Reconstruction Attack on Differential Private Trajectory Protection Mechanisms

Erik Buchholz
University of New South Wales
Data61, Cyber Security CRC
Sydney, NSW, Australia
e.buchholz@unsw.edu.au

Alsharif Abuadbba
CSIRO's Data61
Cyber Security CRC
Sydney, NSW, Australia
sharif.abuadbba@data61.csiro.au

Shuo Wang
CSIRO's Data61
Cyber Security CRC
Sydney, NSW, Australia
shuo.wang@data61.csiro.au

Surya Nepal
CSIRO's Data61
Cyber Security CRC
Sydney, NSW, Australia
surya.nepal@data61.csiro.au

Salil S. Kanhere
University of New South Wales
Cyber Security CRC
Sydney, NSW, Australia
salil.kanhere@unsw.edu.au

ABSTRACT

Location trajectories collected by smartphones and other devices represent a valuable data source for applications such as location-based services. Likewise, trajectories have the potential to reveal sensitive information about individuals, e.g., religious beliefs or sexual orientations. Accordingly, trajectory datasets require appropriate sanitization. Due to their strong theoretical privacy guarantees, differential private publication mechanisms receive much attention. However, the large amount of noise required to achieve differential privacy yields structural differences, e.g., ship trajectories passing over land. We propose a deep learning-based *Reconstruction Attack on Protected Trajectories (RAoPT)*, that leverages the mentioned differences to partly reconstruct the original trajectory from a differential private release. The evaluation shows that our RAoPT model can reduce the Euclidean and Hausdorff distances between the released and original trajectories by over 68 % on two real-world datasets under protection with $\varepsilon \leq 1$. In this setting, the attack increases the average Jaccard index of the trajectories' convex hulls, representing a user's activity space, by over 180 %. Trained on the GeoLife dataset, the model still reduces the Euclidean and Hausdorff distances by over 60 % for T-Drive trajectories protected with a state-of-the-art mechanism ($\varepsilon = 0.1$). This work highlights shortcomings of current trajectory publication mechanisms, and thus motivates further research on privacy-preserving publication schemes.

CCS CONCEPTS

• **Security and privacy** → **Data anonymization and sanitization**; *Privacy protections*; • **Computing methodologies** → Neural networks;

KEYWORDS

Trajectory Privacy, Differential Privacy, Location Privacy, Deep Learning

ACSAC '22, December 5–9, 2022, Austin, TX, USA

ACM Reference Format:
Erik Buchholz, Alsharif Abuadbba, Shuo Wang, Surya Nepal, and Salil S. Kanhere. 2022. Reconstruction Attack on Differential Private Trajectory Protection Mechanisms. In *Annual Computer Security Applications Conference (ACSAC '22), December 5–9, 2022, Austin, TX, USA.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3564625.3564628

1 INTRODUCTION

Due to the omnipresence of smartphones and wearables in our daily lives, a large amount of personal location data is collected every day. The sequence of locations visited by an individual represents a trajectory. This data is valuable for many services such as research [4], market analysis [58], navigation [19, 56], social gaming [39], and most recently for contact tracing in the context of the COVID-19 pandemic [26, 43, 45]. However, significant privacy concerns are associated with the release of location information. The location trajectory of an individual may reveal sensitive information, such as religious, political, or sexual beliefs [2, 47]. For instance, someone with access to the trajectories of a taxi fleet could determine which drivers are practising Muslims based on the correlation of their breaks and mandatory prayer times [17]. Moreover, De Montjoye et al. showed that only four spatio-temporal points suffice to uniquely identify 95 % of individuals [10]. These examples illustrate the need of trajectories for appropriate protection before being released.

To hide the exact route of individual trajectories with the goal to prevent pirate attacks [23], stalking [18] or other security threats, multiple approaches that extend k-anonymity [2, 3, 18, 38, 51, 57] and differential privacy [5, 6, 22, 23, 28, 30] to the trajectory domain have been proposed [47]. However, existing approaches either significantly reduce the utility of the released data or provide limited privacy [46, 47]. K-anonymity based approaches are susceptible to attacks utilising background knowledge, and cannot provide strong privacy guarantees [6, 7, 18, 24, 33]. Therefore, recent research focused on publication mechanisms achieving differential privacy. Due to the high information content of location data, these approaches significantly degrade data utility to achieve privacy protection [33, 48, 49] because there is an inherent trade-off between privacy and utility. The random distortion added by differential private protection mechanisms yields unrealistic trajectories that provide limited utility and can easily be recognised [48] because

they do not take geographical constraints into consideration [37]. For instance, protected trajectories of cars do not follow roads or ship trajectories pass over land. To highlight the risk posed by current publication mechanisms, we address the research question:

RQ1. Can an adversary (partly) reconstruct trajectories from a differential private trajectory release?

We find that the differential private mechanisms, such as the Sampling Distance and Direction (SDD) approach [23], yield trajectories that are structurally distinguishable from unperturbed trajectories. By exploiting these characteristics, we propose a novel Long Short-Term Memory (LSTM)-based Reconstruction Attack on Protected Trajectories (RAoPT) to address the defined research question **RQ1**. The RAoPT model receives trajectories protected with a differential private publication mechanism as input and outputs reconstructed trajectories which are closer to the original trajectories.

Our attack is evaluated on two real-world GPS datasets, the T-Drive [59], and the GeoLife [60] dataset. We evaluate the reduction of the Euclidean and Hausdorff distances between the recovered and original trajectories compared to the distances between the protected and original trajectories. These metrics are commonly used to measure the distance between trajectories [22, 23, 28, 33, 49, 52, 52]. We also measure the increase of the Jaccard index of the trajectories' convex hulls before and after reconstruction, as the convex hull can represent a trajectory's activity space, i.e., the area in which a user is active [27]. A small physical distance to a victim represents a security threat in various settings, e.g., stalkers can follow or intercept a victim [18], or pirates can plan attacks [23].

For an adversary with knowledge about the used protection method and access to ground truth trajectories for the training, our RAoPT model can reduce both distances even for privacy settings with very high privacy guarantees ($\varepsilon \leq 0.1$) by over 98 % in case of Laplace noise-based protection and by 68 % to 84 % considering a state-of-the-art protection mechanism. In these settings, the Jaccard index is increased by at least 180 %. In the realistic scenario that the adversary knows about the protection method (SDD with $\varepsilon = 0.1$) but no ground truth for training is accessible, the Euclidean distance can still be reduced by approx. 40 to 61 %, and the Hausdorff distance by approx. 62 % to 67 % when transferring from one dataset to the other. Even an adversary without background knowledge can achieve reductions of of over 60 % in some settings.

Contributions. Our main contributions are as follows:

- We propose the first LSTM-based reconstruction attack on differential private protected trajectories.
- The RAoPT model can reduce the Euclidean and Hausdorff distances by over 68 % on T-Drive trajectories protected with the Laplace mechanism or a state-of-the-art protection mechanism and $\varepsilon \leq 1$, decreasing the provided level of privacy.
- We show the real-world applicability of the attack via two datasets, namely T-Drive [59] and GeoLife [60], with different granularities and transport modes.
- We open-source our RAoPT model[1] including the considered protection mechanisms and pre-processing scripts.

[1]Our source code is available at: https://github.com/erik-buchholz/RAoPT

This article is organised as follows. We introduce the required background knowledge in Section 2 and define our problem statement and threat model in Sections 2.3 and 2.4, respectively. In Section 3, we discuss related work addressing attacks on protection mechanisms, and approaches utilising deep learning for trajectory protection. Then, we introduce our proposed RAoPT model in Section 4. In Section 5, we provide implementation details and show the results of our evaluations. Subsequently, we discuss our findings in Section 6. Finally, we conclude this paper in Section 7.

2 PRELIMINARIES

In Section 2.1, we first define the term *location trajectory* and provide some general knowledge on trajectory protection mechanisms. The SDD mechanism, which is used for the evaluation of our attack, is explained in Section 2.2. In Section 2.3, we state our problem statement, and in Section 2.4, we present our threat model. We refer readers not familiar with differential privacy to Appendix B.

2.1 Location Trajectory Protection

A location trajectory T consists of a sequence of locations $T = (t_1, \ldots, t_n)$. In the most basic case, each location consists of two values $t_i = (x_i, y_i)$ which can either represent the location within a reference (coordinate) system or *latitude* and *longitude*. For exact localisation, the *altitude* or *elevation* can be added to latitude and longitude as a third coordinate. Many trajectory datasets record a *timestamp* for each location [59, 60]. Lastly, trajectories can be enhanced by semantic information [55], such as Point of Interests (POIs), i.e., the knowledge of whether a location within a trajectory represents a restaurant, shop, or gym. While such additional information can increase the utility of a dataset for analyses, semantic information can be exploited for attacks such as Trajectory User Linking (TUL) [34, 55]. Semantic information can also be added to a dataset retrospectively, e.g., by matching the location points of a trajectory against semantic datasets, such as OpenStreetMap [9].

Protection mechanisms for the release of trajectory datasets target two different scenarios. In the first scenario, a dataset of multiple trajectories is released, each trajectory represents one record of the dataset. Second, one trajectory represents a database and each location can be considered as one record. Moreover, protection mechanisms either rely on *k-anonymity* and related privacy notions, or on *differential privacy*. K-Anonymity follows the intuition of hiding one user in a crowd of users. While this class of privacy notion is very intuitive, it cannot provide strong privacy guarantees [6, 7, 18, 24, 33]. Background knowledge can be exploited to derive knowledge from the protected dataset even when a dataset provides k-anonymity or a similar notion. Therefore, much research has focused on differential privacy (cf. Appendix B for details). The differential private SDD [23] mechanism is described in the following section, as we use this mechanism for our evaluation.

2.2 Sampling Distance and Direction

The Sampling Distance and Direction (SDD) mechanism is one approach to publish location trajectories while providing differential privacy. We consider the SDD mechanism as one target for our reconstruction attack because the mechanism can provide significantly better utility than the standard Laplace mechanism [23]. SDD

is considered as baseline in recent literature on protection mechanisms [7, 41] and follows an intuitive approach. As motivation for the approach, the authors refer to the publication of trajectories from ships in the Singapore Straits because unprotected trajectories could be utilised by pirates to plan and launch attacks [23].

The authors make the assumption that start and end locations are not vulnerable and can be published without protection. Starting from the first location, the exponential mechanism [35] is used to sample the *distance* and *direction* to the next location. The parameters of the exponential mechanism are chosen such that the sampling achieves ε-differential privacy. Moreover, the point defined by the sampled values must lie within a certain distance to the end point, to preserve utility. Otherwise, the sampling is repeated. The evaluation shows that the SDD mechanism achieves significantly better utility than the Laplace mechanism, especially for smaller values of ε (and hence, higher privacy levels) [23]. Hence, we target this particular mechanism for the evaluation of our attack.

2.3 Problem Statement

The protection of ship trajectories with the goal to prevent pirate attacks serves as motivation for the SDD mechanism [23]. Other authors [18] use stalking and interception of individuals as motivation. For both threats, the attackers do not necessarily require the exact coordinates. Instead, the pirates might successfully launch an attack as long as they manage to get into line of sight of the target ship, and for a stalker, an approximate route of the victim suffices.

However, through observation of the output trajectories produced by protection mechanisms, we observe structural differences to the genuine input trajectories. Structural differences include, for instance, that protected trajectories exhibit zigzag patterns not found in genuine ship trajectories or pass very close or over land, which is impossible for real ships (compare Figure 3 and 4 in [23]).

The goal of this article is to highlight the risk posed by a *Reconstruction Attack on Protected Trajectories (RAoPT)* which exploits such characteristics. We define the reconstruction attack as a method to reduce the distance between the original and the reconstructed trajectories (called *OR-Distance*) significantly compared to the distance between the protected and original trajectories (*OP-Distance*). I.e., the locations of the reconstructed trajectory are physically closer to the locations of the original trajectory than the locations of the protected trajectories. Reducing the OR-Distance significantly, potentially to the level that the reconstructed and original trajectory overlap, represents a serious security threat to the users in the released dataset. For instance, in case of ships, this would allow the planning of pirate attacks [23], or in case of individuals, this information could be used by stalkers [18]. While a perfect reconstruction of a protected trajectory is not realistic, a partial reconstruction yielding a close physical distance or even intersection, represents a serious privacy breach and security threat.

2.4 Threat Model

For the evaluation of RAoPT, we assume different levels of background information known to the adversary executing the attack.
Adversary 1: Full Knowledge. The strongest adversary with *full knowledge* knows which mechanism with which parameters have been used for the protection of the released trajectories. Moreover, this adversary has access to unprotected trajectory data with the same distribution as the target dataset for the training of the attack model. This adversary model is the best case for an attack as the model can be trained on data that has the same properties as the real data. However, the assumptions are very strong and not realistic in the real-world. We use this adversary model to examine the influence of different parameters on the reconstruction success.
Adversary 2: Partial Knowledge. The adversary with *partial knowledge* has no access to unprotected data with the same distribution but is aware of the used protection mechanism and parameters. Hence, the adversary must train the attack model with data from a different trajectory dataset, e.g., with a publicly available dataset. Using another dataset during training might lower the attack performance because the trajectories of the training set might not possess the same unique features as the target dataset. The adversary with partial knowledge is more realistic than adversary 1, as an adversary is unlikely to get access to unprotected data from the same source as the target dataset. The assumption that the protection mechanism is public knowledge is not unrealistic as security through obscurity [44] is generally discouraged such that the protection method might be published alongside the dataset. Hence, this adversary targets a real-world scenario.
Adversary 3: No Knowledge. In the worst-case, the adversary has *no background knowledge*, i.e., they have neither access to a dataset with the same distribution nor any information about the used protection mechanism or its parameters. Hence, the attack model has to be trained on data that might display different properties than the dataset that shall be attacked. Accordingly, the reconstruction success will be lower than for the previous two adversaries. We provide an overview of related work in the following section.

3 RELATED WORK

In this section, we first describe existing attacks on trajectory protection mechanisms, and then, summarise works applying deep learning to the trajectory domain.
Existing Attacks. Shao et al. [52] proposed an attack framework called *iTracker* that recovers original trajectories from a differential private release by exploiting the correlation between multiple trajectories. Previous attacks only used the information of a single trajectory and mostly relied on Markov models [52]. The iTracker framework is based on a location sparsity matrix and uses two approximation algorithms to converge towards the most probable original trajectories. The evaluation of the approach shows that iTracker is able to recover trajectories that are more similar to the original ones than any of the trajectories recovered by related work. To the best of our knowledge, this framework represents the most effective attack on differential private trajectory publication mechanisms and highlights that differential private mechanisms can be attacked. While this framework represents the closest work to our approach, iTracker is only evaluated on Laplace noise-based mechanisms. As described in Section 2.2, the Laplace noise-based mechanisms add substantially more distortion to a trajectory than more advanced approaches such as the SDD mechanism. Hence, recovery of Laplace noise-protected trajectories is less challenging than trajectories from more advanced protection schemes, as confirmed by our evaluation in Section 5. Unfortunately, a direct

comparison of our approach to iTracker was not possible, as we could not get sufficient implementation details from the authors to reproduce their results. Moreover, iTracker only utilises time and location information and cannot easily be extended to utilising semantic information. However, semantic properties can be exploited to improve the performance of attack mechanisms [34, 36, 53, 55].

Deep Learning-based Protection Mechanisms. Few approaches exist which apply deep learning to the trajectory privacy domain [7, 31, 48, 49]. However, all these approaches deploy deep learning for the publication of privacy-protected location trajectories. To the best of our knowledge, no attack mechanism based on deep learning exists. In 2018, Liu et al. [31] published a vision paper on the usage of Generate Adversarial Networks (GANs) [32] for the privacy-preserving publication of trajectories. The goal of the approach, called *trajGAN*, is the generation of synthetic trajectories that are similar enough to authentic trajectories to provide high utility for analyses. The proposed framework consists of a generator and a discriminator. While the generator tries to generate realistic trajectories, the discriminator has the goal to distinguish between synthetic and authentic trajectories. These two components learn from each other such that the synthetic trajectories become harder to distinguish from real trajectories with sufficient training. Rao et al. [49] built upon this vision paper and proposed the *LSTM-trajGAN* model, which utilises an LSTM [21] as the main building block for the GAN. This deep learning-based publication approach appears to achieve a better utility-privacy trade-off than traditional publication mechanisms. The evaluation shows that the synthetic trajectories maintain high utility for analyses while achieving a low trajectory user linking accuracy, which is one indicator of the privacy level. However, the usage of a deep learning model incurs higher computational costs than traditional publications schemes and also requires an initial time-consuming training process. Additionally, synthetic trajectories are not suitable for all use cases and the approach requires all input data to originate from a rather small geographical area. Qu et al. [48] follow a similar idea and propose a GAN to create a differential private synthetic dataset to publish location data collected through 5G networks. Chen et al. [7] utilise a Recurrent Neural Network (RNN) to predict a noisy dataset from the original dataset. This noisy dataset is then further processed to release a differential private dataset that hides the original trajectories. The methods used by these approaches inspire the design of our model which we introduce in the following section.

4 RECONSTRUCTION MODEL

In this section, we introduce the LSTM-based RAoPT model to address our research question **RQ1** defined in Section 1:

RQ1. Can an adversary (partly) reconstruct trajectories from a differential private trajectory release?

First, we provide an overview of the attack in Section 4.1. Second, we give an overview of pre-processing and encoding in Section 4.2. The structure of the RAoPT model is described in Section 4.3. Finally, we provide details on the training process in Section 4.4.

4.1 Overview

Differential private publication mechanisms commonly do not take geographical constraints into consideration [37]. The constraints

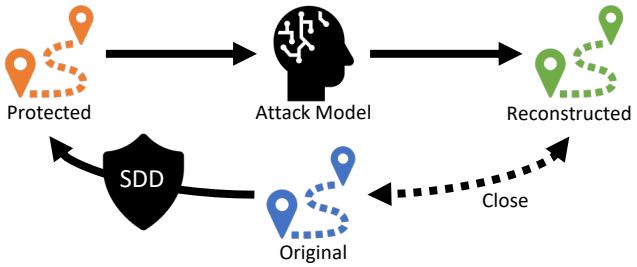

Figure 1: Overview of the attack. A differential private publication mechanism, e.g., the SDD mechanism [23], protects the original trajectories. These protected trajectories serve as input for the model with tries to reconstruct trajectories that shall be as close as possible to the original versions.

of real-world trajectories lead to structural differences between the original and the protected trajectories, such as the following: Ship trajectories generated by protection mechanisms might pass over land or through shallow waters, while genuine ship trajectories do not exhibit such behaviour. Moreover, the introduced randomness can lead to atypical zigzag patterns, whereas ships on the open ocean would most likely choose a straight path. Likewise, realistic taxi or car trajectories have to follow streets, and pedestrians cannot walk through houses and physical barriers. To address **RQ1**, RAoPT exploits these characteristics for a partial reconstruction of the original trajectories from the protected release.

A visual overview of the attack is provided in Figure 1. First, a protection mechanism providing differential privacy, e.g., the SDD mechanism (cf. Section 2.2), protects the *original* trajectories containing private information. Second, the resulting *protected* trajectories are fed into the RAoPT model. This model returns a *reconstructed* trajectory for each protected trajectory. The goal of the model is to generate reconstructed trajectories that are close to the original trajectories. Our attack is successful if the distance between the reconstructed and the original trajectories, called *OR-Distance*, is significantly smaller than the distance between the protected and the original trajectories (*OP-Distance*). Before describing the details of the RAoPT model in Section 4.3, we describe the trajectory encoding used as input for the model in the following section.

4.2 Trajectory Encoding and Pre-Processing

The general representation of location trajectories is described in Section 2.1. In this section, we explain how a trajectory is pre-processed and encoded before it can be used as input for the RAoPT model. Trajectories consist of a sequence of locations which can each be composed of multiple properties. For our attack model, we only utilise the location and time information as these values are the information amount contained in many available datasets [12, 59, 60]. Through usage of the time information, in particular hour-of-day and day-of-week features, we showcase the capability of the RAoPT model to utilise additional semantic knowledge without overstating the attack success by using semantic information that might not be available to a real-world attacker. Due to the usage of an embedding and feature fusion layer (cf. Section 4.3), the model can be extended by further properties, such as POIs. However, the less correlated information per location a dataset provides, the

Point	Latitude	Longitude	Hour of Day	DoW
Dim.	1 (float)	1 (float)	24 (binary)	7 (binary)
1	-0.80	2.34	0 1 ... 0	1 ... 0
2	-0.80	2.33	0 1 ... 0	1 ... 0
...	...	...	0 1 ... 0	1 ... 0
38	1.23	1.45	0 1 ... 0	1 ... 0
39	1.24	1.45	0 1 ... 0	1 ... 0

Table 1: Trajectory Encoding. The table shows an encoded trajectory consisting of 39 locations. Each location contains a latitude, longitude, hour of day and day of week (DoW).

harder a reconstruction becomes. Therefore, only using time and location represents the worst-case for a reconstruction attack.

A matrix represents a trajectory T and each row of this matrix corresponds to one measurement point t_i of the trajectory. An example of an encoded trajectory is shown in Table 1. The location information is represented by latitude and longitude values. However, instead of using these values directly, we compute the offsets from a central reference point. I.e., for the reference point $(lat_0, lon_0) = (40.0, 115.0)$, the location point $(39.2, 117.34)$ is represented as $(-0.80, 2.34)$. This standardisation, motivated by the LSTM-TrajGAN encoding [49], allows the model to better learn the spatial deviation patterns [49]. The time information is dissembled into two one-hot encodings. The hour-of-day is represented by a 24-dimensional binary vector, and the day-of-week by a 7-dimensional vector. Both vectors contain exactly one 1-value. For instance, the first point in Table 1 has been recorded at 2 am on a Monday. Other categorical values could be added to the encoding via similar one-hot encodings. For instance, the encoding could be extended by location types such as gym, shopping centre, or medical centre.

Before feeding the encoding into the RAoPT model, the latitude, and longitude deviations are scaled through max normalisation [54], i.e., all values are divided by the maximal value in the dataset. Moreover, the trajectory is zero padded to the maximal length expected for any trajectory, i.e., rows with all values set to 0 are appended to the bottom of the matrix. These padded rows are marked through a masking layer in the model such that they do not have any influence on the training or reconstruction. After completion of the described pre-processing, the trajectories can be used by the model described in the following section.

4.3 RAoPT Model

The generated encodings can be used for training of the model, or for reconstruction. Figure 2 shows the structure of the RAoPT model. Initially, the protected input trajectories are encoded and pre-processed as described in the previous section.

Masking Layer. The input is masked through a *masking layer* in order to avoid the influence of the padded points on the output.

Embedding Layer. Next, the input is split into three separate features: *location information* (green), i.e., longitude and latitude, *hour-of-day* (light blue), and *day-of-week* (dark blue). In case the trajectories include semantic information, more features could be added here. We treat the hour-of-day and day-of-week as two examples for our model's capability to add semantic information. Any other features can be added in the same way by passing different

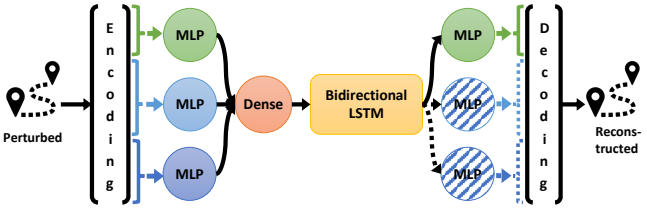

Figure 2: Perturbed trajectories are encoded to a vector containing location (green), and time and/or semantic information (light and dark blue). These features are embedded by MLPs, concatenated in a feature fusion layer, and fed into the bidirectional LSTM layer, followed by the output layers. Finally, the output is decoded to the reconstructed trajectories. The dashed MLPs are optional and not evaluated.

parameters during the model's initialisation. We decided not to include further semantic features in our evaluation because additional information potentially improves the model's training. Thus, the evaluated setting represents the worst-case and allows for better generalizability as not every dataset contains additional semantic information. The model embeds each feature separately through a Multi-Layer Perceptron (MLP) consisting of a dense layer followed by a Rectified Linear Activation (ReLU) activation function. The MLPs use the same weights for all points of the trajectory (realised through a *TimeDistributed layer*). The units of the dense layer depend on the embedded feature. For the location information, we use 64 units, as this value has worked well for the LSTM-TrajGAN approach [49]. For hour and day, we use the same number of units as the size of the encoding, i.e., 24 and 6 units.

Feature Fusion. Then, a dense layer with 100 units and a ReLU activation function fuses the concatenation of the embeddings.

LSTM Layer. Consecutively, we feed the output of the feature fusion into a bidirectional LSTM layer with 100 units. This layer produces one output for each point of the trajectory.

Output Layer. The output of the bidirectional LSTM is processed through separate MLPs which are again applied to each slice of the sequence with the same weights. Each of the MLPs consists of a dense layer followed by a Hyperbolic Tangent (tanh) activation for numerical outputs, such as latitude/longitude, or by a softmax activation in case of categorical output features. As the considered protection mechanisms only perturb the location information, we use two MLPs with one unit each to generate the reconstructed latitude and longitude. If the model should also reconstruct other information such as timestamps, further MLPs can be added. However, adding information which was not perturbed and does not require reconstruction is not beneficial as it distracts the model's from the important values. Moreover, we scale the outputs for latitude and longitude with the inverse scale factor used during pre-processing (cf. Section 4.2) as the outputs of tanh ranges from -1 to 1.

Post-Processing. The outputs of the actual model have a similar format to the encoding after the pre-processing (cf. Section 4.2). To retrieve useful reconstructed trajectories, the pre-processing steps have to be inverted. First, the reference point is added to convert the spatial deviations into absolute latitude and longitude values. Second, the padded points are removed such that the reconstructed trajectory has the same length as the protected trajectory.

Finally, the resulting encoding can be decoded into a trajectory. Thereby, any unperturbed information that was not reconstructed can be added back into the trajectory, e.g., the timestamp if only the locations were reconstructed.

Loss Function. We implemented a custom loss function which computes the Mean Absolute Error (MAE) of the Euclidean distance between the output and the ground truth trajectories used during training. However, instead of computing the Euclidean distance directly by treating latitude and longitude as coordinates, we compute the haversine distance [50] between each pair of locations of the compared trajectories. For trajectories that have been perturbed with very large amounts of noise (which can be identified by invalid latitude or longitude values), we use the standard Mean Squared Error (MSE) loss function instead because our custom loss function requires valid latitudes and longitudes. The loss function concludes the description of our RAoPT model. We provide implementation details and information on hyperparameters in Section 5.5.

4.4 Training

Before the model can be used for the reconstruction of trajectories, it needs to be trained. For the generation of training data, the adversary uses trajectories they have access to, for instance, openly available datasets such as T-Drive [59], GeoLife [60], or Foursquare [12]. The influence of using trajectories with different distribution than the target dataset is evaluated in Section 5.6.5. The adversary perturbs these trajectories with an available protection mechanism. In the best case, they know about the used protection mechanism of the target dataset to attack and use the same mechanism (cf. the threat model in Section 2.4). These generated pairs of original and protected trajectories serve as training data. The protected trajectories represent the model's input, while the original trajectories serve as ground truth. Then, the trained model can be used to reconstruct trajectories from the target dataset. We evaluate the effectiveness of our attack in the following section.

5 EVALUATION

In this section, we present the results of our RAoPT model's evaluation. We begin with a description of the used T-Drive and GeoLife datasets in Section 5.1, followed by the applied pre-processing in Section 5.2. In Section 5.3, we provide information about the protection mechanisms which we consider for the evaluation. Section 5.4 contains information about the used metrics and Section 5.5 provides implementation details of the RAoPT model. Finally, we present and discuss the results of our measurements in Section 5.6.

5.1 Datasets Description

To verify the generalizability of the attack, we base our measurements on two datasets. First, the T-Drive [59] dataset, which consists of the trajectories of 10 357 taxis collected over one week in the area of Beijing. Second, the GeoLife [60] dataset, which contains the trajectories of 182 users with different modes of transportation collected throughout a period of three years. While the T-Drive dataset only contains trajectories of similar types, i.e., cars on a street, the GeoLife dataset is more diverse as it contains walking, hiking, running, cycling, driving, and even flights. The usage of datasets with such different properties allows us to examine the behaviour of our attack in different settings. Both datasets contain latitude, longitude, and timestamp information. Moreover, both datasets allow attributing each trajectory to a certain user (or taxi). In addition, the GeoLife dataset contains the altitude information for each point. However, due to consistency, we do not utilise these values. Neither do we enrich the datasets with any semantic knowledge. Before using the trajectories for our evaluations, we perform a pre-processing step which we describe in the following section.

5.2 Pre-Processing

We undertake a pre-processing step to sanitise the datasets, as clean data is crucial for good deep learning results [29]. First, we remove outliers by deleting all location points which lie outside a bounding box defined by the 99 % percentile for T-Drive and by the 95 % percentile for GeoLife. We use a lower percentile for the GeoLife dataset because it contains locations on other continents which cannot be handled with the reference point approach described in Section 4.2. Second, duplicates are dropped, i.e., locations with the same timestamp. In case both duplicates refer to the same location, the second point is removed. In case the two duplicates correspond to the same timestamp, but the locations differ, we assume that the point with a larger distance to the previous and following location is the outlier which can be removed. Third, speed outliers are removed. As described in Section 5.3, we utilise the SDD mechanism for our evaluation. However, the mechanism requires an upper bound on the speed of any user in the trajectory dataset to be defined. Therefore, we drop all locations that require that a user has travelled at a speed that is faster than the 99 % percentile for the dataset. For the T-Drive dataset, all points indicating a speed over 90 $\mathrm{km\,h^{-1}}$ are dropped, and for GeoLife all speeds over 100 $\mathrm{km\,h^{-1}}$.

Both datasets contain trajectories of varying lengths. The T-Drive dataset only contains one trajectory per taxi, which spans the time period of a week, while the GeoLife dataset contains multiple trajectories of different lengths per user. We define a trajectory as the locations of one uninterrupted trip, for instance, one workday of a taxi-driver in case of the T-Drive dataset. To depict this, we split the trajectories based on a time gap of 11 min (To include trajectories with one GPS reading every 10 min) for T-Drive, and 20 s (the 99 % percentile) for GeoLife. Finally, we remove trajectories that are shorter than 10 locations, as they do not contain much information content. We also remove trajectories longer than a threshold of 100 points for T-Drive and 200 points for GeoLife. The reason for defining an upper threshold is that the deep learning model requires padding of all trajectories to the same size for efficient training and reconstruction. After this pre-processing, the processed T-Drive dataset contains 163 006 trajectories, and the processed GeoLife dataset 90 146, respectively. Now, the trajectories can be protected by a differential private publication mechanism, as described in the following section, to generate the inputs for the RAoPT model.

5.3 Protection Mechanisms

Our attack targets differential private trajectory publication mechanisms. A number of approaches [5, 22, 23, 28, 30, 40] have been proposed, but for most, no implementation is openly available such that a time-consuming re-implementation is required. Related works

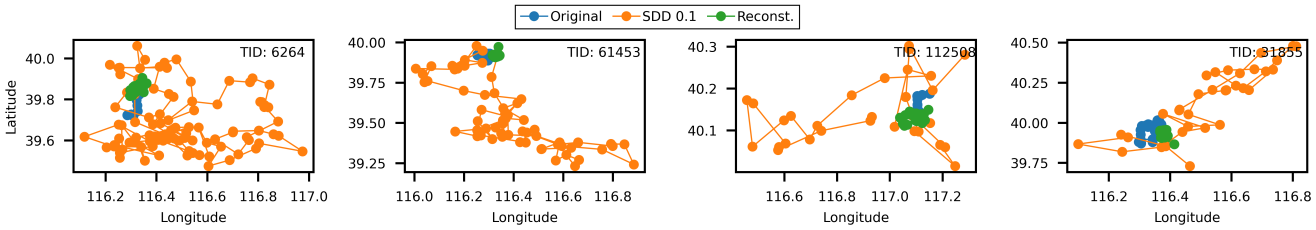

Figure 3: Example Trajectory Reconstruction. The figures show four randomly chosen trajectories from the T-Drive dataset (original), after protection with the SDD mechanism ($\varepsilon = 0.1$), and after reconstruction by the RAoPT model.

based their evaluation on the simple Laplace mechanism (cf. Section 3), however, we assume that reconstruction from more sophisticated approaches adding less total noise is a harder problem. To achieve realistic results, we decided to consider two different protection methods. First, we use a simple Laplace noise based mechanism as a baseline. In particular, we consider the **CNoise mechanism** defined by Jiang et al. [23] because it is the best performing Laplace noised-based mechanism examined in the paper. Second, we utilise the **SDD mechanism** [23] (cf. Section 2.2) which is considered a state-of-the-art protection mechanism [7, 41].

We implemented both mechanisms as close as possible to the descriptions in the original paper, however, had to make a few minor changes to the SDD mechanism. For long trajectories, the mechanism frequently got stuck on line 11 of the original algorithm definition (Algorithm 5 in [23]). To avoid infinite runtime, we restart the entire algorithm in case the inner loop does not terminate after 1000 runs. Moreover, after completion of the standard mechanism, we also perturb the start and end point by sampling a distance and direction from the second (last) point, as not in every scenario start and end point are public knowledge. As we restart the entire algorithm in case of the first modification and add further perturbation in case of the second, both modifications do not lower the level of differential privacy provided by the mechanism.

Both mechanisms require a sensitivity M for each dataset to add the appropriate amount of noise to achieve differential privacy. We choose M to be 16 500 m as this is the sensitivity computed for the T-Drive dataset by multiplying the maximal speed (90 km h^{-1}) with the sampling rate (11 min). The GeoLife dataset would allow for a lower sensitivity due to the finer sampling rate. Due to consistency we choose the larger value 16 500 m for all measurements, as a larger choice is valid while a too low sensitivity breaks differential privacy guarantees. Furthermore, the locations provided in latitude lat and longitude lon need to be converted into Cartesian coordinates before the application of the protection mechanisms. For simplicity, we use offset coordinates from a central point (lat_0, lon_0), as both pre-processed datasets only contain locations within a certain bounding box. With 111 319.44 m as the average distance between two degrees of latitude, we use the following formula:

$$x = 111\,319.44 * \cos lat_0 * (lon - lon_0)$$
$$y = 111\,319.44 * (lat - lat_0)$$

After application of the mechanism, we transform the locations back into latitude and longitude values. In our measurements, we consider different values for the privacy parameter ε (cf. Section B) which, in practice, usually takes values between 0.01 and 10 [16]. For the measurements with the SDD mechanism, we focus on values from this range. For the CNoise mechanism, we additionally consider $\varepsilon = 100$ as the mechanism is faster to apply, and the generated trajectories are very close to the original ones, which is the worst-case scenario for our attack.

5.4 Metrics

The goal of the reconstruction attack is to minimise the physical distance between the locations of the original and the reconstructed trajectories. To measure this distance, we consider three metrics: (1) the Euclidean distance, (2) the Hausdorff distance, and (3) the Jaccard index of the trajectories' convex hulls. Both the Euclidean distance [23, 52] and the Hausdorff distance [22, 28, 33, 49, 52] have been widely used to measure the distance of trajectories. To compute the distance between two locations defined through latitude and longitude, we use the haversine formula [50]. To simplify the discussion, we focus on metrics (1) and (2) in the following. We explain the Jaccard Index and discuss its suitability for our evaluation in Appendix C, and record all results in Table 5 in the appendix.

5.5 Implementation

We implemented the RAoPT model presented in Section 4.3 with Keras [8] contained in TensorFlow 2.4.1 [1] using Python 3.9. Our implementation relies on NumPy [20] in version 1.19.2, and pandas [42] in version 1.4.2. We compute the haversine distance with the haversine library [11]. The model uses the Adam optimiser [25] with a learning rate of 0.001. We choose a batch size of 512, trained our model for maximally 500 epochs, but terminated the training process with an early stop patience of 50 epochs. For other hyperparameters, the default values of the libraries are used. In test cases using one dataset, we use 5-fold cross-validation, i.e., we perform 5 runs, train on 80 % of the dataset and test on the other 20 %. For the test cases using different datasets,we perform 5 independent runs on the entire datasets.

5.6 Results

In this section, we provide the results of our measurements. All bar plots show the average distance reduction (Formula in Appendix D) and the 99 % confidence intervals as error bars. We omit the results for the Jaccard index in the plots to make the appearance clearer. A table containing the results of all performed measurements can be found in Appendix F. We consecutively discuss the adversaries defined through our threat model in Section 2.4, beginning with

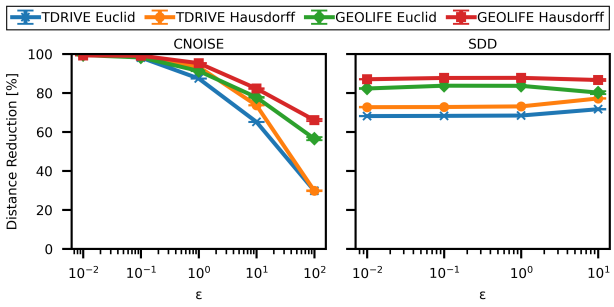

Figure 4: The plot shows the percentage reduction of the OR-Distance compared to the OP-Distance. The left plot shows the results for CNoise, the right plot for the SDD mechanism.

adversary 1 in Sections 5.6.1-5.6.4, followed by adversary 2 in Section 5.6.5, and finally, adversary 3 in Section 5.6.6. In addition, we provide runtime measurements in Appendix E.2.

5.6.1 Adversary 1: T-Drive Dataset. First, we examined the performance of the RAoPT model on the T-Drive [59] dataset with different protection mechanisms. Figure 4 displays the average reduction of the OR-Distance compared to the OP-Distance for both protection mechanisms and different values of ε. In case of the SDD mechanism, the reconstruction attack can reduce the distance to the original trajectories by over 68 % for all choices of ε. The Jaccard index is increased through reconstruction by over 180 % on average. The ε parameter has only limited influence on the outputs of the mechanism due to the way the mechanism is designed. This finding is in-line with the results of the original paper [23].

In case of the CNoise mechanism being used, the reconstruction even reduces the distances by far above 80 % for $\varepsilon \leq 1$. For $\varepsilon = 10$, the distance is still reduced by 65 % (Euclidean distance) and 74 % (Hausdorff distance), respectively. For $\varepsilon = 100$, the reconstructed trajectories are only 30 % closer to the original trajectories. In this setting, the CNoise mechanism barely perturbs the trajectories, such that the protected trajectories are already very close to the originals. However, such a high value for ε is very unlikely to be used in the real world [16] as it cannot provide much privacy. The average Jaccard index is increased in all cases, from 11 % for $\varepsilon = 100$ to an increase by factor 430 573 for $\varepsilon = 0.01$.

By means of illustration, four randomly chosen trajectories, their SDD ($\varepsilon = 0.01$) protected versions, and the reconstruction results are displayed in Figure 3. For all these trajectories from the T-Drive dataset, the reconstructed trajectories are not only significantly closer to the original ones, but the structure, e.g., in terms of density and space between trajectories, is much more similar. This finding is captured by the significant increases of the Jaccard index which resembles the similarity of the trajectories' activity spaces. An adversary with the intention to intercept the user of the original trajectories has a reasonable chance of success by using the reconstructed trajectories. In the following section, we perform the reconstruction attack on our second dataset.

5.6.2 Adversary 1: GeoLife Dataset. To show that RAoPT is generally applicable to different datasets, we performed the same measurements on the GeoLife [60] dataset. The results are also displayed in Figure 4, alongside the results for the T-Drive dataset. The figure depicts that the reduction of the distances is very similar to the T-Drive results, just slightly higher in all cases. We presume that this is caused by the GeoLife dataset being more diverse and hence, the training leads to a more robust model which generalises better to the test set. The average Jaccard indices can even be increased by over 4900 % in all cases except for CNoise with $\varepsilon = 100$, where the increase is still larger than 600 %. Next, we examined the influence of training and testing on datasets protected with different ε.

ID	Mechanism	ε Train	ε Test	Euclidean	Hausdorff
27	CNoise	1.0	10.0	24.3 %	46.2 %
28	CNoise	10.0	1.0	72.5 %	79.3 %
29	SDD	0.1	1.0	68.4 %	73.1 %
30	SDD	1.0	0.1	68.3 %	72.8 %

Table 2: Except for the varied ε, the same parameters have been used for train and test set based on the T-Drive dataset. The table shows the average distance reductions.

ID	Train	Test	ε	Euclidean	Hausdorff
31	CNoise	SDD	1.0	27.7 %	44.9 %
32	SDD	CNoise	1.0	53.0 %	70.3 %

Table 3: All cases use the T-Drive dataset. Except for the protection mechanism, the same parameters have been used for train and test set, including the same ε value.

5.6.3 Adversary 1: Influence of Different ε. In the real world, an adversary might not know the exact parameters used for the released dataset. Therefore, we examined four cases in which the reconstruction model is trained on trajectories that are protected with ε set to a different value than the test set. We consider 0.1 and 1 for the SDD mechanism, as these are in the middle of the common range of values (cf. Section 5.3), and 1 and 10 for CNoise, as lower values for CNoise lead to a level of perturbation that renders the protected trajectories meaningless. Table 2 shows the average distance reduction through the reconstruction attack. Measurement 27 shows that the reconstruction of trajectories protected with CNoise and $\varepsilon = 10$ performs worse when trained on a dataset with $\varepsilon = 1$. This is caused by the fact that CNoise($\varepsilon = 1$) adds substantially more perturbation to the trajectories compared to CNoise $\varepsilon = 10$. Therefore, the model modifies the input trajectories more than necessary. The opposite test case 28 reduces the distances only 15 % less than training on a dataset protected with the same parameters.

In case of the SDD mechanism, the results are very similar to the cases in Section 5.6.1 with training on the same parameters. Both the Euclidean and the Hausdorff distance can still be reduced by over 68 %, and the average Jaccard index can be increased by over 180 %. This observation can be explained by the fact that the outputs of the SDD mechanism are barely affected by the choice of ε (cf. Section 5.6.1). In some cases, not only the parameter, but the entire used protection algorithm might be unknown to the adversary. We examine this situation in the following section.

5.6.4 Adversary 1: Influence of Different Mechanism. Table 3 shows the distance reduction for two cases where the training dataset is protected with a different mechanism than the test set. We fix ε to

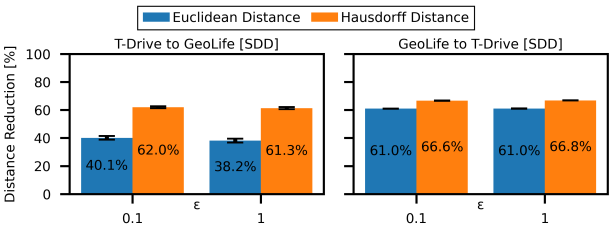

Figure 5: Adversary 2: The figure shows the results for the transfer from one dataset to another. All test cases use the SDD mechanism as protection with different choices for ε.

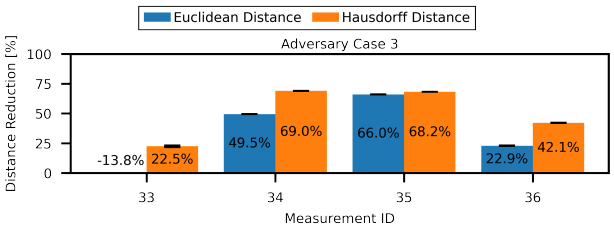

Figure 6: Adversary 3: The figure shows 4 measurements for which all considered parameters have been modified.

1 to determine the influence of a varied mechanism only. While the reconstruction attack still reduces the OR-Distance by at least 27 % in comparison to the OP-Distance, the reconstruction success is significantly smaller than training and testing on datasets protected with the same mechanism. This might be caused by the different amounts of perturbation added by different mechanisms, and by different characteristics of the algorithms. While the protection mechanism being public knowledge is not an unrealistic assumption (cf. Section 2.4), and adversary has rarely access to training data from the same source as the target dataset. Therefore, we consider the case of different datasets in the following section.

5.6.5 Adversary 2: Dataset Transfer. A real-world adversary might not have access to trajectories of the same distribution as the trajectories they try to attack. Therefore, the RAoPT model needs to be trained on a different dataset, e.g., a publicly available dataset. To investigate the attack performance in such a setting, we performed measurements training our model on one dataset and using the other dataset as the test set. The results for protection with the SDD mechanism are shown in Figure 5. The corresponding measurements for CNoise are provided in Appendix E.1.

Figure 5 still shows reductions of approx. 61 % for the Euclidean and approx. 67 % for the Hausdorff distance, when transferring from GeoLife to T-Drive. The opposite direction is less successful with approx. 40 % reduction of the Euclidean and approx. 61 % for the Hausdorff distance. This finding could be caused by the higher diversity of the GeoLife trajectories which include trajectories that are similar to the T-Drive trajectories, while the T-Drive dataset does not contain all transportation modes of the GeoLife dataset. The average Jaccard index shows increases in all mentioned cases.

If the CNoise ($\varepsilon = 1$) mechanism is used for protection, the reconstructed trajectories show an over 90 % reduced Hausdorff distance. The attack reduces the Euclidean distance by 76.7 % (T-Drive to GeoLife) and 82.9 % (GeoLife to T-Drive), respectively. While for CNoise ($\varepsilon = 10$), the transfer from T-Drive to GeoLife cannot reduce the Euclidean distance at all, the other direction can even achieve a reduction by 48 % and 59 % for the Euclidean and Hausdorff distances, respectively.

In conclusion, these measurements indicate that RAoPT is also successful for adversary 2, who does not have access to optimal training data. Therefore, this evaluation highlights the real-world danger posed by the proposed reconstruction attack. To consider an even more realistic scenario, we provide measurements assuming no background knowledge by the adversary in the following section.

5.6.6 Adversary 3: No Background Knowledge. The worst-case scenario for an adversary assumes that no background knowledge about the dataset or protection is known. In particular, the training dataset does not match the testing dataset in terms of properties, the used protection mechanism is unknown as well as the choice of the ε parameter. We picked four such cases, differing both the dataset and the protection mechanism for the training dataset, as well as different common values for ε of either 0.1 or 1. The concrete specifications are shown in Table 4, and the results in Figure 6.

In case 33, the reconstruction is not successful, as it combines the transfer from CNoise protected trajectories to SDD protected trajectories and the transfer from a larger ε to a smaller value. Case 34, which switches the direction of dataset, mechanism and parameter transfer, on the other hand, allows a distance reduction of 49 %/59 % (Euclidean/Hausdorff). Case 35 shows even better reconstruction success with a 66 % reduction of the Euclidean and 68 % for the Hausdorff distance. Finally, case 36 only allows for a limited reduction of 23 %, and 42 %, respectively. Notably, the average Jaccard index is significantly increased through reconstruction for cases 34 and 35, stays nearly unaltered for case 33 (8 % increase) but even shows a decrease for case 36.

These results indicate that even an adversary without background knowledge can execute a successful reconstruction attack and in this way, harm the privacy of contained users. Moreover, the results show that certain parameter choices for the training set can be helpful for a more successful reconstruction. First, training on trajectories protected with less perturbation than the target dataset appears to yield better results, i.e., training on a set with a larger ε parameter or SDD instead of CNoise. Hence, an adversary without knowledge about the protection mechanism should train on a dataset protected with a mechanism adding limited noise. Second, transferring from a more diverse training set seems to be advantageous, which matches deep learning best practices. By choosing appropriate parameters and using a training dataset with characteristics close to the target dataset, an adversary can successfully reconstruct trajectories from the protected set. This reconstruction represents a threat to all users whose data is contained in the released and seemingly protected dataset.

6 DISCUSSION

In this section, we further discuss our findings, mention possible countermeasures, and outline opportunities for future work. The goal of this article was to investigate research question **RQ1**:

ID	DS Train	DS test	Train	Test	ε Train	ε Test
33	T-Drive	GeoLife	CNoise	SDD	1.0	0.1
34	GeoLife	T-Drive	SDD	CNoise	0.1	1.0
35	T-Drive	GeoLife	SDD	CNoise	1.0	0.1
36	GeoLife	T-Drive	CNoise	SDD	0.1	1.0

Table 4: Adversary 3: Specifications of our four worst-case measurements.

RQ1. Can an adversary (partly) reconstruct trajectories from a differential private trajectory release?

Considering the results of our evaluation, we can answer this question affirmatively. The measurements described in the previous section highlight that the proposed Reconstruction Attack on Protected Trajectories (RAoPT) successfully reduces the OR-Distance compared to the OP-Distance in most cases. The results show, that the attack is not limited to a knowledgeable adversary, but an adversary that needs to transfer from one dataset to another can still achieve distance reductions of approx. 60 % or more for most considered protection mechanisms on some datasets.

It is important to understand that our attack does not show a vulnerability of the mathematically proven privacy notion of differential privacy (cf. Appendix B). Rather than targeting the notion itself, the reconstruction attack targets the concrete mechanism of achieving differential privacy. Our results do *not* indicate that all differential private publication mechanisms have to be susceptible to our RAoPT. The issue is rather that current protection mechanism do not consider the characteristics of genuine trajectories sufficiently. Accordingly, the design of improved differential private publication mechanisms is recommended for future work.

As mentioned in Section 3, a direct comparison to the iTracker [52] attack is not possible due to missing implementation details. However, iTracker only targets Laplace perturbation, similar to the CNoise mechanism. In particular, iTracker's evaluation considers the Laplace mechanism with $0.1 \leq \varepsilon \leq 0.9$. As shown in Section 5, the reconstruction of trajectories protected with CNoise and $\varepsilon \leq 1$ allows for very high distance reductions. For adversary 1 with training and testing on sets with the same parameters, RAoPT manages to reduce the distances by 87 % or more. Also, both worst-case measurements targeting CNoise (cf. Section 5.6.6) achieve very high reconstruction rates of 49 % - 69 %.

Countermeasures. The evaluation results clearly show that mechanisms adding less perturbation are less vulnerable to the reconstruction attack. Accordingly, the design of protection mechanisms adding a minimal amount of noise while sufficiently protecting privacy combines high utility with a protection against a reconstruction attack. Approaches such as LSTM-TrajGAN [49] are by design more robust against reconstruction attacks as the generator of the GAN is trained to generate trajectories which are indistinguishable from authentic trajectories. However, as described in Section 3, the approach is not applicable in all scenarios. Accordingly, extending similar approaches to further use cases can represent an effective countermeasure against reconstruction attacks. Apart from that, the reconstruction attack exploits the different characteristics of protected and original trajectories. Hence, designing protection mechanisms that produce trajectories with realistic characteristics can effectively counteract reconstruction attacks.

Future Work. In future work, we intend to highlight the privacy threat of the attack by comparing the success of a TUL attack before and after the reconstruction through RAoPT. Due to the usage of location offsets (cf. Section 4.2), the model can only handle trajectories from a limited geographical area. Future work could look into the generalisation of the attack to trajectories with arbitrary locations. Moreover, the influence of adding semantic features to the trajectories on the reconstruction success could be determined, as semantic information can be exploited for more accurate attacks [34, 46, 49]. Lastly, the focus of further research should lie on the development of novel privacy-preserving trajectory publication mechanisms, which provide both high levels of utility and privacy, and are not susceptible to reconstruction attacks.

7 CONCLUSION

While location trajectories offer huge potential for many use cases such as navigation, marketing, or pandemic control, this datatype is very sensitive because it can reveal religious, political or sexual beliefs. Therefore, trajectory datasets require appropriate protection before publication. Due to its strong theoretical guarantees, differential privacy represents the basis for most recent publication mechanisms. However, the perturbation caused by these publication mechanisms makes it possible to distinguish published trajectories from authentic trajectories. Structural differences, e.g., cars not following roads, can be exploited to partially recover the original trajectories from a differential private publication, and hence, impair the privacy of individuals in the dataset. To highlight these shortcomings, we propose the Reconstruction Attack on Protected Trajectories (RAoPT). The LSTM-based model can significantly reduce the distance of protected trajectories to the original versions. In addition to simple perturbation-based protection, we target the more practical SDD publication mechanism. To measure the success of our attack, we compute the reduction of the Euclidean and Hausdorff distances, as well as the increase of the Jaccard index of the convex hull. On the T-Drive dataset both distances can be reduced by over 68 % while the Jaccard index can be increased by over 180 % for trajectories protected with either protection method and $\varepsilon \leq 1$. An adversary that has to train the RAoPT model on a different dataset can still successfully reconstruct trajectories, as the transfer from the GeoLife to the T-Drive dataset allows for an over 60 % distance reduction considering protection with the SDD mechanism and $\varepsilon = 0.1$ or $\varepsilon = 1$, and a 30 % increased Jaccard index. These results indicate that a reconstruction attack represents a significant privacy threat to existing trajectory publication mechanisms. Thus, further research on improved privacy-preserving publication mechanisms for trajectory datasets is required.

ACKNOWLEDGMENTS

The authors would like to thank UNSW, the Commonwealth of Australia, and the Cybersecurity Cooperative Research Centre Limited for their support of this work. The authors would like to thank all the anonymous reviewers for their valuable feedback.

REFERENCES

[1] Martin Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, et al. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. https://www.tensorflow.org/

[2] Osman Abul, Francesco Bonchi, and Mirco Nanni. 2008. Never Walk Alone: Uncertainty for Anonymity in Moving Objects Databases. In *2008 IEEE 24th Int. Conf. Data Eng.*, Vol. 00. IEEE, 376–385. https://doi.org/10.1109/ICDE.2008.4497446

[3] Osman Abul, Francesco Bonchi, and Mirco Nanni. 2010. Anonymization of Moving Objects Databases by Clustering and Perturbation. *Inf. Syst.* 35, 8 (Dec. 2010), 884–910. https://doi.org/10.1016/j.is.2010.05.003

[4] Hugo Barbosa, Marc Barthelemy, Gourab Ghoshal, Charlotte R. James, Maxime Lenormand, Thomas Louail, Ronaldo Menezes, José J. Ramasco, Filippo Simini, and Marcello Tomasini. 2018. Human Mobility: Models and Applications. *Phys. Rep.* 734 (March 2018), 1–74. https://doi.org/10.1016/j.physrep.2018.01.001

[5] Yang Cao, Yonghui Xiao, Li Xiong, Liquan Bai, and Masatoshi Yoshikawa. 2021. Protecting Spatiotemporal Event Privacy in Continuous Location-Based Services. *IEEE Trans. Knowl. Data Eng.* 33, 8 (Aug. 2021), 3141–3154. https://doi.org/10.1109/TKDE.2019.2963312

[6] Rui Chen, Benjamin C. M. Fung, and Bipin C. Desai. 2011. Differentially Private Trajectory Data Publication. *arXiv* abs/1112.2 (Dec. 2011), 1–12. http://arxiv.org/abs/1112.2020

[7] Si Chen, Anmin Fu, Jian Shen, Shui Yu, Huaqun Wang, and Huaijiang Sun. 2020. RNN-DP: A New Differential Privacy Scheme Base on Recurrent Neural Network for Dynamic Trajectory Privacy Protection. *J. Netw. Comput. Appl.* 168, February (2020), 102736. https://doi.org/10.1016/j.jnca.2020.102736

[8] François Chollet et al. 2015. Keras. https://keras.io

[9] OpenStreetMap Contributors. 2017. OpenStreetMap. https://www.openstreetmap.org/

[10] Yves-Alexandre de Montjoye, César A. Hidalgo, Michel Verleysen, and Vincent D. Blondel. 2013. Unique in the Crowd: The Privacy Bounds of Human Mobility. *Sci. Rep.* 3, 1 (Dec. 2013), 1–5. https://doi.org/10.1038/srep01376

[11] Julien Deniau et al. 2013. Haversine. https://github.com/mapado/haversine/

[12] Dingqi Yang, Daqing Zhang, Vincent W Zheng, and Zhiyong Yu. 2015. Modeling User Activity Preference by Leveraging User Spatial Temporal Characteristics in LBSNs. *IEEE Trans. Syst. Man Cybern. Syst.* 45, 1 (Jan. 2015), 129–142. https://doi.org/10.1109/TSMC.2014.2327053

[13] Cynthia Dwork. 2008. Differential Privacy: A Survey of Results. In *Theory and Applications of Models of Computation*, Manindra Agrawal, Dingzhu Du, Zhenhua Duan, and Angsheng Li (Eds.). Vol. 4978 LNCS. Springer Berlin Heidelberg, Berlin, Heidelberg, 1–19. https://doi.org/10.1007/978-3-540-79228-4_1

[14] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating Noise to Sensitivity in Private Data Analysis. In *Theory of Cryptography Conference*. Vol. 3876 LNCS. Springer, Berlin, Heidelberg, 265–284. https://doi.org/10.1007/11681878_14

[15] Cynthia Dwork and Aaron Roth. 2013. The Algorithmic Foundations of Differential Privacy. *Found. Trends® Theor. Comput. Sci.* 9, 3-4 (2013), 211–407. https://doi.org/10.1561/0400000042

[16] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. 2014. RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response. In *Proc. 2014 ACM SIGSAC Conf. Comput. Commun. Secur.* ACM, 1054–1067. https://doi.org/10.1145/2660267.2660348

[17] Lorenzo Franceschi-Bicchierai. 2015. Redditor Cracks Anonymous Data Trove to Pinpoint Muslim Cab Drivers. https://mashable.com/archive/redditor-muslim-cab-drivers

[18] Fernanda Oliveira Gomes, Douglas Simoes Silva, Bruno Machado Agostinho, and Jean Everson Martina. 2018. Privacy Preserving on Trajectories Created by Wi-Fi Connections in a University Campus. In *2018 IEEE Int. Conf. Intell. Secur. Inform. ISI*. IEEE, 181–186. https://doi.org/10.1109/ISI.2018.8587319

[19] Google. 2022. Google Maps. https://www.google.com/maps

[20] Charles R Harris, K Jarrod Millman, Stéfan J van der Walt, Ralf Gommers, Pauli Virtanen, et al. 2020. Array Programming with NumPy. *Nature* 585, 7825 (Sept. 2020), 357–362. https://doi.org/10.1038/s41586-020-2649-2

[21] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.* 9, 8 (Nov. 1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[22] Jingyu Hua, Yue Gao, and Sheng Zhong. 2015. Differentially Private Publication of General Time-Serial Trajectory Data. In *2015 IEEE Conf. Comput. Commun. INFOCOM*, Vol. 26. IEEE, 549–557. https://doi.org/10.1109/INFOCOM.2015.7218422

[23] Kaifeng Jiang, Dongxu Shao, Stéphane Bressan, Thomas Kister, and Kian-Lee Tan. 2013. Publishing Trajectories with Differential Privacy Guarantees. In *Proc. 25th Int. Conf. Sci. Stat. Database Manag. - SSDBM*. ACM Press, 1. https://doi.org/10.1145/2484838.2484846

[24] Weiwei Jiang and Jiayun Luo. 2022. Graph Neural Network for Traffic Forecasting: A Survey. *Expert Systems with Applications* 207 (Nov. 2022), 117921. https://doi.org/10.1016/j.eswa.2022.117921

[25] Diederik P Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *3rd Int. Conf. Learn. Represent. ICLR 2015 San Diego CA USA May 7-9 2015 Conf. Track Proc.* abs1412.69 (Dec. 2014), 1–15. https://doi.org/10.48550/ARXIV.1412.6980

[26] Robert A. Kleinman and Colin Merkel. 2020. Digital Contact Tracing for COVID-19. *Can. Med. Assoc. J.* 192, 24 (June 2020), E653–E656. https://doi.org/10.1503/cmaj.200922

[27] Jae Hyun Lee, Adam W. Davis, Seo Youn Yoon, and Konstadinos G. Goulias. 2016. Activity Space Estimation with Longitudinal Observations of Social Media Data. *Transportation* 43, 6 (Nov. 2016), 955–977. https://doi.org/10.1007/s11116-016-9719-1

[28] Meng Li, Liehuang Zhu, Zijian Zhang, and Rixin Xu. 2017. Achieving Differential Privacy of Trajectory Data Publishing in Participatory Sensing. *Inf. Sci.* 400–401 (Aug. 2017), 1–13. https://doi.org/10.1016/j.ins.2017.03.015

[29] Peng Li, Xi Rao, Jennifer Blase, Yue Zhang, Xu Chu, and Ce Zhang. 2021. CleanML: A Study for Evaluating the Impact of Data Cleaning on ML Classification Tasks. In *2021 IEEE 37th Int. Conf. Data Eng. ICDE*. IEEE, 13–24. https://doi.org/10.1109/ICDE51399.2021.00009

[30] Qi Liu, Juan Yu, Jianmin Han, and Xin Yao. 2021. Differentially Private and Utility-Aware Publication of Trajectory Data. *Expert Syst. Appl.* 180, March 2020 (Oct. 2021), 115120. https://doi.org/10.1016/j.eswa.2021.115120

[31] Xi Liu, Hanzhou Chen, and Clio Andris. 2018. trajGANs: Using Generative Adversarial Networks for Geo-Privacy Protection of Trajectory Data (Vision Paper). In *Locat. Priv. Secur. Workshop*. github.io, 1–7.

[32] Jia Luo and Jinying Huang. 2019. Generative Adversarial Network: An Overview. *Yi Qi Yi Biao Xue BaoChinese J. Sci. Instrum.* 40, 3 (2019), 74–84. https://doi.org/10.19650/j.cnki.cjsi.J1804413

[33] Tinghuai Ma and Fagen Song. 2021. A Trajectory Privacy Protection Method Based on Random Sampling Differential Privacy. *ISPRS Int. J. Geo-Inf.* 10, 7 (July 2021), 454. https://doi.org/10.3390/ijgi10070454

[34] Lucas May Petry, Camila Leite Da Silva, Andrea Esuli, Chiara Renso, and Vania Bogorny. 2020. MARC: A Robust Method for Multiple-Aspect Trajectory Classification via Space, Time, and Semantic Embeddings. *Int. J. Geogr. Inf. Sci.* 34, 7 (2020), 1428–1450. https://doi.org/10.1080/13658816.2019.1707835

[35] Frank McSherry and Kunal Talwar. 2007. Mechanism Design via Differential Privacy. In *48th Annu. IEEE Symp. Found. Comput. Sci. FOCS07*. IEEE, 94–103. https://doi.org/10.1109/FOCS.2007.66

[36] Anna Monreale, Roberto Trasarti, Dino Pedreschi, Chiara Renso, and Vania Bogorny. 2011. C-Safety: A Framework for the Anonymization of Semantic Trajectories. *Trans. Data Priv.* 4, 2 (2011), 73–101.

[37] Elham Naghizade, Lars Kulik, Egemen Tanin, and James Bailey. 2020. Privacy- and Context-aware Release of Trajectory Data. *ACM Trans. Spat. Algorithms Syst.* 6, 1 (Feb. 2020), 1–25. https://doi.org/10.1145/3363449

[38] Mehmet Ercan Nergiz, Maurizio Atzori, and Yucel Saygin. 2008. Towards Trajectory Anonymization: A Generalization-Based Approach. In *Proc. SIGSPATIAL ACM GIS 2008 Int. Workshop Secur. Priv. GIS LBS - SPRINGL 08*. ACM Press, 52. https://doi.org/10.1145/1503402.1503413

[39] Niantic. 2022. Pokemon GO. https://pokemongolive.com/

[40] Ben Niu, Qinghua Li, Xiaoyan Zhu, Guohong Cao, and Hui Li. 2014. Achieving K-Anonymity in Privacy-Aware Location-Based Services. In *IEEE INFOCOM 2014 - IEEE Conf. Comput. Commun.* IEEE, 754–762. https://doi.org/10.1109/INFOCOM.2014.6848002

[41] Xin Niu, Hongyu Huang, and Yantao Li. 2020. A Real-Time Data Collection Mechanism With Trajectory Privacy in Mobile Crowd-Sensing. *IEEE Commun. Lett.* 24, 10 (Oct. 2020), 2114–2118. https://doi.org/10.1109/LCOMM.2020.3003997

[42] The pandas development team. 2022. Pandas-Dev/Pandas: Pandas. https://doi.org/10.5281/zenodo.6408044

[43] Young Joon Park, Young June Choe, Ok Park, Shin Young Park, Young-Man Kim, et al. 2020. Contact Tracing during Coronavirus Disease Outbreak, South Korea, 2020. *Emerg. Infect. Dis.* 26, 10 (Oct. 2020), 2465–2468. https://doi.org/10.3201/eid2610.201315

[44] Fabien Petitcolas. 1883. La Cryptographie Militaire. *J Sci Mil.* 9 (1883), 161–191.

[45] Atul Pokharel, Robert Soulé, and Avi Silberschatz. 2021. A Case for Location Based Contact Tracing. *Health Care Manag. Sci.* 24, 2 (June 2021), 420–438. https://doi.org/10.1007/s10729-021-09567-z

[46] Vincent Primault, Sonia Ben Mokhtar, Cedric Lauradoux, and Lionel Brunie. 2015. Time Distortion Anonymization for the Publication of Mobility Data with High Utility. In *2015 IEEE Trust.*, Vol. 1. IEEE, 539–546. https://doi.org/10.1109/Trustcom.2015.417

[47] Vincent Primault, Antoine Boutet, Sonia Ben Mokhtar, and Lionel Brunie. 2019. The Long Road to Computational Location Privacy: A Survey. *IEEE Commun. Surv. Tutor.* 21, 3 (2019), 2772–2793. https://doi.org/10.1109/COMST.2018.2873950

[48] Youyang Qu, Jingwen Zhang, Ruidong Li, Xiaoning Zhang, Xuemeng Zhai, and Shui Yu. 2020. Generative Adversarial Networks Enhanced Location Privacy in 5G Networks. *Sci. China Inf. Sci.* 63, 12 (Dec. 2020), 220303. https://doi.org/10.1007/s11432-019-2834-x

[49] Jinmeng Rao, Song Gao, Yuhao Kang, and Qunying Huang. 2020. LSTM-TrajGAN: A Deep Learning Approach to Trajectory Privacy Protection. *Leibniz Int. Proc. Inform.* 177, GIScience (2020), 1–16. https://doi.org/10.4230/LIPIcs.GIScience.2021.I.12

[50] C C Robusto. 1957. The Cosine-Haversine Formula. *Am. Math. Mon.* 64, 1 (1957), 38–40. https://doi.org/10.2307/2309088

[51] Sina Shaham, Ming Ding, Bo Liu, Shuping Dang, Zihuai Lin, and Jun Li. 2021. Privacy Preserving Location Data Publishing: A Machine Learning Approach. *IEEE Trans. Knowl. Data Eng.* 33, 9 (Sept. 2021), 3270–3283. https://doi.org/10.1109/TKDE.2020.2964658
[52] Minglai Shao, Jianxin Li, Qiben Yan, Feng Chen, Hongyi Huang, and Xunxun Chen. 2020. Structured Sparsity Model Based Trajectory Tracking Using Private Location Data Release. *IEEE Trans. Dependable Secure Comput.* 18, 6 (2020), 2983–2995. https://doi.org/10.1109/TDSC.2020.2972334
[53] Xiujin Shi, Junrui Zhang, and Yuan Gong. 2021. A Dummy Location Generation Algorithm Based on the Semantic Quantification of Location. In *2021 IEEE Int. Conf. Artif. Intell. Comput. Appl. ICAICA*. IEEE, 172–176. https://doi.org/10.1109/ICAICA52286.2021.9497903
[54] Dalwinder Singh and Birmohan Singh. 2020. Investigating the Impact of Data Normalization on Classification Performance. *Appl. Soft Comput.* 97 (Dec. 2020), 105524. https://doi.org/10.1016/j.asoc.2019.105524
[55] Zhen Tu, Kai Zhao, Fengli Xu, Yong Li, Li Su, and Depeng Jin. 2019. Protecting Trajectory From Semantic Attack Considering K-Anonymity, l-Diversity, and t-Closeness. *IEEE Trans. Netw. Serv. Manag.* 16, 1 (March 2019), 264–278. https://doi.org/10.1109/TNSM.2018.2877790
[56] Waze Mobile. 2022. Waze. https://www.waze.com/
[57] Yu Xin, Zhi-Qiang Xie, and Jing Yang. 2017. The Privacy Preserving Method for Dynamic Trajectory Releasing Based on Adaptive Clustering. *Inf. Sci.* 378 (Feb. 2017), 131–143. https://doi.org/10.1016/j.ins.2016.10.038
[58] Heng Xu, Xin (Robert) Luo, John M. Carroll, and Mary Beth Rosson. 2011. The Personalization Privacy Paradox: An Exploratory Study of Decision Making Process for Location-Aware Marketing. *Decis. Support Syst.* 51, 1 (April 2011), 42–52. https://doi.org/10.1016/j.dss.2010.11.017
[59] Jing Yuan, Yu Zheng, Chengyang Zhang, Wenlei Xie, Xing Xie, Guangzhong Sun, and Yan Huang. 2010. T-Drive. In *Proc. 18th SIGSPATIAL Int. Conf. Adv. Geogr. Inf. Syst. - GIS 10.* ACM Press, 99. https://doi.org/10.1145/1869790.1869807
[60] Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. 2009. Mining Interesting Locations and Travel Sequences from GPS Trajectories. In *Proc. 18th Int. Conf. World Wide Web (WWW '09)*. Association for Computing Machinery, 791–800. https://doi.org/10.1145/1526709.1526816

A GLOSSARY

GAN Generate Adversarial Network.
LSTM Long Short-Term Memory.
MAE Mean Absolute Error.
MLP Multi-Layer Perceptron.
MSE Mean Squared Error.
OP-Distance Distance between *original* and *protected* trajectory.
OR-Distance Distance between *original* and *reconstructed* traj..
POI Point of Interest.
RAoPT Reconstruction Attack on Protected Trajectories.
ReLU Rectified Linear Activation.
RNN Recurrent Neural Network.
SDD Sampling Distance and Direction.
tanh Hyperbolic Tangent.
TUL Trajectory User Linking.

B DIFFERENTIAL PRIVACY

Differential Privacy [13] represents one of the central privacy notions used to protect personal information. Compared to other notions, it is based on strong theoretical guarantees and provides protection even against adversaries with background knowledge. The main intuition of differential privacy is that the input of any single user or row in a dataset does not significantly change the published result. Accordingly, participation does not harm any user's privacy as the output is approximately the same with or without their data. The mathematical definition is as follows [13]:

Definition B.1 (Differential Privacy). A mechanism $\mathcal{K}$ provides ε-differential privacy if for all datasets D_1 and D_2 differing in at most one element, and all $S \subseteq Range(\mathcal{K})$ holds

$$\mathbb{P}[\mathcal{K}(D_1) \in S] \leq e^{\varepsilon} \times \mathbb{P}[\mathcal{K}(D_2) \in S] \tag{1}$$

For example, the mechanism $\mathcal{K}$ might be a function that computes a noisy average over a dataset. Now if the data of another user is added to the dataset D_1 yielding dataset D_2, the change of the probabilities for the outputs of $\mathcal{K}$ is bounded depending on ε. The smaller ε is chosen, the larger is the provided privacy level. In literature, common values for ε range from 0.01 to 10 [16]. The most common way to design a differential private mechanism is the addition of noise to the output, by using the Laplace mechanism [14], Gaussian mechanism [15], or exponential mechanism [35]. While adding noise from a Laplace distribution to location data is a straight-forward way to achieve differential privacy [23, 52], the sensitivity of location information requires that a high level of noise be added to the published trajectories, such that they cannot provide much utility [23]. Therefore, multiple differential private mechanisms specifically targeting trajectories have been designed [5, 6, 22, 23, 28, 30]. One example is the SDD mechanism which we describe in Section 2.2.

C JACCARD INDEX

In addition to the Euclidean and Hausdorff distances described in Section 5.4, we also measured the Jaccard index of the trajectories convex hulls for each measurement. The convex hull of a trajectory can be used to represent the activity space of a user [27]. Hence, the Jaccard index of two trajectories' convex hull, which is computed by dividing the intersection through the union of two areas, indicates how close the activity spaces are. A Jaccard index of 1 means that the activity spaces are identical, while 0 implies that the activity spaces do not intersect. The Jaccard index has not only been used before to measure trajectory closeness [49], but is particularly suitable to indicate the threat posed by reconstruction. A large Jaccard index for a reconstructed trajectory suggests that an attacker, e.g., a stalker, will find the victim within the activity space of the reconstructed trajectory. The index is better suited than the intersection itself, as it penalises very large activity spaces which include the original trajectory, e.g., a protected trajectory that spans over a large area due to the high noise. Such a large area containing the original trajectory is not helpful for an adversary, as the adversary would only learn that the victim is somewhere within the large area. The smaller the convex hull of the reconstructed trajectory is, the less area needs to be considered/searched by the adversary. Table 5 states the mean Jaccard index before and after reconstruction for all our evaluation cases.

D COMPUTATION OF PERCENTAGE REDUCTION

The percentage reduction stated for our evaluation measurements is computed according to the following formula:

$$\frac{(OP - OR)}{|OP|} * 100$$

Thereby, *OP* refers to the OP-Distance, and *OR* to the OR-Distance. The reduction is computed for each tuple of original, protected, and reconstructed trajectory independently. Then, the average is computed over all the individual reductions. If the percent increase

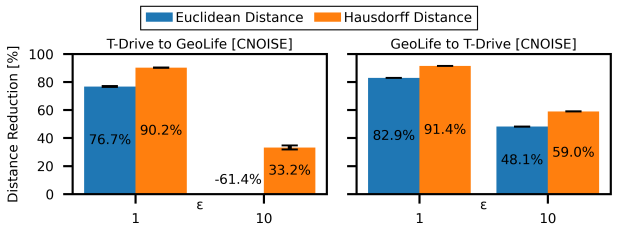

Figure 7: Adversary 2: The figure shows the results for the transfer from one dataset to another. All test cases use the CNoise mechanism as protection with different choices for ε.

of the Jaccard index is mentioned in the paper, it was computed by directly comparing the mean Jaccard index before and after reconstruction. Computing the increase for each sample individually is not directly feasible because the Jaccard index before reconstruction might be 0.

E FURTHER RESULTS

This section contains supplemental results for measurements not included in the main part of the paper. First, we describe the results for adversary 2 (cf. Section 2.4) and the CNoise mechanism in Section E.1. Second, we discuss our runtime measurement in Section E.2

E.1 Dataset Transfer with CNoise Protection.

Figure 7 is the corresponding figure to Figure 5, but with the CNoise mechanism used for protection instead of the SDD mechanism. As for the case with the SDD mechanism, the results indicate that training on a different dataset lowers the reconstruction success while maintaining the general trends. The plot also confirms that the transfer from the GeoLife dataset to the T-Drive dataset is more effective than vice-versa. Interestingly the reconstruction from CNOISE ($\varepsilon = 10$) works with some reduction success when transferring from the GeoLife to the T-Drive dataset, but not at all in the other direction. We do not know the cause of this result.

E.2 Reconstruction Runtime

We performed our performance measurements on a single server (2x Intel Xeon Silver 4208 and 128 GB RAM) running Ubuntu 20.04.01 LTS. The server contains 4 NVIDIA Tesla T4 GPUs with 16 GB RAM each, but we only used a single GPU for the experiments. We measured the time required for reconstruction of a single trajectory, including encoding of the protected trajectory and decoding of the reconstructed one, with a pre-trained model. As the adversary can train the model off-line with an available dataset before the attack, the performance of training is less important than the reconstruction itself which might be performed on-line to directly execute the attack. On the specified hardware, the runtime for the reconstruction of a single GeoLife trajectory protected with SDD ($\varepsilon = 0.1$) lies within the 99 % confidence interval [51.3, 52.1] ms. For the corresponding T-Drive trajectories (cf. Case 7), which are generally shorter, the 99 % confidence interval is [44.8, 45.6] ms. This short reconstruction time underlines the real-world risk of the presented reconstruction attack.

F ALL EVALUATION RESULTS

Table 5 displays all performed measurements. The table specifies which dataset, protection mechanism (abbreviated with *Mech.*) and ε value have been used for the training and test set. The table contains the percentage reduction of the Euclidean and Hausdorff distance after the reconstruction, which is computed as described in Appendix D. Moreover, the table states the mean Jaccard index of the original and protected trajectories (*Jaccard B.*) and of the original and reconstructed trajectories (*Jaccard A.*). A larger value for the Jaccard indicates a higher threat as discussed in Appendix C, and hence indicates the success of our attack. We do not state the percentage increase for the Jaccard Index as the percentages fluctuate strongly and show very large values due to the small Jaccard indices before reconstruction.

ID	Dataset Train	Dataset Test	Mech. Train	Mech. Test	ε Train	ε Test	Euclidean	Hausdorff	Jaccard B.	Jaccard A.
1	T-Drive	T-Drive	CNoise	CNoise	0.01	0.01	99.7 %	99.8 %	1.19e − 7	5.12e − 2
2	T-Drive	T-Drive	CNoise	CNoise	0.1	0.1	98.1 %	99.1 %	1.18e − 5	3.44e − 3
3	T-Drive	T-Drive	CNoise	CNoise	1.0	1.0	87.4 %	93.4 %	1.17e − 3	3.67e − 2
4	T-Drive	T-Drive	CNoise	CNoise	10.0	10.0	65.1 %	73.6 %	7.69e − 2	2.66e − 1
5	T-Drive	T-Drive	CNoise	CNoise	100.0	100.0	29.8 %	29.8 %	5.61e − 1	6.23e − 1
6	T-Drive	T-Drive	SDD	SDD	0.01	0.01	68.1 %	72.7 %	2.46e − 2	7.09e − 2
7	T-Drive	T-Drive	SDD	SDD	0.1	0.1	68.2 %	72.8 %	2.46e − 2	7.03e − 2
8	T-Drive	T-Drive	SDD	SDD	1.0	1.0	68.4 %	73.1 %	2.45e − 2	7.13e − 2
9	T-Drive	T-Drive	SDD	SDD	10.0	10.0	71.7 %	77.2 %	2.22e − 2	8.86e − 2
10	GeoLife	GeoLife	CNoise	CNoise	0.01	0.01	99.4 %	99.2 %	2.01e − 10	1.72e − 5
11	GeoLife	GeoLife	CNoise	CNoise	0.1	0.1	98.2 %	99.1 %	1.96e − 8	8.08e − 4
12	GeoLife	GeoLife	CNoise	CNoise	1.0	1.0	91.3 %	95.3 %	1.98e − 6	1.53e − 3
13	GeoLife	GeoLife	CNoise	CNoise	10.0	10.0	77.7 %	82.2 %	1.88e − 4	9.42e − 3
14	GeoLife	GeoLife	CNoise	CNoise	100.0	100.0	56.5 %	66.1 %	9.13e − 3	6.78e − 2
15	GeoLife	GeoLife	SDD	SDD	0.01	0.01	82.3 %	87.0 %	3.65e − 5	2.55e − 3
16	GeoLife	GeoLife	SDD	SDD	0.1	0.1	83.7 %	87.7 %	3.63e − 5	2.55e − 3
17	GeoLife	GeoLife	SDD	SDD	1.0	1.0	83.6 %	87.7 %	3.60e − 5	2.56e − 3
18	GeoLife	GeoLife	SDD	SDD	10.0	10.0	80.2 %	86.6 %	1.70e − 5	8.86e − 4
19	T-Drive	GeoLife	CNoise	CNoise	1.0	1.0	76.7 %	90.2 %	1.98e − 6	6.62e − 4
20	T-Drive	GeoLife	CNoise	CNoise	10.0	10.0	−61.4 %	33.2 %	1.88e − 4	5.83e − 3
21	T-Drive	GeoLife	SDD	SDD	0.1	0.1	40.1 %	62.0 %	3.63e − 5	8.01e − 4
22	T-Drive	GeoLife	SDD	SDD	1.0	1.0	38.2 %	61.3 %	3.60e − 5	9.23e − 4
23	GeoLife	T-Drive	CNoise	CNoise	1.0	1.0	82.9 %	91.4 %	1.17e − 3	1.66e − 2
24	GeoLife	T-Drive	CNoise	CNoise	10.0	10.0	48.1 %	59.0 %	7.69e − 2	1.23e − 1
25	GeoLife	T-Drive	SDD	SDD	0.1	0.1	61.0 %	66.6 %	2.46e − 2	3.18e − 2
26	GeoLife	T-Drive	SDD	SDD	1.0	1.0	61.0 %	66.8 %	2.45e − 2	3.19e − 2
27	T-Drive	T-Drive	CNoise	CNoise	1.0	10.0	24.3 %	46.2 %	7.69e − 2	5.45e − 2
28	T-Drive	T-Drive	CNoise	CNoise	10.0	1.0	72.5 %	79.3 %	1.17e − 3	2.72e − 2
29	T-Drive	T-Drive	SDD	SDD	0.1	1.0	68.4 %	73.1 %	2.45e − 2	7.21e − 2
30	T-Drive	T-Drive	SDD	SDD	1.0	0.1	68.3 %	72.8 %	2.46e − 2	7.10e − 2
31	T-Drive	T-Drive	CNoise	SDD	1.0	1.0	27.7 %	44.9 %	2.45e − 2	1.16e − 2
32	T-Drive	T-Drive	SDD	CNoise	1.0	1.0	53.0 %	70.3 %	1.17e − 3	1.23e − 2
33	T-Drive	GeoLife	CNoise	SDD	1.0	0.1	−13.8 %	22.5 %	3.63e − 5	3.92e − 5
34	GeoLife	T-Drive	SDD	CNoise	0.1	1.0	49.5 %	69.0 %	1.17e − 3	1.12e − 2
35	T-Drive	GeoLife	SDD	CNoise	1.0	0.1	66.0 %	68.2 %	1.96e − 8	2.86e − 7
36	GeoLife	T-Drive	CNoise	SDD	0.1	1.0	22.9 %	42.1 %	2.45e − 2	9.41e − 3

Table 5: All evaluation cases. This table displays all performed measurements along the percentage reduction of Euclidean and Hausdorff distance through the reconstruction, and the mean Jaccard index [B]efore and [A]fter reconstruction.