PRIVACY PRESERVATION IN HIGH-DIMENSIONAL TRAJECTORY DATA FOR PASSENGER FLOW ANALYSIS

MOEIN GHASEMZADEH

A THESIS

IN

THE CONCORDIA INSTITUTE FOR INFORMATION SYSTEMS ENGINEERING

PRESENTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF MASTER OF APPLIED SCIENCE IN INFORMATION SYSTEMS SECURITY

CONCORDIA UNIVERSITY

MONTRÉAL, QUÉBEC, CANADA

SEPTEMBER 2013

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Moein Ghasemzadeh**

Entitled: **Privacy Preservation in High-Dimensional Trajectory Data for Passenger Flow Analysis**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science in Information Systems Security

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

Dr. Amr Youssef	Chair
Dr. Mohammad Mannan	Examiner
Dr. Samar Abdi	External Examiner
Dr. Benjamin C. M. Fung	Supervisor
Dr. Anjali Awasthi	Supervisor

Approved by Graduate Program Director

2013

Dr. Christopher Trueman, Dean

Faculty of Engineering and Computer Science

Abstract

Privacy Preservation in High-Dimensional Trajectory Data for Passenger Flow Analysis

Moein Ghasemzadeh

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 thesis 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.

Acknowledgments

I would like to express my sincere gratitude to my supervisor, Dr. Benjamin C. M. Fung, for his continuous support of my graduate study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me throughout the research and writing of this thesis. I could not imagine having a better supervisor and mentor for my master's study.

Also, I would like to thank my supervisor, Dr. Anjali Awasthi, for providing the opportunities throughout my graduate study and leading me to working on my thesis.

I would like to offer my special thanks to Dr. Rui Chen for providing me with the files I needed for my research and also for his helpful and valuable comments on my thesis.

I would like to express my deep gratitude to my dear sister Anahita, who is not just a sister, but a true friend to me. I am always so thankful for her guidance and support during my whole life. Also, I would like to thank my siblings Elham, Morteza, Mojtaba, and my in-laws for their unconditional support.

I wish to thank my dear aunt Firuzeh and her great husband Javad, who provided such a warm and loving environment for me. They were particularly supportive, reliable, and always made me feel like I was at home.

Last but not least, I like to thank my parents, who have always been there for me when I needed them. Their love provided my inspiration and was my driving force. I owe them everything I have and I wish I could show them just how much I love and appreciate them. I will be grateful forever for their love and their support. To them I dedicate this thesis.

"Life is what happens while you are busy making other plans." - John Lennon

To my beloved family

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

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, which has several advantages compared to the previous systems. For instance, it has seamless integration with the other transit systems of neighbouring cities. Another advantage is the speed at which users can access the system. As opposed to the magnetic stripe cards previously in use, the contactless smart card is more user-friendly and not only will reduce the risk of becoming demagnetized and rendered useless, but it also does not require patrons to slide the card in a particular way. More importantly, senior and junior passengers register their personal information when they first purchase their cards so that an appropriate fare is charged based on their status.

Automated turnstiles are in place at SCAFC stations to ensure that only people with valid tickets may access the transport. Consequently, passengers leave a trace of reading every time they scan a SCAFC card. A data record in the form of (ID, loc, t), which identifies the passenger's identity, location, and time, is then stored in a central database. The trajectory of a passenger can be represented by a sequence of visited locations, sorted by time.

New constructions occur and new trends emerge as a city evolves. Passenger flow is not static and is subject to change depending on all these uncertainties and developments. Transit companies need to ensure their services evolve with the needs of their passengers and help shape better service in their growth. Hence, 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 flow in an area, with the goal of supporting trajectory data mining [19, 26, 27, 50, 63] and traffic management [33]. By using a probabilistic flowgraph, as shown in Figure 2, an analyst can identify the major trends in passenger flow and hot paths in a traffic network. For example, Figure 2 suggests that 67 percent of passengers who started their journey at location a with timestamp 1 visited location b with timestamp 2. However, sharing passenger-specific trajectory data raises new privacy 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 privacy attack, called an *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 and timestamps. Preventing identity linkage attack is very important in trajectory

Table 1: Raw trajectory database T

ID #	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	$b2 \rightarrow c3 \rightarrow e5 \rightarrow f6$

data sharing and hence is the main goal of this thesis because it is easily performed by an attacker, and upon success it allows the attacker to learn all other locations and timestamps of the victim. Suppose an adversary knows the data record of a target victim, Alice, in Table 1. The adversary also has prior knowledge that Alice visited locations b and c at timestamps 2 and 9, respectively. Then an adversary can associate $ID\#1$ with Alice because $ID\#1$ is the only record containing both $b2$ and $c9$. Consequently, he can find out that Alice has also visited locations a, c, e, and f at timestamps 1, 3, 5, and 6, respectively. ■

This thesis presents a new heuristic method to anonymize a large volume of passenger-specific trajectory data with local 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 [11]. Therefore, data holders need to transform the underlying raw data into a version that is immune to privacy attacks while maintaining the required quality for the recipient's desired analysis. Figure 1 depicts the information flow from passengers to data recipients.

A related, yet different, research area is *Privacy-Preserving Data Mining* (*PPDM*),

Table 2: $(2, 2)$-privacy preserved database T'

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

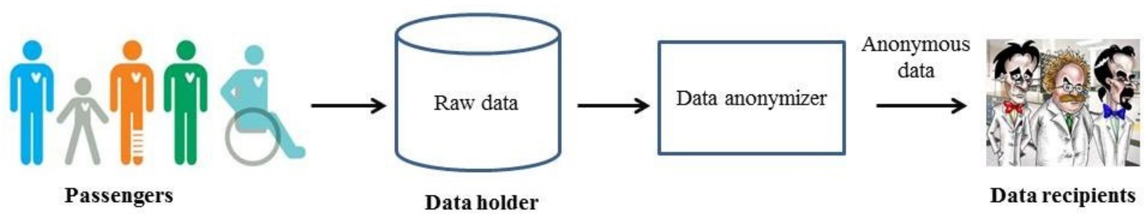

Figure 1: Privacy-preserving data publishing

which aims at releasing privacy-preserving *data mining results*, such as classification models, frequent patterns, or association rules. In the context of passenger flow analysis, releasing data is preferable because data recipients can have greater flexibility in performing their required analysis on the anonymous data. To the best of our knowledge, this is the first work studying trajectory data anonymization for passenger flow analysis.

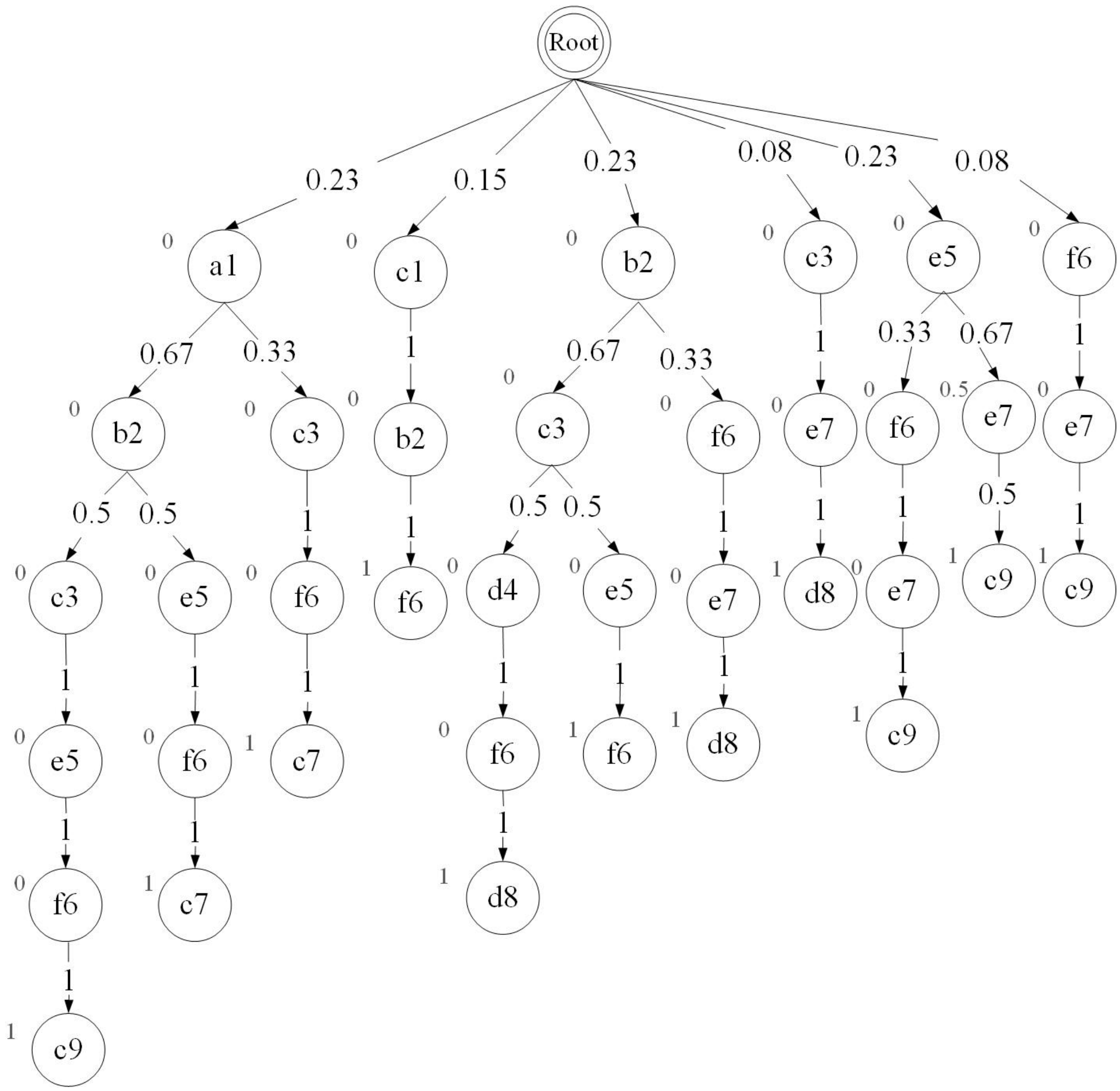

Figure 2: Probabilistic flowgraph of Table 1

1.2 Data privacy and quality trade-off

Several privacy models, such as K-anonymity [48] and its extensions [5,30,35,56,57], 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 [9]. Consider a mass transportation system with 300 metro and bus stations operating 20 hours a day. The corresponding trajectory database would have $300 \times 20 = 6,000$ dimensions. Since K-anonymity requires every trajectory to be shared by at least K

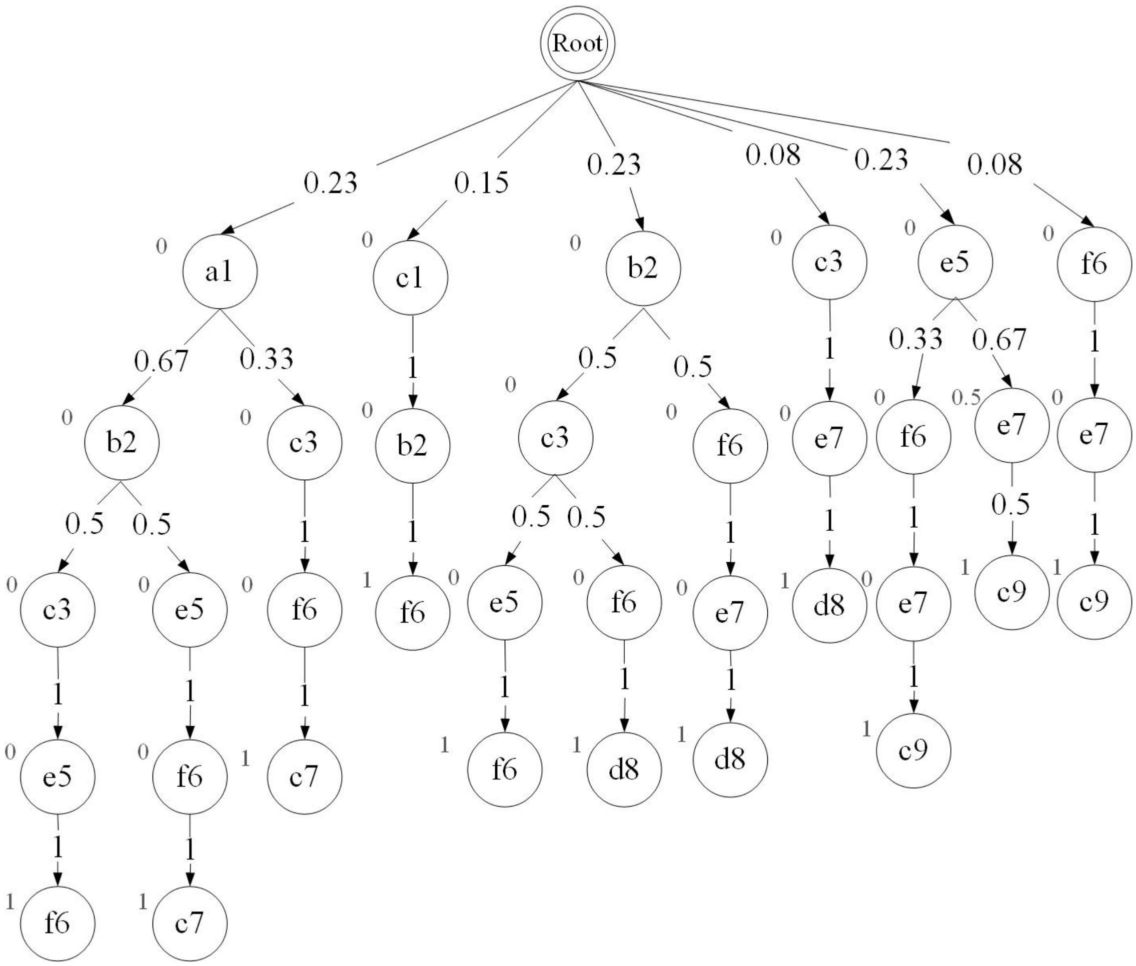

Figure 3: LK-anonymized probabilistic flowgraph of Table 1

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 suppressing most of the data; therefore, the released data is rendered useless for analysis. Furthermore, these privacy models do not consider the sequentiality in the 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 challenge of anonymizing high-dimensional and sparse data, a new privacy model called LK-*privacy* [40] is adopted in this thesis to prevent identity linkage attack. LK-privacy was originally proposed to anonymize high-dimensional relational health data. This new privacy model was built based on the observation that an adversary usually has only limited knowledge about a target victim. Applying the same assumption to trajectory data implies that 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 in order to perform the needed analysis. Anonymous data may be used for different data mining tasks; however, in this thesis 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, Figure 2 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:

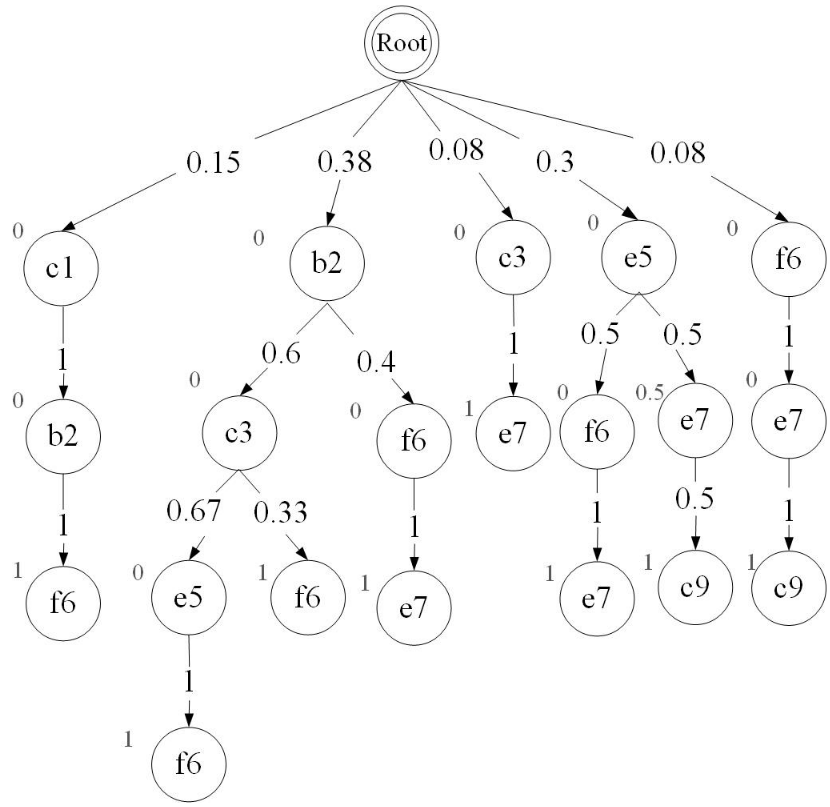

Figure 4: K-anonymized probabilistic flowgraph of Table 1

Example 1.2. Figure 2 depicts the probabilistic flowgraph generated from the raw trajectory data (Table 1). Figure 3 depicts the probabilistic flowgraph generated from Table 2, which satisfies $(2, 2)$-privacy. Figure 4 depicts the probabilistic flowgraph generated from the traditional 2-anonymous data. It is clear that Figure 3 contains more information, including doublet nodes, branches, and transitional probabilities, in the flowgraph than Figure 4. For example, Figure 2 shows that 23% of passengers start their route from $b2$. Figure 3 preserves the same probability, but Figure 4 incorrectly interprets the probability as 38%, resulting in a misleading analysis. This claim is further supported by extensive experimental results in Chapter 5. ■

Generalization, *bucketization*, and *suppression* are the most widely used anonymization mechanisms. In generalization, which can be performed using *global generalization* or *local generalization* [28], specific attributes are replaced by more general attributes. For

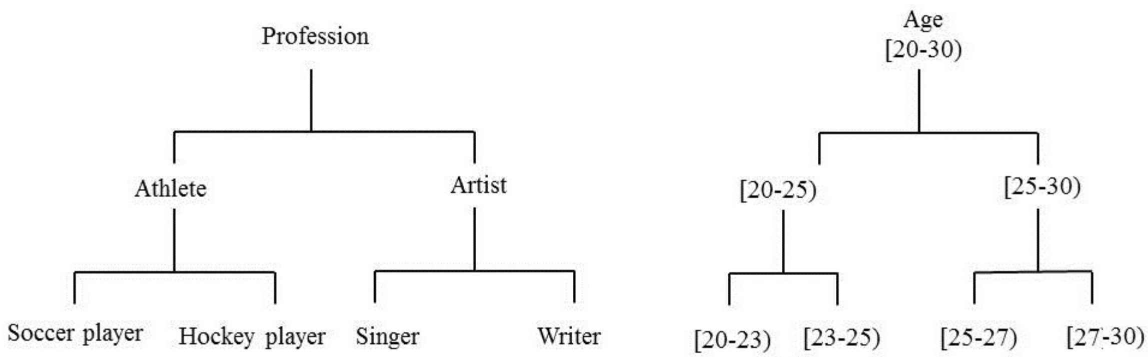

Figure 5: Taxonomy trees for *Profession* and *Age*

example, *Soccer player* and *Hockey player* can be replaced by a more general value *Athlete*. For numerical values, an exact value can be replaced by an interval. Figure 5 depicts the taxonomy trees that generalize specific values to more general ones. Generalization requires the use of taxonomy trees, which are highly specific to a particular application [4]. In many trajectory data applications, such domain specific taxonomy trees are not available. This fact largely hinders generalization's applicability on trajectory data anonymization. Bucketization [36, 58], on the other hand, publishes trajectory data without any modification, but de-associates the relation between *quasi-identifiers (QID)* and sensitive attributes. This mechanism fails to protect identity linkage attacks on trajectory data. In addition, a *condensation* approach [4] is proposed for multi-dimensional data publishing. However, for trajectory data, complexity grows exponentially due to the high dimensionality. Furthermore, it is difficult to measure the similarity of trajectories, which is essential to the condensation approach. Therefore, in this thesis, we employ suppression.

LK-privacy can be achieved by global suppression or local suppression of doublets. A *global suppression* on a doublet d means that *all* instances of d are removed from the data. Global suppression punishes all records containing d by eliminating all instances of d, even

Table 3: Globally suppressing $c9$ from Table 1

ID #	Trajectory
1	$a1 \rightarrow b2 \rightarrow c3 \rightarrow e5 \rightarrow f6$
2	$e5 \rightarrow f6 \rightarrow e7$
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	$f6 \rightarrow e7$
9	$e5 \rightarrow e7$
10	$b2 \rightarrow f6 \rightarrow e7 \rightarrow d8$
11	$a1 \rightarrow c3 \rightarrow f6 \rightarrow e7$
12	$c1 \rightarrow b2 \rightarrow f6$
13	$b2 \rightarrow c3 \rightarrow e5 \rightarrow f6$

if the privacy threat is caused by only one instance of d. Table 3 illustrates globally suppressing doublet $c9$ from Table 1, in which all instances of $c9$ are removed from the table. In contrast, a *local suppression* on a doublet d means that *some* instances of d are removed while some remain intact. Local suppression [12, 41] 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. Suppose that in Table 1, $c9$ from $ID\#1$ causes the privacy violation; applying local suppression on $c9$ in Table 4 results in removing the exact instance of $c9$ from $ID\#1$ rather than removing all instances of $c9$. In this thesis, we employ a hybrid approach of local and global suppression with the goal of maintaining high quality of data for passenger flow analysis with feasible computational complexity.

1.3 Contributions of the thesis

Based on the practical assumption that an adversary has only limited background knowledge of a target victim, we adopt and modify the LK-privacy model for trajectory data anonymization, which prevents identity linkage attacks on trajectory data. This thesis

Table 4: Locally suppressing $c9$ from Table 1

ID #	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	$b2 \rightarrow c3 \rightarrow d4 \rightarrow f6 \rightarrow d8$
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	$b2 \rightarrow c3 \rightarrow e5 \rightarrow f6$

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. All previous privacy works on trajectory data anonymization consider a different information requirement. None of those focus on preserving information quality for generating passenger flowgraphs as discussed in this thesis. Second, we design a hybrid approach that makes use of both global and local suppressions to achieve the requirements of both data privacy and information quality to overcome the challenges of anonymizing high-dimensional and sparse trajectory data. Third, we present a method to measure the similarity between two probabilistic flowgraphs in order to evaluate the difference in 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.

1.4 Thesis organization

The rest of the thesis is organized as follows:

Chapter 2 provides a literature review on traffic and passenger flow analysis and summarizes some common privacy models for relational, statistical, transaction, and trajectory data.

Chapter 3 provides the formal definitions of the input trajectory database, the LK-privacy model, and the passenger flowgraph.

Chapter 4 describes the anonymization algorithm for achieving LK-privacy.

Chapter 5 evaluates the impact of anonymization on the information quality of the flowgraph and efficiency of our proposed methods on synthetic and real-life data.

Finally, Chapter 6 concludes the thesis and outlines possible future research directions.

Chapter 2

Literature Review

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

2.1 Flow analysis

Palleta et al. [45] present a pilot system that helps public transportation system companies optimize the passenger flow 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 flow. Halb et al. [20] 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 et al. [1] 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, and the details of the vehicles in a similar cluster are dispersed through the cluster head. Once this information is obtained from the server, the vehicle with the cluster head status uses the VANET infrastructure to share the required information with its neighborhood.

2.2 Anonymizing relational and statistical data

K-anonymity [47–49], *ℓ-diversity* [35], and *confidence bounding* [56] are common models that prevent privacy attacks against 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. It is based on the concept of generalization by substituting attribute values with generalized values with the objective of minimal distortion while preventing identity linkage.

Machanavajjhala et al. [35] present the *homogeneity attack* and *background knowledge attack* to illustrate that K-anonymity does not provide the claimed privacy guarantee. Consequently, they propose a new privacy model called ℓ-diversity that requires each equivalence class to contain at least ℓ *well-represented* sensitive values. Li et al. [30] present *skewness attack* to illustrate that ℓ-diversity also fails to prevent a privacy attack when the overall distribution of a sensitive attribute is skewed. Instead, they propose a privacy model called *t-closeness* that requires the distribution of a sensitive attribute in any *quasi-identifiers* (*QID*) group to be close to the distribution of the attribute in the overall table. By utilizing the *earth mover's distance (EMD)*, t-closeness measures the closeness between two distributions of sensitive values and requires the closeness to be within t.

Wang et al. [56] present a method to limit the privacy threat by taking into account a set of *privacy templates* specified by a data owner. Such templates formulate individuals' privacy constraints in the form of association rules. Wong et al. [57] propose a new privacy model called (α, K)-*anonymization* by integrating both K-anonymity and confidence bounding into a single privacy model.

However, Kifer [24] illustrates that since an injector framework [31] uses a variation of random worlds or independent and identically distributed random variables for reasoning about privacy, their method is likely to underestimate the risk of disclosure. Kisilevich et al. [25] 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. [37] 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.

Nergiz et al. [44] propose *MultiR* K-anonymity, which achieves K-anonymity on multiple relational tables based on the assumption that a relational database contains a person specific table, PT, and a set of tables $T_1, \ldots, T_n$ where PT contains a person identifier, Pid, and some sensitive attributes; and T_i, for $1 \leq i \leq n$, contains some foreign keys, some attributes in QID, and sensitive attributes. *MultiR* K-anonymity ensures that for each record r in the join of all tables $PT \bowtie T_1 \bowtie \ldots \bowtie T_n$, at least $k-1$ records share the same QID with r.

The above privacy models do not focus on an adversary's background knowledge, but it is reasonable to assume that in real-life privacy attacks an adversary has prior knowledge about the victim. Therefore, more recent works focus on an adversary's background knowledge. Li et al. [31] propose the *injector* framework to model an adversary's background knowledge by mining negative association rules, which is then used in the anonymization process. This is achieved based on a rationale that if certain facts or knowledge exist in a database, the authors should be able to find them using data mining techniques.

Enforcing traditional privacy models on high dimensional relational data usually results in suppressing most of the data [3], thus rendering the released data useless for future analysis. Mohammed et al. [40] 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 real-life privacy attacks, it is less likely that an adversary knows all locations and timestamps of a target victim because a significant amount of effort would be required to gather all prior knowledge from different locations at different times. Thus, it is reasonable to assume that the adversary's background knowledge is bounded by at most L doublets of locations and timestamps that the target has visited. In this thesis, we follow a similar assumption of an adversary's background knowledge and adapt the privacy notion for trajectory data.

Dwork [13] 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 [61]. In particular, for passenger flow analysis, achieving differential privacy may not be able to provide meaningful data utility. Furthermore, Machanava et al. [34] indicate that the resulting data is untruthful due to the uncertainty (e.g., Laplace noise) introduced for achieving differential privacy.

2.3 Anonymizing transaction data

Anonymizing high dimensional transaction data has been widely studied in [10, 18, 21, 51, 53, 54, 59, 60]. In general, this problem setting does not take into account the sequentiality of the data that is an important factor in our problem. Time contains important information for trajectory data mining, specially for passenger flow analysis. Consider two trajectories $a1 \rightarrow c3$ and $c1 \rightarrow a3$. They have the same locations and timestamps but in a different order, and thus they are different from each other. In order to study the passengers' flow, it is necessary to take into consideration the sequentiality of the data. However, this increases the chances for an adversary to exploit such a difference for a successful linkage attack. Therefore, such privacy protection models are not applicable to our problem, and anonymizing trajectory data requires additional efforts.

Ghinita et al. [18] propose a permutation method that groups transactions with close proximity and then associates each group to a set of mixed sensitive values. Sensitive values are then randomized within groups to achieve anonymity. This can work when attributes can *a priori* be partitioned into different quasi-identifiers (QID) and sensitive values. They model the adversary's background knowledge as an arbitrary number of non-sensitive values. Their bucketization-based approach limits the probability of inferring a sensitive value to a specified threshold while it preserves correlations among values for frequent pattern mining. Terrovitis et al. [53] propose an algorithm to K-anonymize transactions by *global generalization* based on some given taxonomy trees in which there are no quasi-identifiers, any item of the sets could be sensitive, and the items of the sets themselves are exploited to tie sets of items to individuals. Depending on the adversary's point of view, they consider both sensitive and nonsensitive data as potential quasi-identifiers and potential sensitive data. Terrovitis et al. [54] improve the quality of data by introducing a local recoding method to achieve anonymity.

He and Naughton [21] argue that the method in [53] does not provide as much privacy protection as K-anonymity, and by introducing *local generalization* they extend their method, which improves data quality. However, generalization does not fit trajectory data well 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. [16] 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. [60] 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. In their proposed method, the set of transactions must be (h, K, p)-coherent in order to achieve anonymization for set-valued data. If not, the item needs to be globally suppressed, which means deleting the item from all transactions that contain it. This privacy criterion ensures that for any p item combination that is nonsensitive, there are at least K transactions in the database containing these items, within which at most h percent of transactions contain some sensitive items. It uses the parameter p to model the adversary's prior knowledge, which offers flexibility in anonymization based on the power of the adversary. (h, K, p)-coherence also has the advantage of incorporating a kind of diversity (of the sort originally introduced in the ℓ-diversity [35]) in the resulting anonymization. Although the above method is improved in [59] by preserving frequent itemsets instead of preserving item instances, and it addresses the high dimensionality concern, the authors 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. [59, 60] achieve their privacy model merely by global suppression, which significantly hinders information quality on trajectory data.

Tassa et al. [51] improve the quality of K-anonymity by introducing new models: $(K, 1)$-, $(1, K)$-, and (K, K)-*anonymity* and K-*concealment*. They argue that $(K, 1)$-, $(1, K)$-, and (K, K)-anonymity do not provide the same level of security as K-anonymity.

K-concealment, on the other hand, provides a 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. Cao et al. [6] propose *ρ-uncertainty*, which bounds the confidence of inferring a sensitive item from both sensitive and non-sensitive items to ρ. They assume that an adversary has some background knowledge of sensitive items. The privacy is achieved by global suppression for both sensitive and non-sensitive items and global generalization for only non-sensitive items.

Chen et al. [10] study the releasing of a 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 that 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 thesis.

2.4 Anonymizing trajectory data

With the increase in use of location-aware devices, more trajectory data has been collected from such devices that provide vast opportunities for researchers to study and analyze the passenger flow. Yet, sharing such information may cause privacy violation of passengers. Some recent works [2,7,8,14,15,22,38,39,43,46,52,62] study anonymization of trajectory data from different perspectives. Based on the assumption that trajectories are imprecise, Abul et al. [2] 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 transit data and RFID data. Trujillo-Rasua et al. [55] 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.

Hoh et al. [22] propose the uncertainly-aware path cloaking algorithm to provide privacy protection for GPS traces. To decrease the identification of trajectories, they selectively remove trajectories with the goal of confusing an attacker. Due to the high dimensionality of trajectory data, Pensa et al. [46] and Terrovitis et al. [52] study privacy protection in *sequential data*, which is a simplified type of trajectory data. Pensa et al. [46] 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 [2], Pensa et al. [46] transform a sequence into another form by inserting, deleting, or substituting some items. First, they build a prefix tree using the raw sequences in the raw database. Then the prefix tree is pruned to ensure that all branches are with a support greater than K. Based on *longest common subsequence (LCS)*, all pruned infrequent sequences are re-appended to the prefix tree. Finally, an anonymous database is built by using the prefix tree.

Based on the assumption that different adversaries have different background knowledge of a victim, Terrovitis et al. [52] propose that the data holder should be aware of *all* such adversarial knowledge. The objective is to prevent an 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. [62] provide privacy protection by utilizing an innovative notion of K-anonymity based on spatial generalization in the context of *moving object databases* (*MOD*). In their proposed algorithm timestamps are considered as the QIDs, and it is assumed that privacy attacks are conducted based on an *attack graph*. They propose two different anonymization algorithms, *extreme union* and *symmetric anonymization*, based on the assumption that different moving objects may have different quasi-identifiers; thus, anonymization groups associated with different objects may not be disjoint. A moving object database satisfies K-anonymity if every node in the attack graph G has at least degree K, and G is symmetric. They identify and generalize the anonymization groups into common regions to the QIDs while minimizing information loss by measuring the reduction in the probability of determining the position of an object over all timestamps between the raw MOD and the anonymous MOD.

Monreale et al. [42] propose a method to ensure K-anonymity by transforming trajectory data based on *spatial generalization*. Hu et al. [23] 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 model called *local enlargement*. They also incorporate an adversary's background knowledge to increase the sustainability of their proposed algorithm against privacy attacks. Nergiz et al. [43] present a generalization-based approach to provide privacy protection for trajectory data by applying K-anonymity, which limits an adversary's ability to re-identify individuals in a trajectory database. They consider an adversary's background knowledge to be a limited part of a trajectory, in which case he may be interested in the rest or in the whole trajectory of an individual, which he may use to infer some sensitive information about the victim. Privacy protection is achieved in two steps. First, the trajectory database is K-anonymized so that every trajectory is indistinguishable from $K - 1$ other trajectories. Second, the data is reconstructed by sampling from anonymized data to prevent further leakage.

Chen et al. [8] 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. [7] 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 recent works [9, 14, 15, 38] study preventing identity linkage attacks over trajectory data but with different information requirements. Fung et al. [14, 15] propose LKC-privacy for anonymizing high-dimensional RFID data, which prevents linkage attacks and overcomes special challenges of high-dimensional RFID data such as high-dimensionality, sparseness, and sequentiality. Global suppression is employed in their method, which leads to less information quality. Similarly, Mohammed et al. [38] study anonymizing high-dimensional trajectory data to overcome linkage attacks while addressing the special challenges of trajectory data anonymization. Chen et al. [9] propose local suppression for anonymizing trajectory data to improve the quality of the data. They present an anonymization framework to preserve both instances of spatio-temporal doublets and frequent sequences in trajectory data.

[14, 15] focus on minimal data distortion and [9, 38] focus on preserving maximal frequent sequences. None of these works focuses on preserving information quality for generating passenger flowgraphs. In contrast, the main goal in this thesis is to preserve

both spatio-temporal data privacy and information quality for passenger flow analysis. By using a sequence of local and global suppression, our proposed algorithm efficiently and effectively addresses the special challenges in trajectory data anonymization for passenger flow analysis.

Chapter 3

Problem Description

The input trajectory database, the LK-privacy model, and the passenger flowgraph are formally defined in this chapter.

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 who entered 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 $(loc_i t_i)$. 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 the form of

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

where ID is the unique identifier of a passenger (e.g., smart card number), $\langle (loc_1 t_1) \rightarrow \ldots \rightarrow (loc_n t_n) \rangle$ is a trajectory, and $y_i \in Y_i$ are relational attributes, such as job, sex, and age. Following 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 [17, 29, 35, 48, 58] have been previously developed for relational QID data; in this thesis we focus on anonymizing the trajectories, 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 [40] 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 $q = \langle (loc_1 t_1) \rightarrow \ldots \rightarrow (loc_q t_q) \rangle$, where $0 < |q| \leq L$. Using this background knowledge, an 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, $T(q)$ is a set of records that contains the record of victim V. 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 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 a certain number of records K.

Definition 3.2 (LK-privacy)**.** Let L be a user-specified threshold indicating the maximum length of the adversary's background knowledge. A trajectory database T satisfies *LK-privacy* if, and only if, for any non-empty sequence q with length $|q| \leq L$ in T, $|T(q)| \geq K$, 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 [17, 32] suggest that anonymization algorithms can be tailored to better preserve information quality if the quality requirement is known in advance. In this thesis, 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 doublet $\{d_x, d_y\}$ 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 Figure 2, 50% of the passengers who have visited $\langle e5 \rightarrow e7 \rangle$ will then visit $c9$. The remaining 50% of passengers terminate 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:

$$Info(d) = \alpha(d) \times w_\alpha + \beta(d) \times w_\beta + \gamma(d) \times w_\gamma \quad (1)$$

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, and w_α, w_β, and w_γ are the weights on the α, β, and γ functions, respectively. The weights, $0 \leq w_\alpha, w_\beta, w_\gamma \leq 1$ and $w_\alpha + w_\beta + w_\gamma = 1$, 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 Figure 2. $\alpha(b2) = 3$ 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 $b2$. Suppose $w_\alpha = 0.5$, $w_\beta = 0.3$, and $w_\gamma = 0.2$. $Info(b2) = 3 \times 0.5 + 5 \times 0.3 + 6 \times 0.2 = 4.2$. ■

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 local minimal impact on the passenger probabilistic flowgraph. ■

Chapter 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 $\alpha(d)$, $\beta(d)$, and $\gamma(d)$ (Equation 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 $0 < |q| \leq L$ 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 $0 < |q| \leq L$. 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 $|q_1| = 2 \leq L$ 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 working 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, Chen et al. [9] 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 [9]. ■

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

Algorithm 1 Identifying minimal violating sequences (MVS)

Require: Raw trajectory database T
Require: Thresholds L, K
Ensure: Minimal violating sequences MVS
1: $C_1 \leftarrow$ all distinct doublets in T;
2: $i \leftarrow 1$;
3: **while** $i \leq L$ and $C_i \neq 0$ **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** $MVS = MVS_1 \cup \cdots \cup MVS_{i-1}$;

$q_2 = \langle c3 \rightarrow d4 \rangle$ is a violating sequence but not a MVS because $d4$ is a violating sequence.

■

Chen et al. [9] prove 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 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, a self-join of the non-violating sequence set, NVS_{i-1}, is conducted (Line 13). Two sequences $q_x = (loc_1^x t_1^x) \rightarrow \ldots \rightarrow (loc_i^x t_i^x)$ and $q_y = (loc_1^y t_1^y) \rightarrow \ldots \rightarrow (loc_i^y t_i^y)$ can be joined if the first $i-1$ doublets are identical and $t_i^x < t_i^y$. The joined sequence is $(loc_1^x t_1^x) \rightarrow \ldots \rightarrow (loc_i^x t_i^x) \rightarrow (loc_i^y t_i^y)$. 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 suppressions for LK-privacy is *NP-hard* [9]. 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's and would have local minimal impact on the passenger flowgraph. Equation 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 Equation 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 $PrivGain(d)$:

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

Most of the previous works on trajectory anonymization [14, 15, 38] 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 [9]. 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 $|T'(a1 \rightarrow b2)| = 1 < K$, where T' 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 a $|MVS|$ 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

Algorithm 2 Check validity of a local suppression

Require: Trajectory database T
Require: Thresholds L, K
Require: A doublet d in a minimal violating sequence m
Ensure: A boolean indicating if locally suppressing d from m is valid
1: $D' \leftarrow \{d' \mid d' \in D, d' \in T(m), d' \in (T(d) - T(m))\}$;
2: $MVS1 \leftarrow \{m1 \mid m1 \in MVS, |m1| = 1\}$
3: $MVS' \leftarrow \{m' \mid m' \in MVS, d \in m, MVS(d)\} \cup MVS1$;
4: Remove all doublets, except for d, in MVS' from D';
5: $Q \leftarrow$ all possible sequences with size $\leq L$ 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;

valid if it does not generate any new MVS [9]. ■

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 summarizes the 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

Algorithm 3 Anonymize trajectory data

Require: Trajectory database T
Require: Thresholds L, K
Ensure: 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 $\neq 0$ **do**
5:     Select a doublet d with the highest score from its MVS m;
6:     **if** d is a local suppression **then**
7:         $MVS' \leftarrow \{m' \mid m' \in MVS, d \in m' \wedge T(m') = T(m)\}$;
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';

$Score$ 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.

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. We generate the flowgraph whose computational time is equal to $\sum_{i=1}^{|T|} |t_i|$, where $|T|$ is the number of records in T and $|t|$ is the number of doublets in each record. Usually, the number of doublets in a records is small and it is reasonable to $|t_i| = |t|$. Hence, the cost is bounded to the size of the database, $|T|$. In the second step, we identify all MVS. Here the most expensive operation is scanning the raw trajectory database T once for all sequences in each candidate set C_i. The cost is $\sum_{i=1}^{L} |C_i|i$, where $|C_i|$ is the size of candidate set C_i. Since C_1 consists of the all size-one sequences, its size would be the number of distinct doublets in T that is the number of dimensions, $|s|$. By self-joining W_1, which consists of all size-one and non-violating sequences from C_1, C_2 is generated; therefore ,the upper bound of C_2 is $|s|(|s| - 1)/2$. However, for $i \geq 3$, the size of the candidate sets does not increase significantly because for all candidates, the two sequences need to share the same prefix in order to perform the self-join and be the future candidate for MVS. Also, the pruning process in Algorithm 1 greatly reduces the candidate search space. Therefore, a good approximation is $C \approx |s|^2$. However, in the worst case, the computational cost of the second step is bounded by $O(|s|^L|T|)$, where $|T|$ is the number of records in T. The third step is the anonymization process, which includes calculating scores for each MVS in table $Score$, and then removing all MVS iteratively. The most costly operation is to check if the instances of the doublets in $MVS(T)$ are valid for local suppression. The number of instances of doublets in $MVS(T)$ is less than $\sum_{i=1}^{L} |C_i|i$, and thus is also bounded by $|s|^L$. For every instance in $MVS(T)$, it is necessary to call Algorithm 2 at most twice, and in the worst case, for each call all records in T need to be scanned. Hence, the cost is still bounded by $O(|s|^L|T|)$. By incorporating all steps, the complexity of the entire algorithm is $O(|s|^L|T|)$. In addition to the theoretical analysis above, the scalability of our algorithm is further experimentally validated in Chapter 5.2.

Chapter 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 measure $\varphi(G, G')$ 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 $\alpha(d)$, $\beta(d)$, $\gamma(d)$, $\alpha(d')$, $\beta(d')$, and $\gamma(d')$; computes the ratios among them; and then sums up the ratios, denoted by $aSum$, $bSum$, and $cSum$ (Lines 5-16), 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 18. Line 19 returns the similarity measure φ, which is a weighted sum of the ratios.

We could not directly compare our proposed algorithm with previous works [2, 9, 46, 52, 62] on trajectory data anonymization because their proposed solutions do not consider

Algorithm 4 Comparing two flowgraphs

Require: Flowgraph G
Require: Flowgraph G'
Require: Weights $w_\alpha, w_\beta, w_\gamma$
Ensure: Similarity measure φ

$UL \leftarrow \{d \mid d \in G\}$;
$UL' \leftarrow \{d' \mid d' \in G'\}$;
Sort UL and UL' by time and location;
$i \leftarrow 0$;
for each $d \in UL$ **do**
    for each $d' \in UL'$ **do**
        if $d = d'$ **then**
            $aSum \mathrel{+}= \frac{\alpha(d')}{\alpha(d)}$;
            if $\beta(d) \neq 0$ **then**
                $bSum \mathrel{+}= \frac{\beta(d')}{\beta(d)}$;
            else
                $i{+}{+}$;
            end if
            $cSum + = \frac{\gamma(d')}{\gamma(d)}$;
        end if
    end for
end for
$\varphi \leftarrow \frac{aSum}{|G|} \times w_\alpha + \frac{bSum}{|G|-i} \times w_\beta + \frac{cSum}{|G|} \times w_\gamma$;
return φ;

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 hours, 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 hours, where the time granularity is set to the hour level. The properties of the two experimental data sets are summarized in Table 5.

[1] www.stm.info

Table 5: Experimental data set statistics

Data sets	Records $|T|$	Dimensions $|s|$	Data size (Kbytes)	Data type
Metro200K	200,000	696	12,359	Synthetic
STM514K	514,213	3120	12,910	Real-life

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. High-dimensional trajectory data are usually sparse. Consider passengers in transit systems. Among all available locations and all possible timestamps, they may visit only a few locations at a few timestamps, making the trajectory of each individual relatively short. In real-life trajectory data the average length of a trajectory equals to 4 sequences. Therefore, it is reasonable to set $L = 3$. Setting L to a greater value means that the adversary has almost complete knowledge about his victim, which in turn means there is no need for further identity linkage attack.

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 two scenarios with different weights.

5.1.1 Scenario 1

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 $w_\alpha = 0.5$, $w_\beta = 0.3$, and $w_\gamma = 0.2$.

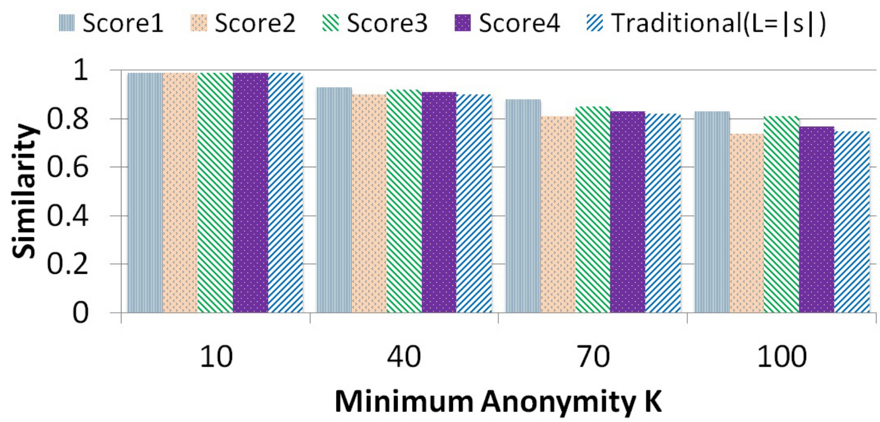

Fig a: Metro200K

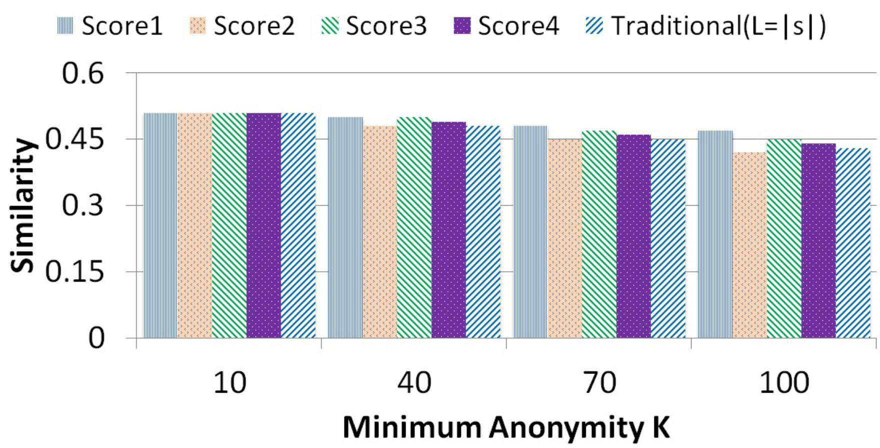

Fig b: STM514K

Figure 6: Similarity vs. $K(L = 3, w_\alpha = 0.5, w_\beta = 0.3, w_\gamma = 0.2)$

Figure 6.a depicts the similarity measure φ of the two flowgraphs before and after the anonymization for $L = 3$ and $10 \leq K \leq 100$, with different $Score$ functions on the *Metro200K* data set. When $K = 10$, the similarity is 0.99, indicating almost no information has been lost in terms of the flowgraph. As K increases, the similarity decreases. This shows a 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

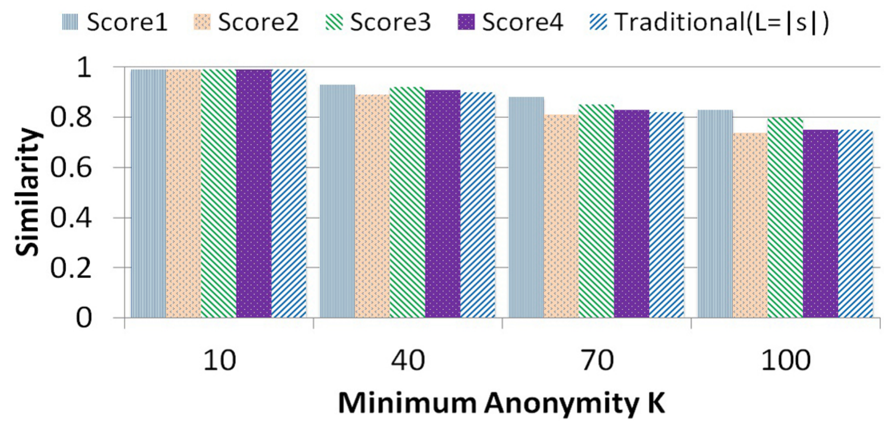

Fig a: Metro200K

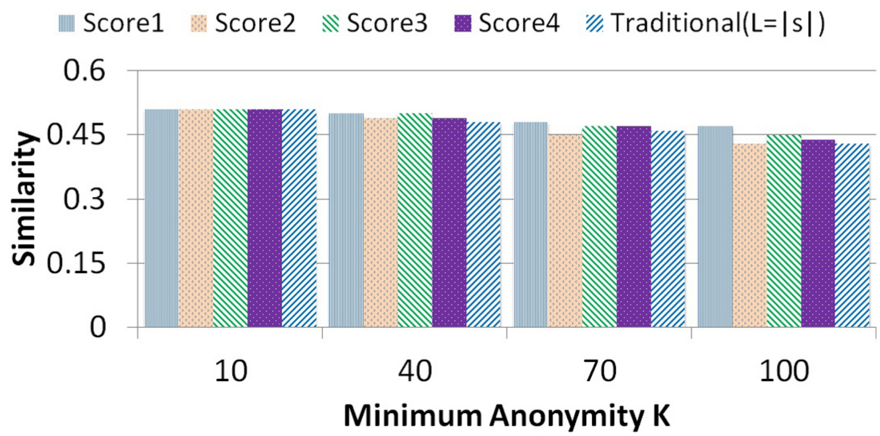

Fig b: STM514K

Figure 7: Similarity vs. $K(L = 3, w_\alpha = 0.3, w_\beta = 0.5, w_\gamma = 0.2)$

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 $10 \leq K \leq 100$. The p-values for $Score1$, $Score2$, $Score3$, and $Score4$ in Figure 6.a are 1.75E-3, 1.28E-2, 5.67E-4, and 1.58E-3, respectively. Figure 6.b depicts the similarity measure φ of the flowgraphs before and after the anonymization for $L = 3$ and $10 \leq K \leq 100$ with different $Score$ functions on the *STM514K* data set. Similar trends can be observed. The p-values for $Score1$, $Score2$, $Score3$, and $Score4$ in Figure 6.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 2

In this scenario, the weights are set at $w_\alpha = 0.3$, $w_\beta = 0.5$, and $w_\gamma = 0.2$, with $L = 3$ and $10 \leq K \leq 100$. The results in Figures 7.a and 7.b in this scenario indicate that our proposed algorithm still performs best, suggesting that our method is robust against different weights and different scenarios of flowgraph analysis. The behaviour 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 α in Scenario 1 is much higher than the weight α in Scenario 2.

The results further confirm that our score functions in general produce better information quality than K-anonymity, except for $Score2$, 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 $10 \leq K \leq 100$. The p-values for $Score1$, $Score2$, $Score3$, and $Score4$ in Figure 7.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 Figure 7.b are 3.98E-3, 5.0E-2, 4.5E-3, and 2.88E-3, respectively, showing that the benefit is statistically significant at $\alpha = 5\%$.

5.2 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 $|T|$

Figures 8.a and 9.a illustrate the runtime of our algorithm on a data set with 4,000 dimensions and sizes ranging from 400,000 records to 1,200,000 records. In Figure 8.a we

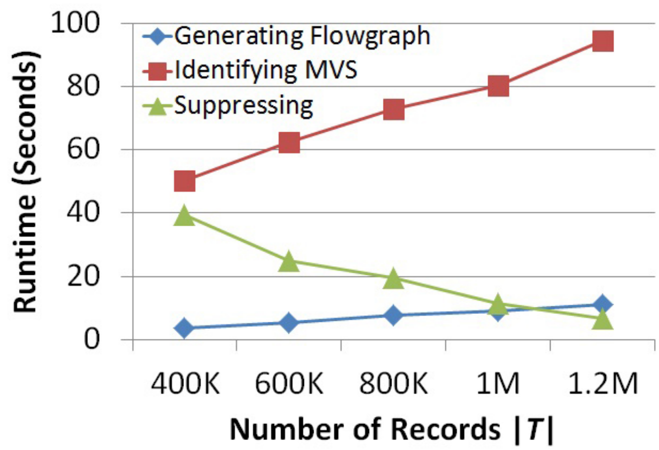

Fig. a: Runtime vs. records

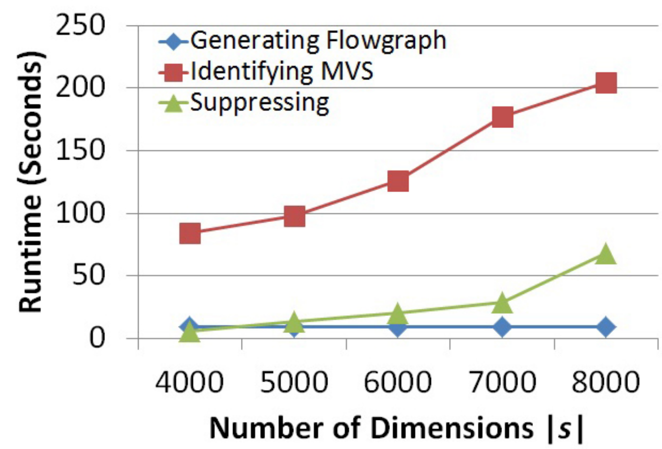

Fig. b: Runtime vs. dimensions

Figure 8: Scalability ($L = 3$, $K = 30$)

observe that the runtime for generating the flowgraph is linear and proportional to the number of records. The algorithm takes less than 15 seconds 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.

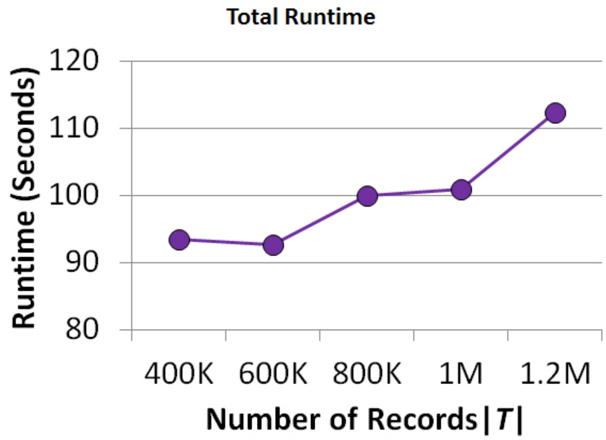

Fig. a: Runtime vs. records

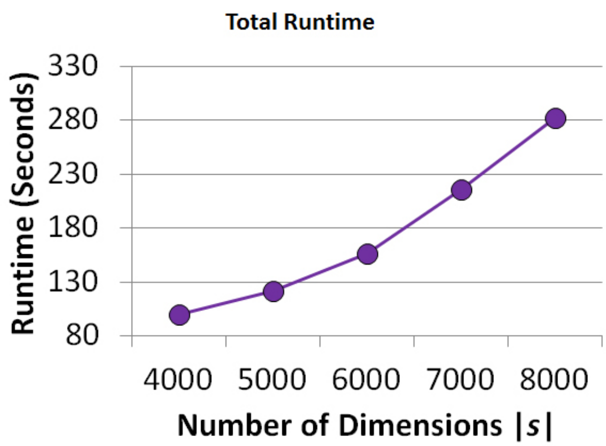

Fig. b: Runtime vs. dimensions

Figure 9: Scalability ($L = 3$, $K = 30$)

5.2.2 Effect of dimensionality $|s|$

Figures 8.b and 9.b depict the runtime of our algorithm on a data set of 1 million records, with the number of dimensions (number of distinct doublets) ranging from 4,000 to 8,000. Figure 8.b 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.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

With the advancement of the use of technology in transportation companies there is a strong tendency toward sharing information for analysis purposes. Consequently, sharing high-dimensional passenger-specific trajectory data raises new privacy concerns that cannot be appropriately addressed by traditional privacy protection techniques.

In this thesis, we study the problem of anonymizing high-dimensional trajectory data for passenger flow analysis. We demonstrate that applying traditional K-anonymity to 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 on uncertain data. 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.

6.2 Future work

By deploying various location-aware devices in transportation systems, such as contactless smart cards or RFID cards and GPS receivers, massive volume of spatio-temporal trajectory data is generated daily. Such data can be used to study and monitor the traffic flow in the anonymized trajectory data. In this thesis, the focus is specifically on preserving information quality for passenger flow analysis. By utilizing a probabilistic flowgraph, the proposed method in this thesis can be applied to study the anonymization of trajectory data for traffic flow analysis which studies the interaction between vehicles, drivers, and even infrastructures such as highways and traffic control devices. Hence, our future work will focus on preserving the privacy of high-dimensional data for traffic flow analysis.

We will study the anomyziation of high-dimensional trajectory data for traffic flow analysis in a transportation system by considering two aspects. First, it is more efficient and beneficial for a transportation company to designate routes with the shortest travel time for their passengers. Second, the company requires to take into consideration the availability of transportation utilities such as bus stops. Considering both the shortest travel time and the availability benefit both the transportation company and the passengers. By reducing the travel time the passengers would reach their destination sooner, while it would reduce the costs for transportation company, as well. On the other hand, routes with shorter travel time should not decrease the availability of transportation' utilities for passengers. For example, if the company places its bus stations in a way which leads to a short travel time, but the stations are far away from passengers access, it is more likely that few passengers would use that particular bus line. Therefore, it is required to preserve the paths in the network which have the shortest travel time and provide the most availability to the passengers. Consequently, in our future work the proposed probabilistic flowgraph can be incorporated in an anonymization framework in a transportation system to provide both better monitoring and optimization of the designated bus routes and privacy preservation.

Bibliography

[1] Sajimon Abraham and Paulose S. Lal. Spatio-temporal similarity of network-constrained moving object trajectories using sequence alignment of travel locations. *Transportation Research Part C: Emerging Technologies*, 23:109–123, 2012.

[2] Osman Abul, Francesco Bonchi, and Mirco Nanni. Never walk alone: Uncertainty for anonymity in moving objects databases. In *Proceedings of the 24th IEEE International Conference on Data Engineering*, pages 376–385, 2008.

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

[4] Charu C. Aggarwal, Charu C. Aggarwal, Philip S. Yu, and Philip S. Yu. A condensation approach to privacy preserving data mining. In *Proceedings of International Conference on Extending Database Technology*, pages 183–199, 2004.

[5] Jianneng Cao and Panagiotis Karras. Publishing microdata with a robust privacy guarantee. *Proceedings of the VLDB Endowment*, 5(11):1388–1399, 2012.

[6] Jianneng Cao, Panagiotis Karras, Chedy Raïssi, and Kian-Lee Tan. ρ-uncertainty: inference-proof transaction anonymization. *Proceedings of the VLDB Endowment*, 3(1-2):1033–1044, 2010.

[7] Rui Chen, Gergely Acs, and Claude Castelluccia. Differentially private sequential data publication via variable-length n-grams. In *Proceedings of the ACM conference on Computer and Communications Security*, pages 638–649, 2012.

[8] Rui Chen, Benjamin C. M. Fung, Bipin C. Desai, and Nériah M. Sossou. 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*, pages 213–221, 2012.

[9] Rui Chen, Benjamin C. M. Fung, Noman Mohammed, Bipin C. Desai, and Ke Wang. Privacy-preserving trajectory data publishing by local suppression. *Information Sciences*, 231:83–97, 2013.

[10] Rui Chen, Noman Mohammed, Benjamin C. M. Fung, Bipin C. Desai, and Li Xiong. Publishing set-valued data via differential privacy. *Proceedings of the VLDB Endowment*, 4(11):1087–1098, 2011.

[11] Chris Clifton and Tamir Tassa. On syntactic anonymity and differential privacy. In *Proceeding of the 29th IEEE International Conference on Data Engineering Workshops (ICDEW)*, pages 88–93, 2013.

[12] Lawrence H. Cox. Suppression methodology and statistical disclosure control. *Journal of the American Statistical Association*, 75(370):377–385, 1980.

[13] Cynthia Dwork. Differential privacy. In *Proceedings of the 33rd International Colloquium on Automata, Languages and Programming (ICALP)*, pages 1–12. 2006.

[14] Benjamin C. M. Fung, Khalil Al-Hussaeni, and Ming Cao. Preserving RFID data privacy. In *Proceedings of the 2009 IEEE International Conference on RFID*, pages 200–207, 2009.

[15] Benjamin C. M. Fung, Ming Cao, Bipin C. Desai, and Heng Xu. Privacy protection for RFID data. In *Proceedings of the 2009 ACM symposium on Applied Computing*, pages 1528–1535, 2009.

[16] Benjamin C. M. Fung, Ke Wang, Rui Chen, and Philip S. Yu. Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys (CSUR)*, 42(4):14, 2010.

[17] Benjamin C. M. Fung, Ke Wang, and Philip S. Yu. Anonymizing classification data for privacy preservation. *IEEE Transactions on Knowledge and Data Engineering*, 19(5):711–725, 2007.

[18] Gabriel Ghinita, Yufei Tao, and Panos Kalnis. On the anonymization of sparse high-dimensional data. In *Proceedings of the 24th IEEE International Conference on Data Engineering (ICDE)*, pages 715–724, 2008.

[19] Fosca Giannotti, Mirco Nanni, Fabio Pinelli, and Dino Pedreschi. Trajectory pattern mining. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 330–339, 2007.

[20] Wolfgang Halb and Helmut Neuschmied. Multimodal semantic analysis of public transport movements. In *Proceedings of 4th International Conference on Semantic and Digital Media Technologies (SAMT)*, pages 165–168, 2009.

[21] Yeye He and Jeffrey F. Naughton. Anonymization of set-valued data via top-down, local generalization. *Proceedings of the VLDB Endowment*, 2(1):934–945, 2009.

[22] Baik Hoh, Marco Gruteser, Hui Xiong, and Ansaf Alrabady. Preserving privacy in GPS traces via uncertainty-aware path cloaking. In *Proceedings of the 14th ACM Conference on Computer and Communications Security*, pages 161–171, 2007.

[23] Haibo Hu, Jianliang Xu, Sai Tung On, Jing Du, and Joseph K. NG. Privacy-aware location data publishing. *ACM Transactions on Database Systems (TODS)*, 35(3):18, 2010.

[24] Daniel Kifer. Attacks on privacy and definetti's theorem. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data*, pages 127–138, 2009.

[25] Slava Kisilevich, Lior Rokach, Yuval Elovici, and Bracha Shapira. Efficient multidimensional suppression for k-anonymity. *IEEE Transactions on Knowledge and Data Engineering*, 22(3):334–347, 2010.

[26] Jae Gil Lee, Jiawei Han, Xiaolei Li, and Hector Gonzalez. Traclass: trajectory classification using hierarchical region-based and trajectory-based clustering. *Proceedings of the VLDB Endowment*, 1(1):1081–1094, 2008.

[27] Jae Gil Lee, Jiawei Han, and Kyu-Young Whang. Trajectory clustering: a partition-and-group framework. In *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data*, pages 593–604, 2007.

[28] Kristen LeFevre, David J. DeWitt, and Raghu Ramakrishnan. Incognito: Efficient full-domain k-anonymity. In *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*, pages 49–60, 2005.

[29] Kristen LeFevre, David J. DeWitt, and Raghu Ramakrishnan. Mondrian multidimensional k-anonymity. In *Proceedings of the 22nd International Conference on Data Engineering*, pages 25–25, 2006.

[30] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and ℓ-diversity. In *Proceedings of the 23rd IEEE International Conference on Data Engineering*, pages 106–115, 2007.

[31] Tiancheng Li and Ninghui Li. Injector: Mining background knowledge for data anonymization. In *Proceedings of the 24th IEEE International Conference on Data Engineering*, pages 446–455, 2008.

[32] Tiancheng Li and Ninghui Li. 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*, pages 517–526, 2009.

[33] Xiaolei Li, Jiawei Han, Jae-Gil Lee, and Hector Gonzalez. Traffic density-based discovery of hot routes in road networks. In *Proceedings of 10th International Symposium on Advances in Spatial and Temporal Databases*, pages 441–459, 2007.

[34] Ashwin Machanavajjhala, Johannes Gehrke, and Michaela Götz. Data publishing against realistic adversaries. *Proceedings of the VLDB Endowment*, 2(1):790–801, 2009.

[35] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. ℓ-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):3, 2007.

[36] David J. Martin, Daniel Kifer, Ashwin Machanavajjhala, Johannes Gehrke, and Joseph Y. Halpern. Worst-case background knowledge for privacy-preserving data publishing. In *Proceeding of the 23rd IEEE International Conference on Data Engineering (ICDE)*, pages 126–135, 2007.

[37] Nissim Matatov, Lior Rokach, and Oded Maimon. Privacy-preserving data mining: A feature set partitioning approach. *Information Sciences*, 180(14):2696–2720, 2010.

[38] Noman Mohammed, Benjamin C. M. Fung, and Mourad Debbabi. Walking in the crowd: anonymizing trajectory data for pattern analysis. In *Proceedings of the 18th*

ACM Conference on Information and Knowledge Management, pages 1441–1444, 2009.

[39] Noman Mohammed, Benjamin C. M. Fung, and Mourad Debbabi. Preserving privacy and utility in RFID data publishing. *Technical Report 6850, Concordia University*, 2010.

[40] Noman Mohammed, Benjamin C. M. Fung, Patrick C. K. Hung, and Cheuk-Kwong Lee. Centralized and distributed anonymization for high-dimensional healthcare data. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 4(4):18:1–18:33, 2010.

[41] David Molnar and David Wagner. Privacy and security in library RFID: issues, practices, and architectures. In *Proceedings of the 11th ACM Conference on Computer and Communications Security*, pages 210–219, 2004.

[42] Anna Monreale, Gennady Andrienko, Natalia Andrienko, Fosca Giannotti, Dino Pedreschi, Salvatore Rinzivillo, and Stefan Wrobel. Movement data anonymity through generalization. *Transactions on Data Privacy*, 3(2):91–121, 2010.

[43] Mehmet Ercan Nergiz, Maurizio Atzori, and Yucel Saygin. Towards trajectory anonymization: a generalization-based approach. In *Proceedings of the SIGSPATIAL ACM GIS 2008 International Workshop on Security and Privacy in GIS and LBS*, pages 52–61, 2008.

[44] Mehmet Ercan Nergiz, Chris Clifton, and Ahmet Erhan Nergiz. Multirelational k-anonymity. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 21(8):1104–1117, 2009.

[45] Lucas Paletta, Siegfried Wiesenhofer, Norbert Brandle, Oliver Sidla, and Yuriy Lypetskyy. Visual surveillance system for monitoring of passenger flows at public transportation junctions. In *Proceedings of the 2005 IEEE Intelligent Transportation Systems*, pages 862–867, 2005.

[46] Ruggero G. Pensa, Anna Monreale, Fabio Pinelli, and Dino Pedreschi. Pattern-preserving k-anonymization of sequences and its application to mobility data mining. *Privacy in Location-Based Applications*, page 44, 2008.

[47] Pierangela Samarati. Protecting respondents' identities in microdata release. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 13(6):1010–1027, 2001.

[48] Pierangela Samarati and Latanya Sweeney. Generalizing data to provide anonymity when disclosing information. In *Proceedings of the 17th ACM SIGACT-SIGMOD SIGART Symposium on Principles of Database Systems (PODS)*, volume 17, pages 188–188, 1998.

[49] Latanya Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.

[50] Lu-An Tang, Yu Zheng, Jing Yuan, Jiawei Han, Alice Leung, Wen-Chih Peng, Thomas La Porta, and Lance Kaplan. A framework of traveling companion discovery on trajectory data streams. *ACM Transaction on Intelligent Systems and Technology*, 2012.

[51] Tamir Tassa, Arnon Mazza, and Aristides Gionis. k-concealment: An alternative model of k-type anonymity. *Transactions on Data Privacy*, 5(1):189–222, 2012.

[52] Manolis Terrovitis and Nikos Mamoulis. Privacy preservation in the publication of trajectories. In *Proceedings of the 9th International Conference on Mobile Data Management (MDM)*, pages 65–72, 2008.

[53] Manolis Terrovitis, Nikos Mamoulis, and Panos Kalnis. Privacy-preserving anonymization of set-valued data. *Proceedings of the VLDB Endowment*, 1(1):115–125, 2008.

[54] Manolis Terrovitis, Nikos Mamoulis, and Panos Kalnis. Local and global recoding methods for anonymizing set-valued data. *Very Large Data Bases Journal (VLDBJ)*, 20(1):83–106, 2011.

[55] Rolando Trujillo-Rasua and Josep Domingo-Ferrer. On the privacy offered by (k, δ)-anonymity. *Information Systems*, 38(4):491–494, 2013.

[56] Ke Wang, Benjamin C. M. Fung, and Philip S. Yu. Handicapping attacker's confidence: an alternative to k-anonymization. *Knowledge and Information Systems*, 11(3):345–368, 2007.

[57] Raymond Wong, Jiuyong Li, Ada Fu, and Ke Wang. (α, k)-anonymous data publishing. *Journal of Intelligent Information Systems*, 33(2):209–234, 2009.

[58] Xiaokui Xiao and Yufei Tao. Personalized privacy preservation. In *Proceedings of the 32nd ACM SIGMOD International Conference on Management of Data (SIGMOD)*, pages 229–240, 2006.

[59] Yabo Xu, Benjamin C. M. Fung, Ke Wang, Ada Wai-Chee Fu, and Jian Pei. Publishing sensitive transactions for itemset utility. In *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM)*, pages 1109–1114, 2008.

[60] Yabo Xu, Ke Wang, Ada Wai-Chee Fu, and Philip S. Yu. Anonymizing transaction databases for publication. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 767–775, 2008.

[61] Yin Yang, Zhenjie Zhang, Gerome Miklau, Marianne Winslett, and Xiaokui Xiao. Differential privacy in data publication and analysis. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, pages 601–606, 2012.

[62] Roman Yarovoy, Francesco Bonchi, Laks V.S. Lakshmanan, and Wendy Hui Wang. Anonymizing moving objects: how to hide a MOB in a crowd? In *Proceedings of the 12th International Conference on Extending Database Technology*, pages 72–83, 2009.

[63] Kai Zheng, Yu Zheng, Nicholas Jing Yuan, and Shuo Shang. On discovery of gathering patterns from trajectories. In *Proceeding of 2013 IEEE International Conference on Data Engineering (ICDE)*, 2013.