Walking in the Crowd: Anonymizing Trajectory Data for Pattern Analysis

Noman Mohammed
CIISE, Concordia University
Montreal, QC
Canada H3G 1M8
no_moham@ciise.concordia.ca

Benjamin C. M. Fung
CIISE, Concordia University
Montreal, QC
Canada H3G 1M8
fung@ciise.concordia.ca

Mourad Debbabi
CIISE, Concordia University
Montreal, QC
Canada H3G 1M8
debbabi@ciise.concordia.ca

ABSTRACT

Recently, trajectory data mining has received a lot of attention in both the industry and the academic research. In this paper, we study the privacy threats in trajectory data publishing and show that traditional anonymization methods are not applicable for trajectory data due to its challenging properties: high-dimensional, sparse, and sequential. Our primary contributions are (1) to propose a new privacy model called LKC-privacy that overcomes these challenges, and (2) to develop an efficient anonymization algorithm to achieve LKC-privacy while preserving the information utility for trajectory pattern mining.

Categories and Subject Descriptors

H.2.7 [**Database Administration**]: [Security, integrity, and protection]

General Terms

Algorithms, Performance, Security

Keywords

Privacy, anonymity, trajectory data

1. INTRODUCTION

In recent years, there has been an explosive growth of location-aware devices such as RFID tags, GPS-based devices, cell phones, and PDAs. The use of these devices facilitates new and exciting location-based applications that consequently generate a huge collection of trajectory data. Recent research reveals that these trajectory data can be used for various data analysis purposes such as city traffic control, mobility management, urban planning, and location-based service advertisements. Clearly, publication of these trajectory data threatens individuals' privacy since these raw trajectory data provide location information that identifies individuals and, potentially, their sensitive information. We use an example to illustrate the privacy threats and challenges of publishing trajectory data.

Table 1: Raw trajectory and health data

ID	Path	Diagnosis	...
1	$\langle b2 \rightarrow d3 \rightarrow c4 \rightarrow f6 \rightarrow c7 \rangle$	AIDS	...
2	$\langle f6 \rightarrow c7 \rightarrow e8 \rangle$	Flu	...
3	$\langle d3 \rightarrow c4 \rightarrow f6 \rightarrow e8 \rangle$	Fever	...
4	$\langle b2 \rightarrow c5 \rightarrow c7 \rightarrow e8 \rangle$	Flu	...
5	$\langle d3 \rightarrow c7 \rightarrow e8 \rangle$	Fever	...
6	$\langle c5 \rightarrow f6 \rightarrow e8 \rangle$	Diabetes	...
7	$\langle b2 \rightarrow f6 \rightarrow c7 \rightarrow e8 \rangle$	Diabetes	...
8	$\langle b2 \rightarrow c5 \rightarrow f6 \rightarrow c7 \rangle$	AIDS	...

Table 2: Anonymous data with $L = 2$, $K = 2$, $C = 50\%$

ID	Path	Diagnosis	...
1	$\langle d3 \rightarrow f6 \rightarrow c7 \rangle$	AIDS	...
2	$\langle f6 \rightarrow c7 \rightarrow e8 \rangle$	Flu	...
3	$\langle d3 \rightarrow f6 \rightarrow e8 \rangle$	Fever	...
4	$\langle c5 \rightarrow c7 \rightarrow e8 \rangle$	Flu	...
5	$\langle d3 \rightarrow c7 \rightarrow e8 \rangle$	Fever	...
6	$\langle c5 \rightarrow f6 \rightarrow e8 \rangle$	Diabetes	...
7	$\langle f6 \rightarrow c7 \rightarrow e8 \rangle$	Diabetes	...
8	$\langle c5 \rightarrow f6 \rightarrow c7 \rangle$	AIDS	...

EXAMPLE 1.1. A hospital wants to release the patient-specific trajectory and health data (Table 1) to a data miner for research purposes. Each record contains a *path* and some patient-specific information, where the *path* is a sequence of *pairs* $(loc_i t_i)$ indicating the patient's visited location loc_i at time t_i. For example, $ID\#2$ has a path $\langle f6 \rightarrow c7 \rightarrow e8 \rangle$, meaning that the patient has visited locations f, c, and e at time 6, 7, and 8, respectively. Without loss of generality, we assume that each record contains only one sensitive attribute, namely, diagnosis, in this example. We address two types of privacy threats:

Identity linkage: If a path in the table is so specific that not many patients match it, releasing the trajectory data may lead to linking the victim's record and, therefore, her diagnosed disease. Suppose the adversary knows that the data record of a target victim, Alice, is in Table 1, and Alice has visited $b2$ and $d3$. Alice's record, together with her sensitive value (AIDS in this case), can be uniquely identified because $ID\#1$ is the *only* record that contains $b2$ and $d3$. Besides, the adversary can also determine the other visited locations of Alice, such as $c4$, $f6$ and $c7$.

Attribute linkage: If a sensitive value occurs frequently together with some sequence of pairs, then the sensitive information can be inferred from such sequence even though the exact record of the victim cannot be identified. Suppose the adversary knows that Bob has visited $b2$ and $f6$. Since

CIKM'09, November 2–6, 2009, Hong Kong, China.

two out of the three records ($ID\#$1,7,8) containing $b2$ and $f6$ have sensitive value AIDS, the adversary can infer that Bob has AIDS with $2/3 = 67\%$ confidence. ■

Many privacy models, such as K-anonymity [7] and its extensions [5][11], have been proposed to thwart privacy threats caused by identity and attribute linkages in the context of relational databases. These privacy models are effective for anonymizing relational data, but they are not applicable to trajectory data due to two special challenges.

(1) High dimensionality: Traditional K-anonymity requires every path to be shared by at least K records. Due to *the curse of high dimensionality* [2], most of the data have to be suppressed in order to achieve K-anonymity. For example, to achieve 2-anonymity on the path data in Table 1, all instances of $\{b2, d3, c4, c5\}$ have to be suppressed.

(2) Data sparseness: Consider patients in a hospital or passengers in a public transit system. They usually visit only a few locations compared to all available locations. Anonymizing these little-overlapping paths poses a significant challenge for traditional anonymization techniques because it is difficult to identify and group the paths together. Enforcing traditional K-anonymity on high-dimensional and sparse data would render the data useless.

1.1 Privacy and Utility

Traditional K-anonymity and its extended privacy models assume that an adversary could potentially use any or even all of the QID attributes as background knowledge to perform identity or attribute linkages. However, in real-life privacy attacks, it is very difficult for an adversary to acquire *all* the visited locations and timestamps of a victim because it requires non-trivial effort to gather each piece of background knowledge from so many possible locations at different times. Thus, it is reasonable to assume that the adversary's background knowledge is bounded by at most L pairs of $(loc_i t_i)$ that the victim has visited. Based on this assumption, we define a new privacy model called *LKC-privacy* [6] for anonymizing high-dimensional and sparse spatio-temporal data.

While protecting privacy is a critical element in data publishing, it is equally important to preserve the utility of the published data because this is the primary reason for publication. In this paper, we aim at preserving the *maximal frequent sequences* (*MFS*) because MFS often serves as the information basis for different primitive data mining tasks on sequential data, such as trajectory pattern mining [4].

1.2 Contributions

Our contributions can be summarized as follows. First, based on the practical assumption that an adversary has only limited background knowledge, we formally present a new privacy model, called LKC-privacy, to address the special challenges of anonymizing high-dimensional, sparse, and sequential trajectory data (Section 3). Second, we present an efficient anonymization algorithm to achieve LKC-privacy while preserving maximal frequent sequences in the anonymous trajectory data (Section 4). Experimental results are omitted due to space limitation.

2. RELATED WORK

Anonymizing High-Dimensional Data. There are some recent works on anonymizing high-dimensional transaction data [9][12][13]. The methods presented in [9][12][13] model the adversary's power by a maximum number of known items as background knowledge. This assumption is similar to ours, However, a transaction is a *set* of items, but a moving object's path is a *sequence* of visited location-time pairs. Sequential data drastically increases the computational complexity for counting the support counts as compared to transaction data because $\langle a \rightarrow b \rangle$ is different from $\langle b \rightarrow a \rangle$. Hence, their proposed models are not applicable to spatio-temporal data.

Anonymizing Moving Objects. Some recent works [1][8][14] address the anonymity of moving objects. In [1], the authors assume that every trajectory is continuous. This assumption is valid for GPS-like devices where the object can be traced all the time, but it does not hold for RFID-based moving objects. The privacy model proposed in [8] assumes that different adversaries have different background knowledge and thus the data holder needs to have the background knowledge of all the adversaries. In reality, such information is difficult to obtain. Yarovoy et al. [14] consider time as a QID attribute. However, there is no fixed set of time for all moving objects, or rather each trajectory has its own set of times as its QID. It is unclear how the data holder can determine the QID attributes for each trajectory. Again, none of these works [1][8][14] aim at achieving anonymity and preserving maximal frequent sequences of the trajectories, which is the main theme of our paper.

3. PROBLEM DEFINITION

A trajectory database T is a collection of records in the form $\langle (loc_1 t_1) \rightarrow \ldots \rightarrow (loc_n t_n) \rangle : s_1, \ldots, s_p : d_1, \ldots, d_m$, where $\langle (loc_1 t_1) \rightarrow \ldots \rightarrow (loc_n t_n) \rangle$ is the path, $s_i \in S_i$ are the sensitive values, and $d_i \in D_i$ are the quasi-identifying (QID) values of an object. Identity and attribute linkages via the QID attributes can be avoided by applying existing anonymization methods for relational data [3][5][10]. In this paper, we focus on eliminating identity and attribute linkages via trajectory data as illustrated in Example 1.1.

3.1 Privacy Model

As explained in Section 1.1, we assume that the adversary knows at most L pairs of location and timestamp that victim has previously visited. We use q to denote such an a priori known sequence of pairs, where $|q| \leq L$. $T(q)$ denotes a group of records that contains q. A record in T *contains* q if q is a subsequence of the path in the record. For example in Table 1, $ID\#1, 2, 7, 8$ contains $q = \langle f6 \rightarrow c7 \rangle$, written as $T(q) = \{ID\#1, 2, 7, 8\}$. Based on background knowledge q, the adversary could launch identity and attribute linkage privacy attacks (Example 1.1). To thwart the identity and attribute linkages, we require that every sequence with a maximum length L in the trajectory database has to be shared by at least a certain number of records, and the ratio of sensitive value(s) in every group cannot be too high. Our privacy model, *LKC-privacy*, reflects this intuition.

DEFINITION 3.1 (LKC-PRIVACY). Let L be the maximum length of the background knowledge. Let S be a set of sensitive values. A trajectory database T satisfies *LKC-privacy* if and only if for any sequence q with $|q| \leq L$,

1. $|T(q)| \geq K$, where $K > 0$ is an integer anonymity threshold, and

2. $P(s|q) \leq C$ for any $s \in S$, where $0 < C \leq 1$ is a real number confidence threshold. ■

LKC-privacy generalizes several traditional privacy models. K-anonymity [7] is a special case of LKC-privacy with $C = 100\%$ and $L = |d|$, where $|d|$ is the number of dimensions, i.e., number of distinct pairs, in the trajectory database. Confidence bounding [10] is a special case of LKC-privacy with $K = 1$ and $L = |d|$. (α, k)-anonymity [11] is also a special case of LKC-privacy with $L = |d|$, $K = k$, and $C = \alpha$. Thus, the data holder can still achieve the traditional models, if needed.

3.2 Utility Measure

The measure of data utility varies depending on the data mining task to be performed on the published data. In this paper, we aim at preserving the maximal frequent sequences.

Definition 3.2 (Maximal frequent sequence). For a given minimum support threshold $K' > 0$, a sequence x is *maximal frequent* in a trajectory database T if x is frequent and no super sequence of x is frequent in T. ■

The set of MFS in T is denoted by $U(T)$. Our data utility goal is to preserve as many MFS as possible, i.e., maximize $|U(T)|$, in the anonymous trajectory database.

3.3 Problem Statement

LKC-privacy can be achieved by performing a sequence of suppressions on selected pairs from T. In this paper, we employ *global suppression*, meaning that if a pair p is chosen to be suppressed, *all* instances of p in T are suppressed. Global suppression retains exactly the same support counts of the preserved MFS in the anonymous trajectory database as there were in the raw data. In contrast, a local suppression scheme may delete *some* instances of the chosen pair and, therefore, change the support counts of the preserved MFS. The property of data truthfulness is vital in some data analysis, such as traffic analysis.

Definition 3.3 (Trajectory Anonymity for MFS). Given a trajectory database T, a LKC-privacy requirement, a minimum support threshold K', a set of sensitive values S, the problem of *trajectory anonymity for maximal frequent sequences (MFS)* is to identify a transformed version of T that satisfies the LKC-privacy requirement while preserving the maximum number of MFS with respect to K'. ■

Finding an optimum solution for LKC-privacy is NP-hard. Thus, we propose a greedy algorithm to efficiently identify a reasonably "good" sub-optimal solution.

4. ANONYMIZATION ALGORITHM

Given a trajectory database T, our first step is to identify all sequences that violate the given LKC-privacy requirement. Section 4.1 describes a method to identify violating sequences efficiently. Section 4.2 presents a greedy algorithm to eliminate the violating sequences with the goal of preserving as many maximal frequent sequences as possible.

4.1 Identifying Violating Sequences

An adversary may use any sequence with length not greater than L as background knowledge to launch a linkage attack.

Algorithm 1 MVS Generator

Input: Raw trajectory database T
Input: Thresholds L, K, and C
Input: Sensitive values S
Output: Minimal violating sequence $V(T)$

```
1: X_1 ← set of all distinct pairs in T;
2: i = 1;
3: while i ≤ L or X_i ≠ ∅ do
4:    Scan T to compute |T(q)| and P(s|q), for ∀q ∈ X_i, ∀s ∈ S;
5:    for ∀q ∈ X_i where |T(q)| > 0 do
6:       if |T(q)| < K or P(s|q) > C then
7:          Add q to V_i;
8:       else
9:          Add q to W_i;
10:      end if
11:   end for
12:   X_{i+1} ← W_i ⋈ W_i;
13:   for ∀q ∈ X_{i+1} do
14:      if q is a super sequence of any v ∈ V_i then
15:         Remove q from X_{i+1};
16:      end if
17:   end for
18:   i++;
19: end while
20: return V(T) = V_1 ∪ ... ∪ V_{i-1};
```

Thus, any non-empty sequence q with $|q| \leq L$ in T is a *violating sequence* if its group $T(q)$ does not satisfy condition 1, condition 2, or both in LKC-privacy in Definition 3.1.

Example 4.1. Let $L = 2$, $K = 2$, $C = 50\%$, and $S = \{AIDS\}$. In Table 1, a sequence $q_1 = \langle b2 \rightarrow c4 \rangle$ is a violating sequence because $|T(q_1)| = 1 < K$. A sequence $q_2 = \langle b2 \rightarrow f6 \rangle$ is a violating sequence because $P(AIDS|q_2) = 67\% > C$. However, a sequence $q_3 = \langle b2 \rightarrow c5 \rightarrow f6 \rightarrow c7 \rangle$ is not a violating sequence even if $|T(q_3)| = 1 < K$ and $P(AIDS|q_3) = 100\% > C$ because $|q_3| > L$. ■

A trajectory database satisfies a given LKC-privacy requirement, if all violating sequences with respect to the privacy requirement are removed, because all possible channels for identity and attribute linkages are eliminated. A naive approach is to first enumerate all possible violating sequences and then remove them. This approach is infeasible because of the huge number of violating sequences. Consider a violating sequence q with $|T(q)| < K$. Any super sequence of q, denoted by q'', in the database T is also a violating sequence because $|T(q'')| \leq |T(q)| < K$.

To overcome this bottleneck of violating sequence enumeration, our insight is that there exists some "minimal" violating sequences among the violating sequences, and it is sufficient to achieve LKC-privacy by removing only the minimal violating sequences.

Definition 4.1 (Minimal violating sequence). A violating sequence q is a *minimal violating sequence* (MVS) if every proper subsequence of q is not a violating sequence. ■

Example 4.2. In Table 1, given $L = 3$, $K = 2$, $C = 50\%$, $S = \{AIDS\}$, the sequence $q = \langle b2 \rightarrow d3 \rangle$ is a MVS because $\langle b2 \rangle$ and $\langle d3 \rangle$ are not violating sequences. The sequence $q = \langle b2 \rightarrow d3 \rightarrow c4 \rangle$ is a violating sequence but not a MVS because its subsequence $\langle b2 \rightarrow d3 \rangle$ is a violating sequence. ■

Every violating sequence is either a MVS or it contains a MVS. Thus, if T contains no MVS, then T contains no violating sequences.

Table 3: Initial *Score*

	b2	d3	c4	f6	c7	e8
PrivGain	3	1	3	1	1	1
UtilityLoss (+1)	4	4	2	5	6	5
Score	0.75	0.25	1.5	0.2	0.16	0.2

Table 4: *Score* after suppressing *c*4

	b2	d3	f6
PrivGain	2	1	1
UtilityLoss (+1)	4	3	4
Score	0.5	0.33	0.25

Algorithm 1 presents a method to efficiently generate all MVS. Line 1 puts all the size-1 sequences, i.e., all distinct pairs, as candidates X_1 of MVS. Line 4 scans T once to compute $|T(q)|$ and $P(s|q)$ for each sequence $q \in X_i$ and for each sensitive value $s \in S$. If the sequence q violates the LKC-privacy requirement in Line 6, then we add q to the MVS set V_i (Line 7); otherwise, add q to the non-violating sequence set W_i (Line 9) for generating the next candidate set X_{i+1}, which is a self-join of W_i (Line 12). Two sequences $q_x = \langle (loc_1^x t_1^x) \rightarrow \ldots \rightarrow (loc_i^x t_i^x) \rangle$ and $q_y = \langle (loc_1^y t_1^y) \rightarrow \ldots \rightarrow (loc_i^y t_i^y) \rangle$ in W_i can be joined only if the first $i-1$ pairs of q_x and q_y are identical and $t_i^x < t_i^y$. The joined sequence is $\langle (loc_1^x t_1^x) \rightarrow \ldots \rightarrow (loc_i^x t_i^x) \rightarrow (loc_i^y t_i^y) \rangle$. Lines 13-17 remove a candidate q from X_{i+1} if q is a super sequence of any sequence in V_i because any proper subsequence of a MVS cannot be a violating sequence. The set of MVS, denoted by $V(T)$, is the union of all V_i.

EXAMPLE 4.3. Consider Table 1 with $L = 2$, $K = 2$, $C = 50\%$, and $S = \{AIDS\}$. $X_1 = \{b2, d3, c4, c5, f6, c7, e8\}$. After scanning T, we divide X_1 into $V_1 = \emptyset$ and $W_1 = \{b2, d3, c4, c5, f6, c7, e8\}$. Next, from W_1 we generate the candidate set $X_2 = \{b2d3, b2c4, b2c5, b2f6, b2c7, b2e8, d3c4, d3c5, d3f6, d3c7, d3e8, c4c5, c4f6, c4c7, c4e8, c5f6, c5c7, c5e8, f6c7, f6e8, c7e8\}$. We scan T again to determine $V_2 = \{b2d3, b2c4, b2f6, c4c7, c4e8\}$. We do not further generate X_3 because $L = 2$. ■

4.2 Eliminating Violating Sequences

We propose a greedy algorithm to transform the raw trajectory database T to an anonymous table T' with respect to a given LKC-privacy requirement by a sequence of suppressions. In each iteration, the algorithm selects a pair p for suppression based on a greedy selection function. In general, a suppression on a pair p in T increases privacy because it removes minimal violating sequences (MVS), and decreases data utility because it eliminates maximal frequent sequences (MFS) in T. Therefore, we define the greedy function, $Score(p)$, to select a suppression on a pair p that maximizes the number of MVS removed but minimizes the number of MFS removed in T. $Score(p)$ is defined as follows:

$$Score(p) = \frac{PrivGain(p)}{UtilityLoss(p) + 1} \quad (1)$$

where $PrivGain(p)$ and $UtilityLoss(p)$ are the number of minimal violating sequence (MVS) and the number of maximal frequent sequence (MFS) containing the pair p, respectively. A pair p may not belong to any MFS, resulting in $|UtilityLoss(p)| = 0$. To avoid dividing by zero, we add 1 to the denominator. The pair p with the highest $Score(p)$ is called the *winner* pair, denoted by w.

Table 3 shows the initial $Score(p)$ of every candidate pair for Table 1. Initially, c4 is suppressed since it has the highest score. After suppressing c4, we update the score of the remaining candidate pairs (Table 4). In the next iteration, $b2$ is suppressed and thus all the remaining MVS are removed. Table 2 shows the resulting anonymized table T' for $(2, 2, 50\%)$-privacy. Due to space limitation, we do not elaborate how we efficiently calculate and update the $Score(p)$ of the candidate pairs.

5. CONCLUSION

We proposed a new LKC-privacy model based on the assumption that an adversary has limited background knowledge about the victim. We also presented an efficient algorithm for achieving LKC-privacy with the goal of preserving maximal frequent sequences, which serves as the basis of many data mining tasks on sequential data.

6. ACKNOWLEDGMENTS

The research is supported in part by Discovery Grants (356065-2008) and Canada Graduate Scholarship from the Natural Sciences and Engineering Research Council of Canada.

7. REFERENCES

[1] O. Abul, F. Bonchi, and M. Nanni. Never walk alone: Uncertainty for anonymity in moving objects databases. In *IEEE ICDE*, 2008.

[2] C. C. Aggarwal. On k-anonymity and the curse of dimensionality. In *VLDB*, 2005.

[3] B. C. M. Fung, K. Wang, and P. S. Yu. Anonymizing classification data for privacy preservation. *IEEE TKDE*, 2007.

[4] F. Giannotti, M. Nanni, D. Pedreschi, and F. Pinelli. Trajectory pattern mining. In *ACM SIGKDD*, 2007.

[5] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. ℓ-diversity: Privacy beyond k-anonymity. *ACM TKDD*, 2007.

[6] N. Mohammed, B. C. M. Fung, P. C. K. Hung, and C. K. Lee. Anonymizing healthcare data: a case study on the blood transfusion service. In *SIGKDD*, 2009.

[7] P. Samarati and L. Sweeney. Generalizing data to provide anonymity when disclosing information. In *PODS*, 1998.

[8] M. Terrovitis and N. Mamoulis. Privacy preservation in the publication of trajectories. In *MDM*, 2008.

[9] M. Terrovitis, N. Mamoulis, and P. Kalnis. Privacy-preserving anonymization of set-valued data. In *VLDB*, 2008.

[10] K. Wang, B. C. M. Fung, and P. S. Yu. Handicapping attacker's confidence: An alternative to k-anonymization. *KAIS*, 2007.

[11] R. C. W. Wong, J. Li., A. W. C. Fu, and K. Wang. (α,k)-anonymous data publishing. *Journal of Intelligent Information Systems*, in press.

[12] Y. Xu, B. C. M. Fung, K. Wang, A. W. C. Fu, and J. Pei. Publishing sensitive transactions for itemset utility. In *IEEE ICDM*, 2008.

[13] Y. Xu, K. Wang, A. W. C. Fu, and P. S. Yu. Anonymizing transaction databases for publication. In *ACM SIGKDD*, 2008.

[14] R. Yarovoy, F. Bonchi, L. V. S. Lakshmanan, and W. H. Wang. Anonymizing moving objects: How to hide a MOB in a crowd? In *EDBT*, 2009.