Original Article

A mobile database security approach with emphasis on privacy using location-based services with k-anonymity model

Mohammadreza Mollahosini Ardakani[1], Fatemeh Zahedi[2]*, Zahra Zahedi[3]

[1] Assistant professor, computer Engineering Department, Meybod Branch, Islamic Azad University, Meybod, Iran. [2] PhD student, Faculty of Engineering, Department of Computer Science, Apadana, Institute of Higher Education, Shiraz, Iran. [3] Masters in Computer Engineering, Faculty of Engineering, Department of Computer Science, Apadana, Institute of Higher Education, Shiraz, Iran.

Correspondence: Fatemeh Zahedi, PhD student, Faculty of Engineering, Department of Computer Science, Apadana, Institute of Higher Education, Shiraz, Iran. Email: zahediifa@gmail.com

ABSTRACT

Continuous developments in mobile networks and positioning technologies have created a strong market pressure for location-based services (LBS). Examples include location-based emergency services, location-based service ads, and location-sensitive billing. One of the major challenges in deploying LBS systems extensively is the privacy of location-based data. Without security, the widespread deployment of location-based services endangers the privacy of mobile users and exposes significant vulnerabilities to exploitation. In this article, we describe a customizable identity model to protect the privacy of location data. Our model has two unique features. First, we provide a customizable framework for supporting variable naming with the k variable, allowing a wide range of users to take advantage of location privacy security with personalized personalization requirements. Second, we design and develop a spatial and temporal hidden cipher algorithm called Clique Cloak, which provides location anonymity to LBS provider mobile users. The secrecy algorithm is run by a location protection broker on a trusted server, which hides node-related messages by hiding location information in messages to reduce or prevent privacy threats before sending them to the LBS provider. Makes cell phones anonymous. Our model enables each message sent from a mobile node to specify the level of anonymity that is desired as well as the maximum time and space tolerance needed to maintain the anonymity. We generate the effect of the undercover algorithm under artificial conditions using realistic artificial location data using real road maps and traffic volume data. Our experiments show that the k-anonymity location model with multi-dimensional secrecy and the k-adjustment parameter can obtain a high guarantee of anonymity k and high resistance to location privacy threats without significant performance penalty. The results shows after presenting the proposed solution, the mobile dataset was introduced along with the criteria for deviation ratio, execution and prediction accuracy. Providing information and predictive accuracy has improved significantly.

Keywords: Mobile Database, Privacy, Location Based Services, K Anonymity

Introduction

The use of mobile databases is rapidly increasing due to the continued growth of higher-capacity hardware devices and more powerful CPUs together with the rapid development of wireless technology. Mobile devices are gradually being used for database applications such as sales, ordering, customer relationship management. Data access method and management of them has changed completely according to these applications are mobile. Instead of using a centralized database, these applications are closer to the application's independence and performance enhancement. There are various definitions of a mobile database. Some researchers define the mobile database as a database that is stored on mobile devices such as laptops, PDAs and mobile

Access this article online	
Website: www.japer.in	**E-ISSN:** 2249-3379

How to cite this article: Mohammadreza Mollahosini Ardakani, Fatemeh Zahedi, Zahra Zahedi. A mobile database security approach with emphasis on privacy using location-based services with k-anonymity model. J Adv Pharm Edu Res 2020;10(S1):137-148.
Source of Support: Nil, Conflict of Interest: None declared.

phones [1]. Other researchers have identified it as a distributed database that is available in mobile mode [2]. Others see it as a collection of distributed databases, fixed databases and ad-hoc databases and scattered information repositories. A distributed database is a location for a mobile database that addresses the requests of mobile users [3]. Vijay Kumar presents the reference architecture (Fig.1) for the mobile database [2]. In this architecture, the fixed host (FH) and base station (BS) are connected via high-speed wired network. One or more base stations are connected to the Base Station Control Center (BSC), which coordinates the operation of the base station by the Mobile Switching Center (MSC). Some basic data processing capabilities are incorporated in the base stations so that they can be Synchronize with Database Servers (DBs) Unlimited mobility in PCS and GSM is enabled by wireless communication between the base station and the mobile unit (MU). Combine data with PCM or GSM, any database server can access any fixed base station or host. The combination of MSC and PSTNleads to the [1] connection of the mobile database system to the outside world.

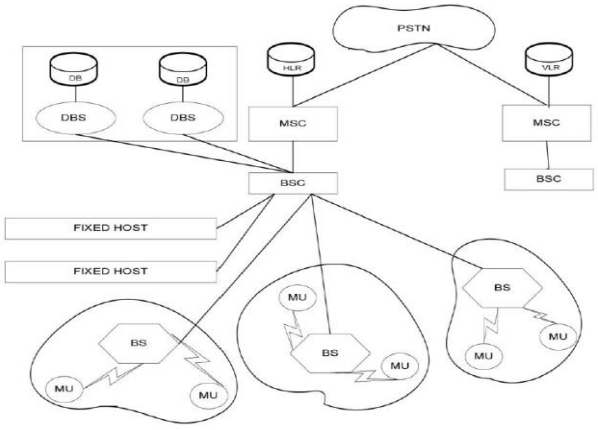

Figure 1: References Architect [2]

Apart from the reference architecture, there are three other types of mobile database architectures: client architecture/ Server, server architecture/ agent / client architecture, peer-to-peer architecture.

With the advent of wireless communications with the Global Positioning System (GPS), location based services (LBS) have attracted considerable attention and are becoming one of the fastest growing services [4-6]. In an LBS, a mobile user sends a location request (LSP) containing their location and interests. The LSP locates a POI[2] near the user's current location (for example, garages, car parks, supermarkets, and restaurants) according to user interests. However, the LSP may violate user privacy [7, 8]. By collecting user requests, an unreliable LSP can infer personal information about the user, such as his location, preferences, mode of transport, and possibly his health [9, 10]. Even worse, the LSP can disclose user private information for financial or other third party business interests. As a result, protecting user privacy is beneficial. Different approaches have been proposed to reduce the risk of loss of privacy in LBSs. Most of them adopt a fully trusted third party architecture (CTP) [11, 12], which acts as anonymous. When a user issues a request, it sends it to anonymous to remove the user's caller identity before sending the request to the LSP. In addition to removing the caller identity, the anonymous also frequently blurs the user's location with "cloaking". Cloaking is where the user's actual location is replaced by a loop. The user is somewhere in this loop, like other K-1 users, thereby providing K-anonymity [13, 14]. However, it destroys the POI, resulting in the POI results being refined to anonymous to return more accurate results to the user. While it is likely that the user will bypass the results almost immediately, it is anonymous to store them so that they can respond to future requests. They can receive their POI service directly from anonymous stored results, no need to submit their requests to the LSP.

Nowadays, enormous amount of location information will be produced by digital devices such as smartphones [15], smart watches [16], smart glasses [17, 18], navigators [19], traffic information systems [20, 21], closed-circuit cameras [22], Internet of Things (IoT) sensors and robots [23], medical devices, biomimetic [24], drones and satellite [19], etc. Meanwhile, related to the location information, sub-sequence matching, time-series analysis, and trajectory pattern mining [25] have been employed, but they have a well-known limitation of safety assurance. To overcome the limitation, some studies have been conducted about the anonymity of location data such as K-anonymity [26], I-diversity [27], t-closeness [28], location semantics [29], differential privacy [17, 30]. They have been usually employed on the point-based location information, but they are not appropriate to the anonymize trajectory database. In addition, the researches have revealed a critical drawback in that they have greatly restricted data utility (e.g., 95% of data removal) [31]. Basically, the stronger to stress on securing anonymity, the smaller is the data utility. Without a fundamental breakthrough that solves this dilemma, it is inevitable that the negative effect of utility constraint will be continued, since personal information is the more important one to be secured (e.g., GDPR in EURO [32]). Furthermore, differential privacy for queries has been widely used to protect individual privacy. This returns a changed result by adding and removing Laplace noise to an original result for the queries to protect individual privacy included in database [20, 30].

Researchers have proposed multiple solutions for solving privacy disclosure problems caused by LBSs. The existing privacy protection methods for LBSs mainly include data encryption, pseudo addresses, space conversions and anonymity areas [33–38]. These methods are mostly focused on the location data, without delving into the relationships between the location data and the users, or the privacy implications of the location data. It is difficult to capture the significance of real-time human activities. Therefore, a growing number of researchers have studied location privacy protection based on semantics, with a view toward achieving a deeper level of protection. The protection of semantics-based moving object trajectories has also become a focus of more research [39–41]. With the increasing awareness of semantic information in trajectories, trajectory protection

[1] Public Switched Telephone Network

[2] Point of Interests

methods have gradually developed into methods based on semantics. Monreale et al. classified locations in order to generate generalized user access addresses, which enabled the creation of anonymity trajectory datasets that ensured that the probability of identifying user IDs and accessing sensitive locations was lower than a given threshold [42]. Lee et al. also imposed a threshold on the information obtainable by adversaries [43]. They suggested exploring location semantics by observing users' length of stay. Moreover, the ratio of suppressed frequent sequences is a direct indication of anonymized data quality for trajectory pattern mining [44, 45].

Modeling

K-Anonymity

k- Anonymity is a model that addresses the question, "How can a data holder publish a copy of his private data with scientific assurances that the data subject cannot be re-identified while the data are they actually useful?" [46] For example, a medical institution may want to publish a medical record table. Even if people's names can be replaced with dummy identifiers, some features (so-called pseudo-identifiers) can extract confidential information. For example, the date of birth, zip code, and gender characteristics in the disclosed table can determine a person individually. Joining such a table with some other information sources available, such as the voter list table, which contains records containing traits that form pseudo-identities as well as individuals' identities, medical information can be easily linked to individuals. k - Anonymity prevents such privacy violations by ensuring that any personal record exists only if there are at least k - 1 other (distinct) individuals that are not identifiable by the pseudo - identifiable values of the former.

Location Based K-Anonymity

In the context of LBS and mobile users, location anonymity requires that location information contained in a message sent from a mobile user to LBS is not detectable from at least k-1 messages other than messages from different mobile nodes. [47]. In general, anonymity in LBS depends on the confidence of the individuals involved and should be examined in several layers on the network stack. In this article, we will solve the problem of having an k location in the application layer by accessing LBS providers with anonymous location information. The location protection algorithm uses location-to-location secrecy to convert any original message from a mobile node to a privacy-protected message with a k-anonymity guarantee. As shown in [47], anonymity k can be used to prevent attacks such as confined space detection and observation detection. The former discloses identity by joining a known community of a message, while the latter by identifying location information from external viewing to a message.

k-Anonymous Location Information

In order to record different privacy conditions and ensure different levels of service quality, each message from the mobile node also specifies its level of anonymity (k value), location tolerance, and time tolerance. The primary task of the anonymous server is to convert the location of each message received from the cellular nodes into another message that can be transferred (k-anonymously) to the LBS provider (by sudden transmission). The basic idea under the K-anonymity model is twofold. In the first step, by reducing the accuracy of the location by enlarging the revealed spatial area, one can maintain a certain degree of location anonymity, such that the other k-1 mobile nodes exist in the same spatial region. This approach is called spatial abduction. Second, it is possible to delay the sending of messages to obtain anonymity of the location until the mobile nodes visit the same area located by the sender. This is called temporary abduction. We express the set of messages received from mobile nodes as S. We formally define the messages in the S series as follows:

$$m_s \in S : \left\langle u_{id}, r_{no}, \{t, x, y\}, k, \{d_t, d_x, d_y\}, C \right\rangle$$

Messages are identifiable by the sender ID, the message reference number pair (u_{id}, r_{no}), in the S. Set of messages from the same cellular node have the same sender ID but have different reference numbers. In a received message, x, y, and t together form the three-dimensional spatiotemporal point of the message that is marked as $P(m_s)$. The coordinates (x, y) refer to the location of the mobile node in two-dimensional space (ie, the x-axis and the y-axis), and timestamp t refers to the time the mobile node was present. That position (time dimension: message axis). The k value of the message indicates the minimum level of anonymity you want. The value of k = 1 means that the message is not anonymous. A value of k> 1 means that the converted message is assigned a spatially-hidden cache that is no longer detectable by at least k-1 converted messages, each with a different mobile node. Larger k values indicate a higher degree of anonymity. The message value d_t represents the user-specified time tolerance. This means that the converted message must have a spatially-hidden box whose layout contains no point more than t. Likewise specify the tolerances according to the spatial dimensions. The values of these three parameters depend on the external LBS requirements and user settings depending on the quality of service. For example, longer spatial tolerance may lead to more accurate results than location-dependent service requests, and longer temporal tolerance may result in longer message delays. Let $\Phi(v, d) = [v - d, v + d]$ be a function that extends a numerical value v to a range by amount d. Then, we denote the anonymity constraint box of a message m_s as $B_{cn}(m_s)$ and define it as:

$$\left(\Phi(m_s.x, m_s.d_x),\ \Phi(m_s.y, m_s.d_y),\ \Phi(m_s.t, m_s.d_t)\right)$$

The field C in m_s denotes the message content. We denote the set of transformed (anonymized) messages as T. We formally define the messages in the set T as follows:

$$m_t \in T : \langle u_{id}, r_{no}, \{X : [x_s, x_e], Y : [y_s, y_e], I : [t_s, t_e]\}, C \rangle$$

For each message m_s in S, there exists at most one corresponding message m_t in T. We call the message m_t, the transformed format of message m_s, denoted as $m_t = R(m_s)$ Concretely, if $m_t = R(m_s)$, then $m_t.u_{id} = m_s.u_{id}$ and $m_t.r_{no} = m_s.r_{no}$. The (u_{id}, r_{no}) fields of a message in T should be replaced with a dummy identifier before the message can be safely exported to the LBS provider. In a transformed message, $X : [x_s, x_e]$ denotes the extent of the spatiotemporal cloaking box of the transformed message on the x-axis, with x_s and x_e denoting the two end points of the interval. The definitions of $Y : [y_s, y_e]$ and $I : [t_s, t_e]$ are similar with y-axis and t-axis replacing the x-axis, respectively. We denote the spatio-temporal cloaking box of a transformed message as $B_{cn}(m_t)$ and define it as $(m_t.X, m_t.Y, m_t.I)$. The field C in m_t denotes the message content. We describe how the fields of a transformed message in set T relates to its counterpart in set S, in the following subsection.

k-anonymity Constraints

The following basic characteristics must be kept in mind a raw message m_s in S and its transformed format m_t in T:

- **Spatial Containment:**

$$m_s.x \in m_t.X . m_s.y \in m_t.Y$$

- **Spatial Resolution:**

$$m_t.X \subset \Phi(m_s.x, m_s.d_x)$$
$$m_t.Y \subset \Phi(m_s.y, m_s.d_y)$$

- **Temporal Containment:**

$$m_s.t \in m_t.I$$

- **Temporal Resolution:**

$$m_t.I \subset \Phi(m_s.t, m_s.d_t)$$

- **Content Preservation**

$$m_s.C = m_t.C$$

Spatial containment and containment needs simply indicate when the message hidden box has become, $B_{cl}(m_t)$, should contain the spatiotemporal point $P(m_s)$ of the original message m_s. The spatial resolution and temporal resolution requirements indicate that, for each of the three dimensions, the amount of latency-time hidden message box must be within the range defined by the tolerance value specified in the original message. This is equivalent to saying that the converted message hidden box should be in the main message restriction box, i.e. $B_{cl}(m_t) \subset B_{cn}(m_s)$.

Content retention is a trivial feature, which ensures that the content of the message as it stands, is copied from the original message to the converted message. We officially record the essence of K-anonymity with the following requirement, which states that, for a message m_s in S and its transformed format m_t in T, the following condition must hold:

- **Location-based k-anonymity**

$$\exists T' \subset T, s.t. m_t \in T', |T'| \geq m_s.k$$
$$\forall \{m_{t_i}, m_{t_j}\} \subset T', m_{t_i}.u_{id} \neq m_{t_j}.u_{id}$$
$$\forall m_{t_i} \in T', B_{cl}(m_{t_i}) = B_{cl}(m_t)$$

Requires the k-anonymity requirement for each converted message m_t, there has to be at least m_s.k − 1 other transformed messages with the same spatiotemporal cloaking box, each from a different mobile node. A key challenge for the spatiotemporal cloaking algorithm is to find a set of messages within a minimal spatiotemporal cloaking box that satisfies the above conditions.

Evaluation Metrics

An important measure of success is to evaluate the effectiveness of the k-anonymization pattern. Specifically, the main purpose of the encryption algorithm is to maximize the number of successful messages in accordance with the k-anonymity constraints of the location. In other words, we want to maximize | T |. Success rates can be defined as the percentage of messages that are successfully anonymized (transformed), i.e. 100 * |T|/|S|. Other important measures of efficiency include the level of relative anonymity, relative temporal resolution, relative resolution, and message processing time. The first three are measures related to quality of service, while the last is a performance measure.

The relative anonymity level is the measure of the degree of anonymity generated by the hidden algorithm and normalized by the level of anonymity required for messages**.** We define relative anonymity level over a set of transformed messages $T' \subset T$ as:

$$\frac{1}{|T'|} \sum_{m_t = R(m_s) \in T'} \frac{|\{m | m \in T \wedge B_{cl}(m_t) = B_{cl}(m)\}|}{m_s.k}$$

Note that relative anonymity level cannot go below 1.

Spatial relative resolution is a measure of spatial resolution provided by a secrecy algorithm that has been normalized by the minimum acceptable spatial resolution defined by spatial tolerances. We define relative spatial resolution over a set of transformed messages $T' \subset T$ as:

$$\frac{1}{|T'|}\sum_{m_t=R(m_s)\in T'}\sqrt{\frac{2*m_s.d_x*2*m_s.d_y}{\|m_t.X\|*\|m_t.Y\|}}$$

where $\|l\|$, when applied to an interval l, gives its length. Higher relative resolution values indicate more effective secrecy achieved with a smaller spatial clay region.

Relative Time Resolution is a measure of the temporal resolution provided by a secrecy algorithm that is defined with ordinary acceptable minimum time resolution defined by ordinary time tolerances. We define relative temporal resolution over a set of transformed messages $T' \subset T$ as:

$$\frac{1}{|T'|}\sum_{m_t=R(m_s)\in T'}\frac{2*m_s.d_t}{\|m_t.I\|}$$

Higher relative resolution values indicate more effective secrecy, which results in a shorter time interval and, consequently, less delay. Relative spatial and temporal resolutions cannot reach below 1.

The message processing time is a measure of the performance of the execution time of the caching algorithm. If computing power is available to handle high-speed incoming messages, message processing time may become an issue. We use the average processor time required to process 10^3 messages as the message processing time.

Data Structures

We briefly describe the four main data structures that are used in our algorithm.

• Message Queue, Q_m : Message queue is a simple FIFO queue, which collects the messages sent from the mobile nodes in the order they are received. Messages from this queue are displayed by the algorithm to be processed.

•Multi-dimensional Index I_m : Multidimensional index is used to efficiently search for locations and locations of messages. For each message, say m_s, in the set of messages that are not yet anonymized and are not yet dropped according to expiration condition (specified by the temporal tolerance), I_m contains a three dimensional point $P(m_s)$ as key, together with the message m_s as data. The index is implemented using an in-memory $R^* - tree$ in our system.

• Constraint Graph, G_m :The constraint graph is a dynamic graphical memory that contains messages that have not yet been anonymised and have not yet fallen due to expiration. The multi-dimensional index I_m is mainly used to speed up the maintenance of the constraint graph G_m, which is updated when new messages arrive or when messages get anonymized or expired.

• Expiration Heap, H_m :The mass expiration is a medium stack, sorted by the deadline of the messages. For each message, say m_s, in the set of messages that are not yet anonymized and are not yet dropped due to expiration, H_m contains a deadline $m_s.t + m_s.d_t$ as key, together with the message m_s as $data^*$. Mass expiration is used to detect expired messages (that is, messages that cannot be successfully unspecified), so that they can be logged out and eliminated.

Algorithm Details

We assume that cellular nodes will not send new messages unless their previous messages are anonymized or explicitly deleted due to anonymous hidden algorithm expiration. The pseudo-code of this algorithm is given in Algorithm 1. The algorithm works by sequencing messages from the queue and processing them to be k-anonymous in four steps.

The first step is to update the data structures with the new message and integrate the new message into the constraint graph. When a message, m_s is popped from the queue, it is inserted into the index I_m using $P(m_{s_c})$, inserted into the heap H_m using $m_{s_c}.t + m_{s_c}.d_t$ and inserted into the graph G_m as a node. Then the edges with vertex m_{s_c} are constructed in the constraint graph G_m by searching the multi-dimensional index I_m using the spatiotemporal constraint box of the message, i.e. $B_{cn}(m_{s_c})$, as the range search condition. The messages whose spatiotemporal points are contained in $B_{cn}(m_{s_c})$ are candidates for being m_{s_c}'s neighbors in the constraint graph. These messages (referred to as N pseudo-N code) are filtered based on whether their space-time constraint boxes contain a filter $P(m_{s_c})$. The ones that pass the filtering step (excluding m_{s_c} itself) become neighbors of m_{s_c} in the constraint graph. See lines 7-16 in the pseudo code.

The second step is to apply the local-k search algorithm to find a clique in the constraint graph. In local-k search, we try to find a

clique of size $m_{s_c}.k$ that includes the message m_{s_c}. The pseudo-code of this step is presented separately in Algorithm 2 as a function of local-k search. Note that the local-k search function is called algorithm 1 (line 18) with parameter k set to $m_{s_c}.k$. Before beginning the search, a set $U \subset nbr\left(m_{s_c}\right)$ is constructed such that for each element $m_s \in U$, we have $m_{s_c}.k \le k$ (line 4). This means that the neighbors of m_{s_c} whose anonymity values are higher than k are simply discarded from U, as they cannot be anonymized with a clique of size k. Once this is done, the set U is iteratively filtered until there is no change (lines 5-13). At each filtering step, each message $m_s \in U$ is checked whether it has at least k−2 neighbors in U. If not, the message cannot be part of a clique that contains m_{s_c} and has size k. After the set U is filtered, the possible cliques in $U \cup \left\{m_{s_c}\right\}$ that contain m_{s_c} and have size k are enumerated and if one satisfying the k-anonymity requirements is found, the messages in that clique are returned. Although the general problem of finding cliques in a graph is NP-Complete, up to values of k = 10, (where k = 5 is considered as a good level of anonymity [10]) the search step does not form a bottleneck.

The third step is to generate k-annonized messages that are sent to external LBS providers. If clicked, the messages in the click (labeled as M in pseudo-code) are anonymized by assigning MBR from the spatio-temporal points of the messages in the clique, $B_m(M)$ as their cloaking box. Then they are removed from the graph Gm, as well as from the index I_m and the heap H_m. This step is detailed in the pseudo code through lines 19-27. If you find a clique, the message stays inside the system. It may be collected and anonymous later when processing a new message or may be lost due to expiration. In the following sections, we discuss more advanced methods for searching for clues.

The fourth step is cleaning the expiration stack. After processing each message, we examine the expiration stack for each expired message. The message at the top of the stack expires and is checked if it expires. Such a message cannot be anonymous and is excluded. This step is repeated until the message reaches the deadline before the current time. Lines 28-37 are pseudo-code related to message expiration.

Algorithm 1 Clique-Cloak Algorithm

1: {G_m is the constraint graph}
2: {Q_m is the queue of incoming messages}
3: {I_m is the index on spatio-temporal points of messages}
4: {H_m is the min-heap consisting of message deadlines}
5: **while** TRUE **do**
6: **if** $Q_m \neq \emptyset$ **then**
7: $m_{s_c} \leftarrow$ Pop the first item in Q_m
8: Add m_{s_c} into I_m with $P(m_{s_c})$
9: Add m_{s_c} into H_m with $(m_{s_c}.t + m_{s_c}.d_t)$
10: Add the message m_{s_c} into G_m as a node
11: $N \leftarrow$ Range search I_m using $B_{cn}(m_{s_c})$
12: **for all** $m_s \in N, m_s \neq m_{s_c}$ **do**
13: **if** $P(m_{s_c}) \in B_{cn}(m_s)$ **then**
14: Add the edge (m_{s_c}, m_s) into G_m
15: **end if**
16: **end for**
17: { Find set of messages M in G_m s.t. $m_{s_c} \in M$, $|M| = m_{s_c}.k$, $\forall_{m_s \in M}, m_s.k \le |M|$, and M forms a clique in G_m }
18: $M \leftarrow$ local-k_Search$(m_{s_c}.k, m_{s_c}, G_m)$
19: **if** $M \neq \emptyset$ **then**
20: **for all** m_s in M **do**
21: Output transformed message $m_t \leftarrow \langle m_s.u_{id}, m_s.r_{no}, B_m(M), m_s.C\rangle$
22: Remove the message m_s from G_m
23: Remove the message m_s from I_m
24: Pop the topmost element in H_m
25: **end for**
26: **end if**
27: **end if**
28: **while** TRUE **do**
29: $m_s \leftarrow$ Topmost item in H_m
30: **if** $m_s.t + m_s.d_t < now$ **then**
31: Remove the message m_s from G_m
32: Remove the message m_s from I_m
33: Pop the topmost element in H_m
34: **else**
35: **break**
36: **end if**
37: **end while**
38: **end while**

Algorithm 2 local-k_Search(k, m_{s_c}, G_m)

1: $U \leftarrow \{m_s | m_s \in nbr(m_{s_c})$ and $m_s.k \le k\}$
2: **if** $|U| < k - 1$ **then**
3: **return** $\emptyset$
4: **end if**
5: $l \leftarrow 0$
6: **while** $l \neq |U|$ **do**
7: $l \leftarrow |U|$
8: **for all** $m_s \in U$ **do**
9: **if** $(|nbr(m_s) \cap U| < k - 2)$ **then**
10: $U \leftarrow U \backslash \{m_s\}$
11: **end if**
12: **end for**
13: **end while**
14: Find any subset $M \subset U$, s.t. $|M| = k-1$ and $M \cup \{m_{s_c}\}$ forms a clique
15: **return** M

$Nbr-k$ Search

When searching for a clique in the constraint graph, it is essential to ensure that the newly received message, say m_{s_c}, should be included in the clique. If there is a new clique formed due to the entrance of m_{s_c} into the graph, it must contain m_{s_c}. However,

instead of searching a clique with size $m_{s_c}.k$, we can try to find out the biggest clique that includes $m_{s_c}.k$, of course making sure that all messages inside the clique has a k value at most equal to the size of the clique. There are two strong motivations behind the approach. First, by anonymizing a larger number of messages at once, it can provide higher success rate which also results in better performance, as the graph will become less crowded. Second, by anonymizing messages that have smaller k's together with messages that have larger k's, it can provide higher relative level of anonymity. $Nbr-k$ search takes the latter approach. Its pseudo code is given in Algorithm 3 as the $Nbr-k$ Search function.

Algorithm 3 nbr-k_Search(m_{s_c}, G_m)

```
1: if |nbr(m_sc)| < k − 1 then
2:    return ∅
3: end if
4: L ← {m_s.k | m_s = m_sc ∨ m_s ∈ nbr(m_sc)}
5: for all distinct k ∈ L in decreasing order do
6:    if k < m_sc.k then
7:       return ∅
8:    end if
9:    M ← local-k_Search(k, m_sc, G_m)
10:   if M ≠ ∅ then
11:      return M
12:   end if
13: end for
14: return ∅
```

$Nbr-k$ search first collects the set of k values the new message m_{s_c} and its neighbors $nbr\left(m_{s_c}\right)$ have, denoted as L in the pseudo code. The k values in L are considered in decreasing order until a clique is found or k becomes smaller than $m_{s_c}.k$ (in which case the search returns empty set). For each k ∈ L considered, a clique of size k is searched by calling the local k Search function with appropriate parameters (see line 9). If such a clique can be found, the messages within the clique are returned. To integrate $Nbr-k$ search into the Clique Cloak algorithm, we can simply replace line 18 of the Algorithm 1 with the call to $Nbr-k$ Search function.

Experimental and Results

Mobile Database

We have created a mobile data tracking generator that simulates moving cars on the road and makes use of situational information from simulation requests. Each machine generates several messages during the simulation. Each message using a zipf parameter of 0.6 indicates the anonymous level (k value) of the list {5, 4, 3, 2}, k = 5 being the most popular. The spatial and temporal tolerance values of the messages are selected independently using the normal distributions whose default parameters are listed in Table 1. Whenever a message is generated, the message originator waits for the message to be anonymized or reduced, then waits for the normal value of the distributed time, called wait time, which is also specified by the default parameters in Table 1. All parameters take their default values, unless otherwise stated. We change many of these parameters to observe the behavior of the algorithms in different settings.

Table 1. Message generation parameters

Parameter	Default value
anonymity level range	{5,4,3,2}
anonymity level zipf param	0.6
mean spatial tolerance	100 m
variance in spatial tolerance	40 m^2
mean temporal tolerance	30 s
variance in temporal tolerance	$12m^2$
mean inter-wait time	15 s
variance in inter-wait time	$6s^2$

Table 2: Car movement parameters

mean of car speeds for each road type	$\{90, 60, 50\} km/h$
std.dev. in car speeds for each road type	$\{20, 15, 10\} km/h$
traffic volume data	$\{2916.6, 916.6, 250\}$ *per hour*

Implementation of the anonymization phase

The functions that are implemented for anonymization are briefly explained below.

- **Read Data**: This function reads the data of each cluster separately from the file.
- **Set Attribute Type:** This function specifies the type of each attribute. Each attribute can have Insensitive, Sensitive, Identifying and Quasi-identifyin values. This function is an ARX library function.
- **Set Hierarchy:** This function specifies how to generalize records. This function is an ARX library function.
- **Set K-anonymity:** Using this function, K-anonymity parameters are specified for anonymity. This function is an ARX library function.
- **Set L-diversity:** Using this function, L-diversity parameters are identified for anonymity. This function is an ARX library function.
- **Set T-closeness:** This function determines the parameters of the T-closeness for anonymity. This function is an ARX library function.
- **Anonymize Dataset:** Anonymizes the dataset by receiving the parameters k, l and t. This function is an ARX library function.

Introducing evaluation criteria

- **Diversion ratio**

Consider the value of an attribute in a record that is not publicized. There was no distortion in this case. However, if the value of an attribute in a record is expanded to a more general value in the classification tree (hierarchical extension), then the degree of distortion on which the attribute is made is proportional to the degree of generalization on that attribute. For example, if the update value in the classification tree is close to the root of the node, the distortion rate is greater: thus, the degree of distortion of the attribute values depends on how high the generalization tree is. For example, the value for which the generalization is not performed is at the zero height of the tree. If only one level is generalized, it falls to the height of the generalization tree.

Here h_{ij} is the height of the generalized value of the attribute A_i from t_j. To calculate the rate of distortion of the whole dataset, we calculate the sum of the total distortion of the generalized values: which $distortion = \sum_{ij} h_{ij}$ for calculating of diversion ratio, Distortion of developed database should be divided by the Distortion of the same set where all data values are fully generalized to the highest level. In other words, they are generalized at the root level.

- **Execution Time**

Another criterion used is the runtime of the algorithm. The runtime is the time it takes for an algorithm to run correctly and without error so that the results are visible at the end of the algorithm.

- **Predicting accuracy**

Another criterion used is precision. The published data is used for various purposes including knowledge extraction, prediction, and so on. Using this criterion the extent to which these goals will be achieved with the selected features will be examined. In order to clarify the problem, the confusion matrix will be expressed first.

Table 3: Confusion matrix

		Predicted	
		Positive	Negative
Actual	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

The most important criterion for determining the efficiency of an accuracy classification algorithm is the classification rate 1, which calculates the accuracy of an entire classification category. In fact, this criterion is the most popular and most general criterion for calculating the efficiency of classification algorithms, which indicates that the designed cluster correctly classifies a few percent of the entire set of experimental records.

Classification accuracy is (CA) obtained using the following equation, which states that the two values of TP and TN are the most important values that should be maximized in a binary problem. (In multi-batch problems, the values on the original diameter of this matrix, which are deducted from the calculation of CA, should be maximal.)

$$CA = \frac{TP + TN}{TP + FN + TN + FP}$$

Results

The proposed method is evaluated with runtime criteria, deviation ratio and prediction accuracy. The following results are followed.

- **Execution time**

The average runtime of the $k-nbr$ algorithm and the proposed algorithm were calculated and measured five times and the results are shown in the following figures. The diagrams in the vertical axis indicate the runtime in seconds, and the horizontal diagram shows the number of pseudo- attitude. Each graph has different values of k that are listed below each graph but the values are l = 2 and t = 2.

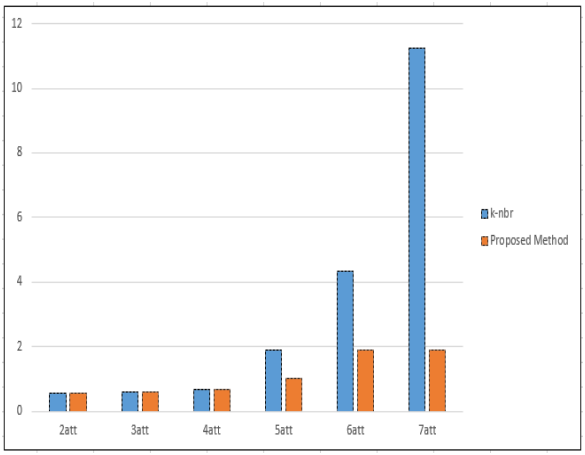

Figure 1: Comparison chart of execution time with K=2

As can be seen in Fig. 1, both algorithms behave similarly when the number of features is low, and with the increase in the number of features, the execution time of the proposed method does not change much. But this increase in features in the k-nbr model also leads to a significant increase in runtime.

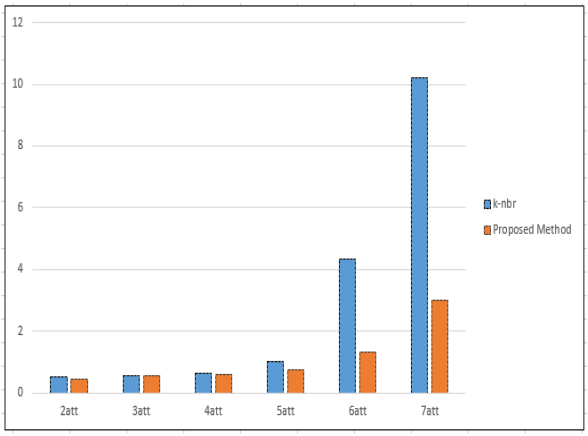

Figure 2: Comparison chart of execution time with K=5

Fig.2 shows the run time of the proposed method and the $k-nbr$ with the number of features between 2 and 7 and k = 5. Fig.3 is executed with k = 10 and its time is measured.

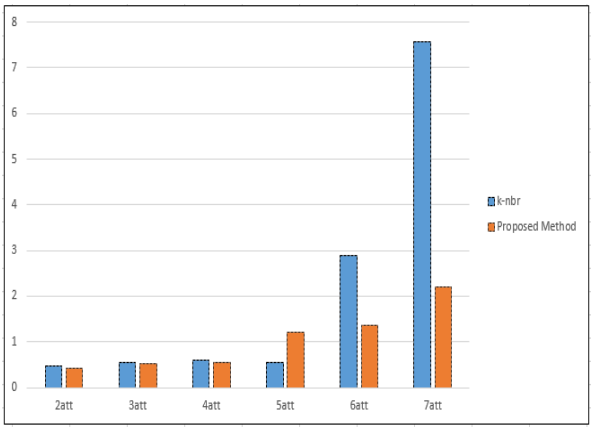

Figure 3: Comparison chart of execution time with K=10

Regarding execution time graphs, it can be concluded that increasing the number of features and increasing the value of K are effective on execution time, but overall the shape and behavior of the graphs do not change.

- **Diversion ratio**

The amount of deviation ratio was compared between the two k-nbr privacy models and the proposed method. The results of the experiment can be seen in the following figures. The values of L and T are set to 2.

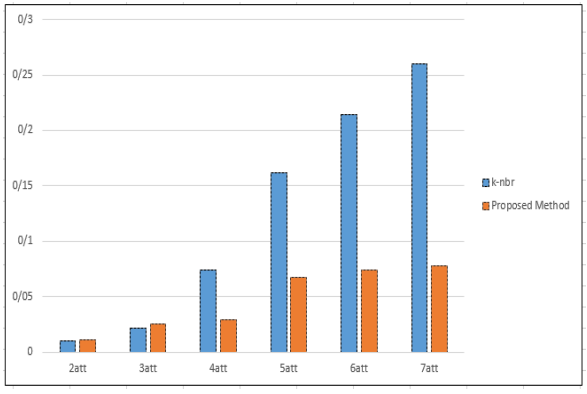

Figure 4: Comparison chart of deviation ratio with K = 2

In the above fig., the deviation ratio with the number of features is seen from 2 to 7. The vertical axis specifies the deviation ratio and the horizontal axis to the number of features. In the k-nbr model, due to weighting of features and deletion of various features by priority, the end features are most similar. For this reason, in Fig.4 we see that the data set with two properties and the application of the k-nbr algorithm has a lower deviation ratio. However, with the addition of more features, the proposed model is much better and has less information loss than the k-nbr model.

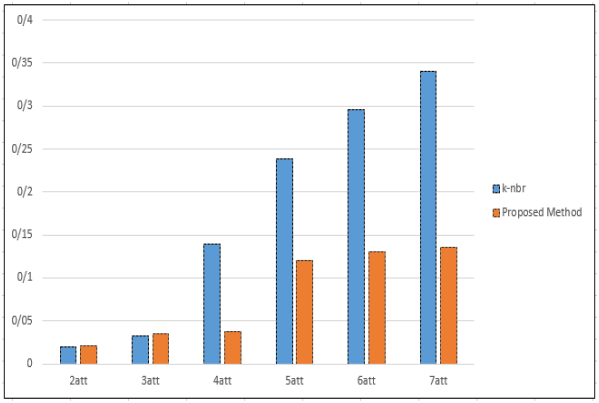

Figure 5: Comparison chart of deviation ratio with K = 5

Figure 5 shows the increase in data loss compared to the previous graph by increasing K. But we also see behavior similar to the previous chart. The same is evident in Fig.6. In Fig.6, the value of k is equal to 10, and again we see an increase in data loss. The reason for this can be stated that by increasing the value of k, the data becomes more secure against disclosure attacks and as a result more data will be lost. Increasing k is directly related to increasing the level of security of the output tables.

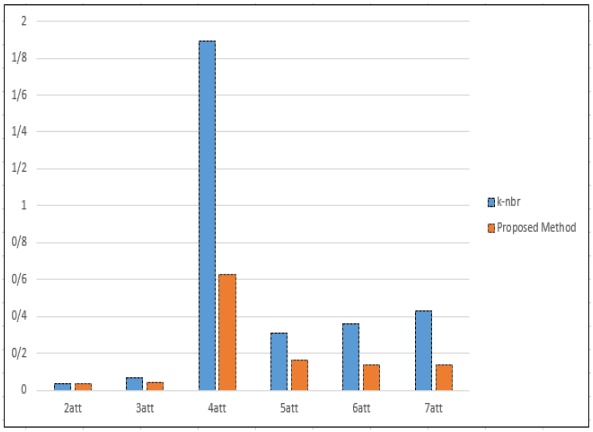

Figure 6: Comparison chart of deviation ratio with K = 10

- **Predicting accuracy**

The last criterion to be evaluated is the accuracy criterion. By using this criterion, the utility of the selected features is measured. As you can see in Fig. 7, the prediction accuracy using the selected features in the proposed model is higher than the k-nbr model, and also the prediction accuracy increases with increasing number of features.

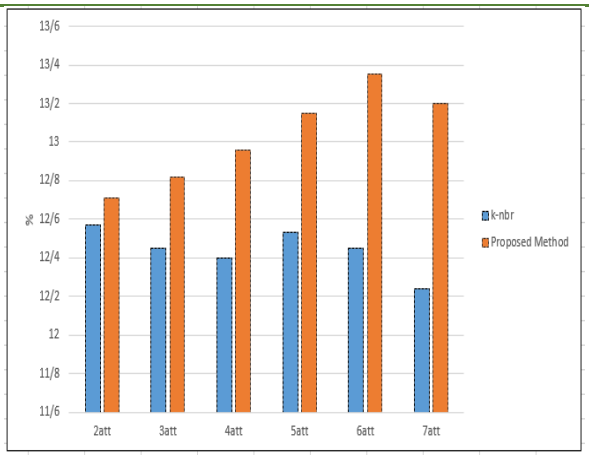

Figure 7: Comparison of prediction accuracy

At the end of the experiment we will review and summarize the results. Initially, the proposed method was evaluated in terms of execution time, which had superiority and less time than the similar model. It should be noted that with increasing k value, data security will be increased in terms of privacy, which in the k-nbr model will be increased computational load. Then the deviation ratio was investigated and the results showed that the proposed algorithm had a significant decrease in the amount of information loss compared to the k-nbr model. Finally, using the prediction accuracy criterion, the objective function prediction was evaluated based on the selected features that the proposed method has superiority over each aspect. In the proposed algorithm set, a new approach on how to select features for anonymization and dissemination of data is proposed which can have a major impact on the privacy of data for dissemination in the future.

Conclusion

We propose a customizable K-anonymity model to provide location privacy. Our model has two unique features. First, it allows each cellular node to be defined in the most expensive single messages, at least by its anonymity level, as well as by a high degree of inaccuracy by temporal and spatial cipher algorithms. Second, it implements this model using a clustered-space-time hidden ramp algorithm, Clique Cloak, which can match the messages sent by mobile nodes to the k-location of the location while the messages are anonymous and accurate. This study presents strategies to increase privacy and safety of spreadsheet data sets. The issue of privacy refers to the care of the personal information held by the recipient of the information and it can be said that the request is how personal data collected for social purposes should be maintained and used by the organization that collected it. In pursuit of these goals, there were a number of ways that challenged this issue in two ways. The first challenge refers to the level of privacy of individuals using the proposed solutions; and the second challenge concerns the extent to which information is lost after the proposed solutions are implemented. According to the aforementioned, the purpose of this study is to develop and reduce the amount of information loss in a three layer algorithm. Finally, by using the features of feature selection to select pseudo-identities, the weighting stage of the features from the three-layer algorithm was completely followed by the partial preprocessing and anonymization step, which greatly improved the three-layer algorithm. This solution reduces the amount of data loss and runtime and also increases prediction accuracy. In the proposed method, firstly, pseudo-attributes were selected using the feature selection method. Then clustering was performed on selected features and finally anonymization operation was performed on clusters and confidential data set was generated. After presenting the proposed solution, the mobile dataset was introduced along with the criteria for deviation ratio, execution and prediction accuracy. Providing information and predictive accuracy has improved significantly. The following describes the innovations and achievements of the proposed method:

- Use attribute selection operations for pseudo- attributes
- Use class to select features
- Use front-facing selection to select features
- Reduce execution time
- Reduce data loss
- Increase the prediction accuracy

The results of this study are very useful and useful for statistical organizations, medical centers, insurance, etc., which deal with the publication of data sets and porotecting the privacy of customers and their accountants will have a huge impact.

References

1. Ouri Wolfson, "Mobile Database", Encyclopedia of Database Systems, Part 13, 2009; 1751. http://www.springerlink.com/content/n72wu51n4056524g/fulltext.html, Springer Science+Business Media, LLC 2009, 10.1007/978-0-387-39940-9_1362.
2. Kumar V. Mobile database systems. Wiley-Interscience; 2006 Aug 25.
3. Xia Y, Helal A. A dynamic data/currency protocol for mobile database design and reconfiguration. InProceedings of the 2003 ACM symposium on Applied computing 2003 Mar 9 (pp. 550-556).
4. Wang X, Pande A, Zhu J, Mohapatra P. STAMP: Enabling privacy-preserving location proofs for mobile users. IEEE/ACM transactions on networking. 2016 Jan 18;24(6):3276-89.
5. Liu Q, Wang G, Li F, Yang S, Wu J. Preserving privacy with probabilistic indistinguishability in weighted social networks. IEEE Transactions on Parallel and Distributed Systems. 2016 Oct 4;28(5):1417-29.
6. Jiang W, Wang G, Bhuiyan MZ, Wu J. Understanding graph-based trust evaluation in online social networks: Methodologies and challenges. ACM Computing Surveys (CSUR). 2016 May 23;49(1):1-35.
7. Zhang S, Wang G, Liu Q, Abawajy JH. A trajectory privacy-preserving scheme based on query exchange in mobile social networks. Soft Computing. 2018 Sep 1;22(18):6121-33.

8. Wang T, Zhou J, Huang M, Bhuiyan MZ, Liu A, Xu W, Xie M. Fog-based storage technology to fight with cyber threat. Future Generation Computer Systems. 2018 Jun 1;83:208-18.
9. Sun Y, Chen M, Hu L, Qian Y, Hassan MM. ASA: Against statistical attacks for privacy-aware users in Location Based Service. Future Generation Computer Systems. 2017 May 1;70:48-58.
10. Qi L, Zhang X, Dou W, Hu C, Yang C, Chen J. A two-stage locality-sensitive hashing based approach for privacy-preserving mobile service recommendation in cross-platform edge environment. Future Generation Computer Systems. 2018 Nov 1;88:636-43.
11. Gedik B, Liu L. Protecting location privacy with personalized k-anonymity: Architecture and algorithms. IEEE Transactions on Mobile Computing. 2007 Nov 21;7(1):1-8.
12. Schlegel R, Chow CY, Huang Q, Wong DS. User-defined privacy grid system for continuous location-based services. IEEE Transactions on Mobile Computing. 2015 Jan 7;14(10):2158-72.
13. Zhang S, Liu Q, Lin Y. Anonymizing popularity in online social networks with full utility. Future Generation Computer Systems. 2017 Jul 1;72:227-38.
14. Zhang Y, Tong W, Zhong S. On designing satisfaction-ratio-aware truthful incentive mechanisms for $ k $-anonymity location privacy. IEEE Transactions on Information Forensics and Security. 2016 Jul 7;11(11):2528-41.
15. Wang L, Yang D, Han X, Wang T, Zhang D, Ma X. Location privacy-preserving task allocation for mobile crowdsensing with differential geo-obfuscation. InProceedings of the 26th International Conference on World Wide Web 2017 Apr 3 (pp. 627-636).
16. Giannotti F, Nanni M, Pinelli F, Pedreschi D. Trajectory pattern mining. InProceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining 2007 Aug 12 (pp. 330-339).
17. ElSalamouny E, Gambs S. Differential privacy models for location-based services. Transactions on Data Privacy, 2016; 9(1): 15–48.
18. You TH, Peng WC, Lee WC. Protecting moving trajectories with dummies. In2007 International Conference on Mobile Data Management 2007 May 1 (pp. 278-282). IEEE.
19. Lee JG, Han J, Li X, Cheng H. Mining discriminative patterns for classifying trajectories on road networks. IEEE Transactions on Knowledge and Data Engineering. 2010 Sep 2;23(5):713-26.
20. Chen R, Fung BC, Desai BC, Sossou NM. Differentially private transit data publication: a case study on the montreal transportation system. InProceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining 2012 Aug 12 (pp. 213-221).
21. Hua J, Gao Y, Zhong S. Differentially private publication of general time-serial trajectory data. In2015 IEEE Conference on Computer Communications (INFOCOM) 2015 Apr 26 (pp. 549-557). IEEE.
22. Huo Z, Meng X, Hu H, Huang Y. You can walk alone: trajectory privacy-preserving through significant stays protection. InInternational conference on database systems for advanced applications 2012 Apr 15 (pp. 351-366). Springer, Berlin, Heidelberg.
23. Xu F, Tu Z, Li Y, Zhang P, Fu X, Jin D. Trajectory recovery from ash: User privacy is not preserved in aggregated mobility data. InProceedings of the 26th international conference on world wide web 2017 Apr 3 (pp. 1241-1250).
24. Han S, Bae HJ, Kim J, Shin S, Choi SE, Lee SH, Kwon S, Park W. Lithographically encoded polymer microtaggant using high-capacity and error-correctable QR code for anti-counterfeiting of drugs. Advanced Materials. 2012 Nov 20;24(44):5924-9.
25. Kim Y, Han J, Yuan C. Toptrac: Topical trajectory pattern mining. InProceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 2015 Aug 10 (pp. 587-596).
26. Sweeney L. k-anonymity: A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems. 2002 Oct;10(05):557-70.
27. Machanavajjhala A, Gehrke J, Kifer D, Venkitasubramaniam M. Venkitasubramaniam, l-diversity: Privacy beyond k-anonymity, in: IEEE ICDE 2006, art. 24, 2006.
28. Li N, Li T, Venkatasubramanian S. t-closeness: Privacy beyond k-anonymity and l-diversity. In2007 IEEE 23rd International Conference on Data Engineering 2007 Apr 15 (pp. 106-115). IEEE.
29. Lee B, Oh J, Yu H, Kim J. Protecting location privacy using location semantics. InProceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining 2011 Aug 21 (pp. 1289-1297).
30. Dwork C. Differential privacy: A survey of results. InInternational conference on theory and applications of models of computation 2008 Apr 25 (pp. 1-19). Springer, Berlin, Heidelberg.
31. Li T, Li N. On the tradeoff between privacy and utility in data publishing. InProceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining 2009 Jun 28 (pp. 517-526).
32. Parliament and of the European Union, Regulation on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (Data Protection Directive), Official Journal of the European Union, 2011; L119: 1–88.
33. Liu L. From data privacy to location privacy: models and algorithms. InProceedings of the 33rd international conference on Very large data bases 2007 Sep 23 (pp. 1429-1430).

34. Nergiz ME, Atzori M, Saygin Y. Towards trajectory anonymization: a generalization-based approach. InProceedings of the SIGSPATIAL ACM GIS 2008 International Workshop on Security and Privacy in GIS and LBS 2008 Nov 4 (pp. 52-61).
35. Ma CY, Yau DK, Yip NK, Rao NS. Privacy vulnerability of published anonymous mobility traces. InProceedings of the sixteenth annual international conference on Mobile computing and networking 2010 Sep 20 (pp. 185-196).
36. Terrovitis M, Mamoulis N. Privacy preservation in the publication of trajectories. InThe Ninth International Conference on Mobile Data Management (mdm 2008) 2008 Apr 27 (pp. 65-72). IEEE.
37. Huo Z, Meng X, Hu H, Huang Y. You can walk alone: trajectory privacy-preserving through significant stays protection. InInternational conference on database systems for advanced applications 2012 Apr 15 (pp. 351-366). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-29038-1_26.
38. You, T. H., Peng, W. C., Lee, W. C. Protecting moving trajectories with dummies. In Proceedings of the international workshop on privacy-aware location based mobile services, 2007.
39. Richter KF, Schmid F, Laube P. Semantic trajectory compression: Representing urban movement in a nutshell. Journal of Spatial Information Science. 2012(4):3-0. https://doi. org/10.5311/josis.2012.4.62.
40. Ying JJ, Lee WC, Weng TC, Tseng VS. Semantic trajectory mining for location prediction. InProceedings of the 19th ACM SIGSPATIAL international conference on advances in geographic information systems 2011 Nov 1 (pp. 34-43). https://doi.org/10.1145/2093973.2093980.
41. Elragal A, El-Gendy N. Trajectory data mining: integrating semantics. Journal of Enterprise Information Management. 2013 Sep 20;26(5):516-35. https://doi.org/10.1108/JEIM-07-2013-0038.
42. Monreale A, Pinelli F, Trasarti R, Giannotti F. Wherenext: a location predictor on trajectory pattern mining. InProceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining 2009 Jun 28 (pp. 637-646).
43. Lee WC, Krumm J. Trajectory preprocessing. InComputing with spatial trajectories 2011 (pp. 3-33). Springer, New York, NY. https://doi.org/10.1007/978-1-4614-1629-6_1.
44. Gao H, Chu D, Duan Y, Yin Y. Probabilistic model checking-based service selection method for business process modeling. International Journal of Software Engineering and Knowledge Engineering. 2017 Aug;27(06):897-923.
45. Giannotti F, Nanni M, Pinelli F, Pedreschi D. Trajectory pattern mining. InProceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining 2007 Aug 12 (pp. 330-339).
46. Sweeney L. k-anonymity: A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems. 2002 Oct;10(05):557-70.
47. Gruteser M, Grunwald D. Anonymous usage of location-based services through spatial and temporal cloaking. InProceedings of the 1st international conference on Mobile systems, applications and services 2003 May 5 (pp. 31-42).