr\{U_n = X_n\} = \mathrm{E}_{X_n} \Pr\{U_n = X_n|X_n\} = 1/|\mathscr{X}|.$$

If the original process X is not independent, uniformly distributed, all samples must be replaced to make it so, thereby maximizing the entropy rate. Consequently, $\rho_{\mathrm{crit}} = 1$, and δ_{crit} can be obtained from the relationship between ρ and δ above, simply by setting $\rho = 1$.

As for improved replacement, we resort to Theorem 2 in [110] and the concept of critical redundancy, which takes on the value $1 - \frac{1}{|\mathscr{X}|p_{\max}}$ in the notation of this work. According to this, for any $\rho \geqslant 1 - \frac{1}{|\mathscr{X}|p_{\max}}$, the PMF of the replaced samples R_n is

$$r(x) = \frac{1}{\rho}\frac{1}{|\mathscr{X}|} + \left(1 - \frac{1}{\rho}\right) p(x).$$

Proceeding as in the first part of this proof,

$$\delta = \rho(1 - \Pr\{X_n = R_n\}),$$

but now

$$\Pr\{X_n = R_n\} = \sum_x p(x)\, r(x),$$

from which the expression for δ in the second part of the theorem follows.

For i.i.d. processes, the problem is mathematically equivalent to that formulated in [110], and ρ_{crit} becomes the critical redundancy defined shortly before Theorem 2 in the cited work, in the form expressed in the statement of the theorem we prove here.

The case for processes with memory requires complete replacement to achieve independence of the samples, not merely uniform distribution, just as in the case of uniform replacement. But for $\rho = 1$, the replacement strategy R_n becomes uniform, and the analysis for uniform replacement above applies here as well. □

List of Acronyms

AP Access Point

API Application Programming Interface

BTS Base Transceiver Station

CDMA Code Division Multiple Access

CDR Call Detail Record

CellID Cell Identifier

CSV Comma Separated Values

DDR Data Detail Record

GPS Global Positioning System

GSM Global System for Mobile Communications

HMM Hidden Markov Model

HSPA High Speed Packet Access

JSON JavaScript Object Notation

LAC Location Area Code

LBS Location-Based Service

LBSN Location-Based Social Network

LTE Long Term Evolution

M2M Machine-to-Machine

MABR Minimum Area Bounding Rectangle

MAC Medium Access Control

MANET Mobile Ad-Hoc Network

MCC Mobile Country Code

MIT Massachusetts Institute of Technology

MNC Mobile Network Code

PET Privacy-Enhancing Technology

PMF Probability Mass Function

POI Point Of Interest

PPM Prediction by Partial Matching

RFID Radio Frequency IDentification

SDC Statistical Disclosure Control

UC3M University Carlos III of Madrid

UMTS Universal Mobile Telecommunications System

XML eXtensible Markup Language

References

[1] S. Akoush and A. Sameh. Mobile user movement prediction using bayesian learning for neural networks. In *Proceedings of the 2007 ACM International Conference on Wireless Communications and Mobile Computing (IWCMC '07)*, pages 191–196. ACM, 2007. 19, 20

[2] M. Alfalayleh and L. Brankovic. Quantifying privacy: A novel entropy-based measure of disclosure risk. *arXiv preprint arXiv:1409.2112*, 2014. 122

[3] J. A. Alvarez-Garcia, J. A. Ortega, L. Gonzalez-Abril, and F. Velasco. Trip destination prediction based on past gps log using a hidden markov model. *Expert Systems with Applications*, 37(12):8166–8171, 2010. 18, 19, 20

[4] T. Anagnostopoulos, C. Anagnostopoulos, and S. Hadjiefthymiades. Efficient location prediction in mobile cellular networks. *International Journal of Wireless Information Networks*, 19(2):97–111, 2012. 19

[5] C. A. Ardagna, M. Cremonini, E. Damiani, S. De Capitani di Vimercati, and P. Samarati. Location privacy protection through obfuscation-based techniques. In *Proceedings of the 21st Annual IFIP Working Conference on Data and Applications Security*, volume 4602 of *Lecture Notes in Computer Science (LNCS)*, pages 47–60, Redondo Beach, CA, United States, July 2007. Springer-Verlag. 122

[6] D. Ashbrook and T. Starner. Using gps to learn significant locations and predict movement across multiple users. *Personal and Ubiquitous Computing*, 7(5):275–286, 2003. 18, 20

[7] I. M. Averin, V. T. Ermolayev, and A. G. Flaksman. Locating mobile users using base stations of cellular networks. *Communications and Networks*, 2:216–220, September 2010. 9

[8] P. Baumann, W. Kleiminger, and S. Santini. The influence of temporal and spatial features on the performance of next-place prediction algorithms. In *Proceedings of the 2013 ACM International Conference on Pervasive and Ubiquitous Computing (UbiComp '13)*, pages 449–458. ACM, 2013. 20

[9] R. Becker, R. Cáceres, K. Hanson, S. Isaacman, J. M. Loh, M. Martonosi, J. Rowland, S. Urbanek, A. Varshavsky, and C. Volinsky. Human mobility characterization from cellular network data. *Communications of the ACM*, 56(1):74–82, 2013. 1, 11

[10] R. A Becker, R. Cáceres, K. Hanson, J. M. Loh, S. Urbanek, A. Varshavsky, and C. Volinsky. A tale of one city: Using cellular network data for urban planning. *IEEE Pervasive Computing*, 10(4):0018–26, 2011. 11, 13

[11] M. Berlingerio, F. Calabrese, G. Di Lorenzo, R. Nair, F. Pinelli, and M. L. Sbodio. Allaboard: a system for exploring urban mobility and optimizing public transport using cellphone data. In *Machine Learning and Knowledge Discovery in Databases*, pages 663–666. Springer, 2013. 11, 13

[12] A. Bhattacharya and S. K. Das. Lezi-update: An information-theoretic framework for personal mobility tracking in pcs networks. *Wireless Networks*, 8(2/3):121–135, 2002. 20, 81, 83, 101

[13] F. Bohnert and I. Zukerman. Personalised pathway prediction. In *User Modeling, Adaptation, and Personalization*, pages 363–368. Springer, 2010. 20

[14] C. Boldrini and A. Passarella. Hcmm: Modelling spatial and temporal properties of human mobility driven by users' social relationships. *Computer Communications*, 33(9):1056–1074, 2010. 2, 15

[15] N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi. Optimal geo-indistinguishable mechanisms for location privacy. In *Proceedings of the 2014 ACM Conference on Computer and Communications Security (SIGSAC '14)*, pages 251–262. ACM, 2014. 121

[16] D. Brockmann, L. Hufnagel, and T. Geisel. The scaling laws of human travel. *Nature*, 439(7075):462–465, 2006. 11, 15

[17] N. Caceres, J. P. Wideberg, and F. G. Benitez. Deriving origin destination data from a mobile phone network. *IET Intelligent Transport Systems*, 1(1):15–26, 2007. 150

[18] N. Caceres, J. P. Wideberg, and F. G. Benitez. Review of traffic data estimations extracted from cellular networks. *IET Intelligent Transport Systems*, 2(3):179–192, 2008. 150

[19] F. Calabrese, G. D. Lorenzo, and C. Ratti. Human mobility prediction based on individual and collective geographical preferences. In *Proceedings of the 13th IEEE International Conference on Intelligent Transportation Systems (ITSC 2010)*, pages 312–317. IEEE, 2010. 18, 20

[20] K. Chatzikokolakis, C. Palamidessi, and M. Stronati. A predictive differentially-private mechanism for mobility traces. In Emiliano De Cristofaro and StevenJ. Murdoch, editors, *Privacy Enhancing Technologies*, volume 8555 of *Lecture Notes in Computer Science*, pages 21–41. Springer International Publishing, 2014. 121

[21] M. Chen, X. Yu, and Y. Liu. Mining moving patterns for predicting next location. *Information Systems*, 54:156–168, 2015. 20

[22] R. Chen, G. Acs, and C. Castelluccia. Differentially private sequential data publication via variable-length n-grams. In *Proceedings of the 2012 ACM Conference on Computer and Communications Security (CCS '12)*, pages 638–649. ACM, 2012. 121

[23] C. Cheng, R. Jain, and E. van den Berg. Location prediction algorithms for mobile wireless systems. In *Wireless Internet Handbook*, pages 245–263. CRC Press, Inc., 2003. 79

[24] E. Cho, S. A Myers, and J. Leskovec. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD '11)*, pages 1082–1090. ACM, 2011. 15

[25] S.-B. Cho. Exploiting machine learning techniques for location recognition and prediction with smartphone logs. *Neurocomputing*, 2015. 18, 19, 20

[26] J. G. Cleary and W. J. Teahan. Unbounded length contexts for ppm. *The Computer Journal*, 40(2 and 3):67–75, 1997. 20, 83

[27] V. Colizza, A. Barrat, M. Barthelemy, A.-J. Valleron, and A. Vespignani. Modeling the worldwide spread of pandemic influenza: baseline case and containment interventions. *PLoS Medicine*, 4(1):95, 2007. 1

[28] I. Constandache, S. Gaonkar, M. Sayler, R.R. Choudhury, and L. Cox. Enloc: Energy-efficient localization for mobile phones. In *Proceedings of the 28th IEEE International Conference on Computer Communications (INFOCOM 2009)*, pages 2716 –2720, April 2009. 8

[29] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, New York, second edition, 2006. 21, 122

[30] G. Danezis. Introduction to privacy technology. Research talk, Katholieke University Leuven, Computer Security and Industrial Cryptography (COSIC), 2007. 121

[31] Y.-A. de Montjoye, C. A. Hidalgo, M. Verleysen, and V. D. Blondel. Unique in the Crowd: The privacy bounds of human mobility. *Scientific Reports*, 3, March 2013. 124

[32] Y. De Mulder, G. Danezis, L. Batina, and B. Preneel. Identification via location-profiling in gsm networks. In *Proceedings of the 7th ACM Workshop on Privacy in the Electronic Society (CCS '08)*, pages 23–32. ACM, 2008. 120

[33] C. Díaz. *Anonymity and Privacy in Electronic Services*. PhD thesis, Katholieke University Leuven, December 2005. 122

[34] C. Díaz, S. Seys, J. Claessens, and B. Preneel. Towards measuring anonymity. In *Proceedings of the 2002 Workshop on Privacy Enhancing Technologies (PET 2002)*, volume 2482 of *Lecture Notes Computer Science (LNCS)*, pages 54–68. Springer-Verlag, April 2002. 122

[35] R. Dingledine. Free Haven's anonymity bibliography, 2009. 121

[36] G.M. Djuknic and R.E. Richton. Geolocation and assisted gps. *Computer*, 34(2):123 –125, February 2001. 8

[37] J. Doyle, P. Hung, R. Farrell, and S. McLoone. Population mobility dynamics estimated from mobile telephony data. *Journal of Urban Technology*, 21(2):109–132, 2014. 11, 14

[38] M. Duckham and L. Kulit. A formal model of obfuscation and negotiation for location privacy. In *Proceedings of the 3rd International Conferences on Pervasive Computing*, volume 3468 of *Lecture Notes in Computer Science (LNCS)*, pages 152–170, Munich, Germany, May 2005. Springer-Verlag. 121

[39] M. Duckham, K. Mason, J. Stell, and M. Worboys. A formal approach to imperfection in geographic information. *Computers, Environment and Urban Systems*, 25(1):89–103, 2001. 121

[40] C. Dwork. Differential privacy. In *Encyclopedia of Cryptography and Security*, pages 338–340. Springer, 2011. 121

[41] N. Eagle, A. Clauset, and J. A. Quinn. Location segmentation, inference and prediction for anticipatory computing. In *AAAI Spring Symposium: Technosocial Predictive Analytics*, pages 20–25, 2009. 12, 15

[42] N. Eagle, Y. de Montjoye, and L. M. A. Bettencourt. Community c omputing: Comparisons between rural and urban societies using mobile phone data. In *Proceedings of the International Conference on Computational Science and Engineering (CSE'09)*, volume 4, pages 144–150. IEEE, 2009. 11

[43] N. Eagle and A. Pentland. Reality mining: Sensing complex social systems. *Personal and Ubiquitous Computing*, 10(4):255–268, 2006. 12, 37

[44] Y. Elovici, C. Glezer, and B. Shapira. Enhancing customer privacy while searching for products and services on the World Wide Web. *Internet Research*, 15(4):378–399, 2005. 122

[45] L.M. Feeney and M. Nilsson. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In *Proceedings of the 20th IEEE International Conference on Computer Communications (INFOCOM 2001)*, volume 3, pages 1548 –1557 vol.3, 2001. 9

[46] M. Ficek. CRAWDAD data set ctu/personal (v. 2012-03-15). Downloaded from http://crawdad.org/ctu/personal/. (last access: October 26th, 2015), March 2012. 11

[47] J. Freudiger, R. Shokri, and J.-P. Hubaux. Evaluating the privacy risk of location-based services. In *Proceedings of the 15th International Conference on Financial*

Cryptography and Data Security (FC '11), FC'11, pages 31–46, Berlin, Heidelberg, 2012. Springer-Verlag. 120

[48] W. Gao and G. Cao. Fine-grained mobility characterization: steady and transient state behaviors. In *Proceedings of the 11th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '10)*, pages 61–70. ACM, 2010. 11

[49] Y. Gao, I. Kontoyiannis, and E. Bienenstock. Estimating the entropy of binary time series: methodology, some theory and a simulation study. *Entropy*, 10(2):71–99, 2008. 21

[50] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, June 2008. 11, 15, 16, 20, 124, 150

[51] K. Gopalratnam and D. J. Cook. Online sequential prediction via incremental parsing: The active lezi algorithm. *IEEE Intelligent Systems*, 22(1):52–58, 2007. 20, 84

[52] P. Grassberger. Estimating the information content of symbol sequences and efficient codes. *IEEE Transactions on Information Theory*, 35(3):669–675, 1989. 27

[53] S. Grauwin, S. Sobolevsky, S. Moritz, I. Gódor, and C. Ratti. Towards a comparative science of cities: Using mobile traffic records in new york, london, and hong kong. In *Computational Approaches for Urban Environments*, pages 363–387. Springer, 2015. 14

[54] H. He, Y. Qiao, S. Gao, J. Yang, and J. Guo. Prediction of user mobility pattern on a network traffic analysis platform. In *Proceedings of the 10th International Workshop on Mobility in the Evolving Internet Architecture (MobiCom '15)*, pages 39–44. ACM, 2015. 19, 20

[55] J. Hightower and G. Borriello. Location systems for ubiquitous computing. *Computer*, 34(8):57 –66, August 2001. 9

[56] S.-S. Ho and S. Ruan. Differential privacy for location pattern mining. In *Proceedings of the 4th ACM International Workshop on Security and Privacy in GIS and LBS (SIGSPATIAL '11)*, pages 17–24. ACM, 2011. 121

[57] L. Hufnagel, D. Brockmann, and T. Geisel. Forecast and control of epidemics in a globalized world. *Proceedings of the National Academy of Sciences of the United States of America*, 101(42):15124–15129, 2004. 1

[58] S. Isaacman, R. Becker, R. Cáceres, S. Kobourov, M. Martonosi, J. Rowland, and A. Varshavsky. Identifying important places in people's lives from cellular network data. In *Pervasive Computing*, pages 133–151. Springer, 2011. 11, 13, 14

[59] S. Isaacman, R. Becker, R. Cáceres, S. Kobourov, M. Martonosi, J. Rowland, and A. Varshavsky. Ranges of human mobility in los angeles and new york. In *Proceedings*

of the IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), pages 88–93. IEEE, 2011. 11, 13

[60] S. Isaacman, R. Becker, R. Cáceres, S. Kobourov, J. Rowland, and A. Varshavsky. A tale of two cities. In *Proceedings of the 11th Workshop on Mobile Computing Systems & Applications*, pages 19–24. ACM, 2010. 11, 13

[61] S. Isaacman, R. Becker, R. Cáceres, M. Martonosi, J. Rowland, A. Varshavsky, and W. Willinger. Human mobility modeling at metropolitan scales. In *Proceedings of the 10th International Conference on Mobile systems, Applications and Services*, pages 239–252. ACM, 2012. 11, 13

[62] B. S. Jensen, J. E. Larsen, K. Jensen, J. Larsen, and L. K. Hansen. Estimating human predictability from mobile sensor data. In *Proceedings of the 20th IEEE International Workshop on Machine Learning for Signal Processing (MLSP 2010)*, pages 196–201. IEEE, 2010. 12, 16

[63] J. Jeong, M. Leconte, and A. Proutiere. Mobility prediction using non-parametric bayesian model. *CoRR*, abs/1507.03292, 2015. 19, 20

[64] D. B. Johnson and D. A. Maltz. Dynamic source routing in ad hoc wireless networks. In *Mobile computing*, pages 153–181. Springer, 1996. 2

[65] C. Kang, X. Ma, D. Tong, and Y. Liu. Intra-urban human mobility patterns: An urban morphology perspective. *Physica A: Statistical Mechanics and its Applications*, 391(4):1702–1717, 2012. 14

[66] D. Karamshuk, C. Boldrini, M. Conti, and A. Passarella. Human mobility models for opportunistic networks. *IEEE Communications Magazine*, 49(12):157–165, 2011. 15

[67] A. Khan, Sk. K. A. Imon, and S. K. Das. An energy efficient framework for localization and coverage in participatory urban sensing. In *Proceedings of the 39th IEEE Conference on Local Computer Networks (LCN 2014)*, pages 193–201. IEEE, 2014. 8

[68] A. Khan, Sk. K. A. Imon, and S. K. Das. Ensuring energy efficient coverage for participatory sensing in urban streets. In *Proceedings of the 2014 International Conference on Smart Computing (SMARTCOMP 2014)*, pages 167–174. IEEE, 2014. 8

[69] A. Khan, Sk. K. A. Imon, and S. K. Das. A novel localization and coverage framework for real-time participatory urban monitoring. *Pervasive and Mobile Computing*, 23:122–138, 2015. 8

[70] M. Kim, D. Kotz, and S. Kim. Extracting a mobility model from real user traces. In *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM 2006)*, volume 6, pages 1–13, Barcelona, Spain, April 2006. 11

[71] Y.-J. Kim and S.-B. Cho. A hmm-based location prediction framework with location recognizer combining k-nearest neighbor and multiple decision trees. In *Hybrid Artificial Intelligent Systems*, pages 618–628. Springer, 2013. 18, 19, 20

[72] N. Kiukkonen, J. Blom, O. Dousse, D. Gatica-Perez, and J. Laurila. Towards rich mobile phone datasets: Lausanne data collection campaign. In *Proceedings of the 7th International Conference on Pervasive Services (ICPS 2010)*, Berlin, Germany, 2010. 12

[73] P. Kontkanen, P. Myllymaki, T. Roos, H. Tirri, K. Valtonen, and H. Wettig. Topics in probabilistic location estimation in wireless networks. In *Proceedings of the 15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2004)*, volume 2, pages 1052 – 1056 Vol.2, September 2004. 9

[74] I. Kontoyiannis, P. H. Algoet, Y. M. Suhov, and A. J. Wyner. Nonparametric entropy estimation for stationary processes and random fields, with applications to english text. *IEEE Transactions on Information Theory*, 44(3):1319–1327, 1998. 27

[75] T. Kuflik, B. Shapira, Y. Elovici, and A. Maschiach. Privacy preservation improvement by learning optimal profile generation rate. In *User Modeling*, volume 2702 of *Lecture Notes in Computer Science (LNCS)*, pages 168–177. Springer-Verlag, 2003. 122

[76] K. Laasonen. Clustering and prediction of mobile user routes from cellular data. In *Knowledge Discovery in Databases: PKDD 2005*, pages 569–576. Springer, 2005. 19, 20

[77] A. LaMarca, Y. Chawathe, S. Consolvo, J. Hightower, I. Smith, J. Scott, T. Sohn, J. Howard, J. Hughes, F. Potter, J. Tabert, P. Powledge, G. Borriello, and B. Schilit. Place lab: Device positioning using radio beacons in the wild. In *Pervasive Computing*, pages 116–133. Springer, 2005. 12

[78] J. K Laurila, D. Gatica-Perez, I. Aad, O. Bornet, T.-M.-T. Do, O. Dousse, J. Eberle, and M. Miettinen. The mobile data challenge: Big data for mobile computing research. In *Nokia Mobile Data Challenge 2012 Workshop*, Newcastle, UK, June 2012. 12

[79] J.-K. Lee and J. C. Hou. Modeling steady-state and transient behaviors of user mobility: formulation, analysis, and application. In *Proceedings of the 7th ACM international Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '06)*, pages 85–96. ACM, 2006. 11, 17, 61

[80] K. Lee, S. Hong, S. J. Kim, I. Rhee, and S. Chong. Slaw: A new mobility model for human walks. In *Proceedings of the 27th IEEE Conference on Computer Communications (INFOCOM 2009)*, pages 855–863. IEEE, 2009. 1, 149

[81] N. Li, T. Li, and S. Venkatasubramanian. t-Closeness: Privacy beyond k-anonymity and l-diversity. In *Proceedings of the IEEE International Conference on Data Engineering (ICDE 2007)*, pages 106–115, Istanbul, Turkey, April 2007. 123

[82] D. Lian, X. Xie, V. W. Zheng, N. J. Yuan, F. Zhang, and E. Chen. Cepr: A collaborative exploration and periodically returning model for location prediction. *ACM Transactions on Intelligent Systems and Technology*, 6(1):8, 2015. 18, 20

[83] X. Liang, X. Zheng, W. Lv, T. Zhu, and K. Xu. The scaling of human mobility by taxis is exponential. *Physics A: Statistical Mechanics and its Applications*, 391(5):2135–2144, 2012. 11

[84] M. Lin and W.-J. Hsu. Mining gps data for mobility patterns: A survey. *Pervasive and Mobile Computing*, 12:1–16, 2014. 11, 15, 17

[85] M. Lin, W.-J. Hsu, and Z. Q. Lee. Predictability of individuals' mobility with high-resolution positioning data. In *Proceedings of the 14th ACM International Conference on Ubiquitous Computing (UbiComp '12)*, pages 381–390, Pittsburgh, Pennsylvania, United States, 2012. ACM. 11, 16

[86] M. Lin, W.-J. Hsu, and Z. Q. Lee. Modeling high predictability and scaling laws of human mobility. In *Proceedings of the 14th IEEE International Conference on Mobile Data Management (MDM 2013)*, volume 2, pages 125–130. IEEE, 2013. 16

[87] M. Lin, Z. Q. Lee, and W.-J. Hsu. Uncovering temporal and spatial localities in individuals' mobility. In *Proceedings of the 14th IEEE International Conference on Mobile Data Management (MDM 2013)*, volume 1, pages 257–262. IEEE, 2013. 11, 16

[88] X. Lu, E. Wetter, N. Bharti, A. J. Tatem, and L. Bengtsson. Approaching the limit of predictability in human mobility. *Scientific Reports*, 3, 2013. 11, 16

[89] Q. Lv, Y. Di, Y. Qiao, Z. Lei, and C. Dong. Spatial and temporal mobility analysis in lte mobile network. In *Proceedings of the 2015 IEEE Wireless Communications and Networking Conference (WCNC 2015)*, pages 795–800. IEEE, 2015. 19, 20

[90] A. Machanavajjhala, J. Gehrke, D. Kiefer, and M. Venkitasubramanian. l-Diversity: Privacy beyond k-anonymity. In *Proceedings of IEEE International Conference on Data Engineering (ICDE 2006)*, page 24, Atlanta, GA, United States, April 2006. 121

[91] W. Mathew, R. Raposo, and B. Martins. Predicting future locations with hidden markov models. In *Proceedings of the 14th ACM International Conference on Ubiquitous Computing (UbiComp '12)*, pages 911–918. ACM, 2012. 19, 20

[92] J. McInerney, S. Stein, A. Rogers, and N. R. Jennings. Exploring periods of low predictability in daily life mobility. In *Nokia Mobile Data Challenge 2012 Workshop*, Newcastle, UK, June 2012. 102

[93] A. Monreale, F. Pinelli, R. Trasarti, and F. Giannotti. Wherenext: a location predictor on trajectory pattern mining. In *Proceedings of the 15th ACM international Conference on Knowledge Discovery and Data Mining (SIGKDD '09)*, pages 637–646. ACM, 2009. 18, 20

[94] M. Morzy. Mining frequent trajectories of moving objects for location prediction. In *Machine Learning and Data Mining in Pattern Recognition*, pages 667–680. Springer, 2007. 18

[95] A. Noulas, S. Scellato, R. Lambiotte, M. Pontil, and C. Mascolo. A tale of many cities: universal patterns in human urban mobility. *PloS One*, 7(5):e37027, 2012. 11, 13

[96] A. Noulas, S. Scellato, N. Lathia, and C. Mascolo. Mining user mobility features for next place prediction in location-based services. In *Proceedings of the 12th IEEE International Conference on Data Mining (ICDM 2012)*, pages 1038–1043. IEEE, 2012. 18

[97] A. Oganian and J. Domingo-Ferrer. A posteriori disclosure risk measure for tabular data based on conditional entropy. *SORT. 2003*, 27(2), 2003. 122

[98] A.-M. Olteanu, K. Huguenin, R. Shokri, and J.-P. Hubaux. Quantifying the effect of co-location information on location privacy. In *Privacy Enhancing Technologies*, Lecture Notes in Computer Science, pages 184–203. Springer International Publishing, 2014. 121

[99] J. R. B. Palmer, T. J. Espenshade, F. Bartumeus, C. Y Chung, N. E. Ozgencil, and K. Li. New approaches to human mobility: using mobile phones for demographic research. *Demography*, 50(3):1105–1128, 2013. 14

[100] L. Pappalardo, F. Simini, S. Rinzivillo, D. Pedreschi, F. Giannotti, and A.-L. Barabási. Returners and explorers dichotomy in human mobility. *Nature Communications*, 6:8166+, September 2015. 16

[101] J. Parra-Arnau, D. Rebollo-Monedero, and J. Forné. Optimal forgery and suppression of ratings for privacy enhancement in recommendation systems. *Entropy*, 16(3):1586–1631, March 2014. 122

[102] J. Parra-Arnau, D. Rebollo-Monedero, J. Forné, J. L. Muñoz, and O. Esparza. Optimal tag suppression for privacy protection in the semantic Web. *Data and Knowledge Engineering*, 81–82:46–66, November 2012. 124

[103] K. Pearson. The problem of the random walk. *Nature*, 72(1865):294, 1905. 2

[104] S.K. Pulliyakode and S. Kalyani. A modified ppm algorithm for online sequence prediction using short data records. *IEEE Communications Letters*, 19(3):423–426, March 2015. 20

[105] A. Rahmati and L. Zhong. Context-based network estimation for energy-efficient ubiquitous wireless connectivity. *IEEE Transactions on Mobile Computing*, 10(1):54–66, January 2011. 12

[106] G. Ranjan, H. Zang, Z.-L. Zhang, and J. Bolot. Are call detail records biased for sampling human mobility? *ACM SIGMOBILE Mobile Computing and Communications Review*, 16(3):33–44, 2012. 11, 17, 76

[107] C. Ratti, S. Williams, D. Frenchman, and R. M. Pulselli. Mobile landscapes: using location data from cell phones for urban analysis. *Environment and Planning B Planning and Design*, 33(5):727, 2006. 150

[108] J. Reades, F. Calabrese, and C. Ratti. Eigenplaces: analysing cities using the space-time structure of the mobile phone network. *Environment and Planning B: Planning and Design*, 36(5):824–836, 2009. 150

[109] J. Reades, F. Calabrese, A. Sevtsuk, and C. Ratti. Cellular census: Explorations in urban data collection. *IEEE Pervasive Computing*, 6(3):30–38, 2007. 13

[110] D. Rebollo-Monedero and J. Forné. Optimal query forgery for private information retrieval. *IEEE Transactions on Information Theory*, 56(9):4631–4642, 2010. 122, 123, 124, 128, 131, 160

[111] D. Rebollo-Monedero, J. Forné, and J. Domingo-Ferrer. From t-closeness-like privacy to postrandomization via information theory. *IEEE Transactions on Knowledge and Data Engineering*, 22(11):1623–1636, November 2010. 123

[112] D. Rebollo-Monedero, J. Parra-Arnau, C. Diaz, and J. Forné. On the measurement of privacy as an attacker's estimation error. *International Journal of Information Security*, 12(2):129–149, April 2013. 122

[113] I. Rhee, M. Shin, S. Hong, K. Lee, S. J. Kim, and S. Chong. On the levy-walk nature of human mobility. *IEEE/ACM Transactions on Networking*, 19(3):630–643, 2011. 1, 2, 11, 15, 51

[114] A. Rodriguez-Carrion, C. Campo, and C. Garcia-Rubio. Recommendations on the move. In *Recommender Systems for the Social Web*, pages 179–193. Springer, 2012. 146

[115] A. Rodriguez-Carrion, S. K. Das, C. Campo, and C. Garcia-Rubio. Impact of location history collection schemes on observed human mobility features. In *Proceedings of the 2014 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pages 254–259. IEEE, 2014. 145

[116] A. Rodriguez-Carrion, C. Garcia-Rubio, and C. Campo. Performance evaluation of lz-based location prediction algorithms in cellular networks. *IEEE Communications Letters*, 14(8):707–709, 2010. 146

[117] A. Rodriguez-Carrion, C. Garcia-Rubio, C. Campo, A. Cortés-Martín, E. Garcia-Lozano, and P. Noriega-Vivas. Study of lz-based location prediction and its application to transportation recommender systems. *Sensors*, 12(6):7496–7517, 2012. 146, 154

[118] A. Rodriguez-Carrion, C. Garcia-Rubio, C. Campo, Alberto Cortés-Martín, E. Garcia-Lozano, and P. Noriega-Vivas. Analysis of location prediction performance of lz algorithms using gsm cell-based location data. In *Proceedings of the 5th International Symposium on Ubiquitous Computing and Ambient Intelligence (UCAmI 2011)*, 2011. 146

[119] A. Rodriguez-Carrion, C. Garcia-Rubio, C. Campo, and S. K. Das. Analysis of a fast lz-based entropy estimator for mobility data. In *Proceedings of the 2015 IEEE International Conference Pervasive Computing and Communication Workshops (PerCom Workshops)*, pages 451–456. IEEE, 2015. 146

[120] A. Rodriguez-Carrion, D. Rebollo-Monedero, J. Forné, C. Campo, C. Garcia-Rubio, J. Parra-Arnau, and S. K. Das. Entropy-based privacy against profiling of user mobility. *Entropy*, 17(6):3913–3946, 2015. 146, 159

[121] A. Sadilek and J. Krumm. Far out: Predicting long-term human mobility. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, Toronto, Canada, July 2012. AAAI Press. 11, 20

[122] P. Samarati. Protecting respondents' identities in microdata release. *IEEE Transactions on Knowledge and Data Engineering*, 13(6):1010–1027, 2001. 121

[123] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-Anonymity and its enforcement through generalization and suppression. Technical report, SRI Int., 1998. 121

[124] S. Scellato, M. Musolesi, C. Mascolo, V. Latora, and A. T. Campbell. Nextplace: a spatio-temporal prediction framework for pervasive systems. In *Pervasive Computing*, pages 152–169. Springer, 2011. 18, 20

[125] D. Schulz, S. Bothe, and C. Körner. Human mobility from gsm data-a valid alternative to gps. In *Nokia Mobile Data Challenge 2012 Workshop*, Newcastle, UK, June 2012. 17

[126] A. Serjantov and G. Danezis. Towards an information theoretic metric for anonymity. In *Proceedings of the 2002 Workshop on Privacy Enhancing Technologies (PET 2002)*, volume 2482, pages 41–53. Springer-Verlag, 2002. 122

[127] A. Sevtsuk and C. Ratti. Does urban mobility have a daily routine? learning from the aggregate data of mobile networks. *Journal of Urban Technology*, 17(1):41–60, 2010. 150

[128] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379–423, 1948. 21

[129] B. Shapira, Y. Elovici, A. Meshiach, and T. Kuflik. PRAW – The model for PRivAte Web. *Journal of the American Society for Information Science and Technology*, 56(2):159–172, 2005. 122

[130] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, and J-P Hubaux. Quantifying location privacy. In *Proceedings of the 2011 IEEE Symposium on Security and Privacy (SP 2011)*, pages 247–262, May 2011. 120

[131] R. Shokri, G. Theodorakopoulos, P. Papadimitratos, E. Kazemi, and J.-P. Hubaux. Hiding in the mobile crowd: Location privacy through collaboration. *IEEE Transactions on Dependable and Secure Computing*, 11(3):266–279, 2014. 121

[132] K. Sohn and D. Kim. Dynamic origin–destination flow estimation using cellular communication system. *IEEE Transactions on Vehicular Technology*, 57(5):2703–2713, 2008. 150

[133] C. Song, T. Koren, P. Wang, and A.-L. Barabási. Modelling the scaling properties of human mobility. *Nature Physics*, 6(10):818–823, 2010. 11, 15, 16, 20, 76

[134] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási. Limits of Predictability in Human Mobility. *Science*, 327(5968):1018–1021, February 2010. 11, 16, 20, 48, 58, 59, 76, 124, 126

[135] L. Song, U. Deshpande, U. C. Kozat, D. Kotz, and R. Jain. Predictability of wlan mobility and its effects on bandwidth provisioning. In *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM 2006)*, 2006. 19

[136] L. Song, D. Kotz, R. Jain, and X. He. Evaluating next-cell predictors with extensive wi-fi mobility data. *IEEE Transactions on Mobile Computing*, 5(12):1633–1649, 2006. 20, 89, 95

[137] M. Spitz. CRAWDAD data set spitz/cellular (v. 2011-05-04). Downloaded from http://crawdad.org/spitz/cellular/ (last access: October 26th, 2015), May 2011. 11

[138] A. Sridharan and J. Bolot. Location patterns of mobile users: A large-scale study. In *Proceedings of the 32nd IEEE International Conference on Computer Communications (INFOCOM 2013)*, pages 1007–1015. IEEE, 2013. 11, 14, 76

[139] J. Steenbruggen, E. Tranos, and P. Nijkamp. Data from mobile phone operators: A tool for smarter cities? *Telecommunications Policy*, 39(3):335–346, 2015. 13

[140] A. Striegel, S. Liu, L. Meng, C. Poellabauer, D. Hachen, and O. Lizardo. Lessons learned from the netsense smartphone study. *SIGCOMM Computer Communication Review*, 43(4):51–56, August 2013. 12

[141] J. B. Sun, J. Yuan, Y. Wang, H. B. Si, and X. M. Shan. Exploring space–time structure of human mobility in urban space. *Physica A: Statistical Mechanics and its Applications*, 390(5):929–942, 2011. 13

[142] X. Sun, H. Wang, J. Li, and T. M. Truta. Enhanced p-sensitive k-anonymity models for privacy preserving data publishing. *Transactions on Data Privacy*, 1(2):53–66, 2008. 121

[143] G. Theodorakopoulos, R. Shokri, C. Troncoso, J.-P. Hubaux, and J.-Y. Le Boudec. Prolonging the hide-and-seek game: Optimal trajectory privacy for location-based services. In *Proceedings of the 13th ACM Workshop on Privacy in the Electronic Society*, pages 73–82. ACM, 2014. 122

[144] J. L. Toole, Y.-A. de Montjoye, M. C. González, and A. Pentland. Modeling and understanding intrinsic characteristics of human mobility. In Bruno Gonçalves and Nicola Perra, editors, *Social Phenomena*, Computational Social Sciences, pages 15–35. Springer International Publishing, 2015. 15

[145] T. M. Truta and B. Vinay. Privacy protection: p-Sensitive k-anonymity property. In *Proceedings of the International Workshop on Privacy Data Management (PDM 2006)*, page 94, Atlanta, GA, United States, 2006. 121

[146] M. P. Ventero Peña. *Analysis and Implementation of an iOS Application to Collect Mobility Data Based on Wireless Networks.* Bachelor's thesis, University Carlos III Madrid, March 2015. 154

[147] J. S. Vitter and P. Krishnan. Optimal prefetching via data compression. *Journal of the ACM*, 43(5):771–793, 1996. 82

[148] A. S. Voulodimos and C. Z. Patrikakis. Quantifying privacy in terms of entropy for context aware services. *Identity in the Information Society*, 2(2):155–169, 2009. 122

[149] P. Wang, M. C. González, C. A Hidalgo, and A.-L. Barabási. Understanding the spreading patterns of mobile phone viruses. *Science*, 324(5930):1071–1076, 2009. 1

[150] P. Wang, T. Hunter, A. M. Bayen, K. Schechtner, and M. C. González. Understanding road usage patterns in urban areas. *Scientific Reports*, 2, 2012. 13

[151] Y. Wang, N. J. Yuan, D. Lian, L. Xu, X. Xie, E. Chen, and Y. Rui. Regularity and conformity: Location prediction using heterogeneous mobility data. In *Proceedings of the 21th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD '15)*, pages 1275–1284. ACM, 2015. 18, 20

[152] S. B Wicker. The loss of location privacy in the cellular age. *Communications of the ACM*, 55(8):60–68, 2012. 120

[153] W. Wu, Y. Wang, J. B. Gomes, D. T. Anh, S. Antonatos, M. Xue, P. Yang, G. E. Yap, X. Li, and S. Krishnaswamy. Oscillation resolution for mobile phone cellular tower

data to enable mobility modelling. In *Proceedings of the 15th IEEE International Conference on Mobile Data Management (MDM 2014)*, volume 1, pages 321–328. IEEE, 2014. 17

[154] G. Yavaş, D. Katsaros, Ö. Ulusoy, and Y. Manolopoulos. A data mining approach for location prediction in mobile environments. *Data & Knowledge Engineering*, 54(2):121–146, 2005. 19

[155] J. J.-C. Ying, W.-C. Lee, and V. S. Tseng. Mining geographic-temporal-semantic patterns in trajectories for location prediction. *ACM Transactions on Intelligent Systems and Technology*, 5(1):2, 2013. 18, 19

[156] M. L. Yiu, C. S. Jensen, X. Huang, and H. Lu. SpaceTwist: Managing the trade-offs among location privacy, query performance, and query accuracy in mobile services. In *Proceedings of IEEE International Conference on Data Engineering (ICDE 2008)*, pages 366–375, Cancun, Mexico, April 2008. 122

[157] J. Yuan, Y. Zheng, and X. Xie. Discovering regions of different functions in a city using human mobility and pois. In *Proceedings of the 18th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD '12)*, pages 186–194. ACM, 2012. 14

[158] Y. Yuan and M. Raubal. Extracting dynamic urban mobility patterns from mobile phone data. In *Geographic Information Science*, pages 354–367. Springer, 2012. 14

[159] Y. Zhao. Standardization of mobile phone positioning for 3g systems. *IEEE Communications Magazine*, 40(7):108 –116, jul 2002. 8, 9

[160] Y. Zheng, Q. Li, Y. Chen, X. Xie, and W.-Y. Ma. Understanding mobility based on gps data. In *Proceedings of the 10th ACM International Conference on Ubiquitous Computing (UbiComp '08)*, pages 312–321. ACM, 2008. 11, 15

[161] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma. Mining interesting locations and travel sequences from gps trajectories. In *Proceedings of the 18th ACM International Conference on World Wide Web (WWW '09)*, pages 791–800. ACM, 2009. 11, 15

[162] X. Zhou, Z. Zhao, R. Li, Y. Zhou, J. Palicot, and H. Zhang. Human mobility patterns in cellular networks. *IEEE Communications Letters*, 17(10):1877–1880, 2013. 16

[163] J. Ziv and A. Lempel. Compression of individual sequences via variable-rate coding. *IEEE Transactions on Information Theory*, 24(5):530–536, 1978. 20, 81