Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

School of Computer Science and Statistics

Privacy Aware Travel Assistant

Supervisor: Melanie Bouroche

August 19, 2022

A dissertation submitted in partial fulfilment
of the requirements for the degree of
MSc (Computer Science - Data Science)

Declaration

I hereby declare that this dissertation is entirely my own work and that it has not been submitted as an exercise for a degree at this or any other university.

I have read and I understand the plagiarism provisions in the General Regulations of the University Calendar for the current year, found at `http://www.tcd.ie/calendar`.

I have also completed the Online Tutorial on avoiding plagiarism 'Ready Steady Write', located at `http://tcd-ie.libguides.com/plagiarism/ready-steady-write`.

Signed: ____________________ Date: ____________________

Abstract

In the era of digital assistants, travel assistants are widely used in everyday travel to easily access route updates during the journey. On-route travel assistants requires continuous user location updates from the start of the journey, during the journey and until the end. Currently there are various approaches to protect the static user location such as location obfuscation, anonymity and perturbation but the constant updating makes location correlation protection more challenging. Spatio-temporal relations between the adjacent location points weakens the traditional as well as advanced privacy preserving mechanisms and thus the trajectory must be independent of correlation between those location points.

To address this issue, we propose using Local Differential Privacy with RAPPOR mechanism to construct a privacy-protection mechanism that is based on user locations predicted using a Markov process, rather than actual user locations. This aids ensuring the user's location privacy while also maintaining the correlation between the true trajectory location points, hence improving the utility of the system. The proposed approach has been evaluated in three different ways confirming the improvement of the privacy preserving mechanism RAPPOR when integrated with the location prediction algorithm based on Hidden Markov Model, comparing the actual trajectory and the predicted trajectory based on Markov process and finally identifying the best range of privacy budget values for a travel assistant that preserves privacy while retaining data utility.

Contents

List of Figures

List of Tables

1 Introduction

This paper's main objective is to safeguard privacy concerns in travel aides so that users can travel without worrying about their personal information being compromised. But before looking into exactly what the privacy issues related to a Travel Assistant are, this chapter will provide a brief look into what a Travel Assistant is and why it is needed.

1.1 Motivation

The development of technology has made traveling in cities more and more exhausting. It takes substantially longer when there are numerous options from various forms of transportation. Due to personal preferences, it may be important to select a specific method of transportation or route. When traveling frequently, it is impossible to manually prepare ahead to prioritize comfort over expense or vice versa. People frequently follow their regular commute routes without even looking at the latest traffic reports (33).

Many of these problems can be avoided with the aid of travel planners, but using one to find the best recommended route is only possible at the start of the trip (32). Even after making advance arrangements and taking the required steps, the most exasperating undertaking is attempting to beat traffic problems, extraordinary accidents, or new and emergency constructions and arrive at the destination on time. With the recent development in wireless communications location-based services (LBS) can offer constant traffic or route updates with respect to the user's present location to make dealing with such everyday concerns easier and can assist in resolving unforeseen problems while traveling (19). Frequent route updates necessitate ongoing user location sharing with third parties and location service providers, raising privacy hazards (45). Therefore today's necessity is for location-based services that are privacy aware.

1.2 Travel Assistant

Unlike trip planners, which offer support only at the beginning of the journey, Travel Assistants are these location-based services (LBS) that continuously aid the user from the beginning of the journey, during the journey, and until the end (32).

Travel assistants are smarter versions of travel planners, offering real-time updates on the routes chosen. Passengers or users can get real-time information about the route, such as the estimated time to the next stop on the route or the estimated time to the traveller's destination, via travel assistants. Giannopoulos has listed multiple benefits of on-route planning such as the possibility to induce people to travel at non-peak hours, use more 'traffic friendly' destinations, share vehicles with other travellers, reducing journey-time uncertainty, making real-time route changes to avoid congestion, choosing the most appropriate connections/interchanges thus maximising use of spare capacity, minimising waiting times, and increasing 'Inter-modality' in passenger transport services (19). Thus, with so many advantages, a Travel Assistant is a necessity to keep up with today's fast-paced world.

1.3 Challenges

While location-based services have shown to benefit both individuals and society, the increasing exposure of users' location data creates serious privacy concerns. The user must provide his current location information to the LBS server in order to receive the required service. Users may face major threats if this information enters into the hands of malevolent opponents which can be humiliating or even dangerous, as in situations of stalking or burglaries that get induced by knowing when people were not present in appropriate locations (7).

When considering a travel planner, the location is only shared once when the service is queried, and simple privacy protection mechanisms such as the obfuscation or perturbation of the location data are sufficient. When it comes to a Travel assistant, however, the location data is updated continuously to the LBS server and must be protected from the beginning of the journey, during the journey and until the end of the journey. The risk to the user increases as a result of the continual location sharing, which discloses significantly more information about the user's whereabouts and trajectories over the entire journey duration.

Defense mechanism applied against travel planners cannot be used for a Travel assistant since the continuous update of the location enhances the correlation between the various location data making defense against location correlation more challenging.

1.4 Approach

There are various approaches in which LBS servers protect the static location queries including encryption, anonymity, etc. These methods allow the data collectors to collect the original and accurate data and implement location obfuscation, perturbation or encryption techniques helping them to store encrypted data in the LBS servers thus safeguarding original data. In some cases, LBS servers themselves can act as an adversary and extract the original data from the encrypted values. Differential privacy, introduced by Cynthia Dwork *et al* in 2014 (16) enables the data collectors to draw conclusions and overall statistics from the individual data. Thus, instead of recording individual data, LBS retains the conclusions and overall statistics of the data such as aggregates, averages, etc (13). However, given that users today are unwilling to share the precise location even for statistics collection, perhaps data collectors might be seen as a threat (48). The trust difficulties with data collectors tampering with the data are resolved by local differential privacy where in the statistics of the individual records are calculated even before passing it to the data collectors. However even in LDP, if the same statistics are retrieved from the data collectors multiple times, adversaries can infer the original data.

The Google's Randomized Aggregatable Privacy-Preserving Ordinal Response (RAPPOR) (49) technique modifies the location data at two different layers, making it more challenging for attackers to deduce the original value. This approach alters the initial value with a predetermined parameter and alters the disturbed value with an instantaneous parameter so that even if an opponent is able to deduce the disturbed value, it will still differ from the initial value, protecting against advance adversaries.

However, even RAPPOR can be subverted because of the strong correlation between location values, which gives the attacker more leeway than anticipated to deduce background information. This work examines a method of location prediction using a Markov model with the goal of reducing the correlation between various places in order to lessen the likelihood of adversary assaults caused by location correlation.

Thus our proposed methodology comprises of a location prediction algorithm based on a Hidden Markov model to suppress the correlation amongst different location data, and a location protection algorithm based on Differential Privacy with RAPPOR to protect the privacy of the location data.

1.5 Contributions

In this paper, we suggest using a Markov model to predict the locations the user might travel, hence reducing the correlation of the original location data, thus boosting the privacy enabled by the RAPPOR mechanism. The Markov model predicts data based on the current state only and thus reduces the correlation on the previous states protecting the location correlation.

The main contributions of this paper are as following:

1. Designing and implementing a location prediction algorithm based on a first order Hidden Markov model to predict the locations of the users considering only the current state and not the previous states, hence masking the auto-correlation of the continuously updated location data.

2. Evaluating the efficiency of the first-order Hidden Markov model to predict location with reduced correlation amongst the location points.

3. Evaluating the efficiency of the proposed approach of using RAPPOR mechanism as the location protection algorithm along with Markov Model for continuous location prediction, which provides both a one-time and a longitudinal privacy framework.

4. Visualizing and understanding the trade-off between utility and privacy along with the management of the privacy budget parameter ϵ value.

1.6 Road map

The rest of this dissertation is organised as follows. Chapter 2 reviews existing work in the field of travel planners, travel assistants and privacy preserving mechanisms. Chapter 3 describes the design and architecture of the privacy preserving mechanism in detail. Chapter 4 describes the features of the travel assistant system and challenges faced during the implementation. Chapter 5 evaluates the 3 different experiments performed during the research. Chapter 6 will provide a conclusion as well as a section on future work.

2 State of the Art

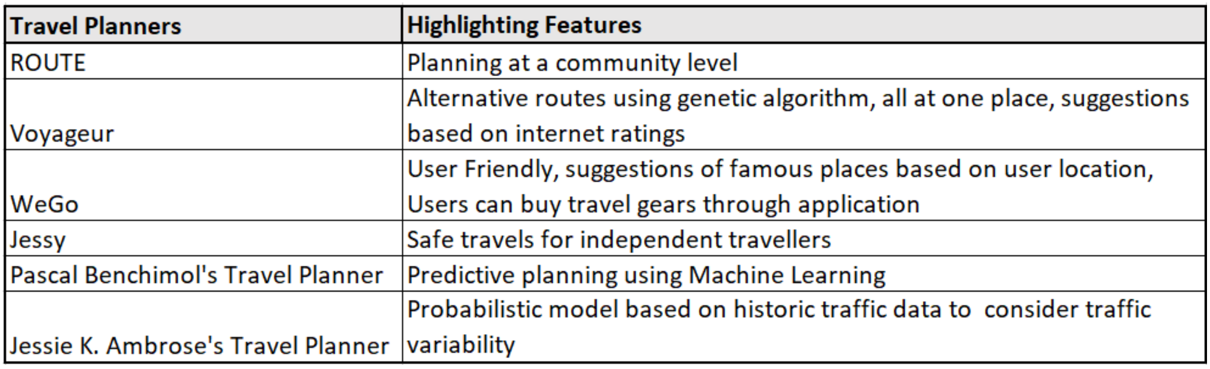
At the moment, privacy protection for location-based services is a major focus. This chapter first reviews existing travel planners, then existing travel assistants. It then highlights their privacy threats, before examining the various existing privacy protection technologies. Additionally it discusses differential privacy in more details, presents the trade-off between utility and privacy and finally concludes with the research question addressed by this work.

2.1 Travel Planners

A journey planner, trip planner, or route planner is a specialized search engine used to find an optimal means of travelling between two or more given locations, sometimes using more than one transport mode. Initial versions of a travel planner computed a route with basic optimizing features such as fastest or cheapest but gradually travel planners considered various parameters before planning a route such as user accessibility preferences (wheelchair accessible, step-free), special interfaces for visually impaired users, etc (39). Although nowadays travel planners also include hotel availability, passenger preferences when selecting the modes of transportation, and even the real time delays of public transportation.

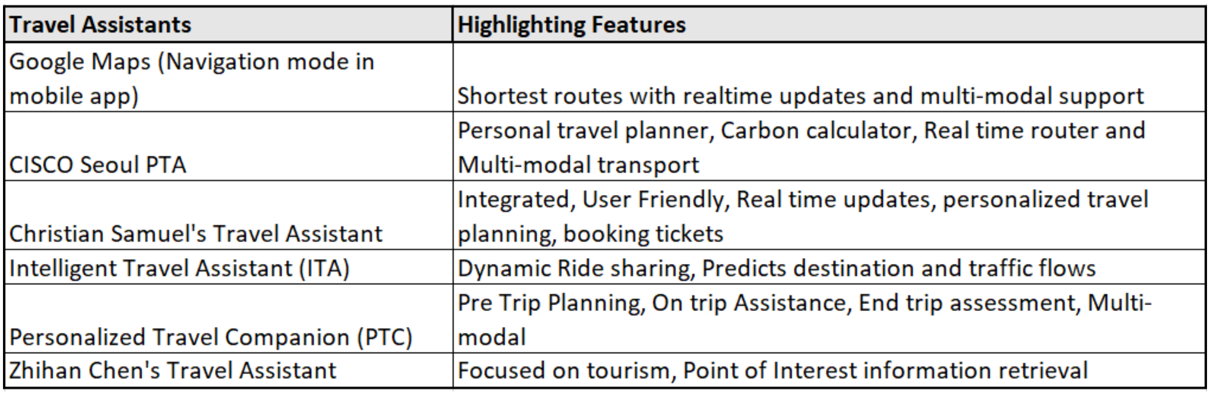
Basically planners provide a map at the start of the journey with the instructions that the user would follow during the journey. Receiving an origin and destination place as input and in return providing a set of possible trips, preferred departure and arrival time along with the best travel mode, travel planners typically provide a static plan with static information and do not interact with users once the plan is delivered. Due to the unavailability of these planners during the journey, plans supporting multi-modal journey are not efficient since multi-modal travelling requires pre-trip planning along with continuous on-trip information and end-trip assessment (32) which travel planners fail to provide. This paper categorizes all the location based services that do not provide continuous on-route assistance as travel planners.

Authors in (6) have considered the traffic variability in account before planning the set of possible routes. Using historical traffic data, 3 different routes associated with different



Travel Planners	Highlighting Features
ROUTE	Planning at a community level
Voyageur	Alternative routes using genetic algorithm, all at one place, suggestions based on internet ratings
WeGo	User Friendly, suggestions of famous places based on user location, Users can buy travel gears through application
Jessy	Safe travels for independent travellers
Pascal Benchimol's Travel Planner	Predictive planning using Machine Learning
Jessie K. Ambrose's Travel Planner	Probabilistic model based on historic traffic data to consider traffic variability

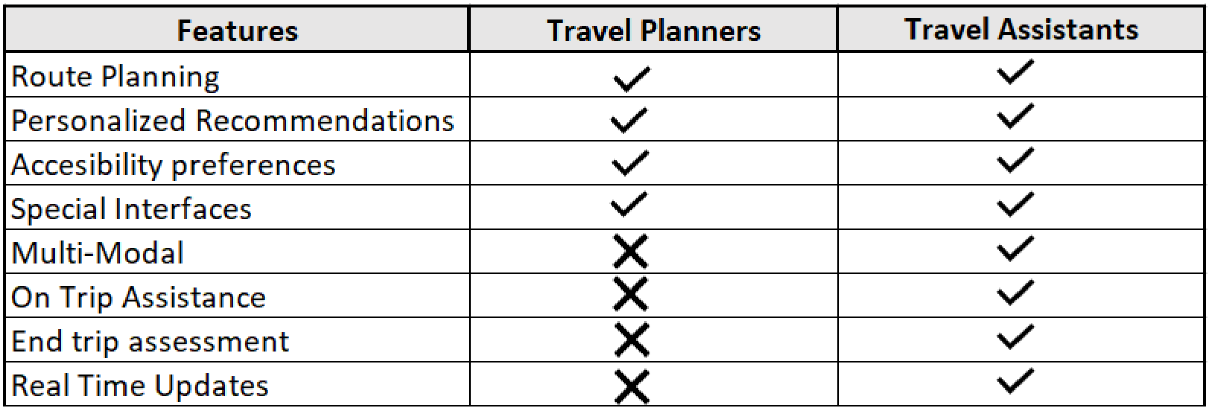
Table 2.1: Travel Planners with highlighting features



Travel Assistants	Highlighting Features
Google Maps (Navigation mode in mobile app)	Shortest routes with realtime updates and multi-modal support
CISCO Seoul PTA	Personal travel planner, Carbon calculator, Real time router and Multi-modal transport
Christian Samuel's Travel Assistant	Integrated, User Friendly, Real time updates, personalized travel planning, booking tickets
Intelligent Travel Assistant (ITA)	Dynamic Ride sharing, Predicts destination and traffic flows
Personalized Travel Companion (PTC)	Pre Trip Planning, On trip Assistance, End trip assessment, Multi-modal
Zhihan Chen's Travel Assistant	Focused on tourism, Point of Interest information retrieval

Table 2.2: Travel Assistants with highlighting features

mean/variance are selected based on the probabilistic model of travel times. Users can choose the consistent travel times trading with the routes that yield high variability in travel time. ROUTE developed by (4) is mainly focused on authorities, mobility service providers along with travellers thus, creating a planner at community level. ROUTE is focused on smart cities to enable authorities to define constraints on movements to manage the current pandemic and thus slow down the spread of the virus and speed up the recovery from the pandemic. Akshen Kadakia *et al* have presented Voyageur (22), where they argue travellers can find the best of everything at one place based on internet ratings. It uses a genetic algorithm at the back end to develop alternative routes for a single plan, and considers preferences, and distances in between different halts promoting efficient planning. Pascal Benchimol *et al* have designed planners with predictive planning using Machine learning to predict the passengers onboard to help the travellers plan accordingly. WeGo (31) is another application which allows backpackers to plan their trips in a very user friendly way with the highlighting feature i.e., to show famous places nearby based on the user's current location. WeGo includes other unique feature such as users can buy travel gears or earn a reward for users whenever they suggest a cuisine.



Features	Travel Planners	Travel Assistants
Route Planning	✓	✓
Personalized Recommendations	✓	✓
Accesibility preferences	✓	✓
Special Interfaces	✓	✓
Multi-Modal	✗	✓
On Trip Assistance	✗	✓
End trip assessment	✗	✓
Real Time Updates	✗	✓

Table 2.3: Location Based Services Important Features

2.2 Travel Assistants

A travel assistant differs from typical map-based direction finders or trip planners by offering assistance throughout the journey and not only at the beginning. It continuously collects the current location data of the user, and updates the real time details such as traffic, or roadblocks, based on the current location by running continuously at the background and staying connected to the user until the end of the journey. The varying features for travel planner and travel assistant are listed in table 2.3.

Currently there are various web-mapping services like **Google Maps** which provide the shortest routes along with the real time updates of accidents, roadblocks and traffic issues throughout the journey. Introduced in 2005 by Google Inc., it is being used widely for travelling from one point to another anywhere throughout the whole world. Google Maps collects and stores data about the users such as the search terms entered, IP address and the latitude and longitude co-ordinates. Google uses encryption for data privacy when in transit and has secured access. It sets a cookie NID in user's browser to optimise services and to provide user personalized services (20). Though Google Maps on web browser works just as a planner but the Navigation mode in mobile application provides real time updates with multi-modal transport support.

Users can protect the data shared with Google Maps in different ways such as by opening the Google Maps with incognito mode in a browser or using Google Maps without signing in. Also, the privacy settings can be modified to control what data can be shared with the user. Sharing less data, on the other hand, results in fewer tailored updates, and does not provide the real-time support and notification from a travel assistant.

The Seoul PTA (12) incorporates "virtual assistant" features that provide transit guidance based on user preferences and trip context (for instance, whether time is more critical than expense for a particular trip), employing real-time traffic and public transportation information. It consists of 4 key features: Transport Information Service

(Public transportation details), Personal Travel Planner (considering user preferences), Carbon Calculator (to manage carbon footprint and travel modes) and Real-Time Router (reroute during a trip in case of disruptions). If a traffic accident occurs while a user is in transit, for example, the service will remember the trip's origin and destination, and reroute proactively.

The Intelligent Travel Assistant (14) implements dynamic ride sharing, wireless communications to send and receive traffic details for assessing traffic conditions while en-route using Global Positioning System (GPS) to track the user's current location. In some cases, when the destination of the user is not available, the authors of this paper propose to predict the future location of the users based on the periodic patterns i.e., by detecting motion patterns amongst the users such as travelling time on weekdays from home to work, and vice versa. Also, to ensure the reliability and credibility of the routes, this system also predicts traffic flows using artificial neural networks.

Another travel assistant explained by Christian Samuel (33), includes integrated inter modal travel information services i.e., new components and interfaces. It mentions the issues faced when multiple interfaces are combined to create an integrated itinerary at one place. Here inter modal includes public transport, personal cars, walking, car-sharing as well. The model architecture of this paper comprises a lot of third-party service providers in its architecture, and shares the user data with all of them, which increases the risk of misuse of private data. The absence of a distributed approach shows the dependency on the centralized server for all the data-related tasks, which makes the server high at risk for adversary attacks. Also the design requires accurate location sharing with the server/data collector which in cases of adversary attacks can fall in malicious hands. The paper fails to explain any location privacy protection algorithms.

The Personal Travel Companion (32) focuses on extensive pre trip planning, personalized multi-modal journey with continuous on-trip assistance and end trip assessments. Along with the integration of in-car navigation and public transport instructions, it helps the user in guiding until the exact destination address, even after getting off at the last public transport stop. Different modes of transport like Tube, Tram, Rail, Bus, Cycle, as well as pedestrian are covered, with options for users to choose the preferences. Using Bluetooth beacons for indoor positioning and GPS for outdoor has made the accuracy and connectivity stronger and allowed door-to-door assistance.

Zhihan Chen's Travel Assistant (11) focuses on tourist assistance providing basic features such as map display, map positioning, geo coding as well as interesting features such as point of interest information during journey.

2.3 Privacy Threats and Risks

In spite of such a great progress in Travel assistants, there are many issues which are unaddressed yet because of the impact to the utility of the services. Privacy is one such issue which is very much in conversation right now. Privacy can mean different for different people at different times. The data at risk can have different sensitive attributes like personal, commercial and research. It is constantly changing and thus these different characteristics of privacy needs to be addressed in the privacy protection mechanisms. Any adversary or third party should not be able to access individual's location data. Leakage of location details can be embarrassing for some, can even be harmful for some. In (48), Martina Ziefle et al, have concluded that users find location data and lifestyle habits as too personal to be shared. Location data can be extracted from the services, media, etc which do not even contain location information in it. For example, A photo posted in social media can be giving location information even if it is not tagged with geographical information but if it contains any unique features. Applications which use accelerometer details without the location details are also fully capable of providing complete trajectory of a user (7).

Location data is collected by all location based services to improve their services. There have been many cases where big companies such as Apple, Google, Microsoft have been reported to store location data of users without users knowledge and conduct privacy breach but they have denied to be aware of the act and claimed to be programming glitches and acted immediately by updating software and thus no direct harm to individuals (7). Some share the location data with the third party services supporting their business, or for advertisements and sponsorships. Due to privacy preserving mechanisms and regulations, these companies do anonymize the data before sharing it with third party. Even if communications are pseudonymous, the spatio-temporal correlation of location traces may serve as a quasi-identifier 2.1. For e.g., work and home locations uniquely identify most of the population of US (18). Studies and evaluations performed in (18) shows the privacy erosion problem arises when even a small amount of information is shared with such services.

But why do these services store user's details? Google mentions in (20) that it might share a user's personal information with other companies, organizations or individuals outside of Google for different reasons. Some free services share location to generate revenue with location-based advertisements (18). Active positioning used in Travel Assistants requires continuous sharing of location details with the trusted server which is very dangerous, compared to the passive positioning where location data is not shared again and again. Location data, when collected repeatedly, forms a time series pattern which are easier to re-identify using various data mining techniques. Authors in (7) have demonstrated that continuous movement data collection is easier to re-identify than episodic since the missing data, and large intervals between the data create more uncertainty.

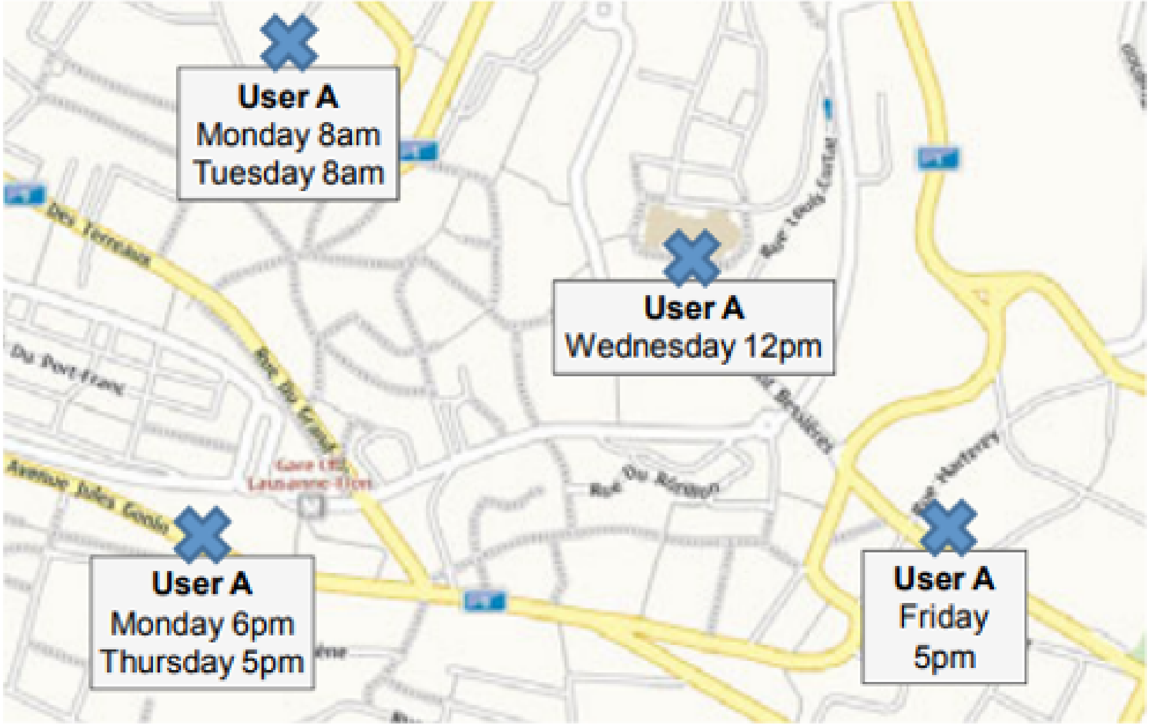


Figure 2.1: Quasi-identification of a user A whose identity is not revealed (18)

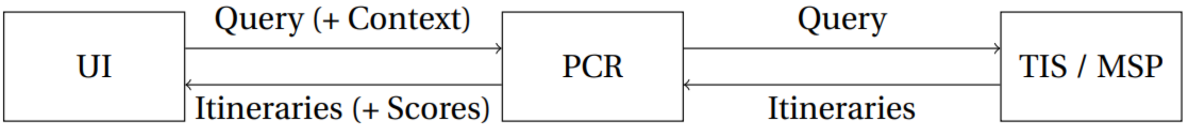


Figure 2.2: Travel information architecture with PCR (33).

In Samuel's approach, figure 2.2 shows how the privacy is secured when UI shares only the query with the Personalized Context aware Recommender (PCR) system and not the context to external Mobility service provider (MSP) to preserve user data privacy, but adversaries nowadays are able to infer background knowledge from the query data as well (21). In a query, it has the format (*UID, t, loc, f*). UID is the user identity (protected), t is the query time, loc is the query location, and f is an interesting keyword that users want to retrieve from LBS servers. Mingming Guo and *et al* have described how improper query protection might make the query search functionality into a weapon for attackers (21).

The privacy policy at Whim (38) another mobility service provider, explains the data they collect from the users, how they use the data and what data is shared with third party users. It mentions that along with the travel and trip data, they may collect data such as the IP address and many more which is not even specified clearly. If the non-personal data is stored with personal data, ultimately making both non-personal and personal data identifiable to the individual. Also, to improve the usability of the service, personal data is shared with third party users, which complies with the Data protection law, but if this data is lost by any of them to the hands of malicious attackers, then can be harmful.

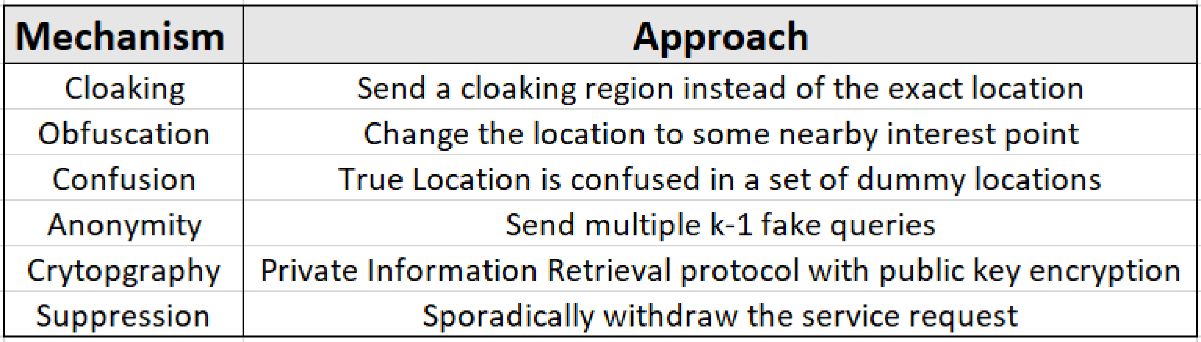
There have been various ways (40) by which users themselves have tried to protect privacy while using such applications, like searching destination with fake source location or by modifying privacy levels to zip codes or city names. However, all these measures do not provide accurate results and require manual efforts of entering source locations.

While (25) suggested the mixing of a user's location with other users' locations in a query so that it's hard to identify the exact or real location of the user, it is not a suitable approach since there will be more damage if all the users' locations are exposed. (40) also explains that the measures where either identity of the user or the location is hidden, it is not difficult nowadays to find out one from the other where every application asks to sign in before the service is provided.

2.4 Privacy Protection Mechanisms (PPM)

Privacy-preserving mechanisms restrain various Location-Based services preventing the misuse of user's private data, but they are rarely used since users are unaware of the privacy threats as they do not intentionally share their location. There are various approaches which protect the privacy for such location-based services using location suppression, obfuscation, confusion and cloaking techniques before sharing the location information to the trusted/untrusted servers. Suppressing location information withdraws the location information from the service during intervals of time. This technique is typically paired with some other privacy mechanism to achieve best results. For instance, location suppression is used to conceal or stop sharing the precise location when using another approach. The location obfuscation or perturbation method includes sharing some nearby point of interest location instead of the exact location which reduces the accuracy of the system. The confusion method involves a set of multiple dummy locations when sharing the single true location to the trusted server or the location service providers. This results in collection of large amounts of noisy data along with the useful data which is not suitable in real world. The location cloaking sends the entire region instead of the exact location during querying the server which utilize long server query processing times (13) (44).

K-anonymity is one of the anonymity method which states that in a set of k elements, no information about k can be inferred from the rest of k-1 data. The basic principle behind k-anonymity is that the user sends to the LBSs provider a collection of k objects including one actual position and k-1 false positions, and so the true location can be hidden in k locations. Haoyu Liu *et al* have proposed privacy preserving k-anonymity scheme (PPKS) which uses probability density function to generate virtual k-1 locations. The chances of



Mechanism	Approach
Cloaking	Send a cloaking region instead of the exact location
Obfuscation	Change the location to some nearby interest point
Confusion	True Location is confused in a set of dummy locations
Anonymity	Send multiple k-1 fake queries
Crytopgraphy	Private Information Retrieval protocol with public key encryption
Suppression	Sporadically withdraw the service request

Table 2.4: Privacy protection mechanisms

quasi identification are higher with k-anonymity techniques and the single anonymizer can act as the single point of failure. Zhang *et al.* in (46) have proposed a system which uses a dual k-anonymity (DKM) mechanism combined with dynamic pseudonyms to overcome the majority of the privacy concerns mentioned in the obfuscation, perturbation and cloaking techniques previously. Here, when sending a query request, a user is assigned a dynamic pseudonym, which is sent to the different anonymizers along with K-1 additional locations. These query requests are sent to the third-party location service provider for another query, and the results are returned to the user via multiple anonymizers. Thus, it is hard for an attacker to identify the user due to dynamic pseudonym. The advantage of the DKM scheme is that a user trajectory cannot be acquired from the Location service provider or a single anonymizer. In addition to offering more protection to the user's trajectory privacy, it provides an effective approach for a single anonymizer's performance bottleneck. However, anonymization techniques are vulnerable to homogeneity attacks where all the k-anonymized values are similar to each other. Moreover, if the adversary has access to the background knowledge, k-anonymity fails to protect the privacy (10).

Cryptography approaches generally ensure strong privacy by using the Private Information Retrieval protocol, which transmits the location data by using public key encryption. Yuwen *et al.* in (30) use an efficient and reliable cryptography approach to ensure the privacy of the transmitted location data, but the delay due to the cryptography approach reduces the utility of the system, and this approach is thus not preferred over other mechanisms.

In summary, in the big data and modern technology era, PPMs such as cryptography, k-anonymity provides privacy protection to a certain limit. Issues such as slowed systems, delayed responses, quasi identification attacks were increased which demanded new mechanisms for privacy protection. Existing location-based services have taken precautions to protect personal information, but there is always a trade-off between privacy and performance such as long query processing times due to cloaking, large amount of useless or dummy data and memory storage due to confusion, inaccuracy of the location due to obfuscation or perturbation, etc (13).

To balance the trade-off between utility and privacy, Cynthia Dwork *et al.* have proposed Differential privacy which resolves most of the aforementioned issues (16) (10). Instead of sharing the exact data, differential privacy shares the aggregated statistics and increases the uncertainty of the data by adding noise to it when sharing with third party service providers.

2.4.1 Differential Privacy (DP)

as explained in (16) is a privacy-preserving mechanism that adds statistically-controlled noise to the user data, thus restricting the identification of the user data by an adversary or any third party. As shown in figure 2.3, the weights of edges are modified in such a way that the total weight of the topology remains same and yet the values are not real for every edge. Differential Privacy eases the process of understanding the trade-off between utility and privacy by introducing the privacy budget ϵ in the design. This privacy budget is the measure used to decide the degree of disturbance. It is used to denote the level of privacy risks since increase in privacy budget causes increase in privacy risk. DP is mainly classified into two types: Local and Global. Global Differential Privacy (GDP) includes centralized approach as shown in 2.4 sending raw data to the trusted data collectors (for e.g., Mike in the image as the trusted curator) and then data collectors extract statistical inferences to send to the untrusted third parties. Whereas Local differential privacy (LDP) applies DP to the raw data at client side and sends statistical conclusions to the data collector (for e.g., Bob in the image as untrusted curator) as well as servers. LDP modifies each and every bit of the data record in such a way that only the obfuscated data is gathered and manipulated. When the amount of data is large enough, the utility becomes acceptable, so the data collector can still find out the accurate frequency of specified attributes.

Global differential privacy (GDP) provides more accuracy as compared to the LDP but is used only when we can guarantee the privacy with data collector. However LDP is more privacy-preserved than GDP and preferred for the designs with distributed systems. GDP is more accurate than LDP, however LDP is as accurate when utilized with large amounts of data. Because of this, LDP is used by Google and Apple to protect their users' data. Additionally, it is more suitable for a distributed architecture since it does not require any trusted and centralized data curator or server and thus the best mechanism for a travel assistant. There are various mechanisms used for perturbation based on the dataset type and the approach of differential privacy. For GDP, Laplace is used for numerical data whereas Exponential is used for categorical data. In LDP generally randomized response mechanism is used since all the users can perturb the data at their end and then send the randomized data to the data collector.

Table 2.5 shows the existing applications of Differential Privacy in a variety of areas wherein

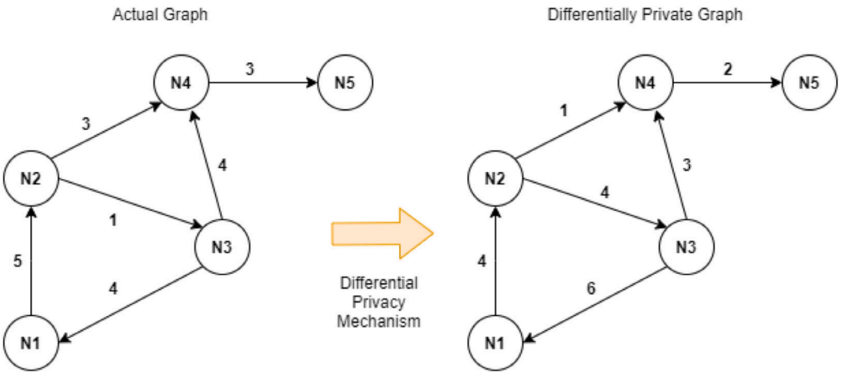


Figure 2.3: DP applied in graph topology preserving edge weight functions (8)

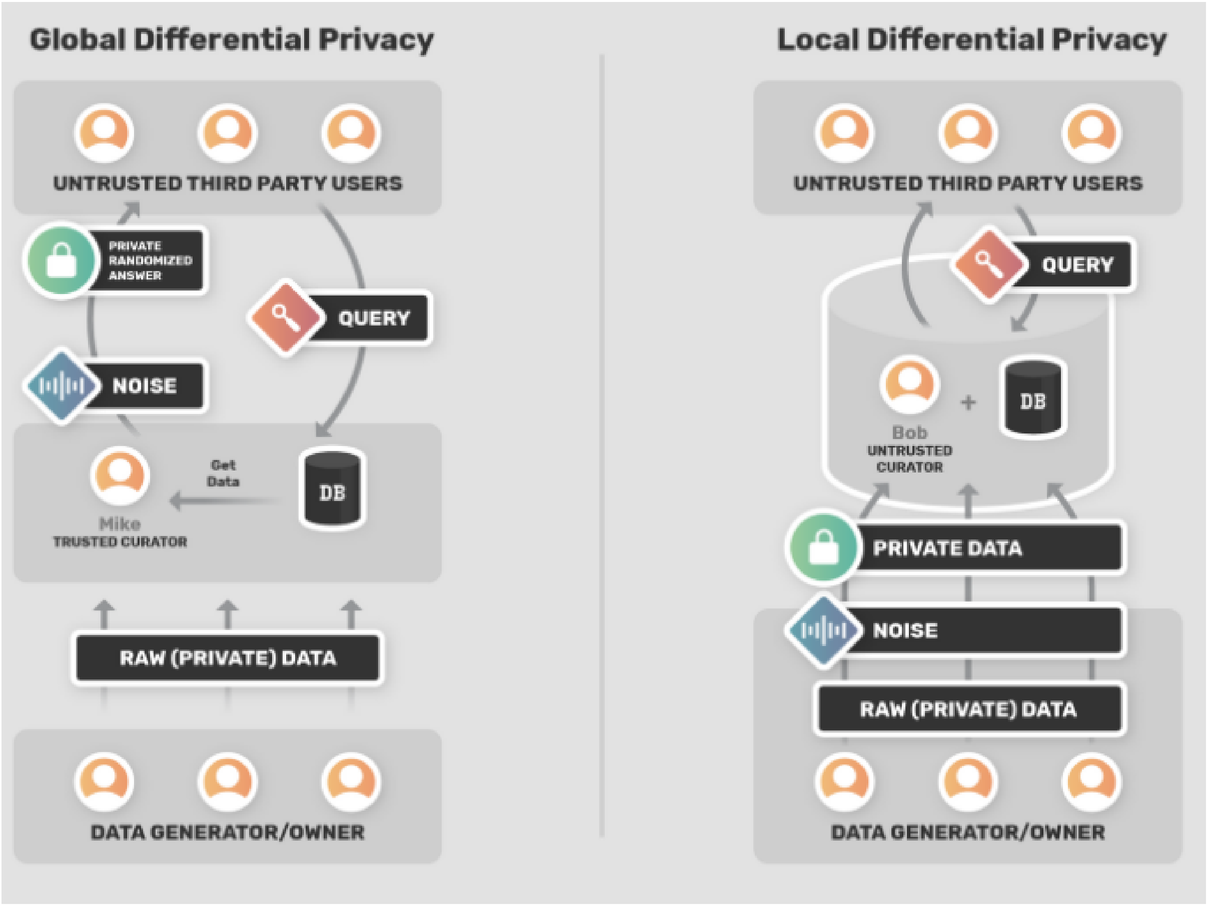
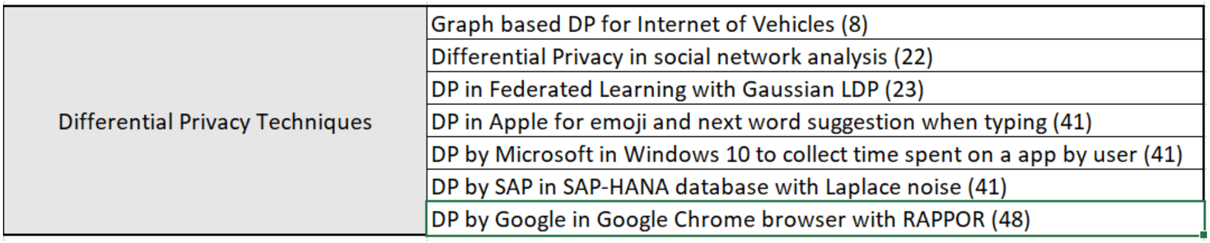


Figure 2.4: Global Differential Privacy vs Local Differential Privacy (17)

Table 2.5: Differential Privacy in real-world applications



Differential Privacy Techniques	Graph based DP for Internet of Vehicles (8)
	Differential Privacy in social network analysis (22)
	DP in Federated Learning with Gaussian LDP (23)
	DP in Apple for emoji and next word suggestion when typing (41)
	DP by Microsoft in Windows 10 to collect time spent on a app by user (41)
	DP by SAP in SAP-HANA database with Laplace noise (41)
	DP by Google in Google Chrome browser with RAPPOR (48)

sensitive user information is at risk (23).

LDP has been applied widely at internet-scale by top companies such as Apple, Microsoft, SAP as well (43). Apple applies LDP to the process of emoji and next words/phrases suggestions when typing (43). Their LDP mechanism used Count Mean Sketch and Hadamard techniques to inject noise, encode and privatization. Count Mean Sketch encodes varied size data into a fixed size matrix, encodes it using variations of SHA-256 hash into vectors, flips each coordinate of the vector with a probability of $1/(1 + e^{\epsilon/2})$, where ϵ is the privacy parameter, whereas the Hadamard technique sends only 1 bit of the entire row to the server to reduce communication cost. Since the noise generated by differential privacy tends to average out across a large number of donations, making it simpler to identify the single user, Apple places severe limits on the amount of contributions per user to safeguard their privacy (1). Microsoft applies LDP in Windows 10 implements DP which includes slight changes in user's data to protect from memoization issue i.e., when adversary can infer data when the query hits multiple times with the same perturbed data but the approach fails when the value changes substantially. By injecting the Laplace noise mechanism into each record of the data before responding to any query, SAP develops LDP to ensure privacy in its database, although it only supports numeric data since Laplace noise only functions well with numeric data (43). Location data is a great example for LDP because for a user it does not matter if the city name is exposed but will definitely matter if the apartment number is exposed which can be managed by changing the ϵ value.

Differential Privacy is applied in location-based services for protecting location privacy in both i.e., for static location queries such as in travel planners and also when sharing location in continuous queries such as in travel assistants. In (37), DP is used to generate an algorithm which perturbs a set of locations adding Laplacian noise and forming a perturbed region instead of a single location and dynamically calculate the perturbation range at different timestamps in continuous queries.

There have been many studies performed to strengthen the location privacy by addressing the location correlation problem. Location correlation in any trajectory results due to the strong relativity between adjacent location points. PPMs perform stronger if the correlation between the location points are reduced. The location correlation is reduced or measured by assuming location trajectories as a Markov process.

Xingxing Xiong *et al* captures temporal correlations with $\epsilon - \delta$-LDP where ϵ is the privacy budget for generalized randomized response (GRR) and δ-location set is the set of user's all possible locations at each timestamp satisfying Markov process. Lu Ou *et al* has addressed the important issue of multi-user location correlation problem in location based services by applying hidden markov model (HMM) with bounded differential privacy applying randomized response mechanism. This paper (27) suggested using HMM to measure and quantify correlation between two user's trajectories and if acceptable then release those trajectories by transforming them using differential privacy.

LDP restricts the performance of the algorithm if the same queries are repeated over time that is, an adversary can infer the real value from the database if the same data is sent over and over again to the database using the same perturbed value. This allows LDP to use memoization, a process which answers the same queries with the same data. However, memoization also can let the adversary to find the real raw value after a point.

Trade-off between utility and privacy

To understand the trade-off between utility and privacy, lets first know what we mean by privacy and utility with respect to a travel assistant system. The privacy here means the degree of disturbance in the location data and utility dictates the availability of the data. The authors of (48) proposed a trade-off framework by using Pareto efficiency. Pareto efficiency, or Pareto optimality, is a state that no one can get better when no one else gets worse. To understand Pareto optimality with respect to a Travel Assistant, is a state where privacy can not get better if utility is not getting worse and vice-versa i.e., if there is increase in data availability and performance of the travel assistant then there is no decrease in the degree with which the location data is preserved. In other words, privacy is inversely proportional to utility. In LDP, if the privacy enhances, the loss of privacy is bound to decrease, which will lead to reduced data utility. Conversely, if the utility of data increases, the utility loss will decrease, which will inevitably lead to lower privacy protection. By definition, the whole system state is always Pareto optimal. The privacy and utility metrics are privacy loss and utility loss respectively.

When the privacy budget ϵ is changed as an independent variable, it will cause changes to the privacy measure and utility measure of the dependent variable. The larger the value of the privacy budget ϵ, the weaker the privacy protection, the greater the privacy loss, the higher the utility, the smaller the utility loss. In other words, ϵ-LDP, the privacy is determined by epsilon. If ϵ=0, which means exp(0) = 1, the true and perturbed values are similar to each other. Therefore, the privacy is ensured perfectly. However, when the ϵ=∞ there is no privacy guarantee. Thus, the choice of ϵ is critical in practice as the increase in privacy risks is in direct proportion to exp(ϵ).

2.5 Research Statement

When using privacy-preserving mechanisms there will always be trade-off in privacy and utility. Though there have been many studies showing this trade-off in various areas (23), to the best of our knowledge, this trade-off is not yet studied in the case of the travel assistant system. A travel assistant provides frequent real-time reports on the user's location and behavior, which can reveal a great deal of personal information. It is not clear to the customers how the privacy impacts the performance of the system. Thus, our research topic is to investigate the trade-off between utility and privacy with appropriate visualizations and studies. This paper will compare and show how the increased privacy directly restricts the performance of the travel assistant and allow users control over data sharing so that they can decide how much data to share for the system's required performance. It seeks to understand the trade-off by contrasting the impact of location correlation on the privacy protection mechanism.

Research Statement 1: Can we use differential privacy to demonstrate the trade off between the utility and privacy of a travel assistant?

Continuous real time updates can generate auto-correlation amongst the adjacent data points due to spatio-temporal relation between them. This correlation will degrade the performance of the privacy preserving mechanism and thus needs to be minimized to strengthen the privacy protection.

Research Statement 2: Can we use location prediction algorithm to reduce the correlation amongst the location data?

Existing privacy preserving mechanism fails to protect under memoization issues i.e., when same query hits the server repetitively in specific intervals of time.

Research Statement 3: Can we use any location protection algorithm to help defeat memoization issue and integrate it with location prediction algorithm to improve the performance of the privacy mechanism?

3 Design

In accordance with the aforementioned objectives, our design for a privacy-aware travel assistant focuses on the location prediction and location protection algorithms. Because of the large domain size, a user's location data can be safeguarded via a local differential privacy technique. Location data, as described in (43), can be generalized over a greater region and regarded as a categorical feature rather than a numerical property (5, 43).

To ensure strong privacy and support the local differential privacy approach, this design will be comprised of distributed computing where the user location data will be perturbed at the user devices itself and will be sent to the server or the data aggregator only after perturbation.

3.1 Location protection algorithm

When choosing the noise mechanism to protect location, Laplace mechanism cannot be applied since the location data is treated as a categorical variable and Laplace only applies to numerical data, and Exponential mechanism is applied for Global/Central Differential Privacy since it requires the design to follow a centralized approach, unlike ours where the users are distributed and there is no trusted third party server with access to all the records. Thus, for a distributed travel assistant where location data needs to be protected, RAPPOR uses randomized response mechanism to add noise.

As stated in (49) traditional randomized responses algorithms fail to provide privacy in case of memoizations. Multiple queries on the same perturbed data or user are also not protected by the traditional randomized responses. Thus, to overcome this problem, we will be using the RAPPOR mechanism to perturb the data which uses two perturbation levels, one is the permanent randomized parameter and the other is instantaneous randomized parameter. The permanent randomized parameter can be used to maintain the longitudinal privacy of the location data whereas the instantaneous randomized parameter will be helpful in one time attacks.

To create n-bit array from the location coordinates data assume A = a1, a2, ...an is the area

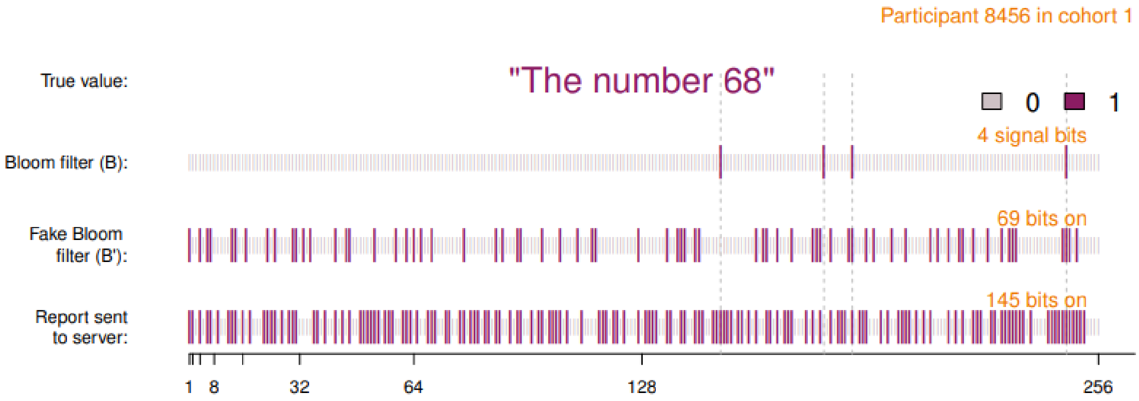


Figure 3.1: Life of a value in RAPPOR (49)

ID defined by the map-based segmentation, where n represents the total number of areas after the map is divided. Determine which ID the current location corresponds to after the participant receives the area ID from the map based segmentation. Suppose a_i (1 <= i < = n) be the ID with the current location. Then, a n-bit array, L (which denotes the current location of a specific user) is defined as,

$$L_j = \begin{cases} 1, & j = i \\ 0, & otherwise \end{cases} \qquad (41)$$

The image in figure 3.1 shows the true value '68' is modified to the bits array with 4 bits on i.e., only 4 bits have value equal to 1 and rest all of them are 0. In the first perturbation layer, this bits array B with 4 bits on will be perturbed statistically to generate bit array B' using the permanent randomized response explained further.

Permanent randomized response: Every bit i in the above n-bit array L, 0 <= i < k in B, is modified to a binary reporting value B' which equals to

$$B'_i = \begin{cases} 1, & \text{with probability } \frac{1}{2}f \\ 0, & \text{with probability } \frac{1}{2}f \\ B_i, & \text{with probability } 1 - f \end{cases} \qquad (49)$$

This B' is memoized and is always used in multiple queries of same data which helps in protecting from long term privacy attacks. Here, it is very important that every report on B should return B' as the perturbed value which helps to protect from adversary attempting to infer the averaging information from multiple noisy versions of it thus allowing multiple queries. The image in figure 3.1 shows the perturbed B' which has 69 bits on whereas the original array had only 4 bits on.

In the next layer, RAPPOR perturbs the perturbed array B' with randomized response mechanism calculated using instantaneous parameter to generate B" which the protects from the one-time attacks of the adversaries.

Instantaneous Randomized Response: Allocate a bit array S of size k and initialize to 0. Set each bit i in S with probabilities p and q as follows

$$P(S_i = 1) = \begin{cases} q, & \text{if } B'_i = 1. \\ p, & \text{if } B'_i = 0. \end{cases} \quad (49)$$

Finally, perturbed S is sent to the server to query the database. As you can see, the report sent to the server in the image has 145 bits on rather than 69 bits on i.e., in B' array there were 69 1s and now when the report is sent to the server it has 145 1s in the bits array.

Thus, this encoding method M as mentioned in (49) (41) satisfies ϵ-differential privacy such that the original value $B \in D$ and perturbed value $B' \in D$ and for any possible $S \subseteq Range(M)$, it gives

$$P[M(B) \in S] \leq e^{\epsilon} P[M(B') \in S](16)$$

. To ensure on route updates even during the journey, the application needs to continuously send current location data to the server for perturbation. But the location data sent for perturbation will not be the actual location data but the predictions of the first order Markov model results which will generate the locations which the user is most likely to visit.

However, RAPPOR also works best only when the user data does not change too often and when there is less correlation amongst the different user data. But in a Travel assistant, with the continuous update of location data to the server, there are high chances of correlation amongst two consecutive locations. If two different location values are correlated then an adversary can infer knowledge from the location correlation problem. In other words, the permanent randomized response parameter ensures longitudinal privacy only when the user data is not changing rapidly and if changing then the values should not be correlated.

To solve this correlation protection problem, we are investigating using a Hidden Markov Model to predict the user location rather than using the accurate/original location data. The Markov model and its variants are used in multiple existing designs (26) (28) to predict the location data to protect the location correlation problems when implementing differential privacy.

3.2 Location prediction algorithm

Markov model is a memory less random process uses actions and transition matrix which will be the movement and the matrix of probabilities will be the result of every movement. First

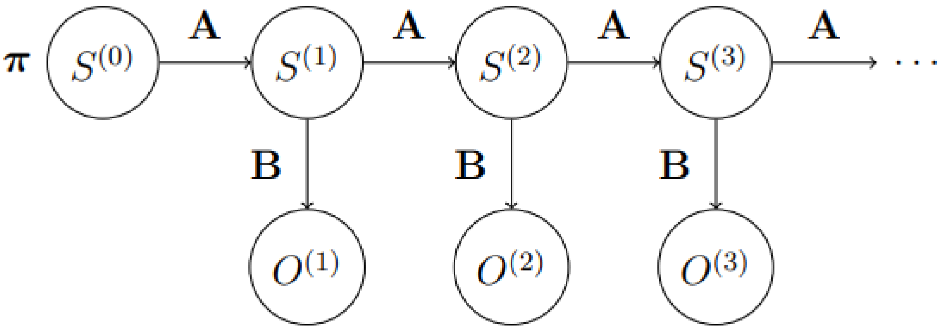


Figure 3.2: Hidden Markov model state and observations (29)

order Markov model uses only the value of the current location for prediction of future location and no dependency on the previous/older values.

Since the true location of the user is unknown, considering it as the state of the system we employ a Hidden Markov Model to predict next location from a given set of observations. The calculated values of the location coordinates are considered to be the possible observations of the system (29).

$$P(Xn+1 = x|X1 = x1,\cdots, Xn = xn) = P(Xn+1 = x|Xn = xn) \quad (42)$$

As stated in (42), in a finite state space Xn: n= 0, 1, 2,.. then *Xn=i* indicates that the object is in state i at time n. If for any n where n >= 0, the above equation is true then the process is a Markov chain.

The transition probability matrix is generated with the help of above equation for consecutive states. One-step transition probability can be expressed as the below equation:

$$Pi,j = P(Xn+1 = xj|Xn = xi) \quad (42)$$

When the above equation arranged in a matrix form we get,

$$P = \begin{bmatrix} P_{1,1} & \cdots & P_{1,j} & \cdots & P_{1,m} \\ \vdots & \ddots & \vdots & & \vdots \\ P_{i,1} & \cdots & P_{i,j} & \cdots & P_{i,m} \\ \vdots & & \vdots & \ddots & \vdots \\ P_{m,1} & \cdots & P_{m,j} & \cdots & P_{m,m} \end{bmatrix} \quad (42)$$

Figure 3.2 shows different states of the system as $S^{(0)}$, $S^{(1)}$, $S^{(2)}$ depending on the observations of the system denoted as $O^{(1)}$, $O^{(2)}$, $O^{(3)}$. In relation to our scenario, the states can be considered as future/next/predicted location whereas observation is the current location coordinates. As stated in (9), the state variable has to be in discrete domain but the observation variables can be in discrete or continuous state.

3.3 Application flow

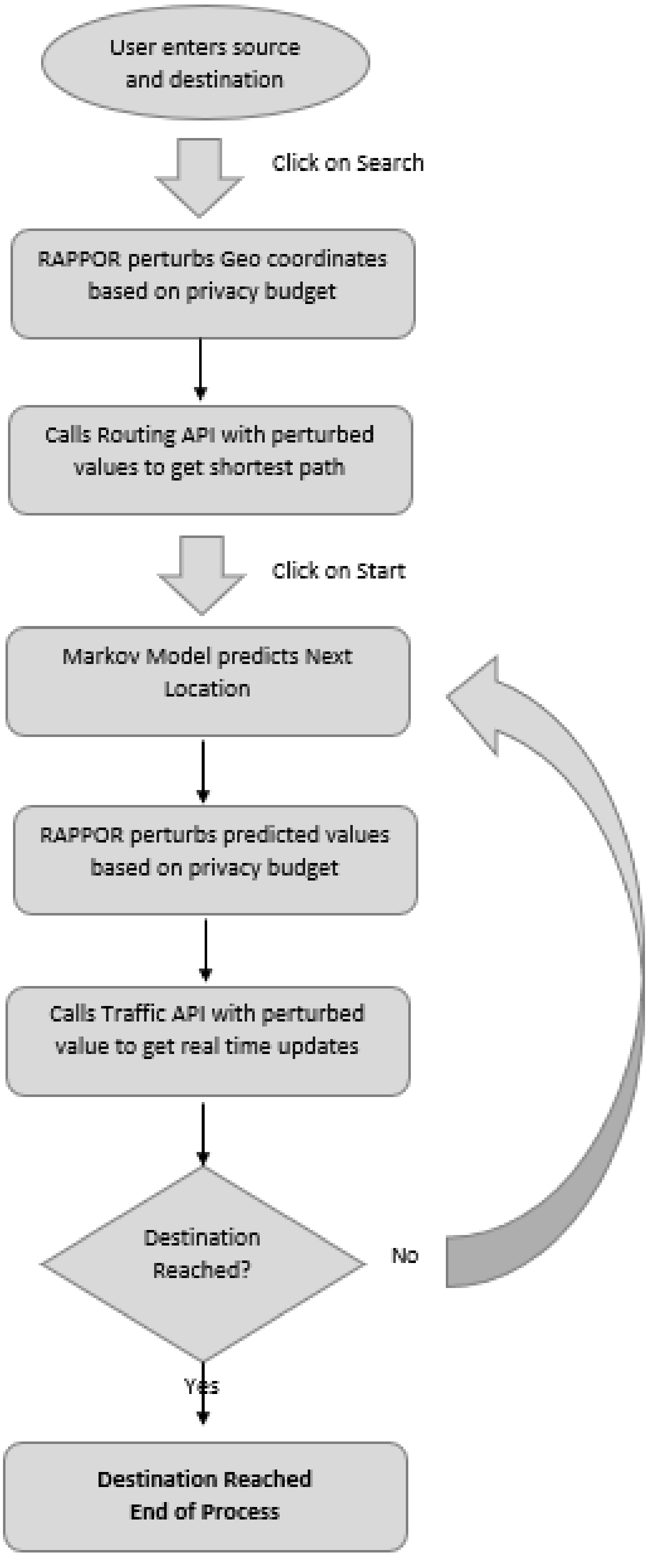

User enters source and destination
Click on Search
RAPPOR perturbs Geo coordinates based on privacy budget
Calls Routing API with perturbed values to get shortest path
Click on Start
Markov Model predicts Next Location
RAPPOR perturbs predicted values based on privacy budget
Calls Traffic API with perturbed value to get real time updates
Destination Reached?
No
Yes
Destination Reached End of Process

Figure 3.3: Architecture Flow of the Privacy Aware Travel Assistant

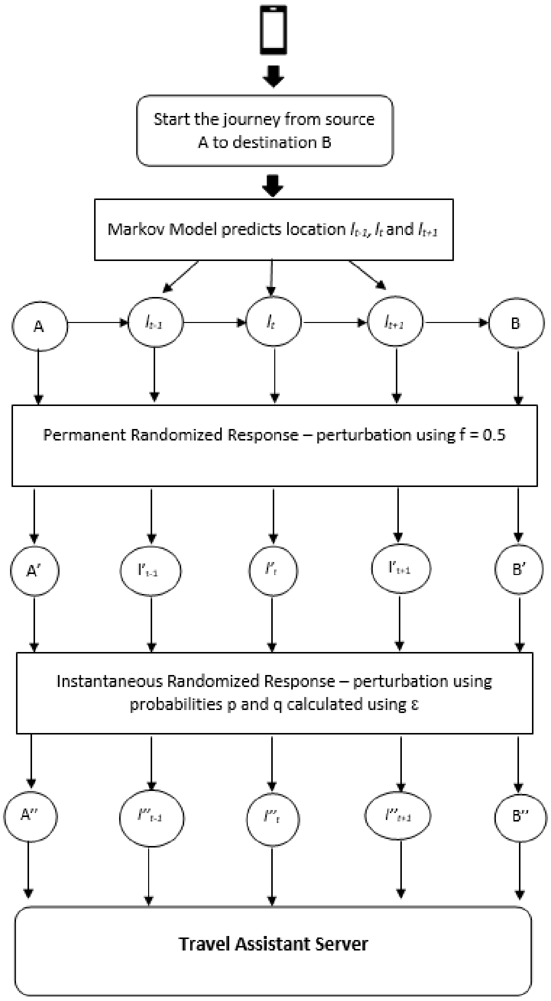

Start the journey from source A to destination B
Markov Model predicts location l_{t-1}, l_t and l_{t+1}
A
l_{t-1}
l_t
l_{t+1}
B
Permanent Randomized Response – perturbation using f = 0.5
A'
l'_{t-1}
l'_t
l'_{t+1}
B'
Instantaneous Randomized Response – perturbation using probabilities p and q calculated using ε
A''
l''_{t-1}
l''_t
l''_{t+1}
B''
Travel Assistant Server

Figure 3.4: Active positioning on click of Start button

3.4 Technology choices

Mobile Application handles the front end and manages the general flow of the application. User can interact with the application through this module to select the routes and get real time updates about the journey. The programming language chosen for the application is Python 3.8 version (2). This choice was made as it is a flexible programming language that allows for quick prototyping. The downside of Python's flexibility manifests with the slower computation times. However, the slower times are still satisfactory in regards to the scope of this dissertation. Kivy is the open-source (24) python library used to build the mobile application to ease the integration on the Android platform. Kivy does not support all the Android features but can be sufficient for the prototype version we are planning to build with this dissertation.

4 Implementation

This section provides a thorough explanation of how the front end screens and back end algorithms are implemented. In addition to the main working of the back end algorithms, this chapter explains the pre requisites required starting with the process of geohashing of location coordinates to bit arrays provided as an input to the RAPPOR algorithm for perturbation. These bit arrays are decoded back using the geohash decoding to generate perturbed location coordinates which are used to get the shortest path from the routing APIs. Real time updates about the journey are shared only when the user starts the journey by clicking on the Start feature available on screen which enables HMM to predict next location based on clustering and perturb the same using RAPPOR before collecting traffic updates from Traffic API.

The first screen in the application visible to the users is shown in 4.1 . It displays the source and destination text fields where the user can input the details of the location. By default, source information is taken from the user's location. When the user doesn't provide any information in the Source field, the IP address of the user is used to determine the user's present location, which is then used as the source. It is mandatory for the user to input the destination address. The user can click the Search button after entering the source and destination addresses.

4.1 Perturbation

4.1.1 Geo Hashing using Hilbert Curve

Amongst the very few RAPPOR implementations, in (35, 41) Differential Privacy with the RAPPOR mechanism is implemented along with map density segmentation mechanisms like Modified Hilbert Curve or Dynamic Hilbert Curve, which helps to map the 2D space to 1D space to ensure spatial correlation and generate binary bits array which are then passed as an input to RAPPOR's first perturbation layer.

When clicked on Search button, the geocode library in Python uses the source and destination addresses to produce the position coordinates and before providing these

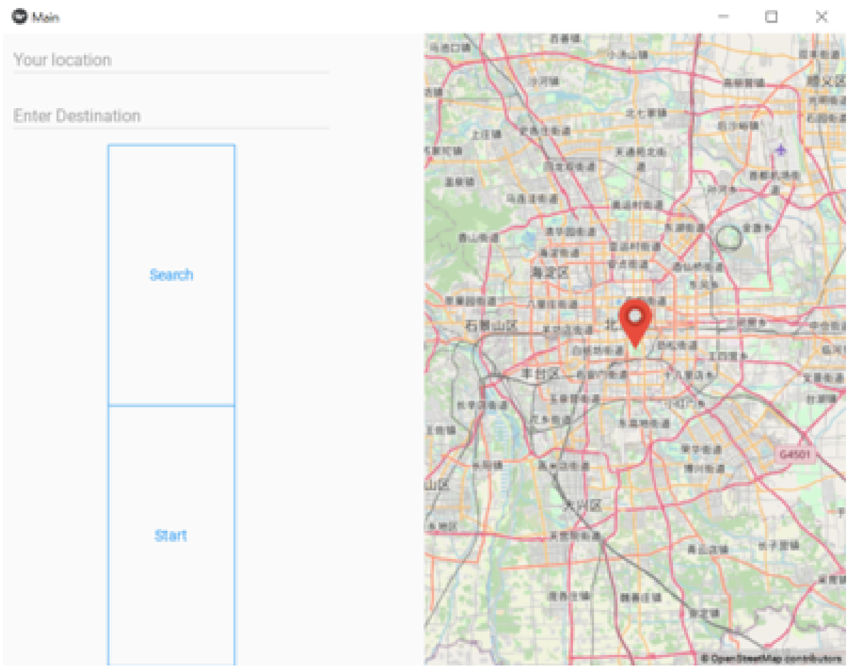


Figure 4.1: Home Screens of the Travel Assistant

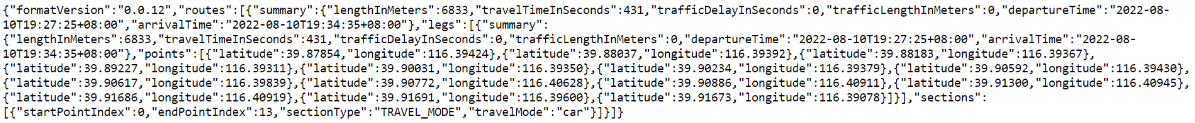
location's coordinates as an input to the RAPPOR mechanism, transforms them to a binary array. Traditional space filling curve, Hilbert curve, is used to convert the 2D mapping to 1D mapping and transforming the location coordinates into bit arrays. This space filling curve covers the whole 2D space in its range and thus includes every point on the 2D space in its 1D curve (3). Since the geohash-hilbert library uses the geo-coordinates but only generates geo hashes in BASE64 encoding using strings and integers both along with '_' and '@' as well, the geo-hashing of hash code to bits array is utilized in conjunction with it to construct the Hilbert curve encoding. Thereafter, the BASE64 encoded hash format is altered to a bits array by utilizing the binary values of each character to convert each character to a bit value.

4.1.2 Permanent Randomized Response

The RAPPOR method is then provided bits array of length 60 as input, and it alters the original bits at two levels before transmitting them to the location service provider. The RAPPOR's first layer, permanent randomized response, alters the bits based on the values of preset parameters (denoted as f). The constant value of f aids in maintaining the longitudinal privacy of the original bits.

4.1.3 Instantaneous Randomized Response

These bits, which were altered at the first layer, are then disturbed once more using the instantaneous parameter chosen with the probability computed based on the privacy budget,



```
{"formatVersion":"0.0.12","routes":[{"summary":{"lengthInMeters":6833,"travelTimeInSeconds":431,"trafficDelayInSeconds":0,"trafficLengthInMeters":0,"departureTime":"2022-08-
10T19:27:25+08:00","arrivalTime":"2022-08-10T19:34:35+08:00"},"legs":[{"summary":
{"lengthInMeters":6833,"travelTimeInSeconds":431,"trafficDelayInSeconds":0,"trafficLengthInMeters":0,"departureTime":"2022-08-10T19:27:25+08:00","arrivalTime":"2022-08-
10T19:34:35+08:00"},"points":[{"latitude":39.87854,"longitude":116.39424},{"latitude":39.88037,"longitude":116.39392},{"latitude":39.88183,"longitude":116.39367},
{"latitude":39.89227,"longitude":116.39311},{"latitude":39.90031,"longitude":116.39350},{"latitude":39.90234,"longitude":116.39379},{"latitude":39.90592,"longitude":116.39430},
{"latitude":39.90617,"longitude":116.39839},{"latitude":39.90772,"longitude":116.40628},{"latitude":39.90886,"longitude":116.40911},{"latitude":39.91300,"longitude":116.40945},
{"latitude":39.91686,"longitude":116.40919},{"latitude":39.91691,"longitude":116.39600},{"latitude":39.91673,"longitude":116.39078}]}],"sections":
[{"startPointIndex":0,"endPointIndex":13,"sectionType":"TRAVEL_MODE","travelMode":"car"}]}]}
```

Figure 4.2: Data returned by Routing API

or epsilon value, which will be the evaluation metric of the trade-off with privacy and utility. This epsilon value is used to determine two probabilities denoted as p and q such that p and q combine as a whole. We have calculated the p and q values using the symmetric unary encoding (36) method as follows.

$$p = e^{\epsilon}/(e^{\epsilon} + 1)(43)$$

$$q = 1/(e^{\epsilon} + 1)(43)$$

These values are then used to perturb the bits based on the B' array i.e., if the $B_i' = 1$ then perturb the it with the bit present in p bits array else perturb the value with the bit present in q bits array. After perturbing each and every bit in the B' array, we get the final array of bits perturbed with both permanent parameter and instant parameter thus protecting the original value with long term and one time attacks both. Then, the decoded bits are converted back to geohashes of characters and integers. The geohash result is then decoded using the geohash-Hilbert library of Python's decode function to yield latitude and longitude values.

4.1.4 Routing

The Routing API is then used to find the shortest path using the perturbed source and destination Geo coordinates data. The routing API being used is TomTom Developer API (34), and it returns a JSON file as shown in 4.2 containing numerous location coordinates constructing the shortest route when called with an API key, source, and destination location coordinates. The routing API currently returns the route considering the mode of transportation as car only but the travel assistant can be further modified to include complete multi-modal support.

Applying the location coordinates, the route is now displayed in the map and is visible to the user. When the user starts the journey by clicking on Start button, the current location coordinates of the user is provided as known data to the location prediction algorithm i.e., Hidden Markov Model.

4.2 Prediction

The Hidden Markov model implementation includes two steps i. clustering of the data before training it to predict the future location and ii. training the data in HMM model from hmmlearn python library.

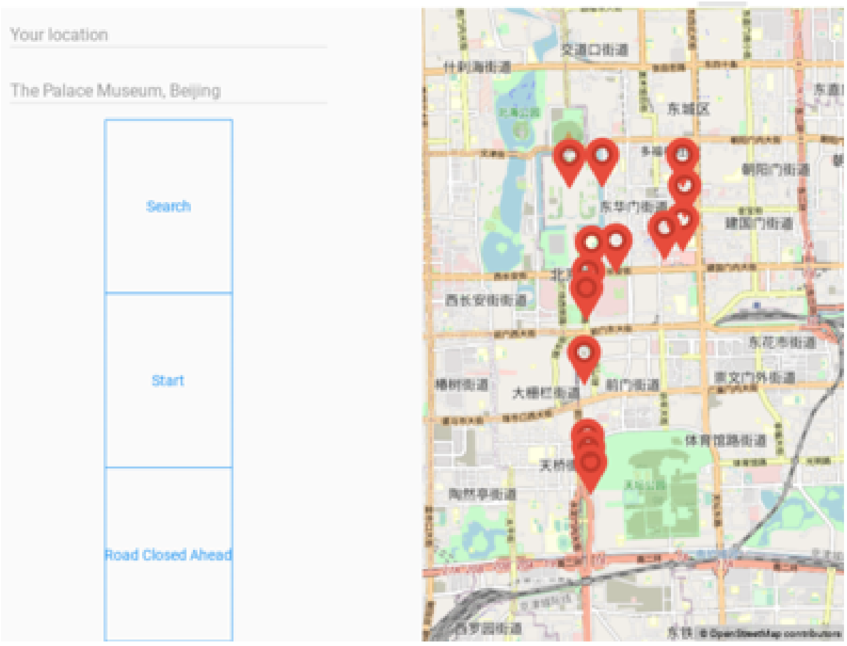
4.2.1 Clustering

When training location data for predicting next location details with HMM, clustering of the geolocation data is necessary since continuous geographic coordinates cannot be used directly for next place prediction. In machine learning, clustering is the process of organizing data into clusters based on shared properties. The use of effective clustering is frequently driven by its predictive abilities. Data are grouped through the process of clustering so that observations within a cluster are more closely related to one another than they are to observations from other clusters. In this project, clusters are significant places generated from the scattered location points all over the space (29) thus reducing the noise from the geolocation dataset.

Amongst various clustering algorithms, Density-Based Spatial Clustering of Applications with Noise also known as DBScan algorithm is the most suitable algorithm for the geo-location data as proved in (29). Unlike Kmeans which is suitable for non-spatial data, DBScan can create any number of clusters based on the data and does not require number of clusters to be decided beforehand which is the most important advantage of DBScan algorithm. DBScan is the most suitable for latitude-longitude pairs because the clustering of spatial data set is based on both minimum cluster size as well as the distance of the each point from each other as compared to k-means which clusters all the points by minimizing the variance amongst the points and not the geodetic distance. The DBScan algorithm generated 153 clusters made from 500k data points from the Geolife dataset.

4.2.2 HMM prediction

Using these clusters HMM model is fitted with the number of components equal to the number of clusters. It is vital to take the data and its nature into consideration before choosing the HMM model from among the various models, such as Gaussian, Multinomial, etc. Since, the latitude-longitude pairs of the data own a continuous nature, Gaussian model is used to predict the next locations (15). To predict next location, the model.predict() function takes the known data as an input.The known data can take the form of geocoordinates for only one location, which is where we are right now in this procedure, or an array of all or some of the previously visited sites. Only the most recent observation from the data available is used to determine the next anticipated location.



Your location

The Palace Museum, Beijing

Figure 4.3: Alert on the screen

The next location predicted by HMM is used to call the Traffic API along with the location coordinates of the destination address. Furthermore, traffic API used is also provided by the TomTom Developer API, returning traffic occurrences between the source and destination GPS locations. The incidents are reported in an array with a specific incident number which classifies the cause of incident for e.g., 1 in the *categoryFilter* string denotes Accident ahead or 8 denotes the road is closed etc. If there is an observed incident then the screen shows an alert to the user as shown in figure 4.3. If there is no incident observed in the Traffic API request and the road works fine then the next location predicted is used to call the Traffic API. This process continues till the user reaches the destination address i.e., current location coordinates of the user is same as of the destination location coordinates. Once the user reaches the destination address, the process is completed and no further actions needed in this transaction.

5 Evaluation

This chapter explains the experiments performed to determine the performance of the previously suggested approach, as well as the metrics used for evaluation of the research statements with respect to the experiments performed. To prove the 3 research questions stated in 2.5, we have performed 3 different experiments.

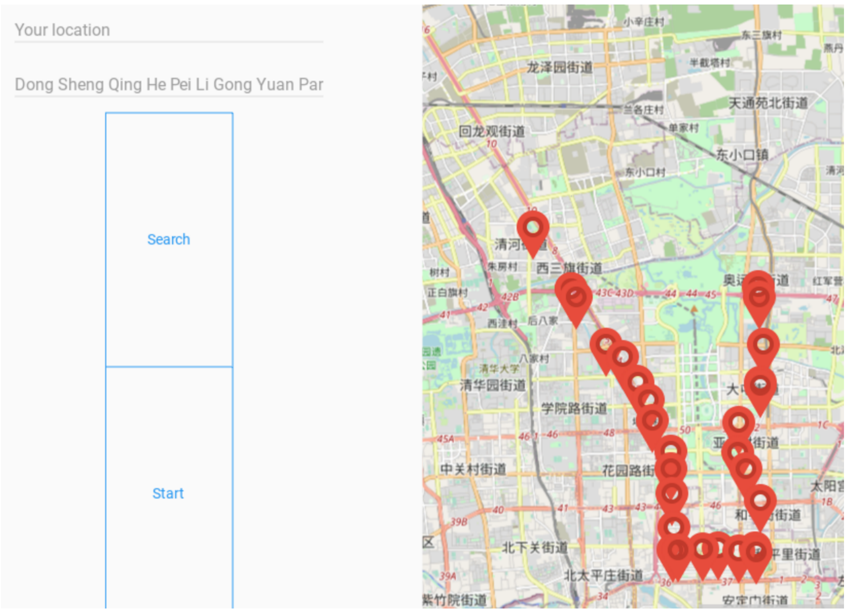
To identify the best privacy budget value, we have examined the utility of the travel assistant with different ϵ values ranging from 0.1 to 10 for 5 different times and using the average values. The metrics used to assess the system's utility and privacy are utility loss and privacy loss. The inaccuracy in the distance computation between the actual or expected values and the perturbed values is how utility loss is determined.

To understand the Markov model performance, we have compared the true trajectory with the trajectory predicted by the markov model.

To confirm if the performance of RAPPOR improves with the reduced correlation, we have compared the true trajectories perturbed with RAPPOR and trajectories predicted by HMM and perturbed with RAPPOR.

5.1 Dataset

To fit and train a hidden markov model for location prediction algorithm, Microsoft's Geolife GPS Trajectory Dataset, a real-world dataset, was used. This dataset was collected in four years period from 2007 to 2011 and includes data on Beijing city with 17,621 trajectory data with latitude, longitude, and altitude information recorded every 5 seconds of interval and was collected over a period of more than 4 years with the full knowledge and consent of 178 users. This dataset recorded a wide range of users' outside movements, such as going to and from work and home as well as certain leisure and sporting activities including dining, shopping, sightseeing, hiking, and cycling. Numerous study areas, including mobility pattern mining, user activity detection, location-based social networks, location privacy, and location suggestion, can benefit from the utilization of this trajectory dataset. The data is recorded covering all major transportation modes such as walk, bike, bus, car, subway, train, airplane,


Your location

Dong Sheng Qing He Pei Li Gong Yuan Par

Figure 5.1: Search results with privacy parameter $\epsilon = 10$

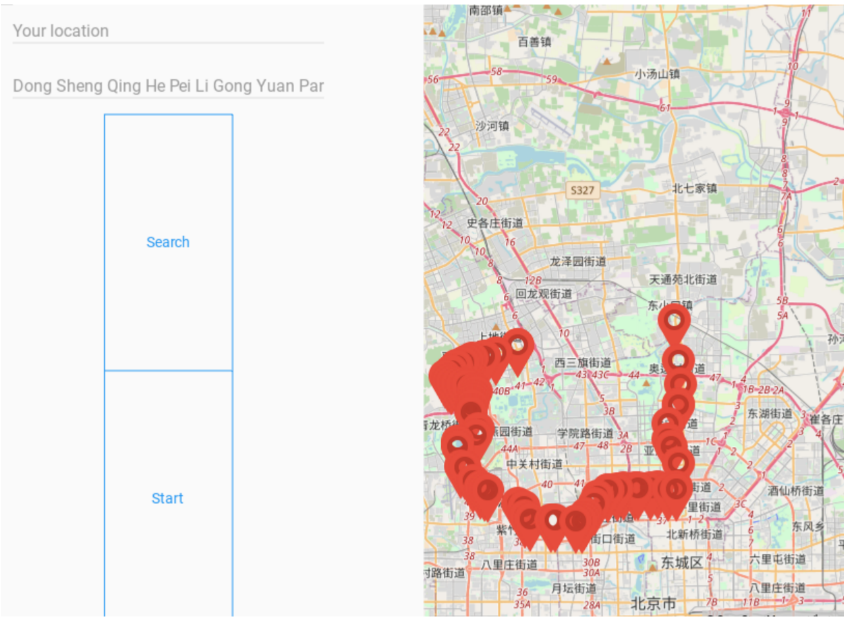
boat, run and motorcycle (47).

5.2 Experiment 1: Trade-off between utility and privacy

Differential privacy allows the users to experiment with the privacy budget value in such a way that its easier to understand the trade-off between utility and privacy of the system. In other words differential privacy allows users to confirm the fact that strengthening privacy of any application will lead to decrease in performance of the application. To determine the optimal privacy budget value is currently a challenge in LDP. As per (43) the privacy budget value ranges from 0.1 to 10 and thus, we have experimented with different privacy budget values such as 0, 0.01, 0.05, 0.1, 0.5, 1, 3, 5, 7, 10. For every ϵ value, the application is tested randomly with the route search from "Beijing International Convention Center, Beijing, China" to "Dong Sheng Qing He Pei Li Gong Yuan Park, Beijing, China" as shown in 5.1 for 5 times and recorded the average values to negate the randomization error which is caused due to the randomized response mechanism used in generating the perturbed values.

This experiment demonstrates how the error rate is decreased with the increase in the privacy budget value and helps to evaluate the importance of error rate in terms of privacy and utility.

Looking at the search results we can easily understand the difference in the performance of the system with the increase in ϵ values. Figure 5.1 shows the route details as expected and



Your location

Dong Sheng Qing He Pei Li Gong Yuan Par

Figure 5.2: Search results with privacy parameter $\epsilon <= 3$

thus we can say that its the best performance of the travel assistant with privacy parameter greater than 5 in contrast to the subpar route shown in figure 5.2 is with low privacy budget where $\epsilon <= 5$. The privacy parameter defines the amount of perturbation in the location data protection. In other words, as the privacy parameter increases the amount of perturbation decreases indirectly reducing the protection and noise in the true data. This in turn increases the performance of the application as shown in figure 5.1.

Distance computation to evaluate utility loss is calculated using Haversine distance formula which is based on the angular distance between two points located on the surface of a sphere.

When the error is higher and the route estimated using perturbed variables diverges significantly from the real or expected path, utility loss is greater. Since maximal secrecy offers a higher privacy protection, privacy loss is computed as the inverse of the error rate in the distance computation of true and perturbed data i.e., inverse of the metric used for utility loss.

As shown in figure 5.3, utility loss is the inaccuracy in the expected routing with true values and actual routing with values perturbed by RAPPOR algorithm. We can conclude that the utility loss is higher when the privacy parameter has low values of 1 and lesser than that and as the ϵ value increases the utility loss reduces thus improving the performance of the application. This is because of the decrease in the perturbation level which leads to the perturbed values as same as true values. Also, when the utility loss is higher such as 140 in figure 5.3, the perturbed values lie so far than the true values that the routing API returns an error as map matching failure which worsens the performance of the travel assistant.

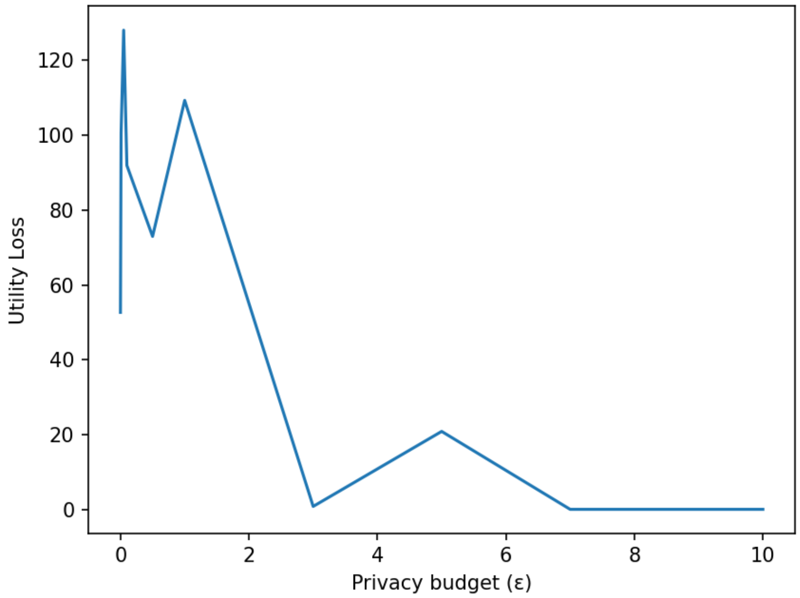


Figure 5.3: Impact of Privacy Budget on Performance for a single search

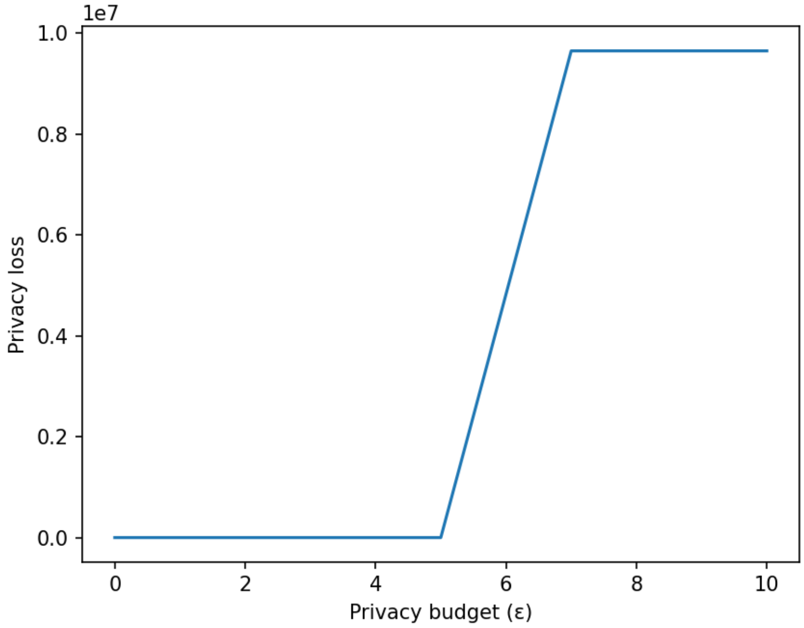


Figure 5.4: Impact of Privacy Budget on Privatization for a single search

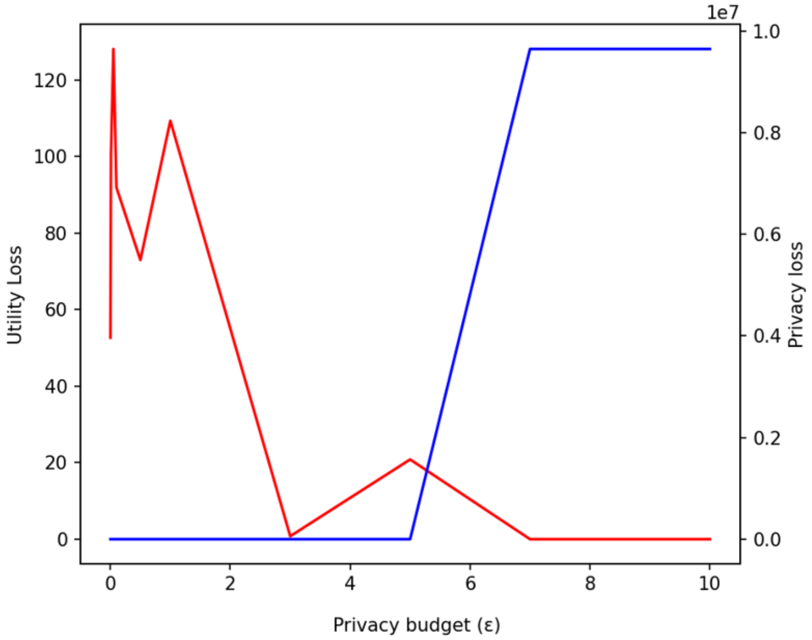


Figure 5.5: Trade-off between utility and privacy for a single search

Fluctuations in the curve between $\epsilon = 0$ and 2 are the results of the randomized perturbations which cause the utility to slightly improve at 0.5 better than at 0 but still is far poorer than at 3.

Figure 5.4 exhibits the privacy loss of the travel assistant system wherein the privacy loss is calculated based on the difference in the true values and the perturbed values. There is no loss of privacy when the ϵ value is less than 5 which denotes that the perturbed values did not reveal the true value at all. Since the perturbation mechanism is directly proportional to e^{ϵ}, when the ϵ is near ∞ giving $e^{\infty} = 0$, thus perturbed value is almost same as true value and privacy loss increases as the ϵ value goes far from 0. There is no fluctuation in the privacy loss curve is due to the wide range of values covered by the curve.

Examining the utility loss and privacy loss for this specific run, we can configure the privacy parameter with values between 3 and 5 for best results of the travel assistant system.

Figure 5.6, 5.7 show the utility loss and privacy loss for 5 different runs separately. The randomization error in the utility loss can be evidently seen due to the random perturbations. Both these plots affirm the ϵ range from 3 to 5 as the best range but when we look at the average results from 5.8, we can see that $\epsilon = 5$ is the overall deviation point for both utility and privacy.

Examining the utility loss and privacy loss for the average of runs, we can configure the privacy parameter with value 5 for best results of the travel assistant system. This trade-off helps to understand that the state which has lowest utility loss and lowest privacy loss is the best optimal state or Pareto optimality (48), achieved when ϵ is at 5. However, if the user

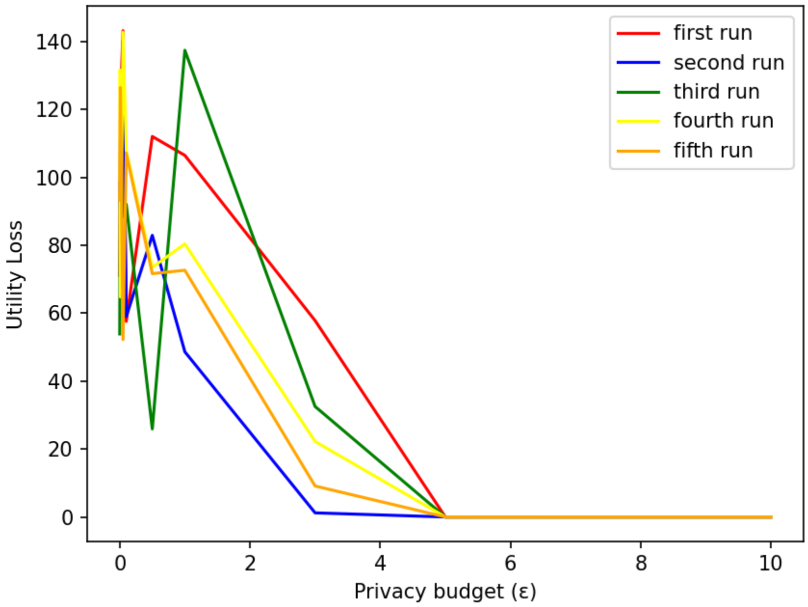


Figure 5.6: Utility loss for 5 different searches

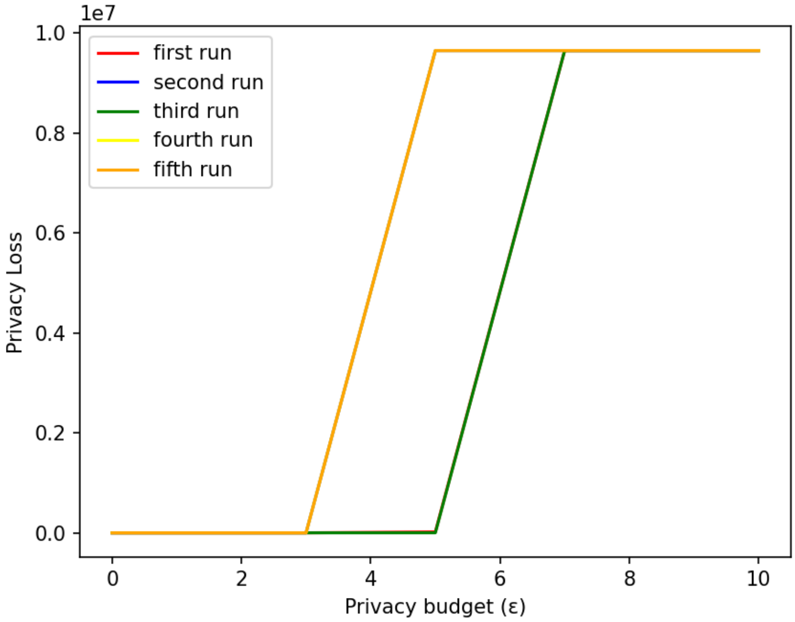


Figure 5.7: Privacy loss for 5 different searches

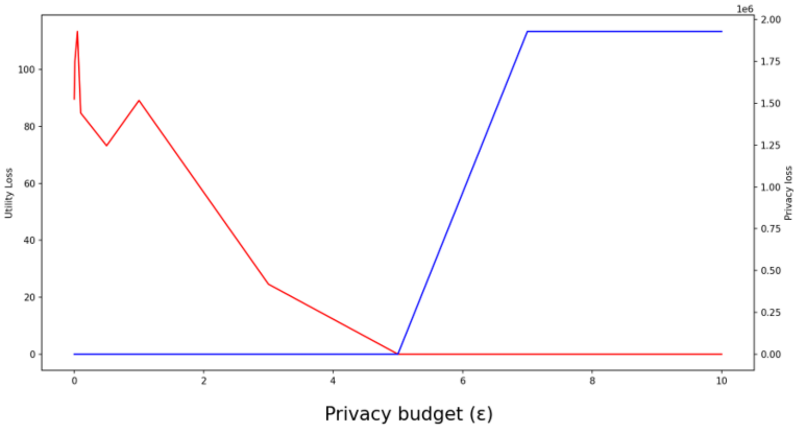


Figure 5.8: Trade-off between utility and privacy with average results

does not want to jeopardize boundary line privacy and prefers to be on the safe side, the privacy budget can be reduced until 3 with a slight trade-off in travel assistant utility.

5.3 Experiment 2: Understand Markov Model Predictions

This experiment is performed using real world dataset explained in section 5.1. To test the hidden markov model, we have plotted the actual locations in the route generated by Routing API and the locations predicted by the HMM model in a scatter plot to examine whether the predicted locations lie in the same region or not. HMM model is trained with only 500k records and thus the predicted locations cover a specific range. Any location selected from outside of Beijing, or specifically which is not present in the training data then the predictions are inaccurate and thus this experiment is performed with the route search from "Beijing International Convention Center, Beijing, China" to "Summer Palace, Beijing, China" which is covered in the training data of the HMM model.

Scatter plot shown in figure 5.9 illustrates that actual locations in blue i.e., the true locations from the user trajectory whereas the ones in red are the locations predicted by the HMM mapped with the longitude in the x-axis and latitude in the y-axis. The spread of red dots indicate the predicted location covers majority area within the true route, allowing for the collection of real time traffic updates throughout the route. Additionally, because the distribution of the projected locations does not follow a course or route, it is difficult for an adversary to create a correlation between the data, decreasing the dependence between the position points. This is made possible by the Markov chain's randomness and memory less property.

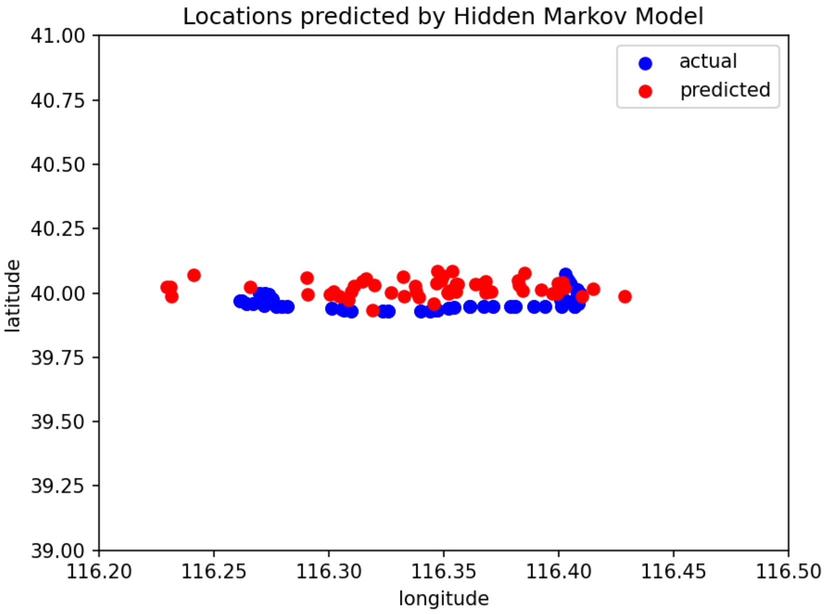


Figure 5.9: True Locations from the user trajectory against the estimated locations by the HMM

5.4 Experiment 3: Comparison of RAPPOR with Markov Model and RAPPOR without Markov Model

This experiment compares the performance of RAPPOR model with and without the correlation of the location data amongst themselves. To demonstrate the impact of dependency, we have configured the privacy parameter to its best value which is 5 and plotted HMM predicted locations and perturbed by RAPPOR as well as true location perturbed by RAPPOR.

Figure 5.10 shows us the trajectories at different stages, blue circles denote the actual route received from Routing API, red circles denote the locations estimated by the HMM model based on the current location and then perturbed by RAPPOR and the green plus signs indicate the actual locations received from Routing API and then perturbed by RAPPOR with an ϵ value of 5. These positions can be mapped with the longitude in x axis and latitude in y-axis.

Comparing the actual values indicated by blue circles and the true locations perturbed by RAPPOR denoted as green plus signs, we can see that even though the RAPPOR values are perturbed, they share the same correlation amongst the location data as the actual trajectory points. When similar data is sent to RAPPOR algorithm with correlations amongst them, there are higher chances of memoization issues which can be minimized significantly if the locations are predicted by using the HMM. HMM does not follow the trajectory and predicts next location based only on current location, thus reducing auto

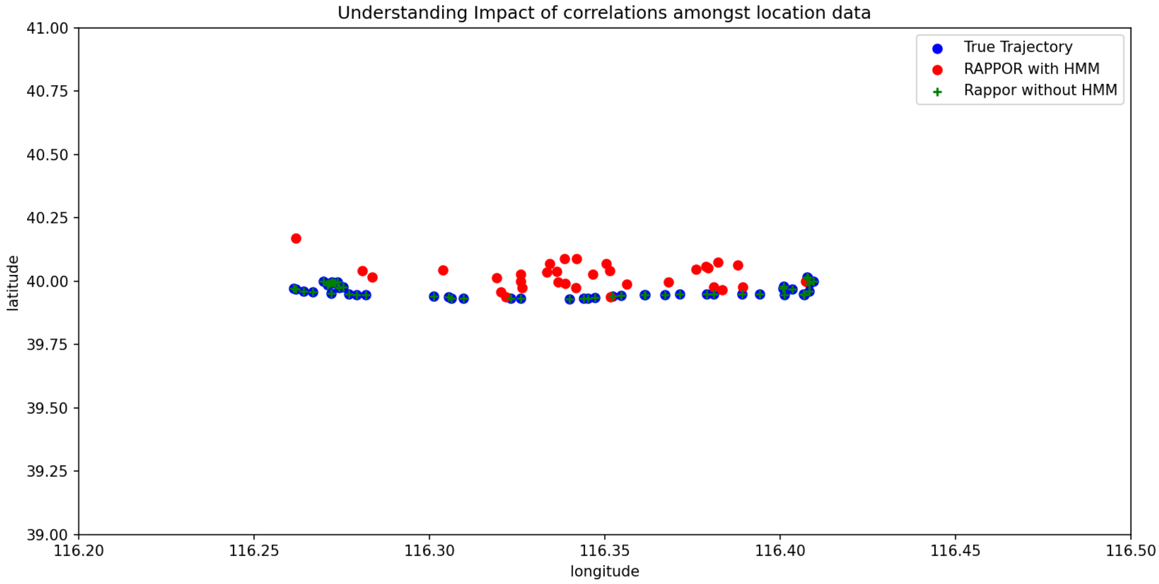


Figure 5.10: Trajectory predicted by HMM and perturbed by RAPPOR vs True trajectory perturbed by RAPPOR

correlation amongst location points.

Results from all the 3 experiments affirm the proposed approach and design. In this travel assistant prototype, best optimal state can be achieved with privacy budget between 3 and 5 which is in the acceptable range as per the authors in (43). Though the location prediction of HMM is bounded with a specific region due to less training data utilized during training the HMM, the predictions are located significantly within the true trajectory without evident correlation amongst them. In the best optimal state, the traditional RAPPOR predictions are almost overlapping the true trajectory positions making it easier for adversaries to correlate the data points amongst each other. Moreover, the suggested approach of RAPPOR with HMM has shown minimal to none correlation amongst the location points. The order of the next location predicted does not follow the location points order from true trajectory thus confusing the adversary and making it more difficult to infer knowledge.

However, due to this mismatch in order of the predictions of the next location when compared with the true locations can degrade the performance of the travel assistant. This utility impact can be explained with an example as follows: Suppose an user is travelling from source A to destination B and starts the journey from A. Suppose next location C predicted is not necessarily the true next location as the order is not same in predicted and actual trajectories and C lies at the end of the journey. Due to this mismatch, an alert can be raised for any traffic incident at the beginning of the journey. This can impact the performance of the travel assistant in both positive and negative ways. Thus, the HMM needs to be trained in the future research to predict the next location in an acceptable range of area.

6 Summary and Conclusion

This chapter summarizes the project from reviewing the state of the art, understanding design and implementation until evaluation of the results. The state of the art chapter defined the features for travel planner and travel assistant clearly. It listed the privacy risks as well as introduced various privacy preserving mechanisms which were used since long back for the protection of static location queries and how the travel assistants were developed to implement differential privacy for protecting continuous location queries. Limitations of differential privacy helped to structure the design of this paper i.e., using location predictions based on Markov process to perturb them with RAPPOR. Our research statement is formed on the basis of this design that evaluates the performance of RAPPOR and HMM in addition to the trade-off between privacy and utility. Design is explained with the help of proper equations and implementations were explained in detail covering both the important algorithms (RAPPOR and HMM) as well as smaller pre requisites such as geohashing and clustering. Experiments performed were evaluated with respect to the research questions stated and the results overall satisfy the suggested approach. The design can be further improvised to include proper evaluation metrics to strengthen the proof of concept.

In this work, we proposed merging the location prediction algorithm with the location protection approach in order to ensure location privacy during real-time updates in a travel assistant. The proposed method employs a hidden markov model to forecast locations during the continuous location sharing essential for a travel assistant, and RAPPOR, the privacy-preserving mechanism, perturbs the predicted locations. The experiment results demonstrate that RAPPOR with HMM minimizes correlation and outlasts regular RAPPOR. The results clearly show that the perturbed trajectory via the standard RAPPOR approach is quite close to the true trajectory and is prone to adversary attacks owing to higher correlations. When the trajectory predicted using HMM and then disturbed by RAPPOR is compared to the trajectory created with RAPPOR, the correlation is less evident in the former and is significantly different from the true trajectory while still covering the whole trajectory range.

Additionally, the design enables users to understand the trade-off between utility and privacy and the impact of the privacy budget. Users can themselves choose the optimal privacy budget value after examining the improvement in the utility with maximized privacy protection. The randomness in the utility curve is justified with the random noise generated from RAPPOR using randomized response mechanism. Results from the multiple runs confirmed the best optimal state to be at $\epsilon = 5$. But if the user wishes to trade the utility of the travel assistant for privacy of the location data, the range of ϵ values from 3 to 5 form the best optimal range.

6.1 Future Work

We identify a few additional research directions in addition to the work to address the aforementioned gaps, as stated in the following.

The evaluation metrics for the trade-off examination needs to be polished in such a way that it speaks to the user in a very relatable manner. For e.g., the utility of the system can be measured with the number of switches in the transport modes. In case of public transports, utility can be measured by comparing if the same mode of transport is suggested.

Due to limitations in processing and memory, HMM in this study is trained only on 500k training data but if we increase the training data, the predictions of HMM might be accurate in the wider region of Beijing. In addition, HMM needs to be trained rigorously to predict the next location in an acceptable range of next locations or a specific area.

Along with location information privacy is a concern when looking at travel assistance applications in various contexts. A lot of user data is used by additional features that offer personalized suggestions, and such data, along with the user's location, has to be protected. Future work on this topic requires to understand the privacy budget to protect such various sorts of data that potentially disclose information.

Bibliography

[1] Differential privacy overview - apple inc..

[2] What's new in python 3.8¶.

[3] Geohashing using hilbert space filling curves, Oct 2019.

[4] Fahed Alkhabbas, Martina De Sanctis, Antonio Bucchiarone, Antonio Cicchetti, Romina Spalazzese, Paul Davidsson, and Ludovico Iovino. Route: A framework for customizable smart mobility planners. In *2022 IEEE 19th International Conference on Software Architecture (ICSA)*, pages 169–179, 2022.

[5] Alvim, Mário and Chatzikokolakis, Konstantinos and Palamidessi, Catuscia and Pazii, Anna. Invited paper: Local differential privacy on metric spaces: Optimizing the trade-off with utility. In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*, pages 262–267, 2018.

[6] Jessie K. Ambrose, Daniel J. Bukovsky, Timothy J. Sedlak, and Scott J. Goeden. Developing a travel route planner accounting for traffic variability. In *2009 Systems and Information Engineering Design Symposium*, pages 264–268, 2009.

[7] Gennady Andrienko, Aris Gkoulalas-Divanis, Marco Gruteser, Christine Kopp, Thomas Liebig, and Klaus Rechert. Report from dagstuhl. *ACM SIGMOBILE Mobile Computing and Communications Review*, 17(2):7–18, 2013.

[8] Atmaca, Ugur and Maple, Carsten and Epiphaniou, Gregory and Dianati, Mehrdad. A privacy-preserving route planning scheme for the Internet of Vehicles. *Ad Hoc Networks*, 123:102680, 09 2021.

[9] David Barber. *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2012.

[10] Sayyada Hajera Begum and Farha Nausheen. A comparative analysis of differential privacy vs other privacy mechanisms for big data. In *2018 2nd International Conference on Inventive Systems and Control (ICISC)*, pages 512–516, 2018.

[11] Zhihan Chen, Bo Wei, and Jingfu Quan. A travel assistant application based on android baidu map. In *2020 International Conference on Intelligent Computing, Automation and Systems (ICICAS)*, pages 299–303, 2020.

[12] Cisco. Personalized Travel Assistant [PTA], Seoul, 2022.

[13] Maria Luisa Damiani. Location privacy models in mobile applications: Conceptual view and research directions. *GeoInformatica*, 18(4):819–842, 2014.

[14] J.F. Dillenburg, O. Wolfson, and P.C. Nelson. The intelligent travel assistant. In *Proceedings. The IEEE 5th International Conference on Intelligent Transportation Systems*, pages 691–696, 2002.

[15] Yongping Du, Chencheng Wang, Yanlei Qiao, Dongyue Zhao, and Wenyang Guo. A geographical location prediction method based on continuous time series markov model. *PLOS ONE*, 13(11), 2018.

[16] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3–4):211–407, aug 2014.

[17] Shaistha Fathima. Global vs local differential privacy, Oct 2020.

[18] Julien Freudiger, Reza Shokri, and Jean-Pierre Hubaux. Evaluating the privacy risk of location-based services. In George Danezis, editor, *Financial Cryptography and Data Security*, pages 31–46, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[19] Giannopoulos, George. The application of information and communication technologies in transport. *European Journal of Operational Research*, 152:302–320, 02 2004.

[20] Google. `https://policies.google.com/privacy`, 2022.

[21] Mingming Guo, Kianoosh G. Boroojeni, Niki Pissinou, Kia Makki, Jerry Miller, and Sitharama Iyengar. Query-aware user privacy protection for lbs over query-feature-based attacks. In *2018 IEEE Symposium on Computers and Communications (ISCC)*, pages 1–7, 2018.

[22] Akshen Kadakia, Devanshi Desai, Urvi Mistry, and Mitchell D'silva. Voyageur. In *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, pages 1–6, 2018.

[23] Muah Kim, Onur Günlü, and Rafael F. Schaefer. Federated learning with local differential privacy: Trade-offs between privacy, utility, and communication. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2650–2654, 2021.

[24] Kivy. Programming guide¶, 2014.

[25] Lee, Ken and Lee, Wang-Chien and Leong, Hong and Zheng, Baihua. Navigational path privacy protection: navigational path privacy protection. pages 691–700, 01 2009.

[26] Hongtao Li, Yue Wang, Feng Guo, Jie Wang, Bo Wang, and Chuankun Wu. Differential privacy location protection method based on the markov model. *Wireless Communications and Mobile Computing*, 2021:1–12, 2021.

[27] Lu Ou, Zheng Qin, Yonghe Liu, Hui Yin, Yupeng Hu, and Hao Chen. Multi-user location correlation protection with differential privacy. In *2016 IEEE 22nd International Conference on Parallel and Distributed Systems (ICPADS)*, pages 422–429, 2016.

[28] Lu Ou, Zheng Qin, Yonghe Liu, Hui Yin, Yupeng Hu, and Hao Chen. Multi-user location correlation protection with differential privacy. In *2016 IEEE 22nd International Conference on Parallel and Distributed Systems (ICPADS)*, pages 422–429, 2016.

[29] Katherine Prinz. Next place prediction with hidden markov models, 2019.

[30] Yuwen Pu, Jin Luo, Ying Wang, Chunqiang Hu, Yan Huo, and Jiong Zhang. Privacy preserving scheme for location based services using cryptographic approach. In *2018 IEEE Symposium on Privacy-Aware Computing (PAC)*, pages 125–126, 2018.

[31] C.G. Raji, Ayman Gafoor, Hijas Ahammed, Aneesh Edavalath, and P.K. Cijas. Wego: An efficient travel assistant application using android. In *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, pages 594–598, 2020.

[32] Karl Rehrl, Stefan Bruntsch, and Hans-Joachim Mentz. Assisting multimodal travelers: Design and prototypical implementation of a personal travel companion. *IEEE Transactions on Intelligent Transportation Systems*, 8(1):31–42, 2007.

[33] Samsel, Christian. *Ubiquitous Intermodal Mobility Assistance*. PhD thesis, 03 2019.

[34] TomTom. Location technology for developers.

[35] Jian Wang, Yanli Wang, Guosheng Zhao, and Zhongnan Zhao. Location protection method for mobile crowd sensing based on local differential privacy preference. *Peer-to-Peer Networking and Applications*, 12(5):1097–1109, 2019.

[36] Tianhao Wang, Jeremiah Blocki, Ninghui Li, and Somesh Jha. Optimizing locally differentially private protocols, 2017.

[37] Ruxue Wen, Rui Zhang, Kai Peng, and Chen Wang. Protecting locations with differential privacy against location-dependent attacks in continuous lbs queries. In *2021 IEEE 20th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 379–386, 2021.

[38] Whim. Whim users privacy policy, 2022.

[39] Wikipedia contributors. Journey planner — Wikipedia, the free encyclopedia, 2022.

[40] Wikipedia contributors. Mobility-as-a-service — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/w/index.php?title=Mobility-as-a-Service&oldid=1072224325`, 2022.

[41] Wang Xiongjian and Yang Weidong. Protection method of continuous location uploading based on local differential privacy. In *2020 International Conference on Networking and Network Applications (NaNA)*, pages 157–161, 2020.

[42] Ming Yan, Shuijing Li, Chien Aun Chan, Yinghua Shen, and Ying Yu. Mobility prediction using a weighted markov model based on mobile user classification. *Sensors*, 21(5), 2021.

[43] Mengmeng Yang, Lingjuan Lyu, Jun Zhao, Tianqing Zhu, and Kwok-Yan Lam. Local differential privacy and its applications: A comprehensive survey. *CoRR*, abs/2008.03686, 2020.

[44] Man Lung Yiu, Christian S. Jensen, Xuegang Huang, and Hua Lu. Spacetwist: Managing the trade-offs among location privacy, query performance, and query accuracy in mobile services. In *2008 IEEE 24th International Conference on Data Engineering*, pages 366–375, 2008.

[45] Li Zhang, Yuwen Qian, Ming Ding, Chuan Ma, Jun Li, and Sina Shaham. Location privacy preservation based on continuous queries for location-based services. In *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 1–6, 2019.

[46] Zhang, Shaobo and Wang, Guojun and Liu, Qin and Wen, Xi and Liao, Junguo. A Trajectory Privacy-Preserving Scheme Based on Dual-K Mechanism for Continuous Location-Based Services. In *2017 IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC)*, pages 1004–1010, 2017.

[47] Yu Zheng, Hao Fu, Xing Xie, Wei-Ying Ma, and Quannan Li. *Geolife GPS trajectory dataset - User Guide*, July 2011.

[48] Ziefle, Martina and Halbey, julian. Users' Willingness to Share Data on the Internet: Perceived Benefits and Caveats. 04 2016.

[49] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 21st ACM Conference on Computer and Communications Security*, Scottsdale, Arizona, 2014.