Highlights

Enhancing Efficiency and Data Utility in Longitudinal Data Anonymization

Fatemeh Amiri[1], David Sánchez, Josep Domingo-Ferrer

- We propose the $(k, \beta)^L$-privacy model to anonymize longitudinal data.
- The model assumes the background knowledge is a subsequence of publicly known L QI values.
- The model restricts the adversary's confidence on inferences on the victim to a predetermined threshold.
- We propose an anonymization algorithm to enforce the model via generalization and suppression.
- Our method outperforms existing methods in terms of data utility and stringent privacy constraints.

[1]Corresponding author, Hamedan University of Technology, Shahid Fahmideh.St, Hamedan, IRAN, f.amiri@hut.ac.ir

Enhancing Efficiency and Data Utility in Longitudinal Data Anonymization

Fatemeh Amiri[1a], David Sánchez[b], Josep Domingo-Ferrer[b]

[a] *Department of Computer Engineering and Information Technology, Hamedan University of Technology, Hamedan, Iran*
[b] *Universitat Rovira i Virgili, Department of Computer Engineering and Mathematics, CYBERCAT Center for Cybersecurity Research of Catalonia, Av. Paisos Catalans 26, Tarragona, E-43007, Catalonia*

Abstract

Longitudinal data are data collected over time on a set of individuals. The fact that information items are accumulated on each individual over time makes longitudinal data particularly privacy-sensitive. Existing anonymization methods are often ill-suited to ensure privacy-preserving publishing of this kind of data, and current privacy models assume unrealistic attacker knowledge. First, we propose the $(k, \beta)^L$-privacy model, which assumes the attacker's knowledge to be a subsequence of L quasi-identifiers, and thus provides a more realistic representation of the information actually available to attackers. This also contributes to enhancing the utility of the protected data. Second, we introduce FCLA, an anonymization algorithm to enforce our privacy model that prioritizes data utility while effectively mitigating identity and attribute disclosures, as well as skewness attacks in longitudinal data. FCLA partitions sequences into groups and anonymizes them independently, a task that is straightforward to parallelize. Reported experiments demonstrate that FCLA outperforms existing methods at preserving data utility while satisfying strict privacy constraints. The time complexity analysis and execution time of FCLA demonstrate that FLCA is more efficient, leading to better scalability.

Keywords: longitudinal data publishing, sequence data publishing, data anonymization, data privacy

[1]Corresponding author, Hamedan University of Technology, Shahid Fahmideh.St, Hamedan, IRAN, f.amiri@hut.ac.ir

1. Introduction

Longitudinal data refer to data continuously collected from the same individuals over time. This type of data is valuable for understanding changes over time or relationships between variables. Anonymizing longitudinal data prior to their release is crucial to safeguard the privacy and confidentiality of the individuals the data refer to and to comply with privacy regulations and laws. However, the absence of anonymization methods in certain longitudinal surveys, such as the Scottish Longitudinal Survey [23], the Youth Survey Luxembourg [25], the Young People and Covid-19 (YAC) [20] surveys, the Framingham Heart Study (FHS) [9], the Child Asthma Management Program (CAMP)[26], and the HIVNET Informed Consent Substudy [4], highlights how challenging and pressing privacy-preserving longitudinal data publishing is.

Table 1 illustrates a sample data set that emulates longitudinal health information, reminiscent of electronic health records (EHRs) and electronic medical records (EMRs). The data set's longitudinal nature emerges from the possibility of a patient's undergoing multiple admissions in a certain lapse of time. These admissions are interconnected and pertain to the same individual. Within this data set, each patient may possess multiple records corresponding to various visits. These visits are meticulously ordered based on their respective visit dates, with each visit being assigned a unique visit ID. The data set comprises six quasi-identifying (QI) attributes, which include PID (patient ID), VID (visit ID), Y (admission year), Z (Zip code), D (number of days since the first claim in each year), and L (length of stay in the hospital). Additionally, there is one sensitive attribute (SA) - Disease. To provide an example, consider Table 1, where patient PID=2 experienced a ten-day hospitalization in 2018 and subsequently had another visit 30 days later, in zip code 43003.

Privacy breaches can occur through individual visits or through multiple visits over time. Adversaries can launch two types of attacks if these data are released: *identity disclosure* and *attribute disclosure*. In identity disclosure, adversaries aim to re-identify individuals from the anonymized data. For example, if an adversary knows that Alice had a visit in 2017 and resides in zip code 42003 since 2018, they can identify Alice's records (those with PID= 4) and, as a consequence, learn all diseases suffered by Alice. Attribute

Table 1: Inpatient Longitudinal Data.

PID	**VID**	**Y**	**Z**	**D**	**L**	**Disease**
1	1	2018	41002	0	21	Infection
2	1	2018	41001	0	10	Cancer
2	2	2018	43003	30	1	Flu
3	1	2018	40012	0	30	Fever
3	2	2019	40012	0	35	Infection
3	3	2020	41001	0	3	Flu
4	1	2017	41001	0	10	Heart attack
4	2	2018	42003	0	10	Flu
4	3	2021	42003	0	35	Hepatitis
5	1	2018	42005	0	3	Fever
5	2	2018	42005	80	10	Cancer
6	1	2017	41001	0	5	Infection
6	2	2019	43002	0	30	Hepatitis
7	1	2018	42016	0	4	Fever
8	1	2020	43003	0	5	Fever
9	1	2019	42016	0	1	Flu
10	1	2018	42016	0	14	Hear attack

disclosure, on the other hand, aims to infer sensitive information on the subjects, even without re-identifying their records. For instance, if Bob had two visits in 2018, the attacker may infer his condition as cancer, even without pinpointing Bob's record.

The k-anonymity model [22] is commonly used to safeguard relational data from privacy breaches by preventing identity disclosure. It guarantees that any specific individual within a data set cannot be distinguished from at least $k-1$ other individuals in the same data set. However, it falls short of protecting against attribute disclosure. To overcome this limitation, extensions of k-anonymity, such as β-likeness [3], t-closeness [14], and l-diversity [18] have been proposed. However, these models are not suitable for longitudinal data due to the latter's high dimensionality and sequential nature. Achieving protection against re-identification and attribute disclosure for longitudinal data containing multiple instances of each attribute at different time points is not easy. Some attempts to address these challenges, such as the methods proposed in [7], [24], [27], and [31] are vulnerable to attribute disclosure attacks. Existing anonymization methods for longitudinal data often fail to produce data sets suitable for emerging clinical research. Effective anonymization requires considering temporal correlations and preventing privacy breaches through any combination of QI values within and across events. ElEmam *et al.* [7] and Tamersoy *et al.* [27] focus on privacy protection against identity disclosure, which may result in sensitive values being overlooked. In contrast, Sehatkar and Matwin [24] propose a partitioning method based on the presence of highly sensitive items in a specified threshold percentage of records. However, it is important to note that these approaches may not offer adequate protection for infrequent sensitive values that are typically encountered in longitudinal data. Additionally, privacy models such as k-anonymity and its extensions assume that attackers have perfect knowledge of all quasi-identifying attributes (QIs) to perform linkage attacks. Extrapolating this to longitudinal data implies assuming attackers with exhaustive knowledge of all the events related to the individual. This is hardly realistic and would require significant data distortion to prevent such perfect knowledge attacks. Some studies, such as those by [30] and [33], explore the concept of limiting the background knowledge available to attackers.

1.1. Contributions and plan

We introduce the $(k, \beta)^L$-privacy model, which limits the adversary's background knowledge to subsequences of publicly known L quasi-identifier (QI) values. This model guarantees that each subsequence of L QI values appears in either zero or at least k records within the longitudinal database. Furthermore, it ensures that confidences for any sensitive value within these k records are at most β times larger than those in the entire data set. Our privacy model offers protection against identity disclosure, attribute disclosure, and attacks based on skewness and semantics [14].

It is unlikely for an adversary to possess complete knowledge of all QI attribute values for every event within an individual's record. Our scenario, which assumes an adversary knowing a subsequence of events, is more realistic than models that assume perfectly knowledgeable attackers. For example, in medical longitudinal data, it is more plausible for an adversary to know specific events about the target individual (such as a visit in 2018 or a two-day hospitalization in 2019) than assuming an adversary with access to the target's entire medical history.

Our proposed model further enhances privacy by allowing for the definition of subsets of sensitive information requiring stricter protection. By applying our privacy model to sequences containing values from these subsets, we can achieve a balance between enhanced privacy and data utility.

We propose an anonymization algorithm that enforces our privacy model using value generalization and suppression techniques. When releasing medical data for general purposes without a predefined use case, it is crucial to ensure that each record in the published data accurately represents a real individual. For example, researchers might need genuine patient records to investigate the side effects of a specific drug. Unlike methods such as swapping, randomization, or synthetic data generation, generalization and suppression preserve the truthfulness of records, even if they reduce the level of detail. This ensures that the released data retain their meaningfulness at the record level and prevents the introduction of fabricated information during anonymization. It is important to note that methods that perturb or synthesize original values can introduce semantic alterations making the anonymized data unsuitable for certain analyses [10].

The proposed algorithm specifically addresses the sequential nature of longitudinal data, operating on multidimensional sequences of events. To preserve data utility, the algorithm partitions the sequence data into clusters based on the similarity of QI values, ensuring each cluster contains at

least k sequences. This clustering approach minimizes data distortion while maintaining data quality and satisfying privacy requirements. We employ a dynamic programming approach, derived from the sequence alignment problem [27], to measure the distance between sequences. Within each cluster, anonymization techniques are applied independently, which ensures that every subsequence of no more than L QI values satisfies our privacy model. Parallel processing of these groups further enhances efficiency. To identify privacy-violating sequences within each cluster, we utilize sequential pattern mining techniques [8].

We ran experiments on three data sets —CMS 2008-2010 DE-SynPUF Data[2], Heritage Health Prize (HHP) claims data set[3] and FDA Adverse Event Reporting System (FAERS) data set[4]— to show the effectiveness of our methods. By evaluating various parameters of our privacy model, we compared the performance of our algorithms with the most similar related work, *i.e.*, the HALT algorithm [24]. Results indicate that our approach outperforms HALT in terms of data utility and privacy while achieving lower time complexity.

The rest of this paper is organized as follows. Section 2 discusses related works. Section 3 defines the problem. Section 4 presents our proposed privacy model. Section 5 introduces our anonymization algorithm designed to satisfy the privacy model. Section 6 provides details about the implementation results. Finally, Section 7 offers conclusions while also outlining potential future work.

2. Related work

Here we review research on anonymization of longitudinal data and related data types, such as transaction data, trajectory data, and sequential data. We will examine the main techniques employed with a view to enhancing their privacy level and their suitability for pure longitudinal data. Given that some of the models employed on the aforementioned data types draw inspiration from relational data models, we begin by examining research on

[2] `https://www.cms.gov/research-statistics-data-and-systems/downloadable-public-use-files/synpufs/de_syn_puf`

[3] `https://foreverdata.org/1015/index.html`

[4] `https://fis.fda.gov/extensions/FPD-QDE-FAERS/FPD-QDE-FAERS.html`

relational data. It is worth noting that relational data models are known to face certain issues, which we also address.

Relational data, typically presented in tabular form, share a fixed set of attributes among records. Various models have been introduced to protect privacy in such data [3, 14, 18, 22]. The k-anonymity model [22] guarantees that each record remains indistinguishable from at least $k-1$ others in terms of its QI values. However, k-anonymity only focuses on QIs (to protect against re-identification), but it neglects confidential attributes (that can be used in attribute inference attacks). In contrast, the l-diversity model [18] mandates that each equivalence class contain at least l well-represented values of the confidential attribute. This model implicitly assumes a uniform distribution of the sensitive attribute values, which limits its effectiveness on skewed data or multiple data releases. The t-closeness model [14] strives to minimize disparities up to a threshold t between the frequency distribution of sensitive attribute values within an equivalence class and the overall microdata table. Finally, the β-likeness model [3] mitigates an adversary's confidence gain through a relative difference measure; it is important to note that this model does not provide protection against identity disclosure.

Transactional data are unstructured and highly-dimensional, with sparsity and no inherent sequential nature. Anonymizing such data without relying on sequentiality has been tackled by studies such as [11] and [28]. However, just applying transaction anonymization methods to longitudinal data does not assure privacy protection, because attackers can exploit the sequential nature of data. Protecting longitudinal data requires specialized techniques to address their unique privacy challenges.

Trajectory data consist in a collection of sequences of spatio-temporal data points belonging to a moving object such as a GPS device, a cell phone, or an RFID tag. Various approaches have been explored for anonymizing this type of data, as discussed in the recent survey [12]. Some works propose protecting trajectory data based on the l-diversity requirement [17, 35]. This type of data can be regarded as a subset of longitudinal data focused on movement. Anonymization methods for trajectory data aim to generalize or suppress the location information [29]. Nonetheless, when these methods are applied to longitudinal data with multiple attributes at each time point, significant information loss can occur. Additionally, the presence of multiple attributes in each event of longitudinal data may affect the accuracy of the distance measure typically used in trajectory data, thereby leading to potential inaccuracies in calculating distances between events.

Longitudinal data refer to multidimensional sequences of events [21]. The anonymization methods proposed by ElEmam *et al.* [7] and Tamersoy *et al.* [27] enforce the k-anonymity privacy model to modify sequences and create indistinguishable groups of k events. Yet these techniques may be vulnerable to attribute disclosure attacks. These methods also fall short of adequately modeling the adversary's background knowledge. In longitudinal data, such as those studied in [27], records contain sequences of (ICD, Age) pairs and DNA sequences. Existing approaches model the adversary's knowledge as combinations of (ICD, Age) pairs, thus failing to account for the multi-dimensionality of events in longitudinal data, which can include multiple quasi-identifiers within each event.

To apply the method in [7] to our problem, we must restrict each event to two quasi-identifiers and assume that the adversary always knows both values during a visit. Sehatkar and Matwin [24] introduced the $(k, C)^L$-privacy model to address these challenges. It combines k-anonymity with restricting the percentage of records containing sensitive values to a maximum of C. However, their proposed method, HALT, examines all patterns with at most L QI values across the data set, without considering QI value similarity, thereby incurring drawbacks such as high cost, information loss, and vulnerability to skewness attacks. Wang et al. [31] presented a privacy model called $MS(k, \theta^*)$-bounding for safeguarding published spontaneous reporting data against privacy attacks. Their model assumes individuals may have multiple events, each with multiple sensitive values. However, their algorithm does not account for the temporal features of longitudinal data. Their anonymization generalizes all events for each individual to create a "super record" (so that each individual has a single super record), and then clusters these super records. This approach can lead to increased information loss compared to methods like [24], which anonymize individual events while considering temporal features.

Parameshwarappa *et al.* [19] proposed novel multi-level clustering-based methods for enforcing the l-diversity model to protect against attribute disclosure. However, their approach assumes fixed-length sequences, whereas we aim to protect longitudinal data with variable-length sequences for each individual.

In summary, while k-anonymity overlooks sensitive attributes –thereby leading to potential attribute disclosure risks–, $(k, C)^L$-privacy often fails to consider less frequent sensitive attribute values –which are prevalent in this data type–. Consequently, neither model provides sufficient privacy protec-

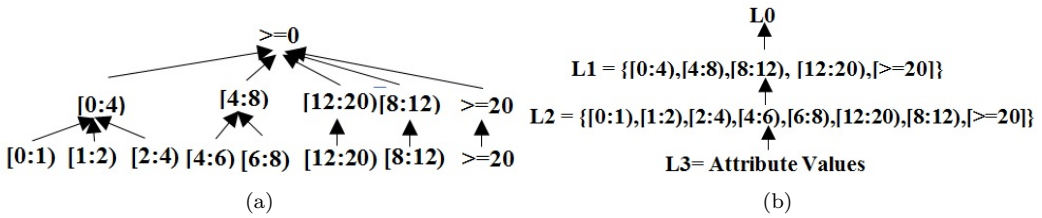


Figure 1: Generalization hierarchies of L and D in terms of weeks: (a) value generalization VH_L, VH_D; (b) domain generalization DH_L, DH_D.

tion against skewness attacks. Such attacks occur when the probability of linking an individual to a sensitive value s_i differs significantly within a set of sequences compared to the overall probability of linking to the set of sensitive values. For example, let us consider $(10, 0.1)$-privacy [24] for a health longitudinal data set, T, where only 0.1% of individuals are infected with HIV. Let Q be a set of sequences in T that contains 10 different sequences, with one occurrence of HIV. In Q, the probability of linking to HIV is 10%, whereas in T it is only 0.1%. This 100-fold increase in probability represents a significant and undesirable information leakage.

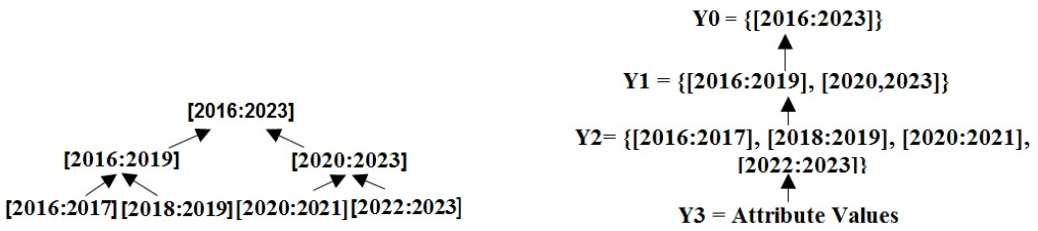
Our work aims to overcome these limitations by proposing an intuitive and distribution-based privacy model. We also introduce a specialized anonymization algorithm designed to implement this novel privacy model.

3. Problem Definition

We model data as consisting of quasi-identifying attributes denoted by $A = \{A_1, A_2, ..., A_n\}$ with their respective attribute domains represented as $R = \{R_1, R_2, ...,R_n\}$. Each QI attribute is associated with a domain generalization hierarchy DH_{A_i} and a value generalization hierarchy VH_{A_i}. A domain generalization is defined to be a set of domains that is totally ordered by direct generalization. Value generalization is a value-level tree, in which edges are direct value generalizations. Additionally, there is a sensitive attribute (SA) with domain values $R_{SA} = \{s_1, s_2, ..., s_m\}$.

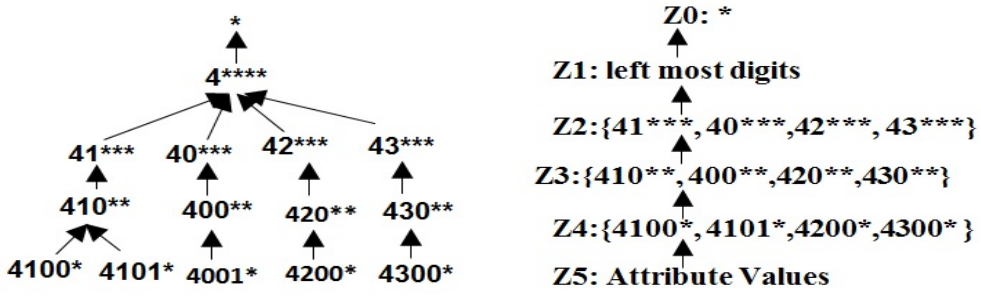
As an illustration, Figures 1 to 3 visually depict the generalization hierarchies for the QIs listed in Table 1. The L and D attributes are generalized in terms of weeks.

The multidimensional sequence data set $T = \{T^1, T^2, ..., T^t\}$ consists of records of the form $T^i = (SID, SE)$, where each record represents a sequence of events $SE = \langle e_1, e_2, ...\rangle$. The events are ordered chronologically based on



(a) (b)

Figure 2: Generalization hierarchy of Y: (a) value generalization VH_Y; (b) domain generalization DH_Y.



(a) (b)

Figure 3: Generalization hierarchies of Z: (a) value generalization VH_Z; (b) domain generalization DH_Z.

the temporal relation $<_t$, such that $e_1 <_t e_2 <_t ... <_t e_r$. Each event e_i contains values for n QIs and one SA, denoted as $(a_1, a_2, ..., a_n, s_j)$, where $a_i \in R_i$ and $s_j \in R_{SA}$. We also assume that attackers may have values for subsets of QIs, but the specific subset may vary for different attackers. The sensitive attribute values can be assumed private (if the attacker had them for a certain configuration of QIs, no attack would be needed).

More specifically, each event in a sequence can be represented as a collection of $(A_i, value)$ pairs, called items. The *value* corresponds to a node in VH_{A_i}. For example, to convert the longitudinal data in Table 1 into a multidimensional sequential data set, the visits of each patient are combined, sorted based on visit dates, and presented as sets of n attributes. Table 2 serves as an illustrative reference for analysis after this transformation. To compare sequences, we introduce the following concepts. For two items $x = (A_i, v)$ and $y = (A_i, v')$, we define y as being more specific than x (denoted as $y \leq_I x$) if either $v = v'$ or v is an ancestor of v' in VH_{A_i}. For example $(L, [0:1)) \leq_I (L, [0:4))$.

Let $e = \{a_1, ..., a_n\}$ and $e' = \{a'_1, ..., a'_n\}$ represent two given events. We say that e is more specific than e' (denoted as $e \leq_E e'$) if, for all $1 < i < n$, $a_i \leq_I a'_i$. For instance, $\{(Y, 2016), (Z, 4100*)\}$ is considered more specific than $\{(Y, [2016:2019]), (Z, 4100*)\}$. Consider two sequences $X = \langle x_1, ..., x_l \rangle$ and $X' = \langle x'_1, ..., x'_{l'} \rangle$. We say that X is more specific than X' (represented as $X \leq_S X'$) if there exists a sequence of indices $1 \leq i_1 < i_2 < ... < i_{l'} \leq l$ where $x_j \leq_E x'_{i_j}$ holds for all $j = 1, 2, ..., l'$, with $l' \leq l$. For example, $\langle \{(Y, 2018), (L, 2)\}, \{(Y, 2019), (Z, 43003)\} \rangle \leq_S \langle \{(Y, [2016:2023]), (Z, 4300*)\} \rangle$.

We also define the length of a sequence, denoted by $len(X)$, as the total number of QI items across all events in X. The items are numbered if their values are not the root of the value generalization hierarchy tree. For example, in the sequence $\langle \{(Y, [2016:2023]), (L, 2)\}, \{(Y, 2019), (Z, 43003)\} \rangle$, there are two events, and $len(X) = 3$.

Definition 1. *The support of a sequence X, denoted as $sup_s(X)$, is the set of sequences in T that are at least as specific as X ($sup_s(X) = \{S^i \in T \mid S^i \leq_S X$), and $|sup_s(X)|$ represents the count of sequences in $sup_s(X)$.*

Definition 2. *The support of an event e', denoted as $sup_e(e')$, is the set of sequences in T that contain an event at least as specific as e'.*

Table 2: Multidimensional sequential data

PID	**Sequence of visits**
1	$\langle\{(Y,2018),(Z,41002),(D,0),(L,21),(Disease,Infection)\}\rangle$
2	$\langle\{(Y,2018),(Z,41001),(D,0),(L,10),(Disease,Cancer)\},$ $\{(Y,2018),(Z,43003),(D,30),(L,1),(Disease,Flu)\}\rangle$
3	$\langle\{(Y,2018),(Z,40012),(D,0),(L,30),(Disease,Fever)\},$ $\{(Y,2019),(Z,40012),(D,0),(L,35),(Disease,Infection)\}$ $\{(Y,2020),(Z,41001),(D,0),(L,3),(Disease,flu)\}\rangle$
4	$\langle\{(Y,2017),(Z,41001),(D,0),(L,10),(Disease,Heartattack)\},$ $\{(Y,2018),(Z,42003),(D,0),(L,10),((Disease,Flu)\},$ $\{(Y,2021),(Z,42003),(D,0),(L,35),(Disease,Hepatitis)\}\rangle$
5	$\langle\{(Y,2018),(Z,42005),(D,0),(L,3),(Disease,Fever)\},$ $\{(Y,2018),(Z,42005),(D,80),(L,10),((Disease,Cancer)\}\rangle$
6	$\langle\{(Y,2017),(Z,41001),(D,0),(L,5),(Disease,Infection)\},$ $\{(Y,2019),(Z,43002),(D,0),(L,30),(Disease,Hepatitis)\}\rangle$
7	$\langle\{(Y,2018),(Z,42016),(D,0),(L,4),(Disease,Fever)\}\rangle$
8	$\langle\{(Y,2020),(Z,43003),(D,0),(L,5),(Disease,Fever)\}\rangle$
9	$\langle\{(Y,2019),(Z,42016),(D,0),(L,1),(Disease,Flu)\}\rangle$
10	$\langle\{(Y,2018),(Z,42016),(D,0),(L,14),(Disease,Heartattack)\}\rangle$

4. Privacy Model

We next present our privacy model, which assumes that the distribution of sensitive attribute (SA) values in a longitudinal data set T is publicly known. Furthermore, we assume that the attacker possesses a certain background knowledge regarding the longitudinal events of victim. The attacker's background knowledge about the sequence of events for the victim includes values of QIs as well as the order of these values. By leveraging publicly available hierarchies for QIs, the attacker can express her knowledge at varying levels of granularity. In real-life scenarios, it is in general not feasible for an attacker to have knowledge of all the QI values for every event. Therefore, it is more realistic to assume that the attacker has partial knowledge, which is limited to a maximum of P values of QIs. This bounded partial knowledge assumption aligns with the constraints and limitations typically encountered in practical situations.

As an example, consider the case where the attacker knows that the victim had a visit between 2018 and 2019 and the victim was admitted to the hospital for 3 days in 2020. The attacker is aware of the hierarchy of the attributes shown in Figures 1 to 3. In this scenario, she knows 3 QI values where two of them occur together in one visit. Such knowledge can be modeled as $X = \langle\{[Y, 2018 : 2019]\}, \{[Y, 2020], [L, 3]\}\rangle$. Equipped with this background knowledge, the attacker can initiate a privacy breach by identifying a matching sequence in the longitudinal data set that aligns with her background knowledge X. A sequence X' is a match to X if it contains X, *i.e.*, $X' \leq_S X$. When examining the data in Table 2, the record with PID # 3 corresponds to sequence X.

To define our proposed privacy model, we compute the confidence of a sensitive value as follows.

Definition 3. *The confidence of a sensitive attribute value s_i in a set of sequences Q reflects the probability of inferring s_i for an individual in Q. It is denoted by $conf\,(s_i|Q) = \frac{|sup_s(Q^{s_i}))|}{|sup_s(Q)|}$, where Q^{s_i} represents the subset of sequences in Q containing s_i. In fact, $conf\,(s_i|Q)$ represents the proportion of sequences in $sup_s(Q)$ that contain s_i.*

When the victim's record is in Q, the gain in confidence for the sensitive attribute value s_i is computed as $q_i - p_i$, where q_i represents the confidence of s_i in Q and p_i represents the confidence of s_i in the sequence data set T.

A positive gain indicates increased correlation between the victim's record and s_i within Q, while a negative gain means reduced correlation, thereby enhancing privacy[5]. Therefore, our approach primarily focuses on positive confidence gains. To mitigate the risk of identity and attribute disclosure, we enforce two conditions in each set of sequences via our privacy model $(k, \beta)^L$-privacy, as follows.

Definition 4 ($(k, \beta)^L$-privacy). *Let $P_c = \{p_1, \ldots, p_m\}$ represent the confidences for all sensitive attribute (SA) values in a longitudinal data set T. We define L as the maximum length of the adversary's background knowledge. Moreover, let Q be a set of sequences with SA confidence $Q_c = \{q_1, \ldots, q_m\}$. The relative distance between the confidences of SA values is determined by $D(p_i, q_i) = \frac{q_i - p_i}{p_i}$. A set of sequences Q satisfies $(k, \beta)^L$-privacy if and only if the following two conditions are met:*

1. *To protect against identity disclosure, it must hold that $|sup_s(Q)| \geq k$.*
2. *To protect against attribute disclosure, it must hold that the maximum relative distance between confidences of SA values is less than or equal to $\min\{\beta, -\ln p_i\}$,* i.e. $\max\{D(p_i, q_i) | p_i \in P_c, p_i < q_i\} \leq \min\{\beta, -\ln p_i\}$, *where β is a positive value.*

By making sure that each individual in the data set cannot be distinguished from at least $k - 1$ other individuals based on QI, Condition 1 in Definition 4 effectively mitigates the risk of identity disclosure.

To preempt both attribute disclosure and skewness attacks, Condition 2 mandates that the difference in confidence of SA values within any group of sequences should not exceed a predefined threshold compared to the overall data set's confidence values. Note that using the absolute difference as a metric would fall short of treating all SA values and their relative differences equitably. Consequently, we have chosen to adopt a relative difference distance approach. Additionally, we propose the use of a maximum threshold, rather than relying on a cumulative distance threshold. Our model is

[5]However, it is important to note that reducing the gain for certain sensitive attribute values may inadvertently result in privacy violations. For example, if the victim's gain for the majority values of religion or sexual orientation is reduced, it means she is more likely to belong to a religion or sexual orientation minority. To address this issue, a potential solution, proposed in [3], is to transform the relative negative gain of a sensitive attribute value into positive gains for other sensitive attribute values.

intentionally designed to accommodate SA values that may be absent from specific groups, hence granting more flexibility in the anonymization process and ultimately preserving higher information quality.

We next prove the monotonicity of both Conditions 1 and 2.

Lemma 1 (Monotonicity). *Assume there are two disjoint groups of sequences Q^1 and Q^2 generated from sequences in the overall data set T. Let SA value $s_i \in S$ have confidence p_i in T, q_i^1 in Q^1, and q_i^2 in Q^2. Additionally, let q_i^3 be the confidence in $Q^3 = Q^1 \cup Q^2$. Both conditions of Definition 4 satisfy monotonicity.*

Proof: The monotonicity property in Condition 1 requires $|sup_s(Q^3)| \geq \max(|sup_s(Q^1)|, |sup_s(Q^2)|)$. For all sequences S and S^0, if $S^0 \leq_S S$, then $|sup(S^0)| \leq |sup(S)|$ [13]. Therefore, $|sup_s(Q^1)| \leq |sup_s(Q^3)|$ and $|sup_s(Q^2)| \leq |sup_s(Q^3)|$ because $Q^1 \leq_S Q^3$ and $Q^2 \leq_S Q^3$. Thus,

$$|sup_s(Q^3)| \geq \max(|sup_s(Q^1)|, |sup_s(Q^2)|).$$

To prove the monotonicity of Condition 2, we need to demonstrate that $D(p_i, q_i^3) = \max\{D(p_i, q_i^1), D(p_i, q_i^2)\}$. Assume there are $n^1 = |sup_s(Q^1 \cap s_i)|$ and $n^2 = |sup_s(Q^2 \cap s_i)|$ sequences with the sensitive value s_i in Q^1 and Q^2, respectively. Then, we have $q_i^1 = \frac{n^1}{|sup_s(Q^1)|}$, $q_i^2 = \frac{n^2}{|sup_s(Q^2)|}$, and $q_i^3 = \frac{n^1+n^2}{|sup_s(Q^1)|+|sup_s(Q^2)|} = \frac{q_i^1 \cdot |sup_s(Q^1)| + q_i^2 \cdot |sup_s(Q^2)|}{|sup_s(Q^1)|+|sup_s(Q^2)|} \leq \max\{q_i^1, q_i^2\}$. Thus, $D(p_i, q_i^3) \geq \max\{D(p_i, q_i^1), D(p_i, q_i^2)\}$. □

Condition 2 limits the confidence level of sensitive values in set Q by requiring $q_i \leq (1 + \min\{\beta, -\ln p_i\}) \cdot p_i$. This constraint can be further understood by decomposing the upper bound defined by $\delta(p_i) = (1 + \min\{\beta, -\ln p_i\}) \cdot p_i$:

- For $0 < p_i \leq e^{-\beta}$, $\delta(p_i) = (1 + \beta) \cdot p_i$.
- For $e^{-\beta} \leq p_i \leq 1$, $\delta(p_i) = (1 - \ln p_i) \cdot p_i$.

The two segments intersect at $p_i = e^{-\beta}$. The function $\delta(p_i)$ monotonically increases within $(0, 1]$, ensuring confidence levels below 1, accounting for value frequency. For rare values ($0 < p_i \leq e^{-\beta}$), confidence is bounded by $(1 + \beta)p_i$. For frequent values ($e^{-\beta} \leq p_i \leq 1$), confidence is restricted to $(1 - \ln p_i)p_i$, ensuring $q_i \leq (1 - \ln p_i)p_i \leq (1 + \beta)p_i$. This framework

guarantees that confidences in Q are at most β times larger than those in T, which implies protection against attribute disclosure. The choice of $-\ln p_i$ is motivated by its properties (see [3]).

Condition 1 of our privacy model ensures that clusters contain no fewer than k elements, thereby protecting against identity disclosure. Condition 2 explicitly quantifies the relationship between the β threshold and positive information gain. Its maximum-distance threshold inherently protects against skewness and semantic attacks [14]. Furthermore, by clearly distinguishing between positive and negative information gain, and allowing for the absence of sensitive attribute values within clusters, Condition 2 provides greater flexibility in anonymization. This results in higher information quality compared to the closely related (k, c)-privacy model. As argued, a measure of relative difference serves our purpose by offering stronger protection for less frequent SA values. Moreover, Condition 2 defines only an upper bound on q_i and allows any q_i value less than p_i , but always disallows q_i values equal to 1 (its upper bound is strictly less than 1). We assert that both of these choices are reasonable for preserving privacy compared to the (k, c)-privacy model.

Our model has several advantages over previous privacy models for longitudinal data. Firstly, it is more intuitive as it measures the risk based on the relative confidence gain of a sensitive value. This allows the data publisher to specify an acceptable maximum relative confidence threshold based on this intuitive measure. Secondly, unlike previous models proposed in [27],[24], our model explicitly quantifies the association between the threshold and the positive gain of information. Finally, our model inherently protects against identity and attribute disclosures, as well as skewness attacks, by enforcing a maximum-distance threshold [3].

Different sensitive attribute values may have varying privacy protection requirements. For instance, certain values like 'flu' for the disease attribute in medical data may be considered less sensitive and acceptable to disclose. To address this variability, we define a subset of sensitive attribute values denoted by $\pi \subseteq SA$, called highly sensitive values, that require stricter protection. In our privacy model, it is crucial to fulfill the second condition for sequences that contain at least one value from π in order to provide enhanced privacy for the specified subset. It is important to note that some works (such as [16, 31]) consider diverse privacy thresholds for sensitive values within anonymization frameworks.

5. The Anonymization Algorithm

In this section, we present FCLA –Fast Clustering-based Longitudinal data Anonymization–, which enforces the model above while striving to preserve data utility. FCLA microaggregates (that is, generates clusters of size at least k [6]) similar sequences based on QIs and implements the privacy model for anonymizing each cluster. Sequence alignment techniques [27] are employed to measure sequence distances, aiding in effective cluster formation.

FCLA (Algorithm 1) uses the longitudinal data set T and privacy model parameters to generate anonymized sequences. Our methodology for the microaggregation component is based on the Maximum Distance to Average Vector (MDAV) algorithm [5], which is an efficient heuristic for creating homogeneous clusters of size at least k. The minimum cluster size k determines privacy, and cluster homogeneity favors utility preservation. First, FCLA defines a distance matrix to keep track of between-cluster distances, and it initializes the distance matrix with the distances between all singleton clusters using sequence alignment techniques. The algorithm begins by selecting a centroid sequence $\hat{c}$ and finding the most distant sequence r. It forms the first cluster G_1 with r and $k-1$ closest unclustered sequences. Additional sequences are added to G_1 until privacy condition 2 in Definition 4 is met for all SA values $s_i \in S$. The process continues by creating new clusters G_i with the farthest unclustered sequences until fewer than $2k$ unclustered sequences remain. If these sequences satisfy privacy condition 2, they form a final cluster; otherwise, they are inserted into other clusters without violating privacy condition 2. After cluster creation, each cluster within *Group* undergoes anonymization in lines 3 to 4 of Algorithm 1. Thus, FCLA identifies and removes sequences violating the privacy requirement to achieve $(k, \beta)^L$-privacy within each cluster G_i. We will provide descriptions of these functions and perform a time complexity analysis using FCLA in the following subsections.

5.1. Identifying sequences that violate privacy

As we assume the attacker possesses a non-empty sequence of L different QI values within or across events, we need to identify the sequences that violate the privacy model. These sequences pose risks of identity and attribute disclosure. A sequence X in T, with a length between 1 and L, violates the privacy requirement if its $sup_s(X)$ violates $(k, \beta)^L$-privacy. We refer to these sequences as violating sequences. For instance, suppose we

Algorithm 1: The Fast Clustering-based Longitudinal Anonymization (FCLA) algorithm

Input: T: data set of Sequences; k, β, L : privacy parameters
Output: Set of anonymized sequences satisfying privacy model

// Microaggregate the sequence data
1 *Group*= Microaggregate the sequences based on QI values
// Generate the anonymized sequences
2 For each cluster of sequences $G_i \in Group$
3 Identify sequences that violate privacy
4 Remove the sequences that violate privacy

consider $k = 2$, $\beta = 1$, $L = 2$, and $R_{SA} = \{Hepatitis, Cancer\}$ with $p_1 = conf(Hepatitis|T) = 0.2$ and $p_2 = conf(Cancer|T) = 0.2$, using Table 2. In this scenario, the sequence $X = \langle\{(Y, 2017)\}, \{(Y, 2019)\}\rangle$ fails to meet the privacy requirement as the record $PID = 6$ matches the sequence X and $|sup_s(X)| = 1$, which is less than the specified threshold of k. Similarly, the sequence $X' = \langle\{(Y, 2018)\}, \{(Y, 2018)\}\rangle$ also violates the privacy requirement as both records $PID = 2$ and $PID = 5$ match sequence X'. Both of these records have the sensitive value $Cancer$ in one of their visits. We have $q_1 = conf(Hepatitis|X') = 0$ and $q_2 = conf(Cancer|X') = 1$. Calculating $D(p_2, q_2)$ gives us $\frac{1-0.2}{0.2} = 4$, which exceeds $\min(\beta, -\ln(p_i)) = \min(1, -\ln(0.2)) = 1$.

One possible approach is to initially identify all potential violating sequences and then proceed with their removal. However, implementing this approach is impractical due to the excessively large number of violating sequences that would need to be enumerated. As a result, this approach becomes infeasible to execute effectively. Consider a violating sequence X with $|sup_s(X)| < k$. Any specific sequence of X', $X' <_S X$, with $|sup_s(X')| > 0$ in T is also a violating sequence because $|sup_s(X')| < |sup_s(X)| < k$. To efficiently identify privacy-violating sequences, we focus on a subset called minimal invalid sequences (MIS). An MIS is defined as a sequence that cannot be further generalized while still violating the privacy requirements.

Let us consider an example with $K = 2$, $\beta = 1$, $L = 2$, and the SA set $R_s = \{Hepatitis, Cancer\}$. In this example, the sequence $S = \langle\{(Z, 41001)\}, \{(L, [4 : 6))\}\rangle$ violates $(k, \beta)^L$-privacy. This violation occurs because $D(p_i, q_i) = 1.5$, which is greater than the specified threshold of $\min(\beta, -\ln(p_i)) = 1$ for *Hep-*

atitis. It is important to note that there is no more general sequence, such as $\langle\{(Z, 4****)\}\rangle$, $\langle\{(Z, 41***)\}\rangle$, $\langle\{(Z, 410**)\}\rangle$, $\langle\{(Z, 4100*)\}\rangle$, $\langle\{(L, [4:6)\}\rangle$ or $\langle\{(L, [4:8)\}\rangle$, that violates the privacy requirements. Hence, S qualifies as an MIS in this case.

Theorem 1. *A longitudinal database T satisfies $(k, \beta)^L$-privacy if and only if it does not contain any minimal invalid sequence (MIS).*

Proof: Let us consider a scenario where a database T does not meet the requirements for $(k, \beta)^L$-privacy, even if it does not contain an MIS. According to the definition of violating sequences, when a database violates the privacy requirements, it must contain violating sequences. However, as per the definition of MIS, a violating sequence either is an MIS itself or contains an MIS. This creates a contradiction with the initial assumption that the database T does not contain any MIS. Therefore, we can conclude that the database T must satisfy $(k, \beta)^L$-privacy because violating sequences cannot exist without the presence of an MIS. □

Algorithm 2 identifies MISs within cluster G_i. To clarify our proposed method, we introduce the concept of direct specializations (dsc). The direct specializations of a value g, denoted as $dsc(g)$, are defined as the set of values u in the domain D that are more specific than g ($u \leq_I g$), and there does not exist any value w in the domain such that $u \leq_I w$ and $w \leq_I g$. For example, dsc([2016:2023])= {[2016:2019],[2020:2023]} regarding VH_Y in Figure 2. Similarly, the direct specializations of an event e=$\{g_1, g_2, ..., g_n\}$ are defined as $dsc_E(e) = \{\hat{e} = (\hat{g}_1, \hat{g}_2, ..., \hat{g}_n)|$ so that there is one value $i \in \{1, ..., n\}$ such that $\hat{g}_i \in dsc(g_i)$ and $\forall j \neq i, g_j = \hat{g}_j\}$.

Algorithm 2 iteratively generates MISs, starting with single events in lines 1 to 12, and expands to incorporate multiple events based on temporal relationships in lines 13 to 29. It begins by generating the most generalized event that covers the root of VH_{A_i}. For example with Table 2, the most generalized event is $\langle$(Y,[2016:2023]),(Z,∗),(D,∗),(L,∗)$\rangle$. Direct specializations of this event are stored in C_1. If an event is valid, it is added to the set V_1 for generating the next candidate set C_2. If an event is invalid, it is included in the set INV_1. The algorithm stores sequences containing i events in the sets C_i and V_i. C_2 is created by performing a self-join operation ($\bowtie$) on V_1 and subsequently eliminating more general sequences and those exceeding length L to reduce the search space for MIS identification. To generate candidate sequences in V_{i+1}, two sequences $X = \langle x_1, \ldots, x_i\rangle$ and $X' = \langle x'_1, \ldots, x'_i\rangle$ in

Algorithm 2: Identifying sequences that violate privacy

Input: G_i: A set of sequences; k,β, L : parameters of the privacy model

Output: Set of anonymized sequences satisfying the privacy model

1 r = The most general event based on $\{DH_{A_1}, ..., DH_{A_n}\}$
2 C_1=$dsc_E(r)$ *//C_1 is a queue of events*
3 $INV_1 = \emptyset$ *// INV_i is a set of MISs*
4 $V_1 = \emptyset$ *// V_i=a set of non-violating sequences*
5 for each $e' \in C_1$ with $len(e) < L$
6 delete e' from C_1
7 If e' does not satisfy $(k, \beta)^L$-privacy in G_i
8 $INV_1 = INV_1 \cup e'$
9 else
10 insert e' into V_1
11 insert $dsc_e(e')$ into C_1
12 generate candidate set C_2 by $V_1 \bowtie V_1$
13 for each sequence $X \in C_2$ do
14 If X is more general than any $v \in V_1$ or $len(X) > L$
15 remove X from C_2
16 $i = 2$
17 while C_i is not empty
18 compute $|sup_s(X)|$ and $conf(s_i|X)$ $\forall$ $X \in C_i$,$s_i \in R_s$
19 for each sequence $X \in C_i$ with $|sup_s(X)| > 0$
20 if X does not satisfy $(k, \beta)^L$-privacy in G_i
21 $INV_i = INV_i \cup X$
22 else
23 insert into V_i
24 i++
25 generate candidate set C_i by $V_{i-1} \bowtie V_{i-1}$
26 for each sequence $X \in C_i$ do
27 if X is more general $\forall v \in V_{i-1}$ or $len(X) > L$
28 remove X from C_i
29 $INV = INV_1 \cup \ldots \cup INV_{i-1}$

V_i are joined if their first $i-1$ events are identical. The candidate sequence is created by appending the event x'_i to X based on specific conditions: if both x_i and x'_i have temporal information, x'_i can be added to X only if its temporal information is neither a descendant nor an ancestor of the temporal information in X; otherwise, x'_i can be added to X only if none of its items is a descendant or an ancestor of any of the items in X. For example, assume $L = 4$, and we have sequences $X = \langle\{(Y, 2018), (L, 2)\}, \{(Y, 2019)\}\rangle$ and $X' = \langle\{(Y, 2018), (L, 2)\}, \{(L, 10)\}\rangle$. In this case, the candidate sequence $X'' = \langle\{(Y, 2018), (L, 2)\}, \{(Y, 2019)\}, \{(L, 10)\}\rangle$ is generated with three events and four items.

5.2. Removing the sequences that violate privacy

Algorithm 3 focuses on removing MISs from cluster G_i. In lines 1 to 8, a multi-domain generalization lattice is created to eliminate MISs within each cluster. Each node in the lattice, denoted as $N = [t_1, ..., t_n]$, corresponds to a vector of n QIs' domains. The values t_i represent the generalization level h for each QI, ranging from 0 to $height(DH_{A_i})$, where $l = 0$ corresponds to the root value and $l = height(DH_{A_i})$ corresponds to the leaf values of DH_{A_i}.

Consider the data set in Table 2. A portion of the corresponding lattice is depicted in Figure 4. Node $N = [Y1, Z1, D0, L0]$ in the lattice means that the values of Y and Z should be generalized to their level-1 descendants, and the values of D and L should be generalized to root level. Its direct generalization is the nodes $N1 = [Y2, Z1, D0, L0]$, $N2 = [Y1, Z2, D0, L0]$, $N3 = [Y1, Z1, D1, L0]$, and $N4 = [Y1, Z1, D0, L1]$.

The algorithm uses a top-down search to construct the lattice, starting from the maximum generalization node. Using direct generalization, child nodes $N_0 \in dsc_e(N)$ are generated and evaluated to eliminate extracted MISs. An MIS, X, can be eliminated by N_0 if at least one item's generalization level in X is higher than in N_0. For example, if $X = \langle\{(L, [0 : 1))\}\rangle$ is an MIS and the generalization level of L at N_0 is 1, X will be removed based on N_0's generalization. In line 8 of Algorithm 3, the child node with the maximum score is selected. For cluster G_i, the score of the sequence set Q under N_0 is given by $Score(N_0) = \frac{|MIS|}{infLoss}$, where $|MIS|$ represents the number of MIS eliminable by N_0, and $infLoss$ denotes the normalized information loss incurred in Q due to anonymization operations on N_0.

A higher-scoring node is better for expansion as it removes more MISs with less information loss. Detailed descriptions of information loss measures can be found in Section 6.1. The search in the generalization lattice stops

Algorithm 3: Removing the sequences that violate privacy

Input: G_i: data set of sequences; k, β, L : parameters of the privacy model

Output: Set of anonymized sequences satisfying the privacy model

```
// Second step: Removing the MISs
1    Node = a node with maximum domain generalization
2    while True
3       children = dsc_e(Node)
4       if ∄c ∈ children such that InfLoss(c) ≤ InfLoss(Node)
5          break
6       else if ∃c ∈ children such that it removes all MIS in INV
7          return c as solution of the sequences in G_i
8       else Node =select c ∈ children with maximum Score()
9    Update the set of INV based on Node
10    if INV ≠ ∅
11       supStrategy = ∅   //This set stores the items to be
 suppressed
12       Items =store all items in INV
13       Sort Items based on their Score()
14       while INV ≠ ∅
15          insert x into Items with maximum Score() into
 supStrategy
16          update INV according to x
```

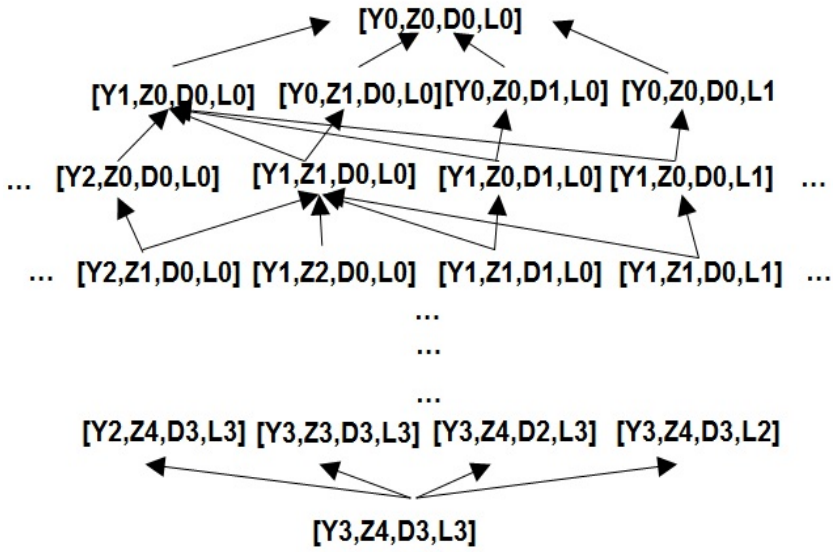


Figure 4: Generalization lattice of Table 2

when the information loss of all child nodes surpasses that of the current node. At line 9 of Algorithm 3, the set INV is updated using the generalization strategy associated with $Node$.

If any MISs remain, a suppression method is employed (lines 10-16). To determine the optimal suppression strategy, all items in INV that cannot be eliminated through generalization are sorted in descending order based on their scores. The score of an item g indicates the number of MISs that contain g and can eliminated by suppressing g. This elimination incurs information loss due to both the generalization strategy of $Node$ and the suppression of items in $\{g\} \cup supStrategy$. The item scores take into account MISs that contain descendant items. In an iterative process, we add the item g with the highest score to the $supStrategy$ set, we remove MISs containing g from the list, and we update the scores of the remaining items. This continues until no more MISs remain. Generalization and suppression strategies may differ across clusters in the $Group$.

5.3. Time Complexity

The time complexity of the FCLA algorithm can be analyzed as follows. In line 1 of Algorithm 1, FCLA initializes the distance matrix with the distances of all singleton clusters using sequence alignment techniques, resulting

in a time complexity $O(t^2 \cdot w^2)$, where t is the number of sequences, and w is the maximum sequence length when dynamic programming is employed for sequence alignments. The algorithm generates clusters based on this distance matrix, resulting in a time complexity $O(t^2)$ for line 1 of Algorithm 1.

This line produces several clusters, with a maximum number of $O(\frac{t}{k})$ clusters. Each cluster contains at least k sequences, satisfying the privacy model constraints. Lines 3 and 4 of the algorithm are executed for each cluster G_i. Anonymization within clusters can be approached as a problem of finding the number of ways to obtain a length L as a sum of integers [36], with a bound $O(F_{G_i}^L)$, where F_{G_i} represents the number of non-minimal invalid sequences with length 1 in cluster G_i. In the worst case, the value of F_{G_i} is given by $n \cdot Height(DH_{A_i})$, where A_i is a QI with the highest domain hierarchy, and n is the number of QIs. Removing MISs with a generalization lattice has a complexity $O(|H| \cdot |MIS|)$, where $|H|$ denotes the number of nodes in the generalization lattice, and $|MIS|$ represents the number of extracted MISs. The total cost complexity of FLCA is $O(t^2 + \frac{t}{k}\{F_{G_i}^L + |H| \cdot |MIS|\})$. While pruning strategies can significantly reduce this complexity in practice, removing MISs through suppression has a similar computational cost.

When comparing with the closest existing work, HALT [24], it must be noted that HALT searches for violating sequences throughout the entire data set with a time complexity $O(F_T^L)$, where F_T represents the number of non-minimal invalid sequences with length 1 in data set T. In contrast, our method operates in parallel on clusters with sizes $k < |G_i| < t$. Thus, FCLA outperforms HALT in terms of time complexity. In the worst-case scenario where FCLA generates one cluster with a size $|t|$, the time complexity of our method is comparable to that of HALT.

6. Experimental Results

In this section, we present an experimental evaluation of our anonymization framework. Given that our study is the first to tackle the challenge of mitigating skewness attacks in privacy-preserving longitudinal data, direct comparisons with the techniques proposed by Terrovitis and Mamoulis [29] and Tamersoy *et al.* [27], primarily aimed at protecting against identity disclosure, are not pertinent. Nevertheless, we conducted experiments to compare our proposed anonymization framework with state-of-the-art privacy protection methods closely related to ours, such as HALT [24]. HALT's primary objective is to achieve k-anonymity while imposing constraints on

the percentage C of records containing sensitive values, to prevent both identity and attribute disclosures. To ensure a fair and meaningful comparison with other methodologies, we adopted the same parameter values as those specified in [24]. Specifically, we varied the privacy parameters, including k, β, and L, within the following ranges: $[5 : 45]$ for k, $[1 : 7]$ for β, and $[2 : 16]$ for L. Experiments were conducted on a workstation with an Intel Core i5 processor and 8GB RAM, running at 2.4 GHz. Our algorithms were implemented in Python. The experiments were performed on three standard health data sets:

- CMS DE-SynPUF, provided by the Centers for Medicare and Medicaid Services (CMS), contains three years (2008 to 2010) of synthetic claims for hospital inpatient services. Each record represents a synthetic inpatient claim (hospital visit). Due to file size limitations, CMS released inpatient data in 20 separate samples. Each file contains 81 attributes, including sensitive information related to patient diagnoses and procedures. To protect privacy, we focused on one sensitive attribute, Diagnosis. For each patient visit, the diagnosis attribute is determined based on values from several related attributes. If any of these attributes contains highly sensitive data, the diagnosis attribute is set to that value; otherwise, a random value from these attributes is assigned to the diagnosis attribute. Some attributes such as CLM_FROM_DT and CLM_THRU_DT representing claim start and end dates, were almost identical to CLM_ADMSN_DT (admission date) and NCH_-BENE_DSCHRG_DT (discharge date) in about 99.8% of records in the 20 sample files. As a result, we discarded CLM_FROM_DT and CLM_THRU_DT considering CLM_ADMSN_DT and NCH_-BENE_DSCHRG_DT as quasi-identifiers. We also included CLM_-UTLZTN_DAY_CNT (length of hospital stay) and CLM_PMT_AMT (claim payment amount) as potential quasi-identifiers, while excluding other payment-related attributes due to low variability or lack of quasi-identifier characteristics. The remaining attributes in the original DE-SynPUFs data are neither sensitive nor quasi-identifiers, and thus, they do not affect the anonymization process. In our study, we identified certain values such as HIV, abortion, abuse, psychosexual disorders, mental retardation, or plastic surgery as more sensitive. These specific values, referred to as the set π mentioned in Section 4, require protection to prevent attribute disclosure. Additionally, we introduced

an extra quasi-identifier, DSFC, as suggested by [24], which tracks the number of days since the initial claim calculation for each patient in each calendar year. We generated five data sets using sample files from CMS DE-SynPUF. Our algorithm was applied to these data sets, and the results were averaged. Each data set comprised 66,253 visits, with a maximum of 12 visits per patient, an average of 8 visits per patient, and 8.12% of high sensitive values in a data set.

- The Heritage Health Prize (HHP) claims data set was part of a data mining competition aimed at predicting patients' hospitalization duration using historical claims data, which included various tables like claims and lab data. Our study focused on HHP claims data, encompassing patient claims over three years. In [7], the anonymization of HHP claims data to prevent identity disclosure attacks was examined. The focus was on attributes Specialty, Place of service (PlaceSvc), Length Of Stay in the hospital (LOS), Days Since the First Service that year (DSFS), (admission) Year, Diagnosis, and CPTCode. We chose DSFS and LOS as our quasi-identifiers. PlaceSvc and Specialty were discarded due to unavailable original values. Instead, we used patient zip codes and claim years as additional quasi-identifiers for each claim. As with the previous data set, we used Diagnosis as the sensitive attribute. The set of highly sensitive values, π, was chosen to be similar to the CMS DE-SynPUF data set. To create a diverse set of synthetic data sets, we generated five variations, each one encompassing 1,000 patients. The average number of events (*i.e.*, claims) per sequence was set at 10, with a maximum of 16 events.

- The FDA Adverse Event Reporting System (FAERS) data set is provided by the U.S. Food and Drug Administration (FDA) and released quarterly. Each report in FAERS is uniquely identified by an attribute called ISR, and contains an attribute CaseID to identify distinct individuals, along with some demographic information such as Weight, Age, and Gender, drug name (Drug) and indication ($INDI_PT$), and reaction information (PT). We used Weight, Age, Gender as quasi-identifiers (QIDs) and CaseID as the individual identifier. We utilized the drug name and created a new feature named "disease". Disease values were constructed by referencing websites such as Drugs.com and wrongdiagnosis.com, which provide information on which drugs

are effective for specific diseases. Data sets from 2004Q1 to 2011Q4 were selected to do experiments, where any record with QID containing missing values was discarded. To create a diverse set of synthetic data sets, we generated five variations, each one encompassing 1,000 patients. The average number of events (*i.e.*, ISR) per sequence was set at 4, with a maximum of 7 events.

Note that the value of β is determined by expert knowledge and context-dependent factors. In this work, we set β based on the percentage of more sensitive values within each data set. For instance, in the CMS DE-SynPUF data set, we identified several values, including those related to HIV, abortion, abuse, psychosexual disorders, mental retardation, and plastic surgery, as more sensitive, consistent with [24]. These more sensitive values constitute a maximum of 8.12% of the data set. Setting $\beta = 1$ results in a frequency threshold of $e^{-\beta} \approx 37\%$, classifying all more sensitive values as "infrequent". This allows the frequency of any sensitive attribute value within any group to be at most $8.12\% \times 2 = 16.24\%$. Consequently, $\beta = 1$ represents a relatively small value. For our experiments, we investigate β values in $\{1, 2, 3, 4, 7\}$. In our privacy model, it is essential to satisfy the second condition for sequences containing at least one value from π. This affords enhanced privacy protection for the specified subset of data. Sequences that do not contain more sensitive values do not satisfy the second condition.

6.1. Information Loss Measures

Different measures have been proposed to quantify information loss incurred by anonymization [1]. We use the normalized certainty penalty [34], denoted as $NCP(A_i, u)$, to measure the information loss of value u from the value generalization hierarchy of A_i. The information loss at node u is determined by the ratio of the subtree size rooted at u to the total number of leaves in the value generalization hierarchy of A_i. If u is a leaf node, the subtree size is considered to be 0. For example, for the value generalization hierarchy of attribute Y and the node [2016:2019], the information loss is calculated as $4/8 = 0.5$, which indicates that [2016:2019] covers four leaf values (2016, 2017, 2018, and 2019), while the total domain size of Y is 8 (2016 to 2023). In a set of sequences S, each sequence comprises multiple events, with each event containing an attribute value A_i. The overall information loss incurred by the generalization strategy $Node$ on a set of sequences, Q,

is defined as

$$InfLoss(Node) = \sum_{X \in Q} \sum_{e \in X} \sum_{A_i \in A} NCP(A_i, u), \tag{1}$$

where X denotes a sequence in Q, $Q \subseteq T$, e represents an event in X, A refers to the set of QIs, and $NCP(A_i, u)$ represents the information loss at value u of attribute A_i.

We utilize a measure introduced by [24] to assess the information loss caused by suppression. When a leaf value u is suppressed, the information loss is 1. The total loss for suppressing all instances of u is given by $count(u)$. For non-leaf values in the value generalization hierarchy, the information loss for suppressing one instance of u is 1 - $NCP(A_i, u)$. Consequently, with a suppression strategy *supStrategy* that involves suppressing multiple items, the total information loss on a set of sequences S can be calculated as

$$InfLoss(supStrategy) = \sum_{u \in supStrategy} count(u) \cdot (1 - NCP(A_i, u)). \tag{2}$$

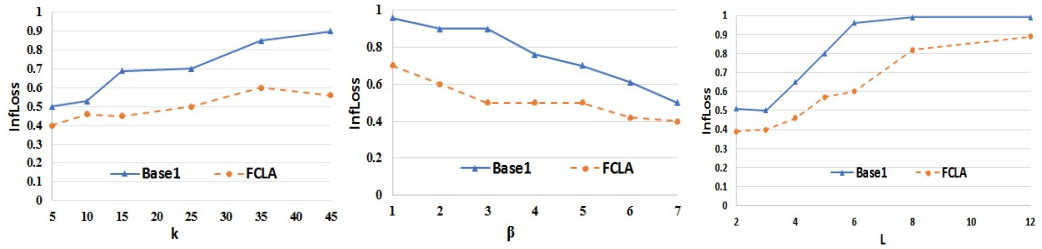
The normalized total information loss on a set of sequences S, considering the combined information loss from both the generalization and suppression strategies, is calculated for every possible anonymization solution by using the following expression:

$$InfLoss = \frac{InfLoss(Node) + InfLoss(supStrategy)}{|S| \cdot |A|}. \tag{3}$$

6.2. Performance of our Proposed Method

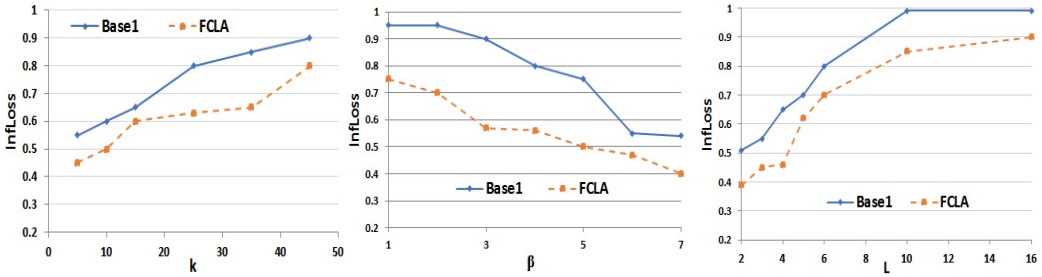
We evaluate the utility preservation of FCLA compared to a baseline algorithm named Base1. Base1 is a clustering-based approach similar to FCLA but without dynamic programming for distance calculations. Instead, Base1 utilizes a list of representative event pairs to compute distances [5]. In the following, we assess the impact of different privacy parameters on the information loss of anonymized data.

Figures 5(a), 6(a), and 7(a) show the information loss analysis of FCLA and Base1 algorithms on the CMS DE-SynPUF, HHP claims, and FAERS data sets. The evaluation was conducted for $\beta = 6$, $L = 3$, and values of k varying from 5 to 45. Increasing k resulted in higher information loss for both algorithms. However, it is noteworthy that FCLA occasionally exhibited a



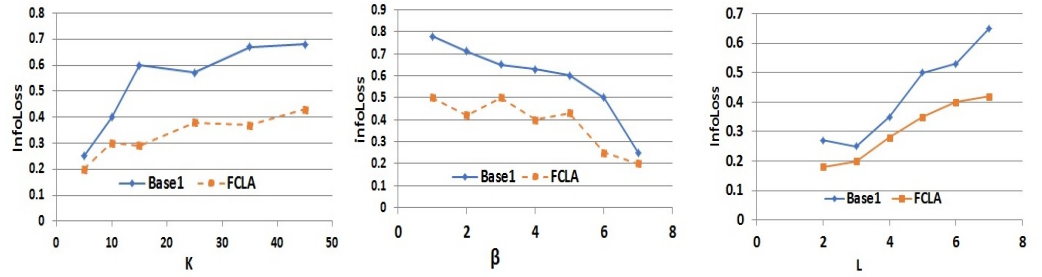
(a) (b) (c)

Figure 5: Information loss of FCLA vs Base1 on the DE-SynPUFs data set when: (a) varying k values, $\beta = 6$, $L = 3$; (b) varying β values, $k = 5$, $L = 3$; (c) varying L values, $k = 5$, $\beta = 6$.



(a) (b) (c)

Figure 6: Information loss of FCLA vs Base1 on the Heritage Health Prize claims data set when: (a) varying k values, $\beta = 6$, $L = 3$; (b) varying β values, $k = 5$, $L = 3$; (c) varying L values, $k = 5$, $\beta = 6$.



(a) (b) (c)

Figure 7: Information loss of FCLA vs Base1 on the FAERS Data set when: (a) varying k values, $\beta = 6$, $L = 3$; (b) varying β values, $k = 5$, $L = 3$; (c) varying L values, $k = 5$, $\beta = 6$.

decrease in information loss for higher k values due to its greedy nature. All in all, FCLA's information loss was lower than that of Base1. This can be attributed to FCLA's utilization of dynamic programming for sequence alignment. This technique helps FCLA preserve data utility by maintaining intrinsic characteristics and patterns.

In Figures 5(b), 6(b), and 7(b), the impact of the different values of β on information loss is shown. With $k = 5$ and $L = 3$, increasing β led to a decrease in information loss. Smaller values of β resulted in a higher number of MIS within the data set. Consequently, more data distortion was required to eliminate these MIS during anonymization. By increasing β, FCLA prioritized reducing information loss by minimizing data distortion. This trade-off between privacy protection and data utility is reflected in the decreased information loss.

In Figures 5(c), 6(c), and 7(c), the impact of the adversary's background knowledge on data utility and information loss is shown. The length of the adversary's knowledge (L) was varied from 2 to the maximum length of the events in the corresponding data set, which were 12, 16, and 7, respectively, while k was fixed at 5 and β was fixed at 6. It is observed that for smaller values of L ($L = 2$ and $L = 3$), the information loss was relatively low. This indicates less presence of MIS of size 2 and 3 in the data, resulting in lower information loss during anonymization. In this experiment, the maximum value of length referred to attackers with perfect knowledge, meaning the length of the known sequences was set to the maximum length in the data set. The results show that the information loss of FCLA was less than that of Base1. However, as L increased, the number of extracted MIS grew, which required more extensive generalizations and suppressions to eliminate them, and hence led to greater information loss. Higher L values indicate an adversary with increased background knowledge. Consequently, more data distortion was required to safeguard privacy, thereby resulting in increased information loss.

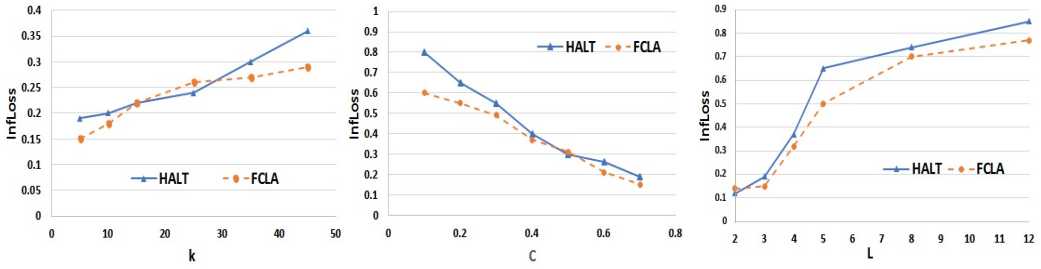
Figures 5(c), 6(c), and 7(c) highlight the importance of considering the adversary's background knowledge when determining the appropriate level of anonymization. Understanding the extent of the adversary's knowledge allows data controllers to make informed decisions about the necessary level of generalization and suppression to achieve a balance between privacy protection and data utility.

6.3. Comparison with other methods

We conducted a comparative analysis of our results with the closely related HALT method [24]. HALT is designed to anonymize longitudinal data while addressing identity and attribute disclosure concerns. However, a direct and fair comparison with HALT was challenging due to the utilization of different privacy models. Nevertheless, our framework offers a $(k, C)^L$-privacy guarantee, similar to that of HALT. Condition 2 in our proposed model differs from Condition 2 in the $(k, C)^L$-privacy model. In $(k, C)^L$-privacy, Condition 2 ensures that the percentage of records in the set of sequences, Q, containing the sensitive attribute (SA) value s_i for any SA values, is below a specified threshold C. In FCLA, we address Condition 2 of the $(k, C)^L$-privacy model in line 1 of Algorithm 1. Moreover, line 7 of Algorithm 2 in FCLA also ensures adherence to the $(k, C)^L$-privacy model.

We took parameter values adjusted in accordance with the recommendations provided in [24]. Firstly, we compared the performance of the two methods in terms of information loss. Figure 8 presents experimental results of HALT and FCLA on the evaluation data set with different parameter values. In Figure 8(a), FCLA outperforms HALT in terms of information loss for varying values of k ($C = 0.7$, $L = 3$), thus achieving lower information loss and better data utility. Similarly, in Figure 8(b), FCLA consistently demonstrates lower information loss compared to HALT for different values of C (with $k = 5, L = 3$). Lastly, Figure 8(c) shows that FCLA exhibits better performance with reduced information loss compared to HALT for different values of L ($k = 5, C = 0.7$). In this experiment, L was varied from 2 to 12, as the maximum length of sequences in the CMS DE-SynPUF data set was 12.

In particular, this means that for $L = 12$ we conducted the experiment with an attacker who had perfect knowledge. By doing so, we effectively demonstrated the superior utility benefits of our methods compared to HALT in scenarios where the attacker possesses complete knowledge. These results highlight the effectiveness of our framework at preserving data utility while meeting the privacy requirements of HALT. Under the same privacy model, HALT identifies patterns that violate the privacy model across the entire data set and generalizes the data without considering the similarity of QI values and data utility. In contrast, our method partitions sequences into groups with similar QI values, thereby preserving utility and handling violating sequences independently within each group. In this way, our method is able to generalize groups in parallel, resulting in reduced time costs. No-



(a) (b) (c)

Figure 8: Information loss of FCLA vs HALT on the DE-SynPUFs data set when: (a) varying k values, $C = 0.7$, $L = 3$; (b) varying C values, $k = 5$, $L = 3$; (c) varying L values, $k = 5$, $C = 0.7$.

tably, our approach differs from HALT in terms of identifying and removing violating sequences.

Finally, we compared the execution times of our algorithm and HALT on the CMS DE-SynPUF data set. To do that, we set $k = 5$ and $C = 0.7$. The L values were selected within the range $[1 : 12]$, as the maximum length of a sequence in the CMS DE-SynPUF data set was taken to be 12. Figure 9 displays the execution times of both FCLA and HALT for various values of L. It is important to emphasize that the Y-axis is represented on a logarithmic scale. Our findings indicate a substantial performance advantage for FCLA over HALT. We next give an explanation for this performance difference. HALT identifies violating sequences by extending a given sequence S through two distinct procedures: one appends a new event consisting of a single item to the end of S, and the other adds a single item to the last event of S. On the other hand, HALT considers all items in the entire data set during both of these procedures. In contrast, our method focuses exclusively on items within a specific cluster, denoted as G_i, where $k \leq |G_i| \leq t$. Furthermore, FCLA employs a search strategy that involves finding violating sequences by joining violating sequences. This results in a smaller search space compared to HALT. It is worth noting that Figure 9 provides empirical evidence that confirms the time complexity analysis presented in Section 5.3.

7. Conclusions and Future Work

We have proposed an anonymization framework for longitudinal data where the adversary's knowledge is limited to a certain number of events.

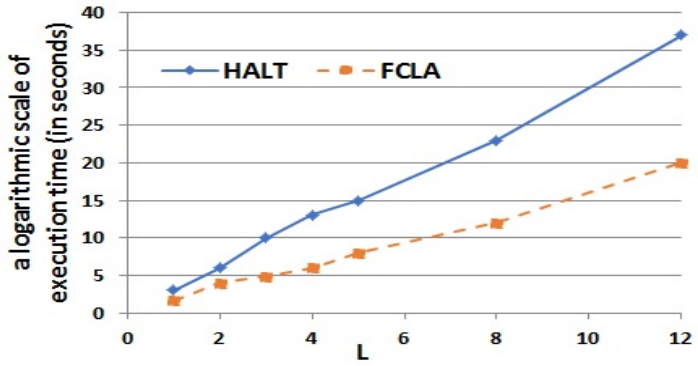


Figure 9: Comparison of the execution times of FCLA and HALT on the DE-SynPUFs data set when varying L values, with $k = 5$ and $C = 0.7$

Our framework can be easily adapted to consider different adversaries with varying degrees of background knowledge, thereby encompassing a variety of privacy requirements. Unlike related works, our approach effectively addresses identity disclosure, attribute disclosure, and skewness attacks in an integrated way. Also, by partitioning data into sequence groups, our method enables efficient parallel groupwise executions. Empirical work confirms that our approach outperforms the most similar state-of-the-art method at preserving both privacy and data utility, as well as in terms of computation time.

As future work, we plan to investigate data republishing strategies in longitudinal data scenarios, addressing a gap in existing dynamic data publishing researches (such as [2],[15], and [32]). In practice, data sources are constantly changing, and data sets may be regularly updated and republished. Current approaches do not handle the unique challenges of anonymizing longitudinal data. Consequently, a privacy violation might arise from any individual release or as a result of combining information from multiple releases.

Acknowledgments

This research was partially supported by MCIN/AEI/ 10.13039/501100011033 (project PID2021-123637NB-I00 “CURLING”), INCIBE/EU-NextGeneration (project “HERMES” and INCIBE-URV cybersecurity chair), the European Commission (project H2020-871042 “SoBigData++”), and the Government of Catalonia (ICREA Acadèmia Prizes to the second and third authors).

References

[1] Amiri F, N. Yazdani, A. Shakery, A.H. Chinae (2016),"Hierarchical anonymization algorithms against background knowledge attack in data releasing," Knowledge Based Systems, 101, pp. 71–89.

[2] Amiri, F., N. Yazdani, A.Shakery, S.S. Ho (2019), "Bayesian-based anonymization framework against background knowledge attack in continuous data publishing". Transactions on Data Privacy, 12(3), pp. 197–225.

[3] Cao J, P. Karras (2012), "Publishing microdata with a robust privacy guarantee," Proceedings of the VLDB Endowment, 5(11), pp. 1388–1399.

[4] Coletti A. S, P. J. Heagerty , A. R. Sheon , M. Gross , B. A. Koblin, D. S. Metzger, G. R. Seage (2003), "Randomized, controlled evaluation of a prototype informed consent process for HIV vaccine efficacy trials," Journal of Acquired Immune Deficiency Syndrome, 32: 161–169.

[5] Domingo-Ferrer J, S. Martínez, D. Sánchez (2022), "Decentralized k-anonymization of trajectories via privacy-preserving tit-for-tat," Computer Communications, 190, pp. 57-68.

[6] Domingo-Ferrer J, J. M. Mateo-Sanz (2002), "Practical data-oriented microaggregation for statistical disclosure control". IEEE Transactions on Knowledge and Data Engineering, 14(1), pp. 189–201.

[7] ElEmam K, L. Arbuckle, G. Koru, L. Gaudette, E. Neri, S. Rose, J. Howard, J. Gluck (2012), "De-identification methods for open health data: the case of the Heritage Health Prize Claims data set," Journal of Medical Internet Research, 14:1, p. e33.

[8] Fournier-Viger P, W. Gan, Y. Wu, M. Nouioua, W. Song, T. Truong, H. Duong (2022), "Pattern mining: Current challenges and opportunities," In International Conference on Database Systems for Advanced Applications, pp. 34–49.

[9] Framingham Heart Study (FHS), `https://www.framinghamheartstudy.org/`

[10] Fung B. C. M, K. Wang, R. Chen, P. S. Yu (2010), "Privacy-preserving data publishing: A survey on recent developments," ACM Computing Surveys (CSUR), 42(4), pp. 1-53.

[11] Gkoulalas-Divanis A, G. Loukides (2012), "Utility-guided clustering-based transaction data anonymization," Transactions on Data Privacy, 5, pp. 223–251.

[12] Jin F, W. Hua, M. Francia, P. Chao, M. E. Orlowska, X. Zhou (2023), "A survey and experimental study on privacy-preserving trajectory data publishing," IEEE Transactions on Knowledge and Data Engineering, 35, pp. 5577–5596.

[13] Kunkle D, D. Zhang, G. Cooperman (2008), "Mining frequent generalized itemsets and generalized association rules without redundancy," Journal of Computer Science and Technology, 23(1), pp. 77-102.

[14] Li N, T.Li, S.Venkatasubramanian (2006), "t-closeness: Privacy beyond k-anonymity and l-diversity," In 2007 IEEE 23rd International Conference on Data Engineering, pp. 106–115.

[15] Li, S., H.Tian, H.Shen, Y.Sang (2021), "Privacy-preserving trajectory data publishing by dynamic anonymization with bounded distortion". ISPRS International Journal of Geo-Information, 10(2), pp. 78.

[16] Lin, W. Y., D.C. Yang, J.T. Wang (2015), "Privacy preserving data anonymization of spontaneous ADE reporting system data set". In Proceedings of the ACM Ninth International Workshop on Data and Text Mining in Biomedical Informatics, pp. 2–16.

[17] Liu X, L. Wang, Y. Zhu (2018), "SLAT: sub-trajectory linkage attack tolerance framework for privacy-preserving trajectory publishing", In 2018 International Conference on Networking and Network Applications (NaNA), pp. 298-–303.

[18] Machanavajjhala P, J.Gehrke, D.Kifer, M.Venkitasubramaniam (2007), "L-diversity: privacy beyond k-anonymity," ACM Transactions on Knowledge Discovery from Data (TKDD), 1(1), pp. 3–es.

[19] Parameshwarappa P, Z. Chen, G. Koru (2021), "Anonymization of daily activity data by using l-diversity privacy model," ACM Transactions on Management Information Systems, 12, pp. 1–21.

[20] Residori C, M.E. Sozio, L. Schomaker, R. Samuel (2020), "YAC Young people and Covid-19. Preliminary results of a representative survey of adolescents and young adults in Luxembourg," Esch-sur-Alzette: University of Luxembourg (UL) and Ministère de l'Éducation nationale, de l'Enfance et de la Jeunesse (MENJE).

[21] Ruiz, N. (2018), "A general framework and metrics for longitudinal data anonymization". In Privacy in Statistical Databases (PSD 2018), Lecture Notes in Computer Science, vol 11126, pp. 215–230.

[22] Samarati P, L. Sweeney (1998), "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," Technical Report. SRI-CSL-98-04.

[23] Scottish Longitudinal Study, `https://calls.ac.uk/ls-units/scottish-longitudinal-study-scotland/`

[24] Sehatkar M, S. Matwin (2013), "HALT: Hybrid anonymization of longitudinal transactions," In: 2013 Eleventh Annual Conference on Privacy, Security and Trust, pp. 127–134.

[25] Sozio M.E, A. Procopio, R. Samuel (2020). "Youth survey Luxembourg. Technical report 2019 ," Esch-sur-Alzette: Ministère de l'Éducation nationale, de l'Enfance et de la Jeunesse (MENJE) and University of Luxembourg (UL).

[26] Szefler S, S. Weiss, A. Tonascia, N. Adkinson, B.R.C. Bender, M. Donithan, H. Kelly, J. Reisman, G. Shapiro, G. Sternberg, R. Strunk , V. Taggart , M. VanNatta, R. Wise, M. Wu, R. Zeiger (2000), "Long-term effects of budesonide or nedocrimil in children with asthma," New England Journal of Medicine, 343(15): 1054–1063.

[27] Tamersoy A, G. Loukides, M.E. Nergiz, Y. Saygin, B. Malin (2012), "Anonymization of longitudinal electronic medical records," IEEE Transactions on Information Technology in Biomedicine, 16, pp. 413–423.

[28] Tassa T, A. Mazza, A. Gionis (2012), "k-Concealment: an alternative model of k-type anonymity," Transactions on Data Privacy, 5, pp. 189–222.

[29] Terrovitis M, N. Mamoulis (2008),"Privacy preservation in the publication of trajectories," In 9th International Conference on Mobile Data Management, pp. 65–72.

[30] Terrovitis, M., N. Mamoulis, P.Kalnis (2008), "Privacy-preserving anonymization of set-valued data". Proceedings of the VLDB Endowment, 1(1), pp.115-125.

[31] Wang. K., B. C. M. Fung, P. S. Yu (2006), "Handicapping attacker's confidence: An alternative to k-anonymization", Knowledge and Information Systems: An International Journal (KAIS), 11, pp. 345-368.

[32] Wang, J.T., W.Y. Lin (2021), "Privacy-preserving anonymity for periodical releases of spontaneous adverse drug event reporting data: algorithm development and validation", JMIR Medical Informatics, 9(10), p. e28752.

[33] Xu Y., K. Wang, A.W.-C. Fu, P. S. Yu (2008), "Anonymizing transaction databases for publication". In Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 767–775.

[34] Xu J., W. Wang, J. Pei, X. Wang, B. Shi, A. Fu (2006), "Utility-based anonymization using local recoding." In Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–790.

[35] Yao L., X. Wang, X. Wang, H. Hu, G. Wu (2019), "Publishing sensitive trajectory data under enhanced L-diversity model." In 20th IEEE International Conference on Mobile Data Management, pp. 160–-169.

[36] Zaki M.J (2001),"Spade: An efficient algorithm for mining frequent sequences," Machine Learning, 42, pp. 31–60.