Protection of Sensitive Trajectory Datasets Through Spatial and Temporal Exchange

Elham Naghizade, Lars Kulik, Egemen Tanin
Computing and Information Systems Department
The University of Melbourne
Melbourne, Australia
{enaghi,lkulik,etanin}@unimelb.edu.au

ABSTRACT

Privacy concerns place a great impediment to publishing and/or exchanging trajectory data across companies and institutions. This has urged researchers to address privacy issues prior to trajectory data release. Currently, privacy preserving solutions distort original data unnecessarily, hence, degrade data utility and make such data less useful for third parties. We consider a trajectory as a sequence of *stops* and *moves*, and propose an approach that exploits features of a trajectory as means for preserving privacy while maintaining a high level of utility. We introduce the concept of sensitivity for stops based on the assumption that they are more vulnerable to privacy threats. We propose an efficient algorithm that either substitutes sensitive stop points of a trajectory with moves from the same trajectory or introduces a minimal detour if a less sensitive stop can not be found on the same route. Our experiments shows that our method balances user privacy and data utility: it protects privacy through preventing an adversary from making inferences about sensitive stops while maintaining a high level of data similarity to the original dataset.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Spatial databases and GIS*

Keywords

Spatiotemporal Database, Location-based Services, Trajectory Privacy

General Terms

Algorithms

1. INTRODUCTION

Fine-grained human mobility traces are captured at an unprecedented level from localisation sensors and GPS-enabled devices. Various applications create an abundant collection of continuous timestamped location data, i.e., trajectory data. However, the precise nature of this data makes individuals subject to various privacy attacks [4]. Consequently, location privacy and the means of preserving it have drawn attention of many researchers [2, 4, 5, 7]. Most studies in the literature aim to preserve the footprint of a trajectory, and emphasise on the importance of protecting its start and end point. Such approaches focus on the spatial aspect of a trip but not on its semantics. Except for the usual trip from home to work and vice versa, the start or end points of a trip are usually not discriminative enough to identify the purpose of a trip. Meanwhile, a trip's semantic can be learned more comprehensively from the stops of a trip that are not the start or end point, i.e., the *intermediate* stops. For instance, a visit to a medical centre followed by a stop at a pharmacy may indicate that it is a *health-related* trip.

We consider stops more vulnerable to privacy threats as they not only expose the purpose of a trip but their semantic plays a vital role in assessing how private a trip is. Protecting stops rather than the whole trajectory can be beneficial in two ways. First, it may alleviate the amount of introduced distortion compared to the original trajectory data. In addition, this method of protecting trajectory data is highly effective when an adversary aims to make further private inferences instead of identifying or tracking an individual.

Recently, some studies [3, 6, 8, 9] aim to protect specific parts of a trajectory that are considered to be more sensitive rather than protecting trajectory as a whole. For instance, Huo et al. [6] assume that most of an adversary's background knowledge is associated with an individual's visited places and hence, preserving such places protects the trajectory privacy. To hide the whereabouts of an individual, they coarsened the location of visited places in the database. However, repeated transitions between coarse and fine granularities in dataset may help an adversary to infer additional information from an individual's path, and hence undermine trajectory privacy. As a result, our approach aims to preserve the uniformity of the dataset in terms of granularity. In addition, in our work stop points are not considered equally private and their level of sensitivity is determined according to their type, e.g. hospital, station, restaurant, as well as other parameters.

A key idea of this paper is to utilize the sequences of stops and moves in a trajectory to protect sensitive stops while ensuring that data granularity remains uniform. This is achieved by first making a sensitive stop a part of a non-sensitive move episode and then substituting it with a less private stop (Figure 1). Given a certain POI (point of interest) density, our *Flip-flop* approach efficiently preserves trajectory privacy through exchanging sensitive stops with less sensitive, and possibly varied, types of POIs. This approach also manages to maintain utility since it selects the POIs that introduce the least distortion to the data.

SSDBM '14 June 30- July 02 2014, Aalborg, Denmark
ACM Copyright 2014 ACM 978-1-4503-2722-0/14/06...$15.00.
http://dx.doi.org/10.1145/2618243.2618278

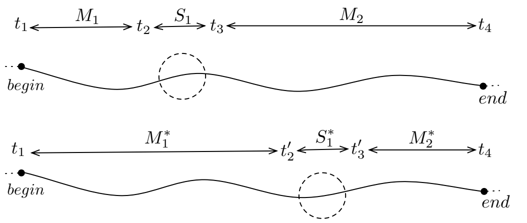

Figure 1: Protecting stops through exchanging trajectory episodes.

2. RELATED WORK

Some studies have focused on *stop points* along a path as the most sensitive parts of a trajectory. The authors in [3] propose an obfuscation technique for online location-based services. This technique considers the geographic context of a stop and provides an uncertainty region based on the POI distribution and users' privacy profile. The work in [9] studies the issue of privacy-aware trajectory publishing by focusing on sensitive stays along a trip. Its goal is to publish a *c-safe* version of the original dataset where the probability of inferring that a person has visited a sensitive stop along visiting a sequence of non-sensitive stops is below a safety threshold. Likewise, in [6] the authors employed generalisation methods so that sensitive places along a user's trip will be replaced by l-diverse zones. In other words, the published dataset contains a set of fine-grained location points along with some cloaked areas, including at list *l* distinct place types, that represent user's stops. Our work shares the same assumption of these studies that it regards stop points of the trajectory as more vulnerable to privacy breaches.

However, publishing a dataset with different levels of location granularities, where an individual spent a significant time in coarser areas, may cause privacy concerns itself. The adversary can easily infer the number of sensitive stops in a trip along with the time they occurred, which is in fact, a privacy breach itself. The adversary may also find a correlation between place types in consecutive stops along a trip, in order to make further inferences. For instance, if a user has stopped in a zone including a hospital, and later has stopped in another zone containing a pharmacy, the adversary may relate these places to make further inferences. It is also possible to refine the obfuscated rectangle if the user takes different paths to get to the same stop point. Moreover, these studies did not take the temporal property of a trajectory into account and decided on the sensitiveness of a place only by considering its position in space. However, the duration of staying in an intermediate stop may play an important role in its sensitivity. Our approach addresses the issues highlighted above.

3. PROBLEM SET-UP

In our work, an adversary is any third party with whom the dataset is shared. The adversary is interested in inferring personal preferences or habits of individuals through the identification of particular visited places and/or the frequency of these visits. We assume the adversary knows that a user's trajectory data may be distorted.

3.1 Measuring Privacy

The ultimate goal of this work is to exchange an individual's sensitive stops with less sensitive POIs in a consistent manner. In our work, every stop is tagged with a place sensitivity rank, r_p, where $0 < r_p \leq 1$. A higher r_p corresponds to a place that a user considers more sensitive. r_p can be determined based on users' privacy setting and/or the underlying application.

Other than the place of a stop, temporal properties of a stop are also of great importance when determining its level of sensitivity. r_p is the minimum sensitivity rank specified by a user and can increase with the duration of a stop. The longer the user stays at a stop, the greater is the sensitivity of a place. To the best of our knowledge, this is the first work that takes the temporal features of a stop into account when trying to protect its privacy. Using the place sensitivity rank, r_p, the duration of a stop, d_s, and the total duration of a trip, d_t, we compute the overall sensitivity of a stop as:

$$r_s = r_p^{\frac{d_t - d_s}{d_t}}$$

where r_s can take any value between r_p and 1. $r_s = 1$ if the entire trip occurs at a stop point.Following the exchange of the sensitive stops with less sensitive POIs, we measure the accuracy of our approach in terms of preserving privacy as its ability to minimise the sensitivity level of a trip:

$$Privacy\ Gain = \frac{\sum_{i=1}^{k}(r_{s_i} - r^*_{s_i})}{max_{sd}} \quad \in [0,1]$$

where k is the number of stops in a trajectory, and max_{sd} is the maximum sensitivity deviation that occurs when choosing the k least sensitive POIs on the route. r_{s_i} and $r^*_{s_i}$ are the sensitivity of the original stop and the substitute POI respectively.

3.2 Measuring Utility

We determine data utility by comparing the original trajectory with its exchanged match and measuring how similar they are. We are not only interested in measuring deviations from the original footprint (spatial projection) of trajectories, but we also need to compute temporal displacements caused by our exchange process. As a result, we adapt a Frećhet-based distance [1], and compute the distance as if a single pair of speed parameterization is available instead of looking for the optimal parameterization that minimises the distance between two trajectories independent of time. Hence, we compute the distance as the maximum spatial distance between every temporally coincident pair of points:

$$Distortion = \max_{i \in [1,n]} [d(\vec{l_i} - \vec{l}^*_i)]$$

where n is the number of points in each trajectory, $\vec{l_i}$ and $\vec{l}^*_i$ are the location points of the original trajectory and exchanged trajectory at time i, and d is the Euclidean distance between two points. Estimating data utility $u \in [0,1]$ requires the definition of perfect utility and worst-case utility concepts. In our work, we assume perfect utility occurs if the original trajectory remains unchanged ($u = 1$), but worst-case utility can be described as a case that given a certain source, destination, and time budget, deviating from the original trajectory as much as possible, hence maximising information loss, i.e., $u = 0$. We then compare the exchanged trajectory, $\mathscr{T}^*$, against the original trajectory, $\mathscr{T}$, to determine utility:

$$Utility = 1 - \frac{Distortion(\mathscr{T}, \mathscr{T}^*)}{Distortion_{max}} \quad \in [0,1]$$

4. FLIP-FLOP APPROACH

Following a pre-processing phase that mainly deals with finding stop points and determining their level of sensitivity, this work aims to protect retrieved sensitive stops through the exchange strategy depicted in Figure 1. Our exchange strategy aims to preserve the overall characteristics of the trajectory, namely duration, regional proximity, and average speed.

4.1 Stop and Move Exchange

As mentioned earlier, the algorithm utilizes POIs when exchanging the sequences of stops and moves of a trajectory. Depending on the result of searching for POIs, two scenarios may occur, namely replacement and displacement. The algorithm may find a less sensitive POI on the same route and *replace* the more sensitive stop point with it (Figure 2, right). Otherwise, the algorithm needs to search for a close POI that does not belong to the present route and to *displace* the sensitive stop with that POI (Figure 2, left). In order to limit the POI search space, we used the properties of an ellipse. Setting any two stop points as the foci of an ellipse, the algorithm is able to limit the search to those POIs inside the ellipse area whose distance from the stops are less than a threshold; this threshold can be increased until a POI is found, however this increase should not exceed the overall boundary of an ellipse that contains all the reachable points from source to destination within the available time budget. In this case the sensitive stop(s) are regarded as non-preservable and the exchange approach fails to protect them. However, our experimental results show that such case is highly unlikely in real world scenarios.

The replacement process completely preserves the trajectory footprint, and only involves temporal modification. However, displacement may cause local changes to both temporal and spatial properties. Nonetheless, displacement is performed in a way to keep these changes as minimally invasive as possible.

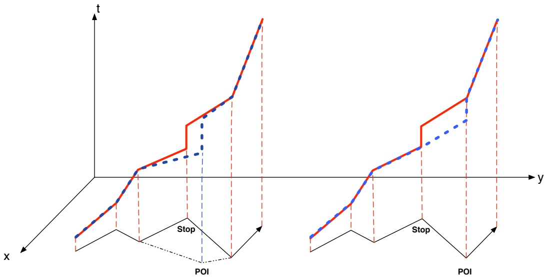

Figure 2: Preserving a sensitive stop through replacement and displacement.

In our work, we also adopt two methods of searching for POIs. Our benchmark method considers a trajectory as a whole and searches for less sensitive POIs exhaustively. Our Flip-flop method, on the other hand, compartments the trajectory into sections, searches for POIs in each section and does the Flip-flop exchange.

4.1.1 Flip-flop Exchange

The flip-flop approach segments the trajectory into episodes of stops and moves (Algorithm 1 lines 3-6). It then considers the move episode between every two consecutive stops and looks for POIs (Algorithm 1, line 7). If it manages to retrieve a less sensitive POI on this route, it replaces the first stop with it (Algorithm 1, lines 8-11). Otherwise, it searches in an ellipse area, whose foci are the two consecutive stop episodes, in order to find a less sensitive POI within this area. Finally, the algorithm displaces the first stop with the retrieved POI (Algorithm 1, lines 12-14).

Algorithm 1: Flip-flop Exchange

```
Input  : Original trajectory dataset, $\mathcal{T}$, list of sensitive stop
         points, $\mathcal{S}_s$, set of all POIs, $P$.
Output: An exchanged trajectory, $\mathcal{T}^*$, with respect to
         sensitive stop points.
1.1  $i \leftarrow 1$;
1.2  $\mathcal{T}^* \leftarrow []$;
1.3  while $i <= length(\mathcal{S}_s)$ do
1.4      $startIndex \leftarrow \mathcal{S}_s[i][ID]$;
1.5      $endIndex \leftarrow \mathcal{S}_s[i+1][ID]$;
1.6      $\mathcal{T}_{loc} \leftarrow \mathcal{T}[startIndex : endIndex]$;
1.7      $\mathcal{P} \leftarrow filter(routePOI(\mathcal{T}_{loc}, P))$;
1.8      if $length(\mathcal{P}) > 0$ then
1.9          $poi \leftarrow leastSensitive(\mathcal{P})$ ;
1.10         $\mathcal{T}^* \leftarrow \mathcal{T}^* + replaceStop(\mathcal{T}^*, \mathcal{S}_s[i], poi)$;
1.11
1.12     else
1.13         $foci_1, foci_2 \leftarrow \mathcal{S}_s[i], \mathcal{S}_s[i+1]$;
1.14         $T_{dis} \leftarrow disStop(\mathcal{T}_{loc}, foci_1, foci_2, P, Axis_{maj})$
             $\mathcal{T}^* \leftarrow \mathcal{T}^* + T_{dis}$;
1.15     $i \leftarrow i+1$;
```

5. EXPERIMENTS

5.1 Experimental Setup

We developed a network-based trajectory simulator to generate GPS trajectories with various modes of transport and multiple intermediate stops. The simulator first chooses two random points within an area ($\approx 12km \times 12km$) in the city of Melbourne and computes the shortest path between them. It then finds all the POIs on this path using OpenStreetMap[1] data and randomly selects k POIs as the intermediate stop points, where $1 \leq k \leq 5$. Generally, the generated trajectories may have a single mode of transport, i.e., using a car, or sometimes up to four transitions between walking and driving. In summary, the generated dataset consists of a total of 750 distinct trajectories with an average length of 23 km. Our underlying network consists of 59680 nodes and 69534 edges.

In order to examine the effect of POI density on Flip-flop's performance, we generate points uniform at random in a given area of Melbourne. Our observations show that most POIs in Melbourne are on streets that are classified as territory and secondary roads. Thus, we mapped the POIs to those roads, rather than distributing them randomly. For varying p_s (average number of POIs per edge), we have approximately 2170 – 65,000 POIs in the network (note that some of the randomly generated points cannot be mapped to any edge). Considering the overall length of the network (2622 km), this POI density creates different urban scenarios, i.e., sparse areas versus more populated areas. The default environment for our experiments is set to reflect the worst cases with a low POI density and high number of long stops. Moreover, we repeat each set of the experiments 20 times and Section 5.2 provides the average of these results.

Flip-flop can be tailored to optimize trajectory privacy or data utility: a privacy-aware version (PFF) searches for the least sensitive POI as the substitute and a utility-aware version (UFF) that selects a POI that minimizes the distance to the original trajectory when exchanging each POI. The PFF and UFF's privacy are evaluated relative to the exhaustive approach (set as 1), which finds the least sensitive POIs as substitutes (Section 3.1). In our experiments we use the exhaustive approach as a baseline to compare the success of UFF and PFF in preserving privacy, and the result of the exhaustive approach is not further discussed due to space constraints. The util-

[1] www.openstreetmap.org

ity is measured with respect to the relation provided in Section 3.2 where we employed the generated trajectories with worst utility to estimate the maximum distortion.

5.2 Experimental Results

Figure 3 shows the average privacy and utility in the dataset for varying POI densities. As expected, low POI densities lead to lower data utility for both variants of Flip-flop because more detours are required from the original trajectory's footprint as less non-sensitive POIs are available along the original trajectory. Figure 3 demonstrates that with an increase in POI density, UFF becomes more successful in maintaining data utility (Figure 3, left). However, the increase in POI density leads to lower privacy levels in UFF since UFF can find POIs at closer distance but not necessarily POIs with the least sensitivity.

Generally, PFF distorts the original data more than UFF but achieves significantly higher privacy levels (Figure 3, right). More specifically, with an increase in POI density, PFF achieves the best possible privacy, i.e., similar privacy levels as the exhaustive approach, although this accuracy comes – as expected – at the cost of a slight loss in utility. As for PFF, an increase in POI density increases the utility level as well. This is mainly due to the ability of PFF to find less sensitive POIs at closer distances to the original sensitive stop.

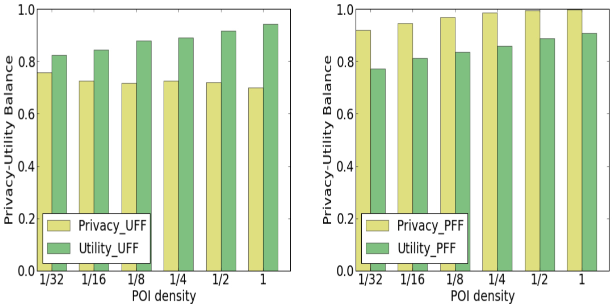

Figure 3: Effect of POI density on the average privacy-utility.

Figure 4 shows that Flip-flop can cope with any POI density and has an almost constant computation time (bottom line). Note that the x axis shows the average number of POIs per kilometer. With lower densities, this time is mostly spent on the displacement process, whereas with an increase in POI density, the computation time is largely spent on the search for POIs along the original route. For low POI densities the exhaustive algorithm could be potentially faster than Flip-flop since a global search increases the probability of finding a POI on the same route and requires less frequent displacements. However, such POI densities (less than $1 POI/km$ of road), are not typical in real world scenarios.

6. CONCLUSION

We propose an algorithm that manages to preserve trajectory privacy with regard to its semantics. To achieve this, we utilize a trajectory's stop and move episodes in order to safeguard sensitive stop points. Our Flip-flop approach not only exchanges sensitive stops with less sensitive POIs more efficiently, but it also results in a high data utility. In the future, we aim to further investigate the effect of POI density on the overall trajectory privacy. We expect that having the POI density of a trip in advance may provide individuals with an estimation of how private their trip can be. In other words, if the POI density of a certain trip is known, the reachable privacy level can be predicted before the trip is disclosed. Similarly, given an area with certain POI densities, a service provider may determine if the required data utility can be achieved with regard to individuals' privacy preferences.

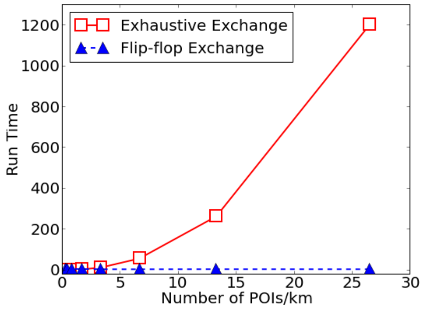

Figure 4: Effect of POI density on Flip-flop's performance.

7. ACKNOWLEDGMENT

This research was partially supported under Australian Research Council's Discovery Projects funding scheme (project number DP110100757) and partially supported by the NICTA Victoria Research Laboratory.

8. REFERENCES

[1] H. Alt and M. Godau. Computing the frechet distance between two polygonal curves. *International Journal of Computational Geometry and Applications*, 5:75–91, 1995.

[2] A. Beresford and F. Stajano. Location privacy in pervasive computing. *IEEE Pervasive Computing*, 2(1):46–55, 2003.

[3] M. L. Damiani, E. Bertino, and C. Silvestri. Protecting location privacy against spatial inferences: the probe approach. In *Proceedings of the ACM SIGSPATIAL 2009 International Workshop on Security and Privacy in GIS and LBS*, pages 32 – 41, 2009.

[4] M. Duckham and L. Kulik. A formal model of obfuscation and negotiation for location privacy. In H.-W. Gellersen, R. Want, and A. Schmidt, editors, *Pervasive Computing*, volume 3468 of *Lecture Notes in Computer Science*, pages 152–170. Springer Berlin Heidelberg, 2005.

[5] M. Gruteser and D. Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *MobiSys*, pages 31 – 42, 2003.

[6] Z. Huo, X. Meng, H. Hu, and Y. Huang. You can walk alone: Trajectory privacy-preserving through significant stays protection. In S.-g. Lee, Z. Peng, X. Zhou, Y.-S. Moon, R. Unland, and J. Yoo, editors, *Database Systems for Advanced Applications*, volume 7238, pages 351 – 366. Springer Berlin Heidelberg, 2012.

[7] J. Krumm. A survey of computational location privacy. *Personal and Ubiquitous Computing*, 13(6):391–399, 2009.

[8] B. Lee, J. Oh, H. Yu, and J. Kim. Protecting location privacy using location semantics. In *SIGKDD*, KDD, pages 1289 – 1297, 2011.

[9] A. Monreale, R. Trasarti, C. Renso, D. Pedreschi, and V. Bogorny. Preserving privacy in semantic-rich trajectories of human mobility. In *Proceedings of the ACM SIGSPATIAL International Workshop on Security and Privacy in GIS and LBS*, pages 47–54, 2010.