IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 10, 2021, accepted January 18, 2021, date of publication January 29, 2021, date of current version February 10, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3055505

Utility-Driven *k*-Anonymization of Public Transport User Data

BHAWANI SHANKER BHATI[1], JORDAN IVANCHEV[1], IVA BOJIC[2], ANWITAMAN DATTA[3], AND DAVID ECKHOFF[1], (Member, IEEE)

[1]TUMCREATE Ltd., Singapore 138602
[2]Senseable City Laboratory, Singapore-MIT Alliance for Research and Technology, Singapore 138602
[3]School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798

Corresponding author: Bhawani Shanker Bhati (bhawani.bhati@tum-create.edu.sg)

This work was supported by the Singapore National Research Foundation through the Campus for Research Excellence and Technological Enterprise (CREATE) Programme.

ABSTRACT In this article, we propose a *k*-anonymity approach that prioritizes the generalization of attributes based on their utility. We focus on transport data, which we consider a special case in which many or all attributes are quasi-identifiers (e.g., origin, destination, ride start time), as they allow correlation with easily observable auxiliary data. The novelty in our approach lies in introducing normalization techniques as well as distance and utility metrics that allow the consideration of not only numerical attributes but also categorical attributes by representing them in tree or graph form. The prioritization of the attributes in the generalization process is based on the attributes' utility and can further be influenced by either automatically or manually assigned attribute weights. We evaluate and compare different options for all components of our mechanism as well as present an extensive performance evaluation of our approach using real-world data. Lastly, we show in which cases suppression of records can counter-intuitively lead to higher data utility.

INDEX TERMS Clustering, k-anonymity, privacy, tap-in tap-out transportation data, utility.

I. INTRODUCTION

Publishing or exchanging datasets is often (rightfully) hindered by privacy protection requirements and by the concern of disclosing too much information. Especially with regards to microdata, that is, data that includes user-specific information, such as the travel routes of single users, the privacy implications are too severe to allow disclosure. This in turn leads to the fact that Open Data initiatives often suffer from data sparsity and that research, particularly in the smart mobility domain, cannot be enriched, evaluated, and validated using real datasets [1]. This often reduces mobility research to an academic exercise, limiting its impact and contribution.

There exists an ample body of related research in the field of privacy-preserving data publishing for microdata [2]. They all share the same problem statement, that is, how can potentially privacy-critical data be sanitized so that the publication of this data has no negative effects on the privacy of the included users. While there are different approaches, e.g., applying noise to statistical queries [3], the conventional methods are more or less based on the *k*-anonymity approach [4]. The idea is to remove all identifiers and then apply generalization (reducing accuracy of an attribute) and suppression (removing an entire row from the output dataset) to create clusters of indistinguishable rows in the dataset with respect to a list of attributes that could allow the identification of an individual, the quasi-identifiers.

Advancements of *k*-anonymity include *l*-diversity [5], *t*-closeness [6], and *m*-invariance [7] which consider the distribution of the sensitive attributes. What these approaches have in common is that their mechanisms focus on the privacy aspects but do not take into consideration impact on or even control of data utility, or, usefulness. If data owners were able to better control (and measure) the utility level of the data they share while at the same time meeting privacy requirements, they might have fewer reservations when it comes to data publishing and sharing.

In this article, we revisit the non-interactive *k*-anonymity approach for mobility data from a utility perspective. Our contributions can be summarized as follows:

- We present a *k*-anonymity mechanism for public transport data, where the separation of sensitive data and

The associate editor coordinating the review of this manuscript and approving it for publication was Gautam Srivastava.

quasi-identifiers is difficult, as it was shown to be possible to derive user identities from origin and destination pairs, travel times, and so on [8], [9].

- To achieve a general approach to control and measure utility, we introduce distance and distortion metrics for heterogeneous attribute types, i.e., numerical, categorical, and graphs.
- We introduce two modes for our entropy-based utility-driven k-anonymity mechanism, an automated one to balance utility loss among all attributes as well as a manual mechanism which allows users to define which attributes to preserve more.
- We introduce a record suppression scheme that imposes constraints for record generalization. We show how record suppression can counter-intuitively benefit data utility when the inclusion of a data row would require too much generalization of other entries.
- We compare different options for all of these components and extensively evaluate the performance of our approach using real-world data.

The remainder of the paper is organized as follows: First we introduce the general concept of k-anonymity and the format of our dataset in Section II. Section III is structured according to the building blocks of our mechanism: First we discuss how to obtain utility values for each attribute. We then show how these utility values can be incorporated into the distance between attributes of various types as well as entire records, with a particular focus on the normalization of those distances. We describe how the distortion of the dataset can be measured before we present our k-anonymity mechanism in detail. In Section IV we present an extensive evaluation of the various building blocks of our system. Related work is discussed in Section V, and Section VI concludes the paper.

II. PRELIMANARIES

In this article, we focus on the utility-driven k-anonymization of public transport data. In k-anonymity and related privacy mechanisms, the columns (or attributes) of a dataset are divided into explicit identifiers, quasi-identifiers (columns that could potentially be used to identify an individual), and sensitive attributes. For example, in a healthcare dataset, the `name` is an explicit-identifier, `age` and `gender` are quasi-identifiers, and `disease` is a sensitive attribute. Naturally, the explicit identifiers are removed and the mechanism then generalizes the quasi-identifiers to an extent that they form so-called equivalence classes of k indistinguishable entries with respect to their quasi-identifiers. In transportation data, however, the separation of sensitive attributes and quasi-identifiers is not so straight-forward. If the origin and destination points of an individual's journey were sensitive attributes then they would not be generalized, potentially allowing the identification of an individual based on easily obtainable auxiliary data such as work and home addresses [8], [9]. Knowing that a certain individual is included in the dataset alongside some information about a trip this individual took could lead to de-anonymization.

TABLE 1. Subset of attributes from transportation dataset and their description. card_id (an explicit identifier) is removed from the dataset before processing.

Attribute	Description	Attribute Type
card_id	Card identification number	Not Included
passenger_type	Passenger type	Categorical
boarding_stop_stn	Boarding station	Categorical
alighting_stop_stn	Alighting station	Categorical
ride_start_time_seconds	Ride start time (seconds)	Numerical
ride_time_seconds	Ride time (seconds)	Numerical
ride_distance	Ride distance (KMs)	Numerical

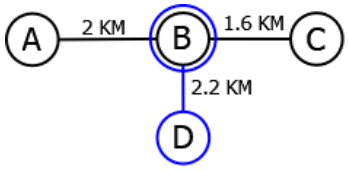


FIGURE 1. An example of public transport showing 4 stations and 2 different lines (blue and black).

Unlike in health or voter data, in mobility data, auxiliary data can be quite easily obtained by simple observation, making the k-anonymity approach quite prone to de-anonymization attacks. To alleviate this problem, we assume that *all* attributes in our examples are quasi-identifiers, meaning each equivalence class would then contain k identical entries. While this assumption improves privacy protection by eliminating some of the common weaknesses of k-anonymity, it will negatively affect utility as generalization and suppression will now incorporate all attributes. To counter this general loss of utility, our mechanism therefore allows users to prioritize the preservation of one attribute over another.

For the sake of simplicity and without loss of generality, we illustrate all concepts with the help of the actual dataset we use for the evaluation. Each record in this dataset consists of a card id, the passenger type, boarding station, alighting station, ride start time, ride time, and ride distance as shown in Table 1.

In what follows, we use this dataset to introduce some definitions required for the remainder of this article.

Original Table An original table or dataset (T) is a set of records $r_1, r_2, \ldots, r_n$ each comprising a sequence of m attribute values $a_1, a_2, \ldots, a_m$.

Example 1 (Original Table): For Figure 1, we consider that Table 2 is an original table containing 8 travel records (r_1, r_2, ..., r_8) with each record having following 6 attributes: Passenger Type (i.e., passenger_type), Boarding Station (i.e., boarding_stop_stn), Alighting Station (i.e., alighting_stop_stn), Ride Start Time (i.e., ride_start_time_seconds), Ride Time (i.e., ride_time_seconds), and Ride Distance (i.e., ride_distance). Note that we use attribute names such as Passenger Type, passenger_type, and passenger type interchangeably in this article.

TABLE 2. An example transportation dataset (original).

Record	Passenger Type	Boarding Station	Alighting Station	Ride Start Time	Ride Time	Ride Distance
r_1	Adult	A	C	9:00	400	3.6
r_2	Adult	A	C	9:00	400	3.6
r_3	C/S	B	D	9:05	320	2.2
r_4	Adult	B	A	10:00	280	2
r_5	SeC	A	D	11:00	520	4.2
r_6	C/S	C	D	9:15	340	3.8
r_7	Adult	B	D	10:00	480	2.2
r_8	SeC	A	D	11:05	520	4.2

C/S - Child/Student; SeC - Senior Citizen

Attribute Type An attribute (or column) within the dataset can be of the type - categorical/qualitative (nominal, ordinal, binary) or numerical/quantitative (discrete, continuous), where:

- Nominal attributes have no order (e.g., race, etc.)
- Ordinal attributes have an order (e.g., performance classification)
- Binary attributes take only two values (e.g., test results pass/fail)
- Discrete attributes take values based on counts (e.g., passenger counts)
- Continuous attributes take any value within a range (e.g., the height and weight of a person)

Example 2 (Attribute Type): In Table 1, the *Attribute Type* column specifies the type of an attribute, with Passenger Type, Boarding Station and Alighting Station as categorical types, and Ride Start Time, Ride Time and Ride Distance as numerical types.

Equivalence Class An equivalence class for table T with respect to attribute set $\{a_1, a_2, \ldots, a_m\}$ is the set of all records in T with identical values (or ranges) for this attribute set.

Example 3 (Equivalence Class): Records r_1 and r_2 (in Table 2) form an equivalence class of size 2 with respect to attribute set {Passenger Type, Boarding Station, Alighting Station, Ride Start Time, Ride Time, and Distance}.

***k*-anonymized Table** A table T_{ano} is said to be k-anonymized with respect to attribute set if each record is identical to at least $k-1$ other records (i.e., size of each equivalence class is greater than or equal to k).

Example 4 (k-anonymized Table): Table 3 is 2-anonymous (i.e., $k = 2$).

Local Recoding To protect privacy, each record in the published k-anonymized dataset needs to be identical to at least $k-1$ other records for a set of attributes with respect to their quasi-identifiers. For this, the k-anonymity approach may need to generalize and/or suppress records. This can be achieved by applying either local recoding or global recoding. While in global recoding a particular value is generalized in the same way for all records, local recoding allows this value to be mapped to different generalized values, depending into which equivalence class the associated entry belongs. Since the local recoding method does not overgeneralize an original table to satisfy privacy requirements as compared to the global recoding method, we apply the local recoding method to generalize the values within the original table to prevent a too high loss of data utility. As local recoding can be more demanding in terms of memory usage and processing time, we also investigate its feasibility for our envisioned use case in Section IV.

TABLE 3. An example 2-anonymized transportation table after applying local recoding, reordered to visualise equivalence classes.

Record	Passenger Type	Boarding Station	Alighting Station	Ride Start Time	Ride Time	Ride Distance
r_1	Adult	A	C	9:00	400	3.6
r_2	Adult	A	C	9:00	400	3.6
r_3	C/S	*{B,C}*	D	*[9:05-9:15]*	*[320-340]*	*[2.2-3.8]*
r_6	C/S	*{B,C}*	D	*[9:05-9:15]*	*[320-340]*	*[2.2-3.8]*
r_4	Adult	B	*{A,D}*	10:00	*[280-480]*	*[2-2.2]*
r_7	Adult	B	*{A,D}*	10:00	*[280-480]*	*[2-2.2]*
r_5	SeC	A	D	*[11:00-11:05]*	520	4.2
r_8	SeC	A	D	*[11:00-11:05]*	520	4.2

C/S - Child/Student; SeC - Senior Citizen; *[x-y]* - Generalized Value

When attributes are generalized or entire records are suppressed, the utility of the dataset decreases. Thus, the main challenge is to form equivalence classes with minimal loss of data utility, meaning that data privacy and utility are conflicting goals. Finding an optimal solution for this trade-off in k-anonymity has been shown to be NP-hard [10], [11] (even for small k, e.g., $k = 2$). Researchers have proposed well-working heuristics [12] on which we will base our mechanism.

Example 5 (Local Recoding): Table 3 is a 2-anonymized dataset obtained after applying local recoding on the original dataset in Table 2. Records forming an equivalence class were grouped together with their generalized values indicated in red.

III. SYSTEM MODEL

The paper aims to develop an approach to achieve k-anonymity that generalizes attributes based on their utility with the goal of minimum distortion of the data. Our approach incorporates the data publisher's preferences with regards to privacy and utility by including parameters for both privacy and utility. We make use of the k-anonymity formulation as a clustering problem where each cluster (or equivalence class) contains at least k records and additionally takes into consideration the utility of an attribute during the formation of clusters. Our proposed k-anonymity approach has two modes of operation: manual and automated. In manual mode, the user is given the option to prioritize certain attributes over others, whereas in automated mode, we use the entropy of an attribute to determine its priority.

In what follows, we give a description of all the components of our k-anonymity mechanism. A table of all used

parameters and variables alongside their descriptions can be found in the appendix in Table 5.

A. UTILITY FACTOR OF AN ATTRIBUTE

In order for our k-anonymity mechanism to consider utility levels, it needs to either be given a utility value for each attribute or able to derive a ranking itself.

In the former case – or manual mode – the publisher decides whether an attribute is assigned a high or low utility factor based on their requirements. For example, if a user prefers information about ride start time to be preserved, they would assign a higher utility factor for the ride_start_time attribute as compared to other attributes. The user can assign a value U_a, $1 \leq U_a \leq U_{\max}$ (for example, $U_{\max} = 5$), where 1 is the lowest utility factor, and $U_{\max}$ the highest. $U_{\max}$ can be chosen based on the required granularity of the relation of attributes.

If the user does not wish to provide manual utility factors, we utilize the Shannon entropy of an attribute, which is a measure of information content [13], where a higher entropy value represents a higher degree of information content. The utility factor (or entropy) of an attribute a, denoted as U_a, is:

$$U_a = \sum_{i \in I} -p_i log_2(p_i) \tag{1}$$

where p_i is the probability of the attribute to take value i ($i \in I$), I being the set of values found for this particular attribute across the dataset. For continuous attributes (and possibly also discrete attributes), the values are discretized into buckets of fixed or flexible size, based on the underlying value distribution to create smaller buckets where the density of the values is higher and larger buckets where the data is sparse.

In general, to calculate the utility factor of an attribute based on entropy, the attribute values within the dataset should be normalized, as two attributes with a large difference in possible values will also have significantly different maximum entropy values.

The utility factor U_a will then be used to influence the generalization process. An attribute with a lower utility factor will be given less weight in terms of utility (or higher level of generalization). The weight w_a of an attribute a is simply its utility factor U_a normalized using the utility factors of all other attributes $a \in A$.

$$w_a = \frac{U_a}{\sum_{i \in A}(U_i)} \tag{2}$$

This weight can then be used when computing distances (Section III-B) and distortion (Section III-D).

B. DISTANCE METRICS

When records are grouped together during generalization, their generalized attributes will be changed to a range between at least the minimum and maximum of their values before generalization. It is therefore important to be able to measure the distance between two values so the algorithm will preferably merge entries closer to each other to reduce the impact on the data quality during generalization. In this section, we provide details on distance metrics considered for different types of attributes.

1) NON-NORMALIZED DISTANCE

The used distance metric is dependent on the attribute type and attribute representation structure. As discussed in Section II, there are different attribute types such as categorical and numerical. Additionally, there may be cases where an attribute type can be represented in different ways, i.e., there are multiple options for the attribute representation structure. For example, a categorical attribute might be represented by either a tree or a graph. The reason behind choosing these separate representations for categorical attributes is that in a tree structure, a domain expert can introduce additional information through deciding which node is a parent of another. For the purpose of applying a distance metric, the nodes could for example hold the value according to their depth in the tree, e.g., a leaf node having the largest value and the root node being represented by a zero. For a graph structure (e.g., for stations in a train network) the vertices usually take the actual values (e.g., the coordinates of a station or their name) of an attribute. Our system model therefore needs to support all attribute types and attribute representation structures by providing distance metrics for each of them.

Example 6 (Attribute Representation Structure): For categorical attributes such as Boarding Station and Alighting Station (from Table 2) taking stations/values A, B, C and D, are represented using a graph structure as shown in Figure 1. The vertices are station name and edges represent the distance and connectivity between stations. A categorical attribute such as Passenger Type, where a domain expert introduces a hierarchy and additional values, such as 'Not Adult' and 'Person', is represented using a tree structure (as shown in Figure 2).

We are now ready to define a non-normalized distance metric for an attribute:

1) Numerical Attributes: The non-normalized distance between two numerical values v_i^N and v_j^N of numerical attribute N is calculated as:

$$D^N(v_i^N, v_j^N) = ||v_i^N - v_j^N|| \tag{3}$$

where $||v_i^N - v_j^N||$ is the distance calculated using a metric such as the Euclidean distance, Manhattan distance, etc.

2) For attributes represented by a tree, the hierarchical tree structure can be constructed based on the frequency of leaf nodes, i.e., leaf nodes with lower frequency are combined to form a common ancestor (for example, in Figure 2, two leaf nodes 'Senior Citizen' and 'Child/Student' are combined to form ancestor 'Not Adult'). The non-normalized distance between any two values (i.e., leaf nodes) of a categorical attribute represented by a tree is captured by subtracting 1 from the

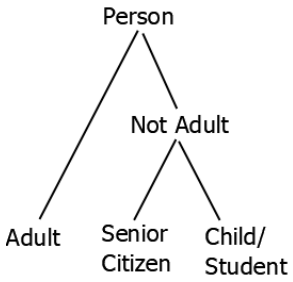


FIGURE 2. Generalization Hierarchy for Passenger Type Attribute.

number of leaf nodes at their common ancestor. For categorical attributes represented by a tree, we calculate the non-normalized distance between two values (v_i^T and v_j^T) as:

$$D^T(v_i^T, v_j^T) = L_{ij}^T - 1 \tag{4}$$

where L_{ij}^T is the number of leaf nodes at the common ancestor of v_i^T and v_j^T.

Example 7: From Table 2 and Figure 2, the non-normalized distance between two values 'Senior Citizen' (in record r_5) and 'Child/Student' (in record r_6) of Passenger Type attribute is = 2-1 = 1.

3) For attributes represented by a graph, the distance between any two values of a categorical attribute is captured by the distance between two vertices in that graph, measuring the shortest path that connects them. For a categorical attribute represented by a graph G, we calculate the non-normalized distance between two values (v_i^G and v_j^G) as:

$$D^G(v_i^G, v_j^G) = sp(v_i^G, v_j^G) \tag{5}$$

where, $sp(v_i^G, v_j^G)$ is the shortest path between v_i^G and v_j^G.

Example 8: From Table 2 and Figure 1, the non-normalized distance between two values 'A' (in record r_1) and 'B' (in record r_3) of the boarding station attribute is 1. Similarly, the non-normalized distance between two values 'A' (in record r_4) and 'D' (in record r_7) of the alighting station attribute is 2.

For weighted graphs $sp(v_i^G, v_j^G) = \sum_{k=1}^{n} w(v_{k-1}^G, v_k^G)$, where $w(v_0^G, v_1^G), \ldots, w(v_{n-1}^G, v_n^G)$ represent the weight of each edge on the shortest path between v_i^G and v_j^G. The weight can represent the actual distance between two nodes, time taken to travel from one node to another, profit or loss made when sending goods from one node to another, etc. Another approach is to use directed graphs where the distance between nodes is the length of the shortest directed path between them, provided at least one such path exists.

4) For categorical attributes that cannot be represented using any attribute representation structure (e.g., a binary attribute), we use the following equation to calculate the distance between any two values v_i and v_j:

$$D(v_i, v_j) = \begin{cases} 0 & v_i = v_j \\ 1 & v_i \neq v_j \end{cases} \tag{6}$$

For the sake of readability, we decided to not include this type in the later sections of this article as our dataset does not contain an attribute of this type.

2) NORMALIZED DISTANCES

Since the non-normalized distance values for attributes are not guaranteed (and also are unlikely) to be on the same scale, we need to normalize them first to be able to compute a *fair* distance between two records without over-representation of one attribute over another. For example, if one attribute was travel time in milliseconds and another attribute was age in years, then without normalization the travel time would be much more prominent and the difference in age would have no impact on an overall distance between two records.

To normalize the distance, we apply the following two methods:

1) Maximum distance: Divide the non-normalized distance by the maximum distance.
   - Numerical Attribute: These attributes can be normalized by their domain size (difference between maximum and minimum values in the domain of a numerical attribute). The normalized distance between v_i^N and v_j^N using the maximum distance method is:

$$\widehat{D^N}(v_i^N, v_j^N) = \frac{D^N(v_i^N, v_j^N)}{max(N) - min(N)} \tag{7}$$

   where the denominator is the domain size of the numerical attribute N.
   - Attribute represented by a tree: The normalized distance between v_i^T and v_j^T using the maximum distance method is:

$$\widehat{D^T}(v_i^T, v_j^T) = \frac{D^T(v_i^T, v_j^T)}{L^T - 1} \tag{8}$$

   where L^T is the total number of leaf nodes in the taxonomy tree of categorical attribute T.
   - Attribute represented by a graph: The normalized distance between v_i^G and v_j^G using the maximum distance method is:

$$\widehat{D^G}(v_i^G, v_j^G) = \frac{D^G(v_i^G, v_j^G)}{Diam(G)} \tag{9}$$

   where $Diam(G) = \max_{v_i^G, v_j^G \in G} D^G(v_i^G, v_j^G)$ is the graph diameter of categorical attribute G.
2) Zero-mean unit-variance: In this method, we first calculate the pairwise distance for each record w.r.t. an attribute, and then utilize the mean and standard deviation of pairwise distances to normalize the distance. This method is required to avoid domination

of one attribute over another as we will show in Section IV.

- Numerical Attributes: The normalized distance between v_i^N and v_j^N using the zero-mean unit-variance method is:

$$\overline{D^N}(v_i^N, v_j^N) = \frac{D^N(v_i^N, v_j^N) - \mu(D^N)}{\sigma(D^N)} \tag{10}$$

  where, $\mu(D^N)$ and $\sigma(D^N)$ are the mean and the standard deviation of the pairwise distance of all values present in the table for numerical attribute N.
- Attributes represented by a tree: The normalized distance between v_i^T and v_j^T using the zero-mean unit-variance method is:

$$\overline{D^T}(v_i^T, v_j^T) = \frac{D^T(v_i^T, v_j^T) - \mu(D^T)}{\sigma(D^T)}, \tag{11}$$

  where $\mu(D^T)$ and $\sigma(D^T)$ are the mean and the standard deviation of the pairwise distance of categorical tree-represented attribute T.
- Attributes represented by a graph: The normalized distance between v_i^G and v_j^G using the zero-mean unit-variance method is:

$$\overline{D^G}(v_i^G, v_j^G) = \frac{D^G(v_i^G, v_j^G) - \mu(D^G)}{\sigma(D^G)}, \tag{12}$$

  where $\mu(D^G)$ and $\sigma(D^G)$ are the mean and the standard deviation of the pairwise distance of categorical graph-represented attribute G.

C. UTILITY-WEIGHTED RECORD DISTANCE

Now that we are able to compute the difference (i.e, distance) between two values of any attribute type, we need to define the distance between two records. This is required to identify records for an equivalence class with minimum generalization. Consider, $A = \{N_1, \ldots, N_p, T_1, \ldots, T_q, G_1, \ldots, G_r\}$ to be the set containing attributes of the original table T, where $N_i (i = 1, 2, \ldots, p)$ are the numerical attributes, $T_j (j = 1, 2, \ldots, q)$ are the categorical tree attributes, and $G_k (k = 1, 2, \ldots, r)$ are the categorical graph attributes. Then, the non-normalized distance between two records r_1 and r_2 is:

$$\begin{aligned} D(r_1, r_2) = &\sum_{i=1}^{p} w_{N_i}(D^{N_i}(r_1[N_i], r_2[N_i])) \\ &+ \sum_{j=1}^{q} w_{T_j}(D^{T_j}(r_1[T_j], r_2[T_j])) \\ &+ \sum_{k=1}^{r} w_{G_k}(D^{G_k}(r_1[G_k], r_2[G_k])), \end{aligned} \tag{13}$$

where D^{N_i}, D^{T_j}, D^{G_k} are the distances between the different attributes. $r_1[N_i]$, $r_2[N_i]$ are the values of the i^{th} numerical attribute (N_i) for records r_1 and r_2. Similarly, $r_1[T_j]$, $r_2[T_j]$ are the values of the j^{th} categorical attribute represented using a tree (T_j), and $r_1[G_k]$, $r_2[G_k]$ are the values of the k^{th} categorical attribute represented using a graph (G_k). Analogue to computing the non-normalized distance $D(r_1, r_2)$, we compute the normalized distances $\widehat{D}(r_1, r_2)$ and $\overline{D}(r_1, r_2)$ by simply using the respective normalized distance functions for each attribute as defined in the previous section. The weights w_{N_i}, w_{T_j}, and w_{G_k} are used to weigh each attribute according to its utility as introduced in Section III-A.

D. DISTORTION

The last metric required for our k-anonymity approach is a metric to measure how much the underlying dataset was distorted due to generalization and suppression. We define the total information distortion (normalized) Γ_{tot} in a k-anonymized table:

$$\Gamma_{\text{tot}} = \Gamma_g + (1 - \Gamma_g) * \Gamma_s \tag{14}$$

where Γ_g is information distortion caused by generalization and Γ_s is information distortion caused by record suppression. By incorporating $(1 - \Gamma_g)$ into the right side of the equation, we ensure a smoother transitioning between distortion values. To calculate Γ_g, we first calculate the information distortion in an equivalence class. The information distortion Γ_g^c in an equivalence class c is given as:

$$\begin{aligned} \Gamma_g^c = |c| * (&\sum_i w_{N_i} \frac{\max(N_i(c)) - \min(N_i(c))}{\max(N_i) - \min(N_i)} \\ &+ \sum_j w_{T_j} \frac{w_{lca} * (L_{lca}^{T_j}(c) - 1)}{L^{T_j} - 1} \\ &+ \sum_k w_{G_k} \frac{max(sp(\cup_{G_k}(c)))}{Diam(G_k)}) \end{aligned} \tag{15}$$

where $\max(N_i(c))$ and $\min(N_i(c))$ are the maximum and minimum values of attribute N_i in c and $\max(N_i)$ and $\min(N_i)$ are the maximum and minimum values of attribute N_i in general. The second term measures the distortion in tree-based attributes with $L_{lca}^{T_j}(c)$ as the number of leaf nodes at the least common ancestor for values of attribute T_j in equivalence class c, and L^{T_j} is the total number of leaf nodes in attribute T_j. For the computation of graph-based attribute distortion in the third term we make use of $\cup_{G_k}(c)$ as the union set of values in c with respect to attribute G_k. Then, $\max(sp(\cup_{G_k}(c)))$ and $Diam(G_k)$ are the maximum values of the shortest paths between two values in c with respect to G_k and the diameter of the graph that represents categorical attribute G_k.

To account for the utility in the computation of the information distortion, we have used the weights assigned to attributes in the automated mode, i.e., w_{N_i}, w_{T_j}, and w_{G_k} are the weights assigned to attributes N_i, T_j, and G_k.

Let EC be the set of all equivalence classes in the anonymized table T_{ano}. Then the information distortion due to generalization in T_{ano} is:

$$\Gamma_g = \sum_{c \in EC} \frac{\Gamma_g^c}{|T_{\text{ano}}|} \tag{16}$$

where $|T_{\text{ano}}|$ is the number of records that are published, i.e., the size of k-anonymity table. To calculate the information distortion caused by record suppression, we use the following method:

$$\Gamma_s = \frac{S}{|T|} \tag{17}$$

where S is the number of records suppressed and $|T|$ is the total number of records in the original table.

While the total distortion gives information about the entirety of the dataset, it might be required to measure the distortion of single attribute to better understand how the data was affected by the k-anonymity approach.

The information distortion in attribute a is calculated as:

$$\Gamma^a = \Gamma^a_g + (1 - \Gamma^a_g) * \Gamma^a_s \tag{18}$$

where Γ^a_g is the information distortion due to generalization and Γ^a_s is the information distortion due to suppression. To calculate the information distortion due to suppression, we use equation 17. The information distortion due to generalization is calculated as:

$$\Gamma^a_g = \frac{\sum_{c \in EC} |c| * \Gamma(\hat{a}^c)}{|T_{\text{ano}}|} \tag{19}$$

where $\hat{a}^c$ represents the generalized value of attribute a in equivalence class c, and $\Gamma(\hat{a}^c)$ is calculated as:

$$\Gamma(\hat{a}^c) = \begin{cases} \dfrac{\max(\hat{a}^c) - \min(\hat{a}^c)}{\max(a) - \min(a)} & a \text{ is numerical} \\ \dfrac{\max\limits_{p,q \in \hat{a}^c; p \neq q} sp(p,q)}{Diam(a)} & a \text{ is categorical (graph)} \\ \dfrac{L(\hat{a}^c) - 1}{L^a - 1} & a \text{ is categorical (tree)} \end{cases} \tag{20}$$

where $\max(\hat{a}^c)$ and $\min(\hat{a}^c)$ are the maximum and the minimum of the generalized value of the numerical attribute. $\max(a)$ and $\min(a)$ refer to the maximum and the minimum value of the numerical attribute before generalization. $sp(p, q)$ is the shortest path between p and q, and $Diam(a)$ is the diameter of the attribute a represented by the graph. $L(\hat{a}^c)$ and L^a are the number of leaf nodes at the generalized value and total number of leaf nodes of the tree attribute.

E. FORMATION OF EQUIVALENCE CLASSES

We can now make use of the attribute weights (Section III-A) and distance metrics (Section III-B) to form equivalence classes such that each equivalence class has a size greater than or equal to k. To form equivalence classes, we implement and compare the two following clustering algorithms: *greedy* and *density*. The process for the formation of an equivalence class is as follows:

1) Selection of cluster head:
   a) In *greedy*, select a random record r as cluster head.
   b) If *density*, calculate the average distance of k nearest neighbors of each record and select record r with the lowest average distance as a cluster head. The average distance μD for record r in a cluster of records K is calculated as:

$$\mu D_r = \frac{\sum_{\hat{r} \in K} D(r, \hat{r})}{|K|} \tag{21}$$

   where $D(r, \hat{r})$ is the distance between record r and $\hat{r}$ as introduced in Section III-B.
2) Selection of cluster members: We can have 2 cases:
   a) When record suppression is not desired, the user does not impose constraints and the algorithm will select $k - 1$ neighbor records of cluster head with minimum distortion.
   b) When record suppression is allowed, the user imposes constraints and the algorithm selects $k - 1$ neighbor records of cluster head with minimum distortion such that no constraints are violated.
3) Repeat above steps until all records are covered, i.e., either assigned an equivalence class or suppressed.

There are different ways to suppress records, such as:

C1: defining an attribute distortion limit, i.e., associating a value with each attribute to limit its generalization [14].
C2: defining a suppression threshold as a percentage of records that can be suppressed during the anonymization process [15] or as a disclosure risk value for each record [16].
C3: defining a level of information loss based on statistical measures (for example, mean, mode, median for numerical attributes), prior to the k-anonymization process.

For the sake of simplicity, all the above methods are called constraints. In this article, we focus on record suppression by applying constraint C1, i.e., distortion limits are defined for each attribute as indicated by the user. To apply constraint C1 on any attribute, we set a limit by which that attribute value can be generalized. If the generalization crosses that limit, then we simply do not form an equivalence class and suppress that record. For example, the user could impose a limit on the *Boarding Station (BS)* attribute to limit the generalization of any given boarding station to four hops. In this case, an equivalence class with two boarding stations A and B will satisfy the constraint if the distance between them is less than or equal to four hops. If the constraint is not satisfied, then the record holding value A is suppressed, given that this record was selected a cluster head. Other records assigned to this cluster head can be reassigned to different clusters.

Algorithm 1 provides the pseudo-code of our proposed k-anonymity approach with following details - (A). Assigning a weight to each attribute (lines 4-8); (B) Formation of equivalence classes (lines 9-29); (C) Generalization / Record Suppression (lines 30-33); and (D) Generation of the k-anonymized table (line 34). The *equivalence_set* consists of all the equivalence classes, and attribute values within

Algorithm 1 Proposed k-Anonymity Approach

Input: Dataset T; k; Generalization hierarchies; Distance metrics; Constraints; Mode of Operation.
Output: T_{ano} (k-anonymized Table).
if $|T| \geq k$ **then**
    if Mode of Operation == manual **then** ▷ Assigning a weight of each attribute
        User-assigned attribute weights
    else
        Determine the utility factor based on entropy.
        Determine the weight of an attribute.
    equivalence_set = {}; *initial_set* = {records in T}; *suppression_set* = {}. ▷ Formation of Equivalence Classes
    while $|initial\_set| \neq 0$ **do**
        if $|initial\_set| \geq k$ **then**
            if *greedy clustering* **then**
                Select a record (r) from initial set and its '$k-1$' neighbors
            else
                Calculate the average distance of each record.
                Select the record with lowest average distance and its '$k-1$' neighbors.
            if k-anonymity with record suppression **then**
                if constraints satisfied **then**
                    Delete the selected record and its '$k-1$' neighbors from *initial_set*.
                    Move these records into *equivalence_set* as equivalence class.
                else
                    Delete the selected record from *initial_set*.
                    Move the selected record into *suppression_set*.
            else
                Delete the selected record and its '$k-1$' neighbors from *initial_set*.
                Move these records into *equivalence_set* as equivalence class.
        else
            Move record r into the best equivalence class
            Delete record r from initial set
    for c in *equivalence_set* **do** ▷ Generalization
        Generalize the attribute values in c.
    for record in *suppression_set* **do** ▷ Record Suppression
        Suppress the record.
    Generate the k-anonymized table ▷ k-anonymized table
else
    Number of records within dataset is less than k

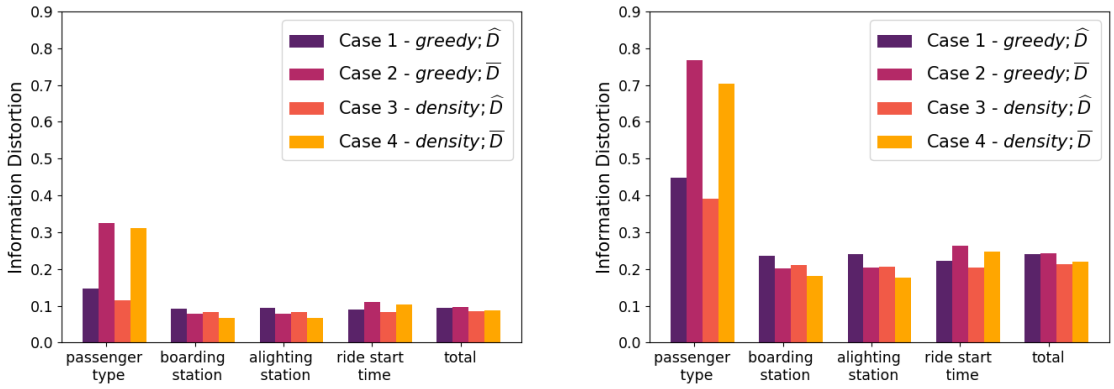
these equivalence classes are generalized. The records within *suppression_set* are suppressed.

IV. RESULTS AND DISCUSSIONS

We discuss the experiments conducted to evaluate the proposed approach. We apply the proposed approach on a real-world public transportation dataset (Section II). We consider tap-in and tap-out data recorded on a single day. Before the experiments, the dataset was pre-processed, where we:

- Cleaned up the dataset to remove obvious errors, e.g., removing journeys outside the operation of the public transport system, negative travel time trips, etc.
- Select four attributes (passenger_type as categorical attribute represented as tree; boarding_stop_stn and alighting_stop_stn as categorical attributes represented as graph; and ride_start_time_seconds as numerical attribute).
- To calculate the entropy, ride_start_time_seconds attribute is discretized by considering an interval of 900 seconds.

For the evaluation of our mechanism, we have removed the ride_time and ride_distance as they can be directly derived from the boarding_stop_stn and alighting_stop_stn attributes. While such direct dependencies may be trivial to discover, others may not be. In a public transport dataset, many, if not all, attributes can exhibit correlation to one or more other attributes. For example, seniors may be less likely to travel during the morning rush hour, stations in a residential area are more likely to be destinations in the evening hours, or the choice of boarding station may be used to predict the destination station to a certain extent. These dependencies



(a) k = 5 (b) k = 20

FIGURE 3. Attribute information distortion and total information distortion (automatic utility weight assignment); Dataset size, $|T| = 10$K records.

may even change, vanish, or be reversed over the course of a day, week, or year. While we do not incorporate a specific mechanism to incorporate these correlations into the utility metric, we avoid a privacy attack where one attribute could be used to de-generalize another attribute by treating every attribute as a quasi-identifier, requiring each entry in an equivalence class to be identical with respect to every attribute.

A. CLUSTERING AND DISTANCE NORMALIZATION

In this first experiment, we compare the different distance normalization and clustering methods without suppression of records during k-anonymization. Since we have introduced two distance normalization methods (max distance $\widehat{D}$ and zero-mean unit variance $\overline{D}$) and two different clustering methods (*greedy* and *density*), we have a total of four possible configurations.

Figure 3 shows the information distortion in each attribute and the total information for k = 5, 20. Utility weights were assigned automatically according to Section III-A. From Figures 3a and 3b, we first observe that the zero-mean unit variance distance normalization ($\overline{D}$) performs better than the configurations using the max distance normalization ($\widehat{D}$). The former method seems to sacrifice more of the passenger type attribute to conserve information in the others, in particular the boarding and alighting stations. Additionally, the ride start is slightly more distorted as it seemed to dominate the other distances for other attributes when using the max distance normalization method.

In the zero-mean unit variance method for distance normalization, the distances for each attribute type is standardized by their mean and standard deviation, reducing the effect of a dominating distance metric when multiple distance metrics are combined. In comparison, the max distance normalization is more sensitive to the minimum and maximum values and the presence of outliers as they would compress the distance values in a narrower range. The robustness of the zero-mean unit variance method to this problem is reflected in the results.

Comparing *greedy* vs *density* clustering, we observe that *density* clustering outperforms *greedy*. This was expected as the *greedy* method is a naive approach that may choose sub-optimal cluster heads leading to unnecessarily coarse generalization, while *density* tries to select a cluster head which is close to at least $k - 1$ other records.

While there are other underlying factors such as attribute homogeneity and value distribution that determine by how much the *greedy* method will perform worse, we found that for a general public transport dataset with a large number of records, it seems to be a viable alternative.

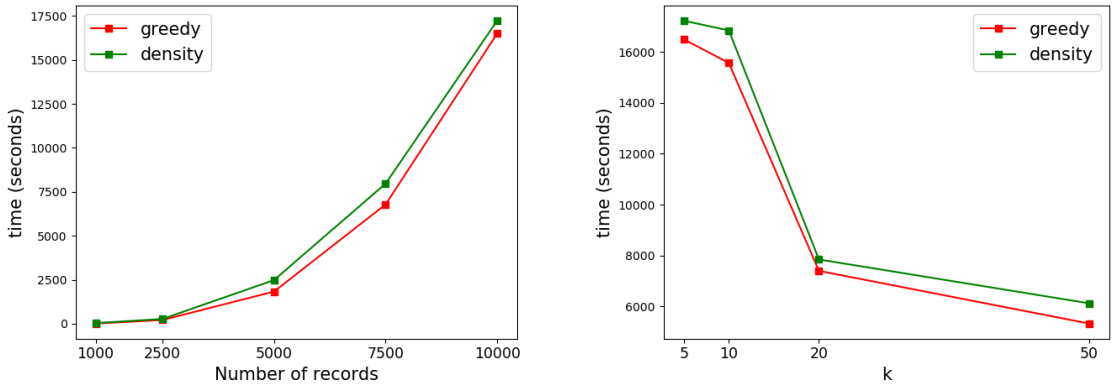
To understand whether one of the methods offers a benefit in terms of computing resources or whether our distance metrics have an effect on the wall-time of either approach, we compared their run-time in Figure 4. We found that no significant differences can be observed between the two approaches, regardless of the dataset size or the privacy parameter k.

We conclude that to achieve lowest distortion during the generalization process, a configuration consisting of the *density* clustering approach combined with zero-mean unit variance normalization performs best. For the remainder of this experiment section, we therefore make use of this configuration.

B. UTILITY-DRIVEN GENERALIZATION

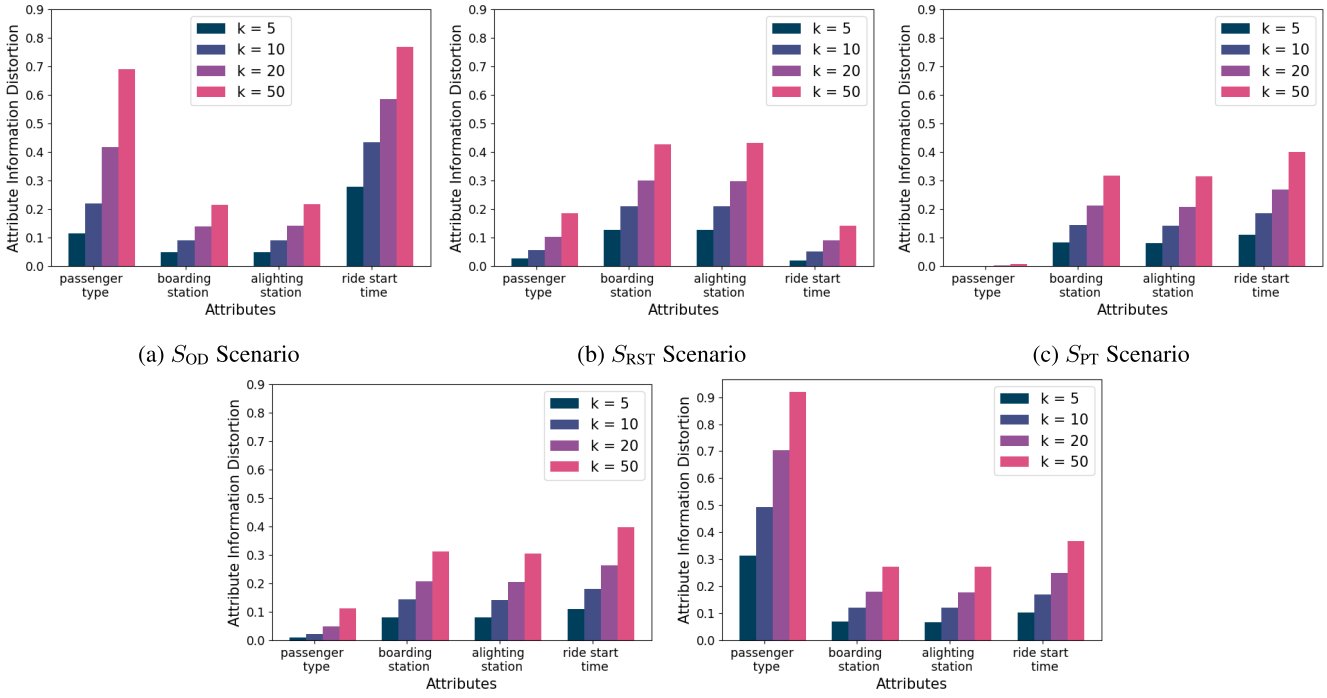
In this set of experiments, we analyze the performance of our utility-driven k-anonymity approach without suppressing any records during k-anonymization. To test the various aspects of our approach, we consider four different scenarios in manual mode (where attribute utility factors are user-defined) and one scenario in automated mode (where attribute utility factor is determined by the system using entropy):

1) Origin-Destination scenario S_{OD}: In this manual scenario, the user wishes to maintain information about boarding and alighting station attributes. The utility factors of boarding_stop_stn and alighting_stop_stn attributes are set to 5, and 1 for all other attributes.



(a) k = 5 (b) $|T|$ = 10k records

FIGURE 4. Simulation time (in seconds) for *greedy* and *density* methods.



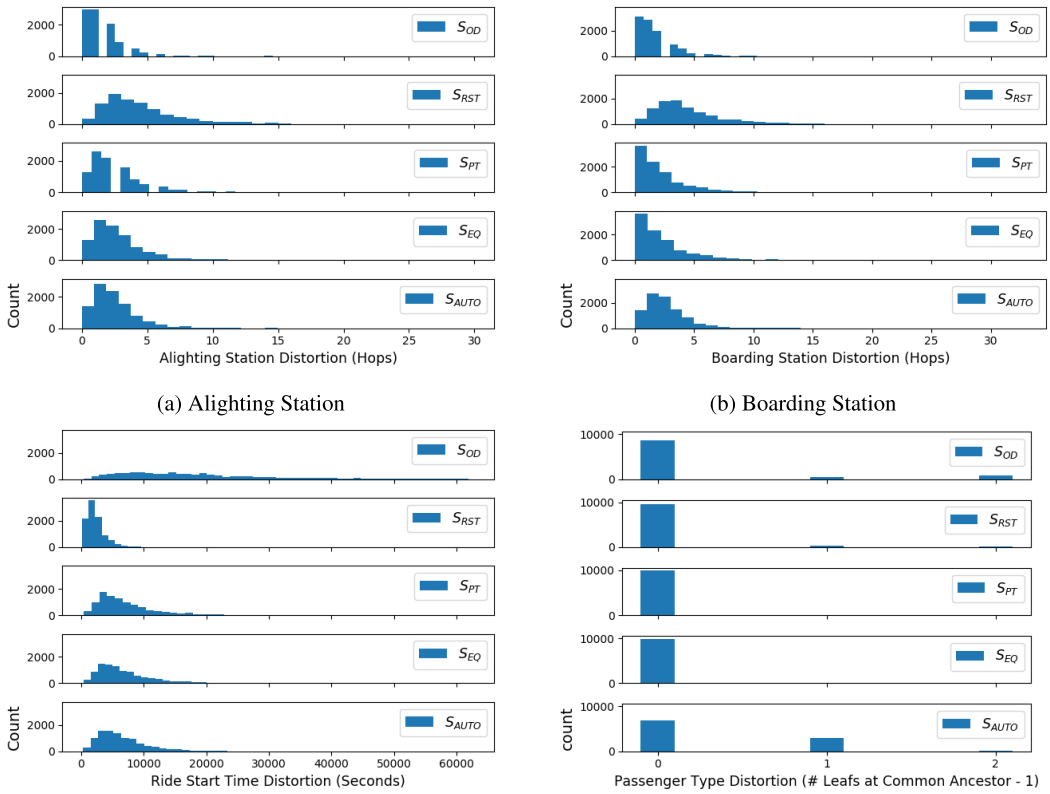
(a) S_{OD} Scenario (b) S_{RST} Scenario (c) S_{PT} Scenario

(d) S_{EQ} Scenario (e) S_{AUTO} Scenario

FIGURE 5. Attribute information distortion for manual and automated utility weights; dataset size, $|T| = 10$K records.

2) Ride Start Time scenario S_{RST}: In this scenario, the user wishes to maintain information about the ride start time attribute. The utility factor of ride_start_time_seconds attribute is set to 5, and 1 for all other attributes.
3) Passenger Type scenario S_{PT}: In this scenario, the user wishes to maintain the information about passenger type attribute, setting the utility factor of passenger_type attribute is set to 5, and 1 for all others
4) Equal scenario S_{EQ}: In this manual scenario, all attributes are given equal utility weights, where the user does not have any preference in which attribute should be preserved.
5) Automated scenario S_{AUTO}: In this automated scenario, attributes are preserved based on their utility determined by the system. The user does not provide any specific weights to attributes, instead the weights are determined automatically based on entropy as described in Section III-A.

Table 4 mentions the weights assigned to attributes during each scenario (manual and automated mode). Please note that these weights have been normalized according to Equation 2. This means that if the user assigns a weight of 5 to both Boarding and Alighting Station and a weight of 1 to Passenger Type and Ride Start Time (see scenario S_{OD}), their



(a) Alighting Station

(b) Boarding Station

(c) Ride Start Time

(d) Passenger Type

FIGURE 6. Attribute distortion (distribution); Dataset size, $|T| = 10K$ records; Privacy parameter, $k = 5$.

TABLE 4. Normalized attribute weight in each scenario.

Scenario	Passenger Type	Boarding Station	Alighting Station	Ride Start Time
S_{OD}	0.083	0.416	0.416	0.083
S_{RST}	0.125	0.125	0.125	0.625
S_{PT}	0.625	0.125	0.125	0.125
S_{EQ}	0.25	0.25	0.25	0.25
S_{AUTO}	0.036	0.328	0.325	0.309

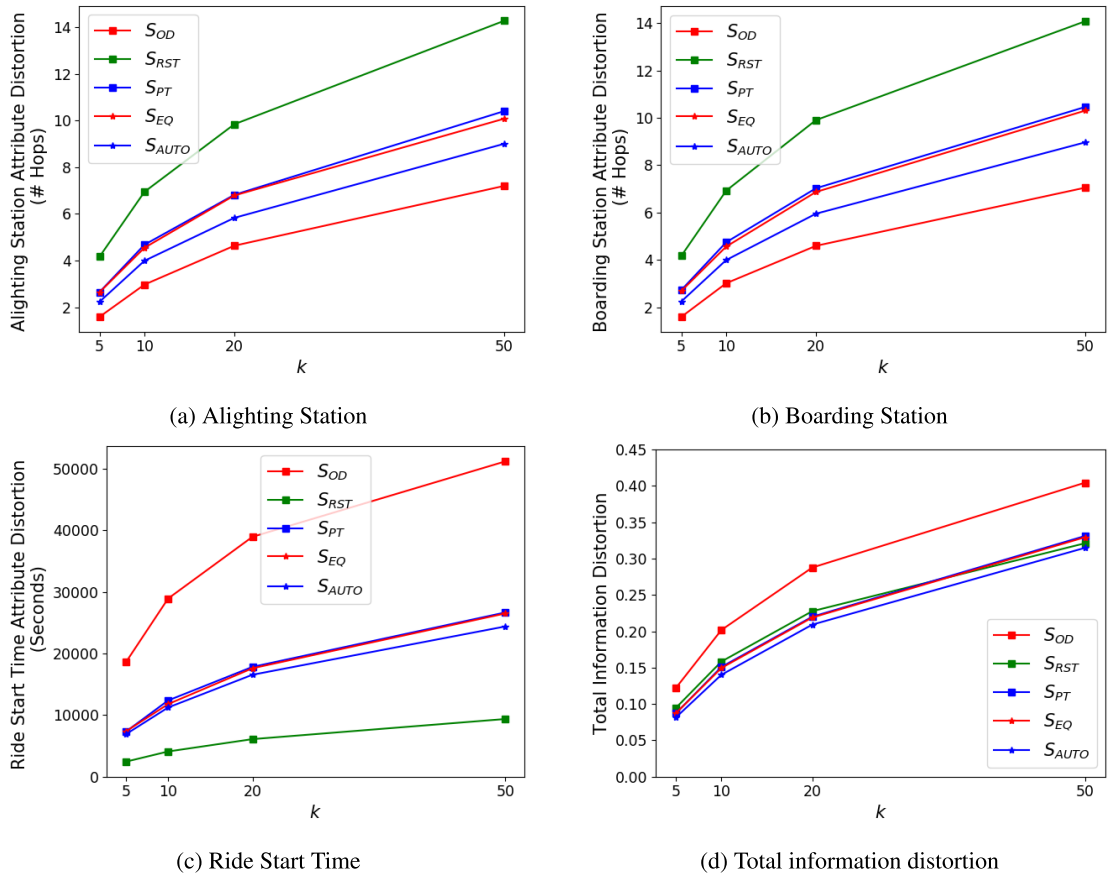
actual weights will be $5/12 \approx 0.416$ and $1/12 \approx 0.083$, respectively.

Figure 5 shows the information distortion in each attribute at $k = 5, 10, 20, 50$ for all the scenarios considered. Naturally, information distortion grows with k as the minimum number of records per equivalence class is larger, requiring more coarse generalization. In terms of utility, each scenario introduces less information distortion in the attribute to which the user assigned a higher utility weight, showing how our approach effectively allows the user to preserve certain attributes better. For example, in scenario S_{OD}, we see that less information distortion is introduced in the boarding and alighting station attributes when compared to other attributes.

In scenario S_{PT}, low information distortion is introduced in the passenger type attribute (0.0 for $k = 5$, 0.007 for $k = 50$), which is due to the fact that the domain size of passenger type attribute is small. Also, scenario S_{AUTO} shows that the information in each attribute is distorted as per its utility, i.e., the passenger type attribute which holds less information is distorted more (0.31 for $k = 5$, 0.91 for $k = 50$) than the ride start time attribute, and boarding and alighting stations are distorted significantly less.

If we compare attribute distortion in the different scenarios (Sub-figures 5a to 5e), we see that, for example, S_{OD} only introduces a distortion of 0.04 for $k = 5$, while S_{RST} already exhibits a distortion in the same attribute of 0.12, caused by attribute prioritization in the generalization process. In S_{AUTO} this distortion lies in-between these two scenarios with a distortion value of 0.06. These trends can also be observed for higher values of k and for other attributes in their respective prioritization scenario.

Figure 6 shows histograms of the attribute distortion for each of the five utility configurations and in Figure 7 we plot the average attribute distortion. Again, the results confirm that our approach is able to take user preferences into consideration in the generalization process, evidenced by the lower distortion of the attributes in the respective configurations. We note that the information distortion is not zero in the respective manual mode, because the shown attributes have a higher domain size, and the number of records considered for experiments is lower. This leads to the situation that there are not enough records to form an equivalence class without also generalizing these attributes. Additionally, our configurations chose a value of 5 for the prioritized attribute, leading to a



(a) Alighting Station (b) Boarding Station

(c) Ride Start Time (d) Total information distortion

FIGURE 7. Average attribute (a-c) and total (d) distortions; Dataset size, $|T| = 10$K records.

ratio of 1 to 5 in the distance metrics. This means that at one point, the distance of the remaining records can be too large in the low priority attributes to outweigh the smaller distance in the priority attribute.

Figure 7d shows the actual information distortion obtained to satisfy the privacy parameter $k \in \{5, 10, 20, 50\}$ for the database with 10k records. We observe that scenario S_{AUTO} exhibits the lowest information distortion because the weights for the attributes are automatically assigned based on their utility. On the other hand, the scenario S_{OD} shows the highest information distortion. This is induced by the distribution of the other attributes (ride start time, passenger type) of the dataset, causing the approach to generalize attributes that are not close together to maintain origin and destination stations.

C. RECORD SUPPRESSION

In our last set of experiments, we analyze the performance of the proposed algorithm when we allow record suppression during k-anonymization. For record suppression, we introduce limits on the attribute value.

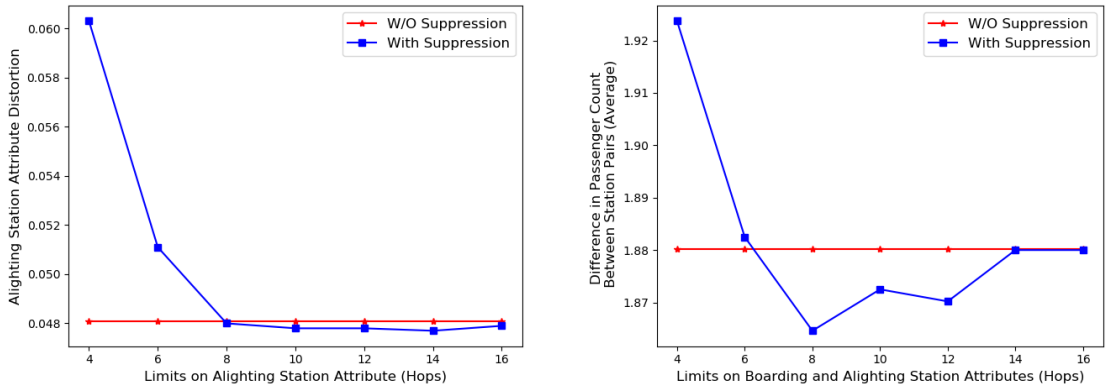
For this experiment, we consider following two configurations:

- For scenario S_{OD}, we limit the boarding and alighting station attribute values to 4, 6, 8, 10, 12, 14, and 16 hops.
- For scenario S_{RST}, we limit the ride start time attribute values to 3600, 5400, 7200, 9000, 10800, 12600, and 14400 seconds.

In addition to the distortion metrics introduced in Section III-B we also take into consideration application-specific metrics to gain better insights into the effects of record suppression:

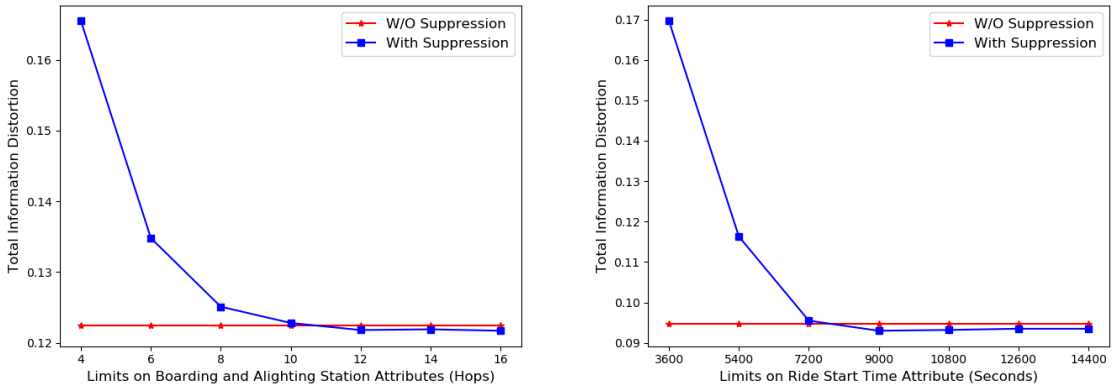
- For scenario S_{OD}: The average of difference in number of passengers travelling between the top 20 pairs of boarding and alighting stations between original and k-anonymized table.
- For scenario S_{RST}: The difference in mean ride start time between original and k-anonymized table. Since we generalize the ride start attribute value into intervals $[\text{rst}_{\text{min}}, \text{rst}_{\text{max}}]$, we shall take minimum value (rst_{min}) to determine the mean ride start time of the k-anonymized table.

Figure 8 shows the distortion in the S_{OD} scenario when suppression thresholds of 4 to 16 are introduced. Figure 8a shows the distortion using the general metric and Figure 8b makes use of the application-specific metric. In both figures we observe that suppression can improve data utility as evidenced by lower values for the distortion of the blue line for threshold values of 8 and higher. This means that sometimes it is better to suppress a record than adding this record



(a) Alighting Station Distortion (b) Difference in Passenger Count

FIGURE 8. Record Suppression in S_{OD} Scenario; Dataset size, $|T| = 10K$; $k = 5$.



(a) S_{OD} (b) S_{RST}

FIGURE 9. Total Information Distortion with and without record Suppression; Dataset size, $|T| = 10K$ records; $k = 5$.

to an equivalence class, causing a coarse generalization of an attribute. To illustrate this, assume an equivalence class c with an attribute a to be generalized to the range [4, 8]. If the only remaining unassigned record has a value for $a = 10$, then adding this record to c, would cause a to be changed to [4, 10] for all members of c, introducing higher distortion. We also observe that information distortion is high when the suppression threshold is low. A low threshold causes the k-anonymization approach to drop a large number of records as it would be unable to form equivalence classes without generalizing attributes more than the threshold.

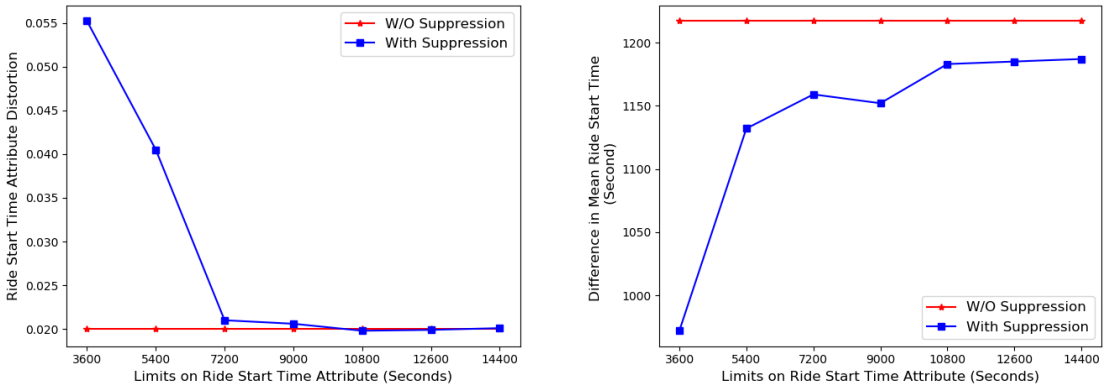
The S_{RST} scenario shown in Figure 10 exhibits similar behaviour in terms of attribute distortion. However, the application-specific metric in Figure 10b suggests that record suppression always performs better. This is caused by this specific metric which always prefers less generalization (i.e., smaller alterations to the ride-start time), showing that applying biased metrics can lead to misleading results. The general-purpose metric takes into account the number of suppressed records and shows that only thresholds of 10800 and higher lead to an improvement in distortion.

Lastly, we analyzed whether this lower attribute distortion will translate to a lower total distortion in both scenarios. Figure 9 confirms this and shows that record suppression lowers the total distortion, both for the S_{OD} and S_{RST} scenario.

In this experiment we manually applied the suppression threshold for an attribute, however, we like to highlight the need for methods that can automatically determine these thresholds. One way to achieve that is to take into account the pairwise distance between attribute values and then use the mean, mode, or other measures of the pairwise distances as a suppression threshold, instead of an iterative manual assignment of a threshold value and a subsequent check whether distortion improved.

V. RELATED WORK

There exists a wide range of k-anonymity privacy mechanisms using different methods to achieve the forming of equivalence classes. For example, researchers have proposed hierarchy-based generalization [17], [18], partition-based generalization [10], and clustering methods [19], [20] [12]. Datafly [17] and Incognito [18] are single-dimensional full-domain generalization methods. Datafly counts the frequency over the attribute set, and if k-anonymity is not satisfied, it generalizes the attribute having the most distinct values until all the records are included in an equivalence



(a) Ride Start Time Distortion

(b) Difference in Mean Ride Start Time

FIGURE 10. Record Suppression in S_{RST} Scenario; Dataset size, $|T| = 10K$; $k = 5$.

class of size at least k. Datafly does not provide minimal generalization [21]. Incognito constructs a generalization lattice, and traverses it using a bottom-up breadth-first search to find the best solution (each node in the lattice represents a solution). LeFevre *et al.* propose Mondrian [10], a multi-dimensional method that recursively partitions the dataset into equivalence classes, each of which contains at least k records. Mondrian chooses an attribute with the widest range of values to perform partitions on that attribute and then splits/partitions the attribute using the median partitioning approach. Since the median-partitioning approach requires a total order for each attribute, it is more suitable for numerical attributes. In the case of the categorical attribute (which does not have proper ordering), partitioning may not be possible, or it will affect the semantics associated with values.

Methods based on clustering such as density-based clustering [20], *k-member* clustering [12], and attribute hierarchical structure [19] have been proposed to overcome the order issue related to categorical attributes. Clustering-based methods utilize distance metrics (defined for both - numerical and categorical attributes) to measure the similarity between records and form equivalence classes. Zhu and Ye [20] provide k-anonymity based on clustering, where equivalence classes are formed using a density metric, which is measured by the k-nearest neighbors distance. Similarly, Li *et al.* [19] and Byun *et al.* [12] propose k-anonymity based on clustering by defining generalization distances between records. Ni *et al.* [22] propose Grading, Centering, Clustering, and Generalization (GCCG), a clustering-based local generalization algorithm. He *et al.* [23] assumes a global order on all possible values in the domain of each attribute, which may not be a reality for categorical attributes. Most of the existing works are either not suitable to handle all attribute types or do not provide a proper definition to distance metrics for all attribute types. We have extended most of the works by providing a distance metric for all attribute types.

Applying any k-anonymity privacy mechanism leads to a reduction of data utility. Thus, researchers have proposed k-anonymity approaches that minimize the utility loss [24]–[28]. Assuming that each attribute has different utility, Xu *et al.* [24] propose a heuristic local recoding method for utility-based anonymization, where authors assign weights to each attribute to reflect its utility. To achieve minimum utility loss, Ye and Chen [25] propose Attribute Utility Motivated k-anonymization (AUM), where authors classify the attributes as key attributes and an anchor attribute (first attribute to be processed). The algorithm starts by creating regions according to anchor attribute and then perform multi-dimensional recoding and partitioning in each region until the equivalence class size is less than k. The authors aim to satisfy the heterogeneous needs of different users. Unlike in our k-anonymity, their approach does not provide an automated mechanism to determine the utility of attributes which ultimately leads to lower utility loss.

Similarly, Kiyomoto *et al.* [26] generate an anonymized table based on the user's requirement satisfying k-anonymity and l-diversity. The anonymization scheme is based on both top-down and bottom-up approaches for global recoding model - full-domain and partial-domain generalization. Global recoding results in a higher loss of data utility when compared to local recoding model. Another drawback of the paper is that it has assumed tree hierarchy for all the attribute types. This approach is therefore not applicable for many application settings, including the assumed application in our paper.

To minimize information loss and achieve privacy protection, Bhaladhare and Jinwala [28] introduce two approaches that produce sub-databases from the original database, i.e., the generated database contains a smaller number of attributes. However, it attains lower data utility at some level when considering the database with all attributes.

Lin *et al.* [29] and Zhang *et al.* [30] aim to minimize the information loss as well as to reduce run-time required to generate an anonymized transactional dataset. For this, [29] proposes PTA, where a divide-and-conquer method is applied to partition the sorted transactions into several parts based on the Hamming distance. Similarly, [30] proposes the ANonymity

for Transactional database (ANT) algorithm to achieve k-anonymity and l-diversity. The authors use k-means clustering to form clusters of highly similar data by considering their non-sensitive items. Although the k-means clustering is fast, it is challenging to determine the optimal value of k, which may result in high information loss. Rajaei *et al.* [31] introduce a greedy algorithm to generate non-overlapping and anonymized groups of network data. The approach aims to reduce information loss by constructing the groups with a minimum number of members and maximum average desirability (which depends on the used utility and privacy metrics).

To further minimize the loss of data utility, Kohlmayer *et al.* [15] and Orooji and Knapp [16], combine generalization with suppression. Both works experimentally show that combining generalization with suppression significantly increases data utility and reduces disclosure risk. Similar to earlier works, we perform experiments by considering record suppression and see how it affects data utility. The earlier works perform record suppression by applying a limit on the number of records to be suppressed. Instead of limiting the number of records that are suppressed, we approach the problem by introducing a generalization threshold to the attribute itself, showing that this can improve data utility.

Campan *et al.* [14] proposed p-sensitive k-anonymization, which introduces a set of generalization thresholds (i.e., multiple limitations) for each attribute, represented by pre-defined tree hierarchies. In our work, we do not primarily focus on record suppression and impose constraints by introducing a single, manually set limitation for each attribute. Our mechanism can be combined with the approach in [14] to create utility-driven p-sensitive k-anonymization.

To overcome the limitations of earlier works, we automatically assign utility weights to attributes and consider a range of different attribute types. Additionally, we consider the case where all the attributes are quasi-identifiers. These assumptions make our approach more practical for real-world deployment. Though the proposed work focuses on transportation data, we have discussed why the underlying ideas are not limited to that use case and that they can be applied to diverse kinds of datasets. Our approach is flexible and hence can be tuned to meet different application and user needs.

VI. CONCLUSION AND FUTURE WORKS

In this article, we tackled the challenge of utility-driven and privacy-preserving transport data publishing based on the k-anonymity approach. Our mechanism allows the user to assign utility weights to each attribute, allowing them to control which attributes the privacy mechanism preserves over other attributes. To maximize utility, we also introduced an automated mechanism to determine those weights. We investigated the performance of generalization methods with our newly introduced distance metrics and presented a mechanism based on the density-based method using zero-mean unit variance distance normalization. We demonstrated the feasibility and effectiveness of our approach using a real-world public transport dataset. Further, we have shown that, in some cases, it is better to suppress records to preserve utility.

Future work includes the automatic determination of suppression threshold values to further improve the data utility of the anonymized table. Another interesting approach would be to attack the problem from a different angle, that is, instead of letting users define k, they define a target utility (or distortion) value and the privacy mechanism automatically determines k and a suitable suppression threshold.

APPENDIX

TABLE 5. Notations and their definition.

Notation	Definition
k	k-anonymity (privacy) parameter
T	original dataset
T_{ano}	k-anonymized dataset
$r_1, r_2, ..$	records within T
A	set of all attributes
U_a	utility factor of attribute a
U_{max}	maximum utility factor
w_a	utility weight of attribute a
$D^N(x,y)$, $D^T(x,y)$, $D^G(x,y)$	distance between two values (x, y) of numerical attribute N, tree-attribute T, and graph-attribute G
L^T_{xy}	number of leaf nodes at common ancestor of x and y
L^T	total number of leaf nodes in the taxonomy tree of attribute T
$sp(x,y)$	shortest path between x and y
$min(N), max(N)$	minimum and maximum values of numerical attribute N
$\widehat{D_N}(x,y)$, $\widehat{D_T}(x,y)$, $\widehat{D_G}(x,y)$	normalized distance between x and y values of numerical attribute N, tree-attribute T, and graph-attribute G, using maximum distance approach
$\mu(D^N), \sigma(D^N)$, $\mu(D^T), \sigma(D^T)$, $\mu(D^G), \sigma(D^G)$	mean and standard deviation of the pairwise distance of numerical attribute N, tree-attribute T, and graph-attribute G
$\overline{D^N}$, $\overline{D^T}$, $\overline{D^G}$	normalized distance between x and y values of numerical attribute N, tree-attribute T, and graph-attribute G, using zero-mean unit-variance approach
$D(r_1, r_2)$	distance between two records r_1 and r_2
Γ_g	information distortion caused by generalization
Γ_s	information distortion caused by suppression
Γ_{tot}	total information distortion
EC	set of all equivalence classes
Γ^a_g	information distortion caused by generalization in attribute a
Γ^a_s	information distortion caused by suppression in attribute a
Γ_a	information distortion in attribute a
S	number of records suppressed
$\|x\|$	cardinality of x
$\hat{a}^c$	generalized value of attribute a in EC c
$Diam(x)$	graph diameter of the attribute a represented as a graph

REFERENCES

[1] D. Eckhoff and I. Wagner, "Privacy in the smart city—Applications, technologies, challenges, and solutions," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 489–516, 1st Quart., 2018.
[2] I. Wagner and D. Eckhoff, "Technical privacy metrics: A systematic survey," *ACM Comput. Surv.*, vol. 51, no. 3, pp. 57:1–57:38, Jun. 2018.
[3] C. Dwork, "Differential privacy," in *Proc. 33rd Int. Conf. Automata, Lang. Program.*, 2006, pp. 1–12.

[4] L. Sweeney, "Simple demographics often identify people uniquely," Carnegie Mellon Univ., Pittsburgh, PA, USA, Working Paper, 2000, vol. 671, pp. 1–34. [Online]. Available: http://dataprivacylab.org/projects/identifiability/paper1.pdf
[5] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "L-diversity: Privacy beyond K-anonymity," *ACM Trans. Knowl. Discovery Data*, vol. 1, pp. 3:1–3:52, Mar. 2007.
[6] N. Li, T. Li, and S. Venkatasubramanian, "T-closeness: Privacy beyond K-anonymity and l-diversity," in *Proc. IEEE 23rd Int. Conf. Data Eng.*, Istanbul, Turkey, Apr. 2007, pp. 106–115.
[7] X. Xiao and Y. Tao, "M-invariance: Towards privacy preserving re-publication of dynamic datasets," in *Proc. ACM SIGMOD Int. Conf. Manage. Data (SIGMOD)*, Beijing, China, 2007, pp. 689–700.
[8] P. Golle and K. Partridge, "On the anonymity of home/work location Pairs," in *Proc. 7th Int. Conf. Pervasive Comput.*, in Lecture Notes in Computer Science, vol. 5538. Nara, Japan: Springer, May 2009, pp. 390–397.
[9] J. Krumm, "Inference attacks on location tracks," in *Proc. 5th Int. Conf. Pervasive Comput. (PERVASIVE)*, in Lecture Notes in Computer Science, vol. 4480. Toronto, ON, Canada: Springer, May 2007, pp. 127–143.
[10] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan, "Mondrian multidimensional K-Anonymity," in *Proc. 22nd Int. Conf. Data Eng. (ICDE)*, 2006, p. 25.
[11] A. Scott, V. Srinivasan, and U. Stege, "K-attribute-anonymity is hard even for K=2," *Inf. Process. Lett.*, vol. 115, no. 2, pp. 368–370, 2015.
[12] J.-W. Byun, A. Kamra, E. Bertino, and N. Li, "Efficient K-anonymization using clustering techniques," in *Advances in Databases: Concepts, Systems and Applications*. Berlin, Germany: Springer, 2007, pp. 188–200.
[13] Q. Xu, "Measuring information content from observations for data assimilation: Relative entropy versus Shannon entropy difference," *Tellus A, Dyn. Meteorol. Oceanogr.*, vol. 59, no. 2, pp. 198–209, Jan. 2007.
[14] A. Campan, T. M. Truta, and N. Cooper, "P-sensitive K-anonymity with generalization constraints," *Trans. Data Privacy*, vol. 3, pp. 65–89, Aug. 2010.
[15] F. Kohlmayer, F. Prasser, and K. A. Kuhn, "The cost of quality: Implementing generalization and suppression for anonymizing biomedical data with minimal information loss," *J. Biomed. Informat.*, vol. 58, pp. 37–48, Dec. 2015.
[16] M. Orooji and G. M. Knapp, "Improving suppression to reduce disclosure risk and enhance data utility," *CoRR*, vol. abs/1901.00716, p. arXiv:1901.00716, Jan. 2019. [Online]. Available: http://arxiv.org/abs/1901.00716
[17] L. Sweeney, "Guaranteeing anonymity when sharing medical data, the datafly system," in *Proc. Conf. Amer. Med. Inform. Assoc. AMIA Fall Symp.*, 1997, pp. 51–55.
[18] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan, "Incognito: Efficient full-domain K-anonymity," in *Proc. ACM SIGMOD Int. Conf. Manage. Data (SIGMOD)*, Baltimore, MD, USA, 2005, pp. 49–60.
[19] J. Li, R. C.-W. Wong, A. W.-C. Fu, and J. Pei, "Achieving K-anonymity by clustering in attribute hierarchical structures," in *Data Warehousing and Knowledge Discovery*. Berlin, Germany: Springer, 2006, pp. 405–416.
[20] H. Zhu and X. Ye, "Achieving K-anonymity via a density-based clustering method," in *Advances in Data and Web Management*. Berlin, Germany: Springer, 2007, pp. 745–752.
[21] P. Samarati, "Protecting respondents identities in microdata release," *IEEE Trans. Knowl. Data Eng.*, vol. 13, no. 6, pp. 1010–1027, Nov./Dec. 2001.
[22] S. Ni, M. Xie, and Q. Qian, "Clustering based K-anonymity algorithm for privacy preservation," *Int. J. Netw. Secur.*, vol. 19, no. 6, pp. 1062–1071, 2017.
[23] X. He, H. Chen, Y. Chen, Y. Dong, P. Wang, and Z. Huang, "Clustering-based K-anonymity," in *Advances in Knowledge Discovery and Data Mining*, P.-N. Tan, S. Chawla, C. K. Ho, and J. Bailey, Eds. Berlin, Germany: Springer, 2012, pp. 405–417.
[24] J. Xu, W. Wang, J. Pei, X. Wang, B. Shi, and A. W.-C. Fu, "Utility-based anonymization using local recoding," in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2006, pp. 785–790.
[25] H. Ye and E. Chen, "Attribute utility motivated K-anonymization of datasets to support the heterogeneous needs of biomedical researchers," in *Proc. AMIA Annu. Symp.*, 2011, pp. 1573–1582.
[26] S. Kiyomoto, Y. Miyake, and T. Tanaka, "Privacy frost: A user-oriented data anonymization tool," in *Proc. 6th Int. Conf. Availability, Rel. Secur.*, Washington, DC, USA, Aug. 2011, pp. 442–447.
[27] K. Doka, M. Xue, D. Tsoumakos, and P. Karras, "K-anonymization by freeform generalization," in *Proc. 10th ACM Symp. Inf., Comput. Commun. Secur.*, New York, NY, USA, Apr. 2015, pp. 519–530.
[28] P. R. Bhaladhare and D. C. Jinwala, "Novel approaches for privacy preserving data mining in K-anonymity model," *J. Inf. Sci. Eng.*, vol. 32, no. 1, pp. 63–78, 2016.
[29] J. C.-W. Lin, Q. Liu, P. Fournier-Viger, and T.-P. Hong, "PTA: An efficient system for transaction database anonymization," *IEEE Access*, vol. 4, pp. 6467–6479, 2016.
[30] B. Zhang, J. C.-W. Lin, Q. Liu, P. Fournier-Viger, and Y. Djenouri, "A (k, p)-anonymity framework to sanitize transactional database with personalized sensitivity," *J. Internet Technol.*, vol. 20, no. 3, pp. 801–808, 2019.
[31] M. Rajaei, M. Haghjoo, and E. K. Miyaneh, "An anonymization algorithm for $(\alpha, \beta, \gamma, \delta)$-social network privacy considering data utility," *J. Univers. Comput. Sci.*, vol. 21, no. 2, pp. 268–305, 2015.

BHAWANI SHANKER BHATI received the Ph.D. degree in engineering from the Indian Institute of Science (IISc), Bengaluru, India, in 2018. He is currently a Research Fellow with TUMCREATE, Singapore. His research interests include ad-hoc networks, communication protocols, ubiquitous computing, security, and privacy in wireless networks.

JORDAN IVANCHEV received the Ph.D. degree in computer science from the Technical University of Munich (TUM), in 2017. He is currently a Senior Research Fellow with TUMCREATE. His research interests include mixed traffic modeling and simulation, automated simulation calibration, social optimal routing strategies, and traffic sensing and control.

IVA BOJIC received the Ph.D. degree in computer science from the University of Zagreb, Croatia. She continued her postdoctoral research at the MIT Senseable City Laboratory, where she joined as a Fulbright Scholar in 2014. After one year at MIT, she moved to Singapore, where she is currently a Research Scientist working with the Singapore-MIT Alliance for Research and Technology. Her research interests include big data analysis, machine learning, data science, and transportation systems.

ANWITAMAN DATTA is currently an Associate Professor with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. He also serves as the Senior Scientific Officer in a consulting role with QPQ.IO. His core research interests include large-scale resilient distributed systems, information security, and applications of data analytics. He is also exploring topics at the intersection of computer science, public policies, and regulations along with the wider societal and (cyber) security impact of technology. This includes the topics of social media and network analysis, privacy, cyber risk analysis and management, cryptocurrency forensics, the governance of disruptive technologies, as well as impact and use of disruptive technologies in digital societies and government.

DAVID ECKHOFF (Member, IEEE) received the Ph.D. degree (Hons.) in engineering in 2016. He was a Visiting Scholar with the Group of Prof. Lars Kulik, The University of Melbourne, Australia, in 2016. He joined the Group of Prof. Alois Knoll, TUMCREATE, Singapore, in October 2016. He is currently the Principal Investigator of TUMCREATE, a joint research Institute by TU Munich and Nanyang Technical University, Singapore. His research interests include privacy protection, smart cities, vehicular networks, and intelligent transportation systems with a particular focus on modeling and simulation.