Transportation Research Part C 39 (2014) 63–79

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Anonymizing trajectory data for passenger flow analysis

Moein Ghasemzadeh [a], Benjamin C.M. Fung [b,*], Rui Chen [c], Anjali Awasthi [a]

[a] CIISE, Concordia University, Montreal, Quebec H3G 1M8, Canada
[b] School of Information Studies, McGill University, Montreal, Quebec H3A 1X1, Canada
[c] Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, Hong Kong

ARTICLE INFO

Article history:
Received 24 August 2013
Received in revised form 6 December 2013
Accepted 6 December 2013

Keywords:
Data privacy
Anonymity
Trajectory
Passenger flow

ABSTRACT

The increasing use of location-aware devices provides many opportunities for analyzing and mining human mobility. The trajectory of a person can be represented as a sequence of visited locations with different timestamps. Storing, sharing, and analyzing personal trajectories may pose new privacy threats. Previous studies have shown that employing traditional privacy models and anonymization methods often leads to low information quality in the resulting data. In this paper we propose a method for achieving anonymity in a trajectory database while preserving the information to support effective passenger flow analysis. Specifically, we first extract the passenger flowgraph, which is a commonly employed representation for modeling uncertain moving objects, from the raw trajectory data. We then anonymize the data with the goal of minimizing the impact on the flowgraph. Extensive experimental results on both synthetic and real-life data sets suggest that the framework is effective to overcome the special challenges in trajectory data anonymization, namely, high dimensionality, sparseness, and sequentiality.

1. Introduction

Over the last few years, transit companies have started using contactless smart cards or RFID cards, such as the EasyCard in Taiwan, the Public Transportation Card in Shanghai, and the OPUS card in Montréal. In 2008, *Société de transport de Montréal* (*STM*), the public transit agency in Montréal, deployed the *Smart Card Automated Fare Collection* (*SCAFC*) system (Pelletier et al., 2011) in its transportation network. Senior and junior passengers have to register their personal information when they first purchase their cards so that an appropriate fare is charged based on their statuses. In the SCAFC system, each passenger leaves a trace of reading in the form of (ID, loc, t), which identifies the passenger's identity, location, and time when she scans her smart card. The trajectory of a passenger is then stored as a sequence of visited locations, sorted by time, in a central database.

Constructions occur and new trends emerge as a city develops. Thus, passenger flows in a city are not static and are subject to change depending on all these uncertainties and developments. Transit companies have to periodically share their passengers' trajectories among their own internal departments and external transportation companies in order to perform a comprehensive analysis of passenger flows in an area, with the goal of supporting trajectory data mining (Giannotti et al., 2007; Lee et al., 2008, 2007; Tang et al., 2012; Zheng et al., 2013) and traffic management (Burger et al., 2013; Li et al., 2007a). For instance, by using a probabilistic flowgraph, as shown in Fig. 1, an analyst can identify the major trends in passenger flows and hot paths in a traffic network. However, sharing passenger-specific trajectory data raises new privacy

* Corresponding author. Tel.: +1 5143983360.
E-mail addresses: mo_gh@ciise.concordia.ca (M. Ghasemzadeh), ben.fung@mcgill.ca (B.C.M. Fung), ruichen@comp.hkbu.edu.hk (R. Chen), awasthi@ciise.concordia.ca (A. Awasthi).

0968-090X/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.trc.2013.12.003

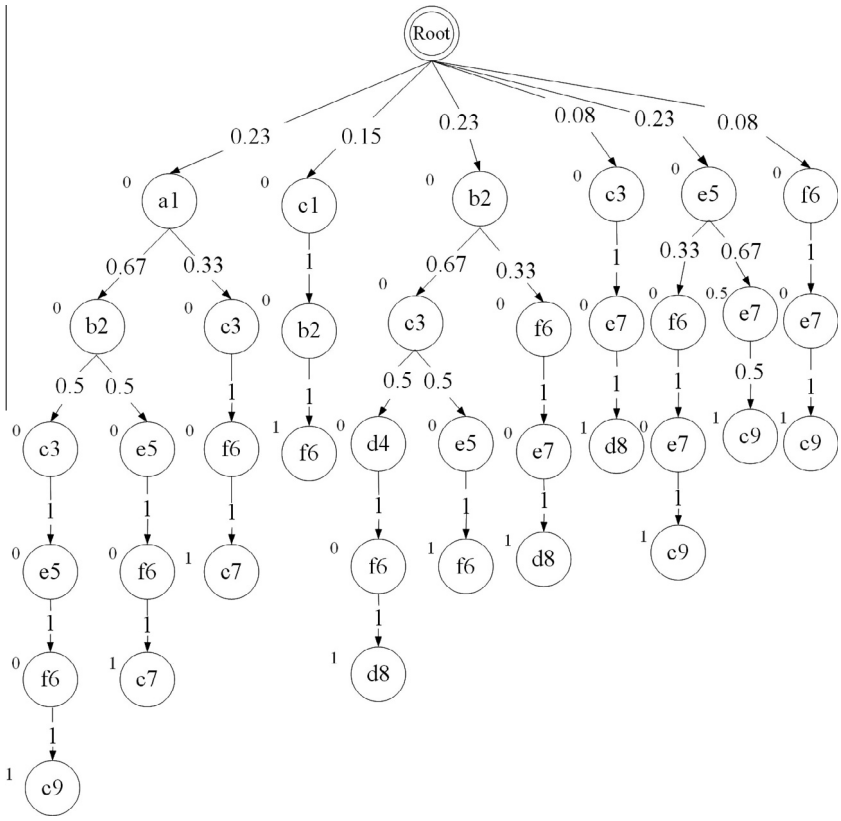

Fig. 1. Probabilistic flowgraph of Table 1.

concerns that cannot be appropriately addressed by traditional privacy protection techniques. Example 1.1 illustrates a potential privacy threat in the context of trajectory data.

Example 1.1 (*Identity linkage attack*). Table 1 shows an example of thirteen passengers' trajectories, in which each trajectory consists of a sequence of spatio-temporal doublets (or simply doublets). Each doublet has the form $(loc_i t_i)$, representing the visited location loc_i with timestamp t_i. For example, $ID\#4$ indicates that the passenger has visited locations c, e, and d at timestamps 3, 7, and 8, respectively. With adequate background knowledge, an adversary can perform a type of privacy attack, called *identity linkage attack*, on the trajectory database and may be able to uniquely identify a victim's record as well as his/her visited locations. Preventing identity linkage attacks is very important in trajectory data sharing because it is easy to be performed by an attacker and upon success, it allows the attacker to learn all other locations and timestamps of the victim. Hence, it is the main goal of this paper. Suppose an adversary knows that the data record of a target victim, Alice, is in

Table 1
Raw trajectory database T.

Series ID #	Series trajectory
1	$a1 \rightarrow b2 \rightarrow c3 \rightarrow e5 \rightarrow f6 \rightarrow c9$
2	$e5 \rightarrow f6 \rightarrow e7 \rightarrow c9$
3	$e5 \rightarrow e7$
4	$c3 \rightarrow e7 \rightarrow d8$
5	$b2 \rightarrow c3 \rightarrow d4 \rightarrow f6 \rightarrow d8$
6	$c1 \rightarrow b2 \rightarrow f6$
7	$a1 \rightarrow b2 \rightarrow e5 \rightarrow f6 \rightarrow e7$
8	$f6 \rightarrow e7 \rightarrow c9$
9	$e5 \rightarrow e7 \rightarrow c9$
10	$b2 \rightarrow f6 \rightarrow e7 \rightarrow d8$
11	$a1 \rightarrow c3 \rightarrow f6 \rightarrow e7$
12	$c1 \rightarrow b2 \rightarrow f6$
13	$b2 \rightarrow c3 \rightarrow e5 \rightarrow f6$

Table 1 and that Alice has visited locations b and c at timestamps 2 and 9, respectively. Then the adversary can associate $ID\#1$ with Alice because $ID\#1$ is the only record containing both $b2$ and $c9$.

This paper presents a new method to anonymize a large volume of passenger-specific trajectory data with minimal impact on the information quality for passenger flow analysis. This work falls into a research area called *Privacy-Preserving Data Publishing* (*PPDP*), which aims at releasing anonymized data for general data analysis or specific data mining tasks (Clifton and Tassa, 2013). To the best of our knowledge, this is the first work studying trajectory data anonymization for passenger flow analysis.

1.1. Data privacy and quality trade-off

Several privacy models, such as *K*-anonymity (Samarati and Sweeney, 1998) and its extensions (Cao and Karras, 2012; Li et al., 2007a; Machanavajjhala et al., 2007; Wang et al., 2007; Wong et al., 2007), have been proposed to thwart privacy threats in the context of relational data. However, these models are not effective on trajectory data due to its high dimensionality, sparseness, and sequentiality (Chen et al., 2013). Consider a mass transportation system with 300 metro and bus stations operating 20 h a day. The corresponding trajectory database would have $300 \times 20 = 6000$ dimensions. Since *K*-anonymity requires every trajectory to be shared by at least K records, most of the data have to be suppressed in order to achieve *K*-anonymity. Moreover, trajectory data are usually sparse because most passengers visit only a few stations within a short period of time. Enforcing *K*-anonymity on sparse trajectories in a high-dimensional space usually results in suppression of most of the data; therefore, the released data are rendered useless for analysis. Furthermore, these privacy models do not consider the sequentiality in trajectories. A passenger traveling from station a to station b is different from the one traveling from b to a. Sequentiality captures vital information for passenger flow analysis.

To overcome the challenges of anonymizing high-dimensional and sparse data, a new privacy model called *LK-privacy* (Mohammed et al., 2010) is adopted in this paper to prevent identity linkage attacks. *LK*-privacy was originally proposed to anonymize high-dimensional relational health data. This privacy model was built based on the observation that an adversary usually has only limited knowledge about a target victim. The same assumption also applies to trajectory data, that is, an adversary knows at most L previously visited spatio-temporal doublets of any target passenger. Therefore, applying the same privacy notion to trajectory data requires every subsequence with length at most L in a trajectory database T to be shared by at least K records in T, where L and K are positive integer thresholds. *LK*-privacy guarantees that the probability of a successful identity linkage attack is at most $1/K$. Table 2 presents an example of an anonymous database satisfying $(2,2)$-privacy from Table 1, in which every subsequence with maximum length 2 is shared by at least 2 records.

While privacy preservation is essential for the data holder, preserving the information quality is important for the data recipient to perform the needed analysis. Anonymous data may be used for different data mining tasks; however, in this paper we aim at preserving the information quality of the probabilistic flowgraph, which is the primary use of trajectory data in passenger flow analysis. A probabilistic flowgraph is a tree where each node represents a spatio-temporal doublet (loc, t), and an edge corresponds to a transition between two doublets. All common trajectory prefixes appear in the same branch of the tree. Each transition has an associated probability, which is the percentage of passengers who take the transition represented by the edge. For every node we also record a termination probability, which is the percentage of passengers who exit the transportation system at the node. As an illustration, Fig. 1 presents the probabilistic flowgraph derived from Table 1.

We present an example to illustrate the benefit of *LK*-privacy over the traditional *K*-anonymity model.

Example 1.2. Fig. 1 depicts the probabilistic flowgraph generated from the raw trajectory data (Table 1). Fig. 2 depicts the probabilistic flowgraph generated from Table 2, which satisfies $(2,2)$-privacy. Fig. 3 depicts the probabilistic flowgraph generated from the traditional 2-anonymous data. It is clear that Fig. 2 contains more information, including doublet nodes, branches, and transitional probabilities, in the flowgraph than Fig. 3. For example, Fig. 1 shows that 23%

Table 2
$(2,2)$-Privacy preserved database T'.

Series ID #	Series trajectory
1	$a1 \rightarrow b2 \rightarrow c3 \rightarrow e5 \rightarrow f6$
2	$e5 \rightarrow f6 \rightarrow e7 \rightarrow c9$
3	$e5 \rightarrow e7$
4	$c3 \rightarrow e7 \rightarrow d8$
5	$b2 \rightarrow c3 \rightarrow f6 \rightarrow d8$
6	$c1 \rightarrow b2 \rightarrow f6$
7	$a1 \rightarrow b2 \rightarrow e5 \rightarrow f6 \rightarrow e7$
8	$f6 \rightarrow e7 \rightarrow c9$
9	$e5 \rightarrow e7 \rightarrow c9$
10	$b2 \rightarrow f6 \rightarrow e7 \rightarrow d8$
11	$a1 \rightarrow c3 \rightarrow f6 \rightarrow e7$
12	$c1 \rightarrow b2 \rightarrow f6$
13	$b2 \rightarrow c3 \rightarrow e5 \rightarrow f6$

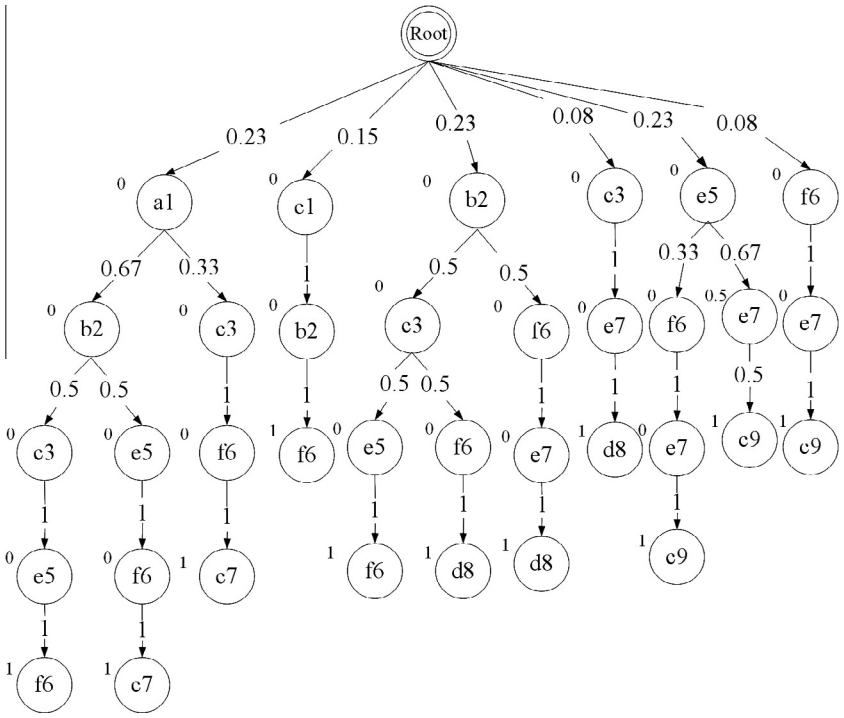

Fig. 2. *LK*-anonymized probabilistic flowgraph of Table 1.

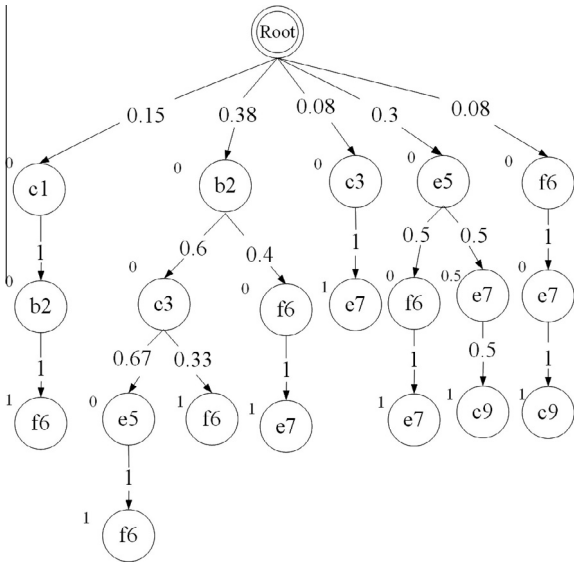

Fig. 3. *K*-anonymized probabilistic flowgraph of Table 1.

of passengers start their route from *b*2. Fig. 2 preserves the same probability, but Fig. 3 incorrectly interprets the probability as 38%, resulting in a misleading analysis. This claim is further supported by extensive experimental results in Section 5.

If probabilistic flowgraph analysis is the goal, why does not the data holder simply build and publish the flowgraph without releasing the data? First, the data holder may not know exactly how data recipients would like to perform the analysis. In real-life scenarios, it is impractical to request the data holder to accommodate the data analysis requests from different data recipients. Second, although the anonymous data is customized for probabilistic flowgraph analysis, the data recipients are

free to perform other types of analysis, such as answering count queries. Third, the recipients can perform interactive data mining on the anonymous data, which requires users' inputs based on the observed data distribution at each step. In general, releasing trajectory data provides data recipients with a greater flexibility of analysis.

LK-privacy can be achieved by global suppression or local suppression of doublets. *Global suppression* on a doublet d means that *all* instances of d are removed from the data. *Local suppression* on a doublet d means that *some* instances of d are removed while some remain intact. Global suppression punishes all records containing d by eliminating all instances of d even if the privacy threat is caused by only one instance of d. In contrast, local suppression eliminates the exact instances causing the privacy violations without penalizing others, and hence preserves more information for data analysis but with the cost of higher computational complexity. In this paper, we employ a hybrid approach, with the goal of maintaining high quality of data for passenger flow analysis with feasible computational complexity.

1.2. Contributions

Based on the practical assumption that an adversary has only limited background knowledge on a target victim, we adapt *LK*-privacy model for trajectory data anonymization, which prevents identity linkage attacks on trajectory data. This paper makes three major contributions. First, this is the first work that aims at preserving both spatio-temporal data privacy and information quality for passenger flow analysis. Second, we design a hybrid approach that makes use of both global and local suppressions to achieve a reasonable tradeoff between data privacy and information quality. Third, we present a method to measure the similarity between two probabilistic flowgraphs in order to evaluate the difference of information quality before and after anonymization. Extensive experimental results on both real-life and synthetic trajectory data sets suggest that our proposed algorithm is both effective and efficient to address the special challenges in trajectory data anonymization for passenger flow analysis.

2. Related work

In this section, we first provide an overview of traffic and passenger flow analysis followed by a review of some common privacy models for relational, statistical, transaction, and trajectory data.

2.1. Flow analysis

Paletta et al. (2005) present a pilot system that helps public transportation system companies optimize the passenger flows at traffic junctions. The system utilizes video surveillance, with the help of AI vision, to monitor and analyze pedestrians' trajectories. Descriptive statistics between different sources and destinations generated from trajectories provide an overview of passenger flows. Later, Halb and Neuschmied (2009) propose an improved system for multi-modal semantic analysis of individuals' movements at public transportation hubs, which is also applicable to other settings, such as consumers' movements in shopping malls.

Abraham and Lal (2012) propose a model to determine the similarity of vehicle trajectories with respect to space and time, which has an important role in many traffic related applications. In their proposed model, they use a remote database to regularly update the trajectories of moving vehicles based on a cellular network. The database server periodically processes the trajectories to form the spatio-temporal similarity set. The details of the vehicles in a similar cluster are dispersed through the cluster head. After obtaining the information from the server, the cluster head vehicle uses the *vehicular ad hoc networks* (*VANET*) infrastructure to share the required information with its neighborhood.

2.2. Relational and statistical data anonymization

K-anonymity (Samarati and Sweeney, 1998), *ℓ-diversity* (Machanavajjhala et al., 2007), and *confidence bounding* (Wang et al., 2007) are common models that prevent privacy attacks on relational data. *K*-anonymity prevents linkage attacks by requiring every *equivalence class* (i.e., a set of records that are indistinguishable from each other with respect to certain identifying attributes) in a relational data table T to contain at least K records. ℓ-diversity requires that the distribution of a sensitive attribute in each equivalence class has at least ℓ *well-represented* values. Wang et al. (2007) present a method to limit the privacy threat by taking into account a set of *privacy templates* specified by the data owner. Such templates formulate individuals' privacy constraints in the form of association rules. Wong et al. (2007) propose a new privacy model called (α, K)-anonymization by integrating both *K*-anonymity and confidence bounding into a single privacy model.

Li and Li (2008) propose a method to model an adversary's background knowledge by mining negative association rules, which is then used in the anonymization process. Kisilevich et al. (2010) propose *K*-anonymity of classification trees using suppression, in which multidimensional suppression is performed by using a decision tree to achieve *K*-anonymity. Matatov et al. (2010) propose anonymizing separate projections of a dataset instead of anonymizing the entire dataset by partitioning the underlying dataset into several partitions that satisfy *K*-anonymity. A classifier is

trained on each projection and then classification tasks are performed by combining the classification of all such classifiers.

Enforcing traditional privacy models on high dimensional relational data usually results in suppressing most of the data (Aggarwal, 2005), thus rendering the released data useless for future analysis. Mohammed et al. (2010) propose the *LKC*-privacy model for high dimensional relational data, which assumes that the adversary's background knowledge is limited to at most L attributes. In this paper, we follow a similar assumption on an adversary's background knowledge and adapt the privacy notion for trajectory data.

Dwork (2006) proposes an insightful privacy notion, called ϵ-*differential privacy*, based on the principle that the risk to a data owner's privacy should not substantially increase as a result of participating in a statistical database. ϵ-differential privacy ensures that the removal or addition of a single database record does not substantially affect the outcome of any analysis. In spite of the rigorous privacy guarantee provided by differential privacy, it has been criticized for not being able to achieve usable information quality in some data analysis tasks (Yang et al., 2012). In particular, for passenger flow analysis, achieving differential privacy may not be able to provide meaningful data utility. Furthermore, Machanavajjhala et al. (2009) indicate that the resulting data is untruthful due to the uncertainty (e.g., Laplace noise) introduced for achieving differential privacy.

2.3. Transaction data anonymization

Anonymizing high dimensional transaction data has been studied widely in Chen et al. (2011), Ghinita et al. (2008), He and Naughton (2009), Tassa et al. (2012), Terrovitis et al. (2008), and Xu et al. (2008a,b). In general, this problem setting does not take into account the sequentiality, which is vital in our problem. Ghinita et al. (2008) propose a permutation method that groups transactions with close proximity and then associates each group to a set of mixed sensitive values. Terrovitis et al. (2008) propose an algorithm to K-anonymize transactions by *generalization* based on some given taxonomy trees. He and Naughton (2009) extend the method in Terrovitis et al. (2008) by introducing local generalization, which gains better quality. However, generalization does not fit trajectory data well. This is because, in real-life trajectory databases, taxonomy trees may not be available, or a logical one for locations may not exist. Moreover, Fung et al. (2010) indicate that the taxonomy tree of trajectory data tends to be flat and fans out; thus, employing generalization leads to more information loss than does employing suppression. This is due to the fact that generalization requires all siblings of a selected node to merge with their parent node, while suppression only removes the selected child nodes.

Xu et al. (2008a,b) extend the K-anonymity model by assuming that an adversary knows at most a certain number of transaction items of a target victim, which is similar to our assumption of limited background knowledge of an adversary. Although their method addresses the high dimensionality concern, it considers a transaction as a *set* of items rather than a *sequence*. Therefore, it is not applicable to our problem, which needs to take into consideration the sequentiality of trajectory data. Furthermore, Xu et al. (2008a,b) achieve their privacy model by merely global suppression, which significantly hinders information quality on trajectory data.

Tassa et al. (2012) improve the quality of K-anonymity by introducing new models : $(K,1)$ -*anonymity*, $(1,K)$ -*anonymity*, and (K,K) -*anonymity* and K-*concealment*. They argue that $(K,1)$-anonymity, $(1,K)$-anonymity, and (K,K)-anonymity do not provide the same level of security as K-anonymity. K-concealment, on the other hand, provides the comparable level of security that guarantees that every record is computationally indistinguishable from at least $K-1$ others with higher quality. In their work, anonymity is typically achieved by means of generalizing the database entries until some syntactic condition is met.

Chen et al. (2011) study the releasing of transaction dataset while satisfying differential privacy. In their proposed method, the transaction dataset is partitioned in a top-down fashion guided by a context-free taxonomy tree, and the algorithm reports the noisy counts of the transactions at the leaf level. This method generates a synthetic transaction dataset, which can then be used to mine the top-N frequent itemsets. Although they claim that their approach maintains high quality and scalability in the context of set-valued data and is applicable to the relational data, their method is limited to preserving information for supporting count queries and frequent itemsets, not passenger flowgraphs which is the main information to preserve in this paper.

Transaction data anonymization methods fail to provide the claimed privacy guarantee when applied on trajectory data because an attacker can utilize sequential information to launch a privacy attack. Consider a sequential data table with two records $[\langle a \rightarrow b\rangle, \langle b \rightarrow a\rangle]$. This table satisfies transactional 2-anonymity but fails to satisfy sequential 2-anonymity. Suppose attacker X knows that a target victim has visited a and b without knowing the order. Suppose attacker Y further knows that the victim has visited a followed by b. This transactional 2-anonymous table can prevent linkage attacks from attacker X but cannot prevent those from Y. In contrast, consider another table $[\langle a \rightarrow b\rangle, \langle a \rightarrow b\rangle]$. This table satisfies both transactional 2-anonymity and sequential 2-anonymity. In fact, any table that satisfies sequential K-anonymity must satisfy transactional K-anonymity. This table can prevent the attacks from both attackers.

2.4. Trajectory data anonymization

Some recent works (Abul et al., 2008; Chen et al., 2012a,b; Fung et al., 2009a,b; Mohammed et al., 2009; Pensa et al., 2008; Terrovitis and Mamoulis, 2008; Yarovoy et al., 2009) study anonymization of trajectory data from different perspectives. The

works can be broadly classified into two categories based on how they model trajectory data. The first category assumes the data is in the form of continuous GPS data (Herrera et al., 2010). Based on the assumption that trajectories are imprecise, (Abul et al., 2008) propose (K,δ)-*anonymity*, in which δ represents a lower bound of the uncertainty radius when recording the locations of trajectories. Based on *space translation*, in (K,δ)-anonymity K different trajectories should exist in a cylinder of the radius δ. However, the imprecision assumption may not hold in some sources of trajectory data, such as passenger data and RFID data. Trujillo-Rasua and Domingo-Ferrer (2013) illustrate that, in general, (K,δ)-anonymity does not offer trajectory K-anonymity for any $\delta > 0$. It only offers this property for $\delta = 0$ when the set of anonymized trajectories consists of clusters containing K or more identical trajectories each.

The second category models trajectory data in the simplified form of *sequential data*, in which detailed timestamps are ignored. Pensa et al. (2008) and Terrovitis and Mamoulis (2008) study the privacy protection on high dimensional trajectory data. Pensa et al. (2008) propose a variant of the K-anonymity model for sequential data, with the goal of preserving frequent sequential patterns. Similar to the space translation method in Abul et al. (2008) and Pensa et al. (2008) transform a sequence into another form by inserting, deleting, or substituting some items. Based on the assumption that different adversaries have different background knowledge of a victim, Terrovitis and Mamoulis (2008) propose that the data holder should be aware of *all* such adversarial knowledge. The objective is to prevent the adversary from obtaining more information about the published sequential data. Although in their specific scenario it is feasible to know all adversarial background knowledge before publishing the sequential data, this assumption is, generally, not applicable to trajectory data. Simplifying trajectory data to sequential data does help overcome the issue of high dimensionality. However, for many trajectory data mining tasks, the time information is essential. Therefore, these approaches fail to satisfy the information requirement for passenger flow analysis.

Yarovoy et al. (2009) provide privacy protection by utilizing an innovative notion of K-anonymity based on spatial generalization in the context of *moving object databases* (*MOD*). They propose two different anonymization algorithms, *extreme union* and *symmetric anonymization*, based on the assumption that different moving objects may have different *quasi-identifiers* (*QID*), thus anonymization groups associated with different objects may not be disjoint. Monreale et al. (2010) propose a method to ensure K-anonymity by transforming trajectory data based on *spatial generalization*. Hu et al. (2010) present a new problem of K-anonymity with respect to a reference database. Unlike previous K-anonymity algorithms that use conventional hierarchy or partition-based generalization, they make the published data more useful by utilizing a new generalization method called *local enlargement*.

Chen et al. (2012b) propose a sanitization algorithm to generate differentially private trajectory data by making use of a noisy prefix tree based on the underlying data. As a post-processing step, they make use of the inherent consistency constraints of a prefix tree to conduct constrained inferences, which lead to better data quality. Later, Chen et al. (2012a) improve the data quality of sanitized data by utilizing the *variable-length n-gram model*, which provides an effective means for achieving differential privacy on sequential data. They argue that their approach leads to better quality in terms of count query and frequent sequential pattern mining. However, these two approaches are limited to relatively simple data mining tasks. They are not applicable for passenger flow analysis.

Some other recent works (Chen et al., 2013; Fung et al., 2009a,b; Mohammed et al., 2009) study preventing identity linkage attacks over trajectory data but with different information requirements. Fung et al. (2009a,b) focus on minimal data distortion and (Chen et al., 2013; Mohammed et al., 2009) focus on preserving maximal frequent sequences. None of these focus on preserving information quality for generating passenger flowgraphs as discussed in this paper.

In this paper, a passenger-specific trajectory is modeled as a sequence of spatio-temporal doublets. We would like to compare this model with other trajectory models in terms of spatial information and temporal information *through the lens of privacy protection*. The spatial distance among different locations is not considered in the anonymization process because the spatial relationship is neither identifying information nor sensitive information. Revealing the spatial relationship, for example, the distance between two bus stations, does not reveal any sensitive information of passengers. Therefore, there is no need to alter the spatial relationship in the anonymization process. An analyst can still utilize the spatial relationship to perform his/her analysis on the anonymous passenger data or the passenger flowgraph. On the other hand, our model does consider the timestamp information, e.g., $\langle a3 \rightarrow b7\rangle$ in the anonymization process because an attacker may utilize the timestamp information to identify a passenger from the released data. Unlike the works that model trajectories as sequential data, e.g., $\langle a \rightarrow b\rangle$, timestamps in our model provide vital information to construct the passenger flowgraph. Furthermore, the anonymous trajecotry data produced by our method can answer queries with timestamps, but sequential data do not share this feature.

3. Problem description

A trajectory database, the *LK*-privacy model, and a passenger flowgraph are formally defined in this section.

3.1. Trajectory database

A typical *Smart Card Automated Fare Collection* (*SCAFC*) system records the smart card usage data in the form of (ID, loc, t), representing a passenger with a unique identifier *ID* entered into the transportation system at location *loc* at time *t*. The

trajectory of a passenger consists of a sequence of spatio-temporal doublets (or simply doublets) in the form of $\langle loc_i t_i \rangle$. The trajectories can be efficiently constructed by first grouping all (ID, loc, t) entries by *ID* and then sorting them by time *t*. Formally, a trajectory database contains a collection of data records in form of

$$ID, \langle (loc_1 t_1) \rightarrow \cdots \rightarrow (loc_n t_n) \rangle, Y_1, \ldots, Y_m$$

where *ID* is the unique identifier of a passenger (e.g., smart card number), $\langle (loc_1 t_1) \rightarrow \cdots \rightarrow (loc_n t_n) \rangle$ is a trajectory, and $y_i \in Y_i$ are relational attributes, such as job, sex, and age. Following the convention, we assume that explicit identifying information, such as name, SSN, and telephone number, has already been removed. The timestamps in a trajectory increase monotonically. Thus, $\langle a3 \rightarrow c2 \rangle$ is an invalid trajectory. Yet, a passenger may revisit the same location at a different time, so $\langle a3 \rightarrow c7 \rightarrow a9 \rangle$ is a valid trajectory. Given a trajectory database, an adversary can perform identity linkage attacks by matching the trajectories and/or the QID attributes. Many data anonymization techniques (Fung et al., 2007; LeFevre et al., 2006; Machanavajjhala et al., 2007; Samarati and Sweeney, 1998; Xiao and Tao, 2006) have been previously developed for relational QID data; we focus on anonymizing the trajectories in this paper, instead.

3.2. Privacy model

Suppose an adversary who has access to the released trajectory database *T* attempts to identify the record of a target victim *V* in *T*. We adopt the *LK*-privacy model from (Mohammed et al., 2010) and customize it for thwarting identity linkage attacks on *T*. *LK*-privacy is based on the assumption that the attacker knows at most *L* spatio-temporal doublets about the victim, denoted by $q = \langle (loc_1 t_1) \rightarrow \cdots \rightarrow (loc_q t_q) \rangle$, where $0 < |q| \leqslant L$. Using this background knowledge, the adversary can identify a group of records, denoted by $T(q)$, that "contains" q. A record *contains* q if q is a subsequence of the record. For example, in Table 1, the records with $ID\#1, 7, 13$ contain $q = \langle b2 \rightarrow e5 \rangle$.

Definition 3.1 (*Identity linkage attack*). Given background knowledge q about victim $V, T(q)$ is the set of records that contains q. If the group size of $T(q)$, denoted by $|T(q)|$, is small, then the adversary may identify *V*'s record from $T(q)$.

For example, in Table 1, if $q = \langle b2 \rightarrow c9 \rangle$, then $T(q)$ contains $ID\#1$ and $|T(q)| = 1$. The attack learns that ID#1 belongs to the victim; therefore, reveals other visited locations and potentially other relational attributes of the victim. To thwart identity record linkage, *LK*-privacy requires every sequence with a maximum length of *L* in *T* to be shared by at least *K* records.

Definition 3.2 (*LK-privacy*). Let *L* be a user-specified threshold indicating the maximum length of adversary's background knowledge. A trajectory database *T* satisfies *LK-privacy* if and only if for any non-empty sequence q with length $|q| \leqslant L$ in $T, |T(q)| \geqslant K$, where $K > 0$ is a user-specified anonymity threshold.

LK-privacy guarantees that the probability of a successful identity linkage to a victim's record is bounded by $1/K$.

3.3. Passenger probabilistic flowgraph

The measure of information quality varies depending on the data mining task to be performed on the published data. Previous works (Fung et al., 2007; Li and Li, 2009) suggest that anonymization algorithms can be tailored to better preserve information quality if the quality requirement is known in advance. In this paper, we aim at preserving the information quality for supporting effective passenger flow analysis. More specifically, we would like to preserve the passenger flow information in terms of a passenger probabilistic flowgraph generated from the anonymized trajectory data. A passenger flowgraph can reveal hot paths and hot spots in different periods of time that may not be apparent from the raw data. This knowledge is also useful for studying the interactions between passengers and the transportation infrastructures.

Definition 3.3 (*Passenger probabilistic flowgraph*). Let *D* be the set of distinct doublets in a trajectory database *T*. A *passenger probabilistic flowgraph* (or simply *flowgraph*) is a tree in which each node $d \in D$, and each edge is a 2-element doublets $\{d_x, d_y\}$ representing the transition between two nodes, with probability denoted by $prob(d_x \rightarrow d_y)$.

The *transitional probability* $prob(d_x \rightarrow d_y)$ captures the percentage of passengers at doublet d_x who moved to d_y. In case $d_x = d_y$, the probability indicates the percentage of passengers who terminated their journey at d_x. Given a node $d_x, \sum prob(d_x \rightarrow d_y) = 1$ over all out-edges d_y of d_x. For example, in Fig. 1, 50% of the passengers who visited $\langle e5 \rightarrow e7 \rangle$ then visited $c9$. The remaining 50% of passengers terminated their journey at $e7$.

The function $Info(d)$ measures the information quality of a distinct doublet d in a trajectory database *T* with respect to the flowgraph generated from *T*:

$$Info(d) = \alpha(d) \times w_\alpha + \beta(d) \times w_\beta + \gamma(d) \times w_\gamma + \delta(d) \times w_\delta \tag{1}$$

where $\alpha(d)$ is the number of instances of d in the flowgraph, $\beta(d)$ is the total number of child nodes of d in the flowgraph, $\gamma(d)$ is the number of root-to-leaf paths containing d in the flowgraph, $\delta(d)$ is the number of trajectories in a trajectory database *T* containing d. $w_\alpha, w_\beta, w_\gamma$, and w_δ are the weights on α, β, γ, and δ functions, respectively. The weights, $0 \leqslant w_\alpha, w_\beta, w_\gamma, w_\delta \leqslant 1$

and $w_\alpha + w_\beta + w_\gamma + w_\delta = 1$, allow users to adjust the importance of each property according to their required analysis. Similarly, the function $Info(T)$ measures the information quality of a trajectory database T by the summation of the information quality $Info(d)$ over all distinct doublets in T with respect to the flowgraph generated from T.

Example 3.1. Consider doublet $b2$ in Fig. 1. $\alpha(b2) = 3$ because three nodes in the flowgraph contain $b2$. $\beta(b2) = 5$ because the three instances of $b2$ have five child nodes in total. $\gamma(b2) = 6$ because six root-to-leaf paths in the flowgraph contain $b2$. $\delta(b2) = 7$ because seven trajectories in Table 1 contain $b2$. Suppose $w_\alpha = 0.4, w_\beta = 0.2, w_\gamma = 0.2$, and $w_\delta = 0.2$. $Info(b2) = 3 \times 0.4 + 5 \times 0.2 + 6 \times 0.2 + 7 \times 0.2 = 4.8$.

3.4. Problem statement

The problem of *trajectory data anonymization for passenger flow analysis* is defined below.

Definition 3.4. Given a trajectory database T and a user-specified LK-privacy requirement, the problem of *trajectory data anonymization for passenger flow analysis* is to transform T into another version T' such that T' satisfies the LK-privacy requirement with maximal $Info(T')$, i.e., with minimal impact on the passenger probabilistic flowgraph.

4. The anonymization algorithm

Our proposed anonymization algorithm consists of three steps. The first step is to generate the probabilistic flowgraph from the raw trajectory database T. The second step is to identify *all* sequences that violate the given LK-privacy requirement. The third step is to eliminate the violating sequences from T by a sequence of suppressions with the goal of minimizing the impact on the structure of the flowgraph generated in the first step. Each step is further elaborated as follows.

4.1. Generating probabilistic flowgraph

To build a probabilistic flowgraph, the first step is to build a prefix tree from the raw trajectories. Each root-to-leaf path represents a distinct trajectory. Each node maintains a count that keeps track of the number of trajectories sharing the same prefix. The transitional probabilities (Definition 3.3) as well as the $\alpha(d), \beta(d), \gamma(d)$, and $\delta(d)$ (Eq. 1) of each distinct doublet d in the trajectory database can be computed from the counts in the prefix tree. The entire step requires only one scan on the trajectory database records.

4.2. Identifying violating sequences

An adversary may use any non-empty sequence with length not greater than L as background knowledge to perform a linkage attack on the trajectory data. By Definition 3.2, a sequence q with $0 < |q| \leqslant L$ in T is a violating sequence if the number of trajectories in T containing q is less than the user-specified threshold K.

Definition 4.1 (*Violating sequence*). Let q be a sequence of a trajectory in T with $0 < |q| \leqslant L$. q is a *violating sequence* with respect to a LK-privacy requirement if $|T(q)| < K$.

Example 4.1 (*Violating sequence*). Consider Table 1. Given $L = 2$ and $K = 2$, the sequence $q_1 = \langle a1 \rightarrow c9 \rangle$ is a violating sequence because $|q_1| = 2 \leqslant L$ and $|T(q_1)| = 1 < K$. However, the sequence $q_2 = \langle c3 \rightarrow e7 \rightarrow d8 \rangle$ is not a violating sequence even though $|T(q_2)| = 1 < K$ because $|q_2| = 3 > L$.

Enforcing the LK-privacy requirement is equivalent to removing all violating sequences from the trajectory database. An inefficient solution is to first generate all possible violating sequences and then remove them. Consider a violating sequence q that by definition has $|T(q)| < K$. Thus, any super sequence of q in T must also be a violating sequence. Therefore, the number of violating sequences is huge, making this approach infeasible to be applied on real-life trajectory data. Instead, we observe that every violating sequence must contain at least one *minimal violating sequence* and eliminating all minimal violating sequences guarantees to eliminate all violating sequences.

Definition 4.2 (*Minimal violating sequence*). A violating sequence q is a *minimal violating sequence* (*MVS*) if every proper subsequence of q is not a violating sequence (Chen et al., 2013).

Example 4.2 (*Minimal violating sequence*). Consider Table 1. Given $L = 2$ and $K = 2$, the sequence $q_1 = \langle b2 \rightarrow c9 \rangle$ is a MVS because $|T(q_1)| = 1 < K$, and all of its proper subsequences, namely $b2$ and $c9$, are not violating sequences. In contrast, the sequence $q_2 = \langle c3 \rightarrow d4 \rangle$ is a violating sequence but not a MVS because $d4$ is a violating sequence.

Chen et al. (2013) proved that a trajectory database T satisfies $(KC)_L$-privacy if and only if T contains no minimal violating sequence. $(KC)_L$-privacy is a generalized privacy model of LK-privacy, so the same proof is applicable to LK-privacy by setting the confidence threshold $C = 100\%$ in the proof.

Algorithm 1. Identifying minimal violating sequences (MVS)

Input: Raw trajectory database T
Input: Thresholds L, K
Output: Minimal violating sequences MVS

1: $C_1 \leftarrow$ all distinct doublets in T;
2: $i \leftarrow 1$;
3: **while** $i \leqslant L$ and $C_i \neq \emptyset$ **do**
4: Scan T once to compute $|T(q)|$, for $\forall q \in C_i$;
5: **for** $\forall q \in C_i$ where $|T(q)| > 0$ **do**
6: **if** $|T(q)| < K$ **then**
7: $MVS_i = MVS_i \cup \{q\}$;
8: **else**
9: $NVS_i = NVS_i \cup \{q\}$;
10: **end if**
11: i++;
12: **end for**
13: $C_i \leftarrow NVS_{i-1} \bowtie NVS_{i-1}$;
14: **for** $\forall q \in C_i$ **do**
15: **if** $\exists v \in MVS_{i-1}$ such that $q \sqsupset v$ **then**
16: $C_i = C_i - \{q\}$;
17: **end if**
18: **end for**
19: **end while**
20: **return** $MVS = MVS_1 \cup \cdots \cup MVS_{i-1}$;

Algorithm 1 presents a procedure to identify all minimal violating sequences, MVS, with respect to a given LK-privacy requirement. First, C_1 contains all distinct doublets, representing the set of candidate sequences with length 1. Then it scans the trajectory database T once to count the support of each sequence q in C_i (Line 4). Then for each q in C_i, if $|T(q)|$ is less than K, it is added to MVS_i (Line 7); otherwise, it is added to NVS_i (Line 9), which will be used to generate the next candidate set C_i in the next iteration. Generating the next candidate set consists of two steps. First, conduct a self-join of the non-violating sequence set, NVS_{i-1} (Line 13). Two sequences $q_x = (loc_1^x t_1^x) \rightarrow \cdots \rightarrow (loc_i^x t_i^x)$ and $q_y = (loc_1^y t_1^y) \rightarrow \cdots \rightarrow (loc_i^y t_i^y)$ can be joined if the first $i-1$ doublets are identical and $t_i^x < t_i^y$. The joined sequence is $(loc_1^x t_1^x) \rightarrow \cdots \rightarrow (loc_i^x t_i^x) \rightarrow (loc_i^y t_i^y)$. This definition assures that all candidates from self-join would be generated only once. Second, for each q in C_i, if q is a super sequence of any sequence in MVS_{i-1}, q will be removed from C_i (Lines 14–18) because by definition q cannot be a minimal violating sequence. Line 20 returns all minimal violating sequences.

Example 4.3. Given $L = 2$ and $K = 2$, the MVS set generated from Table 1 is $MVS(T) = \{d4, a1 \rightarrow c9, b2 \rightarrow c9, c3 \rightarrow c9\}$.

4.3. Removing violating sequences

After all minimal violating sequences are identified, the next step is to eliminate them with the goal of minimizing the impact on information quality for passenger flow analysis. However, finding an optimal solution based on suppression for LK-privacy is *NP-hard* (Chen et al., 2013). Thus, we propose a greedy algorithm to efficiently eliminate minimal violating sequences with a reasonably good sub-optimal solution.

Suppressing a doublet generally increases privacy and decreases information quality. Intuitively, a doublet d is a good candidate for suppression if suppressing it would result in eliminating a large number of MVS and minimal impact on the passenger flowgraph. Eq. (2) measures the goodness of suppressing a doublet d:

$$Score1(d) = \frac{Privgain(d)}{Info(d)} \tag{2}$$

where $PrivGain(d)$ is the number of MVS that can be eliminated by suppressing d and $Info(d)$ measures the information quality of a doublet d defined in Eq. 1. The greedy function considers both data privacy and information quality simultaneously by selecting a suppression with the maximum privacy gain per unit of information loss.

We also define three other functions for comparison. $Score2(d)$ randomly selects a doublet for suppression without considering $PrivGain(d)$ and $Info(d)$:

$$Score2(d) = 1 \tag{3}$$

$Score3(d)$ aims at maximizing $PrivGain(d)$ without considering $Info(d)$:

$$Score3(d) = PrivGain(d) \tag{4}$$

$Score4(d)$ aims at minimizing loss of $Info(d)$ without considering $PrivGain(d)$:

$$Score4(d) = \frac{1}{Info(d)} \tag{5}$$

Algorithm 2. Check validity of a local suppression

Input: Trajectory database T
Input: Thresholds L, K
Input: A doublet d in a minimal violating sequence m
Output: A boolean indicating if locally suppressing d from m is valid
1: $D\prime \leftarrow \{d' | d' \in D, d' \in T(m), d' \in (T(d) - T(m))\}$;
2: $MVS1 \leftarrow \{m1 | m1 \in MVS, |m1| = 1\}$
3: $MVS' \leftarrow \{m' | m' \in MVS, d \in m, MVS(d)\} \cup MVS1$;
4: Remove all doublets, except for d, in MVS' from D';
5: $Q \leftarrow$ all possible sequences with size $\leqslant L$ generated from d after removing super sequences of the sequences in $MVS - T(d)$;
6: Scan $T(d) - T(m)$ once to compute $| q |$;
7: **for** each sequence $q \in Q$ with $| q | > 0$ **do**
8: **if** $| q | < K$ **then**
9: **return** false;
10: **end if**
11: **end for**
12: **return** true;

Most of the previous works on trajectory anonymization (Fung et al., 2009a,b; Mohammed et al., 2009) employ global suppression, which guarantees that globally suppressing a doublet d does not generate new MVS. In other words, the number of MVS monotonically decreases with respect to a sequence of suppressions (Chen et al., 2013). Yet, local suppression does not share the same property. For example, locally suppressing $b2$ from $ID\#1$ in Table 1 generates a new MVS $\langle a1 \rightarrow b2 \rangle$ because the support $|T(a1 \rightarrow b2)| = 2$ decreases to $|T\prime(a1 \rightarrow b2)| = 1 < K$, where $T\prime$ is the database resulted from the local suppression. Identifying the newly generated MVS is an expensive computational process, and there is no guarantee that the anonymization process can be completed within $|MVS|$ number of iterations. To overcome this challenge, a local suppression is performed only if it does not generate any new MVS.

Definition 4.3 (*Valid local suppression*). A local suppression over a trajectory database is *valid* if it does not generate any new MVS (Chen et al., 2013).

Algorithm 2 checks the validity of suppressing a doublet d from a minimal violating sequence m. Let D' be the set of distinct doublets that coexist in both $T(m)$ and $T(d) - T(m)$ (Line 1). Let $MVS1$ be the set of size-one MVS (Line 2). Let MVS' be the union of MVS containing d and $MVS1$ (Line 3). Line 4 then removes all doublets, except for d, in MVS' from D' because such a doublet is already a MVS, or a subsequence of a MVS, and is not a future MVS candidate. Line 5 generates all possible candidates, which can be new MVS. Line 6 scans all records containing d to compute $|q|$ for each $q \in Q$. For each q in Q whose length is less than K, the algorithm returns false, indicating an invalid local suppression.

Algorithm 3. Anonymize trajectory data

Input: Trajectory database T
Input: Thresholds L, K
Output: Anonymous T' satisfying the given LK-privacy requirement
1: Generate *Flowgraph* from database T;
2: Generate $MVS(T)$ by Algorithm 1;
3: Build *Score* table by Algorithm 2;
4: **while** *Score* table $\neq\emptyset$ **do**
5: Select a doublet d with the highest score from its MVS m;
6: **if** d is a local suppression **then**
7: $MVS' \leftarrow \{m' | m' \in MVS, d \in m' \wedge T(m') = T(m)\}$;
8: Suppress the instances of d from $T(m)$;
9: **else**
10: $MVS' \leftarrow MVS(d)$;
11: Suppress all instances of d in T;
12: **end if**
13: Update the $Score(d')$ if both d and d' are in MVS';
14: $MVS = MVS - MVS'$;
15: **end while**
16: **return** the suppressed T as T';

Algorithm 3 summarizes the entire anonymization algorithm. Line 1 generates the flowgraph from the trajectory database, which is then needed to compute *Info* of doublets. Line 2 calls Algorithm 1 to generate all the minimal violating sequences *MVS*. Line 3 calls Algorithm 2 to calculate the score of all doublet instances and stores the results in the *Score* table. In each iteration, a doublet d with the highest score from its MVS m is selected. If the selected suppression d is a local suppression, then Line 7 identifies the set of MVS, denoted by MVS', that will be eliminated due to locally suppressing d, and Line 8 removes the instances of d from the records $T(m)$. If the selected suppression d is a global suppression, then Line 10 identifies the set of MVS, denoted by MVS', that contains d, and Line 11 suppresses all instances of d from T. Line 13 updates the *Score* table for the next round and Line 14 removes the suppressed MVS of d from *MVS*. The algorithm repeats these operations until the *Score* table becomes empty.

4.4. Discussion

Transitional probabilities in a flowgraph are conditional probabilities of the next visited doublets given the passengers' previously visited doublets. Suppressing some instances of a doublet directly affects its related transitional probabilities. Thus, the distortion of transitional probabilities caused by a suppression is indirectly reflected by $\delta(d)$ in the information quality measure $Info(d)$ (Eq. 1). Yet, to *precisely* capture the distortion of transitional probabilities, $Info(d)$ has to be redefined as follows. Let $\{c_1, \ldots, c_m\}$ and $\{c'_1, \ldots, c'_m\}$ be the counts of the child nodes of a node before and after suppression, respectively. The distortion of transitional probabilities due to locally suppressing y instances of c_j is:

$$\sum_i \left| \frac{c_i}{\sum_i c_i} - \frac{c'_i}{\sum_i c'_i} \right| = \sum_i \left| \frac{c_i \sum_i c'_i - c'_i \sum_i c_i}{\sum_i c_i \sum_i c'_i} \right| \tag{6}$$

Since for $i \neq j, c_i = c'_i$, and for $j, c_j - y = c'_j$, we have:

$$\sum_i \left| \frac{c_i \sum_i c'_i - c'_i \sum_i c_i}{\sum_i c_i \sum_i c'_i} \right| = \sum_{i \neq j} \left| \frac{c_i y}{\sum_i c_i (\sum_i c_i - y)} \right| + \left| \frac{y(\sum_i c_i - c_j)}{\sum_i c_i (\sum_i c_j - y)} \right| = \frac{2y(\sum_i c_i - c_j)}{\sum_i c_i (\sum_i c_i - y)} \tag{7}$$

If an item corresponding to c_j occurs in multiple nodes in a flowgraph, then its distortion should be the sum of all these nodes. For global suppression, the equation can be simplified to $\frac{2c_j}{\sum_i c_i}$. Eq. 7 can serve as an information quality measure tailored for quantifying the distortion of transitional probabilities. Yet, the objective of this paper is to preserve the overall flowgraph structure, not limited to preserve only transitional probabilities, after data anonymization. Thus, Eq. (1) is the information quality measure employed in the rest of the paper.

Next, we analyze the computational complexity of our anonymization algorithm. The proposed algorithm consists of three steps. The first step is to generate the flowgraph, which requires one scan on the trajectory database to build a prefix tree. The second step is to identify all MVS in which a good approximation is $O(|s|^2)$, where s is the number of distinct doublets. The worst case scenario is $O(|s|^2|T|)$ (Chen et al., 2013). The third step is to remove all MVS, which is also bounded by $O(|s|^2|T|)$ Chen et al. (2013). In addition to the theoretical analysis above, the scalability of our algorithm is further experimentally validated in Section 5.2.

Table 3
Experimental data set statistics.

Data sets	# Of records $\|T\|$	# Of dimensions $\|s\|$	Data size (Kbytes)	Data type
Metro200K	200,000	696	12,359	Synthetic
STM514K	514,213	3120	12,910	Real-life

5. Experimental evaluation

The experimental evaluation serves two purposes. First, we want to evaluate the impact of anonymization on the information quality of the flowgraph with respect to different privacy parameters and weights. Second, we want to evaluate the efficiency of our proposed algorithm.

To evaluate the impact of anonymization, we introduce a new similarity metric $\varphi(G, G')$ to measure the similarity between the flowgraph G generated from the raw trajectory data and the flowgraph G' generated from the anonymized trajectory data. Algorithm 4 illustrates the procedure for computing $\varphi(G, G')$. First, all distinct doublets of each flowgraph are sorted by time and location (Lines 1–3). Then for each pair of identical doublets $d \in G$ and $d' \in G'$, the algorithm computes $\alpha(d), \beta(d), \gamma(d), \delta(d), \alpha(d'), \beta(d'), \gamma(d')$, and $\delta(d')$, calculates the ratios among them, and then sums up the ratios, denoted by $aSum, bSum, cSum$, and $dSum$ (Lines 5–18), respectively. In case d is a leaf node, $\beta(d) = 0$. To avoid dividing by zero, Line 9 skips the division, uses the counter i to keep track of the number of doublets having $\beta(d) = 0$, and subtracts i from the total number of distinct doublets in Line 19. Line 20 returns the similarity measure φ, which is a weighted sum of the ratios.

Algorithm 4. Comparing two flowgraphs

Input: Flowgraph G
Input: Flowgraph G'
Input: Weights $w_\alpha, w_\beta, w_\gamma$
Output: Similarity measure φ
1: $UL \leftarrow \{d | d \in G\}$;
2: $UL' \leftarrow \{d' | d' \in G'\}$;
3: Sort UL and UL' by time and location;
4: $i \leftarrow 0$;
5: **for** each $d \in UL$ **do**
6: **for** each $d' \in UL'$ **do**
7: **if** $d = d'$ **then**
8: $aSum \mathrel{+}= \frac{\alpha(d')}{\alpha(d)}$;
9: **if** $\beta(d) \neq 0$ **then**
10: $bSum \mathrel{+}= \frac{\beta(d')}{\beta(d)}$;
11: **else**
12: $i{+}{+}$;
13: **end if**
14: $cSum \mathrel{+}= \frac{\gamma(d')}{\gamma(d)}$;
15: $dSum \mathrel{+}= \frac{\delta(d')}{\delta(d)}$;
16: **end if**
17: **end for**
18: **end for**
19: $\varphi \leftarrow \frac{aSum}{|G|} \times w_\alpha + \frac{bSum}{|G|-i} \times w_\beta + \frac{cSum}{|G|} \times w_\gamma + \frac{dSum}{|G|} \times w_\delta$;
20: **return** φ;

We could not directly compare our proposed algorithm with previous works (Abul et al., 2008; Chen et al., 2013; Pensa et al., 2008; Terrovitis and Mamoulis, 2008; Yarovoy et al., 2009) on trajectory data anonymization because their proposed solutions do not consider preserving information in a passenger flowgraph. Thus, we compare our results with the results generated from K-anonymous data.

Two data sets, *Metro200K* and *STM514K*, are used in the experiments. *Metro200K* is a data set simulating the travel routes of 200,000 passengers in the Montréal subway transit system with 29 stations in 24 h, forming 696 dimensions. *STM514K* is a *real-life* data set provided by *Société de transport de Montréal* (STM).[1] It contains the transit data of 514,213 passengers among 65 subway stations within 48 h, where the time granularity is set to the hour level. The properties of the two experimental data sets are summarized in Table 3.

[1] http://www.stm.info.

5.1. Information quality

We evaluate the information quality by calculating the similarity of the raw flowgraph and the anonymized flowgraph in terms of varying K, L, and weights. We also show the benefit of a reasonable L value over the traditional K-anonymity in combination with other parameters. In real-life passenger flow analysis, an analyst may want to emphasize preserving different properties in a passenger flowgraph by adjusting the weights. Thus, we create three scenarios with different weights.

5.1.1. Scenario I

Subway stations provide a unique opportunity for out-of-home marketing. Suppose that a company is granted permission to display their advertisements in the subway stations. The company may request the metro company to share the anonymized trajectory data for research purposes. In this case, it is reasonable to put more emphasis on α, which represents the number of instances of each station in the flowgraph. Accordingly, we set $w_\alpha = 0.5, w_\beta = 0.3, w_\gamma = 0.2$, and $w_\delta = 0$.

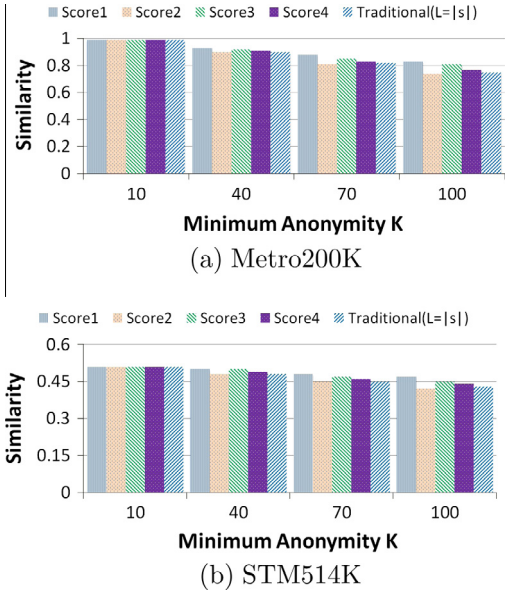

Fig. 4. Scenario I: Similarity vs. K ($L = 3, w_\alpha = 0.5, w_\beta = 0.3, w_\gamma = 0.2, w_\delta = 0$).

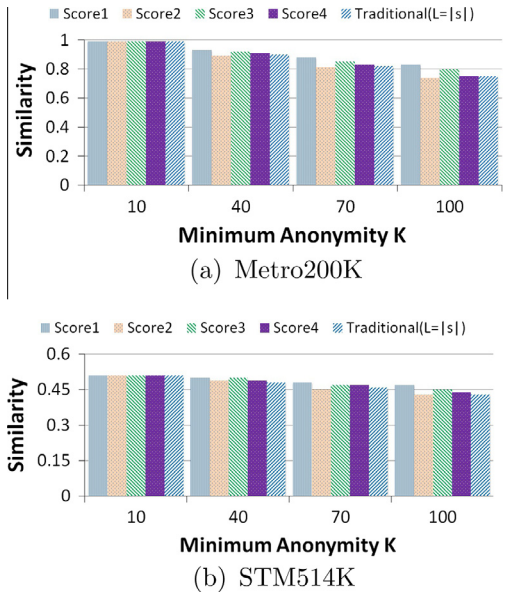

Fig. 5. Scenario II: Similarity vs. K ($L = 3, w_\alpha = 0.3, w_\beta = 0.5, w_\gamma = 0.2, w_\delta = 0$).

Fig. 4(a) depicts the similarity measure φ of the two flowgraphs before and after the anonymization for $L = 3$ and $10 \leqslant K \leqslant 100$, with different *Score* functions on the *Metro200K* data set. When $K = 10$, the similarity is 0.99, indicating that almost no information has been lost in terms of the flowgraph. As K increases, the similarity decreases. This shows the trade-off between data privacy and the information quality of the flowgraph. The results of K-anonymity are achieved by setting $L = |s|$, where $|s|$ is the number of distinct doublets in the given data set. The experimental results suggest that applying LK-privacy does produce less information loss than applying traditional K-anonymity, with respect to passenger flow analysis. To show that the benefit is statistically significant, we conduct a one-tail t-test on the 10 pairs of test cases from $10 \leqslant K \leqslant 100$. The p-values for $Score1, Score2, Score3$, and $Score4$ in Fig. 4(a) are 1.75E−3, 1.28E−2, 5.67E−4, and 1.58E−3, respectively. Fig. 4(b) depicts the similarity measure φ of the flowgraphs before and after the anonymization for $L = 3$ and $10 \leqslant K \leqslant 100$ with different *Score* functions on the *STM514K* data set. Similar trends can be observed. The p-values for $Score1, Score2, Score3$, and $Score4$ in Fig. 4(b) are 2.83E−3, 1.09E−2, 3.8E−4, and 2.18E−2, respectively, showing that the benefit is statistically significant at $\alpha = 5\%$.

5.1.2. Scenario II

In this scenario, the weights are set at $w_\alpha = 0.3, w_\beta = 0.5, w_\gamma = 0.2$, and $w_\delta = 0$ with $L = 3$ and $10 \leqslant K \leqslant 100$. The results in Fig. 5(a) and (b) in this scenario our proposed algorithm still performs best, suggesting that our method is robust against different weights and different scenarios of flowgraph analysis. The behavior of our algorithm is similar in both scenarios. For example, in both scenarios we have almost the same results for $K = 70$, even though the weight w_α in Scenario I is much higher than the weight w_α in Scenario II.

The results further confirm that our score functions in general produce better information quality than K-anonymity, except for *Score*2, which suppresses MVS randomly. To show that the benefit of our proposed algorithm over K-anonymity is significant, we conducted a one-tail t-test on 10 pairs of test cases from $10 \leqslant K \leqslant 100$. The p-values for $Score1, Score2, Score3$, and $Score4$ in Fig. 5(a) are 4.75E-3, 2.8E-3, 4.67E-3, and 9.08E-3, respectively. The p-values for $Score1, Score2, Score3$, and $Score4$ in Fig. 5(b) are 3.98E−3, 5.0E−2, 4.5E−3, and 2.88E−3, respectively, showing that the benefit is statistically significant at $\alpha = 5\%$.

5.1.3. Scenario III

In the final scenario, all weights are equally set to 0.25 with $L = 3$ and $10 \leqslant K \leqslant 100$. The results in Fig. 6(a) and (b) suggest that our proposed algorithm yields less information loss than K-anonymity. The results also suggest that distributing equal weights preserves higher information quality of the flowgraph than the previous two scenarios. To further show that the benefit of our proposed algorithm over K-anonymity is significant, we conducted a one-tail t-test on 10 pairs of test cases from $10 \leqslant K \leqslant 100$. The p-values for $Score1, Score2, Score3$, and $Score4$ in Fig. 6(a) are 4.95E−3, 1.91E−3, 5.01E−3, and 4.58E−3, respectively. The p-values for $Score1, Score2, Score3$, and $Score4$ in Fig. 6(b) are 3.45E−3, 9.08E−2, 2.88E−3, and 4.58E−3, respectively, showing that the benefit is statistically significant at $\alpha = 5\%$.

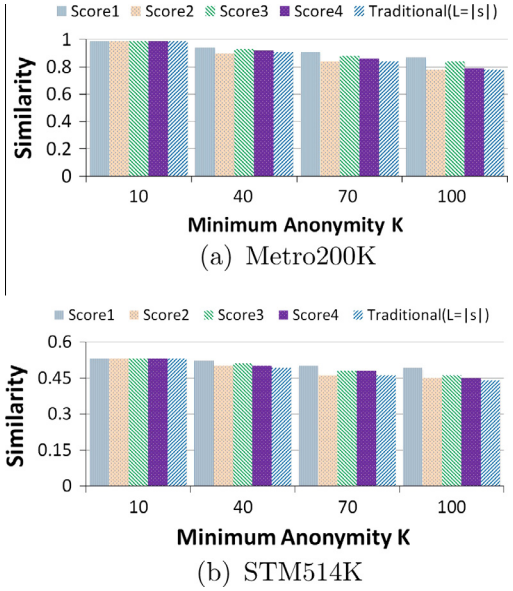

Fig. 6. Scenario III: Similarity vs. K ($L = 3, w_\alpha = 0.25, w_\beta = 0.25, w_\gamma = 0.25, w_\delta = 0.25$).

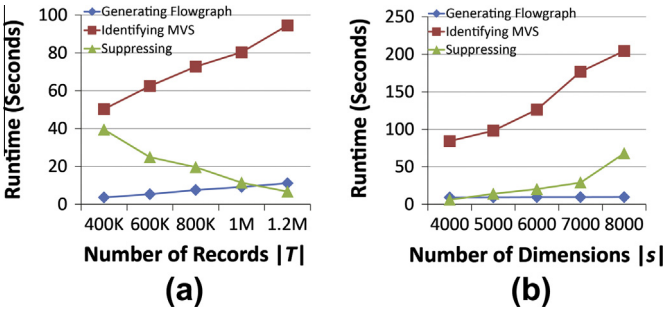

Fig. 7. Scalability.

5.2. Scalability

Next, we demonstrate the scalability of our proposed algorithm on a relatively large trajectory data set. The setting is similar to *Metro200K* but of larger size. Since the complexity is dominated by the number of dimensions $|s|$ and the number of records $|T|$, we examine the performance of our framework with respect to $|s|$ and $|T|$.

5.2.1. Effect of number of records $|T|$

Fig. 7(a) illustrates the runtime of our algorithm on a data set with 4000 dimensions and sizes ranging from 400,000 records to 1,200,000 records. We observe that the runtime for generating the flowgraph is linear and proportional to the number of records. The algorithm takes less than 15 s to generate the flowgraph from 1.2 million records. As $|T|$ increases, the runtime of identifying MVS also increases linearly. The runtime of suppression, however, decreases rapidly as the number of records increases. This is due to the fact that when the number of records increases, there is a substantial reduction in the number of MVS; therefore, it takes less time to suppress them.

5.2.2. Effect of dimensionality $|s|$

Fig. 7(b) depicts the runtime of our algorithm on a data set of 1 million records, with the number of dimensions (number of distinct doublets) ranging from 4000 to 8000. The figure shows that increasing the number of dimensions has no significant effect on the runtime of flowgraph generation. However, when the number of dimensions increases, the runtime of identifying MVS increases because increasing the number of dimensions introduces a larger number of distinct sequences, which in turn increases the number of MVS and the runtime for removing them.

6. Conclusion

In this paper, we study the problem of anonymizing high-dimensional trajectory data for passenger flow analysis. We demonstrate that applying traditional *K*-anonymity on the trajectory data is not effective for flow analysis. Thus, we adapt the *LK*-privacy model for trajectory data anonymization. We present an anonymization algorithm that thwarts identity record linkages while effectively preserving the information quality for generating a probabilistic passenger flowgraph. The originality of our approach derives from the utilization of the probabilistic flowgraph as the measure of information quality in the anonymization process. Extensive experimental results on both real-life and synthetic passenger trajectory data suggest that data privacy can be achieved without compromising the information quality of passenger flowgraph analysis.

Acknowledgment

The research is supported in part by NSERC through Discovery Grants (356065-2013).

References

Abraham, S., Lal, P.S., 2012. Spatio-temporal similarity of network-constrained moving object trajectories using sequence alignment of travel locations. Transport. Res. Part C: Emerg. Technol. 23, 109–123.

Abul, O., Bonchi, F., Nanni, M., 2008. Never walk alone: uncertainty for anonymity in moving objects databases. In: Proceedings of the 24th IEEE International Conference on Data Engineering, pp. 376–385.

Aggarwal, C.C., 2005. On *k*-anonymity and the curse of dimensionality. In: Proceedings of the 31st International Conference on Very Large Data Bases, pp. 901–909.

Burger, M., van den Berg, M., Hegyi, A., De Schutter, B., Hellendoorn, J., 2013. Considerations for model-based traffic control. Transport. Res. Part C: Emerg. Technol. 35, 1–19.

Cao, J., Karras, P., 2012. Publishing microdata with a robust privacy guarantee. Proc. VLDB Endowm. 5 (11), 1388–1399.

Chen, R., Mohammed, N., Fung, B.C.M., Desai, B.C., Xiong, L., 2011. Publishing set-valued data via differential privacy. Proc. VLDB Endowm. 4 (11), 1087–1098.

Chen, R., Acs, G., Castelluccia, C., 2012. Differentially private sequential data publication via variable-length n-grams. In: Proceedings of the ACM Conference on Computer and Communications Security, pp. 638–649.
Chen, R., Fung, B.C.M., Desai, B.C., Sossou, N.M., 2012. Differentially private transit data publication: a case study on the montreal transportation system. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 213–221.
Chen, R., Fung, B.C.M., Mohammed, N., Desai, B.C., Wang, K., 2013. Privacy-preserving trajectory data publishing by local suppression. Inform. Sci. 231, 83–97.
Clifton C., Tassa, T., 2013. On syntactic anonymity and differential privacy. In: Proceeding of the 29th IEEE International Conference on Data Engineering Workshops (ICDEW), pp. 88–93.
Dwork, C., 2006. Differential privacy. In: Proceedings of the 33rd International Colloquium on Automata, Languages and Programming (ICALP), pp. 1–12.
Fung, B.C.M., Wang, K., Yu, P.S., 2007. Anonymizing classification data for privacy preservation. IEEE Trans. Knowl. Data Eng. (TKDE) 19 (5), 711–725.
Fung, B.C.M., Al-Hussaeni, K., Cao, M., 2009. Preserving RFID data privacy. In: Proceedings of the 2009 IEEE International Conference on RFID, pp. 200–207.
Fung, B.C.M., Cao, M., Desai, B.C., Xu, H., 2009. Privacy protection for RFID data. In: Proceedings of the 2009 ACM Symposium on Applied Computing, pp. 1528–1535.
Fung, B.C.M., Wang, K., Chen, R., Yu, P.S., 2010. Privacy-preserving data publishing: a survey of recent developments. ACM Comput. Surv. (CSUR) 42 (4), 14.
Ghinita, G., Tao, Y., Kalnis, P., 2008. On the anonymization of sparse high-dimensional data. In: Proceedings of the 24th IEEE International Conference on Data Engineering (ICDE), pp. 715–724.
Giannotti, F., Nanni, M., Pinelli, F., Pedreschi, D., 2007. Trajectory pattern mining. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 330–339.
Halb, W., Neuschmied, H., 2009. Multimodal semantic analysis of public transport movements. In: Proceedings of 4th International Conference on Semantic and Digital Media Technologies (SAMT), pp. 165–168.
He, Y., Naughton, J.F., 2009. Anonymization of set-valued data via top-down, local generalization. Proc. VLDB Endowm. 2 (1), 934–945.
Herrera, J.C., Work, D.B., Herring, R., Ban, X.J., Jacobson, Q., Bayen, A.M., 2010. Evaluation of traffic data obtained via GPS-enabled mobile phones: the mobile century field experiment. Transport. Res. Part C: Emerg. Technol. 18, 568–583.
Hu, H., Xu, J., On, S.T., Du, J., NG, J.K., 2010. Privacy-aware location data publishing. ACM Trans. Database Syst. (TODS) 35 (3), 18.
Kisilevich, S., Rokach, L., Elovici, Y., Shapira, B., 2010. Efficient multidimensional suppression for k-anonymity. IEEE Trans. Knowl. Data Eng. 22 (3), 334–347.
Lee, J.G., Han, J., Whang, K.-Y., 2007. Trajectory clustering: a partition-and-group framework. InL Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data, pp. 593–604.
Lee, J.G., Han, J., Li, X., Gonzalez, H., 2008. Traclass: trajectory classification using hierarchical region-based and trajectory-based clustering. Proc. VLDB Endowm. 1 (1), 1081–1094.
LeFevre, K., DeWitt, D.J., Ramakrishnan, R., 2006. Mondrian multidimensional k-anonymity. In: Proceedings of the 22nd International Conference on Data Engineering (ICDE), p. 25.
Li, T., Li, N., 2008. Injector: mining background knowledge for data anonymization. In: Proceedings of the 24th IEEE International Conference on Data Engineering, pp. 446–455.
Li, T., Li, N., 2009. On the tradeoff between privacy and utility in data publishing. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 517–526.
Li, N., Li, T., Venkatasubramanian, S., 2007. t-closeness: privacy beyond k-anonymity and ℓ-diversity. In: Proceedings of the 23rd IEEE International Conference on Data Engineering, pp. 106–115.
Li, X., Han, J., Lee, J.-G., Gonzalez, H., 2007. Traffic density-based discovery of hot routes in road networks. In: Proceedings of 10th International Symposium on Advances in Spatial and Temporal Databases, pp. 441–459.
Machanavajjhala, A., Kifer, D., Gehrke, J., Venkitasubramaniam, M., 2007. ℓ-diversity: privacy beyond k-anonymity. ACM Trans. Knowl. Discov. Data (TKDD) 1 (1), 3.
Machanavajjhala, A., Gehrke, J., Götz, M., 2009. Data publishing against realistic adversaries. Proc. VLDB Endowm. 2 (1), 790–801.
Matatov, N., Rokach, L., Maimon, O., 2010. Privacy-preserving data mining: a feature set partitioning approach. Inform. Sci. 180 (14), 2696–2720.
Mohammed, N., Fung, B.C.M., Debbabi, M., 2009. Walking in the crowd: anonymizing trajectory data for pattern analysis. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management, pp. 1441–1444.
Mohammed, N., Fung, B.C.M., Hung, P.C.K., Lee, C.-K., 2010. Centralized and distributed anonymization for high-dimensional healthcare data. ACM Trans. Knowl. Discov. Data (TKDD) 4 (4), 18:1–18:33.
Monreale, A., Andrienko, G., Andrienko, N., Giannotti, F., Pedreschi, D., Rinzivillo, S., Wrobel, S., 2010. Movement data anonymity through generalization. Trans. Data Privacy 3 (2), 91–121.
Paletta, L., Wiesenhofer, S., Brandle, N., Sidla, O., Lypetskyy, Y., 2005. Visual surveillance system for monitoring of passenger flows at public transportation junctions. In: Proceedings of the 2005 IEEE Intelligent Transportation Systems, pp. 862–867.
Pelletier, M.-P., Trépanier, M., Morency, C., 2011. Smart card data use in public transit: a literature review. Transport. Res. Part C: Emerg. Technol. 19, 557–568.
Pensa, R.G., Monreale, A., Pinelli, F., Pedreschi, D., 2008. Pattern-preserving k-anonymization of sequences and its application to mobility data mining. Privacy Location-Based Appl., 44.
Samarati, P., Sweeney, L., 1998. Generalizing data to provide anonymity when disclosing information. In: Proceedings of the 17th ACM SIGACT-SIGMOD SIGART Symposium on Principles of Database Systems (PODS), vol. 17, pp. 188–188
Tang, L.-A., Zheng, Y., Yuan, J., Han, J., Leung, A., Hung, C.-C., Peng, W.-C., 2012. On Discovery of Traveling Companions from Streaming Trajectories. In: Proceedings of the 28th IEEE International Conference on Data Engineering (ICDE), pp. 186-197.
Tassa, T., Mazza, A., Gionis, A., 2012. k-Concealment: an alternative model of k-type anonymity. Trans. Data Privacy 5 (1), 189–222.
Terrovitis, M., Mamoulis, N., 2008. Privacy preservation in the publication of trajectories. In: Proceedings of the 9th International Conference on Mobile Data Management (MDM), pp. 65–72.
Terrovitis, M., Mamoulis, N., Kalnis, P., 2008. Privacy-preserving anonymization of set-valued data. Proc. VLDB Endowm. 1 (1), 115–125.
Trujillo-Rasua, R., Domingo-Ferrer, J., 2013. On the privacy offered by (k,δ)-anonymity. Inform. Syst. 38 (4), 491–494.
Wang, K., Fung, B.C.M., Yu, P.S., 2007. Handicapping attacker's confidence: an alternative to k-anonymization. Knowl. Inform. Syst. 11 (3), 345–368.
Wong, R., Li, J., Fu, A., Wang, K., 2007. (α,k)-Anonymous data publishing. J. Intell. Inform. Syst. 33 (2), 209–234.
Xiao, X., Tao, Y., 2006. Personalized privacy preservation. In: Proceedings of the 32nd ACM SIGMOD International Conference on Management of Data (SIGMOD), pp. 229–240.
Xu, Y., Fung, B.C.M., Wang, K., Fu, A.W.-C., Pei, J., 2008. Publishing sensitive transactions for itemset utility. In: Proceedings of the 8th IEEE International Conference on Data Mining (ICDM), pp. 1109–1114.
Xu, Y., Wang, K., Fu, A.W.-C., Yu, P.S., 2008. Anonymizing transaction databases for publication. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD), pp. 767–775.
Yang, Y., Zhang, Z., Miklau, G., Winslett, M., Xiao, X., 2012. Differential privacy in data publication and analysis. In: Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, pp. 601–606.
Yarovoy, R., Bonchi, F., Lakshmanan, L.V., Wang, W.H., 2009. Anonymizing moving objects: how to hide a MOB in a crowd? In: Proceedings of the 12th International Conference on Extending Database Technology (EDBT), pp. 72–83.
Zheng, K., Zheng, Y., Yuan, N.J., Shang, S., 2013. On discovery of gathering patterns from trajectories. In Proceeding of 2013 IEEE International Conference on Data Engineering (ICDE).