GeoSClean: Secure Cleaning of GPS Trajectory Data using Anomaly Detection

Vikram Patil, Priyanka Singh, Shivam Parikh and Pradeep K. Atrey
Albany Lab for Privacy and Security, College of Engineering and Applied Sciences
University at Albany, State University of New York
Albany, NY, USA
Email: {vpatil, psingh9, sparikh, patrey}@albany.edu

Abstract—Today cloud-based GPS enabled services or Location Based Services (LBS) are used more than ever because of a burgeoning number of smartphones and IoT devices and their uninterrupted connectivity to cloud. However, a number of hacking attacks on cloud raise serious security and privacy concerns among users; due to which many users do not like to share their location information. This poses a challenging problem of availing LBS from the cloud without revealing users location. Also, often GPS receivers record incorrect location data, which can affect the accuracy of LBS. In this paper, we propose a method, called GeoSClean, that not only cleans the GPS trajectory data using a novel anomaly detection scheme but also keeps users location confidential. Anomaly points are detected considering the combination of properties of the GPS trajectory data as distance, velocity, and acceleration. The experimental results validate the utility of the proposed method.

Keywords—Secure data cleaning; GPS trajectory; Anomaly detection;

I. INTRODUCTION

Applications such as driverless/ autonomous cars, cell phones, and IoT devices continuously send streams of the GPS trajectory data to the service provider for real-time analysis and Location Based Services (LBS). LBS includes apps and services like Uber, maps, travel mode detection, traffic congestion detection, fitness trackers, etc. GPS receivers sometimes do not record signals transmitted by satellites accurately due to interference, low signal, sensor malfunction, etc. So the recorded GPS data point deviates from the actual location of the user. This difference in actual and recorded GPS position may result in an error for LBS and affect the utility of these services. In many cases, this error is just a few meters but sometimes it can be huge. In order to eliminate it, LBS providers use some pre-processing techniques to clean or pre-process the data, so that it could be of enhanced utility. Hence, detecting outlier in such geospatial datasets need immediate attention.

Many traditional schemes have been proposed in the literature towards fulfilling this objective of outlier detection. Liu et al. [1] provides summary of them. They can be broadly categorized into four main types: similarity-based, probability distribution-based, entropy-based, and rule-based approaches. In similarity-based approaches, dissimilarity measures are combined with the traditional outlier detection approaches to find the outlier points whereas the data is analyzed via fitting into probability-based distributions in probability distribution-based approaches [2]. Entropy-based methods tend to categorize points as outliers if after removing such points the entropy of the remaining points is minimized while rule-based methods mine rules based on data set at hand and detect anomalies as those points which do not follow them [3]. These approaches are general in nature and are not designed for spatial datasets since they do not consider spatial attributes of GPS data.

Outlier detection has been recently very active for the spatial databases [4]. Basically, the approaches could be categorized as visualization-based, graph-based and statistic-based. Outlying objects are highlighted in visualization-based such as scatterplot [5] and Moran scatterplot [6]. Declaring outlier based on a function where differences are computed for specific observation with respect to its neighbors are employed in graph-based outliers whereas in statistics-based approaches exploit the local inconsistencies to determine outliers such as GLS-SOD, Z [7], median-based Z [8] and iterative-Z [8] based approaches.

Anomaly detection techniques such as point anomaly detection (hypothesis testing/normal distribution), group anomaly detection (log likelihood ratio (LLR) etc.) have been widely used in many fields such as credit card fraud detection, disease spread, event detection, intrusion detection systems, pattern recognition etc. Zheng [9] provides the summary of various data pre-processing techniques. In Mean/Median filter techniques, a predefined window of points is considered and the average of all the points in the window is considered in order to eliminate the erroneous point [9]. In probability distribution based filters such as Kalman and Particle Filters [10], the position of the erroneous point is estimated and replaced by the approximated point. In Heuristics based outlier detection methods, the focus is on eliminating the outlier points from the trajectory instead of approximating them. They determine a threshold by using heuristics for a property like the distance between two successive points etc. Points which exceed the threshold are eliminated accordingly. This provides better accuracy, but finding threshold is still left to heuristics.

Existing methods expose the user’s location to the CSP while processing the data. Hence, the confidentiality and privacy of the data may be compromised. Recent hacking
incidences like Equifax are raising privacy and security concerns. Also, determining the threshold in outlier detection based methods is left to users. Most of the existing methods concentrate on a single property of trajectory such as distance to determine outlier points. However, many spatial properties like velocity, acceleration, synchronized Euclidian distance (SED) [11] can also be considered. [12], [13] describe techniques for secured processing over the cloud.

In this paper, we propose a Z-test (hypothesis testing/normal distribution) based secure point anomaly detection method using the combination of distance, velocity, and acceleration for secured GPS trajectory data preprocessing. To the best of our knowledge, this is the first attempt to provide anomaly detection without revealing users location. Modified haversine formula (Eq. 2) is used to perform secured GPS trajectory data preprocessing. We have extended our previous work [14] for proposing this secure method for anomaly detection. In [14], we proposed the method to securely outsource GPS data to LBS for providing services without revealing users actual location. This is achieved using differences between latitude, longitude and time of successive points and using the modified version of the original haversine formula [15], [14]. Our major contributions in this work can be summarized as follows: (1) We provide a method to clean the GPS trajectory data without sharing actual GPS coordinates. (2) We use the combination of GPS data properties such as distance, velocity, and acceleration for trajectory pre-processing.

The rest of the paper is organized as follows: Section II discusses the proposed method. Section III gives detailed implementation and results. In section IV, we analyze the security of the proposed method. Finally, conclusions along with future work are discussed in section V.

II. PROPOSED METHOD

The workflow of the proposed method is described in Figure 1. It uses point anomaly instead of group anomaly detection technique as the error points are mostly single points instead of the group.

The definitions applicable to the proposed scheme and detailed steps are as follows:

A. Definitions

Definition 1: (Trajectory Pre-Processing) A trajectory $T$ can be said to be pre-processed if it can be represented by a transformed trajectory $T'$ such that $T' = f(T), T' \subseteq T$ and $|T'| \leq |T|$, where $f$ is a pre-processing function and $|\cdot|$ represents the cardinality of the set.

Definition 2: (Anomalous trajectory point) For a GPS trajectory $T$, if $D$, $V$, $A$ are sets of all the points of anomalous distance, velocity, acceleration respectively then a point is said to be anomalous if it belongs to at least two anomalous point sets i.e. if then a given point $p$ can be called as anomalous trajectory point, if it belongs to at least two of three anomalous sets.

$O = \{ p \mid p \in (D \cap V) \cup (V \cap A) \cup (D \cap A) \}; D, V, A \subseteq T$  

This can also be explained with Venn diagram in Figure 2.

B. Algorithm

The algorithm is described in detail as follows: The proposed method accepts Trajectory $T = \{ P_1, \ldots, P_n \}$ where every point consists of latitude, longitude and time attributes, as input for preprocessing.

Step 1 Store the first trajectory point as Key $K = P_1$.

Step 2 Multiple latitude and longitude of every point by $10^6$, and calculate the difference between latitude and longitudes of successive points and store them as differences.

$D = [P_{(i)lat} - P_{(i-1)lat}, P_{(i)long} - P_{(i-1)long}, P_{(i)time} - P_{(i-1)time}]$ \n
$\forall i = 2, \ldots, n.$ (1)

Since latitudes and longitudes generally have six digits after decimal point, after multiplication by $10^6$, their differences will result in real numbers. This is also referred as fixed point arithmetic [16].

Step 3 Store key $K$ on the user’s device and send differences $D$ to the CSP.

Step 4 At CSP side, use modified haversine formula [14] to calculate the distance between points using their differences, velocity and acceleration.

For any two given points $P_1(\text{lat}_1, \text{long}_1), P_2(\text{lat}_2, \text{long}_2)$,
$P_2(lat_2, long_2)$ and if $R$ is the radius of the earth (mean radius = 6,371 km), the distance $d$ can be calculated using modified Haversine formula as follows:

$$a = (\sin\left(\frac{lat_2 - lat_1}{2}\right))^2 + (\sin\left(\frac{long_2 - long_1}{2}\right))^2$$

$$c = 2 \times \arctan2(\sqrt{a}, \sqrt{1 - a})$$

$$d = R \times c$$

(2)

The cosine term in original haversine formula[15] is approximated to 1, to calculate distance between two very close points. Although modified Haversine formula can introduce minor errors in some cases where cosine of latitudes is not closer to 1, it can be further improved by choosing a closer value of the cosine of latitudes of close points. Although modified Haversine formula can be approximated to 1, to calculate distance between two very close points, the differences to actual trajectory since the starting point is minimal. So we can say that the difference between actual and calculated data is approx. 13,432 GPS data points. We first follow the procedure explained in [14], to calculate the differences. We used haversine formula [15] and actual data using differences and cosine of latitudes to calculate the actual distance, velocity, acceleration between two successive points of the trajectory. Then we used approximated haversine formula Eq.2, to determine calculated distance, velocity, and acceleration just from the differential data.

In both cases, we use first 5000 points as training data set to calculate mean ($\mu$) and standard deviation ($\sigma$) of distance, velocity, and acceleration, rest of the points as test data. Table I shows the comparison of the mean ($\mu$) and standard deviation ($\sigma$) of distance, velocity, and acceleration for the actual data using original haversine and calculated data using modified haversine formula.

Figure 3 shows the number of anomalous GPS data points which fall in $[1.645, \infty]$ region for distance, velocity, acceleration, and trajectory anomalous points as described in Definition 2, Figure 2. From both the figures, we can say that the difference between actual and calculated data is minimal. So we can say that the proposed method can be successfully used for GPS trajectory preprocessing in a secured way without revealing users location.

IV. SECURITY ANALYSIS

For the security analysis, we consider users device is secure, the user is honest, CSP is semi malicious and/or curious, any external entity can be considered as an adversary. In the event of adversary getting access to the data stored in the cloud, the adversary will not be able to decode the differences to actual trajectory since the starting point is never stored on the CSP. In case of Linkage attack, the
adversary will not be able to relate the just the differences between GPS points with any other information on the social media. So this model will also be effective in that area.

Remark 1: The proposed method can effectively preprocess the GPS trajectory.

According to Definition 2, the cardinality of set $O$, $|O| = 76$. Let's consider the test data as trajectory $T$ with 8431 GPS data points. So according to Definition 2, the cleaned trajectory $T' = T - O$ and $|T'| = |T| - |O| = 8355$. So we can say that, according to Definition 1, the transformed trajectory is formed after eliminating anomalous points and proposed method can effectively be used for preprocessing the GPS trajectory.

Remark 2: Proposed method can securely detect anomalies on calculated data with actual data.

By observing results in Table I, we can say that using modified haversine formula, we can achieve similar results in calculated data (80 anomalous points) as the actual data (76 anomalous points), as mentioned in Definition 2. Hence, we can say that the proposed method can be used to securely preprocess GPS trajectory with just differences between successive GPS points, without revealing users location.

V. CONCLUSION AND FUTURE WORK

In this paper, we have demonstrated that GPS trajectory data cleaning can be done at the CSP side without revealing users actual location. A novel criterion for classifying a trajectory point as an anomalous point has been proposed in the paper by considering combinations of GPS data properties such as distance, velocity, and acceleration. The hypothesis testing based anomaly detection method has been validated to detect anomalous points with high confidence. We are using hypothesis testing for the representational purpose, but any other anomaly detection techniques can also be used in a secured way using the proposed method. In future, this work can be expanded to process streams of GPS trajectory data to provide services in real time by applying techniques from deep learning and pattern recognition.

REFERENCES


