DSPPTD: Dynamic Scheme for Privacy Protection of Trajectory Data in LBS

Ajay K. Gupta and Sanjay Kumar

1 Introduction

Location-aware service [1–3] is a type of context-aware services in which location is provided as input to the system. The system takes location as input and provides services to the user. The location may be geometric, i.e., latitude and longitude form, or it may be semantic, i.e., near and within. User querying the services provides his or her location to service providers believing that the correct location would improve the quality of services (QoS). However, it led to a risk of disclosure of private and confidential information [4]. It is highly challenging to design efficient trade-off between the QoS and privacy of the mobile user. In location-based services, the user provides his current location to the third party for a service request. An attacker (or untrusted service provider) may make an inference attack [5] through these live locations of the user may infer the personal confidential information regarding his health or lifestyle by observing location, duration of stay, and habits of activity performed by him. So, this is a security and privacy problem. The aim here is to reduce the privacy leakage risk as well as to provide the quality of service.

The general architecture of the cellular mobile environment [15, 16] consists of mobile units (MU), fixed hosts (FHs), and base stations (BSs). The BS has its fixed location, functions with two-way radio, and has some data processing capabilities. The basic function of data and transaction management is done by the database server (DBS). Many BSs and FHs are linked via a high-speed network. Each cell has a limited radio coverage area and a BS to manage mobile clients. The cell

A. K. Gupta (✉)
Indian Institute of Information Technology, Pune, India

S. Kumar
United Services Automobile Association (USAA) – AADC Project Technical Lead (HCL America Inc., America), San Antonio, TX, USA

S. Pandey et al. (eds.), *Role of Data-Intensive Distributed Computing Systems in Designing Data Solutions*, EAI/Springer Innovations in Communication and Computing, https://doi.org/10.1007/978-3-031-15542-0_4

can be seen as a limited bandwidth radio coverage area and normally represented by the shape of a hexagon. In wireless local area networks, it can be treated as a high bandwidth network within the area of the building. The wireless channel is splitted in two channels known as uplink and downlink channels. Here, the first one is utilized for the submission of the mobile client's queries, and the second is used to answer the mobile client queries by the mobile switching stations (MSSs). The base station controller (BSC) is used to control the various BSs. The mobile switching center (MSC) gives commands to BSC to control an appropriate BS. Unlimited mobility in personal communications service, global system for mobile communication, and reachability to any BS or FH facilitate many services being easy to deploy in the real world. The public switched telephone network and MSC connects the databases available for a mobile environment to the outer world.

The mobile transactions run in the frequent disconnection mode. Due to mobility and frequent disconnection behavior of these mobile transactions, they are long-lived. The data and/or user may also move in a mobile environment. Therefore, the mobile transaction may have their associated sub-transactions (cohorts). Among those, some may run on the MSS and some may run on mobile nodes. Due to the disconnection and mobility nature of the transaction, it shares its information of states and also partial results with other transactions. Also, the mobile transaction should fulfill some prerequisites to work well in the environment of mobility. With the mobility nature of nodes, the state of the data object being accessed and the corresponding location information must also move. There should be the availability of the techniques to deal with concurrency, frequent disconnection, and consistency between replicated data objects residing at different locations. The mobile transactions are also executed in a distributed manner, which may be subjected to further restrictions such as limited bandwidth. Evolving commit protocols [17] for a distributed transaction in the presence of mobility is the most challenging task in comparison with the generic environment. Here, the mobile transaction may need to deal with the forced wait or forced abort, if wireless channels (uplink or downlink) are not available at any instant of time, and this could be delayed due to hand off randomly. The mobile transactions might not be in a position to complete its implementation due to the unavailability of full database management system (DBMS) capability [18]. This is the reason why conventional transaction control strategies are not well suited to the mobile environment. If the connection is not possible to mobile nodes or due to high expenses in continuous connection, the mobile host can decide to work in disconnected mode also. Based on the locations of initiation and execution of the mobile transaction, it can be classified into three types. The first category is of those mobile transactions, which are both initiated and executed by a mobile host (MH). The second category is of those mobile transactions [19], which are initiated by fixed host (FH), but executed by the MH. The third category is of those mobile transactions, which are initiated by MH, but executed by both MH and FH. In the mobile transaction environment, where MH initiates transactions but executed completely by FH, the MH requires no record retrieval capability.

The location-dependent information system (LDIS) is an application of context-aware computing in the mobile environment where the transaction is initiated and/or executed by MH. Moflex is an example of this type of scenario. This model primarily stood on dependencies set, suitable goals, and also on rules. Sub-transactions for location-based services are also supported by this model. A further version of the same scenario is pre-serialization, which permits the cohorts of the global transactions to commit independently. The serialization technique permits in releasing the nearby resources in a well time-stamped way. There is need of integrating revised data visualization and indexing technique of data item in LDIS for user-friendly response and faster access rate, respectively. A number of mobile devices together with the personal digital assistants has very small screens. Therefore, the requirement for potential future research work is to consider the mobile system screen size and computational limits when developing lightweight simulation methods for desktop and handheld apps applications. Indexing in LBS is used to get faster search results. Before one can search through the LBS, he has to create a search engine index. It facilitates power saving mode to the client until queried records arrive on the requested channel. Index overhead induced by an LBS implementation certainly affects indexing approach selection. The proportional frequency of queries vs. updates especially favors either query-optimized or update-optimized indexing approaches. The scope of future research is toward an investigation of trajectory and filtering approaches to further enhance the efficiency of these indexing approaches in terms of updating and querying.

1.1 Problem Statement

The past trajectory privacy protection approaches mostly rely on obfuscation of the trajectory locations and add more uncertainty to preserve privacy. However, it is challenging to monitor the trade-off between the efficacy of trajectory privacy security and the usefulness for spatial and temporal behavior, and this problem has not been thoroughly explored or measured in past strategies [6, 10]. The recent analyses concentrate predominantly on the spatial component of trajectory details, whereas other semantics such as thematic and temporal attributes are seldom addressed. In comparison, existing methods depend extensively on manually crafted procedures. If the process is revealed, the initial trajectory details can be recovered. To this end, this study intends to investigate the feasibility of deep learning methods to overcome the above mentioned privacy security challenges in trajectory.

The following points can describe the primary contributions of this work.

1. The edge-based distance measure has been introduced in proposed DSPPTD for k-path trajectory clustering of deep neural network processed trajectory to achieve differential privacy before publishing it. The work discusses an end-to-end solution of deep learning to produce trajectory data supporting differential

privacy. A Gaussian mechanism for synthetic trajectory preparation has been described in this work.
2. The two functions, namely mutual information and Hausdorff distance, are used to measure the intensity of privacy protection and utility of the trajectory data with training deep learning approach.
3. Analysis of the trade-off between privacy protection effectiveness and the usefulness of the new model are made utilizing real-world LBS details.

The rest of this paper is organized as follows: Sect. 2 gives an overview of related work. The deep-learning-based differential privacy protection approach has been described in Sect. 3. We discuss the factors affecting privacy protection effectiveness to verify the utility and privacy trade-off of the proposed policy in Sect. 4. Finally, Sect. 5 concludes this paper.

2 Related Work

With the advancement in mobile technologies, smartphones allow peoples to access numerous LBSs and provide interactive information depending upon location of the user. The study of user' positions and associated confidential information not only enables more sophisticated and reliable user information to be created but also inevitably leads to security and privacy problems. Therefore, this domain needs more research works for the development of location-based technologies to resolve such burning issues [11–13].

There are various reports on the privacy security of dummy-based trajectories. Kido et al. [14] were the first who used the concept of a random move to create dummies. Lu et al. [15] suggested a confidentiality-conscious, dummy-based strategy for preserving consumer data. However, the history details were overlooked by these systems. Niu et al. [16] established a Dummy-T effective privacy security system for the route. It employs the minimum cloaking area and context details to ensure each dummy produced on the trajectories is just like the real one. However, it lacks the actual mobility trend and spatiotemporal association, which leads to the deterioration of the degree of privacy.

The definition of k-anonymity was first introduced for relational databases [17]. If the position of the recipient is indistinguishable from the position of certain k-1 persons, then the query is said to be location k-anonymous. Zhang et al. [18] also suggested caching and spatial K-anonymity (CSKA) policy to improve safety through k-anonymity and caching. This system, though, is not well suited to protection for trajectories. Moreover, past policies are based on user-clustered or centralized architectures. Hence, the workload of the network is high, and the anonymizer could lead the bottleneck performance.

To make sure the optimal distribution of the selected dummy locations, the authors in [19] also provided an enhanced decay lengths (DLs) approach that could expand the cloaking region while retaining a degree of privacy near to the

DLs algorithm. In [15], two approaches of dummy creation were suggested by the authors, notably grid-based and circle-based methods, which take into account the privacy criteria. In [8], the authors developed dual dummy-based techniques to guarantee the k-anonymity of privacy-conscious clients in LBS, recognizing that opponents would exploit side details. The previous approaches have also not understood the information on the side that attackers may exploit while picking dummy locations. While some approaches have taken the side information into account, they have a high processing cost. However, the effective selection of dummy locations in IoT remains a research problem. In K-anonymity systems, Hu et al. [5] applied a credit-incentive framework to maximize the efficiency of selecting dummy roles. Based on the fuzzy reasoning, credit rating contributes to a certain maximum level of probability for each customer. A client can still get help from specific users on the condition that his credit rating passes a certain likelihood threshold amount. It motivates people to assist others in building K-anonymity actively. In a sense, all the above solutions originate directly from the single time LBS position privacy policy [20] and therefore ignore the following two issues:

(a) Protection of communication messages in user's LBS request.
(b) Exposure of the users' real location details due to the continuous importance of query position.

Present findings on the evolution of privacy protection concentrate primarily on two sources of study. One is the hierarchical solution to privacy to combine and mix trajectories from various users such that the detection of person trajectory data is turned into an issue of k-anonymity [19, 21]. Here, the spatial cloaking method utilizes k-anonymous cloaked spatial regions to combine trajectory locations between k-objects and renders these trajectories k-anonymized [22]. The mix-zone strategy often anonymizes trajectory locations in a mix-zone using aliases. It removes the link between the former section and the latter section of the mix-zone trajectory [23].

Additionally, the positions of k trajectories are divided into k-anonymized separate regions first by the generalization-based method and then uniform selection and reassemble k new trajectories by connecting points of each k-anonymized region [24]. A further analysis medium is termed geo-masking, which blurs the positions of actual trajectory details by using spatial dimension interference to cover or change the original positions. However, spatial trends might not be substantially affected [25, 26]; for example, Zandbergen [27] discussed the need to preserve privacy and the spatial usefulness of many forms of geo-masks.

Kwan et al. [28] tested the efficacy of three independent arbitrary geo-masks of perturbation on lung cancer cases in space research. Seidl et al. [29] introduced grid masking and random disruption to data sets from GPS and measured the efficiency of privacy security. Gao et al. [26] studied the efficacy of Twitter data aggregation, Gaussian disruption, random disruption, exploration of the complexity, degree of anonymity, and analytics of each process.

Users may access preference details of the actual position in the implemented system without revealing their location data to the service provider. Beresford et al. [30] proposed anonymous communication techniques, who are first to introduce mixed zones concept. A mixing zone applies to a geographic area where no call back activity has been recorded by any users. The researchers in [31] allow users to swap the pseudonyms if they met in mix zone and also care for user to avoid the use of pseudonym for a larger time. The association of app positions and pseudonyms may, therefore, be disrupted by pseudonyms exchange.

Finally, it may be claimed that these days scientists are energetically researching the privacy concern of query processing [5, 32]. A few worthy survey articles have emerged in recent years addressing privacy problems in LBS—difficulties and probabilistic scope connected with it [7, 33].

3 Our Proposed Scheme

We follow the system approach based on fog computation, as seen in Fig. 1. It is made up of three entities: handheld device, LBS server, and fog server. The fog system is operated by the consumer and installed with enough hard space in the user's spare devices. In the proposed approach, the fog server receives the background information. It applies the DSPPTD policy for protecting trajectory and dependent confidential information from the attacker while providing maximum QoS for the user's query request by the LBS server. The LBS server scans the POIs of users, and it returns the output of the applicant to the fog server after this fog server delivers the relevant results to the customer.

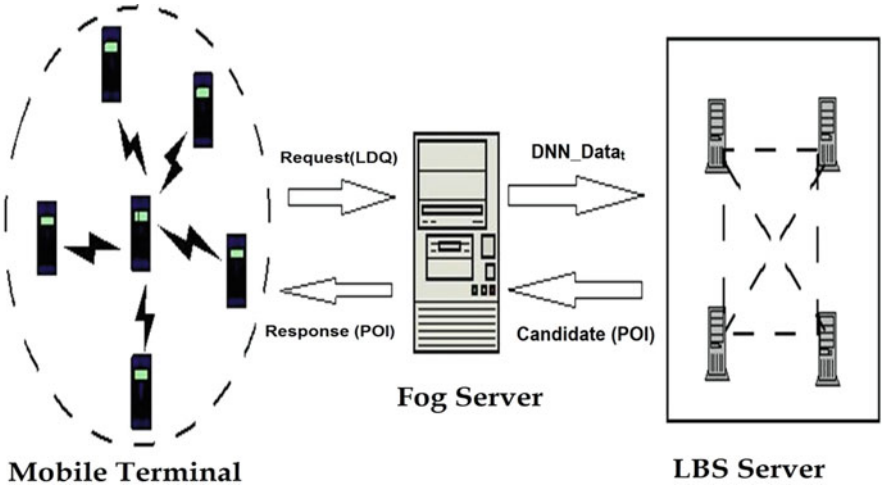

Fig. 1 LBS system structure

Table 1 Key notation used in proposed scheme

Terms	Description	Terms	Description
T	Trajectory	Theta	Skewness parameter based on zipf access distribution
p_j^t	Location point at time t of set j	E	Differential privacy
Data_t	Trajectory original data	DNN_Data_t	DNN processed trajectory data
K	Number of clusters	K	Anonymity degree
E(S)	Entropy of set S	C_{maxH}	Optimized set using entropy
N	System defined variable $N > 2$ k	n	The total number of snapshots
d_r^t	A real location from trajectory at time t	q_j^{t-1}	A query probability for j dummy location at time $t-1$
$TR\left(d_J^t \mid d_j^{t-1}\right)$	Transition probability from time $t-1$ to time t for j dummy locations, where $2>j\geq k$-1	HCR_i^t	Path entropy
M	Randomly selected m locations set from N dummy locations, where $m \leq$ C(2 k-1, N)	m'	Randomly selected m' locations set from 2 k optimized set
D^t	Anonymous set at time t	$q\left(d_j^{t-1}\right)$	Time-dependent query probability at location d_j^{t-1}
$\text{Dist}_{\text{max}}^t$	Separation length from the current position to the next	$\angle(x, y)$	Separation angle between x and y
MI	Mutual information	HD	Hausdorff distance

The concept of cloud fog computing makes server computation resources available in the ground nearer to end-users. In comparison with clustered data centers, these nodes are physically much closer to smartphones, which leads to fast communications between entities. It has the remarkable ability of edge nodes to process and measure large amounts of data under their own, without submitting it to distant servers. Fog computing is an intermediary between external servers and mobile devices. It controls the details that the server can obtain, which can be accessed locally. For this sense, fog is a smart portal that offloads clouds making for more effective data collection, retrieval, and analysis. Table 1 summarizes the notations used in the proposed scheme.

The DSPPTD approach is a trajectory privacy protection that incorporates the deep neural network and structure of the Gaussian system to build privacy-preserving synthetic trajectories as substitutes to actual trajectories for the exchange and publishing of trajectories.

In this paper, we propose a new approach consisting of four main components. The four main components that are implemented by the system include processing, generation, optimization, and release of trajectories. A detailed summary of each unit is given below.

A. Trajectory processing model uses the user's moving scene to generate corresponding high-dimensional data items.
B. Trajectory generator uses a deep neural network, which takes random noise and original location points of trajectories as inputs to generate synthetic trajectories as outputs. The processed trajectory consists of position points in actual timestamps that can shield the original collection of data.
C. Apply k-means clustering for k subregions division of the location trajectory data region with common data points.
D. The trajectory release step involves comparing each clustered "synthetic trajectories datum" to corresponding "real" trajectory and merging accordingly. The process also involves a prejudging mechanism to ensure at least one actual trajectory record can be seen in processed trajectory.

3.1 Trajectory Processing Model

The trajectory is a sequenced series of user movement points where the interval period between two user location points does not reach a fixed threshold T_h. It is represented by $T : p_1 \rightarrow p_2 \rightarrow \cdots \rightarrow p_m$, where, $T_h > p_{i+t}\ t > 0$ with $(m > i \geq 1)$ and $p_i \in P \subset L$. The $|T|$ is the number of samplings ($|T| = m$), and t is defined as the interval of the sampling point. $P = p_1, p_2 \ldots, p_m$ are the arrangement of points known as user movement log, where each point $p_i \in P$ contains p_i.lat, p_i.lng, $p_i.t$, and $p_i.v$ as latitude, longitude, timestamp, and velocity, respectively.

$$p_i = \{p_i.\text{lat}, p_i.\text{lng}, p_i.t, p_i.v\}$$

Also, the location coordinate can change as time passes. Figure 2 provides a distinctly unpredictable glimpse of the initial trajectory data collection. These nodes are related as per the time – series data and thus shape a trajectory. In the equation, a general representation of a record is given below:

	Latitude	Longitude	Time
p_1	28.7041°	77.1025°	22:31
p_2	25.4223°	79.5467°	22.33
.....	.....	.....	.....
p_n	30.2234°	83.3435°	22.43

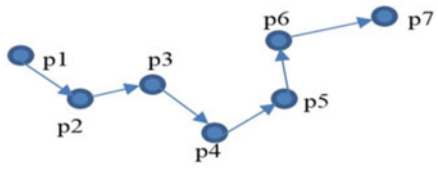

Fig. 2 Log and trajectory for moving person

$$\text{Original Data}: \quad \text{Data}_t : (p_1.\text{lat}, p_1.\text{lng}, p_1.t, p_1.v) \rightarrow (p_2.\text{lat}, p_2.\text{lng}, p_2.t, p_2.v) \rightarrow \rightarrow (p_i.\text{lat}, p_i.\text{lng}, p_i.t, p_i.v)$$

The Gaussian method is used to attain differential anonymity by applying random noise to the time parameter t of the client behavior predicted trajectory results.

The Gaussian method can be described using a data set $D = \{x_1, x_2, \ldots, x_N\}$, privacy parameter ε, global sensitivity Δf of given function f.

In the differential privacy mechanism, with the given sibling data set D and D', the function f sensitivity is represented by Δf as given below:

$$\Delta f = \max_{D \Delta D'} \left\| f(D) - f\left(D'\right) \right\|$$

$D \Delta D^{'}$ is the set of each pair data sets that differs in at most one record.

Theorem 1 For a given output function f: $D^d \rightarrow R^d$, the following function M have (ε, δ)-differential privacy if $\delta > \frac{4}{5} \exp\left(-\frac{\varepsilon\sigma^2}{2}\right)$ and $\varepsilon < 1$.

$$M(f, D) = f(D) + (Y_1, Y_2, \ldots, Y_d)$$

The likelihood of differential privacy is represented by probability δ. The parameter δ bounds the differential privacy level, and its value is smaller than $\left(\frac{1}{|D|}\right)$. The parameter ε is inversely proportional to privacy protection. The Gaussian distribution draw in the form of Y_i $(i = 1, 2, \ldots, d)$ has 0 as the value of mean and $\Delta f \sigma$ as the value of standard deviation, i.e., $Y(0, (\Delta f \sigma))$.

Trajectory data given in the below equation is the trajectory data post-processing the Gaussian noise function value, Gaus $\left(\frac{\Delta f}{\varepsilon}\right)$ to all-time attribute, that can resist an attack through context awareness.

$$\text{Processed Data}: \quad \text{ProData}_t : \left(p_1.\text{lat}, p_1.\text{lng}, \text{p}_1.t + \text{Gaus}\left(\frac{\Delta f}{\varepsilon}\right), \text{p}_1.v\right) \rightarrow \left(p_2.\text{lat}, p_2.\text{lng}, p_2.t + \text{Gaus}\left(\frac{\Delta f}{\varepsilon}\right), p_2.v\right) \rightarrow \rightarrow \left(p_i.\text{lat}, p_i.\text{lng}, p_i.t + \text{Gaus}\left(\frac{\Delta f}{\varepsilon}\right), \ p_i.v\right)$$

3.2 Trajectory Generator

DSPPTD's essential purpose is to increase the performance of trajectory data reporting statistics as well as the scheme's productivity based on maintaining

the differential privacy. Differential privacy frameworks and deep neural network (DNN) deep learning algorithms are the core methodologies applied in this paper. DSPPTD uses differential privacy to offer protection and privacy functionality to LBS apps and uses DNN to efficient trajectory data processing from complex time series. DSPPTD is built for a dynamical object movement, which defines the dynamic model of four components correlated with the speed, latitude, longitude, and time of the users.

3.3 Multilayer Perceptron and Deep Neural Network

A "perceptron" is a known "artificial neuron," forming the "neural" system. This paper first discussed the simplest single hidden layer multilayer perceptron before deep learning-based multilayer perceptron. In general, the multilayer perceptron has the structure in which every location might be represented by way of a single input and a single output neuron and having one hidden layer. Positive weights are typically considered to be excitatory in neural network, whereas negative weights are known to be inhibitory. Training is the method of weight change to build a network that performs some task. The basic architecture of artificial neural network consists of the three components, namely presynaptic connections, which input x_i, synaptic influence, which is modeled using real weights w_i, and neuron reaction, which is a nonlinear weighted inputs function f.

As shown in Fig. 3, x_1, x_2, and x_3 are given as inputs to the perceptron, which produces a single binary output. Piecewise linear and sigmoid are examples of output or response function. The equation for sigmoid and piecewise linear is given below:

$$f(x) = \frac{1}{1 + e^{-\lambda x}} \qquad f(x) = \begin{cases} x, & if \ \ x \geq \theta \\ 0, & if \ \ x < \theta \end{cases}$$

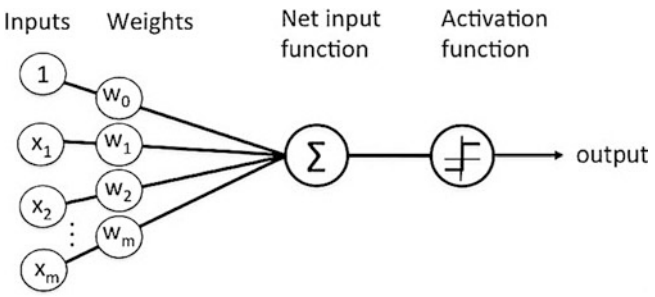

Fig. 3 Perceptron in ANN

The functioning of the human brain is imitated by employing neural network technology for understanding pattern recognition rather than passing the input through the different layers of simulated neural connection. "Artificial neural networks" have an "input layer," at least one "hidden layer" in-between and an "output layer." In "feature hierarchy," specific sorting and order types are carried out in each layer. To deal with unlabeled or unstructured data is among the practical uses of these neural networks. Figure 3 shows the perceptron in artificial neural network (ANN). The leftmost layer refers to "input neurons" present in the "input layer." The rightmost layer refers to the "output neurons" present in the "output layer." The middle layer refers to the "hidden layer," which does not contain the "neurons" of input or the output.

One of the downsides of the "neural network" is cost work slope processing. One of the quicker ways to deal with slope processing is "error back propagation," which gives an in-depth knowledge of changing the metrics toward the system's behavior. The "deep neural network gives the hierarchical composition of the "linear" and "nonlinear" activation function. We propose using "deep neural networks" or "deep learning." In this proposed work, the system considers an input layer, two hidden layers, and a final output layer. The former layers and output layer have been evolved the activation function sigmoid.

The three steps involved in back propagation preparation are listed below:

1. Training set: Neural network uses a collection of input–output patterns for training.
2. Test set: For assessment of neural network performance, another collection of input–output patterns are used.
3. Learning rate: It is a scalar parameter used to determine the change rate, which is similar to phase size in numerical integration.

Network error is used as termination criteria or as an indicator for desired training of the neural network. Root mean square error (RMSE) and sum squared error (SSE) are the two most important indicators commonly used in most of the neural network applications. The equations for root mean square error (RMSE) and total sum squared error (SSE) are given below:

$$\text{RMSE} = \sqrt{\frac{2^{*}\text{TSSE}}{\#\text{patterns}^{*}\#\text{outputs}}}$$

$$\text{TSSE} = \frac{1}{2} \sum_{\text{patterns}} \sum_{\text{outputs}} (\text{desired} - \text{actual})^2$$

The deep neural network processed trajectory data is a new trajectory that may mask the original data set for the trajectory. Using this training model, we can only get more anonymous data according to specific points in the complex trajectory. The

model facilitates the avoidance of loading complete data sets inside the standard procedures and leads to running time reduction. The trajectory data given in the below equation is the trajectory data after deep neural network processing of the trajectory data:

$$\begin{aligned} \text{DNNData}_t : (p_{1\text{DNN}}.\text{lat}, p_{1\text{DNN}}.\text{lng}, p_{1\ \text{DNN}}.t, p_{1\text{DNN}}.v) \rightarrow \\ (p_{2\text{DNN}}.\text{lat}, p_{2\ \text{DNN}}.\text{lng}, p_{2\ \text{DNN}}.t, p_{2\ \text{DNN}}.\text{v}) \rightarrow \\ \rightarrow (p_{\text{iDNN}}.\text{lat}, p_{\text{i DNN}}.\text{lng}, p_{\text{i DNN}}.t, p_{\text{i DNN}}.v) \end{aligned}$$

The trajectory data generation procedure can be described by Algorithm 1 as given below:

Algorithm 1: Differential Privacy Generation of Trajectory Data

Input: Trajectory Original Data (Data_t)
Output: DNN processed Trajectory Data (DNN_Data_t)
Begin
For_ALL Data_t
  For_ALL $t \neq 0$ in Data_t
    $\Delta f = \text{Gaus}\left(\frac{\Delta f}{\varepsilon}\right) = \max_{D \Delta D'} \left\| f(D) - f(D') \right\|$
    $t_{dnn} = t + \Delta f$
    $lat_{dnn} = DNN_{lat}(Pro\_Data_t, t_{dnn})$
    $lon_{dnn} = DNN_{lon}(Pro\_Data_t, t_{dnn})$
    $v_{dnn} = DNN_{v}(Pro\_Data_t, t_{dnn})$
    $(lat_{dnn}, lon_{dnn}, t_{dnn}, v_{dnn}) \rightarrow DNN\_Data_t$
  End_For
End_For
Return DNN_Data_t
End

3.4 K-Paths Trajectory Clustering

Partition-based approaches are more like clustering techniques that are categorized before processing by the count of clusters (or centers). A parameter k ($k \leq n$, n is the data point count in the data set) is needed to set the count of final data partitions. The cluster is represented by partitions, which must require at least one data point. Partition-based approaches involve techniques of k-medoids and k-means. In [] and [], two improved variants of k-means and k-medoids are described. The k-means algorithms have been utilized in several clustering projects. The central concept is to locate k cluster centers randomly and then in an iterative manner, a grouping of the piece of data according to the divergence to the nearest clustering center until all clustering centers converge.

This step involves the k subregions division of the location trajectory data region of common points. Here, the positions data with the same timestamp t is first segmented. Then k subregions or groups are identified with similar data points, and initial centroid corresponding to each subregion is chosen. If, at any instance, the area covers a more significant number of mobile users than the threshold, then k needs to be revised accordingly. In clustering, the location data with closer trajectories are merged into a common cluster.

The k-paths trajectory clustering process can be defined as given below:

Given a set of trajectories T: $p_1 \rightarrow p_2 \rightarrow \cdots \rightarrow p_m$, the goal of the k-paths is to divide the n trajectories into k ($k \leq n$) clusters groups $C = \{C_1, C_2, \ldots, C_k\}$ to minimize the below objective function:

$$O = \arg\min_{C} \sum_{j=1}^{k} \sum_{p_i \in C_x} \text{Dist}\,(p_i, \mu_x)$$

where each clusters C_x have their centroid path μ_x, which is an element of the set of paths in road network directed graph G [34], and Dist is the measure of the Euclidean distance between two trajectories.

The k-means and k-paths can be differentiated based on the following four points:

(a) In a Euclidean space, trajectories can differ in length rather than fixed-length vectors.
(b) A trajectory length estimate "Dist" must be specified for two trajectories.
(c) We cannot locate the centroid direction μ_x by merely measuring the average value with each trajectory throughout the cluster. Analogous to a version of k-means named k-medoids [35], it is possible to use a current trajectory as the centroid path.

Let EH, ALH are the edge histograms and accumulated length histograms, respectively. The terms ub(i) and lb.(i) be the T_i to its nearest cluster upper bound distance and the T_i to its second nearest cluster lower bound distance, respectively. The terms cd(x) and cb(x) be the centroid drift and centroid bound of μ_x, respectively. The formula for edge-based distance measure used in Algorithm 2 is given below:

$$\text{Edge-Based-Distance}\,(T_1, T_2) = \max\,(|T_1|, |T_2|) - |\,T_1 \cap T_2\,|$$

$|T1|$ and $|T1|$ be the travel length of the total trajectory T_1 and T_2, respectively. In k-path trajectory clustering, the trajectory distance measure "Dist" is replaced by edge-based distance. Therefore, the applied objective function has been revised as given below:

$$O = \arg\min_{C} \sum_{j=1}^{k} \sum_{p_i \in C_x} \text{Edge-Based-Distance}\,(p_i, \mu_x)$$

Algorithm 2: K-Paths Clustering (K, DNN_Data$_t$)

Input: Number of clusters (k), DNN processed Trajectory Data (DNN_Data$_t$)
Output: k centroid paths: {$\mu_1, \ldots, \mu_k$}.
Begin
Centroid paths μ = {μ1, · · · , μk} initialization, t ← 0;
Repeat
    If (t = 0)
        For Each $T_i \in$ DNN_Data$_t$ do
            mini ← +infinity;
            For Each path centroid μ_j do
                lb(i, j) ← Edge-Based-Distance(p_i, μ_j);
                If (min_i > lb(i, j)) then
                    a(i) ← x
                    mini ← lb(i, x)
            End For
            UpdateHistogram(p_i, ALH, EH, a(i));
        End For
    Else
        For Each cluster
            Compute and make changes to centroid bound cb and centroid drift cd
        End For
        For Each trajectory $T_i \in$ DNN_Data$_t$ do
            Compute and make changes to lb and ub;
            If (ub(i) < max(cb(a'(i))/2, lb(i))) then
                a(i) ← a'(i) \\T_i remain in same cluster:
            Else
                mini ← +infinity;
                For Each path centroid μ_x do
                    If (lb(i, x) < ub(i)) then
                        lb(i, x) ← Edge-Based-Distance(p_i, μ_x);
                        If (mini > lb(i, x)) then
                            a(i) ← x
                            mini ← lb(i, x);
                        End If
                    End If
                End For
            End If
            If $a'(i) \neq a(i)$
                UpdateHistogram(p_i, ALH, EH, a(i));
            End If
        End For
    For Each centroid path μj do
        Compute
    $O = \underset{C}{arg\min} \sum_{j=1}^{k} \sum_{p_i \in C_x} \text{Edge-Based Distance}(p_i, \mu_x)$ and update μ_x ;
    End For
    t ← t + 1;
While (t = 0 or μ changed)
Return {μ1, . . . , μk}

The trajectory path-k clustering procedure can be described by Algorithm 2. To adjust the centroid direction in iteration and to determine the objective function system manages two histograms of trajectory for each cluster.

(a) Edge histogram: The edge histogram (EH_j) for given trajectories in cluster C_j has the graph edges frequency information in sorted order. $EH_j(e)$ stands for the edge e frequency, i.e., $EH_j(e) = |e|$, and $EH_j[l]$ stands for the l-th most considerable frequency. In any iteration system, no need to reconstruct the histograms; instead, it holds one histogram progressively for every cluster and refreshes it only as a trajectory passes through in or goes out of this cluster. Many trajectories would continue in the same cluster for further iteration, although there would be few changes to the histogram.
(b) Accumulated length histograms: The critical point is the size in a meter of the trajectories for each entry. This histogram measures the number of trajectories that have this defined size. ALH is ordered by key in ascending order; $ALH_x[l]$ gives the trajectories count in cluster C_x that have a size l.

3.5 Trajectory Release

Trajectory release is the last step, which involves comparing each clustered "synthetic trajectories datum" to corresponding "real" trajectory and merging accordingly. The process also involves a prejudging mechanism to ensure at least one actual trajectory record can be seen in processed trajectory. So when the count of records is zero, it means that the produced trajectory data is a null trajectory and is considered to be irregular. The probability of issuing a null trajectory is further minimized due to the inclusion of the decision process of an irregular course, the reliability of the orbiting assignment is increased, and better data availability has been assured.

4 Performance Analysis of Privacy Protection Scheme

To check the feasibility of our proposed approach and the data availability, we performed specific tests based on TDrive pre-project data from Microsoft research [38], which includes the trajectory details of 10,357 taxis for a week duration. The cumulative points count is about 15 million, for a cumulative trajectory size of nine million kilometers. The evaluation was conducted on Octa-core 3.2 GHz, RAM of 64 GB, Windows 8 operating system, and Intel i7 processor. The processing time overhead of the query and service schedule is assumed to be negligible in the proposed model. Location-based services have drawn millions of users and their digital footprints are massively contained. The query process and interval process are the two modules executed for the simulation of the proposed model.

A location-dependent k-nearest neighbor query (e.g., nearest hospital profile info) is continuously generated by the query process with the exponential distributed query interval. Driven by the assessment process surrounding anonymity, modeling, and uncertainty [36], we are examining the relationship between the efficacy and usefulness of data security. Past policies are based on user-clustered or user-centralized architectures. Hence, the workload of the network is high, and the anonymizer could lead the bottleneck performance. Different from existing work, our suggested methodology integrates the fog server, which processes the data in an IoT gateway or fog node, as it is nearer to the consumer and can be partly managed by the user [37]. For the safety of trajectories, we assume in our system the time-dependent mobility trend, probability of query, and spatiotemporal connection. It produces $k - 1$ dummy positions and trajectories with full entropy, which can render offline and online original trajectory security. Here, we have undertaken two measures, namely, mutual information and Hausdorff distance, to establish this relationship and evaluate the proposed policy. We have a belief that consideration of these measures may assist in choosing and implementing acceptable methods of privacy security for particular situations on the pathway. The two measures, i.e., mutual information and Hausdorff distance, can be defined as given below.

Mutual information: Mutual information (MI) is a measure of privacy protection intensity of a given privacy protection scheme. It is directly proportional to the differential privacy parameter (ε) and inversely proportional to privacy protection intensity. The differential privacy budget is represented by ε, which is also known as the differential privacy parameter.

Hausdorff distance: Hausdorff distance is a method for calculating the difference in a metric space between two sets of points and has been commonly used to calculate the spatial dissimilarity of two trajectories. We measure the Hausdorff distance from each pair of initial trajectories to the synthetic ones. A higher value of Hausdorff distance between trajectories pair represents high dissimilarity of two trajectories, and so it has a reduced set of POI than original trajectory POIs. Therefore, the higher Hausdorff distance value shows a lower utility of given trajectory data for LBS.

From the comparative analysis of past policies such as TSTDA [38], NGTMA [39], and SDD [40] with the proposed state-of-the-art proposed scheme deep neural network-based differential privacy protection policy, it is proven that DSPPTD outperforms the other policy with the highest privacy protection intensity in terms of mutual information (MI) and trajectory data utility in terms of Hausdorff distance (HD) has been computed for all models, which have been depicted in Figs. 4 and 5.

The DSPPTD does have the lowest MI level, which shows that RNN-DP has a higher level of privacy security relative to NGTMA, TSTDA, and SDD methods. In this study, we discover that the level of privacy security is directly linked to ε as depicted in Fig. 4. Because DSPPTD uses the Gaussian method in the data collection step in addition to the exponential method in the data release phase; therefore, the dual differential privacy security protocols provide better privacy protection.

As depicted in Fig. 5, DSPPTD has the smallest HD of the four systems, so the data set for publishing is identical to the initial data collection. DSPPTD has

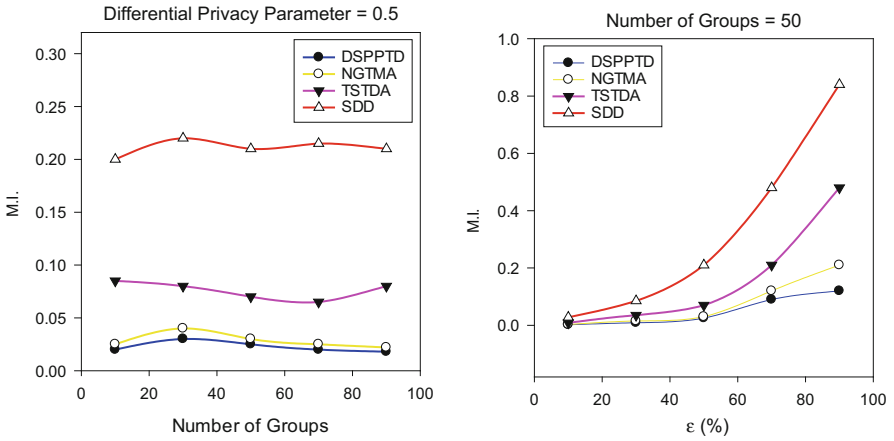

Fig. 4 Effect of number of cluster groups and differential privacy parameter (ε) on mutual information (MI)

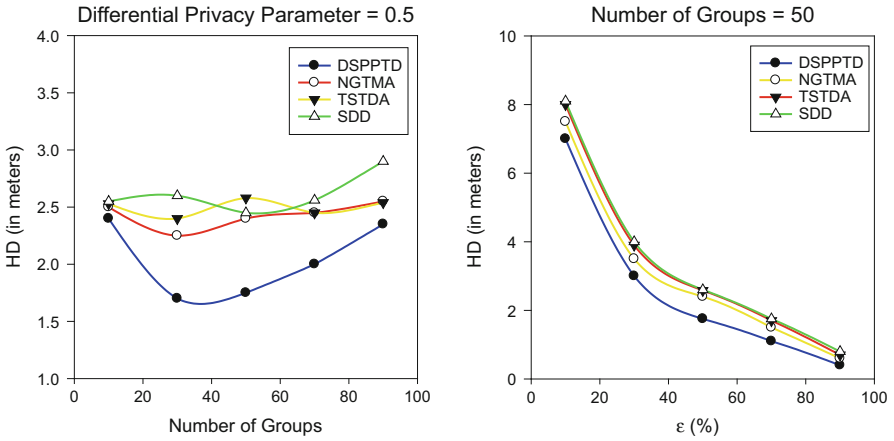

Fig. 5 Effect of number of cluster groups and differential privacy parameter (ε) on Hausdorff distance (HD)

increased the practicability of results. The fundamental explanation for this is the prediction system convergence. The trajectory data redundancy is carried out during the processing of the data. When DSPPTD discovers the data to be incorrect, it removes this data to boost the reliability of the reported trajectory data. The reliability of these data leads to the higher utility of the LBS.

As shown in Fig. 6, DSPPTD seems to have the lowest execution time of algorithms within a separate budget for privacy. The algorithm's execution time comprises of time for generating noise and time for processing trajectories. The execution time of the proposed policy is correlated with the algorithm's time for

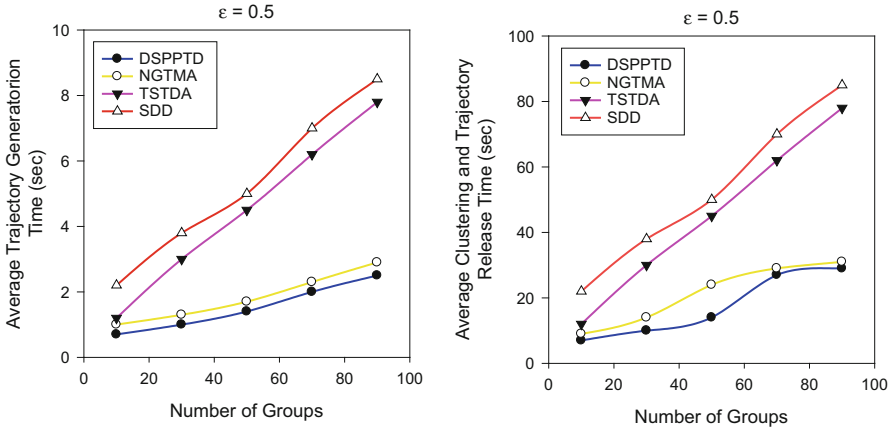

Fig. 6 Comparison of execution time based on number cluster groups

processing trajectories while the noise generation time is the same for all algorithms. The distinction lies in the computation time of trajectories. DSPPTD has the benefit of utilizing the projected model trajectory data collection for the study and is not the time series data conventional processing. So, it has better execution time efficiency. We concluded, therefore, that DSPPTD ensures computing security and availability of data, along with the high efficiency of the device in terms of the running time.

5 Conclusion

In this work, we introduced Dynamic Scheme for Privacy Protection of Trajectory Data (DSPPTD). DSPPTD involve Gaussian framework and double differential privacy requirement focused on deep learning to provide private security and edge computing based on enhanced utility services. For consumer services, a mechanism of dual deep learning-based differential privacy model has been suggested. Via empirical study, we have shown that DSPPTD has more effective privacy security strength, better data efficiency, and overall reliability than state-of-the-art systems currently existing.

Our future research will concentrate on improving the trajectory resemblance loss metric model, expanding our system to global trajectory data sets, creating personalized simulated trajectory data for variable lengths, investigating possible attacks on privacy and security techniques, and assessing the efficacy and usefulness of our system in other trajectory data mining and analytics schemes.

Competent Interest Declaration On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Sun, G., et al. (2017). Efficient location privacy algorithm for Internet of Things (IoT) services and applications. *Journal of Network and Computer Applications, 89*, 3–13. https://doi.org/10.1016/j.jnca.2016.10.011
2. Gupta, A. K., & Shanker, U. (2020). Some issues for location dependent information system query in Mobile environment. In *29th ACM international conference on information and knowledge management (CIKM '20)* (p. 4). https://doi.org/10.1145/3340531.3418504
3. Gupta, A. K., & Shanker, U. (2018). Location dependent information System's queries for Mobile environment. In *Lecture notes in computer science* (pp. 218–226). https://doi.org/10.1007/978-3-319-91455-8_19
4. Zakhary, S., & Benslimane, A. (2018). On location-privacy in opportunistic mobile networks, a survey. *Journal of Network and Computer Applications, 103*, 157–170. https://doi.org/10.1016/j.jnca.2017.10.022
5. Hu, H., Sun, Z., Liu, R., & Yang, X. (2019, July). Privacy implication of location-based service: Multi-class stochastic user equilibrium and incentive mechanism. *Transportation Research Record, 2673*(12), 256–265. https://doi.org/10.1177/0361198119859322
6. Gupta, A. K., & Shanker, U. (2020). OMCPR: Optimal mobility aware cache data prefetching and replacement policy using spatial K-anonymity for LBS. *Wireless Personal Communications, 114*(2), 949–973. https://doi.org/10.1007/s11277-020-07402-2
7. Shen, H., Bai, G., Yang, M., & Wang, Z. (2017). Protecting trajectory privacy: A user-centric analysis. *Journal of Network and Computer Applications, 82*, 128–139. https://doi.org/10.1016/j.jnca.2017.01.018
8. Niu, B., Zhang, Z., Li, X., & Li, H. (2014). Privacy-area aware dummy generation algorithms for location-based services. In *2014 IEEE International Conference on Communications (ICC)* (pp. 957–962). https://doi.org/10.1109/ICC.2014.6883443
9. Indyk, P., & Woodruff, D. (2006). Polylogarithmic private approximations and efficient matching. In *Theory of cryptography* (pp. 245–264). Springer.
10. Gupta, A. K., & Shanker, U. (2020). MAD-RAPPEL: Mobility aware data replacement & prefetching policy enrooted LBS. *Journal of King Saud University – Computer and Information Sciences*. https://doi.org/10.1016/j.jksuci.2020.05.007
11. Gambs, S., Killijian, M., & Cortez, M. N. D. P. (2013). De-anonymization attack on Geolocated data. In *2013 12th IEEE international conference on trust, security and privacy in computing and communications* (pp. 789–797). https://doi.org/10.1109/TrustCom.2013.96
12. Liu, H., Darabi, H., Banerjee, P., & Liu, J. (2007). Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 37*(6), 1067–1080. https://doi.org/10.1109/TSMCC.2007.905750
13. Petrou, L., Larkou, G., Laoudias, C., Zeinalipour-Yazti, D., & Panayiotou, C. G. (2014). Demonstration abstract: Crowdsourced indoor localization and navigation with anyplace. In *IPSN-14 proceedings of the 13th international symposium on information processing in sensor networks* (pp. 331–332). https://doi.org/10.1109/IPSN.2014.6846788
14. Kido, H., Yanagisawa, Y., & Satoh, T. (2005). An anonymous communication technique using dummies for location-based services. In *Proceedings of ICPS* (pp. 88–97).
15. Lu, H., Jensen, C., & Yiu, M. (2008). *PAD: Privacy-area aware, dummy-based location privacy in mobile services*. https://doi.org/10.1145/1626536.1626540
16. Niu, B., Gao, S., Li, F., Li, H., & Lu, Z. (2016). Protection of location privacy in continuous LBSs against adversaries with background information. In *2016 International Conference on Computing, Networking and Communications (ICNC)* (pp. 1–6). https://doi.org/10.1109/ICCNC.2016.7440649
17. Samarati, P., & Sweeney, L. (1998). Generalizing data to provide anonymity when disclosing information (Abstract). In *Proceedings of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on principles of database systems* (p. 188). https://doi.org/10.1145/275487.275508

18. Zhang, S., Li, X., Tan, Z., Peng, T., & Wang, G. (2019). A caching and spatial K-anonymity driven privacy enhancement scheme in continuous location-based services. *Future Generation Computer Systems, 94*, 40–50. https://doi.org/10.1016/j.future.2018.10.053
19. Niu, B., Li, Q., Zhu, X., Cao, G., & Li, H. (2014). Achieving k-anonymity in privacy-aware location-based services. In *IEEE INFOCOM 2014 – IEEE Conference on Computer Communications* (pp. 754–762). https://doi.org/10.1109/INFOCOM.2014.6848002
20. Guan, Z. et al., (2019, January). APPA: An anonymous and privacy preserving data aggregation scheme for fog-enhanced IoT. *Journal of Network and Computer Applications*, *125*, 82–92. https://doi.org/10.1016/j.jnca.2018.09.019.
21. Zhu, H., Yang, X., Wang, B., Wang, L., & Lee, W.-C. (2019). Private trajectory data publication for trajectory classification. In *Web information systems and applications* (pp. 347–360).
22. Gruteser, M., & Grunwald, D. (2003). Anonymous usage of location-based services through spatial and temporal cloaking. In *Proceedings of the 1st international conference on Mobile systems, applications and services* (pp. 31–42). https://doi.org/10.1145/1066116.1189037
23. Palanisamy, B., & Liu, L. (2011). MobiMix: Protecting location privacy with mix-zones over road networks. In *2011 IEEE 27th international conference on data engineering* (pp. 494–505). https://doi.org/10.1109/ICDE.2011.5767898
24. M. Nergiz, M. Atzori, and Y. Saygin, Towards trajectory anonymization: A generalization-based approach. 2008.
25. Hampton, K., et al. (2010, November). Mapping health data: Improved privacy protection with donut method Geomasking. *American Journal of Epidemiology, 172*, 1062–1069. https://doi.org/10.1093/aje/kwq248
26. Gao, S., Rao, J., Liu, X., Kang, Y., Huang, Q., & App, J. (2019, December). Exploring the effectiveness of geomasking techniques for protecting the geoprivacy of Twitter users. *Journal of Spatial Information Science*. https://doi.org/10.5311/JOSIS.2019.19.510
27. Zandbergen, P. (2014, April). Ensuring confidentiality of geocoded health data: Assessing geographic masking strategies for individual-level data. *Advances in Medicine, 2014*, 1–14. https://doi.org/10.1155/2014/567049
28. Kwan, M.-P., Casas, I., & Schmitz, B. (2004, June). Protection of Geoprivacy and accuracy of spatial information: How effective are geographical masks? *Cartographica the International Journal for Geographic Information and Geovisualization, 39*, 15–28. https://doi.org/10.3138/X204-4223-57MK-8273
29. Seidl, D. E., Jankowski, P., & Tsou, M.-H. (2016, April). Privacy and spatial pattern preservation in masked GPS trajectory data. *International Journal of Geographical Information Science, 30*(4), 785–800. https://doi.org/10.1080/13658816.2015.1101767
30. Beresford, A. R., & Stajano, F. (2003). Location privacy in pervasive computing. *IEEE Pervasive Computing, 2*(1), 46–55. https://doi.org/10.1109/MPRV.2003.1186725
31. Liu, X., Zhao, H., Pan, M., Yue, H., Li, X., & Fang, Y. (2012). Traffic-aware multiple mix zone placement for protecting location privacy. In *2012 Proceedings IEEE INFOCOM* (pp. 972–980). https://doi.org/10.1109/INFCOM.2012.6195848
32. Hasan, A. S. M. T., Jiang, Q., & Li, C. (2017, October). An effective grouping method for privacy-preserving bike sharing data publishing. *Future Internet, 9*, 65. https://doi.org/10.3390/fi9040065
33. Li, X., Zhu, Y., Wang, J., Liu, Z., Liu, Y., & Zhang, M. (2018). On the soundness and security of privacy-preserving SVM for outsourcing data classification. *IEEE Transactions on Dependable and Secure Computing, 15*(5), 906–912. https://doi.org/10.1109/TDSC.2017.2682244
34. Gupta, A. K., & Shanker, U. (2020). Study of fuzzy logic and particle swarm methods in map matching algorithm. *SN Applied Sciences, 2*, 608. https://doi.org/10.1007/s42452-020-2431-y
35. Park, H.-S., & Jun, C.-H. (2009). A simple and fast algorithm for K-medoids clustering. *Expert Systems with Applications, 36*(2, Part 2), 3336–3341. https://doi.org/10.1016/j.eswa.2008.01.039
36. Gupta, A. K. (2020). Spam mail filtering using data mining approach: A comparative performance analysis. In S. Shanker & U. Pandey (Eds.), *Handling priority inversion in time-constrained distributed databases* (pp. 253–282). IGI Global.

37. Wang, T., et al. (2017). Trajectory privacy preservation based on a fog structure for cloud location services. *IEEE Access, 5*, 7692–7701. https://doi.org/10.1109/ACCESS.2017.2698078
38. Hua, J., Gao, Y., & Zhong, S. (2015). Differentially private publication of general time-serial trajectory data. In *2015 IEEE Conference on Computer Communications (INFOCOM)* (pp. 549–557). https://doi.org/10.1109/INFOCOM.2015.7218422
39. Li, M., Zhu, L., Zhang, Z., & Xu, R. (2017, March). Achieving differential privacy of trajectory data publishing in participatory sensing. *Information Sciences, 400*. https://doi.org/10.1016/j.ins.2017.03.015
40. Jiang, K., Shao, D., Bressan, S., Kister, T., & Tan, K.-L. (2013). *Publishing trajectories with differential privacy guarantees*. https://doi.org/10.1145/2484838.2484846