Context and Semantic Aware Location Privacy

THÈSE N^{O} 7057 (2016)

PRÉSENTÉE LE 8 JUILLET 2016

À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS
LABORATOIRE DE SYSTÈMES D'INFORMATION RÉPARTIS
PROGRAMME DOCTORAL EN INFORMATIQUE ET COMMUNICATIONS

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Berker AĞIR

acceptée sur proposition du jury:

Prof. B. Faltings, président du jury
Prof. K. Aberer, directeur de thèse
Prof. U. Hengartner, rapporteur
Dr T. G. Papaioannou, rapporteur
Prof. A. B. Ford, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2016

To my family...

Aileme...

Abstract

With ever-increasing computational power, and improved sensing and communication capabilities, smart devices have altered and enhanced the way we process, perceive and interact with information. Personal and contextual data is tracked and stored extensively on these devices and, oftentimes, ubiquitously sent to online service providers. This routine is proving to be quite privacy-invasive, since these service providers mine the data they collect in order to infer more and more personal information about users.

Protecting privacy in the rise of mobile applications is a critical challenge. The continuous tracking of users with location- and time-stamps exposes their private lives at an alarming level. Location traces can be used to infer intimate aspects of users' lives such as interests, political orientation, religious beliefs, and even more. Traditional approaches to protecting privacy fail to meet users' expectations due to simplistic adversary models and the lack of a multi-dimensional awareness. In this thesis, the development of privacy-protection approaches is pushed further by (i) adapting to concrete adversary capabilities and (ii) investigating the threat of strong adversaries that exploit location semantics.

We first study user mobility and spatio-temporal correlations in continuous disclosure scenarios (e.g., sensing applications), where the more frequently a user discloses her location, the more difficult it becomes to protect. To counter this threat, we develop adversary- and mobility-aware privacy protection mechanisms that aim to minimize an adversary's exploitation of user mobility. We demonstrate that a privacy protection mechanism must actively evaluate privacy risks in order to adapt its protection parameters. We further develop an Android library that provides on-device location privacy evaluation and enables any location-based application to support privacy-preserving services. We also implement an adversary-aware protection mechanism in this library with semantic-based privacy settings.

Furthermore, we study the effects of an adversary that exploits location semantics in order to strengthen his attacks on user traces. Such extensive information is available to an adversary via maps of points of interest, but also from users themselves. Typically, users of online social networks want to announce their whereabouts to their circles. They do so mostly, if not always, by sharing the type of their location along with the geographical coordinates. We formalize this setting and by using Bayesian inference show that if location semantics of traces is disclosed, users' privacy levels drop considerably. Moreover, we study the time-of-day information and its relation to location semantics. We reveal that an adversary can breach privacy further by

exploiting time-dependency of semantics. We implement and evaluate a sensitivity-aware protection mechanism in this setting as well.

The battle for privacy requires social awareness and will to win. However, the slow progress on the front of law and regulations pushes the need for technological solutions. This thesis concludes that we have a long way to cover in order to establish privacy-enhancing technologies in our age of information. Our findings opens up new venues for a more expeditious understanding of privacy risks and thus their prevention.

Keywords: location privacy, privacy concerns, location semantics, mobile applications, privacy sensitivities, inference attacks, bayesian inference, bayesian networks

Résumé

Avec toujours plus de puissance de calcul et l'amélioration de leur capacités sensorielles et de communication, les appareils intelligents ont modifié et amélioré la façon dont nous traitons, percevons, et interagissons avec l'information. Les données contextuelles personnelles sont largement stockées et exploitées sur ces appareils et, souvent, envoyées aux fournisseurs de services en ligne. Cette façon de faire s'est révélé être très envahissant pour la sphère privée, puisque ces fournisseurs de services extraient les données qu'ils recueillent afin de déduire de plus en plus de renseignements personnels sur les utilisateurs.

Protéger la vie privée avec la croissance des applications mobiles est un défi crucial. Le suivi continu des utilisateurs avec l'heure et l'emplacement expose leur vie privée à un niveau alarmant. L'historique des positions peut être utilisé pour déduire les aspects intimes de la vie de l'utilisateur, tels que ses intérêts, son orientation politique, ses croyances religieuses, et plus encore. Les approches traditionnelles pour protéger la vie privée ne parviennent pas à répondre aux attentes des utilisateurs en raison de modèles adverses simplistes et d'absence de multidimensionnalité. Dans cette thèse, nous poussons plus loin le développement d'approches de protection de vie privée (i) en l'adaptant aux capacités réelles de l'adversaire et (ii) en enquêtant sur la menace des adversaires qui exploitent la sémantique des lieux.

Tout d'abord, nous étudions la mobilité des utilisateurs et les corrélations spatio-temporelles des déplacements de l'utilisateur dans des scénarios de suivi continu (par ex. applications de mesures continues), dans lequel plus un utilisateur révèle son emplacement, plus il devient difficile de le protéger. Pour contrer cette menace, nous développons des mécanismes de protection de la vie privée, tenant compte des adversaires et de la mobilité, qui visent à minimiser l'exploitation par un adversaire de la mobilité de l'utilisateur. Nous démontrons qu'un mécanisme de protection doit activement évaluer les risques sur la vie privée afin d'adapter ses paramètres de protections. Nous développons également une bibliothèque Android qui fournit une évaluation du risque directement sur l'appareil et qui permet à toute application basée sur la localisation de préserver la vie privée de ses utilisateurs. Nous implémentons dans cette bibliothèque un mécanisme de protection conscient des adversaires potentiels et avec des paramètres de confidentialité fondés sur la sémantique.

De plus, nous étudions les effets d'un adversaire qui exploiterait la sémantique des lieux pour renforcer ses attaques sur les déplacements des utilisateurs. Ces informations détaillées sont disponibles pour un adversaire via des cartes de points d'intérêt, mais aussi via les utilisateurs

eux-mêmes. En règle générale, les utilisateurs de réseaux sociaux en ligne veulent annoncer leurs allées et venues à leurs cercles d'amis. Ils le font la plupart du temps, sinon toujours, en partageant leur activité avec les coordonnées géographiques. Nous formalisons ce cadre et en utilisant l'inférence bayésienne montrons que si la sémantique des lieux visités est divulgué, le niveau de vie privée des utilisateurs s'en trouve considérablement réduit. De plus, nous étudions les informations temporelles et de leur relation à la sémantique du lieu. Nous découvrons qu'un adversaire peut encore mieux infiltrer la vie privée en exploitant les dépendances sémantico-temporelles. Nous implémentons et évaluons, dans le même cadre, un mécanisme de protection tenant compte des différentes sensibilités.

La bataille pour la vie privée exige une conscience sociale et de la volonté pour gagner. Cependant, la lenteur des progrès sur le front de la loi et de la réglementation pousse le besoin de solutions technologiques. Cette thèse conclut que nous avons encore un long chemin à parcourir pour établir des technologies améliorant la confidentialité dans notre ère de l'information. Nos résultats ouvrent la voie à une compréhension plus rapide des risques pour la vie privée, et ainsi à leur prévention.

Mots clefs : protection des données de localisation, préoccupations concernant la vie privée, sémantique de la localisation, applications mobiles, sensibilités de la vie privée, attaques par inférence, inférence bayésienne, réseaux bayésiens

Acknowledgements

I would like to express my gratitude to people who contributed to this thesis and my life as a Ph.D. student.

First and foremost, I would like to thank Prof. Karl Aberer for believing in me, supporting and guiding me through uncertain situations. I understood the virtue of leading better after countless meetings that ended with a clear and motivated mind of mine. Thank you, Karl.

I also would like to thank all my co-authors who contributed to this thesis through not only collaboration and research, but also their enlightening discussions on everything and friendship: Dr. Kévin Huguenin, Dr. Jean-Paul Calbimonte, Prof. Jean-Pierre Hubaux, Prof. Urs Hengartner, Dr. Thanasis Papaioannou, Dr. Rammohan Narendula and Dr. Iris Safaka.

I appreciate the time and dedication my thesis committee devoted for reviewing this thesis: Prof. Boi Faltins, Prof. Urs Hengartner, Prof. Bryan A. Ford and Dr. Thanasis Papaioannou. Thank you for the fruitful discussions and your comments.

I am thankful to all my colleagues from LSIR and LCA1 who contributed to this thesis in many ways: Hamza, Nevena, Igor, Mathias, Julien, Erman, Martin, Rameez, Murtuza, Hossein, Jean-Eudes, Alevtina, Alexandra, Mehdi, Tri, Julia, Zhixian, Hung, Amit, Hao and Tian for the discussions, challenges and the great environment they generated for making Ph.D. life manageable. My special thanks go to Kévin and Julien for their wonderful friendships as office mates. I enjoyed every bit of moment we shared to talk about Ph.D., technology, movies, jokes and everything else in life.

Switzerland is a very beautiful country, but as every other beautiful thing, it is worth more when you share it. My life in Lausanne was only complete with many great friends that kept me going in this journey. My Lausanne family, i.e., Cem, Başak, İsmail, Ece, Kerem, Gizem, Nariye, Meriç, Can, Suat, Ahmet, Onur, Handan, Dilan, Merve, Buğra, Egeyar, Burak, Berat, Duygu, Barış, Onur, Cansu, Victor, Lyusja, Nastya, Vlad, Cengiz, Işıl, Şakir and many other great people were a source of joy, contributed to my life and helped me grow further in many ways. I am indebted to you all.

The last, chaotic, year of this journey would be devastating without the support and love of Gökçen, who helped me through difficult times. Your presence always calmed me and helped me clear my mind. Thank you for all the encouragement, happiness and sharing my most

important moments in life so far and also walking with me many years to come.

Finally, I want to thank my family and particularly my mother, Betül. Your unconditional love, support and guidance have been the main driving force for me to become who I am now, with the latest addition of the title 'doctor'. Betül, Yeşim, Rengin, Gönenç, Süleyman, Ali Osman, Tibet, Yadigar, thank you all for everything.

Contents

Introduction

Mobile technologies are evolving at a dazzling rate leading to adoption of always-on, always-connected smart devices by individuals. With ever-increasing computational power, and improved sensing and communication capabilities, these smart devices have altered and enhanced the way we process, perceive and interact with information. Masses are not only information consumers anymore, they have become data producers as well by integrating to the Web with their smartphones, activity trackers and more. Users of these mobile devices (*i.e.,* mobile users) not only publish their opinions, photos or videos online, but also, potentially unknowingly, upload a great deal of their personal activities as a direct result of contextual sensing abilities of these devices and ubiquitous connections to service providers. Applications built on top of this type of technology can use contextual information to produce added value, as they can provide recommendations, predictions and other processed information back to the users.

However, it is obvious that with these developments in the last two decades, the threat to privacy of mobile users has been an ever-increasing side effect. This subject has several *battlegrounds* such as law, policy-making, regulations and technology. Thereby, privacy issues have received substantial attention from the research community, though insufficient when compared to the advances in technology that make privacy vulnerable. With the invention and widespread adoption of mobile devices, the technology has caught individuals off-guard and online users took the ride of rapid consumption of smart devices and gadgets, unaware of how vulnerable their privacy has become. Unfortunately, laws, policies and regulations tend to take a long time to be defined and implemented, and usually after the technology is put in place. Currently, they have not caught up with the technology. The exposure of PRISM program created by the National Security Agency (NSA) of the United States of America (USA) to collect large scale data for mass surveillance [55], dubbed the *PRISM scandal* by many and rightly so, proved that the threat to individuals' privacy is real. Another example is the recent request by Federal Bureau of Investigation (FBI) in the USA from Apple to help hack iPhones. The threat is clarified by the Apple CEO Tim Cook himself in a letter to their customers [33]:

> [The FBI has] asked us to build a backdoor to the iPhone. Specifically, the FBI wants us to make a new version of the iPhone operating system, circumventing several important security features, and install it on an iPhone recovered during

> the investigation. In the wrong hands, this software — which does not exist today — would have the potential to unlock any iPhone in someone's physical possession.

Recently, the European Union started to take a firmer stand on protection of individuals' data especially after the exposure of the PRISM program. A European court issued the "Right to be Forgotten", which dictates search engines to remove any link to private data on European citizens if they request so [45] (unless the data is published in a journalistic work or the request clashes with freedom of expression). The applicability of this decision and how search engines can implement such a mechanism are still in question. In summary, there is much to do on the side of laws and regulations for privacy whereas the threat to privacy is imminent. Therefore, it is important to develop solutions on the technology side to protect privacy while laws, policies and regulations catch up with the pervasive privacy challenges.

In information systems, privacy can be defined as the control of users over their personal information in terms of managing, editing and deleting. As the aforementioned examples present, ubiquitous connectivity and data flow to service providers hinder this control. Therefore, technological solutions for addressing privacy challenges are in fact tools for providing users the ability to control their personal information. Notwithstanding the research efforts on designing such technological solutions, *i.e., privacy-protection mechanisms* (PPM), an adversary having access to a user's data may try to extract further personal information based on his existing knowledge regarding that user, the context and the relationship between various dimensions of the targeted information. Therefore, a potential adversary's capabilities and knowledge must be concretely defined and addressed when designing PPMs.

In this thesis, we focus on location tracking of mobile users by mobile applications. Increasingly more people use GPS-enabled mobile devices to enjoy location-based services and location-based social networks. Users of such applications provide location information to the service providers in return for useful information, such as the location of the nearest restaurant, cinema or nearby friends, or simply to keep their friends posted about their activities. Many of these mobile applications are presented as free, but in fact, they obtain fine-grained user traces that can be used to infer more personal information: the price a user pays for benefiting from such services is her location data, which is detrimental to her privacy. The online technology community and news agencies draw attention to the problem of location privacy every now and then pointing to the fact that the risks are actual [95, 106]. This problem was broadly investigated by the research community, focusing mostly on geographical location privacy and related protection mechanisms [67, 68]. It was also shown how an adversary can locate/track users' whereabouts based on location samples that are potentially anonymized and/or obfuscated, and on mobility history (e.g., [68, 100]).

There are three main assumptions in the approach to tackling the privacy issues presented in this thesis:

- An adversary may have potentially incomplete background information, when performing an attack on user traces, including users' mobility history and location semantics,
- The existing online systems (e.g., aggregation servers, location-based services and location-sharing based services) are not willing to make extensive changes to their system architectures to support privacy protection at the expense of utility and/or time, but may accept obfuscated locations, which means that the required changes to their systems would be minimal,
- It is difficult to establish and maintain a trusted third-party that handles the protection of privacy on behalf of users, and therefore the solutions for protecting location privacy should be implemented on user devices.

These assumptions are in line with the current systems in place given the minimal potential changes, and therefore realistic. They enable the implementation of related solutions to be deployed sooner than those that require extensive system changes or from-scratch system designs. On the other hand, it has to be noted that many online applications must inevitably be redesigned with a privacy-friendly approach.

Contributions

In this thesis, we focus on the context of location information in mobile applications and the related privacy issues. More specifically, we consider the correlations among user events in spatio-temporal dimensions and also the effect of location semantics (*i.e.,* the type of locations) on location privacy. In this sense, we formally define and model adversaries, develop obfuscation-based protection mechanisms against them, and evaluate how successful the adversaries are in various scenarios. Overall, the approach of the thesis to research on location-privacy can be summarized by the relationship between an adversary and a user trying to maximize their gains against each other by adapting to different dimensions and properties of location in a continuous game (as illustrated in Figure 1).

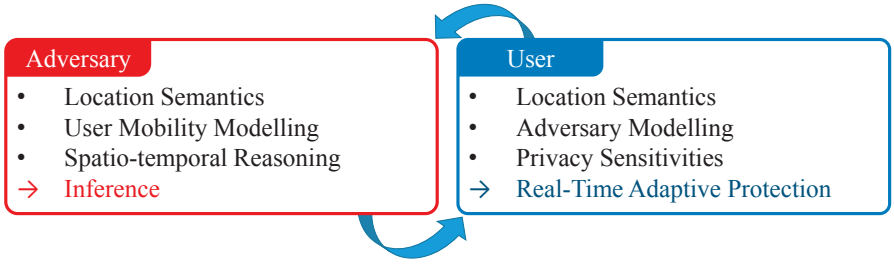


Figure 1 – We embrace a two-sided medallion in privacy research: The adversary and users move in opposite directions. The adversary obtains both public and side information about users in order to breach privacy. Conversely, users should anticipate the adversary's capabilities in order to protect their location privacy adaptively.

In **Part I**, we tackle the location-privacy issues in continuous information disclosure scenarios

such as sensing systems. In these scenarios, the privately-held mobile devices generate geo-tagged events very frequently (e.g., every 1, 5 or 10 minutes) and submit them to an aggregation server (which may or may not be a service provider). The main problem in these scenarios is the correlation between disclosed events from a spatio-temporal perspective, namely the speed and direction of users, and the users' mobility history that might be known to potential adversaries. We argue that these aspects, that is **the context of location** and the adversary's capabilities, need to be taken into account when protecting location privacy, especially by the user device. Specifically, we:

- propose a location-privacy protection mechanism that adaptively determines the size of a rectangular area that replaces the actual location of the user by taking into account consecutive events she publishes. This is achieved by (i) estimating through Bayesian inference the expected privacy level, and (ii) analyzing the reachability among consecutive events with the maximum possible speed of the user. This novel approach is shown to be superior to non-adaptive protection mechanisms. (**Chapter 2**)
- develop a heuristic algorithm that complements the adaptive protection mechanism presented in Chapter 2 to protect location-privacy better: we model the mobility of users and consider the most probable transitions when determining the obfuscation area. The proposed mechanism is shown to be more effective against a mobility-aware adversary than a random obfuscation approach. (**Chapter 3**)
- develop a location-privacy library on the Android platform that implements the adaptive protection mechanism we propose by extending it with user privacy sensitivities w.r.t. location semantics. The library demonstrates the applicability of adaptive protection mechanisms and location privacy evaluation on user devices. It is shown to be lightweight in terms of performance. (**Chapter 4**)

In **Part II**, we focus on the sporadic event generation by users, which typically occurs in online social networks such as Facebook, Twitter and Foursquare. In such applications, events are generally generated by users on-demand (*i.e.,* not automatically). The main problem we tackle in this type of scenario is that users may disclose the type of their location (*i.e.,* the location semantics), which increases the threat to their location-privacy. In other words, we approach location-privacy issues from a **semantic-aware** point of view. More specifically:

- We show that a semantic-aware adversary is more capable of inferring the true location of users against semantic-oblivious privacy-protection mechanisms than a traditional adversary assumed in most of the related work. (**Chapter 5**)
- We argue, based on the analysis of a real dataset, that location semantics are time-dependent, *i.e.,* people go to specific types of places at specific times of day, and we analyze the potential threats by formalizing a time-aware adversary. (**Chapter 6**)
- We propose a semantic and history aware protection mechanism that specifically takes into account user sensitivities w.r.t. certain types of location in order to protect location-

privacy against semantic-aware adversaries. We demonstrate experimentally that such a protection approach is more successful at protecting the location-privacy of users than static protection mechanisms that do not consider any background information. (**Chapter 6**)

Overall, this thesis models adversary capabilities (considering multiple dimensions of location information) and investigates inference attacks in order to better understand the threats, and presents novel *intelligent* privacy-protection mechanisms for mobile users that run on user devices without relying on third parties.

1 State of the Art

Location privacy attracted a lot of interest from the research community in the last two decades in parallel to the developments in mobile systems. How to protect location from service providers and thus potential adversaries, and how can an adversary infer location traces when protection mechanisms are applied on them have been the main research questions. In this sense, there have been several main directions of research on location privacy. These can be summarized as follows ordered from the most tackled issues to the least:

1. Protection-mechanism design for location-privacy.
2. Evaluation of protection mechanisms w.r.t. real attacks and datasets.
3. User privacy preferences and sensitivities.
4. Semantic location-privacy.
5. Trade-off between privacy and application utility.

Hereafter, we discuss the state of the art in terms of protection approaches that constitute the bulk of this chapter, evaluation of privacy, privacy preferences and utility. The non-exhaustive list of work, but representative of the most-commonly used approaches in the field related to this thesis, is presented in Table 1.1, that positions them w.r.t. three main aspects of location-privacy research (and their titles in Table 1.2 for convenience): protection techniques used, the adversary model they assume, and also how they evaluate their work. More specifically, the first three columns of 'Protection' column represent the main methods adopted in most protection mechanisms in the existing work. These are anonymization, *i.e.,* removing user identities and making it difficult to identify users, obfuscation and perturbation of location for confusing the adversary, and fake data injection to mislead the adversary by distorting his knowledge on user traces. We also analyze the related work w.r.t. various approaches they adopt in privacy protection: Do they take into account the simple mobility constraints such as velocity? Do they take into account the location types, *i.e.,* semantics? Do they provide personalization for the level of privacy users require? Do they consider the adversary capabilities actively and adapt the privacy protection parameters accordingly?

Table 1.1 – State-of-the-art on location privacy w.r.t. which protection approaches they use, whether they implement inference mechanisms and their evaluation approach. Last two rows shows the aspects of location privacy this thesis addresses/considers in respective parts.

	Protection							Adversary				Evaluation					
												Geographical			Semantic		
	Anonymization	Obfuscation & Perturbation	Fake data	Velocity-aware	Semantic-aware	Personalized	Adaptive	History-aware	Landscape-aware	Semantic-aware	Time-aware	Anonymity/Cloak size	Adversary Confusion	Adversary Correctness	Anonymity/Cloak size	Adversary Confusion	Adversary Correctness
[16]	✓			✓					✓			✓					
[41]		✓				✓											
[51]	✓	✓						✓				✓	✓				
[49]	✓	✓				✓						✓					
[89]	✓	✓							✓			✓					
[57]	✓	✓				✓	✓		✓				✓	✓			
[79]	✓	✓						✓	✓			✓	✓				
[98–100]	✓	✓	✓					✓	✓		✓		✓	✓			
[118]		✓			✓	✓		✓	✓	✓				✓			
[112]	✓	✓			✓	✓		✓	✓	✓		✓					
[116]		✓	✓		✓	✓		✓	✓	✓		✓			✓		
[13]		✓							✓								
[50]		✓		✓		✓				✓		✓					
[114]	✓	✓			✓	✓						✓					
[48]		✓		✓		✓		✓									
[34]		✓					✓			✓		✓			✓		
[81]	✓	✓			✓	✓		✓		✓		✓			✓	✓	
[37]	✓	✓			✓	✓	✓	✓	✓			✓	✓	✓			
[21]		✓						✓			✓	✓		✓			
[91]			✓							✓		✓					
Part I		✓		✓		✓	✓	✓			✓		✓	✓			
Part II		✓			✓	✓	✓	✓		✓	✓		✓	✓		✓	✓

Under the 'Adversary' title, we categorize the related work according to the following characteristic of an adversary:

- Taking into account the user mobility history,
- Considering the landscape of the map, *i.e.,* the road network, city topology, *etc.*,
- Exploiting semantics in order to increase inference success,
- Exploiting the user regularity and hence dependence on time dimension.

Table 1.2 – State-of-the-art from Table 1.1 with their titles (in the same order).

Citation	Title
Beresford *et al.* [16]	Location Privacy in Pervasive Computing
Duckham *et al.* [41]	A Formal Model of Obfuscation and Negotiation for Location Privacy
Ghinita *et al.* [51]	MobiHide: A Mobile Peer-to-Peer System for Anonymous Location-Based Queries
Gedik *et al.* [49]	Protecting Location Privacy with Personalized k-Anonymity: Architecture and Algorithms
Pingley *et al.* [89]	CAP: A Context-Aware Privacy Protection System for Location-Based Services
Gruteset *et al.* [57]	Anonymous Usage of Location-based Services Through Spatial and Temporal Cloaking
Meyerowitz *et al.* [79]	Hiding Stars with Fireworks: Location Privacy Through Camouflage
Shokri *et al.* [98–100]	A Distortion-based Metric for Location Privacy Quantifying Location Privacy Quantifying Location Privacy: The Case of Sporadic Location Exposure
Yiu *et al.* [118]	Spacetwist: Managing the Trade-Offs Among Location Privacy, Query Performance, and Query Accuracy in Mobile Services
Xiao *et al.* [112]	p-Sensitivity: A Semantic Privacy-Protection Model for Location-based Services
Xue *et al.* [116]	Location Diversity: Enhanced Privacy Protection in Location Based Services
Ardagna *et al.* [13]	Landscape-aware Location-Privacy Protection in Location-based Services
Ghinita *et al.* [50]	Preventing Velocity-based Linkage Attacks in Location-aware Applications
Xu *et al.* [114]	Feeling-based Location Privacy Protection for Location-based Services
Freni *et al.* [48]	Preserving Location and Absence Privacy in Geo-Social Networks
Damiani *et al.* [34]	The PROBE Framework for the Personalized Cloaking of Private Locations
Monreale *et al.* [81]	C-safety: A Framework for the Anonymization of Semantic Trajectories
Dewri *et al.* [37]	Local Differential Perturbations: Location Privacy under Approximate Knowledge Attackers
Bordenabe *et al.* [21]	Optimal Geo-indistinguishable Mechanisms for Location Privacy
Primault *et al.* [91]	Differentially Private Location Privacy in Practice

Lastly, we provide an overview of how the related work does 'Evaluation' in geographical and semantic dimensions of location privacy. The majority of the work is under anonymity/cloak size columns, which represent the simplistic evaluation approaches that do not reflect well the privacy levels. In summary, these include measuring the level of privacy in terms of size of the anonymity set or size of the obfuscation area generated. The other two types of columns, namely adversary/confusion and adversary/correctness correspond to more rigorous analysis of an adversary's success at inference. Adversary/confusion state whether a related work analytically or experimentally calculates the probability that an adversary infers the actual location (or semantics) of a user. Adversary/correctness shows whether a related work implements an actual attack and experimentally computes, or analytically analyzes the adversary's error in inferring a user's geographical or semantic location.

1.1 Privacy-Protection Mechanisms

The purpose of this section is to give an overview of the most commonly used techniques to protect location-privacy of users in mobile systems. The state-of-the-art we present below generally considers a scenario where users query a location-based service and therefore have the liberty to modify their identities and locations in return of a degraded utility. These works do not support sharing location with friends (with exceptions) or announcing the semantics of the user location.

Many previous works focused on anonymizing users in order to prevent an adversary from linking disclosed location data to the origin user. One idea is replacing the identities of users with pseduonyms when using location-based services so that the service provider would not receive the real identity of the user. However, as Bettini *et al.* [17] point out, multiple location points, consecutively disclosed by a user, can quickly become quasi-identifiers meaning that they can be used to identify the user and hence link the location traces back to her. Beresford *et al.* [16] proposed a solution to this named *Mix-zones*, in which unobserved areas would enable the users to exchange their pseudonyms thus making it difficult to trace individuals. The frequency of changing the pseudonyms affects the level of privacy.

Another well-known approach to protect location-privacy is called k-anonymity [102], a technique borrowed from data privacy field. In location-privacy context, k-anonymity means that a protection mechanism is required to build a geographical cloaking area such that there exists at least $k-1$ other users in it, so that the users are indistinguishable from each other, which is first proposed by Gruteser and Grunwald [57]. The idea is that the adversary should not be able to link the location and query to the actual user. Their system require a trusted third-party to act as the *anonymizer* that gather queries from users and forwards them to the LBS servers after anonymizing them. Note that in order to meet the k-anonymity requirement, they apply geographical cloaking that contains the locations of all k users so that when the users make a new query as part of a new anonymity group, their traces will not be easily constructed.

Xiao *et al.* [112] and Xue *et al.* [116] argue that k-anonymity and geographical cloaking may not be effective against a reasoning adversary that is aware of the semantics of location. They propose protection mechanisms that are semantic-aware and on the same line as l-diversity [77] that tries to minimize the probability an attacker can guess the semantics of user location. The purpose is to generate geographical cloaks that contain diverse semantic tags in order to protect the semantic location-privacy. The downside of these approaches is that they lack extensive evaluation of privacy by challenging their mechanisms with the assumed attack scenarios. Typically, k-anonymity techniques (e.g., [49, 57, 71, 79, 112, 114]) rely on a trusted third-party to cluster the users' queries in order to create anonymity sets. Ghinita *et al.* [51] proposed MobiHide to eliminate the need of a trusted third-party by coordinating users to anonymize their queries in a distributed fashion.

Overall, there have been quite extensive research efforts on anonymization in the context

of location-privacy in mobile systems. This raised the question of how well anonymity is preserved even if some k-anonymity technique is applied. In fact, it has been shown to be quite ineffective [52, 58, 119]: it is relatively easy to link an individual to her anonymized trace, especially by using her mobility history, *etc.*

As can be observed from Table 1.1, obfuscation is the dominantly used technique in the state of the art, as is the case in this thesis. Apart from anonymity, various obfuscation-based protection mechanisms have been proposed with different algorithms and constraints. The idea of obfuscation for location privacy was initially introduced by Duckham and Kulik [41] where the notions of inaccuracy, *i.e.,* giving a location measurement different than the real one, and imprecision, *i.e.,* reporting a large area instead of a precise location, were formalized. Ghinita *et al.* [50], Freni *et al.* [48] and Agir *et al.* [11] (*i.e.,* Chapter 2) take into account the velocity of users when generating geographical cloaks, which is especially important in continuous disclosure scenarios as tackled in Part I of this thesis. Another constraint that affects the privacy level obtained by using an obfuscation mechanism is the knowledge the adversary has. Although there are many obfuscation-based mechanisms proposed, only a handful of them [11, 21, 34, 37, 57] consider the adversary's knowledge on users.

In the general context of location sharing, a number of cryptographic protocols have been proposed (e.g., [27] for private and cheat-proof "mayorship"-badges, one of the main feature of location-based social networks, and [39, 59] for sharing location with friends without the service provider learning the users' locations). Such solutions, however, involve cryptographic operations and require technical modifications of the service. Related cryptographic protocols, which provide privacy-preserving features, are proposed in [82] and [121]. They rely on secure multi-party computations (garbled circuits) and homomorphic encryption schemes, respectively. Such approaches can be applied to, for instance, friend-finding applications without revealing user locations, but they require careful analysis and extension to incorporate the semantic dimension of location in semantic-aware applications. These mechanisms aim to provide privacy-preserving features in specific applications, and it is not straightforward to modify them in order to cover the cases where people want to disclose their current activities, *i.e.,* their location semantics.

We refer the reader to Krumm [67], Toch *et al.* [105] and Wernke *et al.* [108] for more detailed overviews of the related work on privacy-protection mechanisms.

Semantic Location-Privacy

Location is a context-rich piece of information; simple numbers such as coordinates can reveal the type of location a user is at. Regular disclosure of visits by a user can reveal certain patterns and hence lead to the prediction of personal information and even future visits [15]. As a result, there have been approaches to protect the semantics of user locations. The motivation is that a semantic-aware adversary can still infer private information about users, even if their locations are obfuscated, as long as they contain minimal amount of location semantics.

For example, if a user visits a hospital and she makes a query at a location-based service by obfuscating her location, the server can see that she is at a hospital if her geographical cloak only contains the hospital. The fact that she is at a hospital can be sensitive information leading to suspicions about serious illnesses.

Consequently, some of the research efforts (including this thesis) focus on the semantic dimension of location in order to improve location-privacy protection. Xue *et al.* [116], Yiu *et al.* [118], Xiao *et al.* [112] and Damiani *et al.* [34] use location semantics to determine the size of geographical cloaks for user locations. The PROBE framework by Damiani *et al.* [34] propose the most comprehensive mechanism in terms of semantics by taking into account the ratio of each sensitive type of place occupies in a geographical cloak. Xu and Cai [114] also consider location semantics in an implicit way: they argue that public places are where users generally feel comfortable and hence have less privacy-protection requirements. They build their mechanism with this assumption to obfuscate locations accordingly, however, they do not use any semantically annotated map data.

Overall, there is a gap in the state of the art regarding research on semantic location privacy. Even though semantic-aware adversaries are assumed, no concrete attacks and evaluation on semantic location privacy have been done. Moreover, the effect of exploiting location semantics on geographical location privacy has not been extensively investigated.

1.2 Evaluating Privacy

The research efforts on designing location-privacy protection mechanisms for mobile systems have been increasing, yet the evaluation of the proposed mechanisms does not always give a concrete idea about how much information an adversary could obtain. Most of the state-of-the-art assume a reasoning adversary as summarized in the 'Adversary' column of Table 1.1, but do not actually implement an attack and test the effectiveness of their proposed mechanisms. In fact, inference attacks play an important role in privacy evaluation. Attacking a protected user trace considering realistic background information can give good insight about how well the privacy can be preserved. This also brings the question of how to quantify location-privacy. In the literature, several metrics have been proposed for measuring the level of location privacy. For instance, Duckham and Kulik [41] first propose the idea of obfuscating location for protection, but they do not provide any experimental evaluation. Many other related works (e.g., [49]) also use as the evaluation metric the size of the anonymity sets or geographical cloaks obtained by their mechanisms. However, these metrics do not reflect how much an adversary can truly infer. As an improvement, Ghinita *et al.* [51] and Meyerowitz *et al.* [79] not only evaluate their mechanisms by cloak size, but also the a-posterior belief of the adversary considering certain attacks.

A more useful metric is *entropy*, which is an information theoretical approach to privacy measurement. In [38, 94], entropy H is proposed as an anonymity metric to measure the privacy offered by a system. It is defined as $H = -\sum_i p_i \log_2 p_i$, where p_i is the attacker's

estimate of the probability that a participant i is responsible for some observed action. In the context of this thesis, the observed actions consist of reported locations by mobile users at a specific time, thus entropy can be used to measure how well the actual location is hidden in a cloaking area, *i.e.,* the uncertainty of the adversary about the actual location of a user. Although entropy adequately assesses the uncertainty of the adversary, it does not measure the correctness of the adversary's estimation. Krumm [68] demonstrated the success of an inference attack that identifies the home locations of users, which can be used in conjunction with such metrics for privacy evaluation to an extent.

The aforementioned issues are formally addressed by Shokri *et al.* [98] and an evaluation approach w.r.t. the *error* an adversary makes and the *confusion* he has when attacking the location traces of users is proposed. To this end, the error and the confusion of the adversary are measured by assigning probabilities to all possible user events and by calculating the distances between the actual user locations from all the observed locations for each time instant. The distances are then multiplied with their respective probabilities in order to obtain the *expected distortion* LP_u (*i.e.,* location privacy measured as the expected error of the adversary) for a corresponding user. LP_u is given by the following formula:

$$LP_u(t) = \sum_{\Psi} \text{dist}(a_u(t), \Psi(t)) \cdot \mathbf{Pr}(\Psi, t) \tag{1.1}$$

where $LP_u(t)$ is the location privacy of user u at time t and Ψ represents all the observed trajectories of user u. $a_u(t)$ gives the actual location of user u at time t and $\Psi(t)$ is the location on trajectory Ψ at time t. $\mathbf{Pr}(\Psi, t)$ is the probability assigned to trajectory Ψ at time t by the adversary. $dist(\cdot,\cdot)$ is a distance function for two given locations and can be the absolute distance function, in which case the location privacy would be in km or meters.

Shokri *et al.* [99, 100] extended their idea behind the aforementioned distortion-based metric and presented a comprehensive location-privacy quantification framework. This framework formalizes the attack of the adversary, takes into account its background information on users' mobility patterns and calculates users' location-privacy protection levels based on the adversary's *accuracy, correctness* and *certainty* about users' actual trajectories. The authors also propose a software tool, called *Location Privacy Meter* (LPM), which implements this framework. The LPM consists of several attack strategies based on Hidden Markov models. Chen *et al.* [29] further extended this framework and focused on the activity of users at points of interest (POIs) in relation to their mobility. They consider time spent at POIs in order to enhance the attack and evaluate expected error of the attack using a real dataset. However, their evaluation is weak in terms of trajectory length and due to the lack of specificity of obfuscation employed. Furthermore, they do not employ semantic annotations for the POIs. Nevertheless, this kind of improvements on existing work paves the way for better evaluation approaches.

Chatzikokolakis *et al.* [28] utilize the well-known differential privacy concept introduced for databases by Dwork [42] in order to provably protect location-privacy with certain privacy

guarantees. Originally, the concept of differential privacy [42] is a privacy measurement approach for statistical databases. The privacy is measured by the predictability of the existence of a single record based on a statistical result obtained from a database. Chatzikokolakis *et al.*'s [28] use of differential privacy for location inherently evaluates a potential adversary's expected error in inferring the actual user location. Thus, their differential privacy based metric fits the theme and approach of this thesis, enabling as a future work the enhancement of the proposed schemes and evaluation results presented in the following chapters.

Another important issue when evaluating privacy is the conformity of the evaluation results to real life, *i.e.,* what kind of dataset is used to evaluate location-privacy. Obviously, real user traces may not be available to researchers. The San Francisco Cab [90] and Nokia Lausanne Data Collection Campaign [85] datasets contain rich and continuous location traces. Such datasets are useful for the experimental evaluation of location-privacy from a geographical point of view. The datasets are poorer on the semantic side in the sense that there is not a publicly available dataset of semantically annotated location traces that can be used in research. Note that, Yan *et al.* [117] and Krumm [69] propose methods to automatically annotate traces with semantic and even means of transport tags; however, experiments that are based on traces annotated by such methods do not guarantee realistic results. In this thesis, to overcome this problem, we collected geo-tagged tweets that are public through Twitter's public stream [9]. By processing the tweets, we obtained considerable amounts of Foursquare check-ins, that by design include semantic tags for the venues visited. This kind of approach gives us the ability to conduct experimental evaluation on real traces. The downside is that such a dataset contains only sporadic user events and is not useful for evaluation in continuous disclosure scenarios. In the future, building a sound and non-intrusive dataset of semantically annotated traces can accelerate the research on location-privacy.

1.3 Privacy Preferences and Sensitivities

Individuals may have different privacy needs in different contexts. In mobile applications, users' need for privacy may arise from the possibility of inference of personal information from shared data such as location, semantics, *etc.* Also, each individual's privacy requirement can be different from others in the same context. For example, a doctor may feel comfortable sharing his/her location at the hospital he/she works at, but a patient may not have the same comfort. This dimension of privacy needs investigation in order to provide users with sufficient privacy levels by setting healthy protection parameters. This means that users need to understand their privacy needs, perceive the protection mechanisms clearly, and act upon them by determining their sensitivities accurately. Unfortunately, there is no magic solution or one-size-fits-all scheme for privacy preferences or sensitivities of users at the moment.

A 2011 survey by Toch *et al.* [105] approaches the privacy issues in personalization-based systems, which rely on data collection from users. They refer to social studies with users of online social networks and state that users become increasingly privacy-aware and sensitive

due to mass collection and processing of their data by the very systems they use. They conclude that client-side privacy profiles can be useful when the personalization algorithms can run on the client-side as well. Such client-side privacy profiles, in fact, are crucial to give users the control over their privacy and related protection mechanisms.

Toch [103] proposes a crowdsourcing framework for privacy sensitivities in order to build a collective sensitivity profile, which then can be used to manage individual privacy needs of users in context-aware applications. Their approach is to help build sensitivity- and privacy-aware location-based services by automating privacy protection decisions based on inferred sensitivities of users. They also provide a semi-automatic approach to this method and show that sporadic input by users in determining the sensitivity for certain locations or semantics provide a better protection of privacy through increased accuracy of sensitivity inference. Overall, the aim is to minimize the effort by users to set their sensitivities for a privacy-aware mobile application. Such approaches are useful to enhance sensitivity-aware protection mechanisms implemented on user devices.

1.4 Trade-off between Privacy and Utility

Utility depends on the type of application and in mobile systems there are two main categories of applications in terms of utility. In the first one, a data collector receives data from users and then this data is used to generate a value for the data collector. Typical examples include sensing and crowd-sourcing applications, which require an abundant amount of data to do, for instance, research. The other category is the case when the users of the system receive a service from a service provider. Assuming the utility is not dependent on the users' identities, the most utility-friendly protection techniques are pseudonym-based anonymization mechanisms that do not employ obfuscation. Obviously, applying obfuscation on location data degrades the utility of a location-based mobile application.

Krause and Horvitz [66] investigate the trade-off between privacy and utility in the context of web search analytically in order to find a near-optimal balance between them. They quantify the relationship between them using entropy and show that it is possible to provide the users with tools to help them make privacy-preserving decisions when web searching while keeping their utility fairly optimal. Singla and Krause [101] further investigate the trade-off between privacy and utility in the context of participatory sensing applications. They experimentally evaluate the loss in utility as privacy protection requirements increase. They formalize the utility needs of the system, design incentives for contributing users and propose a privacy-protection mechanism to provide users a balanced privacy level: as users provide more information, they get 'paid' by the system as this means higher utility.

Bilogrevic *et al.* [19] study the effect of privacy protection on utility from users' perspective. They conduct a user study for understanding the impact of various protection approaches on the perceived utility of users. Their findings show that users' utility in online social networks is quite dependent on semantics and protecting location in the geographical dimension is less

detrimental to utility compared to protection in the semantic dimension. This kind of work is crucial for better analyzing the utility in privacy research and complements our efforts in this thesis.

1.5 Emerging Privacy Challenges and Directions

With new application directions in mobile systems, new privacy threats, hence new research directions emerge. In the context of location-privacy, one of the new challenges is about co-location privacy, *i.e.,* how disclosing one's location can affect the privacy of those around her. This problem is first studied by Olteanu *et al.* [86] and it has been shown that an adversary can infer individuals' actual locations by using co-location information obtained from social networks such as Foursquare. This kind of problem was first formally defined as *interdependent privacy* and studied by Biczók *et al.* [18], according to whom "the privacy of individual users is bound to be affected by the decisions of others".

Another research direction is privacy-aware decision making in mobile applications. This line of work aims to make sharing of information decisions based on user preferences. One approach to achieve this, as implemented by Bilogrevic *et al.* [20], is to learn from user decisions the privacy preferences and after a learning period, make sharing decisions automatically. This kind of machine-learning based approaches are promising as the smartphones are getting quite powerful and capable of running complex algorithms continuously in the background.

As we will discuss and prove in the following chapters of this thesis, location privacy is subject to multi-dimensional threats. With mobile users sharing their opinions and activities on online social networks, their location becomes more susceptible to inference attacks. For instance, Liu *et al.* [75] show how tweet contents in Twitter can be used to guess types of locations a user visits, whereas Cheng *et al.* [30] demonstrates that tweets can be geo-tagged by exploiting their content. This line of prediction, in other words inference, schemes introduce new challenges regarding location-privacy protection.

Lastly, Barak *et al.* [14] propose a scheme that anonymizes user traces by replacing the geographical location coordinates with semantic tags and clustering users w.r.t. this type of *semantic cloaks*. According to their study on how unique user visits are, semantic cloaking of location traces improves anonymity. In overall, this work suggests to create datasets of semantically cloaked traces rather than geographical traces for data requesters where coordinates are not necessary.

Part I
Context-aware Location-Privacy – Continuous Disclosure Scenarios

2 Adaptive Location-Privacy Protection

The recent advances in sensor technology have led to a wide availability of privately-held low-cost sensors in mobile phones, vehicles, home appliances, etc. In turn, this has led to the development of the participatory sensing paradigm, that enables vast sensor data–from privately-held sensory devices–to be collected. In this paradigm, mobile devices send, either continuously or on demand, sensor data along with their locations and timestamps to an aggregation entity. The participatory sensing paradigm paves the way for innovative applications of great social and business interest, such as air pollution monitoring [84], early earthquake detection [76], and electrosmog exposure. This concept has attracted much attention from the research community, e.g., [35, 40], as it is an alternative to the costly and difficult-to-manage deployment of dedicated sensor-network infrastructures. However, participatory sensing faces serious challenges: data accuracy (*i.e.,* due to the low-cost sensors), user privacy protection, finding incentives for users to contribute to the system, etc. Most importantly, users who are sensitive about their private information, such as their location (and inferred activities), are not expected to be willing to contribute to the system. Therefore, it is necessary in such systems to implement privacy-protection mechanisms such as anonymization of the source, and obfuscation of the location and/or time information attached to data. The usefulness of the sensed data for the corresponding application, however, depends on the accuracy, the availability and the spatio-temporal correctness of the data, all of which are negatively affected by privacy-protection mechanisms. For example, data accuracy decreases when location or time information are obfuscated, hence a trade-off emerges between data utility and user privacy. If a certain scheme that rewards mobile users according to the utility of their sensed data is in place, then, assuming that users are utility maximizers, they can more willingly provide data and enjoy a satisfactory level of privacy-protection.

An adversary who has access to users' spatio-temporal traces can find users' activity schedules [117]. To this end, the main objective of a protection mechanism is to provide untraceability. There exist location-privacy protection mechanisms [67] employing techniques such as data hiding or location obfuscation, with limited effectiveness against powerful adversaries that can exploit spatio-temporal associability of users' observable events, in order to partially or

fully discover user trajectories. This happens because these techniques are employed with static parameters that cannot satisfy the user privacy requirements when the correlation between sequential user actions (*i.e.,* user mobility and data emissions) and user's context lead to excessive leakage of personal information. Therefore, a user can never be sure that a static privacy protection will be successful against adversaries at all times. Another core problem of most of the existing approaches is the assumption that all users require the same level of privacy. This results in an unnecessarily high level of protection for some users and in insufficient protection for others. This second problem was addressed by Xiao *et al.* [111] in the context of anonymization of datasets, which is not directly comparable to our context (where we do not consider anonymization).

In this chapter, we propose an innovative approach for adaptive location-privacy protection in the participatory sensing context. Our objective is to provide the user with a statistical privacy guarantee at the lowest possible utility loss for the application. In order to achieve this objective, we define a personal privacy threshold θ, which is a lower bound on user location privacy. Before taking any privacy-protection action, in order to meet θ at the minimal utility cost, the privacy level of the user is dynamically measured on the user's device and compared with θ. Our adaptive scheme for location-privacy protection is lightweight, realistic and thus easily deployable at mobile devices. We consider two threat models: (a) A semi-honest aggregation server that attempts to extract and exploit private location information based on the emitted sensor data. (b) An active-tracking aggregation server, which employs both the (partial) location history of the user and the emitted data for extracting and exploiting private location information. Using artificial- and real-data traces, we experimentally show that our approach, when feasible, satisfies the personal location-privacy protection requirements, based on the privacy techniques employed. By comparing our results with both real and artificial trajectories, we establish that the effectiveness of our approach is independent of mobility patterns. Moreover, it is shown that our approach increases the utility of the participatory sensing application, as compared to static privacy-protection policies, especially when user mobility history is partially available at the adversary. We experimentally analyze in a thorough manner the trade-off between utility and privacy in the context of participatory sensing. Note that our approach is compatible with most continuous or sporadic location-based applications (including location-based services).

2.1 System Model and Performance Metrics

People are concerned about the potential (though unconfirmed) health risks due to base stations [110]. Therefore, in this chapter, we consider the application of electrosmog monitoring by means of participatory sensing, as this case study fits the continuous data dissemination scenario.

We assume that a mobile user can always submit sensor data using her own data plan through the cellular network. In this context, mobile users (or just "users") sense their environment

and send their sensor data to a certain data-collection entity called an aggregation server (AS). Such data is valuable only if it is accompanied by the location and time information, hence the reported data packets are triplets in the form of $\langle value, location, time \rangle$. Our objective is to provide the user with a statistical privacy guarantee at the lowest possible utility loss for the application in this setting.

In our approach, we avoid relying on a trusted third party, because in reality it is difficult to establish such an entity that is trusted by all participants. Furthermore, we assume that users do not collaborate with each other in order to protect their location privacy, because this approach is energy-costly and enables users to collude in order to breach others' privacy. In such a setting, hiding the identities of mobile users is rather unrealistic in the existing systems where registration by users is generally required. Consequently, we focus on user untraceability and do not consider hiding user identity as a protection mechanism (*i.e.,* the AS knows the source of each sensed data).

For presentation clarity and computational limitations, throughout the remainder of the chapter, we assume the monitored area to be partitioned into cells and the time to be slotted. Henceforth, we use the terms 'location' and 'grid cell' interchangeably. In the remainder of this section, we specify the adversary models, define personalized privacy, and describe metrics for the evaluation of privacy and utility.

2.1.1 Threat Models

We consider two threat models; the adversary in both is assumed to be the AS, who records and exploits the private information that it obtains. The communication between the AS and the users is assumed to be encrypted, and the AS knows the identities of users.

In the first model, the AS is assumed to be semi-honest [24], meaning that it follows the protocols, it does not collude with other entities and it does not tamper with the system to obtain private information about the users. Furthermore, it does not deploy devices to monitor the whereabouts of users (no global or local eavesdropping). As a result, it can only try to infer private information based on the data it collects from the users. In this model, the AS has no background information on the users' mobility.

In the second model, the AS is assumed to be an active adversary and to deploy a limited number of tracking devices constrained by cost and resources. In this regard, we assume that the AS is able to detect user presence in a fraction of locations and it uses the information collected to reconstruct the original traces. For example, the AS can do this by sniffing the control channels of the cellular communication where the handshakes between the users and the base stations are exchanged in clear text. At this point, we argue the AS cannot optimally choose the monitored locations, because it cannot know the location sensitivities of the individuals. One approach would be to monitor the hotspot or generally-sensitive areas (such as hospitals) in a city, but then any other user movement would not be captured.

Moreover, some users might not be very privacy-sensitive to their presence in hotspot areas. Therefore, we assume that the AS chooses randomly the locations to monitor. The AS uses the tracking data to build a *spatio-temporal probability distribution* for each user. For example, on Mondays 9am with probability 0.8 a particular user is at work, and the probabilities assigned to user's other possible locations sum up to 0.2. To reveal the user trajectories, this background knowledge is combined with the location information contained in the emitted data from the mobile users. Note that the notion of the spatio-temporal probability distribution is very generic and can model other kinds of background knowledge as well, *i.e.,* user habits, user location sensitivity, location semantics, etc.

We also assume that, in both threat models, the only other background information that the AS has about the users is their maximum possible speed (also known to the users themselves). Nevertheless, mobile users are assumed to be honest, which means that they do not attempt to tamper with their sensor measurements or collude with the adversary, but they might reduce the data accuracy (in terms of location/time), in order to protect their privacy. Last, we assume no interaction among users; consequently, there is no risk of potentially malicious users aiming to track other ones.

2.1.2 Personalized Privacy

In most of the existing location-privacy protection approaches [67], fixed parameters are statically employed in the proposed mechanisms for all the users. This approach has a negative effect on both the privacy levels of the users and the utility of the system, as will be shown in Section 2.3.

First of all, such a static approach does not take into account the trajectory history of users. It implicitly assumes that a uniform parameter for a particular location-privacy protection mechanism will always provide the same level of protection, which is not the case because spatio-temporal correlation between disclosed events might reveal partial or full trajectories of users.

Another problem resulting from this approach is the negative effect on the utility of the system due to the fact that in some cases the provided location privacy can be much higher than what a user actually wants. For example, a user might still achieve satisfactory privacy-protection by providing four grid cells in an obfuscated area instead of six, and therefore increase the system utility.

According to A. Westin [109], "each individual is continually engaged in a personal adjustment process in which he balances the desire for privacy with the desire for disclosure and communication of himself to others, in light of the environmental conditions and social norms set by the society in which he lives". In this spirit, considering the aforementioned issues about the static/uniform parameter selection, we define personalized privacy as follows:

Definition 1 *Given a set P of protection mechanisms and $P^{a_i} \subseteq P$ being the subset of mechanisms that can be implemented by the user i with ability a_i, the* personalized privacy *for user i with privacy threshold θ_i is defined by the formula:*

$$\exists p \in P^{a_i} : p(z, \omega, H) \geq \theta_i, \tag{2.1}$$

where $z \in Z$ is an instantiation of adversary capabilities Z, $\omega \in \Omega$ is a particular user action from the set of available actions Ω, H is the history of user actions, and $p(.)$ is the privacy level resulting from the mechanism p as estimated by the user.

According to this definition, personalized privacy is the individual's ability to employ all the necessary privacy protection mechanisms so as to adapt to privacy leakage resulting from his/her activities and/or the changing privacy-breaching capabilities of the adversary, as observed by the individual.

2.1.3 Privacy Metric

In our work, we measure location privacy as the expected error of the adversary by calculating the expected distortion as proposed by Shokri *et al.* [98] (see Section 1.2). In this chapter, we define the distance function dist(loc_1, loc_2) as a normalized Euclidean distance that gives distance between locations loc_1 and loc_2 in $[0,1]$. As a result, the computed privacy level is in the interval $[0,1]$, where 0 means no privacy protection and 1 means full privacy protection. This is done by normalizing the actual distance by an upper bound distance per time step (e.g., the maximum driving speed in our case). We choose to normalize it in this chapter for the sake of presenting results with a uniform upper bound on the privacy level.

2.1.4 Utility Metrics

The utility of participatory sensing applications is crucial to their emergence and economic sustainability. The utility in this context depends on the data quality, the data relevance to the application and the data availability. Here, we focus on data quality and availability aspects, namely the data accuracy, the data completeness and the area coverage. We analyze the effect of privacy protection on utility, based on the aspects explained below:

- *Data Accuracy*: As the users report imprecise or coarse-grained location (and/or time) information in their sensed data, an error is introduced in the measurements of other locations(and/or time instants). We measure the data inaccuracy by means of the average absolute error (L_1 norm) introduced to the sensed data due to location/time obfuscation. We express the average absolute error as a percentage of the data range.

- *Data Completeness*: One important factor that affects data availability is data loss; some of the sensed data collected by the users might not be emitted (data hiding) to the AS

due to privacy concerns. We define the data completeness as the percentage of the sensed data received by the AS.

- *Area Coverage*: Another component of the data availability is the percentage of the area of interest, where sensor measurements are done by users. Various privacy-enabling techniques differently affect the size of the total monitored area: e.g. while data hiding tends to decrease it, location obfuscation tends to increase it, as observed by the AS. As higher data hiding and larger obfuscation negatively affect the data completeness and accuracy, we define the area coverage as the fraction of the areas in which data is sensed over all areas for which data is reported by the users. Note that this metric is maximized at 1, *i.e.,* all areas where data is reported correspond to real points of sensor measurements.

2.2 Adaptive Protection Scheme

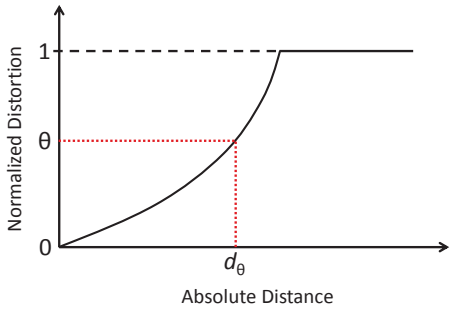


Figure 2.1 – Absolute distance vs. Distortion. θ is the desired privacy threshold and d_θ is the corresponding absolute distance to achieve θ.

In this section, we introduce a simple, yet effective location-privacy protection scheme, that is built upon the existing privacy-preserving techniques of location obfuscation and hiding. The main idea is that before each user submits data, she should be able to estimate locally her expected privacy-level and configure the protection mechanisms accordingly. This requires us to emulate an adversary's attack on user devices; however, due to limitations on processing power and also battery capacities, we need to achieve this by implementing a light-weight approach. The location-privacy quantification framework proposed by Shokri *et al.* [99, 100] is very comprehensive and useful, but it is computationally heavy, as explained in Section 2.1.3. Thus, we employ the distortion-based metric [98] and a Bayesian-network approach on the user-side, in order to calculate locally an estimate of user privacy-level on mobile devices. Note that in the remainder of this section, the term 'node' is used to refer to the users' mobile devices, because it is user's devices where the scheme runs and users do not take action.

We employ location obfuscation for confusing the aggregation server (AS) about the actual location of the sensed data. Location obfuscation is the generalization of the fine-grained location information; we designate its granularity with λ, which is the obfuscation parameter. As stated in Section 2.1, a location is a grid cell, and therefore, an obfuscated area is a set of

grid cells. In our strategy, a reasonable upper bound λ_{max} on λ is assumed, so that the sensor data remains useful for the participatory sensing application.

In our scheme, we want to let people have the privacy protection level they desire. In order to provide this, we define θ, the *personal privacy threshold*, which expresses the desired level (*i.e.,* the lower bound) of expected distortion (*i.e.,* distance) from the actual user location. This privacy threshold depends on the user's sensitivity about her privacy at a particular location, and it can be chosen by a user-specific function of the desired absolute distance from the sensitive location (cf. Figure 2.1).

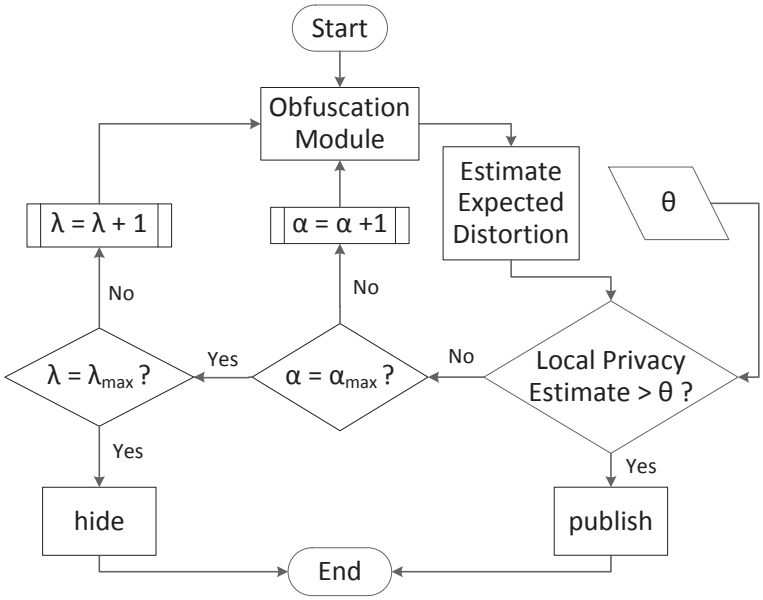


Figure 2.2 – Adaptive Location-Privacy Protection System. Expected distortion estimation keeps track of user history in case of active adversary assumption.

The algorithm for determining the obfuscation is as follows. When a node has data to submit, it calls the location obfuscation module with the lowest λ, *i.e.,* $\lambda = 1$. Then, it provides the output of this module —a set of locations constituting the obfuscation area— to the privacy level estimation module. The estimation is then compared against the node's privacy threshold θ. If θ is reached, then the node submits the data to the AS with the last generated obfuscation area. Otherwise, it increases λ and repeats the process. If λ_{max} is reached, but not θ, then the data is not submitted.

Our obfuscation algorithm randomly determines the obfuscation area as explained in Section 2.2.1. A randomly chosen obfuscation area might be ineffective, whereas another obfuscation area of the same size can provide sufficient privacy protection. Finding the optimal obfuscation area would be time and energy consuming, hence we introduce a limit (*i.e.,* by means of a counter) on the number of obfuscation areas we try: α_{max} per λ level. α is the number of obfuscation areas that have been tried for satisfying θ with the same λ value. As long as θ is

not reached and $\alpha < \alpha_{max}$, another obfuscation area of the same size is generated and privacy level is estimated based on this new area. Otherwise, if $\lambda < \lambda_{max}$, then λ is incremented and the process is repeated. Figure 2.2 shows this adaptive privacy protection strategy as a flowchart.

We explain, in Section 2.2.1, the obfuscation mechanism we employ and in Section 2.2.2 how local estimation is done.

2.2.1 Location Obfuscation Mechanism

The location obfuscation mechanism we employ in our proposed scheme and in the static mechanisms takes two inputs: the obfuscation level λ and the actual location l that is subject to obfuscation. Since the area of interest is discretized, the obfuscation area to be generated consists of a set of grid cells including the actual location/cell l. λ actually encodes the size of the obfuscation area in terms of cells. First, we determine the size $s_x \times s_y$ of the obfuscation area according to λ as follows:

$$s_x := 1 + \lceil \lambda/2 \rceil$$
$$s_y := 1 + \lfloor \lambda/2 \rfloor ,$$

where $\lceil . \rceil$ and $\lfloor . \rfloor$ are the ceiling and floor operations, respectively. Then, the area of size $s_x \times s_y$ cells is randomly positioned over the actual location l. Note that any deterministic choice for this positioning would render the area generalization ineffective in terms of privacy, because the adversary can find the actual location by trying different obfuscation areas iteratively. Randomization avoids the adversary from finding the actual location l, because in this case any of the locations in an obfuscation area is equally likely the actual location without any a priori knowledge. Note that some of the locations in an obfuscation area may be infeasible to reach from observed location in the previous time instant due to the maximum speed constraint. Such constraints are taken into account in the local privacy-level estimation described in the next subsection.

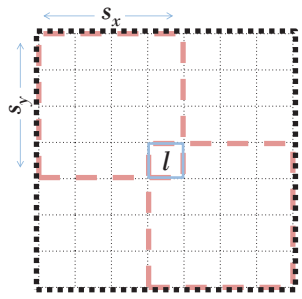


Figure 2.3 – The area in which an obfuscation area of size 4×4 ($s_x = s_y = 4$) can be positioned based on actual location l given $\lambda = 6$.

Figure 2.3 shows an example of this obfuscation mechanism over a gridded area, where the actual location l is in the center and $\lambda = 6$. The obfuscation area can be positioned in the bounding box in this figure, so long as l remains part of it. The two 4×4 squares in the figure represent the top-left and bottom-right possible obfuscation areas with $\lambda = 6$.

2.2.2 Local Privacy-Level Estimation

We calculate locally the expected privacy-level at the node, as explained below. We maintain locally an event linkability graph at each node, as the one depicted in Figure 2.4. Each vertex in this graph represents an event observable by an adversary at the corresponding time instant. An observable event corresponds to a data item associated to a particular location, which is sent to the AS. The linkability graph helps us to identify the trajectories that the AS observes and also to estimate its belief about their authenticity.

In order to build the linkability graph, a node needs to know the geographical topology of the area, *i.e.,* it needs to know the potential connectivity among different locations. It also needs to know the assumptions made by the AS for inferring user trajectories. To this end, one important assumption made by the AS is the maximum possible speed of a mobile node, which also determines the maximum possible distance between sequential vertices in time. A node can extract its own maximum speed from its traces, but it is not practical for the AS to know this value for each individual node. Nevertheless, he can make a global estimation on the average maximum speed and choose it as the upper bound for all the nodes he wants to attack. A node can construct, based on this knowledge, its linkability graph by connecting the vertices (*i.e.,* the observable events) that are adjacent in time and space.

Since the user may have to continuously disclose her data, hiding at a specific time instant does not provide her with full privacy protection. Technically, hiding, as perceived by the adversary, produces yet another obfuscation area that is the maximum feasible one based on the maximum speed and the user's previous reported locations. In practice, for the current time instant, this yields all the locations that are reachable from the locations in the previous time instant, and we consider all such locations as observable events in the privacy-level estimation.

The linkability graph is progressively constructed as new events are produced over time. The vertices corresponding to the current time-instant are connected to the vertices from the previous time-instant, based on the feasibility of being adjacent in space and time. If there are vertices with no children in the previous time instants, then these vertices are identified to be impossible and are removed. The same is applied to the vertices with no parents in the current time-instant. Note that the elimination of vertices needs to be propagated in the whole graph because some vertices in older time instants might lose all their children, which suggests that they are no longer probable locations of the node.

We use the linkability graph and employ the Bayes' rule to calculate the probability that

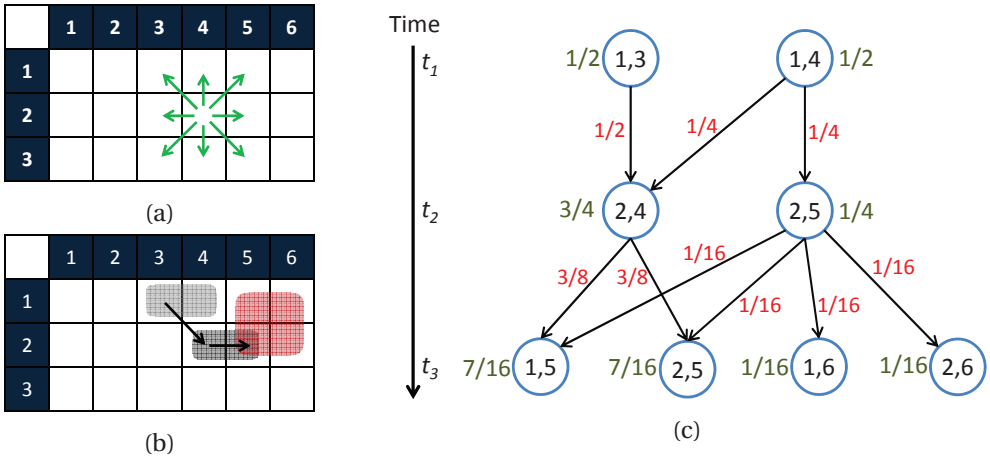


Figure 2.4 – Example of estimation of probabilities of possible trajectories for 3 time steps of a node. (a) Possible moves in a single step for a node on the given area. (b) Real trace of a node and its obfuscation decisions at each time instant. (c) Inferred linkability graph, on which the probabilities of being there are assigned to each edge and vertex. Each vertex on this graph represents an observed event from the node and the indices on the vertices are the location ids w.r.t. the area in Figs. (a) and (b).

an observed event corresponds to the actual location of the node. The first time observed events are inserted to the graph, a uniform probability $1/k$ is assigned to each vertex, as dictated by the k-anonymity employed by the chosen location obfuscation level, where k is the number of vertices. As new vertices are added at a subsequent time-instant, they can only be children of those in the previous time-instant and their probabilities of being genuine are calculated according to the Bayes' rule. Also, after the elimination of impossible events, the probabilities assigned to the siblings or parents of these events are updated and these updates are propagated in the graph. The probability of an event being genuine is depicted in Figure 2.4 as a label beside its corresponding vertex. We explain the calculation of these probabilities following the example of Figure 2.4. Initially, at time t_1, locations $(1,3)$ and $(1,4)$ are reported by the node and thus $\mathbf{Pr}(loc_{t_1} = (1,3)) = \mathbf{Pr}(loc_{t_1} = (1,4)) = 1/2$. Then, at time t_2, the node reports two locations to the AS, namely $(2,4)$ and $(2,5)$. We calculate the probability that location $(2,4)$ is genuine as follows:

$$\begin{aligned}\mathbf{Pr}(loc_{t_2} = (2,4)) &= \mathbf{Pr}(loc_{t_2} = (2,4) | loc_{t_1} = (1,3)) \cdot \mathbf{Pr}(loc_{t_1} = (1,3)) \\ &+ \mathbf{Pr}(loc_{t_2} = (2,4) | loc_{t_1} = (1,4)) \cdot \mathbf{Pr}(loc_{t_1} = (1,4)) \\ &= 1 \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2} = \frac{3}{4}\end{aligned}$$

The same approach is applied for location $(2,5)$ and for the 4 observed locations reported by the node at time t_3. [1]

[1] Note that the size of the obfuscation area at time t_3 is 2×2 (as shown in Figure 2.4-(b)), therefore there are 4 vertices corresponding to 4 reported locations at this time instant.

After having calculated the probability of each leaf vertex being genuine, a node calculates its location privacy according to Equation 1.1. For example, the location privacy LP of the node in our example at time t_3 is calculated as follows:

$$\begin{aligned} LP_u(t) &= \sum_{\Psi} \text{dist}(a_u(t), \Psi(t)) \cdot \mathbf{Pr}(\Psi, t) \\ &= 1 \cdot \frac{7}{16} + 0 \cdot \frac{7}{16} + \sqrt{2} \cdot \frac{1}{16} + 1 \cdot \frac{1}{16}, \end{aligned}$$

where $\text{dist}(\cdot,\cdot)$ stands for the Euclidean distance in this example.

Background Information So far, we have explained how the privacy leakage is estimated locally by a mobile node, under the assumption of no background information about the node's mobility at the adversary side. Now, we consider that some background information on the node's mobility is possessed by the adversary. Specifically, we assume that the adversary has, for each mobile node, a prior spatio-temporal probability distribution with PDF $\pi(X,t)$ over the locations X at time t that is built based on partial leakage of location information. As the mobile node does not know the exact leakage of its mobility, it samples its mobility history and builds a similar prior distribution, in order to accurately estimate its privacy leakage to the adversary by its emitted data. The prior distribution is employed to calculate the transition probabilities between successive locations.

For example, in Figure 2.5, assume a prior spatio-temporal distribution as follows:

$$\begin{aligned} &\pi(X=(1,3), t=t_1) = 1/16,\ \pi(X=(1,4), t=t_1) = 1/8 \\ &\pi(X=(2,4), t=t_2) = 1/10,\ \pi(X=(2,5), t=t_1) = 1/20 \\ &\pi(X=(1,5), t=t_3) = 1/10,\ \pi(X=(2,5), t=t_3) = 1/5 \\ &\pi(X=(1,6), t=t_3) = 1/10,\ \pi(X=(2,6), t=t_3) = 1/20\,. \end{aligned}$$

By employing this prior distribution for calculating the transition probabilities, we derive that:

$$\begin{aligned} \mathbf{Pr}(loc_{t_1} = (1,3)) &= \frac{\pi(X=(1,3), t=t_1)}{\pi(X=(1,3), t=t_1) + \pi(X=(1,4), t=t_1)} = 1/3 \\ \mathbf{Pr}(loc_{t_1} = (1,4)) &= \frac{\pi(X=(1,4), t=t_1)}{\pi(X=(1,3), t=t_1) + \pi(X=(1,4), t=t_1)} = 2/3 \\ \mathbf{Pr}(loc_{t_2} = (2,4) \mid loc_{t_1} = (1,3)) &= 1 \\ \mathbf{Pr}(loc_{t_2} = (2,4) \mid loc_{t_1} = (1,4)) &= \frac{\pi(X=(2,4), t=t_2)}{\pi(X=(2,4), t=t_2) + \pi(X=(2,5), t=t_2)} = 2/3 \\ \mathbf{Pr}(loc_{t_2} = (2,5) \mid loc_{t_1} = (1,4)) &= \frac{\pi(X=(2,5), t=t_2)}{\pi(X=(2,4), t=t_2) + \pi(X=(2,5), t=t_2)} = 1/3 \end{aligned}$$

Then, based on these transition probabilities, $\mathbf{Pr}(loc_{t_2} = (2,4))$ can be calculated again as follows:

$$\begin{aligned}\mathbf{Pr}(loc_{t_2} = (2,4)) &= \mathbf{Pr}(loc_{t_2} = (2,4)|loc_{t_1} = (1,3)) \cdot \mathbf{Pr}(loc_{t_1} = (1,3)) \\ &+ \mathbf{Pr}(loc_{t_2} = (2,4)|loc_{t_1} = (1,4)) \cdot \mathbf{Pr}(loc_{t_1} = (1,4)) \\ &= 1 \cdot \frac{1}{3} + \frac{2}{3} \cdot \frac{2}{3} = \frac{7}{9}\end{aligned}$$

Therefore, the background information can significantly affect the expected distortion that can be achieved by a privacy-protection strategy.

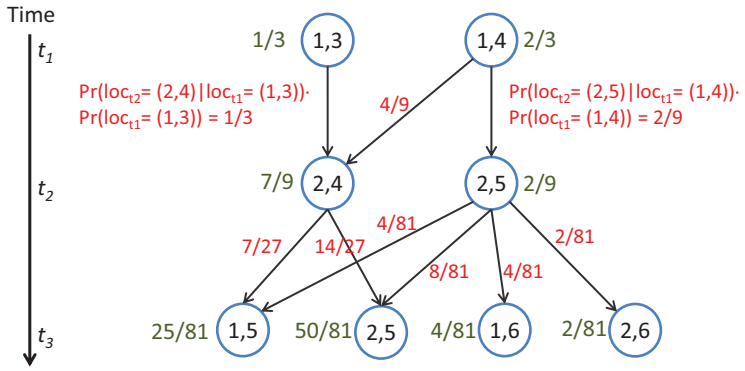


Figure 2.5 – Inferred linkability graph when background information is assumed to be available at the adversary. The prior spatio-temporal distribution is employed to find transition probabilities between successive location.

Complexity Analysis The complexity of our algorithm is dominated by the maintenance of the linkability graph. Each time a data submission is about to be made, the obfuscation module generates a maximum number of L locations constituting an obfuscation area. This operation has time complexity $O(L)$. The pairwise connectivity check between the locations in consecutive time instants takes $O(L^2)$ times. Later, the probabilities assigned to the current observed events are calculated in $O(L)$. Therefore, the time complexity of the whole process is $O(L^2)$. Note that this process has to be repeated until θ is met. The number of repetitions, however, is bounded by a constant maximum obfuscation parameter λ, thus, the total time complexity of the whole estimation and protection operation remains $O(L^2)$.

Shokri *et al.* [99, 100] developed a software tool, called *Location Privacy Meter* (LPM), which implements a quantification framework along with a localization attack based on Hidden Markov models. The purpose of this localization attack is to find the most likely location of a user, at each time instant, among all of her observed locations, based on her observable events both in the past and in the future. The complexity of this attack is $O(TM^2)$ [100] for one trace, where T is the number of time instants and M is the number of locations in the area of interest (*i.e.*, the monitored area). Compared to our simpler, but more lightweight inference scheme, the LPM has worse performance, as we explain in the following example: Given an

area of interest of 20×25 grid cells (*i.e.,* $M = 500$) and a maximum obfuscation parameter $\lambda = 10$, our algorithm's complexity is $O(36^2)$, whereas the complexity of LPM is $O(500^2)$ for one time instant, *i.e.,* almost 200 times slower (even more when considering more time instants).

Our approach is lightweight in terms of space requirements as well. In addition to map topology —which would be required by any client-side location-privacy protection mechanism— our scheme only stores the linkability graph, where each vertex has a probability value and location information. This results in $O(TL)$ vertices and $O(TL^2)$ edges in the worst-case, where T is the number of elapsed time instants. These storage requirements can be easily handled by modern mobile devices, that presumably have several GBs of storage capacity.

2.3 Evaluation

In this section, we assess the performance of our adaptive approach for protecting location-privacy and compare its effectiveness with that of static protection policies in terms of utility and privacy. To this end, we perform simulation experiments, with not only artificial data sets, but also real data traces (explained in Section 2.3.1). The estimate of the privacy level of a user, as observed by the AS, is measured by the LPM [99, 100]. This software tool provides an objective estimate of the privacy level of users, and its output belongs in $[0, 1]$ based on our normalized Euclidean distance function, with 0 meaning no privacy protection and 1 meaning maximum protection.

We replayed real data traces in a simulation environment, that we developed in C++, and ran experiments using artificial data traces (cf. Section 2.3.2). We implemented our adaptive strategy, along with static protection mechanisms of obfuscation and hiding, which works with fixed λ and hiding probability $\Pr_h$, respectively. λ_{max} was set to 10, which means that the largest possible obfuscation area is of size 6×6 grid cells. For expected distortion computation, we used Euclidean distance.

For the scenario in which some background information is assumed to be available at the adversary, we let the AS monitor all user presence in 25 random locations. Given a total of 500 grid cells in the sensed area, the expected number of node events observable at the adversary is given by the formula below:

$$\sum_{i=1}^{500} \frac{25}{500} \cdot (\#\text{ events generated in } l_i) \tag{2.2}$$

In our dataset, each node has around 20,000 events on the average. Given the above formula, the expected number of exposed events of a node corresponds to 1% of its generated events. The mobile node does not know which locations are monitored by the adversary. Although, knowing that the expected total number of its leaked events to the adversary is 1%, it can consider a random 1% of its generated events as the background knowledge available at the adversary. This gives the node a chance to take into account the adversarial background

Table 2.1 – Experiment Parameters.

	Adaptive	**Static**
$\boldsymbol{\theta}$	0.1 - 0.9	N/A
$\boldsymbol{\lambda}$	Adaptive	1 - 10
$\mathbf{Pr}_h$	N/A (Adaptive hiding)	0 - 0.9
# of Nodes (*i.e.*, users)	20	
Monitored Area	25×20	
$\boldsymbol{\lambda}_{max}$	10	

knowledge in the local inference module.

First, we compare our adaptive strategy to combinations of the aforementioned static mechanisms and experimentally prove the ineffectiveness of static policies at satisfying user privacy requirements. Then, we demonstrate that our simple local estimation of privacy is an accurate measurement. Subsequently, we analyze the trade-off between utility (*i.e.,* accuracy, area coverage, data completeness) and privacy for different static policies and our adaptive privacy-enabling policy. To this end, we define two different static policies for a given θ:

- *Avg Static:* This policy defines fixed $\mathbf{Pr}_h$ and λ that meet θ on the average; privacy violations are allowed from time to time.
- *Max Static:* This policy defines fixed $\mathbf{Pr}_h$ and λ so that θ is met most of the time. This is a rather conservative privacy-protection policy.

Note that these static policies employ the obfuscation mechanism described in Section 2.2.1 and apply this mechanism statically with the predefined parameters. They do not consider past or future events of the node when obfuscating the actual location.

Table 2.2 shows the experimentally identified static privacy-protection policies corresponding to each privacy threshold adapted in simulations, and Table 2.1 shows the parameters we have used for the experiments.

2.3.1 Real Data Trace

During the Lausanne Data Collection Campaign (LDCC) [85], run by Nokia Research Center (Lausanne), a dataset of around 200 users was collected. The data was collected over a year from 2009 to 2011, from smart-phones that were provided to the participants. We utilize 20 time-continuous user traces and we consider an area of 1.25×1.00km from this dataset and partition it into 25×20 grid cells. The traces we used in our simulations are one-day long and the time is slotted into 40 instants. We fixed the maximum possible speed to 4 grid cells per time instant after analyzing the maximum speed achieved in the real traces. Finally, for electrosmog measurements, we employ the logged signal strength in dBm from the campaign.

Table 2.2 – Parameters λ, $\Pr_h$ of the Avg and Max static policies experimentally found to satisfy the various privacy thresholds on average and most of the time respectively.

	Avg Static				**Max Static**			
	w/out BK		w/ BK		w/out BK		w/ BK	
θ	λ	$\Pr_h$	λ	$\Pr_h$	λ	$\Pr_h$	λ	$\Pr_h$
0.1	1	0	1	0.2	1	0.1	2	0.3
0.2	1	0.1	3	0.3	1	0.2	3	0.5
0.3	2	0.1	3	0.5	2	0.2	4	0.6
0.4	4	0.1	4	0.6	4	0.2	5	0.7
0.5	4	0.2	6	0.7	4	0.3	7	0.8
0.6	8	0.2	6	0.8	8	0.3	8	0.4
0.7	8	0.4	7	0.9	7	0.5	8	0.5
0.8	9	0.6	9	0.6	9	0.7	9	0.8
0.9	10	0.8	10	0.9	10	0.9	10	0.98

2.3.2 Artificial Data Trace

To facilitate the comparison of our results with artificial data to those obtained with real data, we assume an area of the same size (25×20 grid cells) as in the case of the experiments with real data. We assume 20 mobile nodes that move around with the random waypoint mobility model. The maximum speed is assumed to be 4 grid cells per time slot. At each time slot, a mobile node senses an electrosmog measurement (*i.e.,* the signal strength) and submits through privacy protection mechanisms.

We model the electrosmog generation for our simulations with artificial data as follows. The transmission power of base stations ranges from 10 W to 40 W, depending on the network characteristics; we choose 20 W as the base station transmission power in our setting. The frequency of channel is set to 900 MHz as in GSM. We implement free space path loss on this value for each grid cell. There is one base station centered in the area of interest and it covers the whole area in our simulation. We also apply the Rayleigh fast-fading model upon the free-space path loss to simulate a realistic urban area electromagnetic field distribution. Equation 2.3 shows the free-space path loss PL, where f is frequency in MHz and d is distance in meters. Equation 2.4 shows the Rayleigh distribution, where R is the power in Watt, and σ is the parameter of the Rayleigh distribution; we use the Rayleigh simulator proposed by Komninakis [64] to apply Rayleigh fading in this setup. Note that, for different frequencies, the characteristics of the electrosmog change, but currently the other existing frequencies in use are greater than 900 MHz, which means that the path loss will be much higher. Therefore, the measurements will yield lower values of electrosmog as the distance increases. In this sense, the loss of generality is negligible in regard to our choice of channel frequency.

$$PL = 20\log(f) + 20\log(d) - 27.55 \tag{2.3}$$

$$\mathbf{Pr}(R) = \frac{R}{\sigma^2} e^{-\frac{R^2}{2\sigma^2}} \tag{2.4}$$

2.3.3 Results

Ineffectiveness of Static Policies

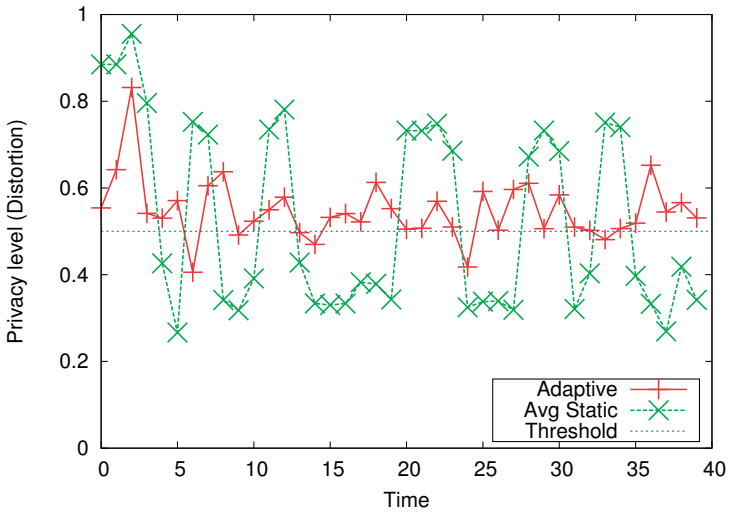
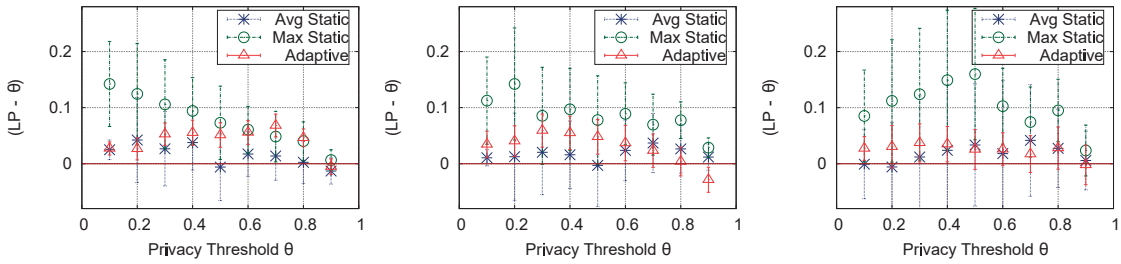


Figure 2.6 – Privacy levels measured by the LPM over time (40 time instants in this example) for part of one real trajectory in cases of adaptive and average static policies given $\theta = 0.5$.

As explained in Section 2.1.2, as nodes move and emit sensor data, the spatio-temporal correlation between events can occasionally violate user θ when a static privacy policy is employed. For example, we consider the time series of electrosmog measurements emitted by a certain real user. We assume that $\theta = 0.5$ for this user. As depicted in Figure 2.6, a static protection policy, which satisfies θ on the average, often results in significant privacy violations. Our adaptive privacy-protection strategy, on the contrary, dynamically adjusts location obfuscation and hiding behavior to almost always meet θ. Note, at this point, that we measure user privacy in an objective way from the AS point of view by employing the LPM in this figure. The LPM tends to be a bit more conservative than the privacy level estimated locally at the node (although highly correlated as shown later), which does not violate the user privacy requirement by definition, as long as hiding is not chosen (hiding is the last resort for a node to protect privacy. If θ is not met with even the largest λ, then it is possible that hiding is also not enough). Another interesting aspect in Figure 2.6 is that our adaptive privacy policy meets θ as minimally as possible, given the employed techniques for location obfuscation.

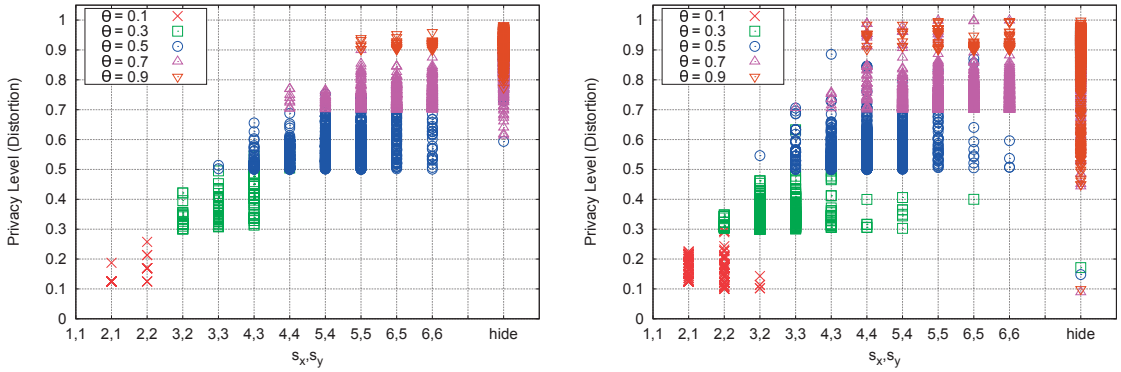
For all nodes from the real-data traces, the adaptive privacy policy needs to use a number of different obfuscation levels and hiding probabilities in order to meet different θ values, as depicted in Figure 2.8. Evidently, due to the fluctuations of the privacy exposure of the users caused by their mobility patterns, a wide spectrum of parameters has to be used for achieving different privacy thresholds. This result is also experimentally verified by the artificial data traces. In addition, as shown in Figure 2.7, the "Avg Static" policy violates thresholds almost half of the time, whereas the adaptive strategy almost always meets them for both real and artificial data. Average values over all users and over all times are plotted in this figure, with a



(a) (b) (c)

Figure 2.7 – Level of privacy achieved by adaptive vs. static policies with (a) real and (b) artificial data traces. (c) Level of privacy achieved in the case of background information available to the adversary with real data traces.

confidence of interval 95%. Note that meeting $\theta = 0.9$ is very strict and sometimes infeasible with the employed location privacy-enabling techniques.

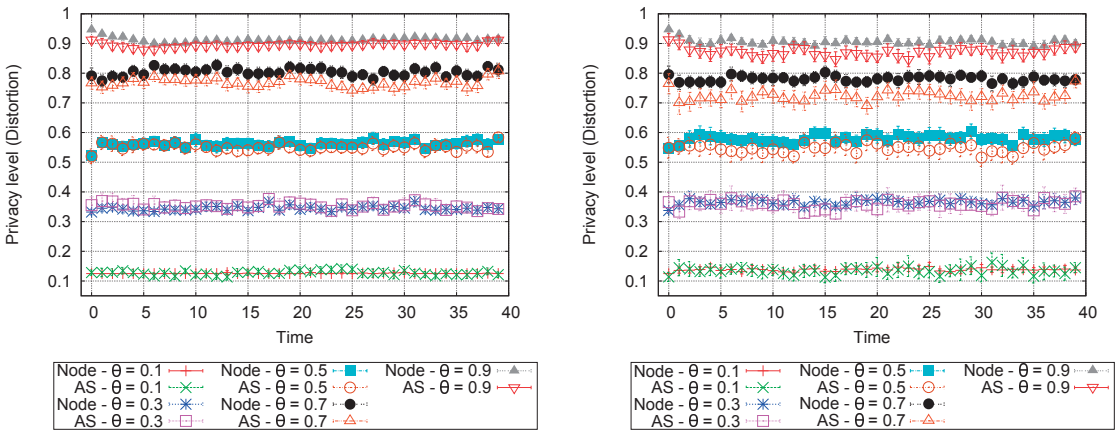


(a) Without background information (b) With background information

Figure 2.8 – Parameters ($\lambda = s_x + s_y - 2$) chosen by the adaptive strategy for all (real) users over all time steps vs. the local estimations of privacy levels (a) without background information and (b) with background information.

Local Estimation of Privacy

Figure 2.9 shows the privacy levels achieved by the adaptive strategy, as estimated locally at the nodes and externally by the LPM for different θ values. These results represent average values and confidence intervals over all nodes. As shown for both real- and artificial-data traces, privacy estimations by our simple approach are highly correlated to the estimations by the LPM (*i.e.,* Pearson correlation > 0.5) for all θ values. Therefore, our simple approach is accurate enough to locally estimate the level of location-privacy of mobile users.



(a) Real data traces

(b) Artificial data traces

Figure 2.9 – Comparison of local privacy estimation (Node) to measurements by LPM (AS) over time with (a) real and (b) artificial data traces.

Utility vs. Privacy

In Figure 2.7, observe that the adaptive protection strategy meets the various privacy thresholds more narrowly, as compared to the "Max Static" policy. As a result, the adaptive strategy is expected to deteriorate the utility of the participatory sensing application less than any static one, while satisfying the privacy requirements of the users. Indeed, the absolute error as a percentage of the data range introduced by the adaptive strategy is lower than the respective errors by the two static policies, as shown in Figure 2.10. The results in this and the subsequent figures are average values over all data items from all users and over all times with a confidence interval of 95%. Note that the results of the real- and the artificial-data traces are similar, despite the significant difference in the mobility behavior of the users.

Moreover, we show the data loss from the two static policies and our adaptive policy in Figure 2.11. Notice that the data loss is significantly lower for reasonable privacy requirements of the users, *i.e.,* lower than 0.8 for real data and always for artificial data. Also, the data loss for $\theta \leq 0.6$ is almost insignificant ($\sim$ 15% or less) for the adaptive policy, and it is double or more for the two static policies for $\theta \geq 0.2$. This was expected, as static policies need to employ a non-zero $\Pr_h$ throughout the sensing process in order to satisfy even low θ values, as opposed to our adaptive strategy that hides sensor data only when needed.

We measure the deterioration of the area coverage by the ratio of the actual sensed area over the total area reported as sensed. As shown in Figure 2.12, this utility metric deteriorates significantly with high θ values. Although, the area coverage degrades smoothly when the adaptive strategy is employed, as opposed to the static policies. Note that in Figures 2.10 and 2.11 there are small fluctuations; this is due to mobility patterns of the users and also the probabilistic nature of data hiding for static policies.

Overall, Figures 2.10, 2.11, and 2.12 clearly demonstrate the *trade-off* between utility and

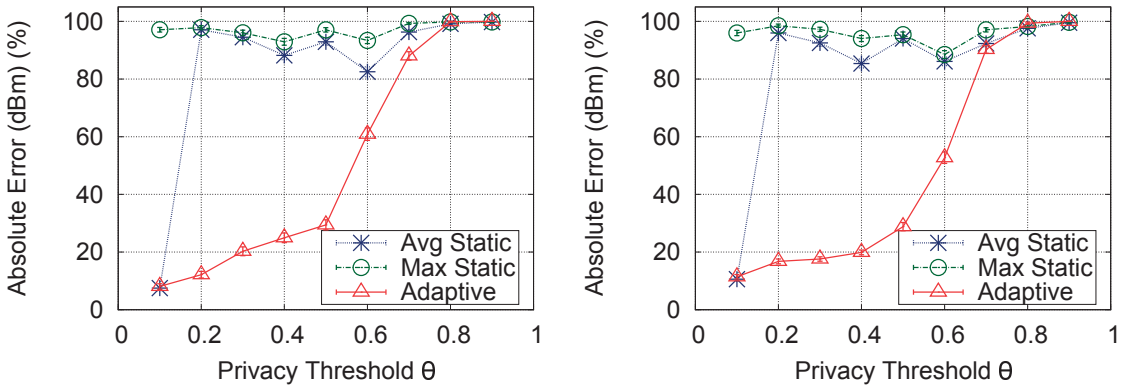


Figure 2.10 – Percentaged absolute error (dBm) for (a) real and (b) artificial traces.

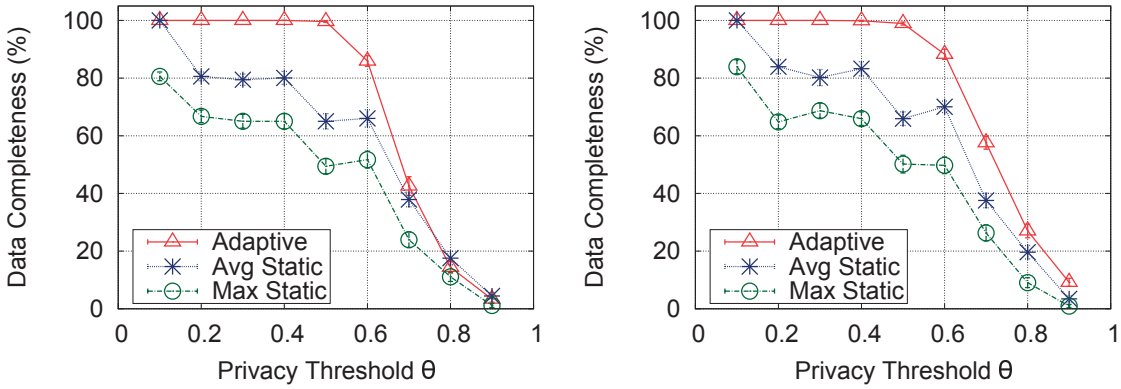


Figure 2.11 – Data completeness with (a) real and (b) artificial traces.

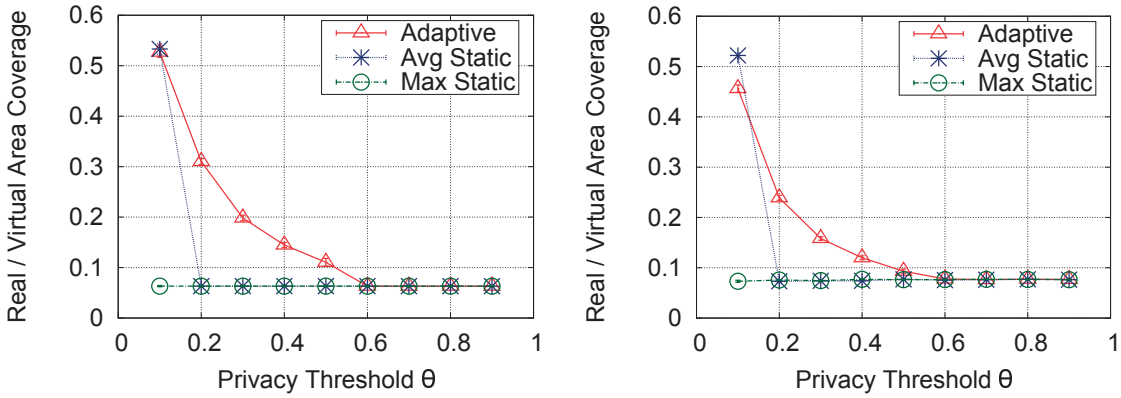


Figure 2.12 – Area coverage with (a) real and (b) artificial traces.

privacy. Our results can be employed to derive feasibility conditions on the application-utility and user-privacy requirements for the realization of a participatory sensing application. Our adaptive privacy-protection strategy dominates any static strategies that involve the same location-privacy protection techniques in terms of utility for any user-privacy requirements that render the participatory sensing application feasible.

Adversary Background Information

Here, we run experiments with the threat model that involves background information at the adversary. The impact of adversarial background information on the chosen privacy parameters by the adaptive privacy-protection strategy is depicted in Figure 2.8(b). As observed therein, our adaptive strategy performs almost as well as in the case of no background information (cf. Figure 2.8(a)). This is due to the fact that the nodes choose obfuscation parameters smartly being aware of the adversary's background information. Moreover, in fact, the prior background information causes the adversary to be biased and therefore enable the nodes to choose parameters lower than before. For example, for $\theta = 0.3$, some nodes chose strategy 2,2 in Figure 2.8b, as opposed to the case in Figure 2.8a. Also, as depicted in Figure 2.7, the user loses some privacy due to the adversary background information, but the amount of loss is negligible. Our adaptive approach is still adaptive enough to protect the user privacy in this threat model, despite the existence of background information at the adversary.

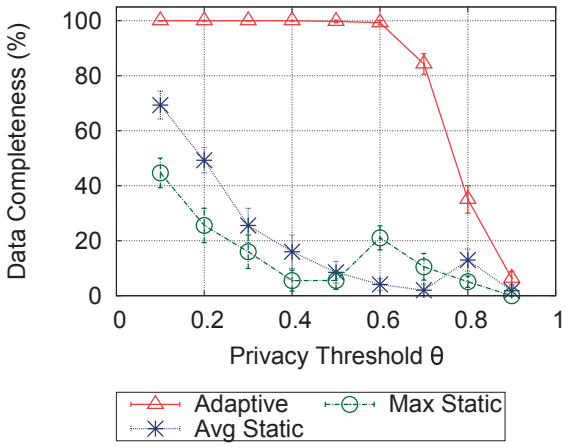


Figure 2.13 – Data completeness in percentage in the case of adversarial background information.

Regarding utility, comparing the data completeness in the cases with background information at the adversary (cf. Figure 2.13) and without background information (cf. Figure 2.11), we observe that they exhibit similar trends. Thus, introducing some background information to the adversary does not cause utility deterioration when the adaptive privacy-protection strategy is employed. However, notice that both static privacy-protection strategies significantly deteriorate utility, since they need to employ larger static parameters than before, in order to meet the various privacy thresholds. Similar findings are observed for the other utility parameters, namely data accuracy and area coverage, when background information is available at the adversary. Therefore, in the presence of background information at the adversary, the employment of an adaptive privacy-protection scheme becomes more important.

2.4 Discussion

In this chapter, so far we have considered the maximum speed of users, known real identities of sensor-data sources, and additional background information on user mobility history as background knowledge at the adversary. Repetitive trajectories of a user can be another type of background knowledge, which, when accumulated at the adversary, can pose additional privacy threats for the user: People generally move in regular patterns (*i.e.,* on a daily basis), for example from home to work in the morning. If a user generates a different obfuscated trajectory each time he moves along the same trajectory, then an adversary can find the real trajectory in time. However, a simple modification to our adaptive privacy-preserving scheme can eliminate this threat. Specifically, a mobile user has to keep track of her privacy preserving actions along repetitive trajectories (*i.e.,* the obfuscated area for each actual location, per repetitive trajectory) and reuse them in the future. In this way, the privacy leakage due to repetitive trajectories would be limited. Moreover, in order to keep storage overhead bounded, a LRU replacement policy could be employed. The experimental evaluation of this approach is left for future work.

Also, an adversary might employ other background information, such as location semantics, which could be of great help for identifying the real user traces and the user activities. This kind of background information can be modeled as probability distributions over space and time for each user and included in the Bayesian inference model employed on the user device. We investigate the effect of location semantics on location privacy in Chapter 5.

Another issue that needs addressing is the usability of our approach. We designed our scheme with an automated software tool embedded in localization modules in mind. The average mobile user would not like having to interact with his device every time his location is used by an application. Therefore, our system should be triggered automatically whenever an application asks for the user's location. Such an automation will provide users with a peace of mind in terms of privacy protection. Furthermore, it might not be straightforward for mobile users to interpret the privacy levels (*i.e.,* the expected distortion values, and hence their θ inputs). Here, the privacy levels need to be conveyed to users in the form of "average confusion from actual location in meters", which can be done using the normalization factor used in the estimation. For example, in our experiments, the normalization is done by 4-hop distance (*i.e.,* the max-speed) which is around 200 meters. In this case, a privacy level of 0.7 yields a confusion of around 140 meters from the actual location.

Last but not least, our system can be extended with location sensitivities, where users input different θ values for their sensitive locations. This would result in an even more dynamic and personalized privacy-protection system. Such an extension can even take into account location semantics, which would enable the batch setting of the user privacy thresholds.

2.5 Related Work

Privacy-preserving participatory sensing has been widely addressed by the research community in the past [31] including privacy of data itself, of data source identity and of user location. The downside of any privacy-preserving mechanism in data-driven applications such as participatory sensing is the potential loss of accuracy or precision in the reported data and/or loss of samples. Krause *et al.* [65] address location privacy and experimentally analyze the trade-off between accuracy and privacy. They employ two methods of location-privacy protection: location obfuscation and sparse querying. The combination of these two methods diversify the users chosen for querying in order to minimize the privacy breach of a single individual user. In this chapter, we significantly enhance the study of the trade-off between utility and privacy by studying the effect of privacy on additional utility aspects apart from accuracy, namely data completeness and area coverage.

According to Xiao *et al.* [111], any privacy-preserving mechanism should consider the personalized privacy requirements of the participating users, because individuals typically have varying privacy requirements. In addition, we argue and experimentally show in Section 2.3 that data utility can also be improved by personalized privacy protection that avoids excessive privacy preservation. Xiao *et al.* [111] formalize personal privacy specifications and apply a data generalization technique for satisfying individual privacy requirements. Gedik *et al.* [49] propose a location-privacy protection mechanism based on personalized k-anonymity for location-based services (LBS). However, they employ a trusted third-party that implements the privacy-protection scheme, which is contrary to our approach; similarly, Vu [107] *et al.* also propose a trusted third-party based k-anonymity approach. In [35, 83], the users might decide to selectively activate sensing (and hide in other times) depending on a variety of factors, such as presence in sensitive locations (home or office), or their current social surroundings (presence of friends or family members). However, hiding is applied not based on a rigorous privacy assessment, but based on a fixed probability value. Minami *et al.* [80] analytically prove that trajectory inference is still possible in a LBS if data hiding is the only mechanism used for location-privacy protection and they suggest designing new policies that consider users' past events, as we do in this chapter in the context of participatory sensing.

In addition to being a client-based location-privacy preserving mechanism, our approach supports continuous location dissemination. Several client-based solutions exist in the literature [60, 61, 89, 96]. SybilQuery [96] generates, for each user's query, $k-1$ other queries so that the LBS server cannot distinguish the real query from the Sybil ones. However, this work requires the user to determine a priori the source and destination of the real query, thus it does not support real-time continuous dissemination. In addition, it does not apply any transformation/obfuscation on the trajectories, which allows an adversary to obtain the full real trajectory, once it is identified partially. A distributed k-anonymity cloaking mechanism is proposed in [60], which identifies neighbors using on-board wireless modules and exploits secure multi-party computation in a collaborative manner in order to compute a cloaking region. However, this work does not support continuous querying. Finally, Jadliwala [61] *et al.*

present a concept called *privacy-triggered communications*, which is a generic framework that fits the work we present in this chapter; however our work differs in detailed utility and privacy analysis. Cappos *et al.* [26] proposes a concrete framework for access control to private sensor data on mobile devices. Their framework wrap the sensors on a mobile device and let privacy filters to be applied on them based on the user's requirements.

Last but not least, other work [32, 36] propose cryptographic approaches for protecting the identity of the participants in participatory sensing. Groat *et al.* [56] consider multidimensional data to evaluate the user privacy, *i.e.,* they consider spatio-temporal dimensions, the sensed data and more. But, they do not take into account the continuous data disclosure, which would be disastrous for the users in case of an attack on a multidimensional scale. More on the privacy issues in participatory sensing applications can be found in the survey paper by Christin *et al.* [31].

In summary, to the best of our knowledge, none of the existing work proposes a location-privacy protection scheme combining the following properties: (i) dynamic estimation of user privacy based on the history of mobility and data submissions, (ii) adaptive satisfaction of personalized privacy requirements, (iii) user-side residence, and (iv) independence of any trusted third parties.

2.6 Summary

In the context of participatory sensing, we have defined a simple, yet effective, adaptive location-privacy protection scheme. Our approach is based on estimating locally in real-time the expected location-privacy level at the user-side, which enables her to adapt her privacy parameters with respect to her mobility, in order to satisfy an *individual* privacy constraint. We have experimentally showed the accuracy of our approach for privacy estimation and the effectiveness of our adaptive privacy-protection strategy, as opposed to static ones. Our adaptive approach achieves more application utility than static policies, and satisfies the individual privacy requirements of the users in case whether background information on the user's mobility history is available to the adversary or not. Furthermore, we have demonstrated the *trade-off* between application utility and user privacy in the context of participatory sensing. As experimentally found, our adaptive privacy-protection scheme is able to maintain high data utility, while satisfying the user privacy requirements. Our results can be used to derive feasibility conditions on the desired application utility and user privacy requirements. The proposed approach is easy to deploy on current mobile devices and supports continuous and sporadic location dissemination by users.

3 Mobility-aware Location-Privacy Protection

Various approaches have been proposed to ensure good levels of location privacy in location-based mobile systems. A commonly adopted approach, as stated in Chapter 1, is to apply obfuscation on locations of users (*i.e.,* to deliberately degrade the quality of location information). However, as discussed in Chapter 2, location obfuscation has shown effectiveness but studies also revealed some weaknesses of the approach in mobile applications where location data was continuously disclosed. Against a reasoning adversary that has access to the geographical context and the mobility of a user, simple obfuscation might prove to be an inadequate privacy-protection mechanism (PPM) and may result in reduced levels of location privacy in successive time steps, as the user moves.

In this chapter, we propose a heuristic location-privacy protection mechanism that is aware of the user mobility when determining the obfuscation area for location in order to minimize the deterioration of location privacy over time. We call this heuristic approach *mobility-aware* location-privacy protection due to its awareness of user mobility history, direction of movement and speed of the user. We experimentally evaluate our heuristic mechanism and show that this approach provides a high level of location-privacy as compared to a random obfuscation mechanism against attackers both with and without knowledge on user history. This algorithm can be combined with the adaptive protection scheme in Chapter 2 in order to benefit from the power of mobility-awareness and adversary-awareness all at the same time.

* The work presented in this chapter is joint work with Iris Safaka and Malik Beytrison.

3.1 Framework

In principle, we use the same framework as in Chapter 2: We consider mobile users equipped with smartphones moving in a geographical area that is discretized to M non-overlapping regions, *i.e.,* $\mathcal{R} = \{r_1, r_2, ..., r_M\}$ in discrete time space $\mathcal{T} = \{t_1, t_2, ..., t_N\}$. They send their location at every time instant $t_i \in \mathcal{T}$ to a server for, e.g., receiving a location-based service or contributing to a sensing application. When sending her data, a user obfuscates her location. An obfuscated location $\mathcal{L}_t$ at time t is a set of locations from $\mathcal{R}$, ie $\mathcal{L}_t \subset \mathcal{R}$.

(a) Actual location of a user (b) Obfuscated location

Figure 3.1 – Location obfuscation example

While performing location obfuscation, a user is revealing a set of c locations, *i.e., c* regions, at each time instant t_i. We will be referring to the parameter c as the *location obfuscation parameter*. Clearly, the value of this parameter is determined by the level of privacy the user wishes for and there exists a trade-off between the utility of the application (location-based service or sensing application) and the user's level of privacy. Deciding on the location obfuscation parameter is out of the scope of this chapter and we will be using a constant value for this parameter.

The following example summarizes the above. The area in Figure 3.1 is discretized into 16 regions and a user is moving in this area. At each time instant, she has to report c locations, out of which, one is her actual location and c-1 are fake. For example, on time instant t_i the user declares the set {5,6,7,9,10,11} (Fig. 3.1b) instead of only 6 (Fig. 3.1a) which is her real position. In this example, the fake locations were selected randomly. In Section 3.3, we will explain how to choose the fake locations so as to minimize the deterioration of privacy level at time instant t_{i+1}.

We use the notation in Table 3.1 in the rest of the chapter.

3.1.1 Adversary Model

We assume a passive and curious adversary (which can be the server), *i.e.,* he will be able to observe the (obfuscated) locations a user reports to the server and try to infer her actual trace from his observations, but never attempt to break protocols or hack otherwise to obtain more information. Furthermore, he may know a user's past traces (which is called background knowledge) and use them in his inference attacks to reduce his confusion. The intuition

Table 3.1 – Table of Notations

$c \geq 2$	location obfuscation parameter
$\mathcal{T} = \{0, 1, 2, \ldots, t\}$	set of time instants
$\mathcal{R} = \{r_1, r_2, \ldots, r_N\}$	set of N distinct regions in the area of interest
$\mathcal{L}_t \subset \mathcal{R}$	set of locations reported by the user at time t
$a(t) \in \mathcal{L}_t$	the actual location of the user at time t
$l_i \in \mathcal{L}_t$	the i_{th} fake location at time t, where $1 \leq i \leq c-1$
$\mathbf{neigh} : \mathcal{R} \rightarrow \mathcal{P}(\mathcal{R})$	function that gives the neighboring locations of a location
$\mathbf{Pr}_{r\rho} : \mathcal{R} \times \mathcal{R} \rightarrow [0, 1]$	the probability to go to region r from region ρ given by Equation 3.3

behind this is that people tend to have regular mobility. The background knowledge he has may or may not be complete. Additionally, he knows the maximum possible speed in terms of regions per time instant, at which a user can move. The adversary's goal is to infer the actual location of a user at each time instant by using his background information and the user's obfuscated trace.

3.1.2 User Mobility Model

In our setup, we are considering location obfuscation in successive time steps. Given the knowledge of the maximum speed, the past behavior and the direction of the user, the transitions between successive cells are characterized by probabilities. An adversary who has knowledge of these probabilities could reduce his uncertainty regarding the user's real location after observing her obfuscated location. The adversary can also benefit from road networks and maps of inaccessible locations when constructing such probabilities. For the sake of simplicity, we do not consider this type of knowledge, but our model is independent of it (*i.e.,* this type of knowledge can be easily integrated into the model). In this subsection we explain the user-mobility prediction models that we use in the design of the heuristic algorithm. We use three mobility models: a history-based, a direction-based and a combination of the two.

History-based Mobility Model

We adapt the human mobility model proposed by Calabrese *et al.* [22], which aims to predict a person's future location based on the individual's past behavior.

We denote the location of a user at time t as $a(t) = \rho$. The model predicts the user's next location $a(t+1)$ using past data. This is done by following a probabilistic approach: A probability is defined for each region $r \in \mathcal{R}$ to be the next location of the user as a function of the user's past behavior. We assume that the behavior of user is periodic over time with period T as modeled in [22]. More precisely, this probability is given by the formula:

$$\mathbf{Pr}_h(a(t+1) = r | a(t) = \rho) = \frac{\sum_{m=1}^{\lfloor t/T \rfloor} f_h(a(t - Tm + 1) = r | a(t - Tm) = \rho)}{\lfloor t/T \rfloor}, \forall r \in \mathcal{R} \tag{3.1}$$

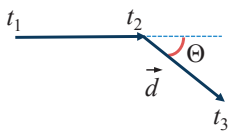
where the frequency f_h on the right hand side is defined as:

$$f_h(a(t+1)=r|a(t)=\rho)=\begin{cases}1 & \text{if } a(t+1)=r \text{ and } a(t)=\rho\\ 0 & \text{otherwise}\end{cases} \tag{3.2}$$

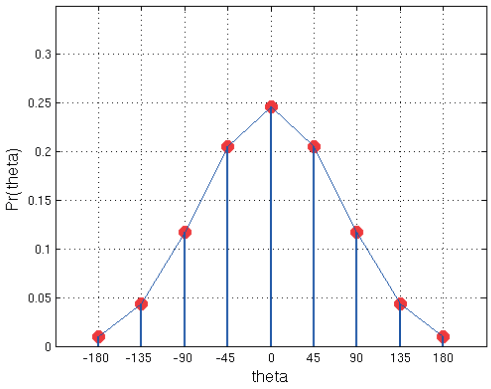
This model says that the probability of a region r to be the next destination of the user is equal to the frequency of visiting that region starting from region r_j during all previous periods $t-T+1, t-2T+1, \ldots$. In [22], experimental results demonstrate a rather promising accuracy of this human-mobility model as compared to original traces used.

Direction-based Mobility Model

Direction-based mobility models are often used to model mobility in ad-hoc networks [23], since they are considered more realistic compared to fluid-flow or random-walk mobility models. The Gauss-Markov mobility model [73] falls into this category of models. Using this model, the mobile-user's next location is predicted based on the information gathered from the user's last location report, velocity and direction. We adopt a simple version of the model: The idea is that it is more probable for a user to continue straight ahead rather than abruptly turning back while moving. The following example aims to give an intuition of this model.



(a) User direction vector $\vec{d}$ with its angle Θ w.r.t. previous direction.



(b) The probability distribution over direction change angle Θ showing that keeping the direction has the highest probability.

Figure 3.2 – The direction-based model that considers the angle of direction change in user movement.

Consider a user at time t and let $\vec{d}$ be the vector of her velocity, *i.e.*, his direction, at time t as shown in Figure 3.2. The probability $\mathbf{Pr}(\Theta)$ that the user will change her direction by $\Theta \in [-180°, 180°]$ degrees at time $t+1$ takes values with respect to a normal distribution. Finally $\mathbf{Pr}_d(a(t+1)=r|a(t)=\rho, a(t-1)=\rho') = \mathbf{Pr}(\Theta)$, where the coordinates of regions $\rho \in \mathcal{R}$ and $\rho' \in \mathcal{R}$ are used to identify the direction $\vec{d}$ at time t, *i.e.*, we need the last two visited regions to determine the movement direction of the user.

The Combined Model

In this Section, we define a model to predict a user's behavior as a combination of the history-based model and the direction-based model:

$$\mathbf{Pr}_{r\rho}(a(t+1)=r|a(t)=\rho)=\alpha\cdot\mathbf{Pr}_h+(1-\alpha)\cdot\mathbf{Pr}_d \tag{3.3}$$

where $\alpha\in[0,1]$ is the combination parameter and can change over time to model occasions when the user's behavior is more likely to be accurately predicted by her history data, if these are enough and available, and occasions where the direction-based model is better suited to predict future movement of the user.

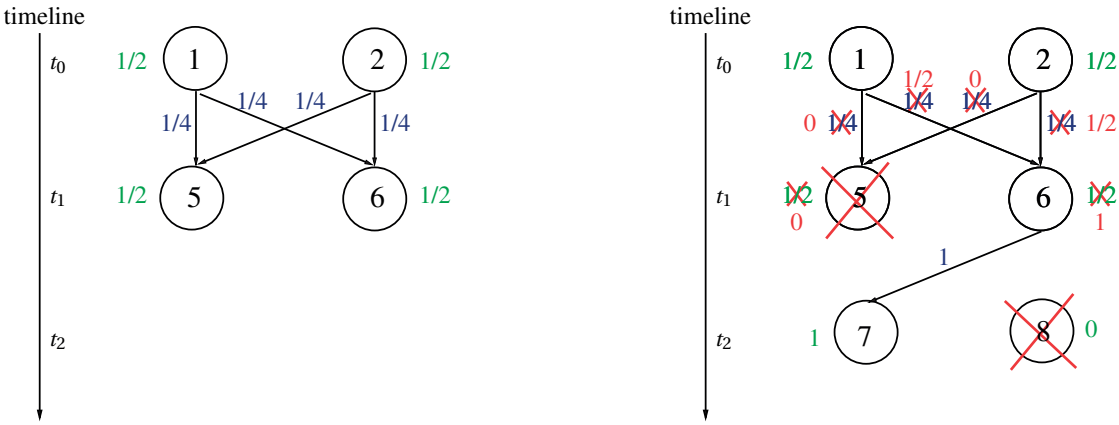
3.2 Problem Statement

Before presenting the heuristic algorithm, we state the algorithm design problem through an example. For demonstration reasons we use a directed *linkability graph* as shown in Figure 3.3, where the vertices are labeled after the reported locations at each time instant and a link between two vertices exists if a transition is possible between them in successive time instants. Each vertex is assigned a *presence* probability $\mathbf{Pr}(a(t)=r)$, where r is also the label of the vertex representing region r. Each link is assigned a transition probability $\mathbf{Pr}(a(t)=r|a(t-1)=\rho)\cdot\mathbf{Pr}(a(t-1)=\rho)\neq 0$, where ρ is the origin vertex and r the destination (zero if the link does not exist). Using Bayesian inference we can calculate that a location r is the real one as follows:

$$\mathbf{Pr}(a(t)=r)=\sum_{\rho\in\mathcal{L}_{t-1}}\mathbf{Pr}(a(t)=r|a(t-1)=\rho)\cdot\mathbf{Pr}(a(t-1)=\rho) \tag{3.4}$$

For the sake of simplicity, we assume in this section that transitions between vertices are equiprobable, therefore $\mathbf{Pr}(a(t)=r|a(t-1)=\rho)$ follows the uniform distribution, where $\sum_{r\in\mathcal{L}_t}\mathbf{Pr}(a(t)=r|a(t-1)=\rho)=1$, $\forall\rho\in\mathcal{L}_{t-1}$. Also, the maximum speed of a user is one region per time instant. We will demonstrate that under these assumptions, a user, who obfuscates her location, can get decreased privacy levels in consecutive time instants. Consider the discretized area seen before and illustrated in Figure 3.1. We set the obfuscation parameter c to 2, the user moves within an area of 16 regions, her maximum speed is one cell per time unit and her trace is {2,6,7}. Ideally, for privacy protection, the linkability graph should not become disjoint and there should always an outgoing edge from every vertex in the previous time instants.

Let's assume that the user reports her actual location $a(t_0)=2$ along with a fake one $l_1=1$ to the server at time t_0, thus $\mathbf{Pr}(a(t_0)=2)=\mathbf{Pr}(a(t_0)=1)=\frac{1}{2}$. At time t_1, the user reports $a(t_1)=6$ and $l_1=5$ and we can compute the probability for each one of those being the real position using Equation 3.4, namely $\mathbf{Pr}(a(t_1)=6)=\mathbf{Pr}(a(t_1)=5)=\frac{1}{2}$. An adversary still has the highest uncertainty regarding the real position (Figure 3.3a). At t_2 the user chooses to report



(a) Privacy level high

(b) Privacy level low

Figure 3.3 – Example showing deterioration in privacy level

$a(t_2) = 7$ and $l_1 = 8$. Apparently, there are some links missing now, since region 7 is impossible to reach from region 5 and region 8 from both 5 and 6 in one time instant. An adversary has now information that would reduce his uncertainty: by exploiting the information at time t_2 and using Bayesian inference, he can recompute the probabilities at time instants t_0 and t_1. Applying Equation 3.4 to the new values she concludes that $\mathbf{Pr}(a(t_2) = 7) = 1$ and $\mathbf{Pr}(a(t_2) = 8) = 0$, meaning that he inferred the actual location of the user (see Figure 3.3b).

One can easily observe that the decrease of privacy level at time t_2 occurred due to the fact that the selection of region 8 as fake was done randomly. If the selection criterion was to select as fake a region that is a direct neighbor of both regions 5 and 6, then the linkability graph would not have become disjoint and no vertex would be removed, thus the uncertainty would have remained high at time step t_2. This is a design specification that we take into account in the proposed heuristic algorithm of the next section.

Moreover, from the analysis above one can notice that deterioration of privacy level might occur due to transition probabilities assigned to links. A sophisticated adversary, that has knowledge of the geographical area and the mobility model of the user, can compute more accurate values of the conditional probability $\mathbf{Pr}(a(t) = r | a(t-1) = \rho)$ instead of simply assuming that they are equiprobable. For example, if a location r is not accessible by the user (even if ρ and r are direct neighbors), the probability $\mathbf{Pr}(a(t) = r | a(t-1) = \rho) = 0$, $\forall \rho \in \mathcal{L}_{t-1}$, resulting in a different value of $\mathbf{Pr}(a(t) = r)$. This is the second design specification taken into account in our heuristic algorithm.

3.3 Mobility-aware Obfuscation Algorithm

In this section, we describe our heuristic obfuscation algorithm that is mobility-aware. The algorithm takes as input the user actual location $a(t)$, her current velocity v_t and the obfuscation parameter c, and returns the set $\mathcal{L}_t$ of c locations, representing the obfuscated location of the user. The algorithm checks the reachable locations from the last time instant w.r.t. v_t and

by using the transition probabilities from the user mobility model, determines the $c-1$ fake locations to be included in $\mathcal{L}_t$ along with the actual location. The principle idea is that the algorithm chooses locations to populate $\mathcal{L}_t$ such that they will be the most probable locations along with $a(t)$ to go to from the locations in $\mathcal{L}_{t-1}$. Hence, the adversary's confusion will potentially be the maximum possible.

There are two base cases for our algorithm, namely for $t = 0$ and $t = 1$. For the sake of presentation and ease of understanding, we first explain the body of the algorithm for $t > 1$ and explain the base cases later. In summary, the algorithm consists of 3 steps. First, it determines the reachable locations at time t from the locations at time $t-1$ w.r.t. the velocity v_t. We call these locations the *candidate* locations $\mathcal{S}$ under consideration for $\mathcal{L}_t$. Secondly, the algorithm chooses $c-1$ locations from $\mathcal{S}$ such that they will have the highest transition probabilities w.r.t. $\mathcal{L}_{t-1}$. Finally the set $\mathcal{L}_t$ is formed by $a(t)$ and the determined $c-1$ locations and returned. The algorithm's pseudo-code is presented in Algorithm 1 with conditions $t = 0$ and $t = 1$ not included for readability purposes. Note that the algorithm has access to all past reported locations, *i.e.,* $\mathcal{L}_k$ for $k < t$, and user history.

Algorithm 1: Mobility-aware Obfuscation Algorithm

Input: $a(t)$, c, v_t
Output: $\mathcal{L}_t$ – the set of c locations acting as the obfuscation area

1 $\mathcal{L}_t = \{a(t)\}$;
2 **for** *each* $\rho \in \mathcal{L}_{t-1}$ **do**
3 Find locations that can be reached from ρ, *i.e.,* **neigh**(ρ, v_t);
4 $\mathcal{I} = \bigcap_\rho$ **neigh**(ρ, v_t) ;
5 $\mathcal{S} = (\mathcal{I} \setminus a(t)) \bigcap$ **neigh**$(a(t), v_t)$;
6 **for** *each* $r \in \mathcal{S}$ *and each* $\rho \in \mathcal{L}_{t-1}$ **do**
7 Compute the probability $\mathbf{Pr}_{r\rho}(a(t) = r | a(t-1) = \rho)$;
8 $i = 0$;
9 **while** $i < c - 1$ **do**
10 $l_i = \arg\max_r \left(\sum_{\rho \in \mathcal{L}_{t-1}} \mathbf{Pr}_{r\rho}\right)$;
11 $\mathcal{L}_t = \mathcal{L}_t \bigcup \{l_i\}$;
12 $i = i + 1$;
13 return $\mathcal{L}_t$

At the beginning of the algorithm, $\mathcal{L}_t$ is initialized to $\{a(t)\}$ as it has to include the actual location of the users. Afterwards, in the first step (lines 2-5), the set $\mathcal{S}$ of candidate locations is determined by finding the set of neighbors of each location j in $\mathcal{L}_{t-1}$ with the limited range v_t, the velocity of the user. Then it finds the intersection of these sets to increase number of combinations of paths in the linkability graph to ensure maximum confusion for the adversary. The set $\mathcal{S}$ of candidate locations is finalized by first removing the actual location $a(t)$ from it and then intersecting it with the neighboring locations of $a(t)$. This last part filters the candidate locations to those within the proximity of the actual user location and also are accessible by all the locations in $\mathcal{L}_{t-1}$.

After determining the set $\mathcal{S}$, the transition probabilities from the locations reported previously, *i.e.,* in $\mathcal{L}_{t-1}$, to the ones in $\mathcal{S}$ are computed in the second step (lines 6-7). In the last step, $c-1$ locations are chosen from $\mathcal{S}$ that provide the highest transition, hence presence, probabilities for time t and inserted to $\mathcal{L}_t$. Note that, in our experiments, we actually implement this step in a way that $\mathcal{L}_t$ consists of locations that form a joint polygon, in other words to avoid disjoint areas in $\mathcal{L}_t$. However, $\mathcal{L}_t$ may form areas of not regular shape (*i.e.,* square, rectangle, *etc.*).

We now explain the details for the base cases of our algorithm, *i.e.,* for $t = 0$ and $t = 1$. For $t = 0$, we do not have any transition to compute due to nonexistence of $\mathcal{L}_{t-1}$. Instead, the algorithm computes the presence probabilities of each location that is a neighbor of $a(0)$, replacing the lines 2-7. These probabilities can easily be computed from the history of user. Step 3 (lines 8-13) remains same except that instead of $\mathbf{Pr}_{r\rho}$ on line 10, the presence probabilities $\mathbf{Pr}(a(0) = k)$ are used. Secondly, for $t = 1$, we do have transitions, however, we cannot check a direction change because we need at least 3 points in space to be able to determine an angle of movement. Hence, we compute the transition probabilities $\mathbf{Pr}_{r\rho}$ based only on the history mobility model explained in Section 3.1.2 on line 7 of Algorithm 1.

Example: We now go briefly through the example presented in Section 3.2 again (where the trace of the user is {2, 6, 7} w.r.t. the area in Figure 3.1) in order to demonstrate the difference between the random selection of locations to report and our heuristic algorithm. We assume a maximum speed of 1 location per time instant with $c = 2$. At $t = 0$ with $a(0) = 2$, the set $\mathcal{S}$ is populated with all the one-hop distance neighbors of $a(0)$, *i.e.,* $\mathcal{S}$ = {1,2,3,5,6,7}. Here we have no access to past traces, so we just pick one location from the set $\mathcal{S}$ to report. Assume that we randomly choose to report location 1 as the fake location. Therefore $\mathcal{L}_0$ = {1,2}. For $t = 1$, we determine the candidate set $\mathcal{S}$ = {1,2,3,5,6,7} based on the previous locations reported. Again, we pick one out of these locations with uniform probability and report $\mathcal{L}_1$ = {5,6}. The algorithm's role becomes apparent now at $t = 2$. For every previously reported location, *i.e.,* in $\mathcal{L}_1$ = {5,6}, we find their neighbors, and intersect the two sets of neighbors to get the set $\mathcal{I}$. We have $\mathcal{I} = \{1,2,5,6,9,10\} \bigcap \{1,2,3,5,6,7,9,10,11\} = \{1,2,5,6,9,10\}$. We then prepare $\mathcal{S}$ as described previously (*i.e.,* according to line 5 of the algorithm): $\mathcal{S}$ = {2,6,10}. Now that we have too many candidate locations to report, we have to compute the probability for all the elements of $\mathcal{S}$. But in this example we cannot compute the probability distribution so we just select a cell that could be reached from the previously reported locations, say cell 6. We finally report $\mathcal{L}_2$ = {6,7}. We can easily see that if we draw the linkability graph for this example we will get a fully connected graph (unlike with the random selection of locations), *i.e.,* we have no transition probability that is zero. Obviously this is only a small example and not all the parameters are taken into account but it gives a good intuition of how the algorithm works and how it differs from a random obfuscation model.

3.4 Evaluation

We evaluate our heuristic algorithm by comparing it to a random obfuscation mechanism and using a dataset of real traces. We evaluate location privacy of users for both mechanisms

using the Location-Privacy Meter developed by Shokri *et al.* [99, 100]. In the remainder of this section, we describe our dataset and methodology, explain the privacy metric we use and present our experimental results.

3.4.1 Dataset and Methodology

We use the real-world traces from the data collection campaign carried out by Nokia in Lausanne region [85] from 2009 to 2011 (*i.e.,* the same dataset as in Chapter 2. The area-of-interest we consider in Lausanne region is of size 1.25 × 1.0 km, which is discretized to 25 × 20 regions for computational limitations. We filter the dataset w.r.t. to this area and choose the users that have at least 15 chunks of 40-event long traces in this area. We train the adversary with the additional traces of each user while obfuscating and attacking one of them. This results in a final set of 33 users. We run our experiments for varying α and c values with heuristic and random obfuscation mechanisms separately. Finally, we attack all the generated obfuscated traces with two attackers (*i.e.,* using the Location-Privacy Meter), one with and one without the background knowledge on users' history.

3.4.2 Measuring Location Privacy

In our scenario, as in the case of Chapter 2, the adversary has access to the obfuscated trace of a user. Her objective is to reconstruct the user's real trace based on this observation. The more accurately she succeeds in the reconstruction, the lower the level of location privacy that we obtain in our system. An effective metric for measuring location privacy is the *distortion-based* metric by Shokri *et al.* in [98].

In order to evaluate the effectiveness of our heuristic obfuscation algorithm, the Location-Privacy Meter (LPM) [99, 100] was used and the expected error of the adversary is employed with Euclidean distance. The observed traces serve as input to the tool as well as a distance Function, and a transition matrix, a matrix that describes which transitions between regions are possible over consecutive time steps, given the geographical area and the maximum speed of the user (in regions per time unit). The output is the level of location privacy in terms of expected distortion in meters at each time instant.

3.4.3 Experimental Results

Here we show the different results we obtained for the different tests we ran. We ran our experiments with varying c and α values for our heuristic algorithm. We ran tests on the obfuscated traces with two different attackers (with and without background knowledge of the users' mobility). All location privacy levels are presented as the expected error of the adversary's inference in meters. We refer to the attacker with background knowledge as the "strong attacker", and the one without background knowledge as the "weak attacker".

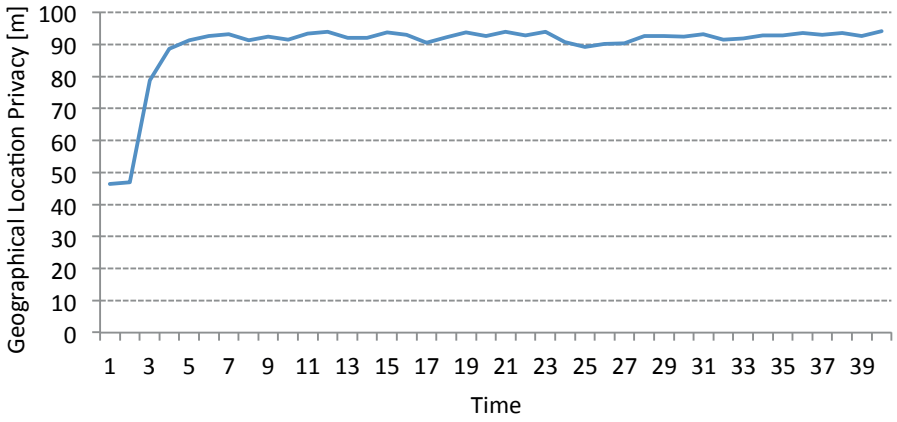


Figure 3.4 – Privacy protection over time for one user, $c = 5$, $\alpha = 0.5$, against the weak adversary.

In Figure 3.4 we show the average location-privacy level of a user over time in meters for $c = 5$ and $\alpha = 0.5$ and against the weak attacker. We observe that the location privacy is relatively constant over time, except the first two time instants. This is due to the fact that we used uniform probabilities when choosing the fake locations for obfuscation in these time instants (*i.e.,* the base cases of the algorithm). But as of the third time instant, our actual heuristic algorithm shows its effect on the location privacy and we observe a jump around 100%. Also, as the algorithm can retain linkability among successive time instants, we see consistent privacy level protection over time as in Chapter 2.

We now show the impact of the parameters c and α on obfuscation. On Figure 3.5, we plot the average privacy levels over all users and all time instants over c obtained against the weak

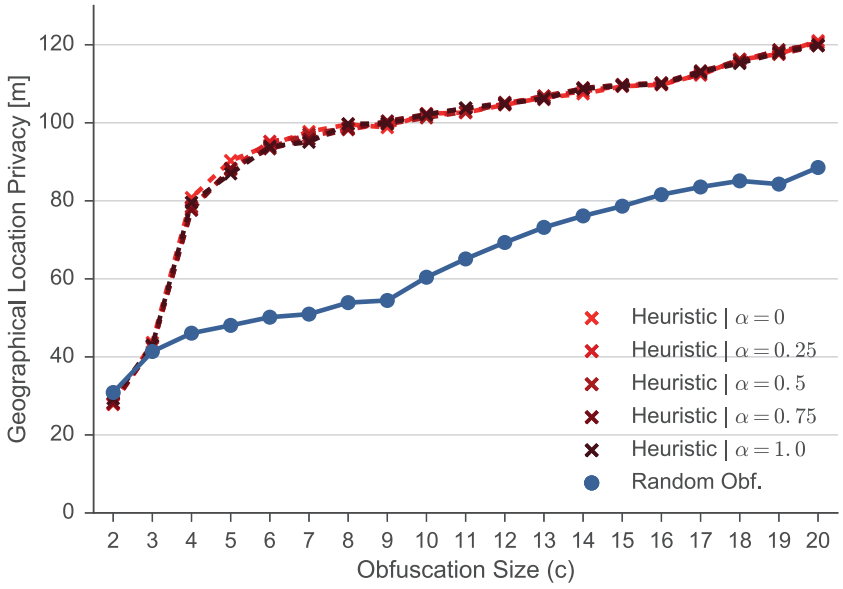


Figure 3.5 – Location privacy over the obfuscation parameter c averaged over all users and time instants, against the weak attacker.

attacker. The results include privacy levels obtained with heuristic algorithm with $\alpha = 0$, with $\alpha = 1$ and a random obfuscation algorithm that generates a randomly placed obfuscation area. We observe that the value of α does not significantly change the results against the weak attacker. When we analyzed the results with the strong attacker, we observed the same trend.

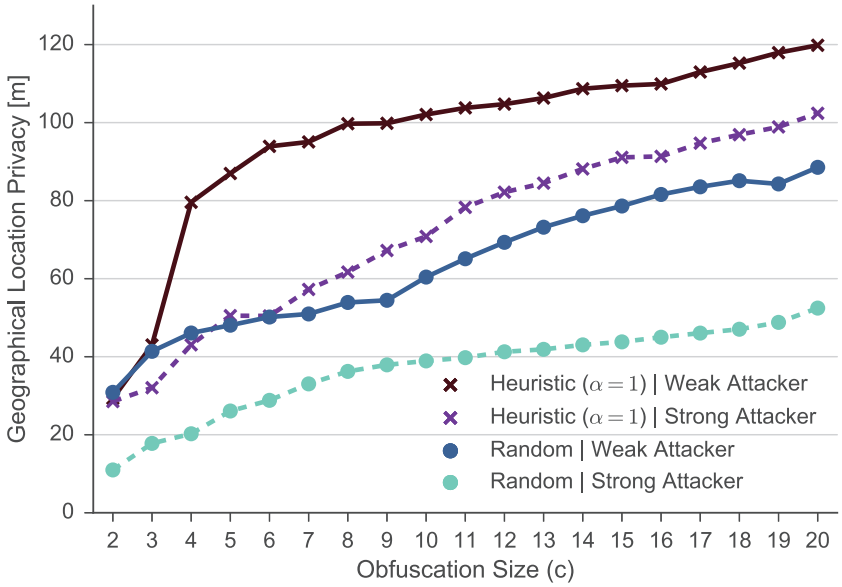


Figure 3.6 – Location privacy over the obfuscation parameter c averaged over all users and time instants for heuristic and random mechanisms against both attackers.

Finally, we compare the random obfuscation and our heuristic algorithm against the two attackers. Figure 3.6 shows the results of the algorithms for different values of obfuscation parameter c, and a comparison w.r.t. two adversary models (*i.e.*, the weak and the strong attackers). For the heuristic algorithm, we averaged the privacy levels with different values for α, (*i.e.*, for $\alpha = 0$, 0.25, 0.5, 0.75 and 1) because the differences were negligible. The first observation on this plot is that the heuristic algorithm clearly outperforms the random one for $c \geq 4$. Even when the heuristic algorithm is attacked by the strong attacker, it still provides a better location-privacy level than the random obfuscation mechanism against the weak attacker. This result demonstrates the importance of considering the mobility of the user when applying protecting location privacy. Obviously, the strong attacker obtains more information than the weak one for both heuristic and random mechanisms. However, the relative loss in the case of heuristic algorithm is considerably less than the loss in the case of random obfuscation. As expected, the location privacy is higher for larger values of c in all cases.

3.5 Related Work

The research community is aware of the privacy risks arising from mobility of users in continuous disclosure scenarios. There have been some attempts [50, 113] to address this issue by proposing velocity or mobility aware protection mechanisms. However, they fail to meet

certain requirements which we addressed in this chapter. Xu *et al.* [113] proposes to analytically consider a user's transition probability distributions among locations in order to derive certain cloaking areas. They achieve this by constructing a linear program and solve it for the optimal solution for building a cloaking area. Their idea is similar to our heuristic approach, yet they do not consider direction-based mobility in their system. Furthermore, they do not evaluate their protection mechanism against a powerful adversary with real traces. Instead, they attack either isolated user events or a short sequence of user events (*i.e.,* 2-3 events) and evaluate the privacy levels with Entropy metric.

Ghinita *et al.* [50] proposes a protection mechanism to protect location privacy in a velocity-aware way. Their model lacks user mobility history in protection mechanism. They also do not evaluate their mechanism's effectiveness against a localization attack with strong adversary assumptions. They do, however, consider sensitive semantic places on the map in order to determine the size and placement of a cloaking area.

Finally, Götz *et al.* [54] propose a mobility-aware protection mechanism based on Hidden Markov models that take into account the knowledge of the adversary. Their proposed mechanism provide optimal solutions for keeping the confusion of the adversary high; however, the system requires an expensive initialization phase, thus requiring offloading some work to a remote server.

3.6 Summary and Discussion

In this chapter, we explored how we can provide a strong obfuscation-based protection mechanism that is mobility-aware. We formulated the choice of obfuscated locations at time instant t, such that the deterioration in the privacy level at time $t+1$ will be minimized while the privacy levels in backward are retained. We proposed a heuristic algorithm that takes into account the mobility of the user (*i.e.,* past behavior and the direction of movement) when applying obfuscation. The effectiveness of the heuristic algorithm was evaluated experimentally and the results are remarkable. This work supports the idea that protection mechanisms must be designed to take into account the user behavior and how the adversary might use such a knowledge against privacy.

As a limitation, although trying to be more realistic in the mobile setup has proved to provide better experimental results, this is not a formal proof that they constitute necessary and sufficient conditions. Working towards identifying these conditions and providing a formal proof is an interesting research direction. Furthermore, the heuristic algorithm can be extended to consider variable location-obfuscation parameter c over time. In fact, it can be merged with the adaptive protection approach in Chapter 2 and implemented on smartphones, e.g., the Location-Privacy library (Chapter 4).

4 Location-Privacy Library on Android Platform

We present in this chapter an Android library we developed to provide location-privacy estimation and protection to mobile users. We implement in it our adaptive privacy-protection mechanism presented in Chapter 2. We separate the privacy estimation algorithm we developed and make it a separate module, thus enabling independent privacy measurement for any additional privacy-protection mechanism or feedback to the user. Furthermore, as a user may require using a different location-privacy protection strategy depending on the type of place where she is, *i.e.,* the *location semantics*, we build a privacy sensitivities setting interface for users. For instance, she may not want to have her location revealed when she is in a hospital. The sensitivities can be set for a specific type of place, *i.e.,* semantic tag, or a geographical region. Our adaptive algorithm takes into account the privacy sensitivity corresponding to the current location of the user and utilizes the privacy estimation module to protect the user's actual location.

Most of the existing approaches to location-privacy protection have not explored these scenarios, and instead provide a static mechanism, *i.e.,* they set a fixed protection parameter such as the size of the obfuscation area. Moreover, as we will see later, very few previous approaches consider the applicability and effectiveness of their methods, let alone implement a real protection program for mobile applications. We try to fill this gap with our Location-Privacy Library that we made available as an open-source project [4]. Location-aware applications can provide location-privacy protection to their users by integrating this library.

We have integrated this library with the generic tinyGSN Android application that integrates heterogeneous sensing devices running on the smartphone, and makes them available as virtual sensors that can interact through the GSN (Global Sensor Networks) middleware [10]. To the best of our knowledge, this is the first Android library/application that adaptively protects location privacy in a way aware of privacy sensitivities and location semantics. Finally, we evaluated this library in terms of performance, resource usage, and expected privacy in a real-life scenario based on air-pollution sensing for personalized health monitoring [5].

4.1 Adaptive Privacy-Protection

In this section, we summarize the adaptive privacy-protection system we use to build our library. As mentioned earlier, this system is based on the work in Chapter 2. As we know, the adversary has a certain knowledge about each user u, more specifically the observed history and the obfuscation mechanism being employed. If the protection mechanism employed by the user uses this information, then it provides privacy-protection in an adaptive manner. In a nutshell, our protection mechanism reasons about the information the adversary has, anticipates on what he can infer from the disclosed data, and decides on the protection strategy accordingly. It also regards the semantics of the user's location and the privacy sensitivities, which it integrates in its decision process. To achieve this, our location-privacy library performs a simulated attack on the user's obfuscated trace and performs an evaluation of the expected privacy level. Then our protection mechanism uses this information to change the protection level if necessary. However, as applications on the user side do not have as much computational power as the adversary has, this needs to be modeled in an efficient way. Thus, the simulated attack in our library is weak, but sufficient to approximate what the adversary can achieve.

We introduce two separate core building blocks in our adaptive privacy-protection scheme: the first is in charge of the local estimation of location-privacy, while the second essentially applies the protection techniques on user locations. Figure 4.1 shows the interactions between these blocks and their dependencies with the user history and sensitivity profile.

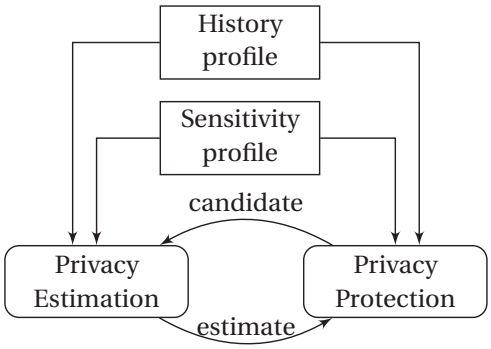


Figure 4.1 – The adaptive protection approach is based on two main building blocks: local privacy-estimation privacy protection. They interact with each other for adaptation in real-time by using user history and the sensitivity profile.

The privacy estimation keeps track of the user's past events (where the user was, and when) and the history profile. Then, it uses Bayesian inference to attack user's own trace as described in Chapter 2. This is achieved by storing user obfuscated events in an inference graph, and updating the graph in real-time as the user generates new events. Privacy estimation fuses information from the Bayesian inference results with history data and computes an estimated privacy level using the expected distortion metric proposed by Shokri *et al.* [98]. This metric is basically an expected value computation on distances between the user's actual location and

the observed locations in his obfuscated event. Note that what the user estimates here is in fact the posterior distribution h resulting from the adversary attack. However, the adversary attacks the obfuscated trace as a whole; the user only attacks the disclosed part of her obfuscated trace. As a result, the user achieves an approximation of h.

Whenever the user generates a new event, the protection component obtains the actual location and the corresponding semantic tag; then it checks the user's sensitivity profile δ_u and drafts an obfuscated location by also considering the user's history profile H_u. It invokes the privacy estimation by passing it the generated obfuscated location and then receives the expected privacy-level as if the user would disclose the current obfuscated location. Upon receiving this estimation of the privacy-level, the protection component checks if it satisfies the user's sensitivities. If it matches the user sensitivity, it discloses the obfuscated location. Otherwise, it adjusts its parameters and generates a new obfuscated location, and goes through the same procedure again. In summary, the protection mechanism iteratively adjusts its obfuscation parameters until the user's sensitivity preferences are satisfied.

4.2 Location-Privacy Protection Library

In order to demonstrate the feasibility and usefulness of our adaptive protection model, we have implemented it as a library for Android [1] version 5.0 (*i.e.,* Lollipop) and above. The library is designed in a way that any Android application that aims to protect its users' location privacy can integrate the library easily and without manually initializing its database (with configuration information such as a semantically-annotated map and a sensitivity profile). We integrated our library to tinyGSN [8, 43], a sensor data collection application for Android devices, in order to test its applicability in real-world scenarios. In this section, we elaborate on the library architecture, the implementation details of its features and how it is integrated to tinyGSN.

4.2.1 Architecture

In this section, we explain in more detail how the theoretical system introduced in section 4.1 is turned to a concrete architecture. Our library consists of four modules that interact with each other and also with the main application hosting the library. This architecture is shown in Figure 4.2.

The interaction rationale among these modules is as follows: The location-privacy protection mechanism, that implements obfuscation for protecting the location, requires a privacy-level estimation whenever it creates an obfuscation area for a given precise location. The privacy estimation module (PEM) provides this functionality and after evaluating any given obfuscation area, returns a privacy-level estimation. Upon this, the protection mechanism decides whether this estimation meets the privacy requirements or not. At this point, the protection mechanism requires the privacy sensitivities of the user based on the current

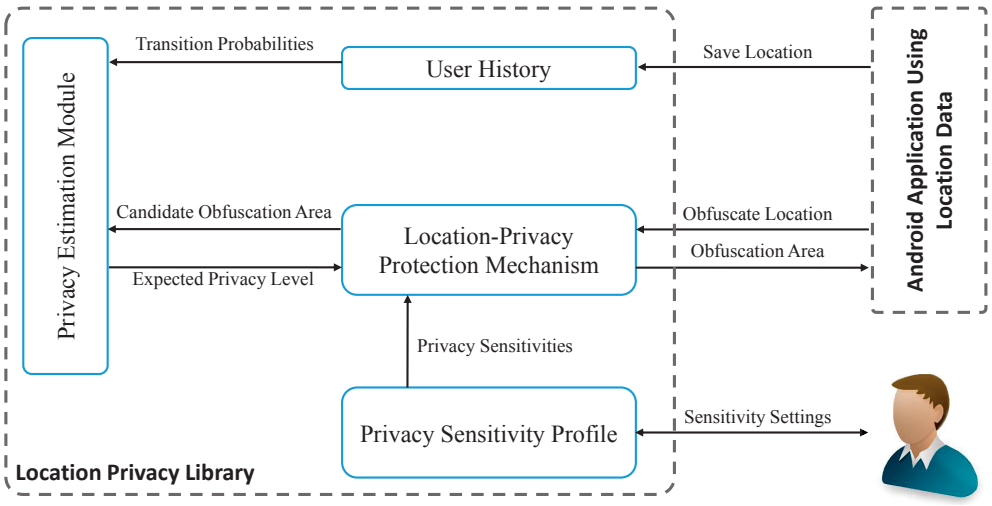


Figure 4.2 – Architecture of the Location-privacy protection library for Android.

location to be able to make the above-mentioned decision. Hence, we introduce the privacy sensitivity profile that stores and provides this information. As the privacy sensitivities are personal, we provide interfaces for the user to modify them as she sees fit, be it semantic or geographical. Coming back to the PEM, the evaluation of a given obfuscation area requires keeping track of previously disclosed locations (*i.e.*, obfuscation areas) and also knowledge about the user's movement patterns. The PEM stores the recent trace of the user locally, but for the movement patterns, it relies on the User History module. Movement patterns are designed to be in the form of transition probabilities, denoting the probability of going to a region r given that the user's current location is ρ. This information is provided by the User History Module, that keeps track of which locations the user visited, at what time, and which location he visited afterwards. Basically, this module gets its location feed of the user from the main Android application that uses the library, but it can potentially connect directly to the Android OS's localization services and save user history frequently.

4.2.2 Implementation

The geographical information in our model is discretized for increasing time and space efficiency of the protection mechanism. In terms of implementation, all the modules in the library use location information in (and converts the geographical coordinates to) a discretized representation. This requires a mapping from coordinates system to a set of regions. For this, we use the Cantor pairing function [25, 74] which is a bijective function $\pi : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ as follows:

$$\pi(x_1, x_2) = \frac{(x_1 + x_2) \cdot (x_1 + x_2 + 1)}{2} + x_2 \tag{4.1}$$

This pairing function allows us to convert geographical coordinates to natural numbers, hence the region ids. For any given geographic coordinates, we first determine the region this point is in based on what size the regions are defined. Due to computational limitations for the PEM,

we use an approximate size of 100×100 m for regions. By using the Haversine formula[1] [92], we determine the top left corner of the region the point is within. After that, we convert the latitude and longitude values of this region's top left corner (that are real numbers) to integers by multiplying them by 10,000 which gives around 11 meters of precision in the geographic coordinate system. The precision changes as we move towards the poles of the Earth; however, this does not affect our application as we do not impose perfect alignment of the regions on the map. Consequently, our library does not require a pre-loaded database of regions, it can download and cache current areas automatically.

Sensitivity Profile

Our adaptive protection mechanism aims to satisfy a desired level of privacy given any location, based on either the location itself or its semantics. To fulfill this objective, it requires a threshold to compare the estimated privacy level obtained from the PEM in order to make a decision regarding the size of the obfuscation area. Our approach in this regard is to define sensitivity values for semantic tags and geographic regions, that act as the threshold. In this sense, we determine a list of semantic tags obtained from OpenStreetMap [6], which is an open and free map project. Moreover, we let users set sensitivity levels for geographical locations in the form of regions as introduced earlier in this section. Whenever a user sets a sensitivity level for a region, this overrides any semantic sensitivity level that might concern this region, for instance, if the sensitivity level in the library for a bar is low and a specific region that contains a bar is set to have high sensitivity level for the user, then the library considers the location to have high sensitivity whenever the user is there.

We provide the user with a configuration panel in the Android App, to customize the sensitivity levels for semantic tags and locations easily through a graphical interface. Figure 4.3 shows the screenshots of this panel. The demonstrated 'min' and 'max' sensitivity levels are represented by 0.0 and 1.0, respectively, and saved to the local SQLite database for each semantic tag and also for the selected geographical locations. For semantic tags, a slider is used for setting the sensitivity level for each semantic tag, whereas for locations, we benefit from Google Maps widget to visually let the user determine and pick the location for which she wants to set a sensitivity level. After picking the desired location, she can simply check the override semantic sensitivity box to determine the location-specific privacy sensitivity level.

User history

In order for the PEM to take into account the most visited places by the user in the inference attack, we keep track of the user's mobility history. The reason behind this is that the adversary could already have obtained information on the user's mobility patterns through privacy-intrusive channels (for example, through an application the user installed on her device that

[1] Haversine formula is used to compute the great-arc distance between two points on a sphere using their latitudes and longitudes.

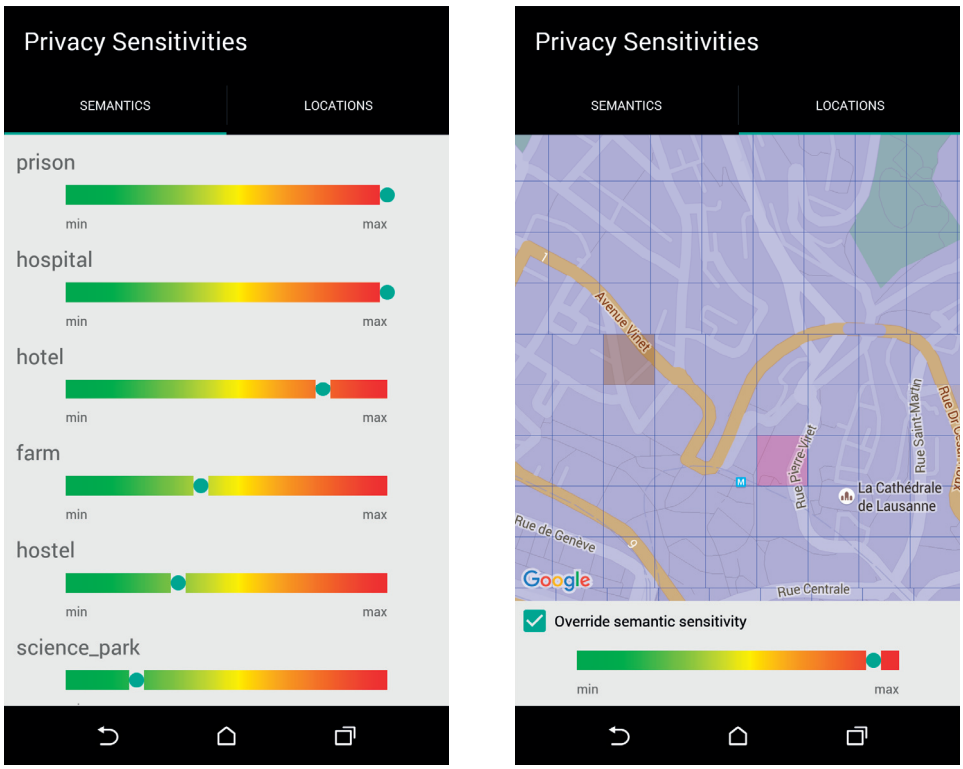


Figure 4.3 – Screenshots of the privacy-sensitivities settings panel. On the left, the user sets the sensitivity levels for various semantic tags. On the right, the user can choose and highlight the location she wants to set the sensitivity level and override the semantic sensitivity levels. The brown region is the currently selected location and red regions are the ones for which specific sensitivity levels are set.

does not have a location-privacy protection function). We keep this information in an SQLite database locally and in the form of transitions. In practice, people have routines and hence their mobility is quite repetitive [53]. Thus, to reflect this, the history should be recorded frequently (for example, once every 10-15 minutes). In our library, we implement a service that can be configured to save the location of the user periodically as well as provide an interface for the main application to take care of invoking the location recording if necessary (and preferred). Each time the location is saved, the current location and time of day are inserted to the database as well as the previous location and time of day. This lets us aggregating the time-dependent movement patterns of the user as transition probabilities. Basically, the PEM requests the transition probability between two locations based on the obfuscation areas generated.

Transition probabilities are calculated simply from the number of transitions observed from one location to another which is then divided by the total number of transitions observed from the same location. As it was shown that human mobility is quite time-dependent [53], we narrow down this to the number of transitions observed within the user's current time period in the day, for instance, taking into account the transitions observed between 12-13 o'clock

in the past. This information is provided to the PEM by getting the count information from the SQLite database and then returning the outcome as a probability. The following formula expresses this calculation:

$$\mathbf{Pr}(r_2|r_1) = \frac{\#(r_1 \rightarrow r_2)}{\sum_{\rho} \#(r_1 \rightarrow \rho)} \tag{4.2}$$

where $\#(r_1 \rightarrow r_2)$ is the number of transitions observed from region r_1 to region r_2 in the user history which is stored in the local SQLite database. $\sum_{\rho} \#(r_1 \rightarrow \rho)$ is the number of all transitions observed from region r_1 to any other region.

Privacy Estimation Module

The PEM is computationally the most complex and hence demanding part of our library. It houses a basic Bayesian inference scheme that determines the regions in an obfuscation area that will be believed to be the actual location of the user from the adversary's point of view. This is achieved by keeping track of the user events, *i.e.,* the obfuscation areas disclosed with the time information and also the actual user events in a *linkability graph* such as the example in Figure 4.4. Basically, each node in this graph represents a region in the obfuscation area disclosed at a given time instant t_i. Whenever a new obfuscation area is generated, it is appended to the graph and the regions in the previous time instant are connected to the new event based on physical reachability between them (based on the maximum speed allowed to go from one place to another, which is set to 70 km/h in our library – the typical speed limit in cities is 50 km/h). Links between the regions are assigned the corresponding transition probabilities obtained from the user history module. This helps us to compute the presence probability per node, *i.e.,* region. In Figure 4.4, the transition probabilities from all the nodes are uniform, meaning that in this example history is not used.

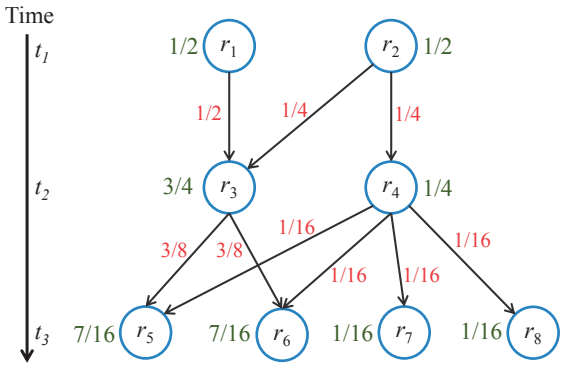


Figure 4.4 – An example linkability graph that shows the transition probabilities between locations on the edges and the presence probabilities derived from them and then assigned to the locations shown next to the nodes.

As more user events are generated (*i.e.,* visited locations), the number of nodes and levels in the linkability graph will explode. To avoid this, we remove old events from the graph and keep

only the most recent 30 events. In this case, the latest probability values assigned to the nodes in the first-level event in the graph are retained. This ensures the correct update of probability propagation in the graph in the event of a link removal.

Finally, the estimation of privacy level is done as follows. As we derive the presence probability for each region in the latest obfuscation area, we calculate the distance between them and the actual region of the user. We use the GPS coordinates of the center points of the regions for this. Then, we calculate the expected distance to the original location by multiplying the distances with their corresponding presence probabilities. This gives an approximation of the adversary's error in guessing the actual location of the user [98]. The formula for this calculation is as follows:

$$LP = \sum_{r \in \mathcal{R}} \mathbf{Pr}(a_u(t) = r) \cdot \mathrm{dist}(r, a_u(t)) \tag{4.3}$$

where $\mathrm{dist}(\cdot,\cdot)$ is the Euclidean distance between two given regions and $a_u(t)$ is the actual location of user u at time t.

Adaptive Protection Mechanism Module

This module consists of the iterative algorithm presented in Chapter 2, which generates an obfuscation area and checks if it satisfies the privacy requirements based on the current location of the user. For this, it starts with an unprotected location first, *i.e.,* no obfuscation area. Note that this still needs to be inserted to the user's trace in the PEM, hence it is submitted to the PEM and an estimation of privacy level is obtained, which is 0.

At this point, the sensitivity level of the current location or its semantics needs to be checked. For this, the module first checks with the sensitivity profile, whether there is a sensitivity level set for the current location specifically. If yes, this sensitivity level is used. Otherwise, the semantics need to be checked. As we cannot load the semantic map information for the whole world, this semantic data needs to be fetched online as needed. OpenStreetMap provides an API, called OverpassAPI, for querying map by providing a bounding box and also a list of semantic tags. The API returns a list of polygons, ways and nodes, that are annotated by one of the semantic tags included in the query. We use this API to automatically obtain semantic details of the user's current vicinity (or her destination). The information is saved to a local Spatialite[2] [7] database that supports geometrical queries such as intersection and minimum bounding box. This functionality eases the determination of the dominant semantic tag in a given region. In this way, we obtain the semantic tag for the current location and query the sensitivity profile accordingly. Note that, fetching data from OpenStreetMap can be perceived as privacy-invasive as well. We avoid this by fetching data for large areas (e.g., 2×2 km) and save it for a long period of time (*i.e.,* cache it). This ensures that no tracking occurs by OpenStreetMap.

[2] Spatialite is a specialized version of SQLite database system that supports geometric objects as column types, such as polygon and point.

Once the sensitivity level is obtained for the current location, the mechanism needs to compare this to the privacy estimation. Note that the estimation is actually a distance value, whereas the sensitivity levels are scalar in the interval $[0,1]$. We use a coefficient Θ at this point that represents the maximum value for the desired level of privacy. We multiply the sensitivity value with Θ to obtain the final threshold for the desired level of privacy. In our library, we have set $\Theta = 0.2$ which is 200 meters (this value can also be set by users). If the estimated level of privacy is less than the desired level of privacy threshold, then the obfuscation mechanism increases the obfuscation area size and generates a new obfuscation area. This obfuscation area is randomly positioned over the current location of the user. A new privacy estimation is requested from the PEM and then compared to the privacy threshold. If it is not met again, the mechanism tries another obfuscation area with the same size again, but positions differently on the current location. Whenever the desired level of privacy is reached with an obfuscation area, then this area is returned to the requesting application and it is retained in the PEM. On the side note, the protection mechanism tries two different obfuscation areas of the same size before incrementing the area size, because a different positioning of an obfuscation area on the map may provide different privacy levels.

TinyGSN Integration

For testing and prototyping, we integrated our library to tinyGSN application [8, 43], that is developed for sensing environmental data on the Android platform. As sensing data is only meaningful with time and location stamps, this application allows setting sensors with location recording (obtained through GPS module). The application allows a user to set wrappers for real sensors to collect data and create virtual sensors to fuse data from any source (sensor wrappers and/or virtual sensors) before recording the final outcome. This flexibility enables collection of multidimensional data together and process them as required. We have modified the virtual sensor implementation of this application to let users enable location-privacy protection in case this virtual sensor uses GPS data. Figure 4.5 shows this modification.

4.3 Evaluation

We evaluate our library on functional and performance aspects; namely, the correctness and adaptiveness of the protection algorithm, its effects on memory, CPU usage and battery (through power consumption) as compared to the case where the library is not employed by tinyGSN [8, 43].

4.3.1 Location Privacy

To evaluate the location privacy, we have used a dataset that consists of GPS traces of users that participate on a health-monitoring data collection campaign (approved by the local ethics committee). The campaign aims to collect environmental data (*i.e.,* air quality, exposure to

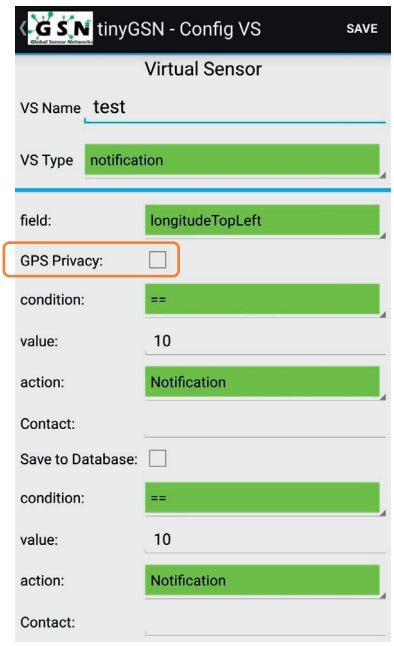


Figure 4.5 – tinyGSN with the Location Privacy Library integrated. “GPS Privacy” checkbox lets the user to enable adaptive location-privacy protection.

pollutant gases) for individuals by asking participants to carry Android smartphones running the tinyGSN application [43] and various air quality sensors. The participants carry the provided devices for one day. The campaign is coordinated by the CHUV, the cantonal university hospital in Lausanne, Switzerland, in the context of OpenSense Project [5]. We obtained GPS traces of 15 participants, length of which vary from 79 to 678 location observations. We loaded these traces to the tinyGSN application integrated with our library and replayed a sensing scenario using these traces. For this experimental evaluation, we set sensitivity levels for a set of semantic tags in the application to reflect the outcomes in different scenarios. Table 4.1 lists these semantic tags and the sensitivity levels assigned to them. All the other semantic tags are assigned a sensitivity level of 0. Note that these sensitivity levels may be different among individuals, but preloading a common sensitivity profile to the library would help users to start using the library immediately. However, this requires a social research that complements the work in this chapter.

Table 4.1 – Privacy sensitivities used in the experiments.

Tag	Sens.	Tag	Sens.	Tag	Sens.	Tag	Sens.
Clinic	1.0	Hospital	1.0	Prison	1.0	Embassy	0.9
Bank	0.8	Nightclub	0.8	Commercial	0.75	Aerodrome	0.7
Police	0.7	University	0.5	Hotel	0.5	Bar	0.4
Fast Food	0.3	Restaurant	0.3	Cemetery	0.3	Zoo	0.3
Stadium	0.25	Train Station	0.15	Post Office	0.10	Museum	0.1

In Figure 4.6, we present the privacy levels estimated by the PEM of the Location Privacy

Library based on the privacy sensitivity settings on the device. Specifically, this graph plots the estimated level of geographical location-privacy vs. the desired level of privacy for each user event for all the users in our dataset. Evidently, the Location Privacy Library manages to satisfy the desired level of privacy in all cases as there is no data point under the threshold line shown as a dashed line. Of course, this graph reflects the local estimation of the privacy level, meaning that a stronger attack might breach some level of privacy even in the case of an adaptive protection approach that considers a certain attack scenario. Nevertheless, it is experimentally shown in [11] that such an attack manages to obtain only very little and hence an adaptive protection approach remains strong.

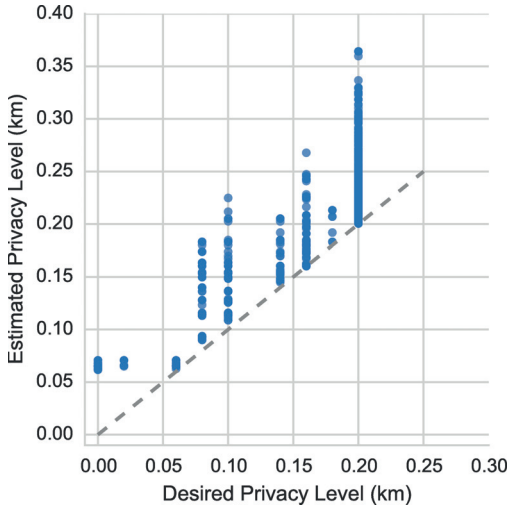


Figure 4.6 – The estimated privacy level of users based on their obfuscated traces as compared to their desired level of privacy measured as an expected error of the adversary in km.

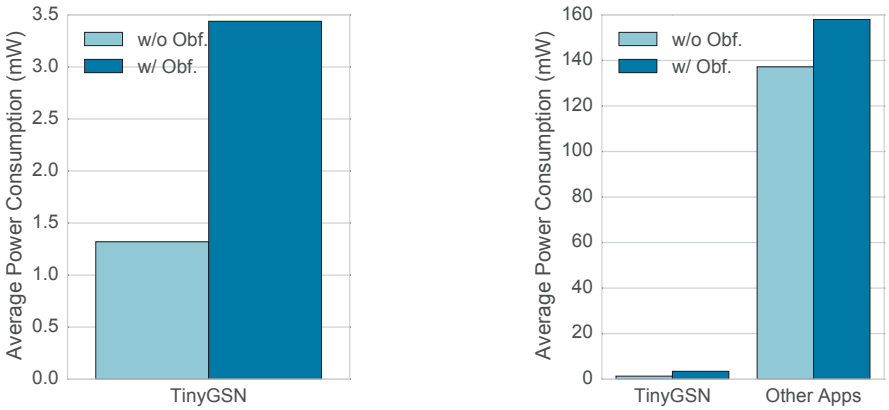
4.3.2 Performance

For performance evaluation, we ran experiments on a Samsung Galaxy S4 smart phone. We evaluate our library in terms of CPU usage time, power consumption and memory usage. We ran tests with four different scenarios:

1. with obfuscation and without active usage of the device by the user
2. with obfuscation and with active usage of the device
3. without obfuscation and without active usage of the device
4. without obfuscation and with active usage of the device

For the power consumption, we benefit from PowerTutor developed by Zhang *et al.* [120] that estimates the power consumption by different modules of an Android device as well as the applications through the CPU. By using this application on our device, we have sampled around 4000 measurements (over 4000 seconds) for each of the four scenarios. We have

plotted the averaged power consumption in Figure 4.7. The average power consumption by TinyGSN is more than doubled when the Location Privacy Library is used (~1.25 mW and ~3.4 mW respectively). However, when compared to the average total power consumption by all the remaining apps on the device, this increase is negligible and not impactful on the battery life.



(a) tinyGSN power consumption

(b) tinyGSN compared to other apps

Figure 4.7 – Average power consumption of tinyGSN per second with and without Location Privacy Library. Our Location Privacy Library introduces considerable amount of CPU-related power consumption compared to the original tinyGSN application as shown in (a), however further analysis reveals that the extra power consumption is negligible as tinyGSN consumes considerably less power compared to the average power consumption of the sum of all other applications on the device as seen in (b).

We observed the memory usage of TinyGSN with our library and observed that the library introduces around 10 to 20MB memory usage. In Figure 4.8, we see that the allocated memory for TinyGSN as observed in Android Studio's debugger, reaches up to 36 MB and then drops to 24 MB (as a result of the garbage collection system). Consequently, our library does not introduce a big overhead in the memory usage.

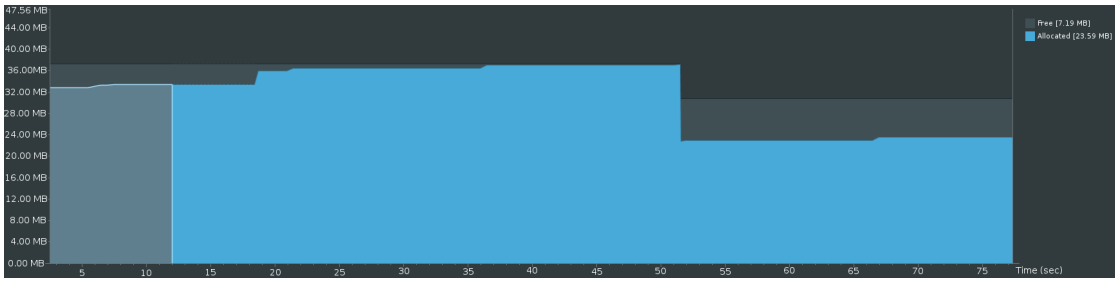


Figure 4.8 – Memory usage of the tinyGSN application with Location Privacy Library.

We determined the CPU usage by using the Android Device Monitor. We sampled 50 measurements for all scenarios. Figure 4.9 shows the CPU Time usage in milliseconds by TinyGSN in all scenarios, where *Data Process* is the operation in the TinyGSN's corresponding virtual

sensor in which the data acquisition from the hardware, processing of the data (including the obfuscation if enabled) and storing is done. We also explicitly plot the time taken by the protection mechanism to show its effect in terms of time delay. As can be seen, TinyGSN runs around ~25 times longer than the usual in the case of without active device usage and around ~10 times longer in the case of with active device usage, when the protection mechanism is enabled. These factors are quite large, however it is important to analyze how this affects the application's functionality. In TinyGSN, the location acquisition service runs every 15 seconds. Normally, this is quite frequent as the context is a sensing scenario. We take this frequency as a baseline and regard it as a lower bound. In our experiments, the maximum real time taken by the data process operation of tinyGSN with obfuscation is 4086.85 milliseconds. As a result, the delay introduced by the Location Privacy Library does not hurt the application's main functionality. We also measured the difference between CPU time usage and real time usage: average time difference between the real time and the CPU time it takes for tinyGSN to process the data with obfuscation is 126.77 milliseconds. On the side note, one strange outcome in Figure 4.9 is that the CPU time for the scenarios with active device usage is less than the case without active device usage. We believe this is caused by the fact that in sleep mode, the CPU switches to lower frequency modes and therefore processing gets slower.

Figure 4.9 – CPU usage of Location Privacy Library within tinyGSN with and without user activity on the device.

4.3.3 Utility

Utility is quite application dependent and for the case of tinyGSN, we do not have access to a sensor data we can use for utility evaluation in this work. Nevertheless, it is intuitive that an adaptive location-privacy protection mechanism generates obfuscation areas of varying sizes instead of setting one-size cloaks. Consequently, varying sizes mean that there will be small obfuscation areas depending on the user's privacy sensitivities and hence cause

less utility loss while still providing enough protection to the user. This was experimentally investigated in Chapter 2 [11] and shown that our adaptive approach degrades the application utility considerably less compared to static obfuscation mechanisms that always generate obfuscation areas of the same size.

4.4 Discussion

The work presented in this chapter is a demonstration of the applicability of an adaptive, yet lightweight, location-privacy protection mechanism for mobile devices, without relying on an external server for estimating the privacy level of the user. This library paves the way for further developments in this direction and enables us to pursue the following issues arising from this implementation. First of all, the representation of privacy sensitivities is not straightforward and our current approach in this matter might be too abstract for many users. This aspect of the work needs improvement and research from a human-computer interaction point of view. Additionally, we assigned default sensitivity values to some semantic tags based on a small-scale survey among the scientists involved in this work and it may not represent a common sensitivity profile in the society. A research on such a sensitivity perception from users is necessary to complement the presented work in this thesis.

Additionally, different people may have different perceptions of the semantics of some locations. This means that personalization in the semantic dimension for sensitivities is required. In the future, we are planning to improve the sensitivity profile interface to let users add personal tags and annotate locations with them. The current implementation of our library lets developers add a user interface that gives users a history of their obfuscated traces and related privacy levels evaluated by the local privacy estimation module. Similar to the case of sensitivities, such a feature requires a social research for communicating the privacy implications to the user and what the privacy levels actually mean.

Last but not least, further analysis of utility in different application types should be studied in relation to location privacy. Depending on the utility requirements, the impact of privacy protection on utility is expected to vary from negligible to substantial. Also, as in the case of participatory sensing, if any additional sensor data is disclosed along with the location traces, it can also be exploited by a potential adversary to infer user location. This dimension of location privacy pose a challenging task and is worth to explore.

4.5 Related Work

Even though there have been numerous works on providing privacy-protection solutions for mobile users in the context of location privacy, there are a handful of implementation attempts for realization of these solutions. In this regard, Fawaz *et al.* [46] analyzed the efficacy of the operating system-level access permission settings for location privacy. They show that the existing designs in the OSs do not give users enough power for access control and propose the

LP-Doctor application on Android that lets users decide per-session access control for location information to location-aware applications. They also proposed a location-privacy protection framework for Android OS called LP-Guardian [47] that is user mobility aware and based on permissions. It differs from our work in the modeling of user mobility and also sensitivity awareness. Nevertheless, their architecture should allow integrating of our library to their framework to benefit from the functions of both approaches.

Enck *et al.* [44] approaches the privacy problem on mobile devices from a violation point of view. They developed the TaintDroid application for the Android platform in order to monitor what privacy information on the user device the applications access (with permission) and potentially abuse it (*i.e.,* excessive data collection). TaintDroid very extensively monitors applications' activity by hooking into private information in the OS and follows if any requested private information is sent to a network device. Though TaintDroid does not offer any protection mechanism against privacy threats, it helps expose privacy-unfriendly applications and act upon that.

Finally, BlurSense [26] and SemaDroid [115] are two data-oriented frameworks developed for mobile devices that controls access to the devices' sensor data and (semi-)automatically allow, block access or provide coarse-grained data to the requester applications. The frameworks' main purpose is to enable privacy-preserving solutions to be easily integrated for various sensor data types. Both applications require the requesting applications to adapt requesting data through these frameworks.

4.6 Summary

We presented an Android library that hosts local privacy level estimation based on user history and an adaptive location-privacy protection mechanism. Our library is based on a novel approach that is aware of location semantics and user sensitivities. It takes into account a sophisticated adversary by emulating his attack and thus estimates the users' location privacy continuously. The library has been integrated with the tinyGSN Android application for participatory sensing, showing that it can be easily embedded in real-life applications. We also evaluated this library in terms of performance, resource usage, and expected privacy in a real-life scenario with participants of an air-pollution campaign.

Semantic-aware Location-Privacy – Sporadic Disclosure Scenarios

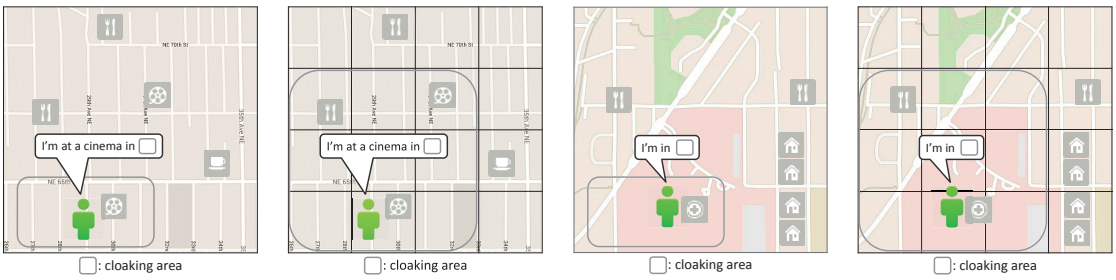
Part II

5 Privacy Implications of Location Semantics

Many online service providers interact with their users on a multidimensional scale. Foursquare, for instance, lets its users check-in at specific nearby venues (selected from the Foursquare database of registered and confirmed venues, e.g., 'Super Duper Burger' in San Francisco), attach pictures and messages to their check-ins and report co-location with other users. Such location check-ins by themselves contain geographical information but also semantic information: For instance, the aforementioned venue is located at '2304 Market St' and is tagged as 'Burger Joint', which is a sub-category of 'Restaurant', which itself is a sub-category of 'Food' in Foursquare categories (see Figure 5.2). Hence, the approach to location privacy from a purely geographical perspective is not sufficient anymore. Additional dimensions of information about the activity of users can be exploited by service providers, thus reducing the effectiveness of existing privacy-protection mechanisms and threatening users' privacy. First, semantic information serves as additional location information: Knowing that a user is in a restaurant reveals some information about her location. Second, semantic information, combined with location information, can be exploited by learning patterns at the semantic level (e.g., people go to cinemas after going to restaurants). Such patterns are already available to (and used by) Foursquare, which makes next-venue recommendations to its users, e.g., "Places people like to go after 'Super Duper Burger': 'Castro Theatre (Movie Theatre, 429 Castro St)'" (see Figure 5.2).

Figure 5.1 depicts two examples where the semantic dimension (*i.e.,* the venue type) of a location can be exploited to infer the actual location and where the semantics of the user's location is not being protected at all. In Figure 5.1a, we observe that a user who visits a cinema discloses that she is in the depicted cloaking area and at a cinema. Because there is only one cinema in this cloaking area, one can easily pinpoint the user. In another example, depicted in Figure 5.1c, a user is at a hospital and wants to protect her location privacy. Unfortunately, her cloaking area is mostly occupied by the hospital, hence even though her exact location might not be pinpointed, the fact that she is at a hospital can be inferred with high confidence.

In this chapter, we consider the case where users disclose not only their (obfuscated) geographical locations but also the (obfuscated) types of venue they are at in the form of check-ins



(a) (b) (c) (d)

Figure 5.1 – Illustrative examples of the privacy threat caused by location semantics.(a) A user reports that she is in the depicted cloaking area and also that she is at a cinema. Her location can be easily pinpointed as there is only one cinema in the user's reported cloaking area, and this cinema occupies a small area compared to the cloaking area. The situation depicted in (b) demonstrates how the issue illustrated in (a) can be reduced by enlarging the cloaking area to include another cinema. An adversary can still narrow down the set of possible locations in the cloaking area, but now there are two locations with the tag cinema. (c) A user at a hospital reports a cloaking area without revealing her semantic information. As the hospital occupies a large proportion of the cloaking area, an adversary can infer that she is at a hospital, thus threatening the user's semantic location privacy. The situation depicted in (d) demonstrates how semantic location privacy can be protected better by generating large cloaking areas to avoid domination of only one type of location in the reported cloaking areas to address the issue illustrated in (c).

on social networks, e.g., "Restaurant, downtown San Francisco". Being able to report such obfuscated information would require to make some modifications on the service. For instance, users could obfuscate their Foursquare check-ins and re-post an obfuscated version of them in a textual message on another social network (e.g., Twitter). Another solution would be that the service provider returns a list of venues to a user based on her coarse-grained location and lets her select a coarse-grained semantic information (e.g., "Food and beverage"). We focus on the semantic dimension of location check-ins and study its effects on location privacy, both at the geographical and semantic levels. To the best of our knowledge, the work presented in this chapter is the first to confront, through data-driven experimentation, semantic information and semantic-aware location privacy protection mechanisms with a practical attack conducted by a concrete adversary. In a nutshell, we formalize the problem and build specific Bayesian networks to model users' behavior on which an adversary runs its inference attacks and we experimentally evaluate both geographical and semantic location privacy under such an adversarial model. In our experiments, we use the semantically-annotated location traces composed of Foursquare check-ins (collected through Twitter's public stream) of hundreds of users distributed across six large cities in North America and Europe. We also rely on a predictive utility model for obfuscated Foursquare check-ins [19]. We show that disclosing information about the type of visited locations, *i.e.*, semantic location-information, decreases geographical location privacy by more than 50% (see Figure 5.8). For instance, in the extreme case where users disclose the precise type of venue they are at, their location privacy drops

Figure 5.2 – Illustration of the information available to location-based social networks such as Foursquare: geographical (*i.e.*, address) and semantic (*i.e.*, venue category) information, semantic mobility profiles (*i.e.*, 'Places people like to go after…'), *etc.* The most relevant pieces of information are circled in red.

by 55% (from 420 m to 190 m). We also present the threat on semantic location privacy that deteriorates quickly as the adversary gains background information on user-mobility profiles, that are easy to build by crawling data publicly available on various social networks. To the best of our knowledge, this is the first work that quantifies semantic location privacy and demonstrates the effects of location semantics on location privacy.

5.1 Background and System Model

We consider mobile users equipped with smartphones that have localization capabilities and Internet connectivity. These users move in a geographical area and make use of location-based online services. We consider that users sporadically report their (potentially obfuscated) locations and, in some cases, semantic information (*i.e.*, the type, in the form of tags such as 'restaurant') of their locations. In this setting, we consider an honest-but-curious service provider that is interested in inferring, based on its observations, users' actual geographical locations and the semantic tags associated with them, if any. Table 5.1 lists the notations used in this chapter. Our model is built on top of Shokri et al.'s [100]; we detail the differences in Section 5.5.

[1] $\mathcal{P}$: Power set.

Table 5.1 – Table of Notations.

$\mathcal{R}$	Set of geographical regions
$\mathcal{S}$	Set of semantic tags
$a_u(t) = (r, s)$	User u's actual location at time instant t, where $r \in \mathcal{R}$ and $s \in \mathcal{S}$
$o_u(t) = (r', s')$	User u's obfuscated location at time instant t, where $r' \in \mathcal{P}(\mathcal{R})$[1]
a_u	Actual trace of user u
o_u	Obfuscated trace of user u
$\mathbf{R}_t, \mathbf{R}'_t$	The actual and obfuscated geographical location variables for time t
$\mathbf{S}_t, \mathbf{S}'_t$	The actual and obfuscated semantic location variables for time t
$h_u(r, r', s, s')$	A PPM modeled as a probability distribution function (PDF) employed by user u (decomposed into $f_u(r, r')$ and $g_u(s, s')$)
q_g, q_s	The PDF output by the inference attack
$\text{dist}^G(\cdot,\cdot), \text{dist}^S(\cdot,\cdot)$	Geographical and semantic distance metrics used for quantifying privacy
$\text{GP}_u(t), \text{SP}_u(t)$	User u's geographical and semantic location privacy at time t

5.1.1 Users

Mobile users with GPS-equipped connected devices move in a given geographical area that is partitioned into M non-overlapping geographical regions/cells $\mathcal{R} = \{R_1, R_2, \ldots R_M\}$. Geographical regions are usually coarse-grained (typically cells associated with cell towers or regular square tiles of a several hundreds of meters). A subset of, or all, the regions in $\mathcal{R}$ contain venues annotated with semantic tags from the set $\{S_1, S_2, \ldots, S_K\}$, *i.e.,* a predefined list of categories (e.g., Foursquare defines such a list, organized as a tree [3] and all registered venues are tagged with such a category). Whenever a venue is visited by a user, it is mapped to the geographical region from $\mathcal{R}$ it falls in. We denote by $\perp$ the semantics of regions for the case when a user is in a geographical region, but does not visit a particular venue with a semantic tag, meaning that her location does not have semantic information. Hence, we define the set $\mathcal{S}$ of semantic tags as the union $\{S_1, S_2, \ldots, S_K\} \cup \{\perp\}$ to cover all semantic cases. Moreover, we consider discrete time instants over a limited-time period $\{1, \ldots, T\}$. Note that the notion of venue types was introduced in the work of Shokri et al. [97].

As users move, they sporadically use online services and share their (potentially obfuscated) locations together with the corresponding (potentially obfuscated) semantic tags. Formally, whenever a user u visits a geographical region r at a time instant $t \in \{1, \ldots, T\}$, she generates an event consisting of her actual geographical region $r \in \mathcal{R}$ and a corresponding semantic tag $s \in \mathcal{S}$. This user event at time t is denoted by $a_u(t) = (r, s)$; in other words, the *actual location* of user u at time instant t is represented by the pair (r, s). We denote by $a_u = \{a_u(1) \ldots a_u(T)\}$ the whole trace of user u.

5.1.2 Privacy Protection Mechanisms

For privacy reasons, users employ privacy-protection mechanisms (PPMs) before reporting their location and semantic information to an online service provider[2] Their privacy goal is to

[2] In the remainder of the chapter, we refer to the *online service provider* as the *service provider* or the *adversary* for short.

prevent the adversary from inferring at what geographical location and in what type of venue they are at. Typically, a PPM, that aims to protect the geographical location of a user, replaces her actual location with another location (*i.e.,* perturbs the location) or with a list of locations (*i.e.,* a cloak), or hides the location information completely. In this chapter, we consider such PPMs and the PPMs that protect the semantic dimension of the location, specifically the semantic tag of a user's event. In particular, these PPMs generalize the semantic tag (*i.e.,* report a parent tag of the venue's actual tag, w.r.t. a tag hierarchy, e.g., replace 'Burger joint' with 'Restaurant'[3] or 'Food') or hide it completely. We assume that a set of PPMs obfuscates a user's actual event at time t independently from her other events at other time instants. Such a PPM model can also cover the cases where the underlying localization technique used by the adversary returns coarse-grained and possibly bogus information about the users.

After applying PPMs on her actual geographical region r and the corresponding actual semantic tag s, a user u reports her obfuscated geographical region r' and the obfuscated semantic tag s' to the service provider. r' (resp. s') is typically a subset of $\mathcal{R}$ (resp. $\mathcal{S}$). We assume that the service provider only observes the obfuscated trace $o_u = \{o_u(t) = (r', s')\}, \forall\, t \in \{1 \dots T\}$ of user u. We model a PPM as a probability distribution function that maps actual events to obfuscated ones (note that in the case of generalization, the PPM is deterministic). Specifically, we denote by functions $h(r, r', s, s')$ the probabilities to generate the obfuscated location/semantic tag r', s' (*i.e.,* $\mathbf{Pr}(r', s'|r, s)$) that constitute the obfuscated event $o_u(t) = (r', s')$ given the actual event $a_u(t) = (r, s)$. Note that the location of a user at a given time instant is obfuscated independently from the other time instants.

Finally, we do not consider collaboration between users to protect their privacy (and prevent loss of privacy from each other). In addition, we assume that users' events are not anonymized.

5.1.3 Adversary

The adversary we consider in this chapter is typically a service provider or an external observer who has access to obfuscated traces of users. He has two main purposes: (1) locate users at specific time instants, and (2) identify the types of the locations a user visits at specific time instants, in terms of the semantic tags associated with them. While carrying out his attack, the adversary takes into account the relationship between geographical and semantic dimensions of location, as explained in Section 5.2. Note that the inference process described below also applies to other adversaries such as users' friends and third party services on which users' check-ins are reposted (e.g., Twitter). However, the amount and the granularity of the information that is available to them can be more limited.

The adversary runs his attack *a posteriori, i.e.,* after having observed the whole obfuscated trace o_u of a user u. Even though the obfuscation of an event is done independently from the other events of the user, the adversary assumes that a user's actual events are correlated and therefore models the users' mobility/behavior. He is assumed to have access to users' (partial)

[3] Note that this is strictly equivalent to reporting the sets of all tags that are sub-categories of tag 'Restaurant'

past events that he exploits to build a mobility profile for each user u, on both the geographical and semantic dimensions. Essentially, a user's mobility profile represents the user's transition probabilities over successive time instants, *i.e.,* between geographical regions and between semantic tags. Formally, such a mobility profile (under a first-order Markovian assumption) is the set of the probability distribution functions $\mathbf{Pr}(r|\rho)$, $\mathbf{Pr}(s|\sigma)$ and $\mathbf{Pr}(r|s)$, where ρ and σ represent the user's previous location and semantic tag (as explained in Section 5.2).

The adversary also knows which PPMs a user u employs and with what parameter(s), *i.e.,* the function h_u. Together with the PPMs and the mobility profile he generates, the adversary performs his attack on a user trace given her obfuscated trace o_u.

5.2 Inference and Privacy

We explain our model of inference and background knowledge of the adversary in the subsequent subsection. In summary, we build two user behavior models by using Bayesian networks [63, 88] under the assumption that people follow a bi-modal Markovian mobility process[4] (along the geographical and semantic dimensions) which we describe below. These models take into account both the geographical and semantic dimensions of the location and also the relationship between them. Based on these two models, we evaluate geographical and semantic location privacy.

5.2.1 Inference and Background Knowledge

We assume that the adversary uses the following simple behavioral user model in the inference process[5]: Users move based on what they plan to do next given their current context, *i.e.,* in this case, their locations and semantic information. We determine the following two scenarios (illustrated in Figure 5.3):

1. The adversary knows the users' geographical transition profile, *i.e.,* the *geographical background*, and assumes that the users move to new locations primarily based on their current locations. The type of place they visit (*i.e.,* semantic tags) depend only on their current locations. For instance, a user might go to a location in downtown after visiting another location in nearby downtown. The semantics of these locations then, for instance, might happen to be a cinema and a restaurant.
2. The adversary knows both the users' geographical and semantic transition profiles, together referred to as *geographical & semantic background*. Unlike the first scenario, in this case the user first determines what type of place she will go to (*i.e.,* her next activity,

[4] This means that a user's events at a given time instant only depend only on that user's event at the immediate past time instant.

[5] Note that the user traces we use in our experiments are real and are not generated from this model. Therefore, the fact that the considered user models rely on a set of simplifying assumptions limits the performance of the inference; as such, the experimental results presented in this chapter constitute a lower bound of the privacy implications of semantic information.

characterized by the semantic tag of the venue she visits next) given the semantic tag of her current location, and then chooses the region she will go to based on the determined next semantic tag and her current location. For instance, if a user is at a restaurant in downtown and wants to go to a cinema, she chooses to go to a cinema that is close to her current location (that she often visits).

For the sake of simplicity for our experimentation, from this point on, we assume that geographical and semantic information are obfuscated independently from each other, using two functions f_u and g_u respectively (note that it is straightforward to include such joint PPMs in our formalism). Joint PPMs could be used to avoid the situations where a user reports a set of geographical locations and a semantic tag such that only some of the reported locations contain a venue with this tag.

We elaborate more on our scenarios and their respective Bayesian networks in the following sections.

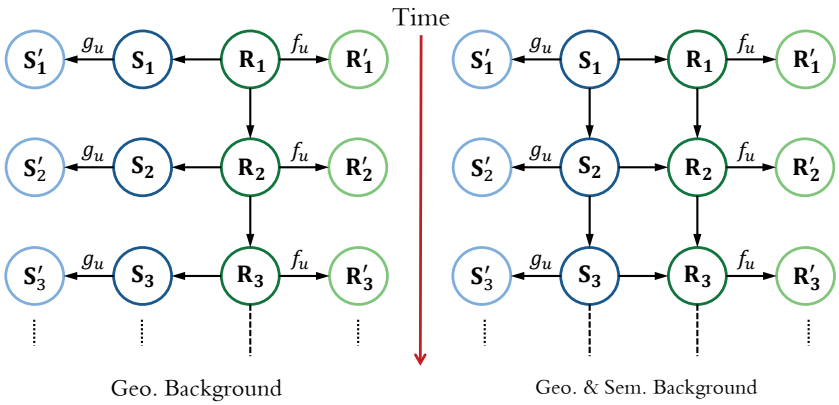


Figure 5.3 – The Bayesian networks representing the user models employed by the adversary. Nodes denote random variables and edges denote probabilistic dependencies between them (e.g., the arrow from $\mathbf{R}_1$ to $\mathbf{R}'_1$ corresponds to the obfuscation function f_u). The model on the left-hand side prioritizes geographical transitions with only geographical background known to the adversary. The model on the right-hand side prioritizes semantic transitions over geographical transitions with both geographical and semantic background. Protection mechanisms work separately on regions and semantic tags and they are independent.

Geographical-Only Background

As stated previously, the adversary has access to the users' geographical transition profile (built from past traces) in this scenario and carries out his attack by using (only[6]) this information as background information. He can correlate the sequential events of a user by using geographical background information, hence we build a Bayesian network in which only the region (*i.e.*, the geographical location) nodes are connected to each other among user events.

[6]The purpose of considering such a limited adversary, used as a baseline, is solely to show the *inference power* of semantic background used in Section 5.2.1.

As the adversary still wants to infer the semantic tags in the user events, semantic nodes are also created and they are dependent on the region nodes. This ensures that the adversary benefits from the semantic information disclosed by the users in his inference, even though he does not have any semantic background information.

This model is illustrated in Figure 5.3 (left), where each line of nodes represent a user event in time, both actual $(\mathbf{R}_t, \mathbf{S}_t)$ and obfuscated $(\mathbf{R}'_t, \mathbf{S}'_t)$, where $\mathbf{R}_t$, $\mathbf{S}_t$, $\mathbf{R}'_t$ and $\mathbf{S}'_t$ represent the random variables for a user's actual and obfuscated events at time t. The conditional probability distributions for the obfuscated events', *i.e.,* for $\mathbf{R}'_t$ and $\mathbf{S}'_t$), are the privacy-protection mechanism distributions f_u and g_u, explained in Section 5.1.2. If a static privacy-protection mechanism (PPM) is used by the users, then these functions map the actual regions and the actual semantic tags to obfuscated regions and obfuscated semantic tags with probability 1 (*i.e.,* for a given region, resp. a semantic tag, the PPM always generates the same obfuscation outcome). More powerful PPMs can be employed and used in this network, e.g., hiding the actual information completely with a given hiding probability.

The remaining conditional probabilities are those of the user's actual semantic tag given her actual location $\mathbf{Pr}(\mathbf{S}|\mathbf{R})$ and the user's next location given her current location $\mathbf{Pr}(\mathbf{R}_{t+1}|\mathbf{R}_t)$. We calculate $\mathbf{Pr}(\mathbf{S}|\mathbf{R})$ based on the semantic tags' associations to regions as the adversary is assumed to have no semantic background information. Essentially, $\mathbf{Pr}(\mathbf{S}|\mathbf{R})$ represents a uniform distribution over all semantic tags associated with a region r, e.g., if a region has 4 semantic tags associated with it, then the probability for each of these tags to be the actual tag given this location is 0.25. Lastly, we compute $\mathbf{Pr}(\mathbf{R}_{t+1}|\mathbf{R}_t)$ by counting the number of transitions among all regions in a user trace and then using the knowledge construction approach from [100].

Geographical and Semantic Background

In this scenario, we consider an adversary that models user mobility-behavior in an activity-driven fashion: A user first determines the type (*i.e.,* the semantic tag) of her next geographical region given the type of her current geographical region; then, she determines the next geographical region given her current geographical region *and* the next semantic tag. For example, a user decides to go to a restaurant, then she chooses which restaurant she wants to go to. Afterwards, she decides to go to a cinema, as she usually does after going to a restaurant. Considering her previous location, she picks the cinema that is most convenient for her. This model is depicted in Fig. 5.3 on the right-hand side.

The conditional probability distributions for the obfuscated events (*i.e.,* $\mathbf{R}'_t$ and $\mathbf{S}'_t$) are the same as in the scenario with only the geographical background knowledge. The transitions between user events, however, now require a semantic-transition distribution ($\mathbf{Pr}(\mathbf{S}_{t+1}|\mathbf{S}_t)$) and a geographical-transition distribution, which is also conditioned on the semantics of the next user-event ($\mathbf{Pr}(\mathbf{R}_{t+1}|\mathbf{R}_t, \mathbf{S}_{t+1})$), meaning that $\mathbf{R}_{t+1}$ depends on the user's current semantic tag $\mathbf{S}_{t+1}$ and her previous geographical region $\mathbf{R}_t$.

The semantic transition distribution $\mathbf{Pr}(\mathbf{S}_{t+1}|\mathbf{S}_t)$ is constructed in the same way the geographical transition distribution $\mathbf{Pr}(\mathbf{R}_{t+1}|\mathbf{R}_t)$ is constructed. However, as we consider geographical and semantic background information separately, the adversary is assumed not to know the distribution $\mathbf{Pr}(\mathbf{R}_{t+1}|\mathbf{R}_t, \mathbf{S}_{t+1})$. In short, the adversary is assumed to have knowledge on $\mathbf{Pr}(\mathbf{S}_{t+1}|\mathbf{S}_t)$, $\mathbf{Pr}(\mathbf{R}_{t+1}|\mathbf{R}_t)$ and $\mathbf{Pr}(\mathbf{R}_t|\mathbf{S}_t)$ to some extent regarding user history. Therefore, he needs to use $\mathbf{Pr}(\mathbf{R}_t|\mathbf{S}_t)$ and $\mathbf{Pr}(\mathbf{R}_{t+1}|\mathbf{R}_t)$ to derive $\mathbf{Pr}(\mathbf{R}_{t+1}|\mathbf{R}_t, \mathbf{S}_{t+1})$. We achieve this simply by normalizing the marginal probability distribution $\mathbf{Pr}\{\mathbf{R}_{t+1}|\mathbf{R}_t\}$ for a given semantic tag s (*i.e.,* over regions that have s) and by combining it with the conditional distribution $\mathbf{Pr}(\mathbf{R}_t|\mathbf{S}_t = s)$. For the rest of the geographical regions, *i.e.,* those that do not have the semantic tag s, the probability is zero. This translates into the following formula:

$$\mathbf{Pr}(\mathbf{R}_{t+1} = r|\mathbf{R}_t = \rho, \mathbf{S}_{t+1} = s) = \tag{5.1}$$

$$\begin{cases} 0 & \text{if } s \notin r \\ \alpha \cdot \dfrac{\mathbf{Pr}(\mathbf{R}_{t+1} = r|\mathbf{R}_t = \rho)}{\sum\limits_{R_m \text{ s.t. } s \in R_m} \mathbf{Pr}(\mathbf{R}_{t+1} = R_m|\mathbf{R}_t = \rho)} + (1-\alpha) \cdot \mathbf{Pr}(\mathbf{R}_{t+1} = r|\mathbf{S}_{t+1} = s) & \text{otherwise} \end{cases},$$

where R_m denotes the set of regions that contain at least one venue with tag s and α is a factor to set the weight of geographical transitions against the probability that $\mathbf{R}_{t+1}$ is r given $\mathbf{S}_{t+1} = s$ (which is derived from the number of visits to a region r given the semantic tag s in the user history). In other words, α is used to control how much importance is distributed among different types of user history, *i.e.,* geographical transitions and steady user events. In our experiments, we set α to 0.5, which we believe is a balanced treatment of user history.[7] Note that considering geographical and semantic background information separately enables the adversary to exploit the semantic mobility of a user's behavior data in one city to infer user events in another city, where he might lack the knowledge.

Note that the aforementioned models might not reflect the users' actual behaviors. However, such models (in particular the Markovian mobility assumption) are widely used in practice (and considered in the literature) as they enable the adversary to develop efficient algorithmic and computational methods to infer the users' locations. The accuracy of the inference attack carried out by the adversary partially depends on how well the user model fits the users' actual behaviors.

5.2.2 Privacy Measurement

Due to different privacy concerns in both geographical and semantic dimensions of location, we measure the privacy level in both dimensions separately. Privacy levels in both dimensions are measured as a function of the expected error of the adversary. The inference based on our Bayesian networks yields probability distributions over regions and semantics that fit this measurement approach. In other words, the output of the inference algorithm is a probability distribution function (PDF) for each node in a given Bayesian network, *i.e.,* the PDF q_g over all

[7] We ran test experiments with different values of α; we observed only small variations ($\sim 5\%$) of the median error, with better results for large values of α (> 0.5).

regions at every time instant for user location and the PDF q_s over all semantic tags at every time instant for user semantic tag. The geographical and semantic privacy levels of a user u at time instant t, denoted by $\mathrm{GP}_u(t)$ and $\mathrm{SP}_u(t)$, are computed as follows:

$$\mathrm{GP}_u(t) = \sum_{m=1}^{M} q_g(R_m, t) \cdot \mathrm{dist}^G(R_m, r), \tag{5.2}$$

$$\mathrm{SP}_u(t) = \sum_{k=1}^{K} q_s(S_k, t) \cdot \mathrm{dist}^S(S_k, s), \tag{5.3}$$

where $\mathrm{dist}^G(\cdot,\cdot)$ and $\mathrm{dist}^S(\cdot,\cdot)$ are geographical and semantic distance functions, and (r, s) is the actual event of user u at time instant t.

We use the Euclidean distance (in the projected coordinate system, *i.e.,* Universal Transverse Mercator or UTM)[8] to compute the geographical distances between two regions by using the projected coordinates of their respective center points. We use the distance metric $d(\cdot)$ from graph-theory (*i.e.,* the length of the shortest path between two nodes) on the category tree to compute the semantic distance between two tags, meaning that if two semantic tags are equal, then the distance is 0, if they have the same parent tag (e.g., 'American restaurant' and 'Burger joint' are both children categories of the 'Restaurant' category), the distance is 2, *etc.* We normalize the semantic distance between two tags by the sum of the tags' depths (*i.e.,* the distance to the root).

$$\mathrm{dist}^S(s, s') = \frac{d(s, s')}{d(\text{'venue'}, s) + d(\text{'venue'}, s')} \tag{5.4}$$

This distance function takes into account the fact that, as one goes deeper in the tree, the graph-distance denotes a less significant semantic difference. For instance, "Italian Restaurants" and "American Restaurants" are not so different but "Food" and "Travel place" are.

5.3 Evaluation

We experimentally evaluate privacy on a real dataset of user traces composed of location check-ins that contain not only geographical location data but also semantic information in most cases (see Section 5.3.1). In our experiments, we study the effects of location semantics on the geographical location privacy by comparing the privacy of users under a semantic-oblivious and a semantic-aware inference attack, in various configurations and with different PPM settings.

5.3.1 Dataset

In order to experimentally evaluate users' semantic location privacy and the effect of semantic information on users' location privacy, we rely on a dataset of real user check-ins, which include geographical and semantic information about the venues visited by the users of a large

[8] Note that we did not take elevation into account in the computation of the geographical distance.

location-based social network. In addition, we rely on a predictive utility model based on user feedback collected through a personalized online survey targeted at Foursquare users ($N = 77$) recruited via the Amazon Mechanical Turk platform. This dataset was collected by the authors of [19] and made available online at https://homepages.laas.fr/khugueni/drupal/datasets. In this section, we give details about our data sources, including the data collection, filtering and processing methodology and general descriptive statistics about the data.

Location Traces with Semantics

Because we could not find large datasets of user check-ins with semantic information, we built our own dataset by running a data collection campaign through crawling. As a starting point, we use a tweet dataset we collected between January 2015 and July 2015 through Twitter's public stream. The dataset contains public geo-tagged tweets (*i.e.,* Twitter lets users to attach their GPS coordinates to their tweets); we focused on six large cities: Boston (MA, USA), Chicago (IL, USA), Istanbul (Turkey), London (UK), New York (NY, USA) and San Francisco (CA, USA). We collected these tweets by identifying users through Twitter's public stream (*i.e.,* ~1% of the Twitter public timeline) and by fetching timelines of these users. A summary of the statistics of the dataset is provided in Table 5.2: We collected location check-in traces of a total of 1065 users. As we collected only public data and we neither interacted with the user nor inferred information not present in the dataset, IRB approval was not required.

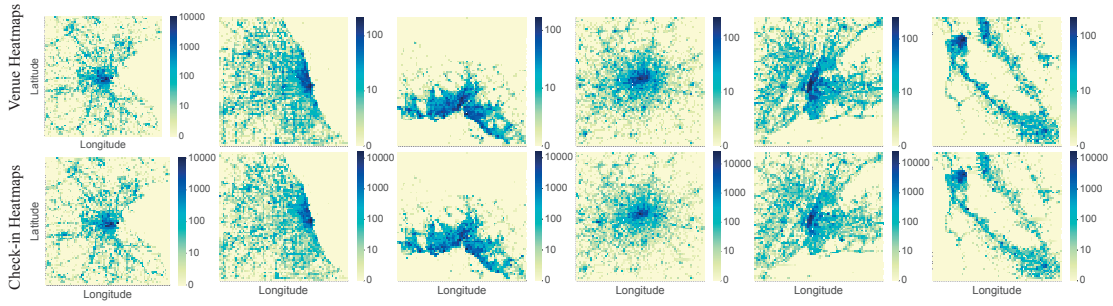
Table 5.2 – Filtered Dataset Statistics

City	Users	Tweets	Check-ins
Boston	79	6,687	5,276
Chicago	136	14,248	11,755
Istanbul	196	22,203	17,005
London	239	18,685	15,018
New York	242	21,249	14,240
San Francisco	173	16,739	13,650

The coordinates embedded in the geo-tagged tweets, however, do not contain semantic information (which we need for our evaluation). To obtain such information, we rely on Foursquare. Foursquare offers its users the option of linking their Foursquare accounts with their Twitter accounts in such a way that, whenever a user checks-in, Foursquare generates an automatic text message with a short URL to the Foursquare check-in and tweets it, along with the GPS

Table 5.3 – Experimental Setup

Number of iterations	10
Size of each area	2.4 × 1.6 km (12 × 8 cells)
Average Proportion of Foursquare tweets per user (*i.e.,* tweets w/ semantic information)	77%



(a) Boston (b) Chicago (c) Istanbul (d) London (e) New York (f) San Francisco

Figure 5.4 – Foursquare venue and check-in heat maps (*i.e.*, count distribution) in six cities from the raw dataset.

coordinates, on the user's Twitter timeline. We select such Foursquare-generated tweets from our Twitter dataset and, for each, we parse the URL to the Foursquare check-in from the tweet text. Using these URLs, we fetch (through the Foursquare API) the corresponding check-in and the venue. For each venue referenced in a check-in of our dataset, we collect rich statistical information such as total number of visits, total unique visitors, rating, etc. Most importantly, we collect the coordinates[9] and the semantic tag(s) (a primary tag and possibly a secondary tag), selected from a pre-defined set of 763 tags (*i.e.*, referred to as Foursquare categories) organized as a tree (see Figure 5.5 for a snapshot of the tree), assigned to the venue. We used Foursquare's definition and implementation of location semantics *as is*. The results of our evaluation are dependent of the underlying semantic model; investigating alternative definitions of location semantics and other categorizations (e.g., from Facebook) is an interesting lead for future work. Because it uses semantic tags (organized as a tree) and because its main feature is to let users check-in at venues, Foursquare constitutes a perfect data source for our evaluation. Note that, unlike in works such as Krumm's [69] in which semantic information is *inferred* from the users' location traces, we use only ground-truth semantic data extracted from the users' check-ins. We show the venue density and the Foursquare tweet density in the considered cities in Figure 5.4, which shows a Foursquare venue heat map and a Foursquare check-in heat map.

In our evaluation, we focus (due to computational limitations) on the tweets and check-ins in small geographical areas of size approximately 2.4 × 1.6 km around the cities of Boston, Chicago, Istanbul, London, New York and San Francisco. We define one such area around each of the six cities, and we divide each of them into 96 cells by using a regular grid of 12 × 8 cells (each of size 200 × 200 m). We determine the most dense such areas and extract users with at least 40 tweets in each region. We further filter out users whose Foursquare tweets (*i.e.*, check-ins) account for less than 50% of all their tweets (*i.e.*, most of the tweets used in the experiments contain venue information). The final dataset contains a total of 1065 users (57%

[9] Note that GPS coordinates in the tweets might slightly differ from registered venue coordinates at Foursquare. In such cases, we use the coordinates of the venues from Foursquare.

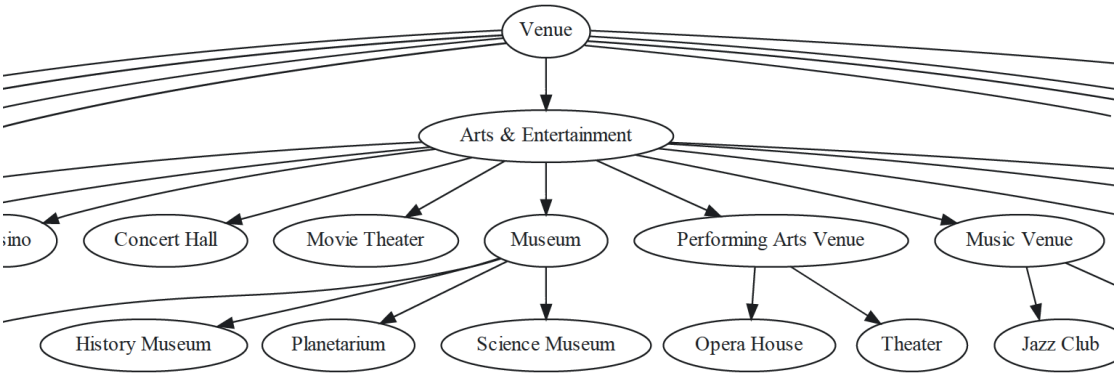


Figure 5.5 – Partial view of the Foursquare category hierarchy that we use as our semantic tag tree in our evaluation. The 'Venue' tag is the root of the category tree.

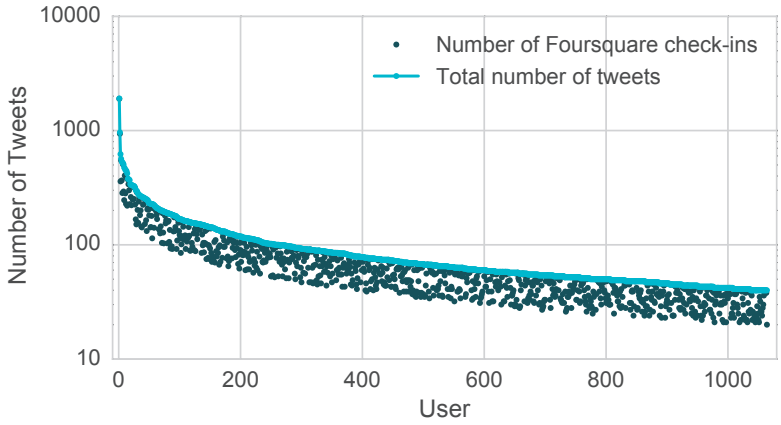


Figure 5.6 – Number of Foursquare check-ins/tweets and the total number of tweets per user (in decreasing order) in the filtered dataset used in our experiments (log-scale on the y-axis).

male, 41% female, 2% unknown); see Table 5.2 for detailed statistics and Figure 5.6 for users' count of Foursquare and total tweets. We included all the tweets of a user in the knowledge construction of the adversary and for each user we use a randomly selected sub-trace of length 5 in each experiment. There are 10,970 venues in our filtered dataset and the tag distribution over these venues is shown in Figure 5.7.

Dissemination of the dataset Although the terms and conditions of Twitter[10] and Foursquare[11] prevent us from making the dataset directly available for download as we need to make sure that the requesting party agrees to comply with these terms, we will be happy to provide our dataset (and the script used for collecting the data) to other researchers upon request.

The dataset contains all the considered check-ins, each of which is characterized by a timestamp, a user id, a geographical location (as reported in the tweet), a geographical location (as

[10] https://dev.twitter.com/overview/terms/agreement-and-policy
[11] https://developer.foursquare.com/overview/venues

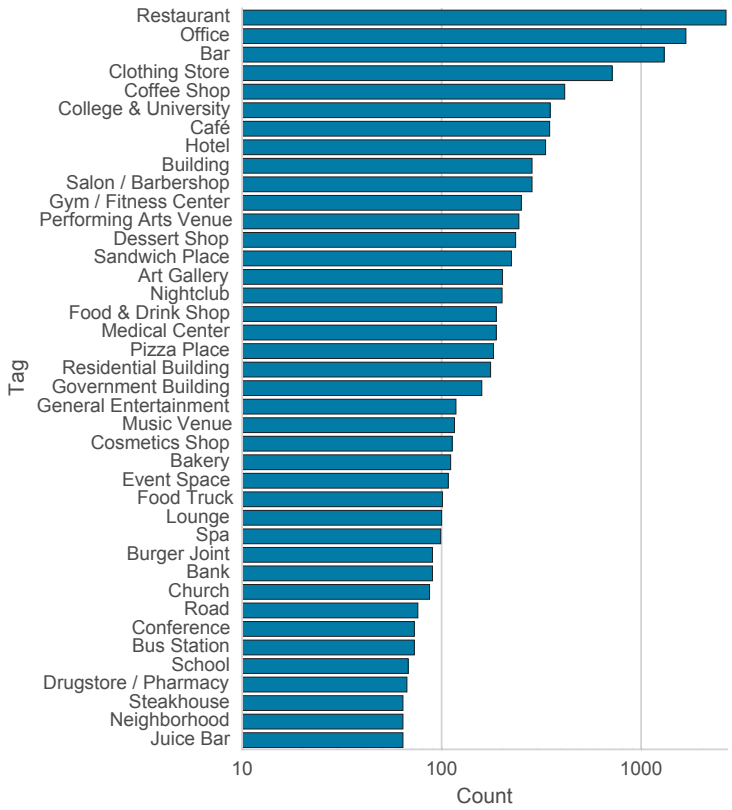


Figure 5.7 – Number of venues per semantic tag in the filtered dataset for the top 40 common tags (log-scale on the x-axis).

reported in the Foursquare venue information), and the Foursquare venue type in the form of a tag, and will be made available in the csv file format. It will also contain a snapshot of Foursquare category tree at the time of data collection.

Predictive Utility Model

Semantic obfuscation, usually achieved through generalization as discussed in the previous sections, is likely to have a negative effect on the utility of the service as perceived by the users. As the notion of (perceived) utility is quite subjective, user feedback is needed to model and quantify the utility implications of the use of obfuscation techniques. In order to build such a model, we rely on a dataset collected and made available by the authors of [19]. The fact that the survey focuses on Foursquare check-ins makes it perfectly adequate for our dataset and hence for our evaluation. In this work, the authors performed a personalized survey with 77 active Foursquare users recruited through Amazon Mechanical Turk. In the survey, each participant was shown 45 of her own past Foursquare check-ins; for each of these check-ins, the participant was presented with four different obfuscated versions of the check-in and she was requested to rate, on a scale from 1 to 5 (where 1 is “not at all” and 5 is “perfectly”), to what extent the purpose of her check-in would still be met if the precise venue location was replaced with the obfuscated version of it. The four obfuscated versions of the check-in were

Table 5.4 – Example of obfuscated check-ins with different combinations of geographical and semantic obfuscation (source: [19]).

Obfuscation levels	Example
Original check-in	The Westin Hotel, 320 N Dearborn St. (Chicago 60654, IL, United States)
Low semantic, Low geographical (Ls-Lg)	At a hotel, on Dearborn St. (Chicago 60654, IL, United States)
High semantic, Low geographical (Hs-Lg)	At a travel & transport place, on Dearborn St. (Chicago 60654, IL, United States)
Low semantic, High geographical (Ls-Hg)	At a hotel, in Chicago (IL, United States)
High semantic, High geographical (Hs-Hg)	At a travel & transport place, in Chicago (IL, United States)

generated by applying the possible combinations of low/high semantic obfuscation (Ls or Hs) and low/high geographical obfuscation (Lg or Hg) as illustrated in Table 5.4 (extracted from the original article). One finding from the article is that semantic obfuscation has a higher negative effect on utility than geographical obfuscation does.

Using this data, to predict the utility of an obfuscated version of a check-in (on a discrete scale from 1 to 5), the authors propose a utility model that relies on a number of features extracted from the users' check-in, including the check-in location, date, time, text, and the venue type. The predictive model proposed in the original paper achieves high accuracy with a median error of around 0.5. In order to quantify utility, we build a simplified version of the predictive utility model proposed in [19] (based on the same data). Our model is based on only two different features: the venue type and the obfuscation level. The median error of our simplified model is 1.1, which is sufficient for our purpose (*i.e.,* exploring the privacy-utility trade-off).

5.3.2 Experimental Setup

Methodology: We partitioned each of the six considered areas (one for each city considered in the dataset) into 96 cells, each identified by an ID, using an 12×8 regular square grid. We then mapped the locations in the users' traces to the corresponding region IDs, and we kept the semantic tag. We implemented our Bayesian network-based models in Python by using the Bayesian Belief Networks library provided by eBay [2]. We applied certain protection approaches (listed below) on the users' traces, obtaining *protected*/observed traces that our Bayesian networks use as observations, and applied the junction-tree inference algorithm [62] which achieves optimal inference. The output of the inference algorithm is a probability distribution function for each unknown (inferred) variable, which we use in our privacy metrics (see Equations (5.2) and (5.3)).

Background Knowledge: In our experiments, the adversary always has geographical background knowledge on the users' history (*i.e.,* transitions). Based on this we have two different scenarios (explained in detail in Section 5.2.1):

1. **Geographical Background:** In this scenario, the adversary is assumed to have knowledge on geographical transition patterns of users and no semantic background information. We run experiments for this scenario by using our first Bayesian network model

that prioritizes the geographical transitions for user behavior introduced in Section 5.2.1. The transition probabilities are estimated from the number of geographical transitions in the whole traces of users.

2. **Geographical and Semantic Background:** The adversary is assumed to have more knowledge about users' histories: transitions in both geographical and semantic dimensions. He also knows the distribution of geographical region visits, given the semantic information on user traces, *i.e.,* how many times a region r was visited, given that the user event's semantic tag was s. This type of background information enables us to use our second Bayesian network model that prioritizes the semantic transitions for event sequences, meaning that the users move by first choosing the semantic tag of the location they want to go to and then determine a geographical region associated with this semantic tag based on their previous location.

In many cases, such information can be obtained by the service provider. In cases where only little background information about individual users is available, the service provider can aggregate data across users with similar profiles.

Protection Mechanisms: We implement geographical and semantic location privacy protection approaches separately, meaning that geographical protection does not take into account the semantic information of the user's actual location, and vice versa. As mentioned above, joint protection mechanisms could be used for improved performance; we leave the design of such mechanisms to future work.

We implement a geographical location-privacy protection mechanism as an obfuscation mechanism that either generates an obfuscation area of a certain size or hides the geographical location completely with a predetermined probability (called the hiding probability λ). This mechanism replaces any given region (*i.e.,* the actual location of a user) with a larger, square area in our map. For instance, a 2×2 obfuscation: (*i*) with probability $1 - \lambda$, generates an obfuscation area consisting of 4 adjacent regions/cells, one being the actual location of the user, or (*ii*) with probability λ, hides the location.

We consider the following four scenarios regarding the semantic protection and, to compare their effects, employ each of them in separate experiments:

1. **No protection**. In this case, we directly disclose the actual semantic tag all the time. From a privacy perspective, this constitutes a worst-case scenario.
2. **Parent-tag obfuscation**. This is a generalization based on the semantic tag tree derived from Foursquare's category hierarchy. In this case, given the actual semantic tag of the user, we determine its parent tag in the tree and disclose this tag as the semantic information of the user's current location. It has been shown, for Foursquare check-ins, that reporting the parent tag of a venue is often sufficient to meet the purpose of the original check-in [19].

3. **Parent-tag obfuscation with hiding**. In this case, we disclose the parent tag of the user's location with probability $1-\lambda$ or hide the semantic information completely with hiding probability λ.
4. **Complete hiding of semantic tags [baseline]**. In this case, we never disclose semantic tags. This corresponds to a pure geographical approach (as taken in previous works); as such it constitutes our baseline.

In our experiments, we employ the geographical protection mechanism in combination with each of the aforementioned semantic protection scenarios with varying hiding probabilities.

5.3.3 Experimental Results

In this section, we analyze the experimental results with different protection mechanisms in various settings.

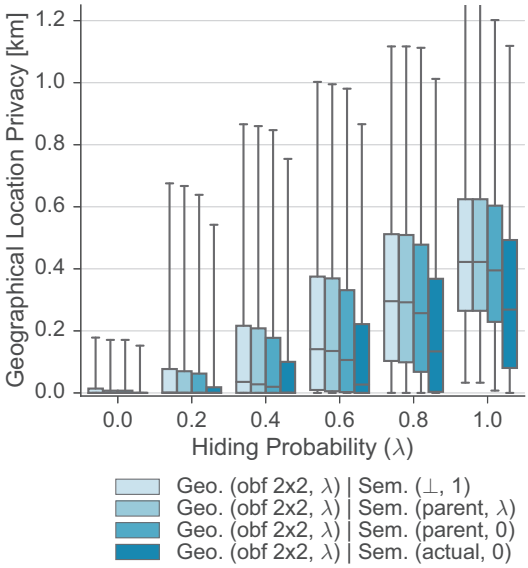
Effect of Semantic Information on Location Privacy

We first analyze the effect of adding semantic information to a user's check-in on her geographical location privacy. We consider four protection scenarios with low to high granularity of semantic information combined with fixed geographical obfuscation over gradual hiding probability λ. Specifically, given a geographical obfuscation parameter (e.g., 2×2 obfuscation), for each λ we evaluate four different semantic protection approaches (explained in Section 5.3.2) that are employed together with the obfuscation mechanism.

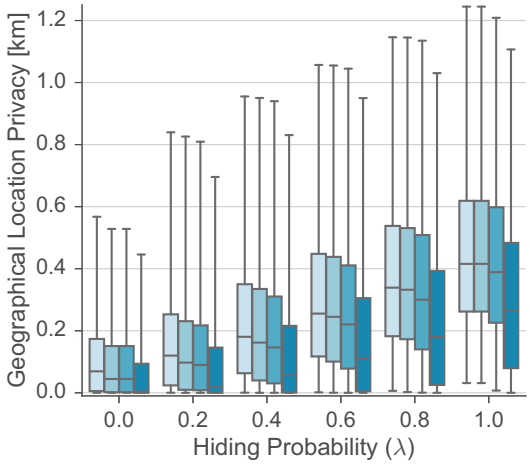
We present the results in Figure 5.8, where the x-axis represents the hiding probability λ (used for geographical obfuscation and parent-tag semantic generalization) and the y-axis represents the geographical location privacy in kilometers (*i.e.,* the distance between a user's actual discretized location and that inferred by the adversary, as described in Equation (5.2)). A privacy of a few hundreds of meters (typically a city-block) provides a reasonable protection against precise localization/tracking and limits the possibility to infer the exact place a user visits or her exact address. We plot the geographical location privacy aggregated over all users, all events and all iterations of simulations for each protection mechanism and hiding probability (λ) pair using box plots. These box plots show the 1st, 2nd, 3rd quartiles of the data and the 98% confidence intervals.

We consider four scenarios (geographical obfuscation and semantic generalization) and plot the corresponding results, e.g., "Geo. (obf 2×2, λ) | Sem. (parent, λ)" means that (1) geographical locations are hidden with probability λ and obfuscated by reporting 2×2 cloaking areas otherwise, and (2) semantic tags are hidden with probability λ and generalized by reporting the parent tag otherwise; the darker a box-plot is, the higher the amount of disclosed information is. In our experiments, we employed both 2×2 and 4×4 cloaking.

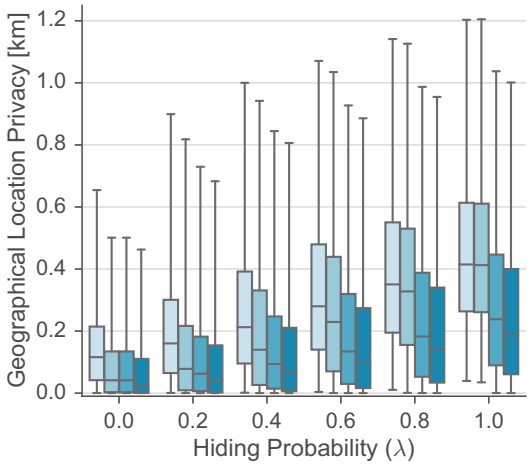
We observe that as we disclose more semantic information, along with the obfuscated geo-



(a) 2 × 2 Obfuscation w/o Sem. Background



(b) 4 × 4 Obfuscation w/o Sem. Background



(c) 4 × 4 Obfuscation w/ Sem. Background

Figure 5.8 – Geographical location privacy over different protection and learning scenarios.

graphical location (from left to right for each λ value), the median location privacy consistently decreases in all cases. For instance, it can be observed in Figure 5.8c ($\lambda = 1$), that disclosing the actual semantic tag decreases the median location privacy by 55% (from 420 m to 190 m) and disclosing the parent tag decreases it by 43%. Also, unsurprisingly, the privacy level increases as we increase the granularity of the location (*i.e.,* from 2 × 2 obfuscation in Figure 5.8a to 4 × 4 obfuscation in Figure 5.8b). Note that for $\lambda = 1.0$, the parent-tag generalization with hiding probability λ is exactly the same as hiding the semantic information completely and, similarly, it is exactly the same as the direct parent-tag generalization (*i.e.,* always disclosing the parent tag instead of the actual tag) for $\lambda = 0.0$. These can be observed in Figure 5.8.

We also analyze the effect of employing semantic background information (*i.e.,* the histories of users' transitions between semantic tags) in the inference process, in addition to the geographical background information that is already employed in all our experiments. We compare the two scenarios where 4 × 4 geographical obfuscation with hiding probability λ is used

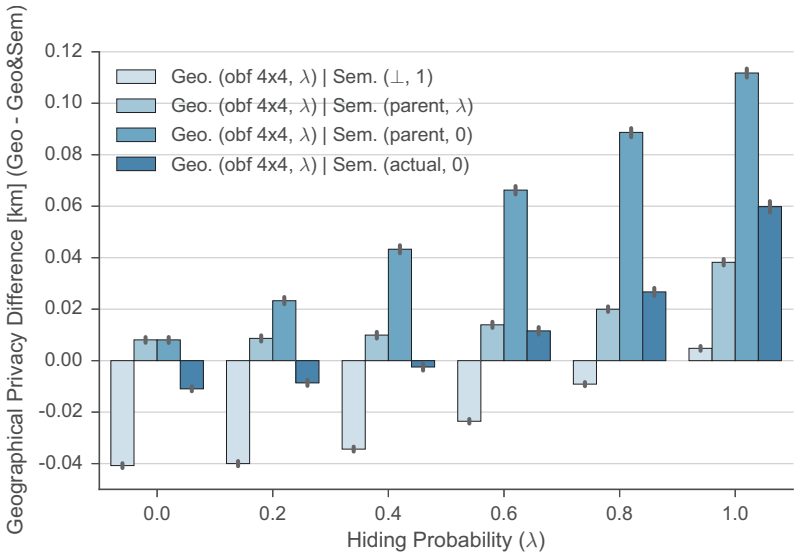


Figure 5.9 – Difference of geographical location-privacy levels between the cases *with* semantic background and *without* semantic background with 4×4 geographical obfuscation and varying λ. When users disclose some semantic information, the performance of the inference increase when using semantic background information about users. Interestingly, hiding the semantic tag of the user event results in the adversary being less successful when he uses the semantic background information.

(*i.e.,* Figures 5.8b and 5.8c, with and without semantic background information respectively). We observe that, for instance in the case of $\lambda = 0.4$, the median geographical privacy decreases when the adversary employs the semantic background information of users. This pattern is visible for most of the cases from *without* semantic background to *with* semantic background. It can also be observed that the semantic background information is very influential on geographical location privacy in the cases of direct parent-tag generalization and semantic disclosure (*i.e.,* the two darkest boxes). We notice that, in some cases (typically for the light case where the semantic information is hidden all the time), the adversary is more confused (and hence less successful) when he employs semantic background knowledge. The main reason for this outcome is that the adversary's knowledge on the semantic transitions of the user is less effective in his attack when the attacked traces' length is short. In general, we observe that employing semantic background knowledge in the inference helps the adversary increase his median accuracy by 10 to 115 meters when the users disclose some semantic information in their traces. This is clear in Figure 5.9, that shows the difference between Figures 5.8b and 5.8c (*i.e.,* the information gain of the adversary between the two scenarios). The reason why the adversary gains more information in the case of parent-tag obfuscation compared to no semantic protection is that when users disclose their semantic tags, their privacy level is already lower; hence the potential information gain of the adversary in *with* semantic background scenario is naturally lower.

Figure 5.10 depicts the average geographical location privacy in each of the six considered cities (with and without semantic background, aggregated over all values of λ and over the two

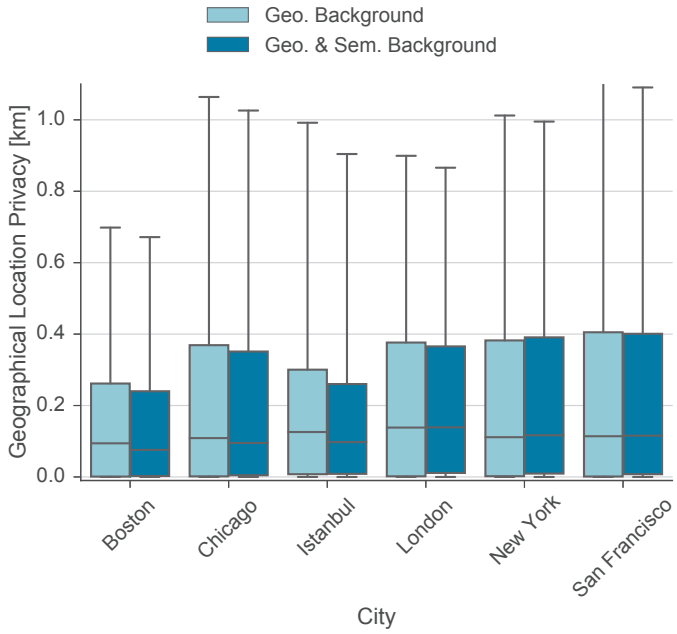


Figure 5.10 – Average geographical location privacy over all users in each considered city.

sizes of the cloaking area). It can be observed that it is quite comparable among cities: Despite the difference in terms of culture and urban planning, we did not observe major differences across cities in terms of user privacy in the presence of semantic information. It can also be observed that semantic background information improves the performance of the inference, thus decreasing users' geographical location privacy. Note that our experiments include some randomness and as a result, in some situations (e.g., New York) the background information slightly misleads the adversary.

Privacy vs. Utility Trade-Off

We now explore the trade-off between privacy and utility by evaluating both location privacy and utility for different levels of obfuscation. To comply with the experimental setup of [19], we consider four protection mechanisms by combining a low or high level of semantic obfuscation with a low or high level of geographical obfuscation as described in Table 5.5 and illustrated in Figure 5.11. We set the hiding probability λ to 0.2.

Table 5.5 – Description of the different obfuscation levels.

Obfuscation	Description
Ls-Lg	Semantic tag, 2×2 geographical region
Hs-Lg	Parent semantic tag, 2×2 geographical region
Ls-Hg	Semantic tag, 4×4 geographical region
Hs-Hg	Parent semantic tag, 4×4 geographical region

We plot the results in Figure 5.12. The points represent the average privacy and utility. It can be observed that the four points corresponding to the different obfuscation levels form

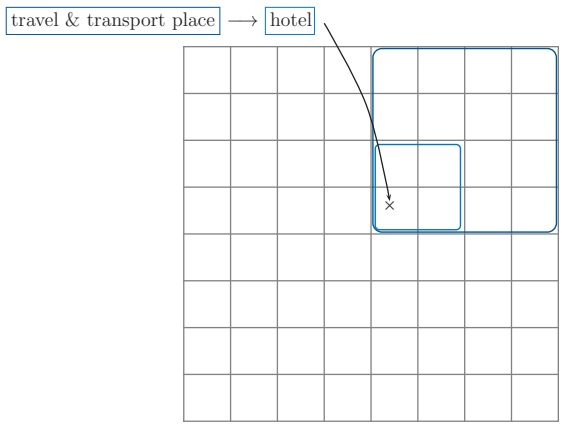


Figure 5.11 – Illustration of the obfuscation levels used in the experiments. Light blue frames denote low levels of obfuscations whereas dark blue frames denote high levels of obfuscation.

a diamond shape: Ls-Lg provides the highest level of utility and the lowest level of privacy; Hs-Hg provides the highest level of privacy but the lowest level of utility; Ls-Hg provides a better level of (location) privacy than Hs-Lg *and* a lower level of utility. This last observation is quite intuitive as geographical obfuscation is expected to protect location privacy better than semantic obfuscation and semantic obfuscation has been proved to be more detrimental to utility than geographical obfuscation has been [19]. This means that, as far as geographical location privacy is concerned, users should always prefer Ls-Hg over Hs-Lg. As for semantic location privacy (which we analyze in detail in the next sub-section), it can be observed that geographical obfuscation is quite beneficial as the use of high geographical obfuscation substantially increases the users' semantic location privacy at a cost of a small decrease in utility. In the case where low semantic obfuscation is used, the semantic location privacy is zero as the users reveal the actual semantic tags of their locations.

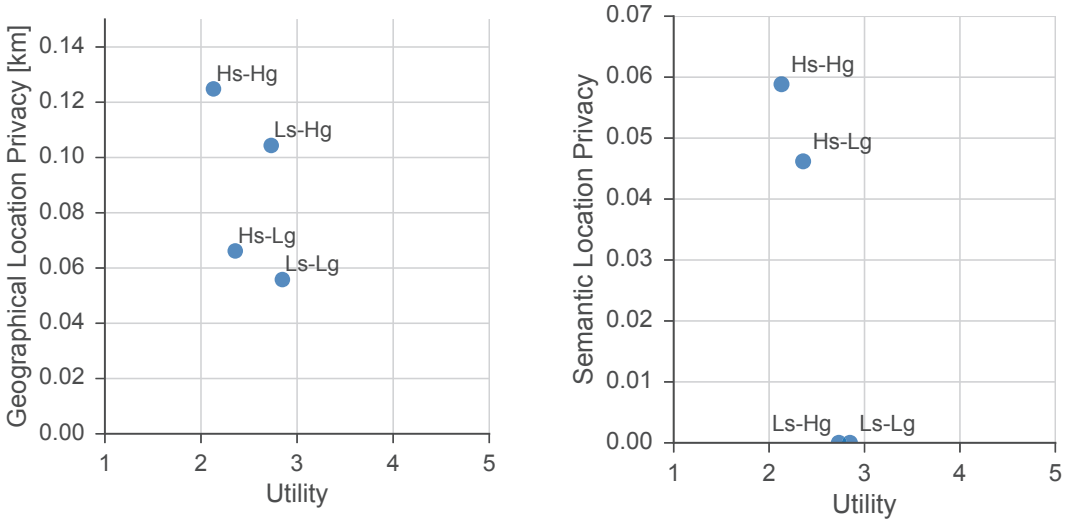
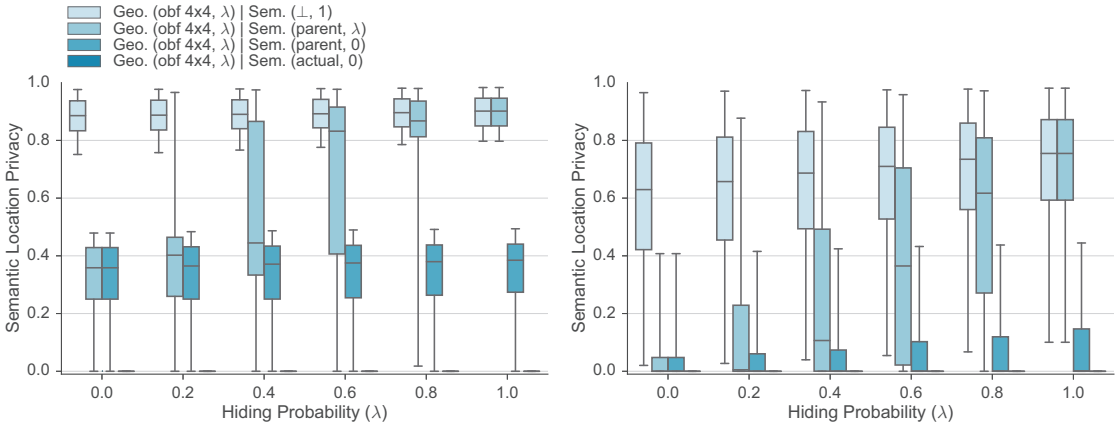


Figure 5.12 – Privacy vs. Utility in four different scenarios (Ls-Lg, Hs-Lg, Ls-Hg, Hs-Hg, for $\lambda = 0.2$).

Semantic Location Privacy

In this section, we evaluate the semantic location privacy and present the loss of privacy in the semantic dimension of location. As in the figures depicting geographical location privacy, we plot the aggregated privacy-level over all users, all simulation iterations and all user events using box plots. The semantic location privacy is calculated as the expected error of the adversary.



(a) 4 × 4 Obfuscation w/o Sem. Background

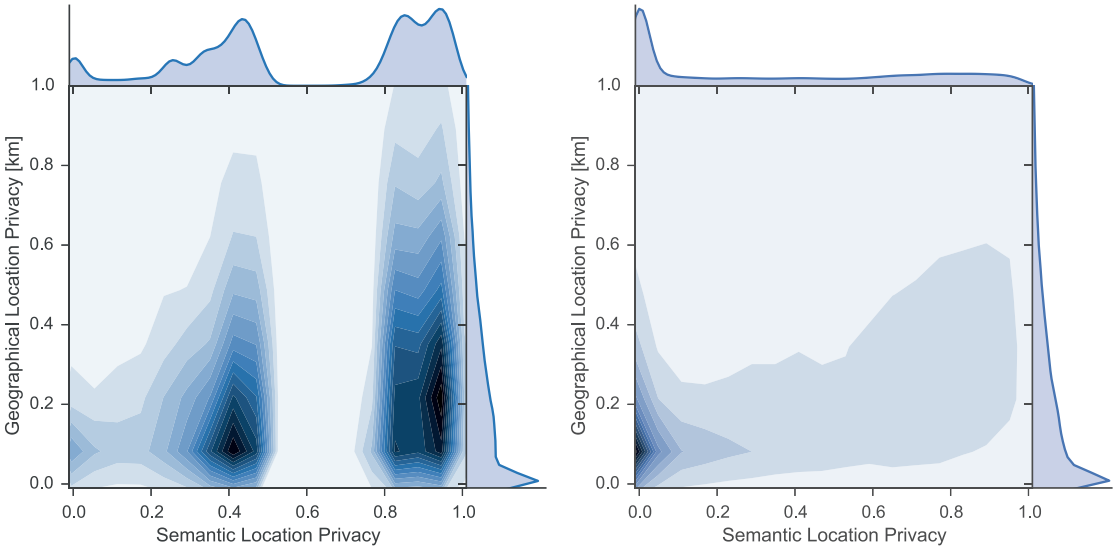
(b) 4 × 4 Obfuscation w/ Sem. Background

Figure 5.13 – Semantic location privacy levels over different protection scenarios with geographical and semantic background knowledge of the adversary.

In Figure 5.13, we present the semantic location-privacy results for 4×4 obfuscation with hiding probability λ in both 'Geographical background' and 'Geographical & Semantic Background' scenarios. In both cases (shown separately in figures 5.13a and 5.13b), as we protect the semantic information in the users' traces less and less (from the lightest boxes to the darkest ones), the semantic location privacy consistently decreases. We also observe that protecting the geographical location privacy more, *i.e.*, increasing the hiding probability λ, also helps increase the semantic location privacy in most of the cases. Whereas, semantic location privacy is naturally always 0 in the case of disclosing semantic information all the time. Moreover, unsurprisingly, when the adversary has semantic background information in addition to the geographical one, he learns more about the users' location semantics in his inference, *i.e.*, the semantic location privacy decreases. However, compared to the geographical dimension, this decrease in the semantic location privacy is more substantial as can be seen in Figures 5.13a and 5.13b: Even if the semantic tags of the user events are hidden all the time, the privacy loss is between 30-50%. The loss reaches up to 80% in other protection scenarios.

Lastly, we present the geographical and semantic location privacy jointly in Figures 5.14a and 5.14b, without and with semantic background information, respectively. These plots represent the density of the privacy levels over both the geographical vs. semantic location privacy (on the $[0,1] \times [0,1]$ planes depicted in Figure 5.14). The darker the plot gets, the more data points there are in the corresponding geographical and semantic intersections.

We exclude the scenario where the semantic tag of the events is always disclosed, because semantic location privacy is always 0 in this scenario, hence it does not contribute to these plots. These figures present the change in the relationship between the geographical and semantic location privacy. The obvious change occurs in the semantic dimension, though the change in the geographical location privacy is non-negligible as well. It can be observed, for instance, that users are somewhat clustered with respect to semantic location privacy. This corresponds to differences between cities in terms of venue distribution and diversity.



(a) Geo. Background

(b) Geo. & Sem. Background

Figure 5.14 – Geographical location privacy vs. semantic location privacy. Note that we excluded the case of 'Sem. (actual, 0)' as it provides no semantic privacy.

Analysis of the Effect of α

Finally, we present the results of our analysis of the effect of parameter α (used in the transition probabilities of the Bayesian model, see Equation (5.1)), in the case where the adversary has access to both geographical and semantic background information. In Figure 5.15, we plot the geographical location privacy obtained for different values of α (with mixed hiding probabilities). We observe that with increasing α (*i.e.*, prioritizing geographical information over semantic information), users obtain higher location privacy (i.e., the adversary is less successful) when they disclose the semantic tag. However, in the cases of hiding the semantic tags and parent-tag cloaking with hiding, the α value has less effect. This shows that an actual adversary could and should tune the model used in the attack, based on his observations, in order to improve the performance of the inference.

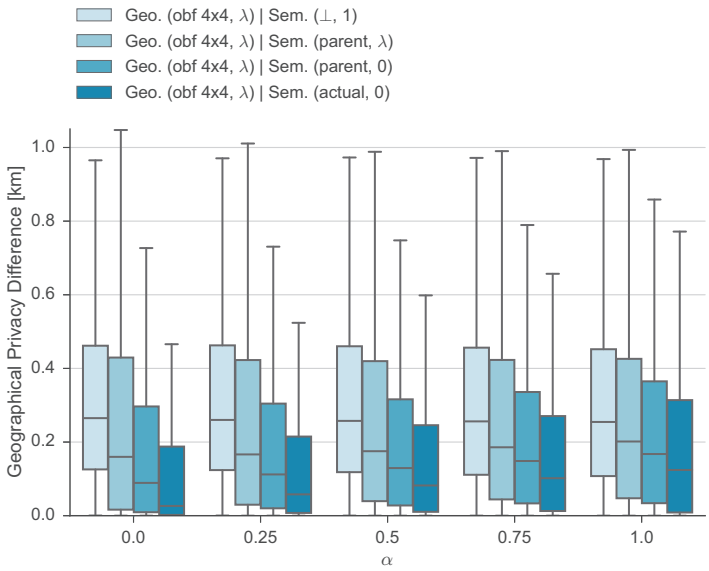


Figure 5.15 – Effect of parameter α on the users' geographical location privacy.

5.4 Discussion

In this chapter, we presented a semantic-aware location inference scheme, which we tested against several simple privacy-protection mechanisms (PPM), to prove that the threat on location privacy is more acute when the semantic dimension of location is taken into account. However, this is just a first step towards developing smarter PPMs, which take into account the semantic dimension of location privacy (together with the geographical dimension). The results we demonstrated in this chapter serve an important purpose: Understanding how to develop joint PPMs that protect geographical and semantic location privacy together and by taking into account user history and profiles. Our work in this chapter enables evaluation of such PPMs by paving the way for testing and adapting them w.r.t. the success of the adversary in an adaptive manner as well as optimizing jointly privacy and utility. As part of future work, we plan to use this framework to develop smarter PPMs. For instance, we intend to consider PPMs such as "If the cloaking area contains only one Burger joint opened at the considered time instant, either increase the size of the cloaking area or use the parent semantic tag, depending on which option brings the lowest utility loss".

A first limitation of this work is the fact that the adversary we considered uses a basic user behavioral model. As such, the results we present constitute a lower bound on the privacy loss: The adversary can actually strengthen his attack by increasing the complexity of the model he uses. For instance, he could exploit the temporal properties of locations and semantics: Users tend to have periodic routines (e.g., daily/weekly), such as staying home at night, going to work or school during the day and having lunch around noon, and venues have characteristic opening hours. By taking into account the time dimension, we could show that the threat is actually greater than what we demonstrate. Furthermore, the information we considered is in fact a subset of what a typical adversary (*i.e.,* a service provider) can collect. The fact that

the adversary has access to geographic and semantic profiles (i.e., background information) may be considered as rather strong. However, such knowledge can be built not only from obfuscated traces, but also by aggregating the data of several similar users, thus building more generic models (as done by Foursquare for next place recommendations).

A second limitation of this work is the size and the nature of the dataset: We considered "only" 1065 users (whom we have only little demographic information about) in six cities, who linked their Foursquare and Twitter accounts and made the tweets generated by Foursquare *public*. Such a sampling method could introduce a bias in the experimental results.

5.5 Related Work

A large amount of work has been devoted to quantifying location privacy, in particular when extra information (*i.e.,* different from location information e.g., co-locations and location semantics) is available to the adversary. [68] is one of the first papers to identify and study inference attacks on location traces. Another notable example, on which our work is partially built, is presented in [99, 100]. In these papers, the authors propose a formal framework to quantify users' location privacy when some (obfuscated) location information is available to the adversary. Their proposed framework relies on hidden Markov models for the location inference process and uses the expected error of the adversary as a metric for location privacy. The work presented in this paper enriches this framework by incorporating the rich semantic information increasingly disclosed by users on social networks. Note that Shokri's framework can be used *as is* to include semantic information by defining a location as a couple (geographical location, semantic location). This however, makes existing techniques for background construction inefficient due to the sparsity of the transition data (although many transitions go from one geographical region to another, the number of transitions from a couple (region, semantic tag) to another is significantly reduced). Also, recent work have shown that moving from Hidden Markov Models to Bayesian networks enables the adversary to take into account more complex information such as co-location [87]. The main differences between our work and Shokri et al.'s are (1) the use of general Bayesian networks to model users' behavior and (2) a two-step background construction (*i.e.,* first semantic, then geographical) to deal with sparse data. Similarly, but orthogonal, to our work, in [87], the authors study the effect of co-location information (e.g., Alice and Bob are at the same (unknown) location at 2pm) on users' location privacy.

On the front of location semantics, several works study the semantic dimension of location information (some of them in the context of privacy). Several works, including [69], [75], [70] and [72], address the problem of identifying the points-of-interest (POIs) users visit, based on location traces. Unlike our work, these works do not consider semantic information *reported* by the users. Hence, obfuscating semantic information is not directly possible. Barak et al. propose an anonymization technique based on semantic cloaking, that is, replacing actual coordinates by *personal* semantic labels such as 'home' (by opposition to the *universal* labels we considered, such as 'restaurant') [14]. Rossi *et al.* [93] benefits from semantic information

in user check-ins in Foursquare to infer user identities based on anonymized user visits and prior information. Some works extend existing location privacy metrics and definitions to take semantics into account. For instance, in [70], the authors propose a location-cloaking technique that ensures that the reported regions have a high semantic diversity in terms of the number of distinct venue types in the area. In [34], the authors propose the PROBE framework for implementing efficient, semantic-aware and personalized location cloaking. The concept of semantic diversity was originally formalized as l-diversity in [77] followed by related models including p-sensitivity [112], location diversity [116] and t-closeness [71]. Again, these works focus mostly on providing formal *semantic* location privacy guarantees by obfuscating *location information*, whereas our work considers both geographical and semantic information and investigates the privacy implications on both dimensions, based on statistical inference. Similarly, in [28], the authors extend the concept of *geo-distinguishability*, which applies differential privacy to location privacy [12], to take into account the semantic diversity of the reported locations. Differential privacy-based frameworks and inference-based frameworks are fundamentally different in their approach to privacy quantification. In [81], the authors propose the notion of C-safety, which not only takes into account semantics but also the sensitivity (in terms of privacy) of the different venue types. Using a taxonomy of venue types, the authors propose an efficient semantic-aware obfuscation mechanism. Our work distinguishes itself from existing works as it incorporates semantic information in the *inference* process to better recover the users' locations, thus demonstrating the sensitive nature and the associated privacy risks of semantic information.

Finally, complementary to our approach, in [19], the authors study the implications of geographical and semantic obfuscation (through generalization) of users' check-ins on their perceived utility; in the evaluation of our work, we make use of the predictive model proposed in this paper.

5.6 Summary

In this chapter, we have investigated the effects of location semantics on geographical location privacy of mobile users. We have considered two essential scenarios, specifically the case when an adversary, without knowing the semantic mobility patterns of the users, exploits the publicly available semantic information on locations, and secondly the case when the adversary knows the semantic mobility patterns of the users, in addition to knowing the location semantics. We have modeled the adversary that is aware of location semantics by using Bayesian networks and demonstrated that disclosing any level of semantic information on the visited locations improves his success.

In summary, both the geographical and semantic location privacy are at greater risk than revealed before, due to the multidimensional nature of location data. When designing privacy-protection mechanisms, our aim must be to protect location privacy on a multidimensional scale, *i.e.*, considering the types of locations.

6 Time-Aware Inference and Sensitive Protection

In the previous chapter, we demonstrated that an adversary can increase his success in obtaining a user's true location if the user discloses some level of semantic information about her location online. This is due to the regularity in people's lives, not only geographically but also semantically. Moreover, people tend to do similar things at regular schedules (e.g., they work during weekdays, have lunch around noon, etc.), which is not taken into account in the aforementioned adversary model.

In this chapter, we develop and present an extended Bayesian model that considers a user's daily habits as compared to a time-oblivious manner that was studied in the previous chapter. Specifically, we analyze our real dataset of geo-tagged tweets that include Foursquare check-ins from the previous chapter and show that the semantics of visited locations exhibit a time dependency. We exploit this aspect in order to study a more complex adversary. The results of our experiments show that exploiting time-dependency indeed helps an adversary gain more information, especially when the semantics are not disclosed at all, compared to a time-oblivious adversary.

Furthermore, inspired by the results from Part I of this thesis, we implement a sensitivity-aware and obfuscation-based protection mechanism and evaluate it in comparison to a static obfuscation mechanism with fixed size parameters. We refer to this protection mechanism simply as *sensitive protection* for the sake of brevity. We use the user's background information while determining the final sensitivity in a probabilistic manner and iteratively increase the obfuscation size if necessary by taking into account the granularity of the disclosed semantic information (e.g., disclosed, generalized, hidden). Note that, however, we do not implement a linkability graph in this scenario as we focus on sporadic event disclosure in this part of the thesis. Our evaluation reveals that a sensitivity and semantic aware approach to privacy protection provides a more preferable experience by keeping up with the desired geographical location-privacy levels of users while not overprotecting.

6.1 Preliminaries

As in the previous chapter, we consider users who use mobile devices to sporadically report their locations, sometimes annotated with a semantic tag. This is the typical behavior in online social networks, where users generate events on-purpose. In this context, the application can serve different purposes, for instance to post an opinion (e.g., on Twitter), or query a nearby location (e.g., from Foursquare), or to check-in (e.g., could be both Twitter and Foursquare). We assume that users may sometimes apply obfuscation or generalization on their locations and semantic tags (if any). Finally, an honest-but-curious adversary tries to infer the actual location and semantic tag of users based on his observations and background knowledge about them.

Users We modify and use the formalization from the previous chapter (Chapter 5) in this work. A user event $a_u(i) = (t, s, r)$ is a tuple where $t \in [0, 23]$ is time of day, $r \in \mathcal{R}$ is the geographical region user u is in and $s \in \mathcal{S}$ is the semantic tag associated to the user's visit in r (*i.e.,* the venue type). $a_u = \{a_u(1), a_u(2), \cdots, a_u(N)\}$ is user u's trace that consists of a total of N events. $\mathcal{R} = \{R_1, R_2, \ldots R_M\}$ represents the area of interest in an application, which consists of non-overlapping (potentially uniform) geographical regions, whereas $\mathcal{S} = \{S_1, S_2, \ldots, S_K, \perp\}$ denotes the set of all semantic tags used to denote the venue types in $\mathcal{R}$ including the non-annotated visits (*i.e.,* with $\perp$, we denote the case when a user visits a region r, but without a specific venue type). Users may generate different numbers of events independently from each other. Finally, we do not consider interactions among users.

Note that since we consider semantically annotated events to be generated by users in a sporadic manner (*i.e.,* on-demand), the context in which these events are published generally requires the users to log in. This means that anonymization techniques are not easily applicable. Even if the service providers are convinced to modify their systems to enable anonymization of users, the anonymization techniques were shown to be not sufficiently effective [52, 58]. Moreover, as discussed in State of the Art (Chapter 1), many anonymization techniques require additional infrastructure (e.g., a trusted third party) or communication costs.

Protection Mechanisms A protection mechanism f_u maps an actual user event $a_u(i) = (t, s, r)$ to an obfuscated event $o_u(i) = (t, s', r')$ with a probability, where $r' \in \mathcal{P}(\mathcal{R})$[1] is an obfuscated geographical location and s' is some generalized semantic tag w.r.t. some rule. Typically, a semantic tag can be generalized by either replacing it with its parent or extending it with all its siblings in a preset tag taxonomy, e.g., Foursquare's hierarchical categories [3]. As it can be seen, we do not consider time obfuscation in our formalization, but our modeling does not restrict it. Lastly, as individuals may have different privacy requirements [104], *i.e.,* sensitivities, in different places and types of places, a protection mechanism f_u may potentially consider users' privacy sensitivities w.r.t. geographical regions or semantic tags,

[1] $\mathcal{P}$ is the power set.

hence be personal to each user.

Adversary The adversary is, as in the rest of this thesis, an honest but curious entity, typically the service provider. We assume that the adversary already has access to user history, which may be incomplete. He may also know which protection mechanism is employed by the users. He observes the obfuscated user traces $o_u = \{o_{u_1}, o_{u_2}, \cdots\}$ and tries to infer the actual user traces $a = \{a_{u_1}, a_{u_2}, \cdots\}$ from them by using the knowledge he has, *i.e.,* the history and the protection mechanism $f_u$with some accuracy.

Privacy metrics In order to evaluate an adversary's attack effectiveness (which is also the ineffectiveness of a protection mechanism), we need metrics that take into account the correctness of the adversary's inference and his confidence, as discussed in the previous chapters as well. We compute the expected error of the adversary both in the geographical and semantic dimensions as was done in the previous chapter. The base formulas to compute the privacy levels for a user event $a_u(i) = (t, s, r)$ is as follows (identical to those in Chapter 5):

$$\mathrm{GP}_u(i) = \sum_{m=1}^{M} q_g(R_m, i) \cdot \mathrm{dist}^G(R_m, r), \tag{6.1}$$

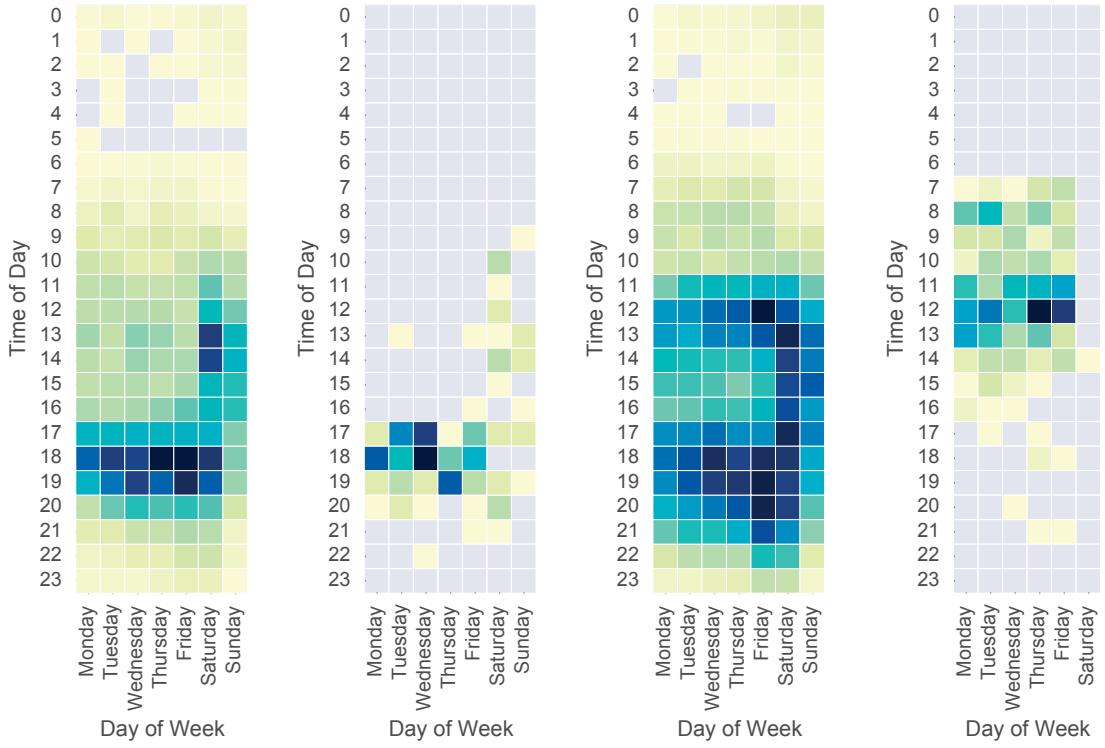
$$\mathrm{SP}_u(i) = \sum_{k=1}^{K} q_s(S_k, i) \cdot \mathrm{dist}^S(S_k, s), \tag{6.2}$$

where $\mathrm{GP}_u(t)$ (resp. $\mathrm{SP}_u(i)$) is the geographical (resp. semantic) location-privacy of user u for event i. q_g and q_s are the marginal distributions for regions and semantic tags as a result of the adversary's attack. In our experiments, we use Euclidean distance for regions (dist^G) and a tree-based graph distance for semantic tags (dist^S). More specifically, we use the distance metric $d(\cdot)$ from graph-theory (*i.e.,* the length of the shortest path between two nodes) on the Foursquare category tree [3] to compute the semantic distance between two tags. We normalize the shortest-path distance between two tags by the sum of the tags' depths (*i.e.,* their respective distances to the root). Please refer to Section 5.2.2 and Equation 5.4 for details.

6.2 Time-Aware Inference

Previously in Chapter 5, we modeled user mobility with the assumption that people first decide what type of place they want to go to and then the location. We did not, however, include the time dimension of the visits in the model as our purpose was to demonstrate that disclosing some level of semantic information decreases the geographical location privacy. We still keep the same assumption on human mobility, but consider that people exhibit patterns in terms of location and semantics in the same time period of a day. Taking into account these patterns, we model our adversary –hence the inference process– accordingly. To validate our assumption, we have analyzed Foursquare check-ins (see Section 6.4.1 for the dataset).

Figure 6.1 shows examples that support the aforementioned idea on the time-dependency of



(a) Collective (b) User A (c) Collective (d) User A

Figure 6.1 – Visit distributions for the Foursquare categories *Arts & Entertainment* in (a) and (b), and *Food* in (c) and (d), showing the different behavior patterns in the city of London collectively (*i.e.*, of all users combined) with comparison of (a) to (c) and by one user with comparison of (b) to (d). The visit concentration goes from light to dark, dark representing the more concentrated cells.

user behavior. The heatmap plots represent the visit distribution over days of week and time of day for users in a dense area of the city of London. Figures 6.1-a and 6.1-c show collective distributions from all users on categories *Arts & Entertainment* and *Food*, respectively. Figures 6.1-b and 6.1-d show distributions on the same categories, respectively, but only from a user A in the same region. For user A, places labelled as Arts & Entertainment are visited mostly on weekdays at around 17:00 - 19:00, and also on Saturday throughout the day. The same user visits (and checks-in at) Food places dominantly during weekdays around noon. The collective behavior shows that people in London go to Arts & Entertainment places throughout the week, including Saturday, and dominantly on the evenings. Additionally, Saturday and Sunday noons also appear to be attractive for visiting such places. On the other hand, places in the Food category are visited quite diversely, yet the clear pattern is that during the weekdays: they are visited dominantly around noon and between 18:00 to 20:00, corresponding to lunch and dinner times. There is more continuity on Saturday and Sunday, which is most probably the result of non-working days for most people and hence less restriction on lunch times. Last but not least, in all plots, we observe that Sunday is the day with the least amount of check-ins leading to the idea that people are less active on Sundays. These examples support our

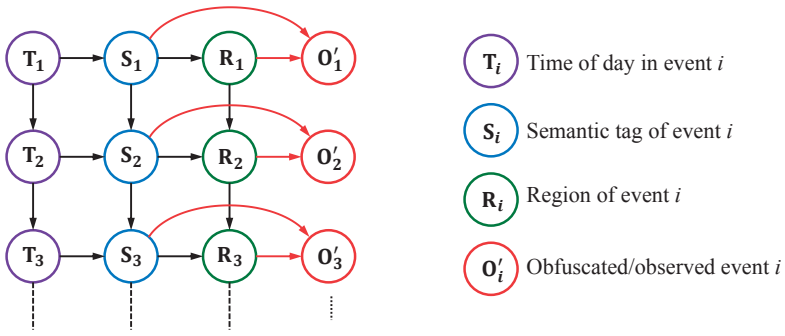


Figure 6.2 – The Bayesian network that models the daily behavior of a user. A daily model is dynamically generated by the adversary upon observing a day of events from a user.

intuition of including the time dimension in user modeling from an adversary's perspective.

Another observation on Figure 6.1 is that visits to a specific type of place have a dependency on time of day, though it is not strict. In other words, a user may not visit a type of place always at the same time of day, but on a variable interval. With this in mind, we model a time-aware adversary for which we generate Bayesian networks for weekdays and weekends separately in an observed user trace o_u. We attack each observed day in o_u independently since causality between successive events in a single day is more likely than across days. As a result, each generated Bayesian network is dynamic (*i.e.,* with varying number of events) and specific to one day (but with knowledge on the whole weekdays or weekend based on what day it is). Figure 6.2 presents a sample Bayesian network with random variables for an event i as $\mathbf{T}_i$, $\mathbf{S}_i$, $\mathbf{R}_i$ and $\mathbf{O}_i$. As it can be observed, the time variables $\mathbf{T}_i$ depend only on the time of day of the previous event (if any). Semantic variables $\mathbf{S}_i$ depend on the time of day of the event and also previous semantic tag. Region variables $\mathbf{R}_i$ depend both on $\mathbf{S}_i$ and $\mathbf{R}_{i-1}$ as a result of our core assumption that people first determine the type of place they want to go to. Finally, the observed event variable $\mathbf{O}_i$ depends on $\mathbf{S}_i$ and $\mathbf{R}_i$, which corresponds to the *probabilistic* PPM f_u that takes as input the semantic tag and region. f_u's output, *i.e.,* the tuple (s', r'), is the evidence for $\mathbf{O}_i$ where s' is the (potentially) generalized semantic tag and r' is the (potentially) obfuscated region. $f_u = \mathbf{Pr}(\mathbf{O}_i = (s', r') | \mathbf{S}_i = s, \mathbf{R}_i = r)$ may or may not be correctly known to the adversary as it might be personalized to user u and hence have some private parameters such as sensitivity levels for different regions and tags.

Except $\mathbf{T}_1$ —the first observed event's time of day— in a Bayesian network, all the variables take value within their domains w.r.t. conditional probability distributions. In our context, we assume that the adversary may have access to background information and hence tries to approximate these distributions with some degree of accuracy.

For inference, we set the observed event i's value (*i.e.,* $\mathbf{O}_i$) to (s', r') from $o_u(i) = (t, s', r')$. Note that $\mathbf{O}_i$ is not directly dependent on $\mathbf{T}_i$ in the Bayesian model and hence does not contain the time of day t, yet the adversary knows at what time the event takes place. This information must be used on $\mathbf{T}_i$. However, as we noted before, a user may visit some place around the

same time of day but not necessarily always at the exact same time. Hence, instead of strictly setting the evidence for $\mathbf{T}_i$ to t in $o_u(i)$, we opt for setting the domain of $\mathbf{T}_i$ to a time range, e.g., $[t-2h, t+2h]$. This automatically considers a variance in timestamps among successive visit patterns as well, for instance going for lunch first around noon and afterwards visiting a coffee shop in the afternoon. The domains of the remaining semantic and region variables can be set to $\mathcal{S}$ and $\mathcal{R}$, respectively, which are the targets of the adversary. The marginal distributions with these settings are obtained by running inference algorithms on the Bayesian network, such as Junction-tree algorithm [62] or belief propagation [88].

6.3 Sensitivity-Aware Protection

We implement a sensitivity and history aware PPM that automatically determines the size of an obfuscation area for a given user event; we call it the *sensitive* PPM for short. In principle, this PPM is similar to the one we implemented in the Location-Privacy Library (see Chapter 4, [4]) with a different approach to processing the sensitivities assigned by the user. Likewise, we let users set sensitivity levels for semantic tags and regions in this PPM as well. The difference is that the sensitive PPM uses these settings in order to obtain an expected sensitivity level instead of using an individual sensitivity level from the user's sensitivity profile. The desired geographical location-privacy is then defined as the expected sensitivity multiplied by the *maximum* desired geographical location-privacy Θ. The exact formula for computing the expected sensitivity $E.S.$ is as follows:

$$E.S. = \sum_r \mathbf{Pr}(\mathbf{R}_i = r) \times \begin{cases} Sens(r) & \text{if } Sens(r) > 0 \\ \sum_s \mathbf{Pr}(\mathbf{S}_i = s) \times Sens(s) & \text{otherwise} \end{cases} \tag{6.3}$$

$$D.P. = E.S. \times \Theta, \tag{6.4}$$

where $Sens(.)$ is the sensitivity level for semantic tag s or region r, and $D.P.$ is the desired level of geographical location-privacy. In this formula, we prioritize any geographical sensitivity setting w.r.t. the semantic sensitivities. It means that if a user sets a sensitivity level for a specific region r, any other sensitivity level corresponding to the semantic tags within region r is overridden. Such a scenario can occur for home and work addresses of an individual, which may have very high (or low) sensitivity levels regardless of the semantics.

In summary, the PPM takes as input the semantic tag and region pair (s, r) as well as the generalization level for s, *i.e.,* the semantic privacy-protection level. Then the geographical obfuscation is determined iteratively by calculating and comparing the estimated geographical location-privacy and $D.P.$. Whenever $D.P.$ is higher than the estimated location geographical location-privacy, we increase the obfuscation size. The reason behind this is that, according to Bilogrevic *et al.* [19], people tend to value the semantics more when valuing the utility they get. Thus, our algorithm is semantic-driven. Note that the maximum protection is to hide the semantic tag/region completely. However, we do not hide both at the same time, *i.e.,* whenever the semantic protection is set as 'hide', we limit the geographical obfuscation

to a maximum area size set as a parameter of the PPM. The idea is that users usually aim to have a utility either on the semantic or the geographical dimension when generating their events sporadically, for example to 'inform about activity', 'appear cool', 'inform about the location', 'get a reward', *etc.* [19]. As a result, it is not logical to hide all the data, otherwise it kills the whole purpose of the application. It also means that there is a trade-off between the semantic and geographical location-privacy levels. The pseudo-code algorithm of this PPM is given in Algorithm 2.

Algorithm 2: Sensitive Protection Mechanism

```
Input: s, r, semantic_protection
s_observed = semantic_generalization(s, semantic_protection);
r_observed = r;
α = 0;
while desired_privacy_level(s_observed, r_observed) > expected_privacy_level(r_observed, r) do
    α += 1;
    if α == max_obfuscation && semantic_protection == 'Hidden' then
        break;
    else
        r_observed = geographical_obfuscation(r, α);
return (s_observed, r_observed);
```

For the semantic location-privacy protection, our algorithm considers three protection levels:

- Disclose: the semantic tag of the user event is not protected. It is disclosed as it is.
- Parent: the semantic tag of the user event is replaced by its parent in a tag taxonomy to decrease the amount of semantic information disclosed.
- Hidden: the semantic tag is hidden altogether and nothing is disclosed regarding the semantic tag of the user event.

For the case when the semantic tag is hidden, as mentioned above, we do not apply hiding on the geographical region even if the desired level of privacy is not met. Computing the expected geographical location-privacy is straightforward: the user (or her device) is aware of her history (*i.e.,* visit counts), thus can compute the probability distributions over $r \in \mathcal{R}$ given $o_u = (t, s', r')$. Using these distributions we calculate and compare the expected level of privacy to the desired level of privacy in order to determine the final size of the obfuscation.

Note that when a user wants to disclose her location's semantic tag, her location sensitivity may be lower than the case when she hides the semantic tag. This is especially true if a user sets a sensitivity level based on a semantic tag. If the semantic tag is protected, *i.e.,* generalized or hidden, and if there exists other semantic tags with lower sensitivity levels in the obfuscation area, then the expected sensitivity level may be lower than the set sensitivity level for the current semantic tag of the user event. This results in a lower desired protection level and

hence a lower geographical location-privacy. Consequently, with this protection scheme we introduce a trade-off between semantic and geographical location-privacy levels. Lastly, this scheme can be extended by letting users determine whether their sensitivities are also time-dependent, for instance going to a bar in the evening may not be sensitive for most people, but it might be during morning.

6.4 Evaluation

We experimentally evaluate geographical and semantic location-privacy w.r.t. our inference scenario and also compare the effects of our sensitive protection mechanism to a static geographical obfuscation. We use real user traces whose majority is semantically annotated (*i.e.,* more than 50% of the events in each trace).

6.4.1 Dataset

We benefited from the same dataset as in the previous chapter. Namely, it is a dataset where geo-tagged tweets from Twitter are joined with Foursquare venues. We matched the tweets and Foursquare venues through the publicly tweeted Foursquare check-ins by users having an account on both social networks. The dataset was collected from January 2015 until the end of July 2015 (please refer to Section 5.3.1 in Chapter 5 for the explanation of how the dataset was collected). We filter the data based on certain criteria in order to have user traces that contain Foursquare check-ins with at least 50% proportion. We further limit the geographical areas we run our experiments on due to computational limits; more specifically, to avoid domain explosion of the random variables in our Bayesian network model. We chose users with trace length of at least 70 in order to avoid data sparsity: shorter traces would not provide enough data for the adversary. The number of users we evaluate in our experiments from six big cities (Boston, MA, USA; Chicago, IL, USA; Istanbul, Turkey; London, UK; New York, NY, USA; and San Francisco, CA, USA) is 690. The areas of interest from these cities are of size 2.4×2.4 km^2.

6.4.2 Experimental Setup

We partitioned each area of interest in our dataset into 12×12 cells (of size 200×200 m). Coordinates from each event in a user trace are mapped to one of these regions, and we used the center point of the regions as the coordinates in our distance calculations. As we rely on Foursquare check-ins obtained through Twitter, we use the Foursquare category tree for semantic generalization. This means that with different sources of location semantics and tag taxonomies, we may obtain different results, but we do not expect the main findings to change drastically in the case of sensitive protection, because the sensitivity levels with the

[2]The essential differences from the filtered dataset used in the previous chapter are the area size and the minimum trace length criteria. This results in a richer dataset required for validating our approach that is based on daily behavior.

new semantic system should automatically adapt.

We implemented our adversary and the daily Bayesian model in Python using the Bayesian Belief Networks library provided by eBay [2]. We used the junction-tree algorithm [62] to optimally infer the final marginal distributions on semantic and region variables with the evidence set from the observed traces.

For the background information of the adversary, *i.e.,* the presence and transition probabilities used in the Bayesian model, we count all the relevant occurrences of events or transitions and apply Laplace smoothing [78] for ensuring non-zero probability for all possible values of random variables. Finally, we normalize the outcome histogram to determine the probability distributions.

As for the time variables in our Bayesian networks, we determine a range to focus on instead of evidencing them on the observed time in the user events, as discussed in Section 6.2. Based on some preliminary investigation we did for various ranges (from ±0 to ±5 hours), we observed that the adversary is most successful when we set the time domain to $[t - 2.5\text{h}, t + 2.5\text{h}]$[3] (*i.e.,* ±2.5 hours). As a result, we use this time range for time variables $\mathbf{T}_i$ in our attacks.

Protection mechanisms We implement two types of privacy-protection mechanisms (PPMs): our sensitive PPM and also a static PPM that generates obfuscation areas of fixed size (which is used in the previous chapter as well). Both are joint protection mechanisms that take a semantic tag, a region, and the amount of semantic protection as input and output a tuple (s', r'). The static PPM also takes the obfuscation size as input, which stays fixed throughout a user trace. In each mechanism the geographical obfuscation is the same, meaning that with a certain size, they will generate the same obfuscation area. We use 2×2 and 4×4 obfuscation sizes in our scenarios, dynamically in the sensitive protection and as a fixed parameter in the static protection. As a result, we have three different geographical protection scenarios:

- Sensitive Protection: the mechanism automatically decides the size of the geographical protection which can be: no protection, 2×2 obfuscation, 4×4 obfuscation or hidden.
- Static Protection with 2×2 obfuscation for all users and events independent of the semantic tag
- Static Protection with 4×4 obfuscation for all users and events independent of the semantic tag

The semantic generalization cases are also three as stated in Section 6.3: 'Disclosed', 'Parent' and 'Hidden', which results in a total of nine combinations of scenarios to run our experiments in.

[3] Obviously, this range yields successful results for the dataset we use, but with different datasets, a different setting may need to be identified for improved inference.

For the sensitivities, we have not come across a real dataset for individuals' privacy sensitivities/preferences w.r.t. location semantics that we can use in conjunction with the Foursquare categories tree. This subject requires its own research based on extensive user studies which is beyond the context of this thesis. For the time being, we rely on our small scale survey of the scientists involved in this work and determine an example sensitivity profile used in our experiments. This profile is presented in Table 6.1. Note that we did not set any sensitivity level for a region as this would be too arbitrary to rely on. The values we use for the semantic tags are nevertheless perceivable.

Table 6.1 – Privacy sensitivities used in the experiments.

Tag	Sens.	Tag	Sens.	Tag	Sens.	Tag	Sens.
Prison	1.0	Doctor's Office	0.9	Hospital	0.9	Medical Center	0.9
Embassy	0.9	Bank	0.8	Nightclub	0.8	Police Station	0.7
Hotel	0.4	Bar	0.4	Restaurant	0.3	Cemetery	0.3
Fast Food	0.3	Office	0.3	University	0.2	Train Station	0.15
Zoo	0.1	Stadium	0.1	Post Office	0.1	Museum	0.1

Cross Validation We adapt a cross-validation approach in our experiments: we slice user traces into chunks of 13 to 17 events (based on the length of each user trace independently) and use each chunk as a test trace while using the remaining chunks as training data for this test case. For example, if a user trace consists of 100 events, and it is sliced into 5 chunks, then we run 5 different experiments with each chunk as test trace to attack while using the 4 remaining chunks to train the Bayesian networks. Each test trace is obfuscated before running the attacks. If the adversary does not have any knowledge on user behavior for a particular day of week (e.g., Wednesday), then we use the aggregated data of the user over all week. In such cases, the adversary would not have enough information to decide in favour of any of the other days. Figure 6.3 shows this experimental framework visually.

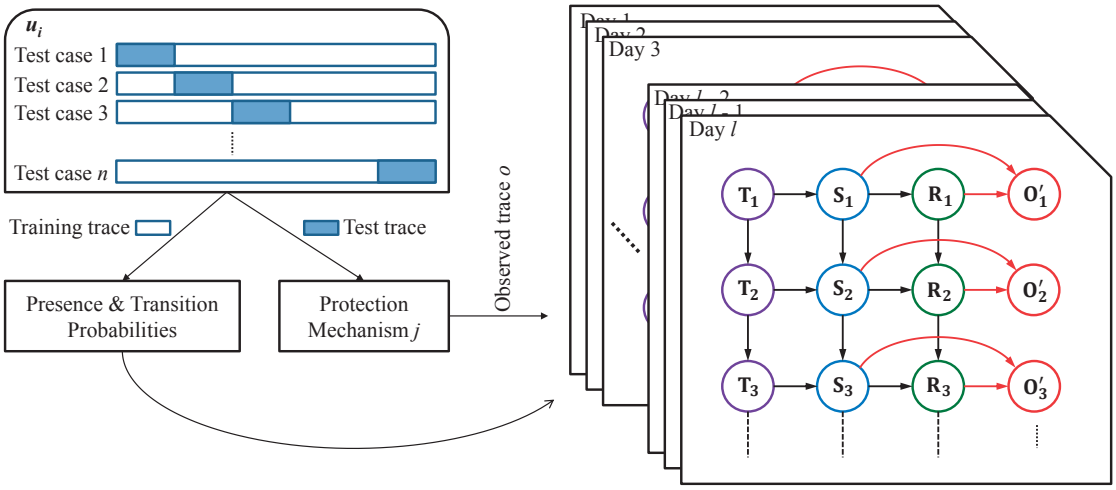


Figure 6.3 – Our approach on inferring location-privacy in a time and semantic aware way in a nutshell.

6.4.3 Results

We present our experimental results on time-aware inference and the proposed sensitive PPM in this section.

Effect of Time-awareness on Inference

Figure 6.4 shows the average geographical location-privacy levels over time of day for various semantic and geographical protection scenarios w.r.t. the *time-oblivious* and the *time-aware* attacks. From left to right, the plots represent results with 2×2 static protection, 4×4 static protection and sensitive protection employed for geographical location-privacy. From top to down, plots refer to different semantic protection scenarios in the order: actual tag, parent-tag generalization and hidden. The first observation is that time-aware attack outperforms the time-oblivious one at inferring the geographical locations of users when the semantic tags are hidden, for all 3 geographical protection scenarios. Also, the time-aware attack outperforms the time-oblivious one noticeably around morning. As we will discuss in the next subsection, this is due to the various activities of users that exhibit regularity. Conversely, the time-aware attack fails to outperform the time-oblivious one (though statistically insignificant) when the semantic tags are generalized by their parent tags with 4×4 static protection on regions. The time-oblivious attack slightly outperforms the time-aware attack in the afternoon and evening. This happens due to the noise introduced by unusual user activities, *i.e.*, some users occasionally leave their usual pattern and visit places that they do not usually go (geographically, semantically, or both). Apparently, when they disclose reduced amount of semantic information about their locations, an adversary is misled by his background knowledge on them. This is emergent also in Figure 6.5 (which shows the same setting for semantic location privacy) for all geographical PPMs with parent-tag generalization. When the semantics are hidden completely, however, the adversary is most successful at exploiting the time dimension on semantics for both geographical and semantic location-privacy. Not surprisingly, when the semantic tags are disclosed to the adversary, the time dimension does not play much role in the inference of geographical locations. The geographical location privacy levels as a result of both attacks in the first row of Figure 6.4 are almost perfectly aligned.

Performance of Sensitive PPM

Now we analyze and discuss the experimental results on sensitive PPM in detail and how it performs as compared to the static PPMs. Note that since the trends in the results are same for both time-aware and time-oblivious attack scenarios, we only present the results with only the time-aware attack scenario in this subsection.

We plot the results of the evaluation of the geographical location-privacy in Figure 6.6a. The figure plots the average geographical location privacy (the average error of the adversary inference in km.), compared to the desired location-privacy, established by the sensitivity

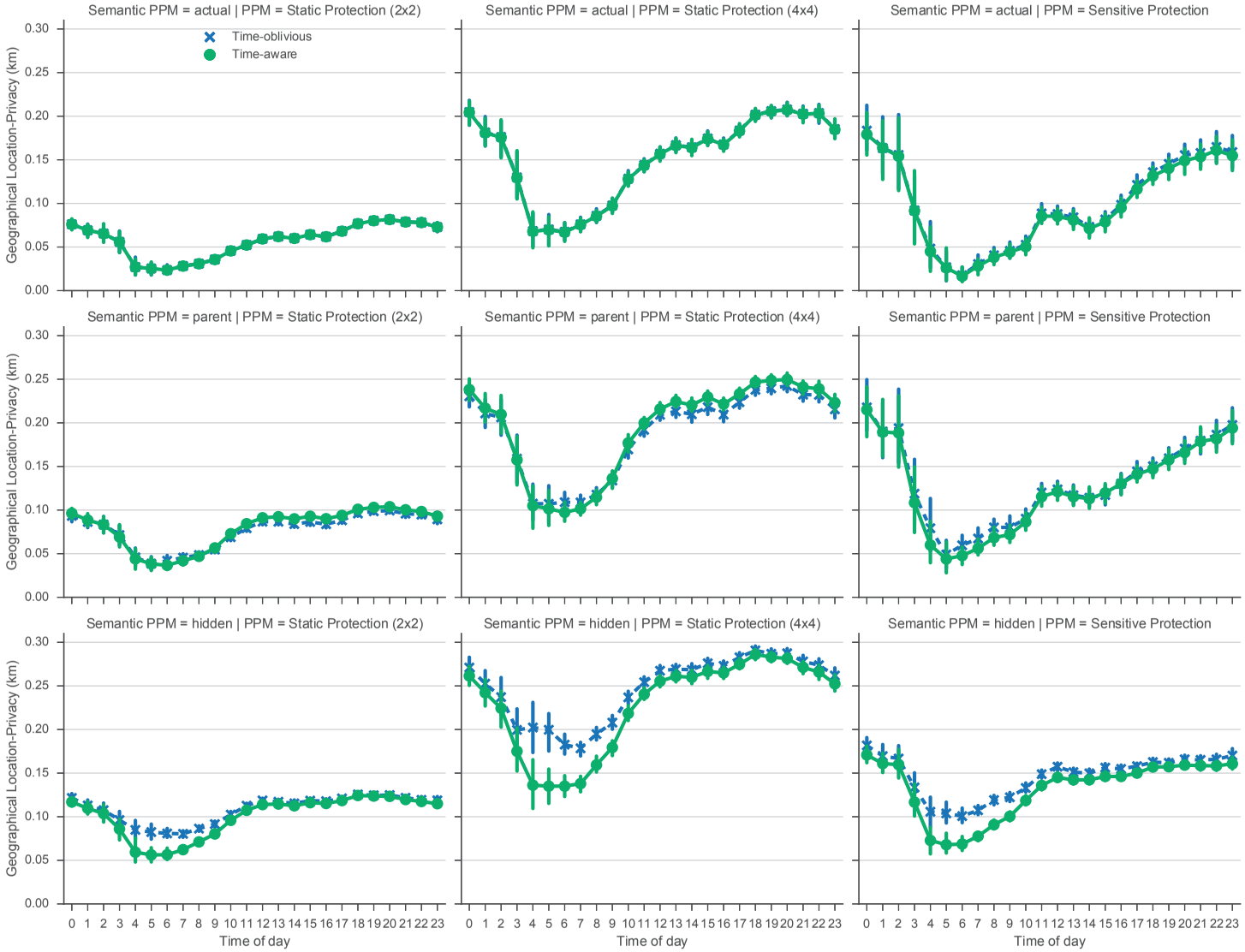


Figure 6.4 – Average geographical location-privacy (with confidence intervals) over time of day for all combinations of privacy protection scenarios w.r.t. time-aware and time-oblivious attacks.

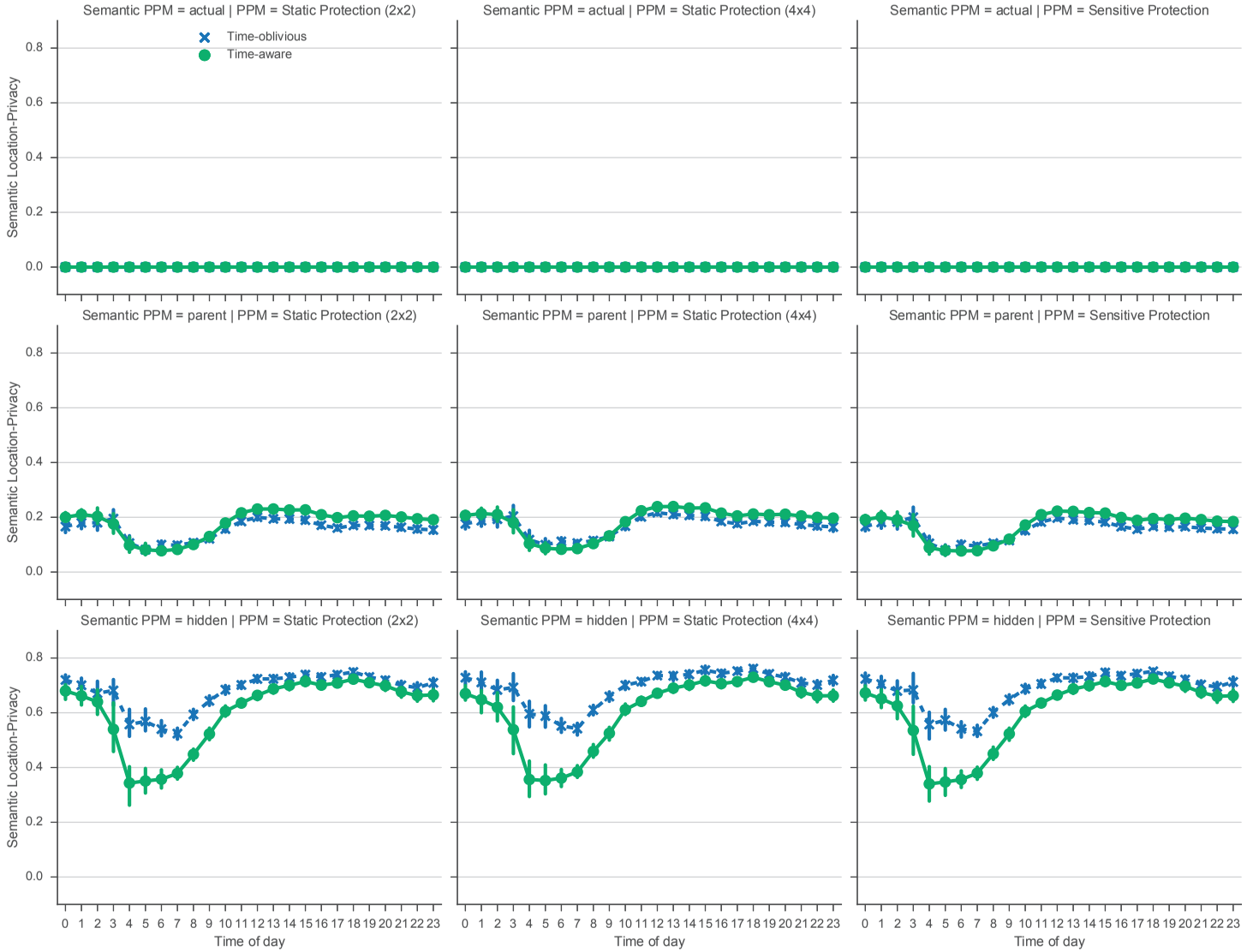
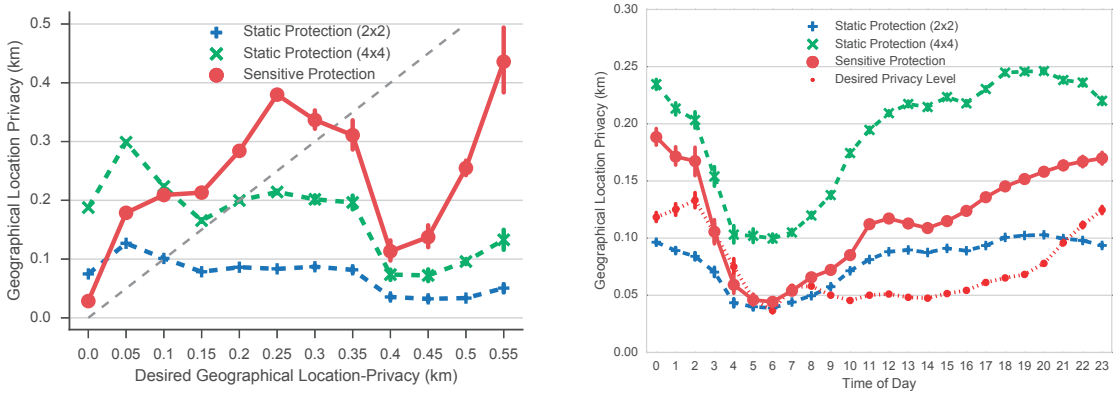


Figure 6.5 – Average semantic location-privacy (with confidence intervals) over time of day for all combinations of privacy protection scenarios w.r.t. time-aware and time-oblivious attacks.



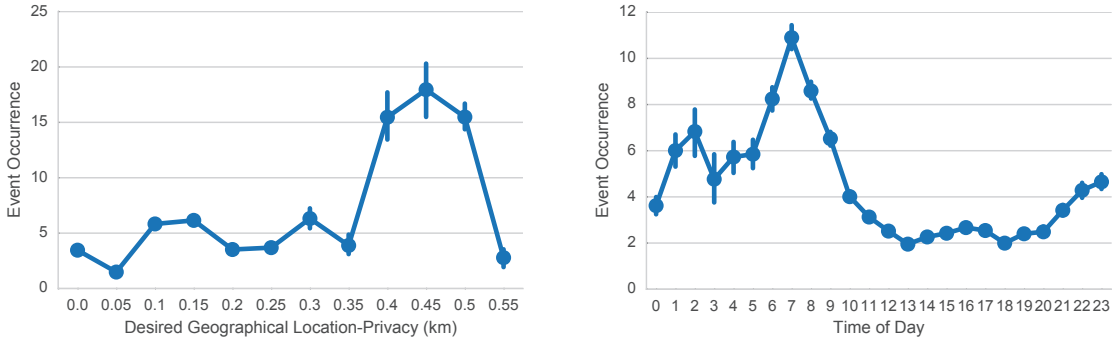
(a) Over the desired level of privacy

(b) Over time of day

Figure 6.6 – Average geographical location-privacy of user events over (a) desired privacy level and (b) time of day.

profile. We plot both the static and the sensitive PPMs. For the static PPM, we plot the results for both a 2×2 and 4×4 obfuscation scheme. The ideal protection level is plotted as a linear dashed segment. As it can be seen, for from low to medium desired privacy levels the sensitive protection mechanism is able to cope with the sensitivity requirements of the users. However, for very high sensitivity levels the mechanism is not able to keep up to the expected level of privacy. One of the explanations for this behavior is the fact that the number of event occurrences at places with high level of privacy is also considerably higher. The distribution of events can be seen in Figure 6.7a, and there is a noticeable peak for events that have a high desired privacy level, which in turn provides the adversary more information to predict the user location more accurately. Nevertheless, these results also show that the sensitive protection outperforms the static protection in most cases, even for the 4×4 obfuscation, and often with a significant difference.

In order to understand the time-variability of the PPMs, we plot in Figure 6.6b the static and sensitive PPMs over the time of the day (*i.e.,* from 0 to 23h). It is visible that the average desired

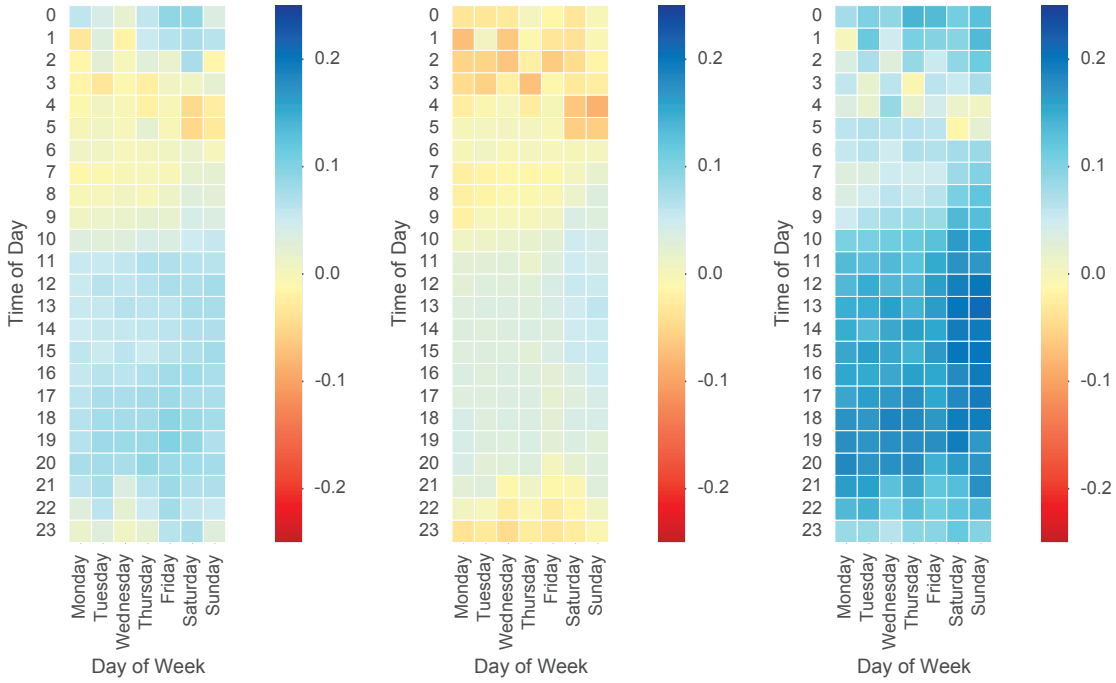


(a) Over the desired level of privacy

(b) Over time of day

Figure 6.7 – Average number of occurrences of the observed events in the adversary's background knowledge over (a) the desired privacy level and (b) time of day.

location-privacy increases during night time (between 10pm and 3 am). When we analyze further to see if this is related to the frequency of events w.r.t. time (see Figure 6.7b), it is clear that the number of events known to the adversary per observed event actually increases starting at 21pm and stays relatively high (compared to afternoon) until morning. As we can see, the 4×4 static obfuscation always provides a higher protection level on average over time-of-day than the sensitive approach (knowing that for higher desired privacy levels, 4×4 static obfuscation may fail). However, this is in reality a case of overprotection, as the 4×4 obfuscation is overprotecting the user location, which could potentially degrade the overall utility of the application.



(a) Sensitive Protection (b) Static Protection (2x2) (c) Static Protection (4x4)

Figure 6.8 – Average of the desired location-privacy subtracted from geographical location-privacy plotted over time of day and day of week for different geographical protection scenarios. 0 means the desired privacy level is met. The sensitive protection approach meets the desired privacy level almost all the time while not overprotecting.

In order to better visualize this effect, Figures 6.8a, 6.8b and 6.8c show heatmaps of average geographical location-privacy minus the desired location-privacy over time of day and day of week, for different geographical protection scenarios. We observe that the sensitive protection manages to meet the desired privacy level on average almost all the time while not overprotecting. In the case of 2×2 static obfuscation, we see that the desired privacy level cannot be met more and more, especially during nights. Figure 6.9 reveals that this trend is due to the number of activities the users engage in throughout the day in each hour. Figure 6.9 shows the number of distinct tags identified in the user events in each hour of day. Not surprisingly, people check-in at more diverse places during the day than night, at which time most people

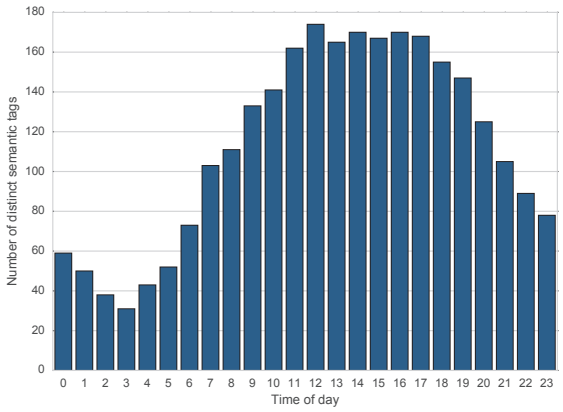
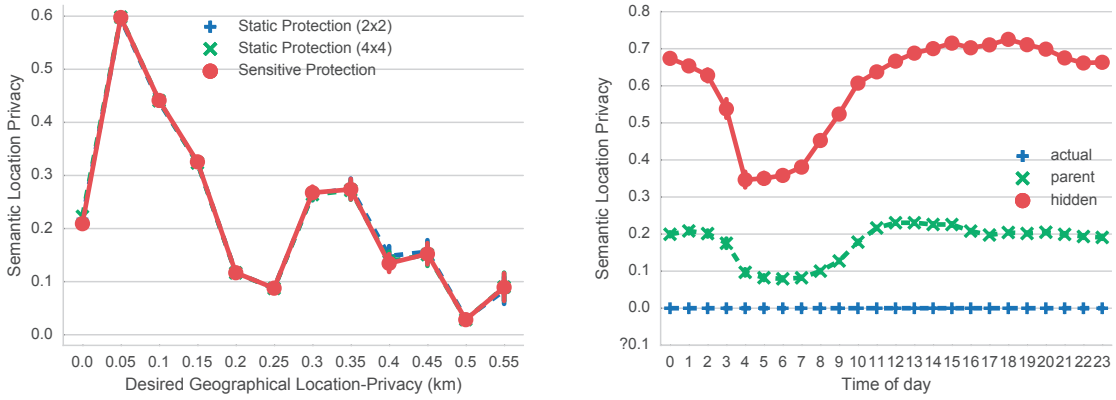


Figure 6.9 – Number of unique semantic tags observed in the user events over time of day.

(who do check-in) visit nightlife places. More diverse activities during the day actually help people to hide their activities and locations easier, because this is reflected in the background knowledge of the adversary as more confusion.

We also present results concerning the semantic location privacy in Figure 6.10. In this case the obtained privacy level is computed as a normalized tree distance between the inferred and the actual semantic tags. As it can be observed both the sensitive and static protection mechanisms throw similar results (Figure 6.10a) This is due to the fact that we applied a fixed semantic protection scenario throughout all the experiments for each geographical PPM. Also, the effect of applying different levels of geographical obfuscation is negligible on the semantic location-privacy; however, it is important to note that these results are obtained for semantically rich and dense areas. We can argue that in places with few semantic tags available, the semantic location-privacy would be more sensitive to the size of geographical obfuscation areas. Moreover, we plot semantic location-privacy over time of day for different semantic privacy-protection mechanisms (PPM) in Figure 6.10b.



(a) Over the desired level of privacy

(b) Over time of day

Figure 6.10 – Average semantic location-privacy of user events over (a) desired privacy level and (b) time of day.

Unsurprisingly, hiding the semantic tags provides the best protection and parent-tag generalization achieves some degree of protection, which might be useful in certain scenarios where the utility is based on semantics. Furthermore, we observe a correlation between the drop in semantic location-privacy and the event occurrence (in conjunction with Figure 6.7b), yet little sign of effect by the number of distinct tags exposed by the users throughout the day (Figure 6.9).

6.4.4 Discussion

The results obtained with the proposed sensitive PPM yield insightful outcomes given that we trained the attacker with the aforementioned dataset. The lack of rich and diverse datasets of real traces limits the research on this subject. Confirming the results obtained in this chapter with additional datasets, and also larger areas of interest is essential. We expect the trends in the results to be similar.

Another question regarding this work would be to seek the possibility to implement a protection mechanism that runs on the user-side similar to the Location-Privacy Library presented in Chapter 4. It is feasible to implement such an extension to the existing Location-Privacy Library, however it is not straightforward to implement a more complex privacy-estimation module that is in line of Part II of this thesis. Such adversary models are more complex and require a lot of resources to attack user traces. Nevertheless, the evaluation in this chapter demonstrates that a sensitivity-aware protection mechanism that takes into account user history is capable of meeting the desired privacy levels up to a certain point against a strong adversary.

6.5 Summary

In this chapter, we presented a time-aware adversary model and the related inference approach that is shown to be stronger than a time-oblivious adversary in various time periods. Additionally, we presented a sensitivity-aware privacy-protection mechanism that automatically protects the geographical location of a user based on the desired level of protection on the semantic dimension and also the privacy sensitivities of the user. This approach to privacy protection is shown to be superior to static protection approaches that do not consider user history, sensitivities and location semantics. The experimental results provide insights that can further push the development of such privacy-protection mechanisms against strong adversaries. Furthermore, we realized the evaluation of protection mechanisms w.r.t. stronger adversaries by actually implementing them and performing attacks to obfuscated traces multi-dimensionally.

Conclusions

In this thesis, we focused on privacy protection and inference attacks regarding location privacy in mobile applications. Service providers such as location-based services and online social networks have almost uncontrolled access to massive amounts of user data enabling them to mine intimate details about individuals. As more and more users are utilizing smart devices every day to interact on the Internet, they unknowingly commit data to various service providers and thus potential adversaries. Users will not stop using the services they get as their access to information is enhanced with mobile applications. Hence, to address this privacy vulnerability, protection mechanisms are designed and proposed to be integrated to such applications. We approached this problem by considering what an adversary can exploit in his attack in relation to user traces. It is essential to realize that protecting and attacking privacy are two sides of the same medallion. This realization helped us build adaptive location-privacy protection mechanisms (*i.e.,* location PPMs) that take into account an adversary's knowledge on user mobility and also a user's privacy needs. On the other hand, we also focused on how to infer users' location traces better by exploiting location semantics and their time dependency, hence demonstrating a more concrete power of a potential adversary.

More specifically, in Part I, we studied user mobility and mobility constraints such as velocity in order to counter the threats by mobility-aware adversaries. In Chapter 2, we designed a privacy-protection approach that relies on local privacy-level estimation based on Bayesian inference. This enabled us to adapt a PPM's parameters and thus create a more resistant privacy protection. Our evaluation of the adaptive protection against a mobility-aware adversary shows that the adaptive approach not only protects users' location privacy better w.r.t. their requirements than a static protection mechanism, but also avoids over-protection and thereby improves application utility. Our findings helped us understand that a PPM must consider the adversary capabilities more concretely unlike the most of the existing work and should be able to estimate the privacy level.

In Chapter 3, we focused on mobility of users in terms of mobility history, direction of movement and velocity. The intuition was that an adversary, who knows a user's most visited places and also reasons about user mobility, can identify the fake parts of a random obfuscation area. Therefore, it is important to generate obfuscation areas with maximum confusion possible. Consequently, we formalized probability distributions over user movements and designed a PPM that uses these distributions and heuristically determines obfuscation areas with the

highest chance to confuse an adversary. The success of the heuristic approach at protecting location privacy proved to be remarkably outperforming compared to random obfuscation approaches when facing a strong adversary. Our findings in this chapter, together with Chapter 2, suggest that strategic PPMs can be developed and deployed on mobile devices that automatically protect location privacy. Merging the adaptive approach with local estimation of privacy level in Chapter 2 with the mobility-aware PPM in this chapter can yield a protection scheme for mobile devices that provides near-optimal privacy levels.

Our work in chapters 2 and 3 inspired us to actually implement and evaluate a protection mechanism on mobile devices to study the applicability of our approaches, as to the best of our knowledge, the implementation of such extensive PPMs on actual devices has not been investigated before. We implemented our adaptive approach in Chapter 2 on Android platform as a library consisting of three main modules: (i) a privacy-level estimation module, responsible for evaluating the expected privacy level of the user, (ii) a PPM that adaptively increases the size of obfuscation to apply based on the feedback from the estimation module, and (iii) a sensitivity module that lets users to provide their privacy sensitivities w.r.t. certain semantic tags and locations. The library is integrated to a real sensing application called TinyGSN [8] and tested in terms of correctness and performance. The results show that our library is lightweight on modern smartphones and runs smoothly, consequently paving the way for making active privacy protection on mobile devices a reality.

In Part II, we investigated the additional power factor of an adversary that exploits semantic dimension of location information, *i.e.,* the types of visited locations. The privacy concern regarding such an adversary rised from the fact that users publish information regarding their locations on online social networks to inform their friends, to state an opinion about a place or simply to check-in. These disclosures provide potential adversaries with additional information that they can exploit in their inference attacks on user traces. We formalized and designed this setting with Bayesian networks in Chapter 5, and performed attacks on semantically-annotated traces obtained from Twitter and Foursquare. We quantified the privacy loss induced by disclosed semantic information along with user location and saw that considering location semantics improves the strength of an attacker remarkably.

In Chapter 6, we further extended our adversary model to include the time dimension of user visits based on our findings that the semantics in user traces exhibit time dependency. We show that this time- and semantic-aware adversary is more successful at inferring the traces of users from protected traces. The results of our evaluation provide a better understanding of location privacy from a multi-dimensional point of view. They also indicate that complex modeling of both PPMs and adversaries is required in order to study location privacy in mobile applications, and to provide strategic ways of protecting to users. Lastly, we implemented an adaptive and sensitivity-aware PPM, similar to the one in Chapter 2 in order to investigate whether privacy sensitivities of users can be protected against such powerful adversaries. Our experiments demonstrate that this is indeed possible, yet significantly challenging if the users already leaked considerable amount of information regarding their sensitive visits in the past.

Consequently, it is clear that location privacy is susceptible to location semantics and habits of users under strong and realistic adversary models. The author believes the findings in this thesis provide a better understanding of location privacy from a multi-dimensional point of view and evaluation techniques with the introduced adversary models. The proposed protection approaches along with the location-privacy library and the adversary models comprise a useful toolbox both for enabling privacy protection in location-based mobile applications and for further research on location privacy.

Bibliography

[1] "Android Operating System," http://www.android.com/, accessed: 2014-08-23.

[2] "Bayesian Belief Network Package," https://github.com/eBay/bayesian-belief-networks, accessed: 2015-08-16.

[3] "Foursquare Category Hierarchy," https://developer.foursquare.com/categorytree, accessed: 2014-09-15.

[4] "Location Privacy Library," https://github.com/LSIR/LocPrivLib/.

[5] "OpenSense Project," http://opensense.epfl.ch, accessed: 2016-04-02.

[6] "OpenStreetMap (OSM)," http://www.openstreetmap.org/, accessed: 2015-09-26.

[7] "Spatialite," http://www.gaia-gis.it/gaia-sins/, accessed: 2015-07-27.

[8] "TinyGSN," https://github.com/LSIR/gsn/tree/master/gsn-tiny, accessed: 2015-05-07.

[9] "Twitter Public Streams," https://dev.twitter.com/docs/streaming-apis/streams/public, accessed: 2013-04-14.

[10] K. Aberer, M. Hauswirth, and A. Salehi, "A Middleware for Fast and Flexible Sensor Network Deployment," in *Proceedings of the International Conference on Very large Data Bases*. VLDB Endowment, 2006, pp. 1199–1202.

[11] B. Agir, T. G. Papaioannou, R. Narendula, K. Aberer, and J.-P. Hubaux, "User-side Adaptive Protection of Location Privacy in Participatory Sensing," *Geoinformatica*, vol. 18, no. 1, pp. 165–191, 2014.

[12] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: Differential Privacy for Location-based Systems," in *Proc. of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2013.

[13] C. A. Ardagna, M. Cremonini, and G. Gianini, "Landscape-aware Location-Privacy Protection in Location-based Services," *Journal of Systems Architecture*, vol. 55, no. 4, pp. 243–254, 2009.

[14] O. Barak, G. Cohen, and E. Toch, "Anonymizing Mobility Data Using Semantic Cloaking," *Pervasive and Mobile Computing*, 2015.

[15] E. Bart, R. Zhang, and M. Hussain, "Where Would You Go this Weekend? Time-Dependent Prediction of User Activity Using Social Network Data," in *Proc. of the International Conference on Weblogs and Social Media (ICWSM)*, 2013.

[16] A. R. Beresford and F. Stajano, "Location Privacy in Pervasive Computing," *IEEE Pervasive Computing*, vol. 2, no. 1, pp. 46–55, Jan. 2003.

[17] C. Bettini, X. S. Wang, and S. Jajodia, "Protecting Privacy Against Location-based Personal Identification," in *Secure Data Management*, 2005, pp. 185–199.

[18] G. Biczók and P. H. Chia, "Interdependent Privacy: Let Me Share Your Data," in *Proc. of International Conference Financial Cryptography and Data Security*, 2013.

[19] I. Bilogrevic, K. Huguenin, S. Mihaila, R. Shokri, and J.-P. Hubaux, "Predicting Users' Motivations behind Location Check-Ins and Utility Implications of Privacy Protection Mechanisms," in *Proc. of the Network and Distributed System Security Symposium (NDSS)*, 2015, pp. 1–11.

[20] I. Bilogrevic, K. Huguenin, B. Agir, M. Jadliwala, and J.-P. Hubaux, "Adaptive Information-Sharing for Privacy-Aware Mobile Social Networks," in *Proc. of UbiComp*, 2013.

[21] N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Optimal Geo-Indistinguishable Mechanisms for Location Privacy," in *Proc. of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2014.

[22] F. Calabrese, G. Di Lorenzo, and C. Ratti, "Human Mobility Prediction Based on Individual and Collective Geographical Preferences," in *Proc. of the IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2010.

[23] T. Camp, J. Boleng, and V. Davies, "A Survey of Mobility Models for Ad hoc Network Research," *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 483–502, 2002.

[24] R. Canetti, U. Feige, O. Goldreich, and M. Naor, "Adaptively Secure Multi-Party Computation," in *Proc. of Symposium on Theory of Computing (STOC)*, 1996.

[25] G. Cantor, *Contributions to the Founding of the Theory of Transfinite Numbers*, 1952.

[26] J. Cappos, L. Wang, R. Weiss, Y. Yang, and Y. Zhuang, "Blursense: Dynamic Fine-grained Access Control for Smartphone Privacy," in *Proc. of Sensors Applications Symposium (SAS),*, 2014.

[27] B. Carbunar, R. Sion, R. Potharaju, and M. Ehsan, "The Shy Mayor: Private Badges in GeoSocial Networks," in *Proc. of the International Conference on Applied Cryptography and Network Security (ACNS)*, 2012.

[28] K. Chatzikokolakis, C. Palamidessi, and M. Stronati, "Constructing Elastic Distinguishability Metrics for Location Privacy," *Proc. on Privacy Enhancing Technologies (PoPETs)*, vol. 2015, no. 2, pp. 156–170, 2015.

[29] X. Chen, A. Mizera, and J. Pang, "Activity Tracking: A New Attack on Location Privacy," in *Proc. of Conference on Communications and Network Security (CNS)*, 2015.

[30] Z. Cheng, J. Caverlee, and K. Lee, "You are Where You Tweet: A Content-based Approach to Geo-locating Twitter Users," in *Proc. of the ACM International Conference on Information and Knowledge Management (CIKM)*, 2010.

[31] D. Christin, A. Reinhardt, S. S. Kanhere, and M. Hollick, "A Survey on Privacy in Mobile Participatory Sensing Applications," *Journal of Systems and Software*, vol. 84, no. 11, pp. 1928–1946, 2011.

[32] D. Christin, C. Rosskopf, M. Hollick, L. A. Martucci, and S. S. Kanhere, "IncogniSense: An Anonymity-Preserving Reputation Framework for Participatory Sensing Applications," in *Proc. of IEEE Conference on Pervasive Computing and Communications (PerCom)*, 2012.

[33] T. Cook, "Apple Customer Letter," February 2016, accessed: 2016-04-20. [Online]. Available: https://www.apple.com/customer-letter/

[34] M. L. Damiani, E. Bertino, and C. Silvestri, "The PROBE Framework for the Personalized Cloaking of Private Locations," *Transactions on Data Privacy*, pp. 123–148, 2010.

[35] T. Das, P. Mohan, V. N. Padmanabhan, R. Ramjee, and A. Sharma, "PRISM: Platform for Remote Sensing using Smartphones," in *Proc. of Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2010.

[36] E. De Cristofaro and C. Soriente, "Short Paper: PEPSI—Privacy-Enhanced Participatory Sensing Infrastructure," in *Proc. of the ACM conference on Wireless Network Security (WiSec)*, 2011.

[37] R. Dewri, "Local Differential Perturbations: Location Privacy Under Approximate Knowledge Attackers," *IEEE Transactions on Mobile Computing*, vol. 12, no. 12, pp. 2360–2372, 2013.

[38] C. Diaz, S. Seys, J. Claessens, and B. Preneel, "Towards Measuring Anonymity," in *Proc. of International Workshop on Privacy Enhancing Technologies (PET)*, 2002.

[39] C. Dong and N. Dulay, "Longitude: A Privacy-Preserving Location Sharing Protocol for Mobile Applications," in *Proc. of the International Conference on Trust Management (IFIPTM)*, 2011, pp. 133–148.

[40] A. Dua, N. Bulusu, W.-C. Feng, and W. Hu, "Towards Trustworthy Participatory Sensing," in *Proc. of USENIX Conference on Hot Topics in Security (HotSec)*, 2009.

[41] M. Duckham and L. Kulik, "A Formal Model of Obfuscation and Negotiation for Location Privacy," *Pervasive Computing*, pp. 243–251, 2005.

[42] C. Dwork, "Differential privacy," in *International Colloquium on Automata, Languages and Programming*, 2006, pp. 1–12.

[43] J. Eberle, "Energy-efficient Continuous Context Sensing on Mobile Phones," Ph.D. dissertation, EPFL, 2015.

[44] W. Enck, P. Gilbert, S. Han, V. Tendulkar, B.-G. Chun, L. P. Cox, J. Jung, P. McDaniel, and A. N. Sheth, "TaintDroid: An Information-Flow Tracking System for Realtime Privacy Monitoring on Smartphones," *ACM Transactions on Computer Systems (TOCS)*, vol. 32, no. 2, p. 5, 2014.

[45] European Comission, "Factsheet on the "Right to be Forgotten" ruling," accessed: 2016-04-11. [Online]. Available: http://ec.europa.eu/justice/data-protection/files/factsheets/factsheet_data_protection_en.pdf

[46] K. Fawaz, H. Feng, and K. G. Shin, "Anatomization and Protection of Mobile Apps' Location Privacy Threats," in *Proc. of USENIX Security Symposium*, 2015.

[47] K. Fawaz and K. G. Shin, "Location Privacy Protection for Smartphone Users," in *Proc. of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2014.

[48] D. Freni, C. Ruiz Vicente, S. Mascetti, C. Bettini, and C. S. Jensen, "Preserving Location and Absence Privacy in Geo-social Networks," in *Proc. of the Conference on Information and Knowledge Management (CIKM)*, 2010.

[49] B. Gedik and L. Ling, "Protecting Location Privacy with Personalized k-Anonymity: Architecture and Algorithms," *IEEE Transactions on Mobile Computing*, vol. 7, no. 1, pp. 1–18, 2008.

[50] G. Ghinita, M. L. Damiani, C. Silvestri, and E. Bertino, "Preventing Velocity-based Linkage Attacks in Location-aware Applications," in *Proc. of the ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2009.

[51] G. Ghinita, P. Kalnis, and S. Skiadopoulos, "MobiHide: A Mobile Peer-to-Peer System for Anonymous Location-Based Queries," in *Proc. of Advances in Spatial and Temporal Databases*, 2007.

[52] P. Golle and K. Partridge, "On the Anonymity of Home/Work Location Pairs," in *Pervasive Computing*, 2009, vol. 5538, pp. 390–397.

[53] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi, "Understanding Individual Human Mobility Patterns," *Nature*, vol. 453, no. 7196, pp. 779–782, 2008.

[54] M. Götz, S. Nath, and J. Gehrke, "Maskit: Privately Releasing User Context Streams for Personalized Mobile Applications," in *Proc. of the ACM SIGMOD International Conference on Management of Data*, 2012.

[55] G. Greenwald and E. MacAskill, "NSA Prism Program Taps into User Data of Apple, Google and Others," *The Guardian*, 2013, accessed: 2016-04-20. [Online]. Available: http://www.theguardian.com/world/2013/jun/06/us-tech-giants-nsa-data

[56] M. M. Groat, B. Edwards, J. Horey, W. He, and S. Forrest, "Enhancing Privacy in Participatory Sensing Applications with Multidimensional Data," in *Proc. of IEEE Conference on Pervasive Computing and Communications (PerCom)*, 2012.

[57] M. Gruteser and D. Grunwald, "Anonymous Usage of Location-based Services Through Spatial and Temporal Cloaking," in *Proc. of the International Conference on Mobile systems, Applications and Services (MobiSys)*. ACM, 2003, pp. 31–42.

[58] M. Gruteser and B. Hoh, "On the Anonymity of Periodic Location Samples," in *Proc. of Conference on Security in Pervasive Computing*, 2005, pp. 179–192.

[59] M. Herrmann, A. Rial, C. Diaz, and B. Preneel, "Practical Privacy-Preserving Location-sharing Based Services with Aggregate Statistics," in *Proc. of the ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec)*, 2014.

[60] H. Hu and J. Xu, "Non-Exposure Location Anonymity," in *Proc. of IEEE International Conference on Data Engineering (ICDE)*, 2009.

[61] M. Jadliwala, J. Freudiger, I. Aad, J.-P. Hubaux, and V. Niemi, "Privacy-Triggered Communications in Pervasive Social Networks," in *Proc. of IEEE International Symposium on World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2011.

[62] F. V. Jensen, "Junction Trees and Decomposable Hypergraphs," Judex Datasystemer, Aalborg, Denmark., Tech. Rep., 1988.

[63] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*, 2009.

[64] C. Komninakis, "A Fast and Accurate Rayleigh Fading Simulator," in *Proc. of IEEE Global Telecommunications Conference (GLOBECOM)*, 2003.

[65] A. Krause, E. Horvitz, A. Kansal, and F. Zhao, "Toward Community Sensing," in *Proc. of International Conference on Information Processing in Sensor Networks (IPSN)*, 2008.

[66] A. Krause and E. Horvitz, "A Utility-Theoretic Approach to Privacy in Online Services," *Journal of Artificial Intelligence Research*, pp. 633–662, 2010.

[67] J. Krumm, "A Survey of Computational Location Privacy," *Personal and Ubiquitous Computing*, vol. 13, no. 6, pp. 391–399, 2009.

[68] ——, "Inference Attacks on Location Tracks," in *Proc. of the International Conference on Pervasive Computing*, 2007.

[69] J. Krumm and D. Rouhana, "Placer: Semantic Place Labels from Diary Data," in *Proc. of the Conference on Pervasive and Ubiquitous Computing (UbiComp)*, 2013.

[70] B. Lee, J. Oh, H. Yu, and J. Kim, "Protecting Location Privacy Using Location Semantics," in *Proc. of ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2011.

[71] N. Li, T. Li, and S. Venkatasubramanian, "t-Closeness: Privacy Beyond k-Anonymity and l-Diversity," in *Proc. of IEEE International Conference on Data Engineering (ICDE)*, 2007.

[72] W. Li, P. Serdyukov, A. P. de Vries, C. Eickhoff, and M. Larson, "The Where in the Tweet," in *Proc. of the Conference on Information and Knowledge Management (CIKM)*, 2011.

[73] B. Liang and Z. Haas, "Predictive Distance-based Mobility Management for PCS Networks," in *Proc. of the Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 1999.

[74] M. Lisi, "Some Remarks on the Cantor Pairing Function," *Le Matematiche*, vol. 62, no. 1, pp. 55–65, 2007.

[75] H. Liu, B. Luo, and D. Lee, "Location Type Classification Using Tweet Content," in *Proc. of the International Conference on Machine Learning and Applications (ICMLA)*, 2012.

[76] H. Lu, W. Pan, N. D. Lane, T. Choudhury, and A. T. Campbell, "SoundSense: Sound Sensing for People-Centric Applications on Mobile Phones," in *Proc. of Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2009.

[77] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy Beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, 2007.

[78] C. D. Manning, P. Raghavan, H. Schütze *et al.*, *Introduction to Information Retrieval*, 2008.

[79] J. Meyerowitz and R. Roy Choudhury, "Hiding Stars with Fireworks: Location Privacy Through Camouflage," in *Proc. of the International Conference on Mobile Computing and Networking*, 2009.

[80] K. Minami and N. Borisov, "Protecting Location Privacy Against Inference Attacks," in *Proc. of ACM Workshop on Privacy in the Electronic Society (WPES)*, 2010.

[81] A. Monreale, R. Trasarti, D. Pedreschi, C. Renso, and V. Bogorny, "C-safety: A Framework for the Anonymization of Semantic Trajectories," *Transactions on Data Privacy*, vol. 4, no. 2, pp. 73–101, 2011.

[82] B. Mood, D. Gupta, K. Butler, and J. Feigenbaum, "Reuse it or Lose it: More Efficient Secure Computation Through Reuse of Encrypted Values," in *Proc. of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2014.

[83] M. Mun, S. Hao, N. Mishra, K. Shilton, J. Burke, D. Estrin, M. Hansen, and R. Govindan, "Personal Data Vaults: A Locus of Control for Personal Data Streams," in *Proc. of ACM Conference on Emerging Networking Experiments and Technologies (Co-NEXT)*, 2010.

[84] M. Mun, S. Reddy, K. Shilton, N. Yau, J. Burke, D. Estrin, M. Hansen, E. Howard, R. West, and P. Boda, "PEIR, the Personal Environmental Impact Report, as a Platform for Participatory Sensing Systems Research," in *Proc. of Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2009.

[85] Nokia Research Center, "Lausanne Data Collection Campaign," https://www.idiap.ch/dataset/mdc, accessed: 2016-05-20.

[86] A.-M. Olteanu, K. Huguenin, R. Shokri, and J.-P. Hubaux, "Quantifying the Effect of Co-locations on Location Privacy," in *Proc. of the Privacy Enhancing Technologies Symp. (PETS)*, 2014, pp. 184–203.

[87] A.-M. Olteanu, K. Huguenin, R. Shokri, M. Humbert, and J.-P. Hubaux, "Quantifying Interdependent Privacy Risks with Location Data," *IEEE Transactions on Mobile Computing*, vol. PP, no. 99, 2016.

[88] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, 2014.

[89] A. Pingley, W. Yu, N. Zhang, X. Fu, and W. Zhao, "CAP: A Context-Aware Privacy Protection System for Location-Based Services," in *Proc. of IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2009.

[90] M. Piorkowski, N. Sarafijanovic-Djukic, and M. Grossglauser, "CRAWDAD Dataset Epfl/Mobility (v. 2009-02-24)," Downloaded from http://crawdad.org/epfl/mobility/20090224, Feb. 2009.

[91] V. Primault, S. B. Mokhtar, C. Lauradoux, and L. Brunie, "Differentially Private Location Privacy in Practice," in *Proc. of International Workshop on Mobile Security Technologies*, 2014.

[92] C. C. Robusto, "The Cosine-Haversine Formula," *The American Mathematical Monthly*, vol. 64, no. 1, pp. 38–40, 1957.

[93] L. Rossi, M. J. Williams, C. Stich, and M. Musolesi, "Privacy and the City: User Identification and Location Semantics in Location-based Social Networks," *Proc. of International AAAI Conference on Web and Social Media (ICWSM)*, 2015.

[94] A. Serjantov and G. Danezis, "Towards an Information Theoretic Metric for Anonymity," in *Proc. of International Workshop on Privacy Enhancing Technologies (PET)*, 2002.

[95] C. N. Service, "Tracking Your Phone: A Look into the Technology," http://marylandreporter.com/2016/05/11/tracking-your-phone-a-look-into-the-technology/, accessed: 20.05.2016.

[96] P. Shankar, V. Ganapathy, and L. Iftode, "Privately Querying Location-Based Services with SybilQuery," in *Proc. of Conference on Ubiqutious Computing (UbiComp)*, 2009.

[97] R. Shokri, J. Freudiger, and J.-P. Hubaux, "A Unified Framework for Location Privacy," in *Proc. of the Workshop on Hot Topics in Privacy Enhancing Technologies (HotPETS)*, 2010.

[98] R. Shokri, J. Freudiger, M. Jadliwala, and J.-P. Hubaux, "A Distortion-based Metric for Location Privacy," in *Proc. of ACM Workshop on Privacy in the Electronic Society (WPES)*, 2009.

[99] R. Shokri, G. Theodorakopoulos, G. Danezis, J.-P. Hubaux, and J.-Y. Le Boudec, "Quantifying Location Privacy: The Case of Sporadic Location Exposure," in *Proc. of the Privacy Enhancing Technologies Symp. (PETS)*, 2011.

[100] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, and J.-P. Hubaux, "Quantifying Location Privacy," in *Proc. of IEEE Symposium on Security and Privacy (S&P)*, 2011.

[101] A. Singla and A. Krause, "Incentives for privacy tradeoff in community sensing," in *First AAAI Conference on Human Computation and Crowdsourcing*, 2013.

[102] L. Sweeney, "k-anonymity: A Model for Protecting Privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 5, pp. 557–570, 2002.

[103] E. Toch, "Crowdsourcing Privacy Preferences in Context-aware Applications," *Personal and Ubiquitous Computing*, vol. 18, no. 1, pp. 129–141, 2014.

[104] E. Toch, J. Cranshaw, P. H. Drielsma, J. Y. Tsai, P. G. Kelley, J. Springfield, L. Cranor, J. Hong, and N. Sadeh, "Empirical Models of Privacy in Location Sharing," in *Proc. of the ACM International Conference on Ubiquitous Computing (UbiComp)*, 2010.

[105] E. Toch, Y. Wang, and L. F. Cranor, "Personalization and Privacy: A Survey of Privacy Risks and Remedies in Personalization-based Systems," *User Modeling and User-Adapted Interaction*, vol. 22, no. 1-2, pp. 203–220, 2012.

[106] D. Tynan, "Why Location Privacy is Important," http://www.itworld.com/article/2752981/mobile/why-location-privacy-is-important.html, accessed: 20.05.2016.

[107] K. Vu, R. Zheng, and J. Gao, "Efficient Algorithms for k-anonymous Location Privacy in Participatory Sensing," in *Proc. of IEEE Conference om Computer Communications (IEEE INFOCOM)*, 2012.

[108] M. Wernke, P. Skvortsov, F. Dürr, and K. Rothermel, "A Classification of Location Privacy Attacks and Approaches," *Personal and Ubiquitous Computing*, vol. 18, no. 1, pp. 163–175, 2014.

[109] A. F. Westin, *Privacy and Freedom*, 1967.

[110] World Health Organization, "Electromagnetic fields and public health," http://www.who.int/mediacentre/factsheets/fs304/en/index.html.

[111] X. Xiao and Y. Tao, "Personalized Privacy Preservation," in *Proc. of ACM SIGMOD Conference on Management of Data*, 2006.

[112] Z. Xiao, J. Xu, and X. Meng, "p-Sensitivity: A Semantic Privacy-Protection Model for Location-based Services," in *Proc. of International Conference on Mobile Data Management Workshops (MDMW)*, 2008.

[113] J. Xu, X. Tang, H. Hu, and J. Du, "Privacy-conscious Location-based Queries in Mobile Environments," *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 3, pp. 313–326, 2010.

[114] T. Xu and Y. Cai, "Feeling-based Location Privacy Protection for Location-based Services," in *Proc. of the ACM conference on Computer and Communications Security (CCS)*, 2009.

[115] Z. Xu and S. Zhu, "SemaDroid: A Privacy-Aware Sensor Management Framework for Smartphones," in *Proc. of the ACM Conference on Data and Application Security and Privacy*, 2015.

[116] M. Xue, P. Kalnis, and H. K. Pung, "Location Diversity: Enhanced Privacy Protection in Location Based Services," in *Proc. of the International Symposium on Location and Context Awareness (LOCA)*, 2009.

[117] Z. Yan, D. Chakraborty, C. Parent, S. Spaccapietra, and K. Aberer, "SeMiTri: A Framework for Semantic Annotation of Heterogeneous Trajectories," in *Proc. of the International Conference on Extending Database Technology (EDBT/ICDT)*, 2011.

[118] M. Yiu, C. Jensen, X. Huang, and H. Lu, "Spacetwist: Managing the Trade-offs Among Location Privacy, Query Performance, and Query Accuracy in Mobile Services," in *Proc. of the IEEE International Conference on Data Engineering (ICDE)*, 2008.

[119] H. Zang and J. Bolot, "Anonymization of Location Data Does Not Work: A Large-scale Measurement Study," in *Proc. of the Annual International Conference on Mobile Computing and Networking (MobiCom)*, 2011.

[120] L. Zhang, B. Tiwana, Z. Qian, Z. Wang, R. P. Dick, Z. M. Mao, and L. Yang, "Accurate Online Power Estimation and Automatic Battery Behavior Based Power Model Generation for Smartphones," in *Proc. of the Intl. Conference on Hardware/Software Codesign and System Synthesis*, 2010.

[121] G. Zhong, I. Goldberg, and U. Hengartner, "Louis, Lester and Pierre: Three Protocols for Location Privacy," in *Proc. of International Workshop on Privacy Enhancing Technologies (PET)*, 2007.

Berker AĞIR

Data Scientist, Privacy Expert, Researcher, Software Developer

+41 76 405 11 87
berkeragir@gmail.com
https://www.linkedin.com/in/berkeragir

EPFL IC IINFCOM LSIR,
BC 146, Station 14
1015 Lausanne, Switzerland

Areas of Interests

Data Privacy, Data Science, Predictive Models, Bayesian Networks and Inference, Social Network Datasets

Experience

2010-2016 **Research Assistant** (EPFL) Lausanne, Switzerland

Thesis: *Context and Semantic Aware Location Privacy*

Thesis Director: Prof. Karl Aberer

- Initiated and managed research on adaptive location-privacy protection mechanisms. I designed and validated solutions based on Bayesian inference, one of which is implemented as an Android prototype by 2 students under my supervision. I finalized the project with publications and an open-source library.
- I developed a crawler in Python to collect geo-tagged Twitter tweets and corresponding Foursquare check-ins which successfully ran through Jan. and Jul. 2015. Processed and structured the data, and designed the MySQL database to enable usage for research.
- I analyzed user behavior in the data collected with Python in terms of location semantics and researched the impact of semantics on geographical location privacy in collaboration with EPFL scientists by modelling users with Bayesian Networks. Also formalized and evaluated *semantic* location-privacy. I extensively used Python data analytics packages such as Pandas, Numpy and Seaborn.
- Developed exercises and unit tests in Java for grading student submissions which successfully ran on Coursera for two introductory programming courses in Java for 2 semesters serving more than 20,000 students online in total.

06/2008-08/2008 **Intern** (Cabot Communications) Izmir, Turkey

As a member of a team developing a new set-top box for TVs, I developed a built-in web interface for users that showed news and weather forecast when the set-top box was booted. I also developed a multiplayer chess game that could be played online via the system. The utilized technologies were PHP, CSS and AJAX.

2006-2008 **Part-time Developer** (Nazar Bilgisayar) Izmir, Turkey

Developed from scratch a sales management software and an accounting software for Windows using MS Visual Basic .NET and MS SQL Server based on older MSDOS applications. Migrated the data from the old applications to the new ones.

Selected Publications

1. Berker Agir, Kévin Huguenin, Urs Hengartner, Jean-Pierre Hubaux. **On the Privacy Implications of Location Semantics**. Proc. on Privacy Enhancing Technologies (PoPETs), 2016.
2. Berker Agir, Jean-Paul Calbimonte and Karl Aberer. **Semantic and Sensitivity Aware Location-Privacy Protection for the Internet of Things**. Privacy Online: Workshop on Society, Privacy and the Semantic Web (PrivOn) 2014.
3. Berker Agir, Thanasis G. Papaioannou, Rammohan Narendula, Karl Aberer and Jean-Pierre Hubaux. **User-side Adaptive Protection of Location Privacy in Participatory Sensing**, in Geoinformatica, vol. 18, num. 1, p. 165-191, 2014.

Education

2010 – 2016	**PhD in Computer and Communication Sciences** École Polytechnique Fédérale de Lausanne (EPFL)	Lausanne, Switzerland
2005 – 2010	**B.Sc. in Computer Science and Engineering** Sabanci University (Graduation GPA: 3.80 / 4.00) • Merit Scholarship (Annual 2/3 of Tuition Fee exemption) • Certificate of High Honor for 6 semesters (2006 – 2009) • Extracurricular Student Activities Award - Theater Club (2007)	Istanbul, Turkey
2001 – 2005	**Karsiyaka Atakent Anatolian High School** (Graduation GPA: 5.00 / 5.00)	Izmir, Turkey

Skills

Development Experience in: Python, C/C++, Java, C#, PHP, JavaScript, SQL

Data Science: Data analytics packages in Python (Pandas, Numpy, MatPlotLib & Seaborn), Bayesian networks, Hidden Markov Models, Clustering, Decision Trees

Development Environments & Platforms: MS Windows, Linux, Android, PyCharm, Eclipse, MS Visual Studio, MySQL, MS SQL Server, PostgreSQL, IBM DB2, Android Studio, Matlab

Other: Git, SVN, HTML, CSS, AJAX, JQuery, LaTeX

Languages

Turkish (native), English (proficient) and French (A2)

Extracurricular Activities

- **TURQUIA 1912 – Turkish Students Association in Switzerland**
  - President (2013-2015), Treasurer (2012-2013), Overseer (2011-2012)
  - Conducted meetings to introduce the association to new people and attracted members.
  - Organized and found sponsors for the traditional annual reception in 2014 and 2015 for about 70-100 attendants including Turkish representatives in Switzerland.

- **IBM Turkey – Software Competition** (2008), Istanbul
  Image-Processing Algorithms Database Web Application Project
  As the core developer of the project, I implemented a Java Servlet-based web application that lets users to upload image-processing algorithms and run them online. Utilized Java Servlets, JSP, IBM DB2 and IBM WebSphere Server. Participated as a group of 3 and ranked 3rd.

- **Civic Involvement Projects** at Sabanci University (http://cip.sabanciuniv.edu/en):
  - Acted as one of the 3 leading volunteers between 2007 and 2010 for organizing a large social event day for ≈2500 school children organized annually on campus. Determined and delegated tasks to volunteers, found sponsors and followed through the plan during the event day.
  - Voluntarily acted as a team supervisor for freshman students to conduct weekly community service for school children.

- I am a long-time leisure swimmer and enjoy swimming in pools and the sea alike. I am also holder of a CMAS 2* diver's certificate. I experienced diverse cultural activities during my bachelor studies such as acting in the theater club and singing in a choir. I also have an interest in amateur photography. Living in Switzerland, I am happily a beginner skier.