Science & Technology Development Journal – Engineering and Technology, 3(SI1):SI82-SI94

Open Access Full Text Article

Research Article

Privacy preserving spatio-temporal databases based on k-anonymity

Anh Truong*

ABSTRACT

The development of location-based services and mobile devices has led to an increase in the location data. Through the data mining process, some valuable information can be discovered from location data. In the other words, an attacker may also extract some private (sensitive) information of the user through the data mining process and this may make threats against the user privacy. For example, the attacker can mine user's location data for deciding the home address of the user. Thus, location privacy protection becomes an important requirement to the success in the development of location-based services. In this paper, we propose a grid-based approach as well as an algorithm to guarantee k-anonymity, a well-known privacy protection approach, in a location database. To do this, we assume that the service server will provide services but in a defined area and the grid will cover the area in which the service server takes effect, Then, the user's location will be hidden in an anonymization area. The anonymization area will be chosen by cells that forms a rectangle area so that this area contains at least k distinct users. Moreover, in practice, the location of a user usually accompanies with a temporal data. And, indeed, the information about the combination of spatial and temporal data may also disclose some other sensitive information of the user. Thus, the paper also proposes an approach for guaranteeing k-anonymity for the combination of spatial and temporal database. The proposed approach considers only the information that has significance for the data mining process while ignoring the un-related information. Finally, the experiment results show the effectiveness of the proposed approach in comparison with the literature ones.

Key words: Location Privacy, Privacy Preserving, data mining, k-anonymity, spatio-temporal databases

Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology, VNU-HCM, Vietnam

Correspondence

Anh Truong, Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology, VNU-HCM, Vietnam

Email: anhtt@hcmut.edu.vn

History

- Received: 29-7-2019
- Accepted: 25-8-2019
- Published: 04-12-2020

DOI : 10.32508/stdjet.v3iSI1.517

Check for updates

INTRODUCTION

Today, advances in location technologies and wireless communication technologies enable the widespread development of location-based services (LBSs) [1]. When using the service, the user may face with risks because the location of the user can disclose some private information. For example, the attacker can keep track of information in each time the user uses the service. From there he can find the area where the user uses the service more frequently. Thus, it is necessary to protect the location information of the user from attacker [2–5].

The user's location privacy should be protected in two stages [6–8]. In the first stage, the location privacy should be protected at the time of using services. One popular method is to obfuscate the location with the service provider in order to hide the user's location information [9,10]. The solution focuses on preventing the user's location from an illegal observation at the time of service calls. We proposed an approach to hide the user's location in [11,12]. In the next stage, the location privacy should be protected at the time when the user's location information is stored in the database for data mining purposes. With this stage, the location information should be hidden before such data are shared to organizations or companies.

In this paper, we focus on protecting the user's location information when such information are stored in the database. It is assumed that when a user uses a location based service, he will provide his real location to the service server and the server will save such location information. Then, some organizations, companies or individuals will collect such location data. By using the data mining process [13,14], they can obtain some valuable information. Because such location information maybe disclose some user's privacy. For example, the attacker can queries the database to get some results, then, he can also link some priori knowledge with the results to get some sensitive information. Therefore, such location data should be protected before they are collected by organizations. Fortunately, some techniques protecting user privacy have been proposed such as k-anonymity, cryptography… [1]. Among them, k-anonymity [15] is one of the most important methods for privacy protection. The

Cite this article : Truong A. **Privacy preserving spatio-temporal databases based on k-anonymity**. *Sci. Tech. Dev. J. – Engineering and Technology;* 3(SI1):SI82-SI94.

main idea behind k-anonymity is that some attributes in data can often be considered as pseudo-identifiers to uniquely identify the records [16]. Thus, such attributes should be also protected.

This paper proposes an technique to anonymize the user's spatio-temporal data to achieve k-anonymity. This approach will use a grid and anonymize the user's location to an anonymization rectangle. The grid must cover all space where the server provides services. This approach considers also the data mining process by finding the area where has more users using the service. The paper also presents an algorithm to connect the location attribute and time attribute in a spatio-temporal database to achieve k-anonymity.

The rest of this paper is organized as follows. In section 2, we briefly summarize related works. Section 3 presents our approach to anonymize the user's location. Next, section 4 introduces an algorithm to connect the location attribute and time attribute to achieve k-anonymity. Finally, section 5 shows some conclusions.

RESEARCH METHODS

k-anonymity

K-anonymity is an approach that protects data from individual identification [17]. Intuitively, k-anonymity states that data must be anonymized in a way such that every combination of values of released attributes can be indistinctly matched to at least k respondents [17]. Recently, some approaches have been introduced to ensure privacy protection when releasing data mining results [17]. With these approaches, we first define the set of attributes, called *Quasi-Identifiers* (QI), whose values can be used, possibly together with external information, to re-identify the real data. For example, data about sex, the ZIP code, date of birth may not explicitly identify an individual but such data cab be linked to external information to obtain name, address and city. Basically, the greater the value of k, we can get the better the protection of privacy. However, if the data is anonymized too much (that means the value of k is too big). This leads to the case that data quality for data mining process is not good. Therefore, we need the balance between data privacy and data quality and it is considered as an important factor in privacy preserving in data mining. We consider an algorithm not only to anonymize the location data but also considering the result of data mining process in this paper.

k-Anonymity in Data Mining

There are two possible approaches to guarantee k-anonymity in data mining [17]:

- Anonymize the original table and perform mining on its k-anonymous version.
- Perform mining on the original table and anonymize the result. This approach can be performed by executing the two steps independently or in combination.

The first approach gives two benefits. First, it guarantees that data mining is safe because data mining is executed on a k-anonymized version of original table. Second, it allows data mining to be executed by others than the data holder, enables different data mining processes and different uses of the data. This is convenient, for example, when the data holder may not know a prior how the recipient may analyze and classify the data. With the second approach, Data mining can then be performed by the data holder only, and only the sanitized data mining results are released to other parties. This may therefore affect applicability [17].

The paper will use the first approach to guarantee k-anonymity. This allows data mining to be executed by many organizations, or companies to obtain suitable results.

GRID-BASED APPROACH TO GUARANTEE K-ANONYMITY FOR LOCATION DATA

When a user uses location services, he must provide his real location to the service server that provides LBSs and this information will be saved in the service server database. After that, attackers collect the user's location information; he can find some sensitive information about the user. For example, an attacker may query the database and a result, which has just one tuple, is returned, he also has knowledge that at this location, there is just one user who is using the service. Therefore, he can decide that this tuple is for this user and can find some private information of this user. Clearly, the linking between the user's location and external knowledge can reveal the private information of the user [18,19].

To protect the information about the user from linking information, k-anonimity technique has been proposed. With this approach, the location of the user is indistinguishable from k-1 other locations. Therefore, the attacker can not distinguish the tuple which actually contains the user's location with other tuples. In this section, we will propose a grid approach to hide the true location of the user. We assume that the service server will provide services but in a defined area and the grid will cover the area in which the service server takes effect. First, we will have some definitions.

Definitions

Definition: A Grid *G* is a uniform grid which contains cells, a cell is not necessary a square-shaped, it can be in a rectangle-shaped but all cells must cover the whole space. The grid has a starting point. Figure 1 a shows a grid with a starting point S.

Definition: An anonymization area includes cells and contains location of some users. Figure 1 b presents an area with a user U.

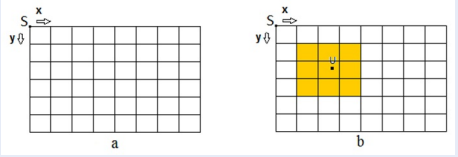

Figure 1: Grid (a) and Anonymization area (b).

The user's location will be hidden in an anonymization area. We will create an anonymization area by choosing cells to form a rectangle area so that this area contains at least k distinct users. Figure 2 shows an anonymization area with three users.

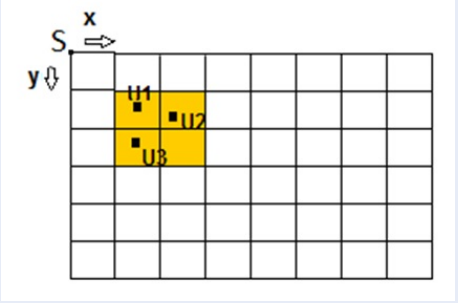

Figure 2: 3-anonymization area.

Moreover, through the data mining process, the data miner usually desires to find areas which have more users using the service. Therefore, the anonymization area should be in well-proportioned shaped. We will consider the example in Figure 3, to obtain 3-anonymity, anonymization area A1 and A2 are acceptable. However, anonymization A1 is better in this example.

Definition: A cell is defined as a pair *(x, y)* where *x* is the order number of this cell in x direction and *y* is the order number in y direction. For example, in Figure 4, cell A is defined as (2, 1).

Definition: The distance of two cells *(x1, y1)* and *(x2, y2)* in direction x is defined as $|x1 - x2|$. Similarly, the

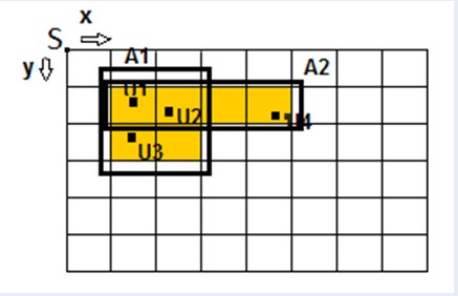

Figure 3: Better anonymization area

distance of two cells in direction y is defined as $|y1 - y2|$. We call the distance of two cell in direction x is *dis_x* and in direction y is *dis_y*.

Dis[(x1, y1), (x2, y2)] = (dis_x, dis_y)

with $\begin{cases} dis\_x = |x1 - x2| \\ dis\_y = |y1 - y2| \end{cases}$

Definition: An anonymization rectangle is defined as *[(x1, y1), (x2, y2)]* where *(x1, y1)* and *(x2, y2)* are the left-top cell and right-bottom cell of cells in the rectangle. For example, the colored rectangle in Figure 4 b is defined as *[(2, 1), (4, 2)]*.

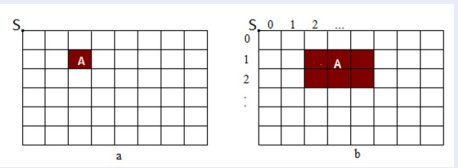

Figure 4: Cell definition (a) and anonymization rectangle definitiion (b).

Normally, data miners usually desire to find an area that the number of the user in it is maximal and this area is usually not too big. Therefore, we also provide thresholds to limit the area. These thresholds are also the limitation that the data miners want to limit the area containing users. Figure 5 shows the maximum area which has threshold *tx* and *ty*.

With our approach, the following rule is considered:

Area A the number of users using the service in A

A is an anonymization rectangle. Our approach will find areas and the number of users in these areas. We want to guarantee k-anonymity in the database; therefore, the number of users in any area must be greater k. Moreover, these areas must smaller the maximum area.

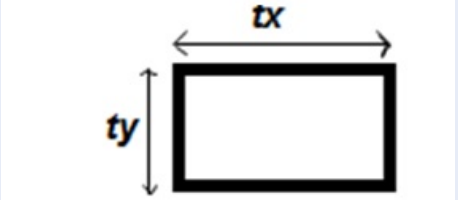

Figure 5: Maximum area with threshold tx and ty

Definition: We have an anonymization rectangle A, threshold tx and ty.

$$\text{A is safe if } \begin{cases} dis\_x \Leftarrow tx \\ dis\_y \Leftarrow ty \end{cases}$$

dis_x is the distance-x of any two cells in A and dis_y is the distance-y of any two cells in A.

Algorithm

With k-anonimity approach, attributes whose values can be used, possibly together with external information, to re-identify the data, will be included in *Quasi-Identifiers* (*QI*). Because the attacker can use the location attribute and link to external knowledge in order to find some sensitive information of the user, the location attribute need to be included into *QI*. The *QI* will be in the following format:

$QI = \{Q_1, \ldots, Q_n, L\}$. L is the location attribute.

We assume that the location attribute does not depend on other attribute of *QI*. Therefore, we can anonymize *QI* through two stages. The first stage is to anonymize $Q_1, \ldots Q_n$ and the second stage is to anonymize the location attribute L. Our approach will focus on the second stage and not care the first stage.

In our approach, we use a grid to anonymize the location of the user. Therefore, we will anonymize the user's location to grid cell at the first step. As discussed before, our approach will consider the rule:

Area A the number of users using the service in A

The algorithm will find all areas that is smaller the maximum area (the area which is defined by thresholds *tx* and *ty*). We will choose areas that have the most users in it. It means that users use the service more frequently in these areas. Certainly, we only choose the area which is safe.

In some cases, areas, which are not safe, are returned. In these cases, additional steps will be operated. We notice that the variable k, which is provided, is usually smaller the minimum number of users in an area, which the data miner desire to receive. Therefore, if the number of users in an area is smaller k, we consider this area is not signification to the data mining process. In these cases, we will anonymize the location of users in this area to the closet anonymized area which is safe.

The algorithm for guaranteeing k-anonymity in location database is described as follows:

Input: k, threshold *tx*, threshold *ty*, location table T

Output: k-anonymization location table T'

Method:

Create a gird which covers the space where the server provides services.

Anonymize all location data of tuples in T to grid cell.

While (exist a tuple which has not been marked)

{

For each tuple in T **and** this tuple has not been marked

{

Find the safe anonymization area and the number of distinct users in this anonymization area is maximum.

}

From the set of anonymization areas has just found, we will choose the area in which the number of distinct users is maximal. We call this area as maximal anonymization area.

If (the number of distinct users in this maximal anonymization area < k)

{

Anonymize all location data of tuples in this area to closet anonymization area which is safe and mark the corresponding tuple in T.

}

Else

{

Anonymize location attribute of users, which belong to this maximal anonymization area, to this area and mark the corresponding tuple in the table T.

}

}

Return anonymized table.

In our algorithm, we ignore the case when the number of distinct users in the maximal anonymization area is smaller k at the first loop. The reason for this had been mentioned above.

To explain the algorithm, we will consider an example: We have a location table with attributes **No.**, **ID**, **Location** and other data as in Table 1. Location attribute is a pair *(a, b)* which describes the true location of the user in x and y direction. Threshold *tx* is 2 (cells), *ty* is 2 (cells) and k is 3.

At the first step, we will anonymize all location data of tuples to grid cell. The grid cell size is 100*100 and the result is in Figure 6.

Table 1: A location table

No.	ID	Location	Data
1	u1	(156, 150)	…
2	u2	(460, 263)	…
3	u1	(335, 158)	…
4	u5	(448, 192)	…
5	u7	(295, 191)	…
6	u8	(388, 284)	…
7	u6	(229, 365)	…
8	u4	(572, 189)	…
9	u5	(649, 118)	…
10	u3	(320, 225)	…
11	u9	(240, 224)	…
12	u11	(127, 358)	…
13	u10	(164, 167)	…

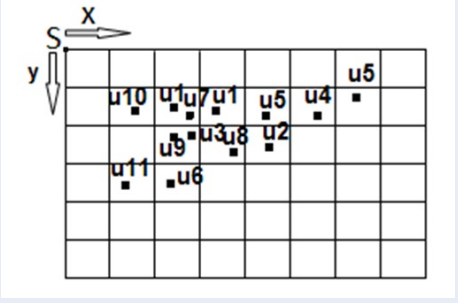

Figure 6: Tuple anonymization to grid cell

At the first loop, we will find the safe anonymization area, in which the number of distinct users is maximum, for each location data of tuple. For example, tuple *No. 9* has two anonymization areas which satisfy the condition, Two areas are described in Figure 7 a. We can choose one among two these areas. The process is similar to other tuples. Finally, the maximal anonymization area is described in Figure 7 b. We choose this area because the number of distinct users in this area is maximal.

Because the number of users in this maximal anonymization area is greater k (value of k is 3). We will anonymize all location of users, which belong to this area and mark the corresponding tuple. In this case, location attribute of tuples *No. 1, 3, 5, 6, 10* and *11* will be anonymized to anonymization rectangle *[(2, 1), (3, 2)]*.

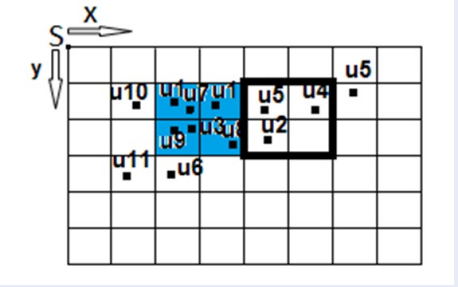

Figure 8: Second loop: maximal anonymization area

At the second loop, the maximal anonymization area in Figure 8 is chosen. Location attribute of tuples *No. 2, 8* and *9* will be anonymized to anonymization rectangle *[(4, 1), (5, 2)]*.

At the third loop, the maximal anonymization area in Figure 9 is chosen. However, the number of distinct users in this area is 2 and this value is smaller k. Therefore, all users in this area will be "moved" to the "closest" anonymization area which is safe. In this case, the maximal anonymization area that was found in the first loop is chosen. Therefore, location attribute of tuples *No. 7* and *12* will be anonymized to anonymization rectangle *[(2, 1), (3, 2)]*.

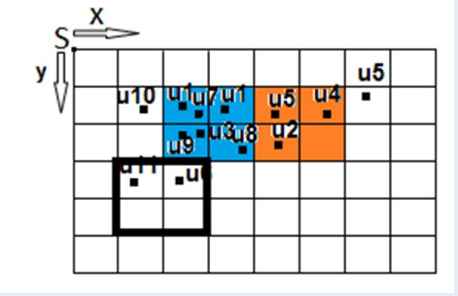

Figure 9: Third loop: maximal anonymization area

Similarity, next loops will be processed in the same way. Finally, we will have Table 2, which satisfies 3-anonymity:

K-ANONYMITY FOR SPATIO-TEMPORAL DATABASES

Discussion

In practice, the location of a user usually accompanies with a temporal data[20,21]. For example, the user A was in location "U_1" and used the service at "March 10, 2010". The information about spatio-temporal

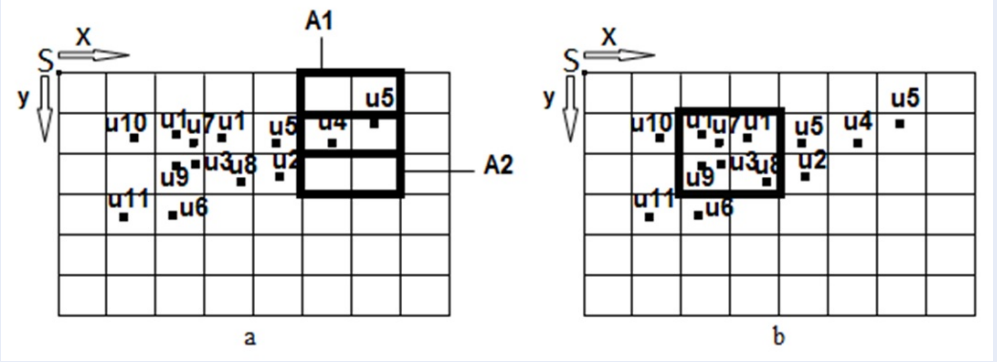

Figure 7: First loop: anonymization for tuple No. 9 (a) and maximal anonymization area (b)

Table 2: 3-anonymity table

No.	ID	Location	Data
1	u1	[(2, 1), (3, 2)]	…
2	u2	[(4, 1), (5, 2)]	…
3	u1	[(2, 1), (3, 2)]	…
4	u5	[(4, 1), (5, 2)]	…
5	u7	[(2, 1), (3, 2)]	…
6	u8	[(2, 1), (3, 2)]	…
7	u6	[(2, 1), (3, 2)]	…
8	u4	[(4, 1), (5, 2)]	…
9	u5	[(4, 1), (5, 2)]	…
10	u3	[(2, 1), (3, 2)]	…
11	u9	[(2, 1), (3, 2)]	…
12	u11	[(2, 1), (3, 2)]	…
13	u10	[(2, 1), (3, 2)]	…

data maybe also disclose some user's sensitive information. In case he also has some knowledge that there is just one user, who used the service at that time and location, he will find that all tuples in the result will belong to a user. Therefore, time attribute also need to be included into *QI*. The structure of *QI* will be:

$QI = \{Q_1, \ldots, Q_n, L, T\}$.

L is location attribute and *T* is time attribute.

The process, which anonymizes this *QI* to guarantee k-anonymity, is also similar to the process proposed in section 3. First, we will anonymize Q_1, Q_2… Q_n attributes. After that, *L* and *T* will be anonymized. We notice that we need to protect both spatial and temporal data. In the previous section, we introduced an approach to anonymize the location of the user. Therefore, we also need an approach to anonymize the temporal data.

Authors extend the notion of k-anonymity in the context of databases with timestamped information in order to naturally define k-anonymous views of temporal data[22]. With this approach, time attributes can be generalized to the most common ones, as year, month or week. For example, Table 3 is the original table. We can generalize time attributes in this table to "*week*". We assume that we have 52 weeks in a year. The value "*2009-01-03*" will be generalized to value "*week 1*". We notice that all time data have the same year. Therefore, the value "*week 1*" also means that this is week 1 of year 2009. Similarly, the value "*2009-01-12*" will be generalized to "*week 2*". Table 4 is a 2-anonymous version of the original table.

Table 3: Original table with time attributes

	QI		Data
	Q	T	
u1	q1	2009-01-03	d0
u2	q1	2009-01-03	d1
u1	q1	2009-01-11	d2
u4	q1	2009-01-12	d3
u5	q2	2009-02-07	d4
u6	q2	2009-02-10	d5

Table 4: 2-anonymous table

UID	QI		Data
	Q	T	
u1	q1	2009-week 1	d0
u2	q1	2009-week 1	d1
u1	q1	2009-week 2	d2
u4	q1	2009-week 2	d3
u5	q2	2009-week 6	d4
u6	q2	2009-week 6	d5

We can also generalize time attributes in original table to "month". Value "*2009-01-03*" will be generalized to value "*2009-month 1*". Similarly, value "*2009-02-07*" will be generalized to "*2009-month 2*". Table 5 is another 2-anonymous version of the original table. For more details, refer to [22].

In order to apply k-anonymity to the database with spatio-temporal data, we notice that the location of a user accompanies with a temporal data. Therefore, when we anonymize the location of the user, we also consider the time attribute, which accompanies with the location attribute. We also notice that the data mining process result may not be significant or data is unusable if we generalize the time attribute too much. We will consider the example in Figure 7. User *u1, u3, u7, u8* and *u9* are in the maximal anonymization area. We add time values to these tuples in the original table as follows:

According to the example in section 3.2, these tuples will be anonymized to the anonymization rectangle *[(2, 1), (3, 2)]*. Therefore, time attributes of these tuples are also generalized to the most common ones. In this case, time attributes will be anonymized to 2009. This interval is too long if the data miner wants statistics in each month. To avoid this case, we can also set a threshold to time attribute. Moreover, we also need to anonymize the time attribute at the same time with the location attribute. We will consider the example in Table 6. The time threshold is 2 months, threshold *tx* is 2 (cells), *ty* is 2 (cells) and *k* is 2. The algorithm in section 3.2 will find all anonymization areas which are safe and choose the maximal anonymization area in which the number of distinct users is maximal. The result is in Figure 7 b. However, as discussed above, this result does not satisfy the time constraint. Therefore, the new algorithm to guarantee k-anonymity in the spatio-temporal database must choose another anonymization area which satisfies both time and location constraints. Figure 10 shows an acceptable result which satisfies all constraints. Location attributes of tuples *No. 3, 5* will be anonymized to *[(2, 0), (3, 1)]* and time attributes will be generalized to *"2009-month 8_9"*. The result is in Figure 10 a. Similarly, location attributes of tuples *No. 6, 10* will be anonymized to *[(2, 2), (3, 3)]* and time attributes will be generalized to *"2009-month 11_12"* as in Figure 10 b. Tuple *No. 1* will be "moved" to maximal anonymization area in Figure 10 b while tupe *No.11* will be "moved" to maximal area in Figure 10 a.

In this paper, we will generalize the time attribute into intervals, which are in the formation *[a, b]*. Value *a* is the lowest and values *b* is the biggest of time value. For example, we will consider the following table:

Table 5: Another 2-anonymous table

UID	QI		Data
	Q	T	
u1	q1	2009-month 1	d0
u2	q1	2009-month 1	d1
u1	q1	2009-month 1	d2
u4	q1	2009-month 1	d3
u5	q2	2009-month 2	d4
u6	q2	2009-month 2	d5

Table 6: A spatio-temporal table

No.	ID	Location	Time	Data
1	u1	(156, 150)	2009-01-03	…
3	u1	(335, 158)	2009-07-09	…
5	u7	(295, 191)	2009-08-12	…
6	u8	(388, 284)	2009-11-07	…
10	u3	(320, 225)	2009-12-25	…
11	u9	(240, 224)	2009-06-19	…

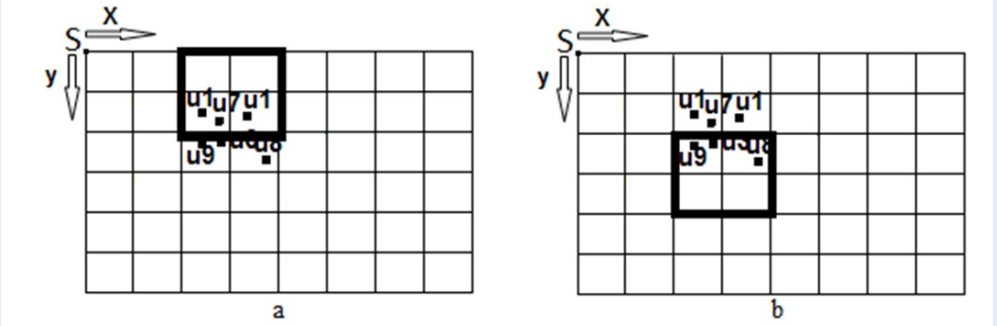

Figure 10: Maximal anonymization areas for spatio-temporal data

Table 7: Time attribute generalization

No.	QI		Data
	Q	T	
1	q1	2010-01-12	d0
2	q1	2010-02-25	d1
3	q1	2010-01-16	d2
4	q1	2010-02-08	d3

With our approach, these time attributes in table 7 will be generalized to *[2010-01-12, 2010-02-25]*.

Similar to the location attribute, which data miners usually desire to find an area that the number of the user in it is maximal and this area is usually not too big, with the time attribute, we also set a threshold *t_time*. Therefore, all intervals, which are the result of generalization, will be smaller than this time threshold. This threshold is also the limitation that the data miners want to limit time intervals. It also means that the data miner will mine the valuable information from the data but time intervals are not bigger than the time threshold.

As defined before, anonymization rectangle *A* is safe if the distance of any two cells in both direction x and y is smaller than thresholds *tx* and *ty*. However, in a spatio-temporal database, the time attribute also accompanies with the location attribute. Therefore, we will redefine the safe anonymization rectangle *A* as follows:

Definition: The distance of two time values, *a* and *b*, is $|a - b|$. We call this subtraction as time-distance(a, b).

Definition: We have an anonymization rectangle *A*, threshold *tx* and *ty* and t_{time}.

$$\text{A is safe if } \begin{cases} dis\_x \Leftarrow tx \\ dis\_y \Leftarrow ty \\ dis\_time \Leftarrow t\_time \end{cases}$$

dis_x is the distance-x of any two cells in *A*, *dis_y* is the distance-y of any two cells in *A*, dis_time is the time-distance of the time value of any two tuples which have location attribute value in A.

Clearly, a trade off between location anonymization and time generalization is very important to achieve k-anonymity in a spatio-temporal database. In the next section, we will introduce an effective algorithm to anonymize location and time attributes in spatio-temporal database in order to guaranteeing k-anonymity.

Algorithm

As discussed before, the structure of *QI* will be:

$QI = \{Q_1, \ldots, Q_n, L, T\}$.

L is location attribute and *T* is time attribute.

To guarantee k-anonymity in database with this structure of *QI*, we can anonymize *QI* through two stages. The first stage is to anonymize $Q_1, \ldots Q_n$ and the second stage is to anonymize the location attribute *L* and *T*. Again, we will not care the first stage and focus on the second stage.

In our algorithm, we will anonymize the location and time value of tuples in the original table to corresponding safe anonymization rectangles and intervals. The algorithm will choose the areas in such a way that the results of the anonymization should be significant to data mining. These areas are also where users use the service more frequently. With other areas which are not good to the data mining, the algorithm will "move" them to closet significant area.

At the first step, we build a grid and then hide the user's location data to the grid. With our approach, we want to find all areas that satisfy all thresholds and the number of distinct users, who uses the service in these areas, is maximal. Therefore, we will find all safe anonymization areas for the time and location value of each tuple, which has been not anonymized, and choose the safe area that the number of distinct users in this anonymization area is maximum. We call this area as *tuple_maximal_anonymization_area*. In our algorithm, the find_safe_max_anoymization() function will be responsible for this work. After the previous step, we get a set *X*, which contains all *tuple_maximal_anonymization_area* for each tuple. From this set, we will choose the tuple_maximal_anonymization_area which has the number of distinct users in this area is maximum. We call this area as *maximal_anonymization_area*. Finally, we will anonymize all the location data and time data of tuples, which belong to this *maximal_anonymization_area*, to corresponding value, namely *maximal_anonymization_area* for location data and an interval for time data. Approaches for anonymizing the location data and time data are discussed before.

Name: k-anonymization Algorithm()

Input: *k*, threshold *tx*, threshold *ty*, threshold *t_time*, spatio-temporal table *T*

Output: k-anonymization location table *T'*

Method:

Create a gird *G* which covers the space where the server provides services.

Anonymize all location data of tuples in *T* to grid cell.

X = Ø

While (exist a tuple which has not been marked)

{

For each tuple in *T* **and** this tupe has not been marked

{

tuple_maximal_anonymization_area = find_safe_max_anoymization();

X = *X* U tuple_maximal_anonymization_area;

}

Maximal_anonymization_area = choose the tuple_maximal_anonymization_area which has the

number of distinct users in this area is maximum from X.

If (the number of distinct users in this maximal anonymization area < *k*)

{

Anonymize all location and time data of tuples in this area to closet anonymization area which is safe.

Mark the corresponding tuple in the table T.

}

Else

{

Anonymize location attribute of users, which belong to this maximal anonymization area, to this area and generalize all time data of these tuples to an interval.

Mark the corresponding tuple in the table T.

}

}

Return anonymized table.

When the number of distinct users in the maximal anonymization area is smaller than *k*. we will consider this area is not significant to the data mining process. Therefore, we will anonymize all location and time data of tuples in this area to "closet" safe anonymization area. We discussed this idea in section 3.2.

The *find_safe_max_anoymization()* function will choose the safe area for the time and location data of each tuple that the number of distinct users in this anonymization area is maximum. At the first step, this function will find all safe areas according to the time and location value which is input parameter. Among them, it will choose the safe area that the number of distinct users in this anonymization area is biggest.

Name: find_safe_max_anoymization()

Input: a tuple *t* contains time and location data, threshold *tx*, threshold *ty*, threshold *t_time*, spatio-temporal table *T*, Grid *G*

Output: tuple_maximal_anonymization_area for tuple t and a set *O* which contain tuples belong to tuple_maximal_anonymization_area

Method:

Arrange *T* in order to tupes *t1, t2, t3* … in *T* will satisfy *t1.time* < *t2.time* < *t3.time*

Give set *Y* = all anonymization area which contain *t.location* and satisfy both thresholds *tx* and *ty*

Variable *count_max* = 0

tuple_maximal_anonymization_area = null;

For each *y* in *Y*

{

Variable *count_y* = 0

R = all tuple *i* in *T* and *i.location* belongs to *y*

For each *t'* in *R* **and** index of *t'* in *R* <= the index of *t*

{

if time-distance(*t'.time, t.time)* <= *t_time*

{

tuple *tp*

For each tuple *tr* in *R* **and** index of *tp* in *R* >= the index of *t*

{

tp = tr;

if time-distance(*tp.time, t'.time)* > *t_time*

Exit For

}

if *index_of_tp_in_R – index_of_t'_in_R – 1 > count_y*

{

count_y = index_of_tp_in_R – index_of_t'_in_R – 1

Give set *O* = all tuples in *R* from *(index_of_t'_in_R)* to *(index_of_tp_in_R* – 1)

}

}

If (*count_y > count_max*)

{

count_max = count_y

tuple_maximal_anonymization_area = y

Remember set *O*

}

}

Return *tuple_maximal_anonymization_area*, *O*

This function will return the *tuple_maximal_anonymization_area* and a set *O* contains tuples which satisfy location and time constraints. *tuple_maximal_anonymization_area* always accompanies with a set *O*. Therefore, when this area is chosen as *maximal_anonymization_area*, all tuples in *O* will be anonymized.

EXPERIMENTS

We show the experiment for the evaluation of the effectiveness of proposed approach. With our tests, the data mining process wants to find the time interval, when users use the service more frequently. The data mining process will work with the original table and k-anonymous table version, which generated by our algorithm. We will compare these two results by getting the overlapped interval between two results. Clearly, the proposed approach is effectiveness if this overlapped interval is large. We will use a ratio to describe this effectiveness:

$$R_{time} = \frac{overlapped\_t}{original\_result\_t}$$

original_result_t is the result which the data mining process works with original table. *overlapped_t* is the overlapped interval between the results of original table and our k-anonymous version. As discussed, the larger the ratio, the more effective the approach.

We will evaluate the approach with a spatio-temporal table with more than 2000 records. The number of distinct users is more than 50. The grid cell size and k are changed in each test case. In each case, we will change the maximum area and maximum time interval which data miner desires to be processed from data mining process. The maximum area is *max area* column and the maximum time interval is *max interval* column in the Table 8.

In Table 8, k will be 5, 10, and 20 for each test. The ratio value is the average of three tests when k is 5, 10 and 20. The result shows that in most case the ratio is larger than 80%. It also means that our approach will generate a k-anonymous version of the original table in which the data mining process can find significant information as when working in original table.

CONCLUSIONS AND DISCUSIONS

In this paper, we propose an technique for anonymizing the spatio-temporal database. With this technique, we can anonymize the location and time data easily. We also consider the data mining process result to develop an algorithm, which tradeoffs between data privacy and data quality.

In the future, we will focus on improving the algorithm in order to guarantee k-anonymity in a big spatio-temporal database more efficiency.

ACKNOWLEDGMENT

This research is funded by Ho Chi Minh City University of Technology, Vietnam National University HoChiMinh City under grant number T-KHMT-2018-90.

CONFLICT OF INTEREST

We claim that there is no conflict of interest in this article.

AUTHOR CONTRIBUTION

Anh Truong is the only author of this article.

REFERENCES

1. Ciriani V, Vimercati SDC, Foresti S, Samarati P. k-Anonymous Data Mining: A Survey. Handbook of Database Security - Applications and Trends ISBN 978-0-387-70991-8, Springer Science and Business Media, LLC. 2008;p. 105–136. Available from: https://doi.org/10.1007/978-0-387-70992-5_5.
2. Samarati P, Sweeney L. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical Report SRI-CSL-98-04, Computer Science Laboratory, SRI International. 1998;.
3. Dang TK, Truong AT. Anonymizing but Deteriorating Location Databases. In the International Research Journal of Computer Science and Computer Engineering with Applications, IPN Mexico. 2012;(46):73–81. Available from: https://doi.org/10.17562/PB-46-9.
4. Bettini C, Mascetti S, Wang XS. Privacy Protection through Anonymity in Location-based Services. Michael, G, Sushil, J (eds), Handbook of Database Security - Applications and Trends Springer. 2008;p. 509–530. Available from: https://doi.org/10.1007/978-0-387-48533-1_21.
5. Cuellar JR. Location Information Privacy. B. Srikaya (Ed.). Geographic Location in the Internet Kluwer Academic Publishers. 2002;p. 179–208. Available from: https://doi.org/10.1007/0-306-47573-1_8.
6. Gedik B, Liu L. Protecting Location Privacy with Personalized k-Anonymity: Architecture and Algorithms. IEEE Transactions on Mobile Computing. 2008;7(1):1–18. Available from: https://doi.org/10.1109/TMC.2007.1062.
7. Bugra G, Ling L. Protecting Location Privacy with Personalized k-Anonymity: Architecture and Algorithms. IEEE Transaction on mobile computing. 2008;.
8. Gidófalvi G, Huang X, Pedersen TB. Privacy-Preserving Data Mining on Moving Object Trajectories. 8th International Conference on Mobile Data Management. 2007;Available from: https://doi.org/10.1109/MDM.2007.18.
9. Vinh C, Truong AT, Tran T. A Privacy Preserving Authentication Scheme in the Intelligent Transportation Systems. 5th International Conference on Future Data and Security Engineering, Ho Chi Minh - Viet Nam. 2018;.
10. Tran T, Truong AT, Vinh C. An Authentication Scheme to Preserve Users Privacy in Intelligent Transportation Systems, SEATUC, Yogyakarta - Indonesia. 2018;.
11. Truong TA, Truong QC, Dang TK. An Adaptive Grid-based Approach to Location Privacy Preservation. Proc of 2nd Asian Conference on Intelligent Information and Database Systems (ACIIDS 2010), Hue City, Vietnam. 2010;.
12. Truong QC, Truong TA, Dang TK. Privacy Preserving through A Memorizing Algorithm in Location-Based Services. Proc of the 7th International Conference on Advances in Mobile Computing and Multimedia (MoMM2009), Kuala Lumpur, Malaysia. 2009;Available from: https://doi.org/10.1145/1821748.1821780.
13. Beresford AR, Stajano F. Mix zones: User privacy in location-aware services. 2nd IEEE Annual Conference on Pervasive Computing and Communications Workshops. 2004;.
14. Bettini C, Wang X, Jajodia S. Protecting privacy against location-based personal identification. 2nd VLDB Workshop on Secure Data Management. 2005;Available from: https://doi.org/10.1007/11552338_13.
15. Sweeney L. Achieving k-anonymity privacy protection using generalization and suppression. International Journal on Uncertainty, Fuzziness and Knowledge-based Systems. 2002;10(5):571–588. Available from: https://doi.org/10.1142/S021848850200165X.
16. Phan TN, Dang TK, Dang AT, Lam TH. A Context-aware Privacy-preserving Solution for Location-based Services. 2018 International Conference on Advanced Computing and Applications, Hochiminh - Viet Nam. 2018;PMID: 30134661. Available from: https://doi.org/10.1109/ACOMP.2018.00028.
17. Truong TA, Dang TK, Kueng J. On Guaranteeing k-Anonymity in Location Databases. In Proc of the 22nd International Conference on Database and Expert Systems Applications (DEXA'11), pages 280-287, LNCS, Springer. 2011;Available from: https://doi.org/10.1007/978-3-642-23088-2_20.
18. Myles G, Friday A, Davies N. Preserving Privacy in Environments with Location-Based Applications. IEEE Pervasive Computing. 2003;p. 56–64. Available from: https://doi.org/10.1109/MPRV.2003.1186726.
19. Ardagna CA, Cremonini M, Vimercati SDC, Samarati P. Privacy-enhanced Location-based Access Control. Michael, G, Sushil, J (eds), Handbook of Database Security - Applications and Trends Springer. 2008;p. 531–552. Available from: https://doi.org/10.1007/978-0-387-48533-1_22.
20. Marco G, Xuan L. Protecting Privacy in Continuous Location - Tracking Applications. IEEE Computer Society. 2004;.
21. Panos K, Gabriel G, Kyriakos M, Dimitris P. Preventing Location-Based Identity Inference in Anonymous Spatial Queries. IEEE Transactions on Knowledge and Data Engineering. 2007;.

Table 8: Results of experiment

tx*ty (m*m)	t_time (days)	k	Cell size (m)	max area (m*m)	max interval (days)	Ratio Rtime (%)
90*90	30	5,10,20	30	200*200	120	88.97
90*90	30	5,10,20	30	400*400	150	87.53
90*90	60	5,10,20	30	200*200	120	88.31
90*90	60	5,10,20	30	400*400	150	86.42
150*150	30	5,10,20	50	300*300	120	84.62
150*150	30	5,10,20	50	500*500	150	85.02
150*150	60	5,10,20	50	300*300	120	84.19
150*150	60	5,10,20	50	500*500	150	83.97
300*300	30	5,10,20	100	500*500	120	81.74
300*300	30	5,10,20	100	800*800	150	80.36
300*300	60	5,10,20	100	500*500	120	79.92
300*300	60	5,10,20	100	800*800	150	80.09

22. Mascetti S, Bettini C, Wang XS, Jajodia S. k-anonymity in databases with timestamped data. Proc of 13th International Symposium on Temporal Representation and Reasoning, IEEE Computer Society. 2006;Available from: https://doi.org/10.1109/TIME.2006.20.

Tạp chí Phát triển Khoa học và Công nghệ – Kĩ thuật và Công nghệ, 3(SI1):SI82-SI94

Open Access Full Text Article

Bài Nghiên cứu

Bảo vệ tính riêng tư cơ sở dữ liệu không-thời gian dựa trên k-anonymity

Trương Tuấn Anh*

Use your smartphone to scan this QR code and download this article

TÓM TẮT

Sự phát triển của các dịch vụ dựa trên vị trí và các thiết bị di động đã dẫn đến việc sinh ra các dữ liệu vị trí. Thông qua quá trình khai phá dữ liệu, các thông tin có ích sẽ được khai thác từ dữ liệu vị trí này. Điều này cũng đồng nghĩa với việc kẻ tấn công có thể lợi dụng để rút trích các thông tin riêng tư của người sử dụng từ các dữ liệu này. Ví dụ, kẻ tấn công có thể xem thông tin vị trí của người dùng để xác định địa chỉ nhà của họ. Bởi vậy, việc bảo vệ thông tin vị trí trở thành một yêu cầu quan trọng. Trong bài báo này, chúng tôi giới thiệu hướng tiếp cận dùng lưới tương thích cũng như một giải thuật để đảm ảo k-anonymity cho các cơ sở dữ liệu vị trí. Để làm điều này, chúng tôi giả thiết rằng các dịch vụ vị trí sẽ cung cấp dịch vụ trong một vùng không gian định trước và một lưới tương thích sẽ được tạo ra trong vùng này. Sau đó, vị trí của người sử dụng sẽ được ẩn danh trong một vùng ẩn danh. Các vùng ẩn danh này sẽ được lựa chọn theo nguyên tắc là có ít nhất k người sử dụng trong vùng ẩn danh. Chúng tôi cũng đề xuất hướng tiếp cận để đảm bảo k-anonymity cho dữ liệu kết hợp cả không và thời gian. Hướng tiếp cận được đề xuất sẽ chỉ xem xét các thông tin có ý nghĩa cho quá trình khai phá dữ liệu trong khi bỏ qua các thông tin không liên quan khác. Cuối cùng, các kết quả thực nghiệm chỉ ra sự hiệu quả của giải pháp đề xuất khi so sánh với các giải pháp khác.

Từ khoá: Tính riêng tư vị trí, Bảo vệ tính riêng tư, khai phá dữ liệu, k-anonymity, cơ sở dữ liệu không-thời gian.

Khoa Khoa học và Kỹ thuật Máy tính, trường Đại học Bách Khoa-ĐHQG-HCM, Việt Nam

Liên hệ

Trương Tuấn Anh, Khoa Khoa học và Kỹ thuật Máy tính, trường Đại học Bách Khoa-ĐHQG-HCM, Việt Nam

Email: anhtt@hcmut.edu.vn

Lịch sử

- Ngày nhận: 29-7-2019
- Ngày chấp nhận: 25-8-2019
- Ngày đăng: 04-12-2020

DOI : 10.32508/stdjet.v3iSI1.517

Check for updates

VNU-HCM Press

Trích dẫn bài báo này: Anh T T. **Bảo vệ tính riêng tư cơ sở dữ liệu không-thời gian dựa trên k-anonymity**. *Sci. Tech. Dev. J. - Eng. Tech.;* 3(SI1):SI82-SI94.