Title:

NOVEL ANONYMIZATION APPROACHES ENSURING PRIVACY: APPLICATION IN E-HEALTH

Abstract: Recently, collecting data has become crucial for most organizations due to the fast growth of data analytics tools that allow better use of the raw collected data ensuring higher added value and positive impact for these organizations. However, the explosive quantity of data that is collected might contain personally identifiable information (PII) that should be protected to be compliant with related laws and regulations.

For example, in the health sector, no doubt that the use of recent Information and Communication Technologies (Cloud computing, Internet of Things, Big Data, Artificial Intelligence, ...) improve communication and access to the right information on the right time and guarantee a high quality of care to patients. However, the collected, stored and processed data by these technologies often include sensitive information that arise new security and privacy concerns. Many approaches and solutions are used to mitigate such issues. In particular, concerning privacy it is widely agreed that anonymization techniques are considered among the most efficient approaches.

In this thesis, we first provide a new detailed classification of the most used cryptographic and non-cryptographic anonymization techniques ensuring privacy. Besides, we evaluate the presented techniques through data completeness, confidentiality and data accuracy criteria. Next, we focus more on three relevant anonymization techniques belonging to Generalization-based approaches that are: *K*-anonymity, *L*-diversity and *T*-closeness techniques. Our second contribution in this thesis concerns a novel way in applying *K*-anonymity principle for quasi-identifier (QI) attributes. In fact, unlike other works, we have used the principle of *K*-anonymity without specifying a prior value of the threshold *K*.

Afterwards, we proposed an algorithm that deals with sensitive attributes by using the principle of *L*-diversity. This algorithm ensures privacy while reducing the correlation loss among attributes. However, *L*-diversity technique cannot resist against the Similarity attack. That is why; we developed two main algorithms that test the degree of proximity for both numerical and categorical attributes. Besides, we have measured the information loss through a utility measurement called Normalized certainty penalty (NCP) before and after applying the anonymization process on categorical attributes. In fact, the combination of these proposed algorithms ensures privacy, preserves data utility and treats both QI and sensitive attributes.

Keywords: Privacy, Anonymization, E-health, *K*-anonymity, *L*-diversity, *T*-closeness, Similarity attack, NCP.

Name : Zakariae EL OUAZZANI

Title : Novel anonymization approaches ensuring privacy: application in e-health

Thesis N° : 220/ST2I

Year : 2021

Année : 2021

Thèse N° : 220/ST2I

École Nationale Supérieure d'Informatique et d'Analyse des Systèmes
Centre d'Études Doctorales en Sciences des Technologies de l'Information et de l'Ingénieur

THÈSE DE DOCTORAT

NOVEL ANONYMIZATION APPROACHES ENSURING PRIVACY: APPLICATION IN E-HEALTH

Présentée par

Zakariae EL OUAZZANI

Le 19/06/2021

Formation doctorale : Informatique
Structure de recherche : Smart Systems Laboratory

JURY

Professeur Rachida AJHOUN PES, ENSIAS, Université Mohammed V, Rabat	**Président**
Professeur Hanan EL BAKKALI PES, ENSIAS, Université Mohammed V, Rabat	**Directeur de thèse**
Professeur Abdelkrim HAQIQ PES, FST, Université Hassan 1er, Settat	**Rapporteur**
Professeur Nabil BENAMAR PES, EST, Université Moulay Ismail, Meknès	**Rapporteur**
Professeur Mostapha ZBAKH PES, ENSIAS, Université Mohammed V, Rabat	**Rapporteur**
Professeur An BRAEKEN Associate Professor, Vrije Universiteit Brussel (VUB), Belgique	**Examinateur**
Professeur Abderrahmane NITAJ Professeur HDR, Université de Caen Normandie, France	**Examinateur**

Acknowledgments

First of all, I am grateful to Allah for the good health and well-being that were necessary to complete this thesis. Then, many thanks are in order. First, I must thank my parents, Mohamed Ben Taleb El Ouazzani and Badia Labzour, my 4 sisters Hanan, Laila, Rajae and Lamiae for motivating me to prepare and write this thesis. No word can describe my gratitude towards my father who helped me a lot in so many ways.

My advisor, Hanan EL BAKKALI, deserves a great deal of thanks for, without her tutelage, this thesis would not have come to be completed.

A special thanks to Prof. An BRAEKEN from VUB (Vrije Universiteit Brussel) for all her efforts and recommendations during and after my research visit at VUB.

The honorable members of the jury, professor Rachida AJHOUN, professor Abdelkrim HAQIQ, professor Nabil BENAMAR, professor Mostapha ZBAKH, professor An BRAEKEN and professor Abderrahmane NITAJ have earned my thanks for reading this thesis all the way to the end and examining whether I was qualified to present this work.

My cousin Abdelghaffar and my friends Hamza Ait El Maalem and Khalid El Mrabet who supported me throughout my thesis. I also want to thank Mounia for her help.

I take this opportunity to express my gratitude to all of the PhD students of the SSLab and all the faculty members of ENSIAS (my former professors as an engineering student).

Finally, I wish to thank every one who has somehow helped me in the realization of this work, Thanks to all!

Abstract

Recently, collecting data has become crucial for most organizations due to the fast growth of data analytics tools that allow better use of the raw collected data ensuring higher added value and positive impact for these organizations. However, the growing quantity of data that is collected might contain personally identifiable information (PII) that should be protected to be compliant with related laws and regulations.

For example, in the health sector, no doubt that the use of recent Information and Communication Technologies (Cloud computing, Internet of Things, Big Data, Artificial Intelligence, ...) improve communication and access to the right information on the right time and guarantee a high quality of care to patients. However, the collected, stored and processed data by these technologies often include sensitive information that arise new security and privacy concerns. Many approaches and solutions are used to mitigate such issues. In particular, concerning privacy it is widely agreed that anonymization techniques are considered among the most efficient approaches.

In this thesis, we first provide a new detailed classification of the most used cryptographic and non-cryptographic anonymization techniques ensuring privacy. Besides, we evaluate the presented techniques through data completeness, confidentiality and data accuracy criteria. Next, we focus more on three relevant anonymization techniques belonging to *Generalization-based* approaches that are: *K-anonymity*, *L-diversity* and T-closeness techniques. Our second contribution in this thesis concerns a novel way in applying *K-anonymity* principle for *quasi-identifier* (QI) attributes. In fact, unlike other works, we have used the principle of *K-anonymity* without specifying a prior value of the threshold K.

Afterwards, we proposed an algorithm that deals with sensitive attributes by using the principle of *L-diversity*. This algorithm ensures privacy while reducing the correlation loss among attributes. However, *L-diversity* technique cannot resist against the Similarity attack. That is why; we developed two main algorithms that test the degree of proximity for both numerical and categorical attributes. Besides, we have measured the information loss through a utility measurement called Normalized Certainty Penalty (NCP) before and after applying the anonymization process on categorical attributes. In fact, the combination of these proposed algorithms ensures privacy, preserves data utility and treats both QI and sensitive attributes.

Keywords: Privacy, Anonymization, E-health, *K-anonymity*, *L-diversity*, *T-closeness*, Similarity attack, *NCP*.

Résumé

Récemment, la collecte de données est devenue cruciale pour la plupart des organisations en raison de la croissance rapide des outils d'analyse de données qui permettent une meilleure utilisation des données brutes collectées garantissant une plus grande valeur ajoutée et un impact positif pour ces organisations. Cependant, la quantité croissante de données collectées peut contenir des informations personnellement identifiables (PII) qui doivent être protégées pour être conformes aux lois et réglementations connexes.

Par exemple, dans le secteur de la santé, il est clair que l'utilisation des récentes Technologies de l'Information et de la Communication (Cloud Computing, Internet des Objets, Big Data, Intelligence Artificielle, ...) améliore la communication et l'accès aux bonnes informations au bon moment et garantie une meilleure qualité des soins aux patients. Cependant, les données collectées, stockées et traitées par ces technologies incluent souvent des informations sensibles (ou "sensitive") qui soulèvent de nouveaux défis en matière de sécurité et de protection de la vie privée (ou "privacy"). De nombreuses approches et solutions sont utilisées pour atténuer ces problèmes. En particulier, en ce qui concerne la protection de la vie privée (ou "privacy"), il est largement admis que les techniques d'anonymisation sont considérées parmi les approches les plus efficaces.

Dans cette thèse, nous proposons tout d'abord une nouvelle classification détaillée des techniques les plus utilisées d'anonymisation cryptographiques et non cryptographiques garantissant la protection de la vie privée (ou "privacy"). En outre, nous évaluons les techniques présentées à travers des critères d'exhaustivité, de confidentialité et d'exactitude des données. Ensuite, nous nous concentrons davantage sur trois techniques d'anonymisation pertinentes appartenant à des approches basées sur la généralisation qui sont : "*K-anonymity*", "*L-diversity*" et "T-closeness". Notre deuxième contribution dans cette thèse concerne une nouvelle manière d'appliquer le principe de "*K-anonymity*" pour les attributs "*quasi-identifier*" (QI). En fait, contrairement à d'autres travaux, nous avons utilisé le principe de "*K-anonymity*" sans spécifier une valeur préalable au seuil K.

Ensuite, nous avons proposé un algorithme qui traite les attributs sensibles (ou "sensitive") en utilisant le principe de "*L-diversity*". Cet algorithme garantit la protection de la vie privée (ou "privacy") tout en réduisant la perte de corrélation entre les attributs. Cependant, la technique "*L-diversity*" ne peut pas résister contre l'attaque de Similarité. C'est pourquoi, nous avons développé deux principaux algorithmes qui testent le degré de proximité pour les attributs numériques et catégoriels. De plus, nous avons mesuré la perte d'information par une mesure d'utilité appelée pénalité de certitude normalisée (NCP) avant et après l'application du processus d'anonymisation sur les attributs catégoriels. En fait, la combinaison de ces algorithmes proposés garantit la protection de la vie privée (ou "privacy"), préserve l'utilité des données et traite à la fois les attributs QI et sensibles (ou "sensitive").

Mots-clés: Protection de la vie privée "Privacy", Anonymisation, Santé-mobile, "K-anonymity", "L-diversity", "T-closeness", Attaque de Similarité, NCP.

Glossary

AES	Advanced Encryption Standard
ARX	Powerful Data Anonymization
CIA	Confidentiality, Integrity and Availability
COPPA	Children's Online Privacy Protection Act
COVID-19	COronaVIrus Disease appeared in 2019
CPGEN	C-mixture based Privacy GENetic
DAC	Discretionary access control
EEA	European Economic Area
EHRs	Electronic Health Records
EMD	Earth Mover's Distance
EMRs	Electronic Medical Records
EU	European Union
FE	Functional Encryption
FPE	Format Preserving Encryption
FTC	Fair Information Practices
GDPR	General Data Protection Regulation
HDFS	Hadoop File System
HE	Homomorphic Encryption
HIPAA	Health Insurance Portability and Accountability Act
HITECHA	Health Information Technology for Economic and Clinical Health Act
ICT	Information and communications technology
ImSLD	Improved scalable l-diversity
IoT	Internet of Things
KL	Kullback-Leibler
LBS	Location-Based Service
MAC	Mandatory access control
MBF	Maximal-Bucket First
MNSACM	Multi numerical sensitive attributes clustering method
MPA	Multi-Party Computation
MSB	Multi-Sensitive Bucketization
MSB-KACA	Multi Sensitive Bucketization K-Anonymity Clustering Attribute Hierarchy algorithm
NCP	Normalized certainty penalty
NIST	National Institute of Standards and technology
NP	Non-deterministic Polynomial-time
OECD	Organization for Economic Co-operation and Development
PCI DSS	Payment Card Industry Data Security Standard
PETs	Privacy Enhancing Techniques

PII	Personally Identifiable Information
PIPEDA	Personal Information Protection and Electronic Documents Act
PKI	Public Key Infrastructure
PM-HCA	Proximity measurement of hierarchical categorical attributes
PTFG	Privacy Technology Focus Group
QI	Quasi-Identifier
RBAC	Role based access control
SMC	Secure Multiparty Computation
TCS	T-Closeness Slicing
T-MSN	T-closeness applied on multiple sensitive numerical attributes
TDS	Top Down Specialization
VC	Verifiable Computation
V-COLD	Variable distinct L-diversity algorithm applied on highly correlated attributes
VD	Variational Distance
VGH	Value Graph Hierarchy
V-KAN	K-anonymity technique without prior value of the threshold K
V-KLT	Variable K-anonymity, L-diversity and T-closeness

Contents

List of Figures

List of Tables

General Introduction

Thesis context and problem statement

Nowadays, data is daily generated at an extraordinary rate from diversified sources: web sites, social networks, connected devices (IoT), etc. These Data could belong to different sectors such as government, business, marketing, education, health, etc [120]. This incessant growth of stored data gives rise to a remarkable interest in data analysis because of the possibilities it can offer to organizations. For instance, data gathered from e-commerce sites may profile customers based on their prior searches and purchases [72]. In another side, healthcare plays a significant role in the society. Improving the efficiency, accuracy, and quality of people's healthcare by using recent information and communication technologies to collect and process patient's data is currently a common objective of all the governments around the world [77]. In fact, healthcare facilities such as hospitals, doctor's offices, and clinics are progressively swapping from paper records to Electronic Health Records (EHRs). Thus, related to various studies performed in various countries like US, Canada and Europe, the use of EHRs continues to increase because of their multiple benefits such as improving the quality of healthcare and minimizing costs [96].

Currently, people around the world are facing an unusual global health emergency due to the COVID-19 pandemic. Fortunately, electronic and mobile technologies have played a significant role in improving the quality of patient's care (for example, by allowing distant access to healthcare services) and accelerating the medical research to fight COVID-19. However, aren't these technologies presenting huge drawbacks, including the violation of patients' privacy? Indeed, the exchange of patient's data between various parties such as healthcare organizations, Cloud providers or even intermediate services like laboratories, pharmacies causes new opportunities for intruders to have unauthorized access to patients' data. In addition, publishing data for research or analysis purposes without removing PII will lead to privacy breach. Nevertheless, even though the identifier is removed, the identity information of a certain user could be detected by using auxiliary information [141]. Consequently, sensitive data related to patients become more vulnerable. So, patients become more concerned about their sensitive information especially when they do not know who may have access to their personal data after being exchanged or downloaded and also how their data will be used or shared [142]. At this level, an important question arises: How can third parties guarantee privacy protection of patients while still leaving them at ease in disclosing their personal data?

For research and analysis goals, it is necessary to limit the risks of disclosure to a reasonable level, while retaining useful information. That is why, effective approaches and algorithms are needed to ensure patients' privacy while preserving sufficient data utility. To this purpose, various privacy-preserving approaches have been suggested in the literature, some of which focus on protecting private data using cryptographic algorithms, others on enforcing access control and many others on anonymization techniques. The anonymization

is an operation used to prevent the identity of a person from being linked to other information. Depersonalization, masking or even obfuscation are other forms of anonymization. An anonymization-based approach is considered efficient when it is impossible to deduce the initial data from the anonymized one even by using a mathematical process [31].

In this thesis, we focus more on three relevant anonymization techniques belonging to *Generalization-based* approaches which are *K-anonymity*, *L-diversity* and *T-closeness*. It is well known that there exist two main types of attributes in the literature, *QI* and sensitive attributes. Certain researchers have found that *QI* attributes could be a threat; for instance, with the "Date of birth", "Zip code", and "Gender", almost 60% of individuals could be identified [72]. Through *QI* attributes, an attacker could identify the majority of the population by performing the Linkability attack and then trying to gather the targeted victim's sensitive attribute values. Besides, with the huge attention and importance given to sensitive data, we tried to develop an algorithm which deals with sensitive attributes using the principle of *L-diversity*. However, although *L-diversity* technique is a good anonymization technique, it cannot resist against the Similarity attack. That is why, we thought about developing combination of algorithms that could ensure privacy, preserves data utility and treats both *QI* and sensitive attributes.

Contributions

This thesis mainly aims to ensure privacy protection while preserving data utility through anonymization techniques. Specifically, we propose three major contributions:

Contribution 1: A classification of anonymization techniques

As a fundamental step, we start by making a general classification of the main anonymization techniques ensuring privacy. These techniques belong to two main categories including cryptographic and non-cryptographic techniques. Besides we compare the presented anonymization techniques according to three relevant criteria which are data completeness, confidentiality and data accuracy [31]. Explicitly, we aim in this first contribution to show the importance of using anonymization techniques in order to ensure privacy. The proposed classification gives the user the possibility to choose the suitable anonymization technique to its specific case.

Contribution 2: A technique to anonymize *QI* attributes

In this contribution, we suggest a privacy preserving technique using the principle of *K-anonymity* technique. Over the years, *K-anonymity* has been treated with great interest as an anonymization technique ensuring privacy when dealing with *QI* attributes. Despite the fact that many algorithms of *K-anonymity* have been proposed, most of them admit that the threshold K of *K-anonymity* has to be known before anonymizing the data set. In our work, a novel way in applying *K-anonymity* for *QI* attributes is presented. It is a new algorithm called *K-anonymity* without prior value of the threshold K [27].

Contribution 3: Techniques to anonymize sensitive attributes

In this contribution, we deal with sensitive attributes. For the reason that sensitive attributes are generally separated, the correlation between these various attributes is lost. In this thesis, we examine the preservation of the data utility by reducing the correlation

loss. Next, we show that the fact of applying the principle of *L-diversity* technique on highly correlated attributes through a vertical partitioning preserves well the data utility and in the same time ensures privacy [30]. Thus, we proposed a new algorithm called "Variable distinct *L-diversity* applied on highly sensitive correlated attributes". Although *L-diversity* is a good anonymization technique, it does not resist against the Similarity attack. That is why, by combining *L-diversity* and *T-closeness* principles a resulting anonymization technique is essential to address the proposed *L-diversity* technique limitation [28].

Despite the fact that many algorithms for *T-closeness* have been proposed in the literature, many of them admit that the threshold T of *T-closeness* is set to a fixed value. In this thesis, we prove that applying *T-closeness* principle for both single and multiple sensitive numerical attributes without fixing the threshold T is more efficient to ensure privacy. For this, we proposed an algorithm called "Variable *T-closeness* for sensitive numerical attributes" which was extended to be able to treat multiple sensitive attributes [26], [29]. Furthermore, we indicate that categorical sensitive attributes must be treated by breaking the similarity between categorical values. Thus, we suggested an algorithm called "Proximity Test for Sensitive Hierarchical Categorical Attributes" [28]. The last algorithm is applied on both a test table and a real fairly large data set. In addition, we evaluate the percentage of information loss through the use of the *NCP* criterion.

Thesis organization

This thesis is organized in four main chapters as follows:

We first start by defining the basic notions in chapter 1. Essentially, we give some privacy and privacy policy definitions. Besides, we highlight the importance of privacy preferences while presenting various privacy laws. Then, we show the privacy concerns in E-health and the different privacy preserving solutions. Afterwards, we present the difference between pseudonymization and anonymization and the case where pseudonymization could be more preferable than a full anonymization. Moreover, we present our general proposed classification of various anonymization techniques in the form of a hierarchy. These techniques correspond to two main approaches including cryptographic and non-cryptographic while the non-cryptographic based approaches category is divided into *Generalization-based* and *Randomization-based* approaches. We conclude the chapter by citing the different cases where a certain technique could be applied by taking into consideration the type of the treated attribute and the reason behind the anonymization.

In chapter 2, we focus on three main techniques related to non-cryptographic based approaches including *K-anonymity*, *L-diversity* and *T-closeness* techniques. We thought to focus more on these techniques because they are able to deal with huge data sets. Besides, the combination of these techniques will give us the opportunity to treat both *QI* and sensitive attributes. These techniques are corresponding to *Generalization-based* approaches in our proposed classification of anonymization techniques. Concerning the *K-anonymity* principle, we are focusing on the *K-anonymity based on Generalization* because generally while applying *K-anonymity* technique, we may generalize the data as well as we may proceed by removing some or all the characters of a *QI* value in the data set. The aim of this chapter is to present related work belonging to the *Generalization-based* approaches already cited and to discuss every technique separately.

Chapter 3 presents our proposed algorithms related to non-cryptographic based approaches. We first deal with *QI* attributes by proposing an algorithm using the principle of *K-anonymity* without prior value of the threshold K called *V-KAN*. Then, we treat sensitive attributes by employing the principle of *L-diversity*. In this context, we proposed a variable distinct

L-diversity algorithm applied on highly sensitive correlated attributes entitled *V-COLD* to ensure privacy and also to preserve the data utility. However, the fact that *L-diversity* technique does not resist against the Similarity attack prompted us to think about addressing this limitation. Thus, we proposed a variable *T-closeness* algorithm intended to treat sensitive numerical attributes. This algorithm was extended to be able to deal with multiple sensitive numerical attributes called *T-MSN*. Finally, we suggested an algorithm using the principle of *T-closeness* and dealing with categorical attributes called *PM-HCA* in order to prevent the adversaries from the Similarity attack. Moreover, we have discussed each algorithm separately.

The last chapter is as a validation of our new hybrid technique in e-health context called *V-KLT* which is based on *K*-anonymity, *L*-diversity and *T*-closeness variable techniques. We first apply *V-KAN* and *V-COLD* algorithms on the *QI* and sensitive attributes existing in our treated test table respectively. Then, we present our experimental results showing the resistance against the Similarity attack using both *T-MSN* and *PM-HCA* algorithms. The final step in this chapter evaluates the final anonymized table through the use of the *NCP* criterion. Besides, we made a comparison between applying the *NCP* criterion on a test table and a fairly large one. In addition, we discussed the results of each step in chapter 4 separately. Finally, we conclude the thesis by a general conclusion reminding the main treated problematic and the suggested contributions. Also, we present some perspectives and possible future research trends.

Chapter 1

Classification of Anonymization Techniques

1.1 Introduction

In this chapter, we will first present the privacy issues by giving privacy definitions and the privacy concerns in e-health and also some privacy enhancing solutions. Then, we will highlight the difference between the pseudonymization and the anonymization approaches. After that, we will give our proposed general classification of various anonymization techniques belonging to both cryptographic and non-cryptographic categories.

1.2 Privacy Issues

This section gives some definitions of privacy, privacy policy, privacy preferences and major related concepts. In addition, it highlights the privacy concerns in digital e-health and presents some privacy enhancing solutions.

1.2.1 Privacy Background

From the very early ages, researchers and philosophers indicated the importance of understanding the meaning of privacy concept. For instance, Solove in [111] argued that the most striking thing about the right to privacy is that nobody seems to have any very clear idea what it is. However, first and before talking about privacy protection, this concept has to be properly understood by users and any organization willing to use or disclose personal data on the internet. Explicitly, Information security means protecting information and information systems from unauthorized access, use, disclosure, disruption, modification or even destruction so that the confidentiality, integrity and availability of information are maintained [37]. In the other hand, privacy ensures that user's data are stored, used and fairly disclosed according to the data owner's preferences.

1.2.1.1 Privacy definitions

In the healthcare field, the National Committee for Vital and Health Statistics describes in [82] the differences between and among privacy, confidentiality, and security this way: "Health information privacy is an individual's right to control the acquisition, uses, or disclosures of his or her identifiable health data. Confidentiality, which is closely related, refers to the obligations of those who receive information to respect the privacy interests of those to whom the data relate. Security is altogether different. It refers to physical, technological, or administrative safeguards or tools used to protect identifiable health data from unwarranted access or disclosure". In the following, some privacy definitions are given for better understanding:

- **Definition 1:** "Privacy is a fundamental human right, enshrined in numerous international human rights instruments. It is central to the protection of human dignity and forms the basis of any democratic society" [51].
- **Definition 2:** "The concept of privacy relates to individual autonomy and each person's control over their own information, this includes each person's right to decide when and whether to share personal information, how much information to share, and the circumstances under which that information can be shared" [101].
- **Definition 3:** "Privacy means controlling all information about oneself, including protecting identity (anonymity), personal information, and information about personal activity" [12].

These definitions link privacy with the person's desire of keeping his sensitive data secret, safe and under control without the interruption of others. Therefore, owing to its strong impact on individual's behaviors, privacy is considered a fundamental right that must be accorded to all people from all ages. Each person has the right to keep his personal information private and has the ability to control the disclosure of that information. Patient's medical histories, family secrets, shopper preferences, vote's results are all examples of sensitive data that require a high-level of privacy protection [32]. How to ensure such right and desire of privacy is far from being simple. At first, each organization that stores or processes personal data should make public its privacy policy to allow verification of compliance with related laws and regulations.

1.2.1.2 Privacy policy

The term privacy policy is well known by researchers, the computer science community and anyone who has ever browsed the internet. But, what is the real explanation of this term, particularly, in e-health ? Does it indicate how the sharing, collection and management of medical data is done to reassure patients ? or is it just used to show how the privacy policy is compliant with privacy laws and regulations? Hence, and in order to clarify the real meaning and purpose of this concept and its importance in e-health, the authors in [32] selected the following three definitions:

- **Definition 1:** "A privacy policy is a written, published statement that articulates the policy position of an organization on how it handles the PII that it gathers and uses in the normal course of business. PII is any information that can be used on its own or with other information to identify, contact, or locate a single person, or to identify an individual in context" [84].

- **Definition 2:** "A Statement that declares a firm's or website's policy on collecting and releasing information about a visitor. It usually declares what specific information is collected and whether it is kept confidential or shared with or sold to other firms, researchers or sellers" [14].
- **Definition 3:** "A privacy policy is a statement or a legal document (in privacy law) that discloses some or all of the ways a party gathers, uses, discloses, and manages a customer or client's data. It fulfills a legal requirement to protect a customer or client's privacy" [10].

Referring to the above definitions, we conclude that ensuring privacy requires that sharing, collecting and managing sensitive information are regulated using privacy policies. These statements or legal documents contain some or all the ways a party manages user's data i.e. what information is collected, how it is collected and under which circumstances this information is used or stored. In the health context, there is a need of privacy policies that allow healthcare providers to access all the relevant information (generally, in a more permissive and timely manner than other contexts) and to share patient's health information with other health providers or relevant stakeholders to make well informed decisions. In critical situations, a healthcare provider should even be able to override the patient's preferences with regards to data sharing [32].

1.2.1.3 Privacy preferences

Patients have the right (as a human right) to decide on themselves to whom disclose their data and under what circumstance and healthcare providers should normally show in their privacy policies that they respond to their patient's preferences [32]. Fortunately, with the recent advance in privacy preserving solutions and the new important updates made in some privacy laws, more and more healthcare providers in developed countries use innovative Information and Communications Technology (ICT) solutions that allow the patient to manage the access to his personal data or to specific types of sensitive information. Therefore, the patient is gradually moving away from a passive to an active role. Yet, even with these patient-centric solutions in place, more efforts are needed to make patients more active regarding the decisions made concerning the usage, disclosure and management of their sensitive medical information. It is also important to give patients the means to verify if their privacy preferences will be taken into account when exchanging or sending their private health information to other health actors. At least, they would be able to easily verify if the privacy policies of these actors are both convenient (according to their preferences) and compliant with relevant privacy laws [32]. Hence, we believe that patients have all the right to express their privacy preferences and since their health conditions or IT literacy do not always allow them to perform this task, researchers and policy makers have to help patients expressing their privacy preferences in an easy and simple manner [32].

1.2.1.4 Privacy laws

The terms of "privacy law", "privacy regulations", "privacy directive" and "privacy act" could be used differently according to the concerned country or region. Yet all of them aim to assure that citizen's PII is properly protected. In the healthcare sector, the privacy laws play an important role in protecting patient's privacy while allowing the flow of health information needed to provide and promote high quality health care and to protect the public's health and well-being [8]. A privacy Law could be defined as a statute that protects a person's right

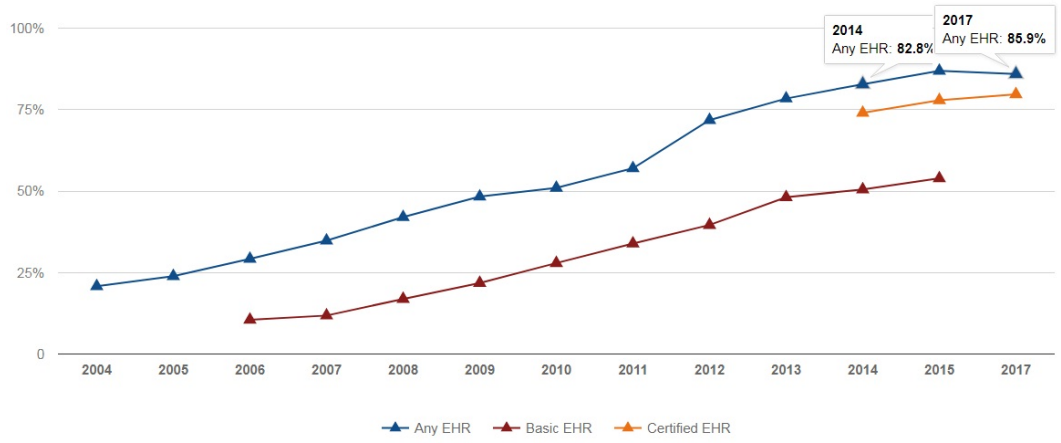

Figure 1.1: Office-based Physician Electronic Health Record Adoption

to be left alone, and governs collection, storage, and release of his or her financial, medical, and other personal information [14]. A privacy Regulation is a kind of rule regulating how certain activity, behavior or data should be protected. Regulations can define two things; a process of monitoring and enforcing privacy legislation and a written instrument containing rules that have privacy law on them [73]. A privacy Act is an act that protects a person against the unauthorized use of personal data by any government agency. For instance, the United States, unlike many other countries, does not have an overarching privacy law that applies to all types of personal information, including health information. However, it has a general health privacy law with broad application that may be extended to e-Health. One of the most important federal laws is the Health Insurance Portability and Accountability Act (HIPAA) which were created to improve the efficiency and effectiveness of the health care system, by encouraging the development of a health information system by establishing requirements and standards and for the electronic transmission of certain health information. In the European Union, the Data Protection Directive 1995/46/EC sets up a regulatory framework aiming to strike a balance between a high level of protection for the privacy of individuals and the free movement of personal data within the EU by setting strict limits on the collection and use of personal data [22].

1.2.2 Privacy concerns in E-health

The concept of digital health, also known as e-Health appeared to refer to the use of ICTs in the healthcare environments. E-health, is seriously improving the quality of patients' care. In fact, with the release of online healthcare applications, more and more healthcare facilities such as hospitals, Clinics and medical offices are moving from paper records to Electronic Medical Records (EMRs) or Electronic Health Records (EHRs).

Actually, the use of EHRs or EMRs keeps increasing because of their various benefits like improving the quality of healthcare and reducing its related costs. Around 2006, 9/10 physicians in the United States manually updated their patient records and stored them in color-coded files [19]. As shown in Figure 1.1 in [80], by the end of 2014, about 8/10 (83%) of office-based physicians were using EHRs, up from 61% during a year before. After that, as of 2017, almost 9/10 (86%) of office-based physicians had adopted any EHR, and almost 4/5 (80%) had adopted a certified EHR. Thus, electronic health records are changing rapidly.

In particular, patients are increasingly involved in the management of their health record

especially when using mobile applications or personalized services provided by healthcare institutions because of the emergence of m-health and more general e-health paradigm. With the adoption of EHRs, physicians can quickly and easily check patient's medical histories to obtain information about their illnesses in order to reduce any risk of complications. Nevertheless, advancements in information and communication technologies have conducted to a situation in which patient's health data faces new threats to privacy [123]. The fact of sharing patient's data between various parties like hospitals, laboratories or even pharmacies gives chances for attackers to an unauthorized access and identity violation. In addition, sensitive health data requires higher level of privacy protection against various threats compared to other kinds of data. Privacy is frequently confused with security. Security deals with the CIA triad composed by Confidentiality, Integrity and Availability, whereas privacy is ensured when it is possible to hide the real identity of the person [98]. From a legal point of view, several laws for protecting personal information were proposed. For example, the Privacy Act 1988 which was introduced to protect the individual's privacy and also to control the manner in which Australian Government agencies and organisations handle personal information. In addition, the Data Protection Directive 95/46/EC which is promulgated in October 1995, it is a European Union (EU) directive that controls how personal data are handled within the EU. Besides, the Personal Information Protection and Electronic Documents Act (PIPEDA) that became a law on 13 April 2000. The act was intended to encourage consumer trust in electronic commerce and to reassure the EU that the Canadian privacy law was able to make the personal information of European citizens safe. In this context, the United States established separate laws and operates according to the HIPAA in 1996, Children's Online Privacy Protection Act (COPPA) in 1998 and finance (Gramm-Leach-Bliley) in 1999. Otherwise, the privacy principles deal with the fundamental rules on how organizations should treat personal information, for example, the Fair Information Practices (FTC) 2000 and Organization for Economic Co-operation and Development (OECD) Privacy Principles in 2010 [9]. A latest law was proposed in EU, called General Data Protection Regulation (GDPR) which became efficient in May 2018. This law concerns all the organizations of the EU, the European Economic Area (EEA) as well as organizations belonging to other countries processing European citizen's data [45], [128]. However, standards and regulations such as HIPAA, The Health Information Technology for Economic and Clinical Health Act (HITECHA) in 2009 and ISO 27779 only offer a baseline of protection for health information instead of ensuring security and privacy of such information [20].

Privacy plays a significant role in healthcare to the point that revealing sensitive patient health data may also lead to reveal other details related to that patient's life [75]. Hence, we talk about medical privacy when patients are allowed to keep their medical records from being disclosed. Patients fear that revealing personal information will affect their insurance and thus, their situation would be embarrassing [2]. However, mobile technologies are very useful especially with chronic diseases, empower elder and even pregnant women. These technologies could remind people to take their medication, enhance health outcomes and medical system effectiveness [16].

It seems that the patient's interest in electronic and mobile technologies is constantly increasing. Based on a survey done by Harris Interactive and ARiA Marketing, patients are more and more interested in exchanging information with their doctors, participating in online communities to receive information about their illnesses and getting personalized medical alerts related to their medical histories from their doctors [44]. Actually, mobile technologies make the communication between patients and their doctors too easy as long as mobile devices are small and easy to hold in comparison with computers. According to Dr. Ted Eytan, the medical director of the Kaiser Permanent Center for Total Health in

Washington: "The idea of storing information on a web site and forwarding it to your doctor seems to make more sense on a mobile phone, because it's something you hold that's yours, that you can "share" with someone" [34]. Then, since patients may have a cell phone, they can take an active role in their own healthcare.

Even though mobile devices are simple to hold and easy to use, they are prone to be stolen or damaged. Besides, sensitive data is often subject to unauthorized use or violation due to the diversity of mobile applications and the way of using mobile devices in various situations such as checking and sending emails, browsing the internet or even social networking. Consequently, there is a need to techniques and practical tools to better control the violation of health data and to ensure patient's privacy. Various recent privacy preserving approaches have been suggested. Most of these solutions are based on cryptography, anonymization or access control techniques.

1.2.3 Privacy Preserving Solutions

Solutions that are specifically designed to protect privacy are often referred to as Privacy Enhancing Techniques (PETs). According to Borking and Raab in [11], PETs could be described as "A coherent system of ICT measures that protect privacy by eliminating or reducing personal data or by preventing unnecessary and/or undesired processing of personal data, all without losing the functionality of the information system".

Among all kinds of data, sensitive health data require a higher level of privacy protection in order to resist against unauthorized adversaries and disclosures. For this end, a number of mechanisms and privacy-preserving approaches have been suggested. Nonetheless, there are not much research works related to privacy preserving healthcare data presenting frameworks that offer a practical view for real life application [49].

Privacy protection is the responsibility of all the stakeholders of e-health including patients, healthcare service providers and also any type of organizations implicated in patient's care. As stated earlier, privacy policies and preferences play an essential role in ensuring patient's privacy. These policies have to comply with local regulations and laws. In fact, privacy policies alone are not sufficient to protect privacy [32].

Several recent privacy-preserving solutions use access control techniques such as Role based access control (RBAC), Mandatory access control (MAC) and Discretionary access control (DAC). Cryptography is evenly considered a crucial mechanism used to protect medical information principally during data transmission and storage. In cryptography, a public key infrastructure (PKI) is a set of roles, policies, and procedures necessary to create, manage, use, distribute, store, and even deny digital certificates [74]. The goal of a PKI is to manage public-key encryption and to ensure the security of electronic transfer of information for a range of network activities like confidential email, e-commerce, internet banking. In addition, some developed encryption-decryption algorithms such as Advanced Encryption Standard (AES) have been broadly used. AES uses a variable-length block cipher of 128 bits, and could run on various platforms from mainframes to desktop computers [118].

Another way to ensure privacy is Data anonymization. It encrypts or removes personally identifiable information from data sets to prevent adversaries from violating the individual's identity [86]. Data anonymization is defined by The Privacy Technology Focus Group (PTFG) as a "Technology that converts clear text data into a non-human readable and irreversible form, including pre-image resistant hashes (e.g., one-way hashes) and encryption techniques in which the decryption key has been removed." [119]. Nevertheless, the anonymization process may lead original data to be removed or modified with the risk of information loss (Data utility may be not preserved). Besides, several algorithms used for data de-identification are

not efficient because in many cases the resulting anonymized data set can be linked with public databases and then the user's identity could be easily revealed [110]. Additionally, most of privacy breaches happen inside the healthcare points of care. Consequently, effective and practical tools to protect medical sensitive data from external and internal threats are always needed. In the following, we present the difference between the pseudonymization and anonymization approaches.

1.3 Pseudonymization Vs Anonymization

Both pseudonymization and anonymization based approaches protect the identity of a person. Although anonymization is the most used, there are some cases where pseudonymization is more suitable since it keeps certain information unencrypted which may cause some problems to the person's life if they are hidden.

1.3.1 Pseudonymization-based approaches

Pseudonymization is intended for ensuring privacy even if it does not represent a fully anonymization and it can be considered as a data minimization measure [45]. Pseudonymization replaces an identifier with a randomly generated one called pseudonym and it generates several identification keys in order to create a connection between distinct information related to individuals [25]. In the following, the most used pseudonymization-based techniques as mentioned in [136].

- **Encryption with secret key:** Since the data set still contains personal data, the owner of the key can trivially identify each data subject by decrypting the same data set. [128].
- **Hash function:** It is a one-way function that returns an output data with fixed size from an unfixed input size. However, this function has the possibility to replace the range of known input values in order to deduce the right value for a special record. Hash functions are generally designed to be relatively fast in the calculation processes even if they fail against Brute Force attack.
- **Keyed-hash function with stored key:** It is a special hash function using a secret key as an additional data entry. A data controller can apply the function on the attribute by employing this secret key. However, when an adversary applies the function without knowing the key, then this technique becomes much harder and impractical since the number of possibilities to be analyzed is large.
- **Deterministic encryption:** It may be assimilated to a technique that chooses a random pseudonym number for every attribute's value in the data set. This pseudonym is then removed from the matching table in order to reduce the risk of Linkability. By using this pseudonymization-based technique, it will be computationally difficult for an adversary to decrypt the function because he or she has to test every possible key every time the tested one is not correct.
- **Tokenization:** It is particularly employed in financial industry to substitute the numbers of an ID card by other values in order to reduce the usefulness of the data for the

adversary. This technique applies unidirectional encryption mechanisms through an indexed function of random produced numbers which are mathematically not determined from the source data.

Generally, pseudonymization does not have a negative effect on the data mining process [45]. However, the reversibility of pseudonymized data could be very significant. For example, in the context of clinical drug trials, it is important that patient's pseudonymized trial data could be reversed if necessary in order to inform the patients about a medically undesirable event. That is why, fully anonymized data in this context might be dangerous and irresponsible. Therefore, in most of the time, a fully anonymization is needed since the main goal is privacy protection. In the following, we will present the general anonymization architecture and we will mention the required tasks to get an anonymizer system.

1.3.2 Anonymization-based approaches

In order to keep sensitive shared data protected, data sanitization is often made before the distribution and analysis processes. And, when the intention of sanitization is privacy protection; then, anonymization, or de-identification based techniques are the most used. Anonymization-based approaches maintain the identity of records existing in the published data set protected against Identity Disclosure attacks by applying some anonymization-based techniques belonging to *Generalization-based* approaches or even *Suppression-based* ones, etc [13].

In fact, the goal of anonymization-based techniques is providing a balance between preserving data utility and ensuring privacy [134]. There are many open source tools made for anonymization such as Powerful Data Anonymization (ARX), the cornell anonymization and hadoop anonymization toolkit [102]. Anonymization is a process used to prevent a person's identity from being connected with other information. Depersonalization, masking or even obfuscation are other forms of anonymization. Furthermore, the anonymization process is considered effective if it is impossible to deduce the original data set from the anonymized one by using a mathematical process [57].

When protecting privacy, the anonymization process is expected to enable big data tools to analyze the data in meaningful ways. So, as shown in Figure 1.2, once the data is anonymized, it can be safely moved to Hadoop File System (HDFS) based storage for example, where it would be disposable for analysts to examine the output. The architecture gives also the possibility to identify and reconstruct the original data set in order to control the unaccustomed behaves of some persons [102]. According to Krizan et al. in [57], an anonymizer system is expected to obey a number of requirements, mainly security and speed.

- **Security** should be very strong to make a backup of the original information, fairly difficult or even impossible. Security is not strong enough even if the used keys are adequately protected, thus, anonymization techniques should be involved. Otherwise, the possibility to apply the masking only on authenticated users would improve the security of the system.
- **Speed** is measured by the number of hidden records in a unit of time. The speed of the used technique gives the ability to determine if the system is able to work online or offline only. Thus, any process should be done within a reasonable time in all cases.

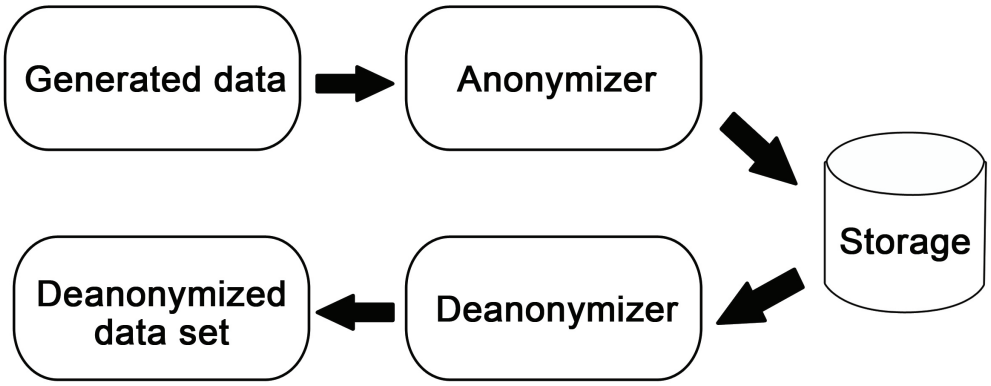

Figure 1.2: General anonymization architecture

In the following, we will present our proposed general classification of cryptographic and non-cryptographic anonymization techniques.

1.4 Classification of anonymization techniques

Before we present our proposed classification, it is necessary to give an insight on the classifications made in some works dealing with anonymization to ensure privacy. For Li and Zhang in [65], the anonymization could be achieved by using cryptographic or non-cryptographic techniques. According to Krizan et al. in [57] and the working party in [136], anonymization techniques are divided into two categories; the first one is called *Randomization-based* approaches while the second one is called *Generalization-based* approaches. Otherwise, *Encryption*, *Randomization-based* approaches, *Bucketization* and *K-anonymity* constitute the classification adopted by Patil and Ingale in [85]. Besides, authors in [98] assume that the anonymization process could be realized by using cryptographic techniques, *K-anonymity*, *L-diversity*, *T-closeness* and *Differential Privacy* techniques. Venifa Mini and Angel Viji in [128] proposed an architecture that ensures privacy against potential security breaches in the cloud through the anonymization of encrypted data. Furthermore, the classification of anonymization techniques made by Wang and Li in [133] includes *Generalization-based* approaches, *Suppression-based* approaches, *Bucketization* and *Perturbation*. In this thesis, we will divide the anonymization techniques into two main classes, cryptographic and non-cryptographic approaches as shown in Figure 1.3.

Figure 1.3 represents a diagram of anonymization techniques helping to ensure privacy. Table 1.1 represents the anonymization techniques already mentioned in Figure 1.3 to evaluate the anonymized published output through data completeness, confidentiality and data accuracy.

One of the components of data quality is data Completeness which indicates the degree of availability of all required data in the data set. Data completeness would be measured by the percentage of missing data entries. All the cryptographic techniques existing in Table 1.1 comprising *HE*, *VC*, *MPC*, *FE* and *FPE* techniques do not delete the data during the anonymization process. According to the Table 1.1, *K-anonymity based on Generalization*, *L-diversity*, *T-closeness*, *Permutation*, *Differential Privacy* and *Shuffling* are the only non-

Table 1.1: The evaluation of anonymization techniques.

Categories	Techniques	Data completeness	Confidentiality	Data accuracy
Cryptography-based techniques	*Homomorphic Encryption (HE)*	Yes	Yes	Yes
	Verifiable Computation (VC)	Yes	No	No
	Multi-Party Computation (MPC)	Yes	Yes	Yes
	Functional Encryption (FE)	Yes	Yes	Yes/No
	Format Preserving Encryption (FPE)	Yes	Yes	No
Generalization-based techniques	*K-anonymity based on Generalization*	Yes	No	Yes
	L-diversity	Yes	No	Yes
	T-closeness	Yes	Yes	Yes
Randomization-based techniques	*K-anonymity based on Suppression*	No	No	Yes
	Permutation	Yes	Yes	Yes
	Differential Privacy	Yes	Yes	No
	Substitution	No	Yes	No
	Shuffling	Yes	No	Yes
	Blurify	No	Yes	Yes
	Nulling Out	No	Yes	No
	Character Masking	No	Yes	No

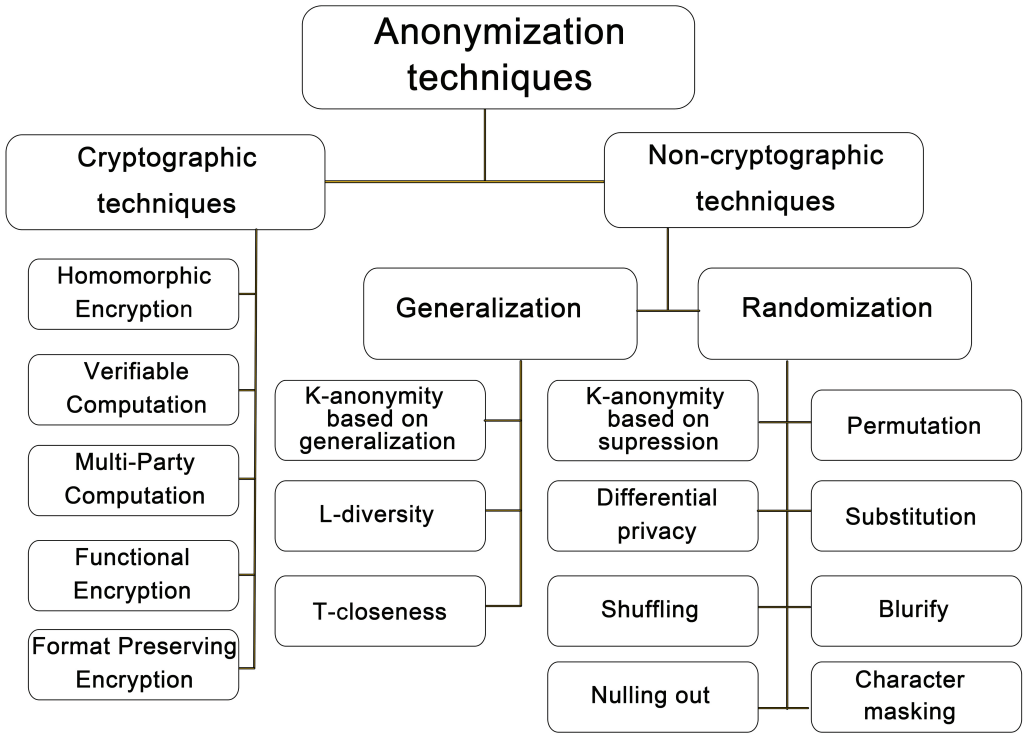

Figure 1.3: Diagram of the proposed classification of anonymization techniques

cryptographic techniques where the data completeness is ensured. In fact, *K-anonymity based on Generalization* technique only substitute values in the data set by more general ones while leaving the number of values in the data set intact. The *L-diversity* and *T-closeness* techniques leave the data as close as possible to its original form. Moreover, the distribution of values using *Permutation* technique remains unmodified. In addition, a copy of the original data set will be maintained when using *Differential Privacy* technique; besides, when employing *Shuffling* technique, the entire data still exists in the anonymized data set.

Regarding the confidentiality criterion, it is ensured when the published anonymized data set does not contain information that could lead to identify a specific person. In fact, the studies show that the confidentiality is not ensured when using *VC* technique. However, it is ensured when using the remaining cryptographic technique from Table 1.1. Concerning the non-cryptographic technique, the confidentiality is guaranteed when using *K-anonymity* and *L-diversity* in the case where the thresholds K and L are high enough respectively. However, this criterion could not be maintained when using *Shuffling* technique if the used algorithm is not appropriate in the anonymization process.

Data accuracy is ensured when using *HE* technique since it enables anyone to compute functions on data while it is encrypted without decrypting it first. Data accuracy is also ensured when employing *MPC* technique since it gives multiple parties the ability to compute a joint function of their inputs. So, everyone knows the correct output of the function, but nobody can get any other information even though some parts may be opponents. However, data accuracy is not ensured when using *VC* technique since it is hard to make the calculation of the proof verification to demonstrate that the results are correct. Besides, data accuracy is not ensured when using *FPE* algorithms because they provide confusion in the output cipher text so that it is computationally indistinguishable from a random process. The remaining cryptographic technique from Table 1.1 is the *FE* technique. This technique may be able to be

useful for researchers if they compute specific functions enabled by the data owner. Concerning non-cryptographic techniques, data accuracy is ensured when using *K-anonymity*, *L-diversity*, *T-closeness*, *Permutation*, *Shuffling* and *Blurify* techniques. In fact, when using *K-anonymity* the real values of individuals still exist in the data set even if they are either generalized or some digits of values are suppressed. Thus, the researchers can make computations on the resulting anonymized data set even if the results are not precise enough. Distinct *L-diversity* technique is the most used among *L-diversity* models where each bucket in the data set contains only distinct values. Thus, computations are possible since all the original values still exist in the anonymized data set. Concerning *T-closeness* technique which is a refinement of *L-diversity* technique, the researchers still have the ability to make computations on the resulting anonymized data set. However, the results taken by researchers through *T-closeness* technique will be less precise than those taken through *L-diversity* technique. Researchers can get useful information when employing *Permutation* technique since the data still exist in the anonymized data set so that several attribute's values can lead to interesting information for statistical goals. It remains *Shuffling* and *Blurify* among non-cryptographic techniques where data accuracy is guaranteed. Since the techniques randomly rearranges values inside one data set's column, researchers can make computations on the resulting anonymized data set because the substituted values are not falsified. The *Blurify* technique involves modifying each value in a column by a particular variance which represents a random percentage of the original value; thus, since the anonymized data set does not contain fake values, the researchers can get useful information. Nevertheless, data accuracy is not ensured when using *Differential Privacy*, *Substitution*, *Nulling Out* and *Character Masking* techniques. In fact, the *Differential Privacy* technique ensures privacy by adding noise to the output of a given function, and consequently it is hard to deduce if a specific record is involved in the data set. Therefore, researchers will find it difficult to make computation on the resulting anonymized data set. For *Substitution* technique, the substituted values can be selected from a given pseudonymization list containing falsified values, then the resulting anonymized data is not useful for researchers. Concerning the two last techniques, *Nulling Out* deletes the sensitive data contained in the data set by removing the whole corresponding column and replacing it with NULL values. The *Character masking* technique is very similar to *Nulling Out* technique since it changes the initial values with a special constant character and replaces certain fields by using a mask character. Consequently, the researchers will get any information form the anonymized data set when using both *Nulling Out* and *Character masking* techniques. In the following, we will discuss every technique mentioned in Table 1.1.

1.4.1 Main Cryptographic anonymization techniques

Nowadays, cloud computing becomes the choice for big data processing and analysis because of its elasticity, resource pooling, low cost and large network access [126]. The cloud providers and tenants might be untrustworthy while storing the data or even while computing on it. Such concerns incite the need for innovative techniques that give a sort of obfuscation to data, such as the use of cryptographic techniques to meet IT security objectives in cloud computing. As cloud storage is progressively used, encryption is becoming essential to ensure both the confidentiality and the integrity of sensitive attributes [7]. Thus, it is crucial to ensure the privacy of individuals. There are various cryptographic anonymization techniques that are particularly applicable to ensure privacy such as identity-based and attribute-based encryption. However, we will focus in this chapter on some techniques that are considered to be the most promising and pertinent including *Homomorphic Encryption*, *Verifiable Computation* and *Multi-Party Computation*. We will also present the *Functional Encryption* technique since

it enables the data owner to supervise and precisely control who can compute on the data. Moreover, a cryptographic anonymization technique called *Format Preserving Encryption* will be highlighted because of its ability to preserve the format of the initial messages.

1.4.1.1 Homomorphic Encryption technique

Homomorphic Encryption (*HE*) is a type of public key encryption techniques. It enables anyone to compute functions on data while it is encrypted without decrypting it beforehand or learning outside information about it [135]. More formally, the encryption scheme is called homomorphic with respect to a function f if and only if there is a corresponding function f' such that the $D_k(f'(E_k(m)) = f(m)$, where D_k and E_k are the decryption and encryption of a message m under the key k respectively [139]. This technique has some weaknesses; in fact, it is inefficient in practical processes and it does not allow the use of different keys for computing on encrypted data [139]. Besides, *HE* is too slow for most of big data analytic applications and it is more expensive in terms of time and space when ensuring privacy compared to common plain-text based systems [126].

Veena in [126] and Wang et al. in [131] mentioned that *HE* is theoretically valid and preserve the confidentiality of the data. However, this cryptographic anonymization technique has more concerns in ensuring the confidentiality instead of the integrity. In the following, we will present an anonymization technique called *Verifiable Computation* that preserves the data integrity.

1.4.1.2 Verifiable Computation technique

Unlike the case of *HE*, the intention now is to be sure that the work is well done without focusing on confidentiality [122]. So, to maintain the integrity of the computation, several methods could be employed without using cryptographic tools. The simplest approach is replication by outsourcing the computation to multiple servers, after that, taking the most common answer as correct. However, *Verifiable Computation* (*VC*) works only when the server failures are not correlated, which is a rare case [47]. The *VC* is a cryptographic anonymization technique that ensures the integrity of externalized computations without doing any assumption on the server failure rate or correlation of failures. In *VC*, accurate operations on big data are redistributed to a third party called prover (receiver) [122]. The data owner provides his or her data with a specification of the desired computation to a certain entity known as the prover which is generally more powerful. After that, this prover delivers the outcome of the specified computation with some "compelling argument" or "proof" in order to demonstrate that the results are correct [122], [139]. Although, it is hard to make the calculation of the proof verification, this anonymization technique allows the data owner to check the integrity of the computation.

1.4.1.3 Multi-Party Computation technique

Multi-Party Computation (*MPC*) is considered more efficient compared to *HE* and *VC* cryptographic anonymization techniques which are still too slow for the majority of big data analytics applications [122]. Thus, there is a necessity to employ another technique that addresses the limitations of the techniques presented above. The *MPC* technique gives multiple parties the ability to compute a joint function of their inputs. So, everyone knows the correct output of the function, but nobody can get any other information even though some parts

may be opponents [127], [129]. Most *MPC* schemes share the incoming data among the participating nodes whereby no set of less than t shares reveals anything about the incoming data. By computing these input shares, the nodes can generate shares of the output which can be reconstructed by the receiver if he/she wants to get the actual output. Since the cloud nodes only see individual shares of the data, they do not get any information about the data or the output computation [139], [41]. This cryptographic anonymization technique gives the opportunity to leverage the presence of honest parties, without knowing exactly the honest party. In addition, no single party can gain information about the data.

1.4.1.4 Functional Encryption technique

Functional Encryption (*FE*) is a public key encryption technique which employs functional and regular secret keys in order to decrypt the data. These keys allow the access to the result of the associated function measured on the data instead of decrypting it [47]. The *FE* is used when a data owner wants to allow certain computations to be made on his/her sensitive data and in the same time desires to maintain a strict control on the data and computations. For instance, suppose that users need to make spam filtering on their encrypted messages without accessing to the content of those messages, then, the most suitable cryptographic anonymization technique is *FE*. This technique fulfils both access control properties of attribute-based encryption and computation on encrypted data. The *FE* technique enables a data owner to precisely specify the functions they can apply and also control who can compute on this data [47].

1.4.1.5 Format Preserving Encryption technique

Format Preserving Encryption (*FPE*) transforms the plain-text related to a specific format into an encrypted text of the same format. For instance, a social-security number is encrypted into another social-security number while both of them have the same format. In 2016, the National Institute of Standards and Technology (NIST) released *FPE* standard schemes which are made with Feistel networks by employing block ciphers [122]. In particular, *FPE* attracts companies with existing software that need to protect their data in accordance with national standards such as HIPAA, Payment Card Industry Data Security Standard (PCI DSS) and also the European Commission's GDPR [122], [7].

1.4.2 Main Non-Cryptographic anonymization techniques

As shown in Figure 1.3, we have two general categories including cryptographic and non-cryptographic anonymization techniques. Here, we tried to divide the non-cryptographic category itself into *Generalization-based* and *Randomization-based* approaches.

1.4.2.1 Generalization-based techniques

The *Generalization* is the first family of non-cryptographic anonymization techniques and it consists of making the attributes of data subjects more widespread by changing their scale or order of size [1]. The *Generalization* could help in preventing Singling out attack, but it is not helpful in all cases. In fact, the *Generalization* needs specific and sophisticated quantitative techniques for preventing Linkability and Inference attacks [136]. Besides, the *Generalization-based* techniques are considered the most common to anonymize a data set in order to ensure privacy in big data [92]. Moreover, by applying the *Generalization-based* algorithms, the data

set's values are replaced with more general ones based on Value Graph Hierarchy, either on Taxonomy tree as mentioned in [54] and [92].

1.4.2.1.1 *K-anonymity* technique Anonymization changes the format of the original data in order to protect personal or private information. In a broad sense; there exist two main attribute types including *QI* and sensitive attributes. The *QI* attributes may lead to identify an individual existing in a certain data set when they are linked with other attributes in external data sets. The *K-anonymity* is achieved when all the records belonging to a set of *QI* attributes cannot be distinguished from at least *K-1* other records in the data set [52], [94]. Moreover, every record in a k-anonymized data set has a maximum probability *1/k* of being identified [94]. In addition, the confidentiality of the published data is better ensured when the value of the threshold K is high enough [27], [54]. The *K-anonymity based on Generalization* protocol works as follows; in the first step, it separates *QI* attributes from sensitive ones. After that, it makes sure that *QI* attributes are generalized according to the threshold K. Later, it verifies if the generalized *QI* are indistinguishable from at least *K-1* other records, then, it inserts them into the anonymized resulted table, otherwise the procedure is repeated [27]. The main idea of *K-anonymity based on Generalization* technique is ensuring that there are identical values within each bucket when making a horizontal partitioning. By applying *K-anonymity* principle, an adversary is not able to detect the real values corresponding to a certain individual. In practice, optimal *K-anonymity* is a Non-deterministic Polynomial-time (NP) hard problem, thus, different approaches come to address the *K-anonymity* limitation like *L-diversity* and *T-closeness* models [57]. Next, the second non-cryptographic anonymization technique called *L-diversity* will be presented.

1.4.2.1.2 *L-diversity* technique Since sensitive attributes include confidential information belonging to a specific individual, they need more protection compared to *QI* attributes [52], [3]. The model of *L-diversity* is introduced to address the shortcomings of *K-anonymity*. The *L-diversity* technique is a form of group-based anonymization and it aims to ensure privacy by partitioning the data sets into several buckets [99]. Thus, the huge scale of big data is minimized in terms of representation [52]. This technique ensures that each sensitive attribute has at least L distinct values within each bucket [1], [13]. The *L-diversity* technique is achieved when the resistance against Background Knowledge attack is possible [94], [107]; besides, the technique can ensure that sensitive attributes would have actually the same frequency [21]. In addition, it is impossible to implement the Inference attack against an L-diverse data set with certitude of 100% [136]. In the literature, there exist three models of *L-diversity* which are Distinct, Entropy and Recursive models [27], [91]. However, the distinct *L-diversity* technique is the most used where each bucket in the data set contains only distinct values. In the following, we present the *T-closeness* technique which comes to address the *L-diversity* technique limitation.

1.4.2.1.3 *T-closeness* technique The *T-closeness* is a refinement of *L-diversity* and aims to create equivalent classes, also called buckets, which look like the initial distribution of attributes in the original table. This technique is efficient when it is necessary to remain the data as close as possible to their original form [136], [99].

When the distance between the distribution of a sensitive attribute in the equivalence class and the distribution of the attribute in the whole table is less than a threshold T, then the equivalence class is called "have *T-closeness*" [98], [91]. Actually, the *L-diversity* technique is not able to resist against several attacks and the most critical one is called the Similarity

attack. Despite the fact that the sensitive values are distinct within each bucket after applying the *L-diversity* technique, the semantic significance of these distinct values may be similar and thus, the information may be disclosed [98]. In the following, we present the different techniques belonging to *Randomization-based* approaches.

1.4.2.2 Randomization-based techniques

The *Randomization* is the second family of non-cryptographic anonymization techniques. It alters the veracity of the data in order to remove the strong relationship between the data and the individual. If the adversary has enough confusion concerning the data; then, he/she can no longer identify an individual [136]. The *Randomization-based* techniques can be applied when collecting the data and also during the data pre-processing steps [93]. However, it will not reduce the singularity of records itself since each record will always be derived from a single data subject. The *Randomization-based* approaches could be combined with the *Generalization-based* approaches in order to produce stronger privacy guarantees [136]. In the following, we will list some techniques related to the *Randomization-based* approaches.

1.4.2.2.1 *K-anonymity* technique The main idea of *K-anonymity based on Suppression* is hiding the values of some attributes by using an asterisk "*" while the concept of *K-anonymity* is ensured with respect to the chosen attributes [27], [68]. The *K-anonymity based on Suppression* protocol is described as follows; first, it separates *QI* attributes from sensitive ones; after that, it substitutes some *QI* attributes by the special character "*". Later, it checks if the suppressed *QI* attributes are equal to the original ones. In the end, they are inserted in the table, otherwise the procedure is repeated. One advantage of using the *K-anonymity based on Suppression* is the impact of substitution of the actual value with "*" which makes unauthorized users confused. However, it becomes impossible to make a backup of the original data set [85]. Besides, it causes a huge amount of information loss and therefore, the data utility is not preserved [143]. The *K-anonymity based on Suppression* process is very close to *K-anonymity based on Generalization* process. The difference is that the first one modifies the values within a column by substituting for example some digits of an attribute with an asterisk and the second one focuses on dispersing the range of values by making them more general.

1.4.2.2.2 Permutation technique The *Permutation* technique can be considered as a particular way of adding noise to data. However, the generation of an exact amount of noise could be a challenging task. Also, changing the values of attributes may slightly not provide enough privacy. Alternatively, the *Permutation* technique modifies the values in the data set by substituting a record with another one. This technique could be applied during the anonymization process between minimum or maximum values corresponding to each consecutive buckets existing in the data set [28]. Such permutation between records will ensure that the range and the distribution of values will remain unchanged, but the link between values and individuals will not. If two or more attributes have a logical or statistical relationship and are swapped independently, then, such correlation will be destroyed. Therefore, it might be important to permute a set of linked attributes while breaking the logical correlation; otherwise, an attacker may identify the permuted attributes and reverse the permutation [136]. In addition, it is essential to isolate sensitive attributes from the original data set and then, apply the permutation process on the corresponding sensitive values in order to take benefit from protecting personal data and to prevent the adversary from retrieving valuable information.

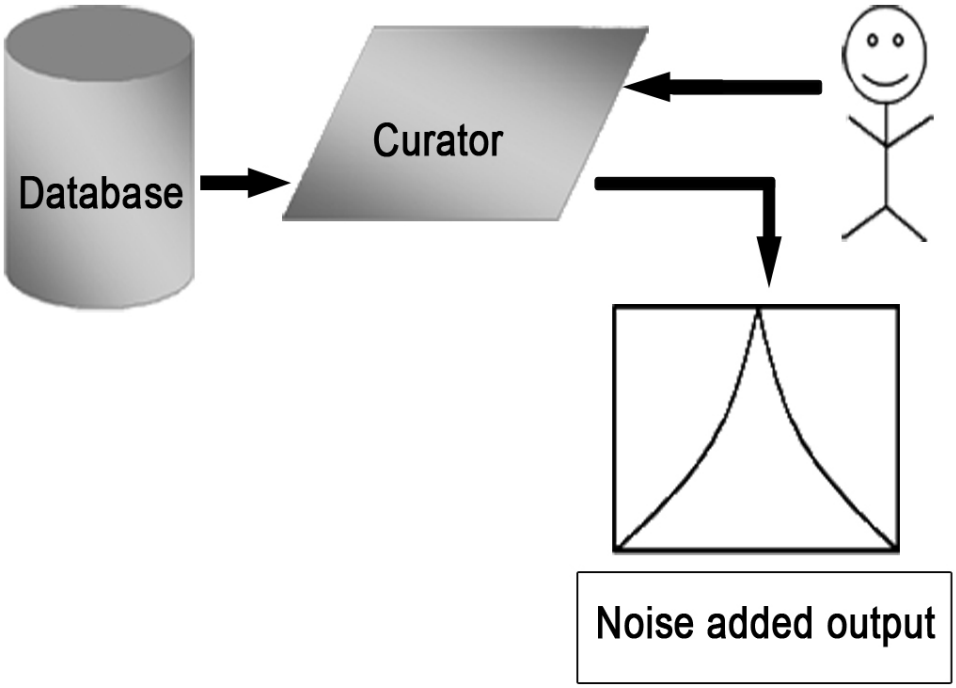

Figure 1.4: Achieving *Differential Privacy* technique [109]

The *Permutation* technique itself is not able to ensure privacy in big data. However, it must be combined with other generalized anonymization techniques such as *T-closeness* technique [28], [136].

1.4.2.2.3 Differential Privacy technique The *Differential Privacy* technique is an anonymization technique which is very suitable for big data as it does not allow the degradation of system's speed. In addition, it is very hard for an attacker to deduce the presence or the absence of an individual. Furthermore, when two different data sets produce almost the same output, then the opponent is unable to determine the real targeted data set [48], [109]. Besides, when the amount of the data becomes important, the *Differential Privacy* technique becomes less efficient since the original data could be estimated from the perturbed data [109]. Figure 1.4 illustrates the required process in order to achieve *Differential Privacy* technique.

As shown in Figure 1.4, the *Differential Privacy* technique is achieved by making a curator between the database and the user/analyst. Once the user or analyst makes a request, it is received by the curator that accesses the impact of privacy by calculating the sensitivity of information; after that, the curator sends the request to the database and waits to receive the clean response [109]. One of the strengths of using the *Differential Privacy* technique is highlighted when the data sets are delivered to authorized third parties in order to reply to a particular request instead of releasing a single data set. Therefore, the *Differential Privacy* technique ensures privacy by adding noise to the output of a given function, and consequently an adversary cannot deduce if a specific record is involved in the data set [17]. However, this technique has some weaknesses, for instance, when making numerous requests; an attacker could be able to identify a particular individual through two or more answers [136]. Besides, the technique is not efficient for privacy preserving when processing a data set including highly correlated attributes [24]. The ability to generate the right quantity of noise to be added to

the output is considered as a challenging issue when using the *Differential Privacy* technique [136].

1.4.2.2.4 Substitution technique The *Substitution* is an anonymization technique which consists of substituting the values within a data set in a random way or even through a list of data similar to the original data set values [6], [108]. The substituted values can be selected either from a given pseudonymization list containing falsified values [57]. The *Substitution* technique is highly adequate when the anonymization intends to preserve the appearance and the feel of current data [108]. However, preparing a considerable amount of substituted information to be accessible for every substitution is a challenging task. For instance, to sanitize names, a fairly extensive random list of names must be prepared; and to sanitize the phone numbers, a huge list of fake phone numbers is needed; nevertheless, the capacity to produce an invalid data is very difficult [78]. In this context, an efficient substitution requires a list of data whose size is equal to or larger than the size of data requiring substitution. Thus, if the data set contains a huge amount of data without having enough substituted data, then, the *Substitution* technique will not be the best technique for anonymization.

1.4.2.2.5 Shuffling technique The *Shuffling* or *Data Swapping* technique is similar to *Substitution* technique but the anonymized data is derived from the column itself. This technique randomly rearranges values inside one data set's column while maintaining the order in the other columns [57]. The *Shuffling* technique is useful when it is essential to keep the aggregated values in their original form. Moreover, it could process columns with a single constraint [108]. The data migrates between lines until there is no possible correlation in the data set. However, there is a risk when using the *Shuffling* technique since the source data still exists. So, an adversary with some significant information can deduce the original data. Another problem is the selection of the algorithm used to shuffle the data; at that time, the data may be simply unshuffled if the adversary could deduce the *Shuffling* algorithm.

For instance, if the *Shuffling* algorithm works by swapping the data existing in a column between every two lines, then, the interested party would not make a big effort to get a backup of the original data set. It is true that the *Shuffling* technique is quick; however, a particular attention should be paid when using a modern and advanced algorithm to randomize the lines in the data set [78]. In fact, it is more secure to apply the *Shuffling* technique on huge data sets because tracing the original values is harder [57]. Although this technique preserves the data integrity, it may be insufficient especially when the amount of records in the data set is tiny.

1.4.2.2.6 Blurify technique The *Blurify* technique gives the opportunity to dissemble the data in a reasonable way. It involves modifying each value in a column by a particular variance which represents a random percentage of the original value [57]. The *Blurify* technique considerably changes the data in order to make it untraceable by any adversary. For instance, a salary details column could have a random variance of ±10 %. Certain values might be higher and others could be lower, but they would not be too far away from their original range [78]. The *Blurify* technique is also called *Number and Variance* technique in some literature researches; besides, it is generally useful when dealing with numeric or birth date data [108]. For example, financial data like salaries are increased or decreased randomly for a particular variance percentage [57], and birth date data could be used through an arbitrary range of

±120 days. Actually, this range conceals the personally identifiable information, whereas the distribution is still preserved [78].

1.4.2.2.7 Nulling Out technique The *Nulling Out* is an anonymization technique that removes the sensitive data existing in the data set by eliminating the whole corresponding column and replacing it with NULL values [108]. The *Nulling Out* technique cannot usually be employed on non-nullable columns of the data set [57]. In general, the test teams require a non-nullable data for their processing. Although, this technique is simple, it is not much desirable and it may not be appropriate if an assessment has to be conducted on the data [78], [108]. For instance, it would be impossible to query accounts of customers if vital information like names and other customer details are null values. The *Nulling Out* technique could also be called as *Truncating Data* technique, and it is helpful in some situations where the data is not very important [78].

1.4.2.2.8 Character Masking technique The *Character Masking* technique is similar to *Nulling Out* technique; it changes the initial value with a special constant character [57]; and it substitutes certain fields by a mask character. This technique strongly hides the contents of the data while maintaining the same format and reports [78]. For example, a credit card number could be viewed such as: *4346 6454 0020 5379* and after applying the masking, the information would appear like: *4346 XXXX XXXX 5379*. The *Character Masking* technique efficiently eliminates a great part of the sensitive content of the record while conserving the appearance and feel. Thus, much care has to be provided to make sure that a sufficient amount of data is masked in order to insure privacy. An operation of masking like: *XXXX XXXX XXXX 5379* would remove much information about the credit card number but the technique would be strong and rapid when dealing with a data in a particular and unchanging format [78]. If many appropriate cases should be treated, then *Character Masking* will be slow and find it difficult to manage and could possibly leave some data without being masked.

1.5 Conclusion

In order to ensure privacy, many anonymization techniques were presented in this chapter belonging to *Generalization-based* and *Randomization-based* approaches. However, choosing an anonymization technique instead of another one is a challenging issue. In the following, numerous cases are cited:

- If the used data set contains only *QI* attributes; then, the most adequate technique is *K-anonymity* based on *Generalization* or *Suppression* or even both of them. Besides the more the threshold K of *K-anonymity* is high the more the technique is powerful.
- If the data set includes only sensitive attributes; then, *L-diversity* technique is suggested among other anonymization techniques. However, since *L-diversity* technique cannot resist against the Similarity attack, *T-closeness* technique must be involved in the anonymization process.
- When the handled data is qualitative; the use of *Generalization-based* techniques is advisable since they do not remove the data from the original data set.

- When the manipulated data is quantitative; it would be recommended to use *Randomization-based* techniques since this type of data involves removing or aggregating variables.
- Since the encryption increases the size of data, the system's speed is degraded; thus, the employment of non-cryptographic techniques is appropriate when the speed of anonymization is not a priority for the user.
- The anonymization techniques such as *Nulling Out* and *Character Masking* may be favored when the data set contains a trivial or powerless content.
- The *Differential Privacy* would be the most suggested technique when the data set contains secret information which needs to be disrupted by adding a particular noise to the data.
- When it is necessary to save a copy of the original data set; then the most suitable techniques to use are *T-closeness*, *Permutation*, *Differential Privacy* and *Shuffling*.
- If the data set includes both *QI* and sensitive attributes; then, the combination of *K-anonymity*, *L-diversity* and *T-closeness* techniques would be useful and will make a balance between ensuring privacy and preserving data utility.

Chapter 2

Generalization-based Anonymization Techniques: State of the Art

2.1 Introduction

In this chapter, we will present the existing works belonging to *Generalization-based* approaches. This chapter will be divided into three parts corresponding to *K-anonymity*, *L-diversity* and *T-closeness* based approaches. In addition, we will make three discussion sections including the advantages and limitations of using the treated approaches.

2.2 *K-anonymity* based approaches

K-anonymity is an anonymization technique that deal with *QI* attributes. We had shown in the previous chapter that *K-anonymity* technique could correspond to both *Generalization-based* and *Suppression-based* approaches depending on the utilization of this technique. In the following, we present some works relating to *K-anonymity* based approaches.

2.2.1 Related work

The *K-anonymity* technique has been extensively studied in the literature in order to ensure privacy when dealing with *QI* attributes. Although, many *K-anonymity* algorithms have been proposed, most of them consider that the privacy parameter K of *K-anonymity* has to be known before applying the anonymization process. For example, Xie et al. in [137] made a combination of diverse techniques to ensure privacy of medical data. They first use *K-anonymity* technique to be sure that the attacker cannot detect the real identity of the individual. Then, the authors use random *Perturbation* technique to randomly change certain information and in the last step they use *Secure Multiparty Computation* (*SMC*) to encrypt the information exchanged between the sites in order to ensure that each participant only has access to his or her own input and output data. Besides, Tu et al. in [124] expect that *K-anonymity* technique is fulfilled when consolidating the same full-length paths shared by users into an anonymized data set. The authors propose an algorithm that grants *K-anonymity*, *L-diversity*, and *T-closeness* of paths through *Generalization-based* approaches while ensuring

the smallest loss of spatio-temporal granularity. Then, the algorithm constantly merges paths from the original mobility data set in order to obtain a new generalized data set composed of all the merged paths able to resist against the Re-identification and the Semantic attacks.

In addition, Natesan et al. in [76] introduce an adjusting learning model to protect the privacy of *K-anonymity* location. They develop a framework that would help users to effectively choose and operate their privacy preferences. Also, the model gives the opportunity to obtain context-based privacy from the anonymizers. Thus, based on the analysis of a set of factors that generally influence the choice of privacy profile, a learning model is built to help users making the right decisions by protecting their location-based privacy. In another study, Forster et al. in [38] propose a generic method to decentralize *K-anonymity* of location data by using a distributed secret sharing algorithm, the approach of location and time specific keys. Forster et al. in [38] describe their method in the context of a privacy-friendly traffic flow analysis system. Moreover, Fei et al. in [35] develop a two-tier schema for the preservation of privacy based on the principle of *K-anonymity* while reducing the cost of protecting privacy. Specifically, the schema divides the users into groups to maximize the level of privacy. Then, in each group, one proxy is chosen to generate fictitious locations and share the returned results from location-based service (LBS) provider. After that, on each group, an auction mechanism is suggested to decide the payment of each user to the proxy as a benefit, which finally fulfills the balanced budget and incentive compatibility.

Later, Kavitha et al. in [54] present a large-scale data anonymization model using the MapReduce framework with Top Down Specialization (TDS) in *K-anonymity*. The "Two Phase TDS" approach includes map and reduce phases. In the first one, the high dimensional data set is divided into small data sets. Those data sets are anonymized in parallel and generate anonymized intermediate results. In the second phase, the intermediate results are joined to each other and then anonymized to produce consistent K-anonymous data. Moreover, Pramanik et al. in [87] propose a novel clustering approach to attain *K-anonymity* technique with minimum loss of information. In this approach, no data records are totally removed. The authors principally do not use *Suppression-based* techniques in the proposed model because the suppression seriously damages both data quality and data utility. The proposed algorithm supports a data publishing process in a way this data will not be deformed. Besides, Kundalwal et al. in [59] proposed a hybrid technique by using two different techniques, a technique based on the restriction of the query set size and *K-anonymity* technique to ensure the privacy of individuals. The first technique is used to prevent the existing sensitive data in the data set from Inference attack, while *K-anonymity* helps in protecting the data set against Linking attack. The authors give a calculation method to estimate the value of the threshold K of *K-anonymity* technique. Furthermore, Arava and Lingamgunta in [5] suggest a systematic approach using an algorithm called "the adaptive *K-anonymity*". The goal is to find the best selection of p seeds to group the records by creating clusters to calculate *T-closeness*. The authors in [5] assume that the proposed anonymization approach gives good results in terms of information loss and execution time.

2.2.2 Discussion

The *K-anonymity* technique is very suitable when dealing with *QI* attributes. It is known that the chance for re-identification is less when the value of K is high [117]. Kavitha et al. in [54] benefit from maximizing the value of the threshold K when using the MapReduce framework with TDS in *K-anonymity*. In addition, it is well known that the isolation and the detection of a person within a group of K users are impossible because the person's attributes are shared by these K users [126], [136]. Thus, when *K-anonymity* principle was used in [137],

the attacker was not able to detect the real identity of the individual as well as when Tu et al. in [124] expected that the *K-anonymity* technique is fulfilled when consolidating the same full-length paths shared by users into an anonymized data set. Moreover, the principle of *K-anonymity* is used in the technique developed by Fei et al. in [35] in order to reduce the cost of protecting privacy. The *K-anonymity* technique has other advantages as it is easy to implement [117] and also the probability of knowing an individual is less than *1/k* by ensuring that each sensitive attribute is hidden in the scale of K groups [126].

However, the *K-anonymity* is inappropriate for numerical sensitive attributes and does not protect the relationship between sensitive attributes in a data set [126]. Besides, it takes a long time processing [117]. Although the K-anonymized table may lead to lose a great amount of information and consequently the utility may be compromised in such a way any query returns minimum of K matches [117], [136]. Pramanik et al. in [87] assume that *K-anonymity* may be helpful to minimize the information loss when using it in their anonymization process. In another side, Kundalwal et al. in [59] used the *K-anonymity* principle to protect the data against Linking attack. Nevertheless, the *K-anonymity* technique does not resist against various attacks such as Background Knowledge, Homogeneity and Inference attacks. In fact, *K-anonymity* is susceptible to Background knowledge attack since it does not ensure privacy against attackers using outside knowledge. In addition, *K-anonymity* cannot resist against Homogeneity attack when there is a diversity in sensitive attributes and therefore an attacker may deduce some values of these sensitive attributes[117], [136]. It remains that *K-anonymity* technique is unable to protect the data against Attribute Disclosure attack even if it gives enough protection against Identity Disclosure attack [126], [117]. In the following, we present the related work concerning *L-diversity* based approaches along with a discussion containing some advantages and disadvantages of techniques using *L-diversity* in their processing.

2.3 *L-diversity* based approaches

The *L-diversity* is an anonymization technique which correspond to *Generalization-based* approaches. This technique only deals with sensitive attributes.

2.3.1 Related work

Generally, the *L-diversity* based approaches aim to ensure privacy when dealing with sensitive attributes. In most of cases, *L-diversity* is applied on a data set while the threshold L is fixed to a specific value. Besides, the degree of correlation between attributes is not considered. For instance, Praveena Priyadarsini et al. in [88] proposed an enhanced *L-diversity* algorithm able to diversify several sensitive attributes without dividing the data set. The proposed algorithm attempts to support multiple sensitive attributes for *L-diversity* by applying certain conditions to determine the size of the bucket. Moreover, Praveena Priyadarsini et al. in [88] accommodate the values corresponding to the sensitive categorical attributes within each bucket by setting the value of the threshold L based on the occurrence of distinct values in the whole column. Besides, Sei et al. in [105] suggest a privacy model called (*L1*, ..., *Lq*)-*diversity* that could be applied on data sets including various sensitive *QI* attributes. The proposed method in [105] does not make any modifications on the original data set, but it adds various random values to each attribute to achieve (*L1*, ..., *Lq*)-*diversity* while the threshold L is set to a fixed value.

Moreover, Oishi et al. in [81] presented (L, d)-*semantic diversity* algorithm considering the resemblance of sensitive attribute values within each bucket by adding distances to settle

the problem of impossibility to satisfy the threshold L of *L-diversity*. The algorithm in [81] satisfies *L-diversity* through a method based on adding a Boolean indicator to every sensitive attribute without generalizing the *QI* attributes. Also, Gaoming et al. in [39] proposed a (K, L, θ)-*diversity* model based on clustering to reduce the information loss and increase the usefulness of data. The algorithm in [39] takes as input three parameters, the thresholds K and L correspond to *K-anonymity* and *L-diversity* techniques respectively and the parameter θ corresponds to the degree of privacy preserving. Additionally, a new technique using the principle of *L-diversity* is presented by Sei and Ohsuga in [103], which randomizes sensitive attributes belonging to each individual. The method in [103] is divided into two parts; the first one concerns the data holder where *L-1* random values are generated and added to a sensitive attribute in the whole original data set. The second one concerns the data user where he/she has the possibility to identify the *QI* attributes that should be analyzed based on the relation between the *QI* attributes and the sensitive ones.

Furthermore, Chakraborty and Tripathy in [15] proposed (α, L)-*diversity* and recursive (α, C, L)-*diversity* techniques. Both eigen vector centrality and noise node addition concepts are used in the process in order to create an anonymized network. In the other side, Tu et al. in [125] proposed a heuristic algorithm in order to get an approximate solution. The algorithm meets *L-diversity* principle for protecting trajectory privacy through specific *Generalization* while guaranteeing the smallest loss of spatio temporal granularity. Besides, Kulkarni and Murugan in [58] proposed an algorithm called C-mixture based Privacy GENetic (CPGEN) algorithm in order to ensure privacy. The method in [58] combines the genetic algorithm with C-mixture theory for privacy measurements. The C-mixture is a new privacy measure, which integrates various privacy constrains belonging to both *K-anonymity* and *L-diversity* principles. Moreover, Susan and Christopher in [121] suggested an anonymization technique by combining the advantages of *Anatomization*, and an improved *Slicing* technique using both *K-anonymity* and *L-diversity* principles to treat high dimensional data sets, which include several sensitive attributes. Actually, the *Anatomization* approach reduces the information loss and the *Slicing* algorithm preserves the correlation and data utility.

Furthermore, Mehta and Rao in [69] proposed an Improved Scalable *L-diversity* (ImSLD) algorithm for scalable anonymization. Before applying the algorithm, the data set was first K-anonymized and then the authors in [69] applied the improved scalable *L-diversity* by checking for L distinct sensitive values within each equivalence class. In order to protect the individual's privacy, Zhu et al. in [146] proposed a ζ-safe (L, K)-*diversity* privacy algorithm belonging to *Generalization-based* approaches and segmentation of records satisfying *L-diversity* within each cluster. This privacy technique ensures that the signatures of each record remain consistent or have no intersection in all versions. Zhu et al. in [146] ensured that their proposed algorithm is able to resist against the Linking attack while preserving the data utility. The technique can also be applied on a data set containing multiple records. Besides, Zheng et al. in [144] suggested a clustering-based *L-diversity* algorithm by considering both *K-anonymity* and *L-diversity* principles. Concerning the *L-diversity* principle, Zheng et al. in [144] added other highly identical records to the cluster in order to find L distinct sensitive attributes. In addition, every generated cluster is used to select the new cluster centroid in the clustering process.

2.3.2 Discussion

The *L-diversity* is very suitable for sensitive attributes. Normally this anonymization technique reduces the dimensions of the data set [117]. However, Praveena Priyadarsini et al. in [88] proposed an enhanced *L-diversity* algorithm able to diversify several sensitive attributes

without dividing the data set. When employing *L-diversity* technique, sensitive attributes would have actually the same frequency [21], [117] and the records corresponding to an individual cannot be distinguished in the database [136]. Nevertheless, the data set may be completed by random values. For instance, Sei et al. in [105] suggested a technique using *L-diversity* principle which does not make any modifications on the original data set but adds various random values to each attribute in the data set. Besides, Oishi et al. in [81] considered the resemblance of sensitive attribute values within each bucket by making distances to settle the problem of impossibility to satisfy the threshold L of *L-diversity*. Another technique using the principle of *L-diversity* is presented by Sei and Ohsuga in [103], which randomizes the sensitive attributes belonging to each individual. Also, Zheng et al. in [144] added other highly identical records to a cluster in order to find L distinct sensitive attributes. In fact, satisfying *L-diversity* technique through random values leads to information loss and thus the proposed technique using *L-diversity* principle cannot preserve data utility. Nevertheless, Gaoming et al. in [39] proposed a (K, L, θ)-*diversity* model based on clustering giving the possibility to reduce information loss and increase the usefulness of data. Same as Susan and Christopher in [121] who suggested an anonymization technique employing both *K-anonymity* and *L-diversity* principles helping to preserve the correlation and the data utility.

The *L-diversity* can ensure that sensitive attributes would have actually the same frequency [21], [117]. In addition, when using *K-diversity* technique the records corresponding to an individual cannot be distinguished in the database [136]. It is impossible to implement the Inference attack against an *L*-diverse database with certitude of 100 % [136]. Besides, even if Zhu et al. in [146] assume that they proposed an algorithm that can preserve high data utility and able to resist against the Linking attack, *L-diversity* is vulnerable to various attacks like Skewness, Homogeneity, Background Knowledge and also Similarity [21], [117]. In the following we present several works using *T-closeness* technique and we make a discussion by presenting the advantages and limitations of some works employing *T-closeness* in their anonymization process.

2.4 *T-closeness* based approaches

The *T-closeness* is an anonymization technique that belongs to *Generalization-based* approaches. This technique deals with sensitive attributes and comes to address the *L-diversity* technique limitations.

2.4.1 Related work

The *T-closeness* technique has been widely studied in the literature in order to ensure privacy when dealing with numerical and non-numerical sensitive attributes. Although, many algorithms for *T-closeness* have been proposed, most of them assume that the privacy threshold T of *T-closeness* must be set to a fixed value. Some researches using the *T-closeness* principle focus more about single sensitive attribute, whereas others apply the anonymization process on multiple sensitive attributes or even on both of them. For example, Roy and Jena [95] proposed a way of determining the parameter T and applied *T-closeness* technique for multiple sensitive attributes instead of single sensitive attribute. In [95], the partitioning classes of sensitive attributes are the only information needed to apply *T-closeness* for multiple sensitive attributes. It is also mentioned in [95] that it is important to know the value of the threshold T in advance so as to unnecessarily anonymize data set over requirement. In addition, Sei et al. [105] presented a privacy model called ($t1$,...,tq)-*closeness*, which can process data sets

including more than one sensitive *QI* attribute. Sei et al. in [105] proposed a method composed of two algorithms. The first one is simple but efficient general anonymization algorithm for (*t1*,. . . ,*tq*)-*closeness*, which is conducted by data holders; the other one is a new reconstruction algorithm which can reduce the error between the reconstructed and the original values depending on the purpose of each data analyzer.

Whereas, Qinghai et al. in [89] proposed a privacy-preserving data publishing method with the name of Multi numerical sensitive attributes clustering method (MNSACM), which uses the ideas of clustering and Multi-Sensitive Bucketization (MSB) to publish micro data with multiple sensitive numerical attributes. Qinghai et al. in [89] anonymized the original data set based on the *Generalization-based* techniques of all the *QI* attributes existing in the data set. The procedure consists of putting tuples that have the same generalized *QI* value in the same bucket. Then, the order of the sensitive numerical values is changed and consequently the proximity breach is prevented according to authors in [89]. Radha and Vatsavayi in [90] applied multi sensitive attribute bucketization and clustering in order to generate an anonymized data set with low information loss and low suppression ratio. We notice that the authors in [90] used the same algorithm mentioned in [89] by conducting experiments on real data set. Saraswathi and Thirukumar [100] ensure that *T-closeness* technique could be applied over Multi Sensitive Bucketization *K-anonymity* Clustering Attribute Hierarchy algorithm (MSB-KACA). The authors in [100] employed Earth Mover's Distance (EMD) to calculate the distance between the sensitive values existing in an equivalence class and the whole data set. Saraswathi and Thirukumar in [100] admit that the minimum calculated distance is considered as the threshold T in order to equitably disperse attributes in the data set. Finally, the authors in [100] obtain a divided data set satisfying the principle of *T-closeness* technique by ensuring privacy and preserving utility.

Further researches combine the *T-closeness* principle with other techniques in order to enhance the anonymization process or to address some limitations. For instance, Domingo-Ferrer in [23] explored the formal similarity between ϵ-*Differential Privacy* and *T-closeness* when anonymizing the data set. Moreover, he highlighted how *T-closeness* and ϵ-*Differential Privacy* techniques are linked to each other regarding anonymization of data sets. Moreover, Domingo-Ferrer in [23] showed that *T-closeness* and ϵ-*Differential Privacy* effectively furnish related privacy safeguards when applied to off-line data release. In addition, Soria-Comas and Domingo-Ferrer [114] indicated that *T-closeness* is considered as one of the extensions of *K-anonymity* technique which can produce ϵ-*Differential Privacy* in data publishing when $t = \exp(\epsilon)$. The authors in [114] supplied a computational procedure based on *Bucketization* for reaching *T-closeness* and ϵ-*Differential Privacy* in data publishing. Moreover, Song et al. [113] proposed an enhanced *T-closeness* privacy protection way which gives a measure of semantic privacy degree and also a specific implementation of the algorithm. Song et al. in [113] gave two specific algorithms. The first one is called Top-Down (T, A)-*closeness* algorithm and the other one is known as (T, A)-*closeness* based on genetic classification algorithm. In addition, Soria-Comas et al. in [115] introduced Microaggregation as an alternative technique for *Generalization-based* and *Suppression-based* techniques in order to generate K-anonymous data sets. Soria-Comas et al. in [115] presented *T-closeness* as a technique that offers one of the strictest privacy guarantees. Moreover, they showed how the K-anonymous T-close data sets are generated by using Microaggregation. The contribution of Soria-Comas et al. in [115] consists of three microaggregation-based algorithms for *T-closeness*. The first algorithm is called *T-closeness* through microaggregation and merging of microaggregated groups of records, the second one is entitled *K-anonymity-first T-closeness* aware microaggregation algorithm and the last one is known as *T-closeness-first* microaggregation algorithm.

Furthermore, in order to encounter the request of data owners demanding a high level of

privacy preserving, Mingzheng et al. in [70] developed a new technique called *T-closeness* Slicing (TCS) to safeguard the data against the Similarity and Skewness attacks. The suggested technique is based on *T-closeness* principle and slicing technique by isolating *QI* attributes from sensitive ones. Then, it rearranges the data set related to sensitive attributes. In the last step, TCS permutes between lines corresponding to all non sensitive columns [70]. By the end, Wang et al. in [132] proposed a maximal-bucket first (MBF) algorithm to achieve (L, e)-*diversity*. The goal is to split an original data set into various equivalence classes satisfying (L, e)-*diversity* constraint. First, the MBF algorithm puts all the records with e-similar sensitive values in the same equivalence class. According to a semantic hierarchy, two values are considered similar if they have the common parent on the tree; otherwise, the two values are comparative dissimilar if they have the common great-grandparent. Second, the MBF algorithm selects records from various equivalence classes to form sequentially equivalence classes based on the size of buckets until the equivalence classes are (L, e)-diverse. The algorithm in [132] repeats the process of constructing equivalence classes until it cannot construct a new equivalence class satisfying (L, e)-*diversity* constraint. After that, the algorithm joins the remaining records to the generated equivalence classes provided that the diversity of the equivalence classes is still achieved. Finally, the algorithm removes the remaining records which cannot be joined to any equivalence class.

2.4.2 Discussion

Just like *L-diversity* technique, *T-closeness* technique is very suitable when dealing with sensitive attributes in such a way the records corresponding to an individual cannot be distinguished in the data set [136]. This technique measures the spacing between two probabilistic distributions which resemble to each other [21] to be able to resist against the Similarity attack. In fact, Qinghai et al. in [89] proposed a privacy-preserving data publishing method where the order of the sensitive numerical values is changed in a way the proximity breach is prevented. In addition, Song et al. in [113] proposed an enhanced *T-closeness* privacy protection way which gives a measure of semantic privacy degree helping to protect data against the Similarity attack. The *T-closeness* permits also to prevent the data set from Skewness attack [117]. Actually, Mingzheng et al. in [70] developed a new technique called TCS to safeguard the data against both Similarity and Skewness attacks. Besides, it is impossible to implement the Inference attack against a T-close database with a certitude of 100 % [136].

Although the computational procedure to enforce *T-closeness* is complex as mentioned in [117], Roy and Jena [95] proposed a technique using *T-closeness* principle in a way that the value of the threshold T of *T-closeness* is known in advance so as to unnecessarily anonymize the data over requirement. Among the limitations of *T-closeness* is that the correlation is lost between attributes since each attribute is generalized separately as mentioned in [117]. Besides, the data utility is damaged when T is very small [21], [117]. Here, come Radha and Vatsavayi in [90] to apply *T-closeness* on multiple sensitive attributes in a way the generated anonymized data set ensure privacy with low information loss. Also, Saraswathi and Thirukumar [100] assume that *T-closeness* technique could be applied over multiple sensitive attributes by obtaining a divided data set where the privacy is ensured and the data utility is preserved. Still *T-closeness* a technique that offers one of the strictest privacy guarantees as mentioned by Soria-Comas et al. in [115].

The motivation of Wang et al. in [132] is fairly similar to ours. However, the fact that the MBF algorithm places all records with e-similar sensitive values into the same set has several drawbacks. First, we could have in the table some buckets containing distinct values which are comparative dissimilar. So, processing these buckets is a waste of time because

they do not lead to Similarity attack. Second, the algorithm needs to check all the values within all buckets existing in the table in order to detect e-similar sensitive values which is not the case in our proposed algorithm in [28]. We do not need to process all the values within buckets. Thus, if we find that two values are comparative dissimilar, we move to the consecutive bucket. Besides, to achieve distinct and comparative dissimilar bucket, we only treat buckets containing semantic similar values.

In addition, the MBF algorithm in [132] selects records from different buckets to constitute an equivalence class sequentially according to the size of buckets until the equivalence class is (L, e)-*diversity* is not practical because normally we work on an L-diverse table which may include buckets containing semantic similar values. In our proposed algorithm, we work on an L-diverse table containing buckets having the maximum possible size of distinct values because the great interest is to ensure privacy as much as possible. Besides, the MBF algorithm adds the remaining records to the generated equivalence classes on the condition that the added records do not destroy the diversity of the equivalence class. However, this will cost time processing because the algorithm recycles the process to construct equivalence classes many times even if it is not needed because the best is to only treat buckets needing anonymization. Also, the fact that the algorithm removes the remaining records which cannot be added to any equivalence class is a bad thing because even if it ensures privacy, the MBF algorithm fails to preserve the data utility. However, in our proposed algorithm, we do not suppress the values and in the same time we do not add fake values to complete the size of buckets. In the following we conclude this chapter by making a summary table representing the anonymization techniques already mentioned in this chapter.

2.5 Conclusion

In this chapter, we have tried to present a large part of the work done in the literature which uses anonymization techniques corresponding to *Generalization-based* approaches. In the following, a summary table of the existing works using *K-anonymity*, *L-diversity* and *T-closeness* anonymization techniques. Normally *K-anonymity* technique only treats *QI* attributes which in combination with external information may allow leakage. However, Kundalwal et al. in [59] consider *QI* attributes as sensitive ones since they may present a threat to privacy violation and thus the authors in [59] use *K-anonymity* technique in their proposed technique by applying it on sensitive attributes instead of *QI* ones. In addition, authors in [59] and [5] are the only ones in Table 2.1 who proposed techniques using *K-anonymity* principle while treating single, multiple, numerical and non-numerical attributes at the same time.

If *K-anonymity* technique treats *QI* attributes in most cases, the *L-diversity* technique deals with sensitive attributes. Nevertheless, Sei et al. in [105] uses *L-diversity* technique in their proposed anonymization technique while treating *QI* attributes. Obviously, a technique using *L-diversity* deals with numeric or non-numerical attributes or even both at the same time. However, even if privacy is ensured to some degree when using the *L-diversity* technique, there is still a privacy leakage since the semantic relationship between values within buckets in the data set is not taken into account. Belonging to Table 2.1, all the authors using *T-closeness* technique in their works only treat sensitive attributes. Besides, Song et al. [113] are the only authors who do not deal with numerical attributes among the other works using *T-closeness* technique; however, based on Table 2.1, the technique proposed by Mingzheng et al. [70] is the only one which treat both numerical and non-numerical attributes. In the next chapter, we will present our hybrid anonymization technique called *V-KLT* which is composed of four proposed algorithms.

Table 2.1: Summary table of works using *K-anonymity*, *L-diversity* and *T*-closeness techniques.

Used technique	Authors	Type of attributes	Single attribute	Multiple attributes	Numerical attributes	Non-numerical attributes
K-anonymity	Xie et al. in [137]	*QI*	Yes	Yes	Yes	No
	Tu et al. in [124]	*QI*	Yes	Yes	Yes	No
	Forster et al. in [38]	*QI*	Yes	Yes	Yes	No
	Fei et al. in [35]	*QI*	Yes	Yes	Yes	No
	Kavitha et al. in [54]	*QI*	Yes	Yes	No	Yes
	Pramanik et al. in [87]	*QI*	Yes	Yes	No	Yes
	Kundalwal et al. in [59]	Sensitive	Yes	Yes	Yes	Yes
	Arava and Lingamgunta in [5]	*QI*	Yes	Yes	Yes	Yes
L-diversity	Praveena Priyadarsini et al. in [88]	Sensitive	No	Yes	No	Yes
	Sei al. in [105]	*QI*	No	Yes	Yes	Yes
	Oishi et al. in [81]	Sensitive	Yes	No	No	Yes
	Sei and Ohsuga in [103]	Sensitive	No	Yes	Yes	Yes
	Susan and Christopher in [121]	Sensitive	No	Yes	Yes	Yes
	Mehta and Rao in [69]	Sensitive	No	Yes	Yes	Yes
	Zhu et al. in [146]	Sensitive	Yes	Yes	Yes	No
T-closeness	Roy and Jena [95]	Sensitive	No	Yes	Yes	No
	Sei et al. [105]	Sensitive	No	Yes	Yes	No
	Qinghai et al. in [89]	Sensitive	No	Yes	Yes	No
	Radha, D. and Vatsavayi, K. in [90]	Sensitive	No	Yes	Yes	No
	Saraswathi and Thirukumar [100]	Sensitive	No	Yes	Yes	No
	Song et al. [113]	Sensitive	Yes	Yes	No	Yes
	Mingzheng et al. [70]	Sensitive	No	Yes	Yes	Yes
	Wang et al. [132]	Sensitive	Yes	No	No	Yes

Chapter 3

Proposition of a New Hybrid Technique *V-KLT* Ensuring Privacy

3.1 Introduction

There exist two main types of attributes in the literature, which are *QI* and sensitive attributes. The *QI* attributes represent a set of attributes that may be used to indirectly identify a person such as "Zip code", "Age" and "Gender". However, *QI* attributes may become sensitive if they are associated to each other [105]. In the other hand, the sensitive attributes are considered as private personal attributes [104]; for example, personal identification, diseases, sexual orientation. The sensitive attributes may cause disclosure problem of privacy if the values related to these sensitive attributes are collected by malicious parties [66].

In this section, we will present our four main proposed algorithms composing our hybrid anonymization technique called *V-KLT*. The first one deals with *QI* attributes and it is called *V-KAN*. This proposed algorithm makes reference to applying *K-anonymity* technique without fixing the value of the threshold *K* beforehand. Then, we will propose another algorithm which treats sensitive attributes with the name of *V-COLD* and has the meaning of applying variable distinct *L-diversity* algorithm only on highly correlated sensitive attributes. By making distinct values within each bucket through *L-diversity* technique, we ensure privacy and when applying the principle of *L-diversity* technique on highly correlated attributes, we preserve the data utility. However, our proposed algorithm *V-COLD* cannot resist against the Similarity attack and therefore there is still privacy leakage after the anonymization process. For this reason, we have suggested two other algorithms called *T-MSN* and *PM-HCA* by applying *T-closeness* technique on multiple sensitive numerical and hierarchical categorical attributes respectively.

3.2 Our algorithm *V-KAN* based on *K-anonymity*

As an anonymization technique, *K-anonymity* has been widely studied in the literature and many related algorithms were suggested. The *K-anonymity* is reached when all the records in a set of *QI* are indistinguishable from at least *K-1* other records in the data set [107]. The *K-*

anonymity is a technique that could belong to both *Generalization-based* and *Randomization-based* techniques depending on the treated *QI* attribute. For example, if we have “Age” as an attribute; then, it is more suitable to apply the *Generalization* on the "Age" attribute by putting its related values into intervals. In this case, we are using *K-anonymity based on Generalization* where values are replaced by more general ones based on Value Graph Hierarchy (VGH), either on Taxonomy tree as mentioned in [54]. In the other side, when we have an attribute such as "Zip code"; then, the most suitable is to apply *K-anonymity based on Suppression* where we hide parts of the values by an asterisk "*" [85]. Still, most of researches assume that the privacy parameter K of *K-anonymity* has to be set to a fixed value before the beginning of the anonymization process. Contrary to this idea, we thought that setting a specific value to the threshold K will limit the power of privacy. In the following, we present our proposed algorithm called *V-KAN*.

3.2.1 Algorithm presentation

The *V-KAN* algorithm could be applied on a huge data set since it does not care about the number of lines existing in such data set [27]. Moreover, our proposed algorithm proceeds by putting all identical combinations with respect to the chosen *QI* attributes into buckets until treating all the existing rows in the data set. Bellow, we present our proposed algorithm.

Algorithm 1 *K-anonymity* without prior value of the threshold K

Require: Test table including *QI*
Ensure: Anonymized Table
T = Original Table
$QIRT$ = *QI* rest of table $T = \{\emptyset\}$
$AQIT$ = Anonymized *QI* table $= \{\emptyset\}$
$i = 2$;
1. Create *QI bucket1* table from the original table so that all combinations of the chosen attributes are identical in all the rows of the created table;

2. Insert in $QIRT$ table lines other than those existing in *QI bucket1* table;
while $QIRT$ is not empty **do**
3. Create *QI bucketi* table from $QIRT$ so that all combinations of the chosen attributes are identical in all the rows of the created table;

4. Update $QIRT$ table by inserting lines other than those existing in *QI bucketi* table;
i++;
end while
return All the created *QI bucketi* and *QI bucket1* are grouped in $AQIT$ table;

The *V-KAN* algorithm is proposed to deal with *QI* attributes. First, we create a table called *QI bucket1* where we put the identical rows with respect to the chosen *QI* attributes. Then, we create another table called "QIRT" where we put the remaining rows from the original test table except the rows already existing in the first created table *QI bucket1*. Later, we reapply the algorithm on the "QIRT" instead of the original test table until the "QIRT" becomes empty. Finally, we return all the created *QI buckets* . Our proposed algorithm

processes both numerical and non-numerical *QI* attributes. Moreover, it does not care about the number of rows existing in the data set which gives us the opportunity to apply this technique on high dimensional data sets. In addition, the resulting time complexity of *V-KAN* algorithm is $O(q \times N^2)$ where q represents the number of the treated *QI* attributes and N represents the number of values within each column in the data set.

3.2.2 Discussion

Actually, the originality of our proposed algorithm lies in constructing buckets without setting a specific value to the threshold K of *K-anonymity*. Thus, *V-KAN* algorithm reduces the time processing since we do not need to browse all the data set's columns to know the number of lines in such data set. In addition, when using *V-KAN* algorithm, we are not supposed to know the number of occurrences of each existing value in the data set. Also, we are not obliged to add fake values to buckets in order to reach a specific value of the threshold K and therefore, we gain data utility. The fact of fixing the value of the privacy parameter K before applying the algorithm forces us to be satisfied with a certain degree of privacy even if we could have the highest possible value of the threshold K. And consequently, ensuring privacy as much as possible. In the following, we present our proposed algorithm dealing with sensitive attributes.

3.3 Our algorithm *V-COLD* based on *L-diversity*

Because each attribute is universally separated, the correlation between different sensitive attributes is lost. This will be a major problem when performing analysis about data utility [55]. Thus, we have to reduce the correlation loss between attributes by grouping highly correlated attributes together. However, even if the data utility is preserved by dividing the huge data set into various data sets containing only highly correlated attributes, the challenge of ensuring privacy remains a crucial issue when sharing a data set that contains personal information [81]. Current information technologies create vast amount of data characterized by velocity, volume and veracity. So, publishing this data increases the possibility of privacy violation. That is why, privacy protection is considered as one of the most hurdles issues [4].

In order to ensure privacy, there are several anonymization techniques in the literature treating sensitive attributes. One of them is called *L-diversity* using horizontal partitioning. The main idea behind *L-diversity* technique is that the values corresponding to sensitive attributes are well represented in each bucket [106]. In this section, a new algorithm of data anonymization is proposed called *V-COLD*. This algorithm is applied whatever the type of attributes is numerical or non-numerical. Actually, the algorithm is divided into two main parts. The first one is intended for preserving data utility by grouping highly correlated attributes together in several small data sets. The determination of the highly correlated attributes is done by using "Pearson" correlation tool. Although, "Pearson" tool only processes numerical values, we used an intermediate algorithm that converts non-numerical values into numerical ones. The second part of the algorithm applies the *L-diversity* technique on highly correlated attributes by splitting the data set horizontally into buckets including distinct values with respect to the treated sensitive attributes in order to ensure privacy. By proposing *V-COLD* algorithm, we try to prove that the distinct *L-diversity* technique must only be applied on data sets containing highly correlated attributes to be an effective anonymization technique.

3.3.1 Algorithm presentation

Our proposed algorithm is divided into two main parts. The first one preserves the data utility by grouping every two highly correlated attributes' columns into several data sets. Concerning the second part of the algorithm, it ensures privacy by applying distinct *L-diversity* technique on each data set including highly correlated attributes.

3.3.1.1 Preliminary Step

With data analysis techniques, precious information could be extracted. In data analysis, Correlation is a well-known mathematical and statistical method for analyzing the compatibility of huge data sets [53]. Since each attribute is generally separated and thus distinguishable, the correlation between various attributes is lost. This is considered as an inherent issue to make efficient analysis of attribute correlations [55]. In order to reduce the correlation loss and thus to preserve the published data utility, a partitioning approach is proposed in [145] based on the lexicographic and non-sensitive attributes sorted by correlation (between sensitive and non-sensitive attributes). Otherwise, authors in [55], [145], [116] and [79] used vertical partitioning by grouping attributes into small data sets according to the correlation existing between these attributes where only highly correlated ones are grouped together. The main idea is to break the association between columns while preserving the relationship within each grouped highly correlated attributes [55], [116].

Actually, the small generated data sets minimize the large dimension of the original data set and preserve better data utility than *Generalization-based* and *Bucketization* approaches [116], [79]. Besides, as mentioned in [55], [121], [116] and [79] the *Slicing* technique preserves data utility because highly correlated attributes are grouped together. The evaluation of the correlation between the pairs of attributes could be realized through several correlation tools depending on the type of the treated attributes. For instance, "Pearson" correlation tool is utilized to evaluate the correlation between two continuous attributes [79], whereas mean-square contingency coefficient is a chi-squared test of correlation between two categorical attributes [145], [79]. In this section, we used "Pearson" tool to identify the highly correlated attributes whether they are numerical or not. In the case we have non-numerical attributes in the data set, we convert the non-numerical values into numerical ones by using a proposed intermediate converting algorithm. Indeed, we assigned the same number to similar values in each column of the treated data set. This conversion will give us the opportunity to process different types of data. The "Pearson" correlation tool is used to calculate the degree of linear correlation between two numerical attributes through the equation 3.1 [36].

$$\frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x}) \sum_i (y_i - \bar{y})}} \tag{3.1}$$

where $\bar{x}$ is the mean of x variable and $\bar{y}$ is the mean of y variable.

The resulting correlation values are in the range $[-1.0, +1.0]$. After calculating "Pearson" correlation coefficient for all the pairs of attributes existing in the data set, we identify those corresponding to the highest value in order to apply the distinct *L-diversity* principle on a data set containing only highly correlated attributes.

3.3.1.2 Anonymization steps

Most of anonymization techniques existing in the literature are applied before publishing the data set [61]. Some of these techniques deal with *QI* attributes and others deal with sensitive

ones. In this section, a technique using the principle of distinct *L-diversity* is suggested dealing with sensitive attributes. Besides, the proposed variable *L-diversity* technique does not take into consideration any prior value of the threshold *L*. Furthermore, the principle of *L-diversity* has been introduced to improve traditional data mining that preserves privacy. The *L-diversity* is considered as an important technique in privacy protection. It is a group based form of anonymization used to ensure privacy in huge data sets by minimizing the huge scale of big data in term of representation [52]. The *L-diversity* model (Distinct, Entropy, Recursive) is an extension of the *K-anonymity* technique, which deals with *QI* attributes [18], [27].

The *L-diversity* ensures that an adversary needs *L-1* values using background knowledge to deduce *L-1* possible values of a sensitive attribute in order to violate the individual privacy [39], [33]. In other words, an Equivalence Class (EC), also called bucket is deemed to satisfy *L-diversity* if there are at least *L* "well-represented" values related to the treated sensitive attributes [106], [33], [112]. Then, the whole data set is deemed to satisfy *L-diversity* when every bucket existing in that data set satisfies *L-diversity* [33], [112]. Our proposed algorithm *V-COLD* applies the principle of distinct *L-diversity* without a prior value of the threshold *L* which means that the value of *L* is not fixed beforehand. Thus, there is an opportunity to maximize this value in order to ensure privacy as much as possible. In the following, we present our proposed algorithm which applies the principle of variable distinct *L-diversity* on highly correlated attributes.

Our algorithm is divided into two main parts. The first one identifies every two highly correlated attributes among all sensitive attributes existing in the original data set. The second one presents the process of applying *L-diversity* principle. The anonymization process is applied on a Table containing *N* tuples and *L* attributes constituting the fields of a structure.

In the first part, from line *6* to line *22* in the algorithm, the identification of every two highly correlated attributes is realized through a correlation tool called "Pearson". Since the data set could contain both numerical and non-numerical attributes and also "Pearson" tool processes only numerical attributes, we convert non-numerical values into numerical ones. Then, we calculate the correlation coefficient between every two attributes *p* in the Original Table. After that, we save the indexes *indi* and *indj* of the attributes corresponding to the highest correlation coefficient *hc*.

In the second part, from line *23* to line *48* in the algorithm, we apply distinct *L-diversity* on the two sensitive attributes having the highest correlation. We start by identifying the distinct values corresponding to the first attribute, then, we put the corresponding tuples in *D1* Table, the remaining tuples are put in *RT* Table. However, Table *D1* may still contain non distinct values with respect to the second attribute. Then, we copy tuples containing distinct values in Table *D1* with respect to the second attribute in Table *D2*. Besides, we add the remaining tuples in *D1* to *RT* Table. Thus, *D2* is the *L*-diverse Table containing distinct values with respect to the two highly correlated attributes. Once the process ends, we clear the Original Table and we copy the content of *RT* Table in the original table and we repeat the process of applying distinct *L-diversity* until *RT* Table is empty. Concerning the time complexity, the part where we convert the non-numerical values into numerical ones (Line5) is of the order of $O(q \times N)$. Thus, the resulting time complexity of *V-COLD* algorithm is $O(q \times N^2)$ where *q* represents the number of the treated sensitive attributes and *N* represents the number of values within each column in the data set.

Algorithm 2 Variable distinct *L-diversity* algorithm on highly correlated attributes

Require: Test table including sensitive attributes

Ensure: Anonymized Table

```
OriginalTable[1 → N] struct attr1(String) attr2(String)...attrL(String) end struct
D1[1 → N] struct attr1(String) attr2(String)...attrL(String) end struct
D2[1 → N] struct attr1(String) attr2(String)...attrL(String) end struct
RT[1 → N] struct attr1(String) attr2(String)...attrL(String) end struct
Conversion(OriginalTable)
hc ← 0
indi ← 0
indj ← 0
p ← 0
find ← 0
i ← 1
while i < L − 1 do
j ← i + 1
    while j < L do
p = pearson(OriginalTable[.].attr[i], OriginalTable[.].attr[j])
if hc < p then
    hc ← p
    indi ← i
    indj ← j
end if
j + +
    end while
i + +
end while
repeat
D1.put(OriginalTable[0])
i ← 1
    while i < N do
find ← 0
if D1.Contains(OriginalTable[i].attr[indi]) then
    find ← 1
    if find == 1 then
        RT.put(OriginalTable[i])
    else
        D1.put(OriginalTable[i])
    end if
end if
i + +
    end while
D2.put(D1[0])
i ← 1
    while i < D1.length() do
find ← 0
if D2.Contains(D1[i].attr[indj]) then
    find ← 1
    if find == 1 then
        RT.put(D1[i])
    else
        D2.put(D1[i])
    end if
end if
i + +
    end while
Clear(OriginalTable)
Copy(OriginalTable, RT)
until RT.isEmpty()
```

3.3.2 Discussion

Our proposed algorithm *V-COLD* starts by defining the highly correlated attributes among the totality of sensitive attributes existing in the original treated data set. The reason behind selecting the highly correlated attributes is the fact that they enable the preservation of the data utility. Besides, the anonymization of small data sets instead of a huge one allows to better manipulate the data and to perform analysis in a simple way. Then, we apply the distinct *L-diversity* principle on each generated small data set containing only highly correlated sensitive attributes. This gives us the possibility to ensure privacy and preserve the data utility. The privacy is ensured since the values within each bucket in the data set are distinct. Besides, the data utility is preserved when the relationship between each two highly correlated sensitive attributes still exist. We have chosen to use "Pearson" tool in order to calculate the correlation between each two consecutive attributes since it is the most commonly used among the other existing correlation tools. Although, the anonymized data set includes distinct values within each bucket after applying *V-COLD* algorithm, it suffers from the Similarity attack because the values within some or all buckets may have a common semantic relation. In the following sections, we present two main algorithms dealing with both sensitive numerical and categorical attributes in order to tackle the issue of the Similarity attack.

3.4 Our algorithm *T-MSN* based on *T-closeness* for Sensitive Numerical attributes

In this section, we will present the reason behind suggesting another technique called *T-MSN* in our anonymization process. In addition, We will propose two anonymization algorithms ensuring privacy. The first one is applied on a data set containing one sensitive numerical attribute; then, we extended our proposed algorithm to also deal with multiple sensitive numerical attributes.

3.4.1 Algorithm presentation

3.4.1.1 Problem position

The *L-diversity* technique by definition means that an adversary has to know *L-1* pieces of background knowledge to be able to eliminate *L-1* possible sensitive attribute values in order to breach privacy [39]. In addition, we remind that we tried in our proposed algorithm *V-COLD* to diversify the values within each bucket with respect to sensitive attributes. Let's look at "Salary" column in Table 3.1, an adversary could easily recognize that a person of *22* years age is corresponding to *bucket1* and certainly has a low salary (Less than *5k*). Then, the idea is to diversify the numerical values corresponding to every bucket in the table in such a way an adversary will find for example a fairly general salary range.

Actually, the *L-diversity* technique itself is unable to resist against Homogeneity and Background Knowledge attacks as cited in [21]. This technique is also inadequate to prevent attribute disclosure. Moreover, *L-diversity* is restricted in its hypothesis of adversarial knowledge. Thus, it is possible for an adversary to obtain information about a sensitive attribute as long as he or she has information about the global distribution of this attribute. Here, we underline the problem of proximity between sensitive numerical values corresponding to every bucket in the data set. Thus, *L-diversity* is not able to resist against the Similarity attack. In

Table 3.1: Table containing sensitive numerical attributes

ID	Age	Salary	Loan	Bucket
1	[20,26]	3k	900	1
2	[20,26]	4k	1200	1
3	[20,26]	5k	1500	1
4	[27,30]	6k	1800	2
5	[27,30]	11k	3300	2
6	[27,30]	8k	2400	2
7	[31,35]	7k	2100	3
8	[31,35]	9k	2700	3
9	[31,35]	10k	3000	3

order to address this limitation, we tried to remove the notion of semantics between numerical values corresponding to every bucket in the data set separately. In the following, we will first present our proposed algorithm treating only one sensitive numerical attributes. Then, an extended version of it will be presented to deal with multiple sensitive numerical attributes.

3.4.1.2 Particular case of treating one sensitive numerical attribute

An equivalence class, also called as bucket is considered to have *T-closeness* if the distance between the distribution of a sensitive attribute in this class and the distribution of the same attribute in the entire data set is no more than a threshold T. In addition, a data set is said to have *T-closeness* if all equivalence classes have *T-closeness* [114], [52]. The *T-closeness* technique aims to prevent attribute disclosure from occurring. This is fulfilled by ensuring that the distribution of the sensitive attributes in each bucket is similar to their distribution in the whole data set [115]. In order to resist against the Similarity attack, we have to calculate the distance between the distribution of a sensitive attribute in an equivalence class and the distribution of the attribute in the whole table. In other words, we calculate the distance between two probabilistic distributions.

There exist several methods to calculate the desired distance such as Variational Distance (VD) or Kullback-Leibler (KL) distance which is proposed by SONG Yang et al. in the classification method based on genetic algorithm to achieve *T-closeness* algorithm [113]. Nevertheless, Earth Mover's Distance (EMD) is the most common choice for calculating the desired distance [105]. When the *T-closeness* technique was introduced, the EMD was suggested as it measures the minimal amount of work needed to transform a distribution to another one through moving probability mass between each other [114]. In other words, the EMD is used to define the resemblance between distributions and it is calculated through Equation 3.2.

$$d[B;Q] = \frac{1}{n-1} \times \frac{1}{n} | m \sum_{i=1}^{m} w_i - \sum_{i=1}^{n} v_i | \tag{3.2}$$

with:
B : the sensitive attribute column of the selected bucket,
Q : the sensitive attribute column of all existing buckets,

n : number of values in Q,
m : number of values in B,
w_i and v_i are elements of B and Q respectively.

The Equation 3.2 represents the adapted and ameliorated version of an equation given in [95], [62] since we have detected that several values are missing when trying to apply it. Lately, more details about Equation 3.2 will be highlighted when presenting our proposed algorithm using *T-closeness* technique and treating multiple sensitive numerical attributes. The Equation 3.2 calculates the distance between the distribution of a sensitive attribute in a bucket and the distribution of the same attribute in the whole table. In the following, we present our proposed preliminary algorithm called Variable *T-closeness* for one sensitive numerical attribute.

Algorithm 3 Variable *T-closeness* for one sensitive numerical attribute

Require: Test table including a *Sensitive numerical attribute*
Ensure: Sliced T-close table
D : distance table
T : original table
RT : Rest of table
SB : Sliced T-close table
$RT = \{\emptyset\}$
$SB = \{\emptyset\}$
1. Calculate distance for all buckets existing in Table T by using the mathematical expression in Equation 3.2
2. Insert the calculated distances in D;
3. Insert into SB the bucket which has the minimum distance as d in D;
4. Insert into RT the rest of buckets which have a distance bigger than d;
5. Update RT by permuting the lines that have the minimum value of the chosen sensitive numerical attribute of every two consecutive buckets;
6. Truncate D table;
while RT is not empty **do**
  7. Calculate distance for all buckets existing in Table RT by using the mathematical expression in Equation 3.2
  8. Insert the calculated distances in D;
  9. Insert into SB the bucket which has the minimum distance as d in D;
  10. Delete the bucket existing in D from RT table;
  11. Update RT by permuting the lines that have the minimum value of the chosen sensitive numerical attribute of every two consecutive buckets;
  12. Truncate D table;
end while
return SB

This proposed algorithm could be applied on a data set containing maximum one sensitive numerical attribute. The algorithm starts by calculating the distance for all buckets existing in the test table by using Equation 3.2. Then, the calculated distances are inserted in a table called "Distance table". This step allows us to define the bucket corresponding to the minimum distance d which means that the current bucket includes distinct and divergent

values. The first bucket corresponding to the minimal value will be inserted in a table called *SB* while the remaining buckets will be permuted. In this algorithm we have chosen to permute between the lines that have the minimum value of the treated sensitive numerical attribute in every two consecutive buckets existing in the data set. We could also choose to permute between the lines that have the maximum value of the treated sensitive numerical attribute. Next, we calculate the distances of the remaining buckets after permutation to add the new bucket corresponding to the minimum value to *SB* table and then, the permutation will be applied on the other buckets except the one with the minimum distance d until we process all the buckets existing in the original data set. The resulting time complexity of this algorithm treating only one sensitive numerical attribute is $O(N^2)$ where N represents the number of values corresponding to the treated sensitive numerical attribute's column. In the following, we extend the work to be able to treat multiple sensitive numerical attributes through our algorithm called *T-MSN*.

3.4.1.3 General case of treating multiple sensitive numerical attributes

In the case of multiple sensitive numerical attributes, an equivalence class satisfies the principle of *T-closeness* if its distribution of multiple sensitive numerical attributes is close to the distribution of these attributes in the whole data set [138]. As said previously, the main goal of our work in this section is to ensure privacy while resisting against the Similarity attack. The solution uses a mathematical equation which measures the resemblance between distributions in the data set. Li et al. in [62] assume that the distance between B and Q could be calculated through the Equation 3.3.

$$d[B;Q] = \frac{1}{n-1}\sum_{i=1}^{n}|\sum_{j=1}^{i}(w_j - v_j)| \tag{3.3}$$

According to Table 3.1, B_1 includes the values of "Salary" attribute belonging to *bucket1* and Q includes the values of "Salary" attribute belonging to all existing buckets. In the following the expression of B_1 and Q.
$B_1 = \{w_1 = 3k, w_2 = 4k, w_3 = 5k\}$
and
$Q = \{v_1 = 3k, v_2 = 4k, v_3 = 5k, v_4 = 6k, v_5 = 11k, v_6 = 8k, v_7 = 7k, v_8 = 9k, v_9 = 10k\}$

Based on the experiments done by Li et al. in [62], when applying Equation 3.3 on *bucket1*, *bucket2* and *bucket3* of "Salary" attribute corresponding to Table 3.1, the probability of optimal mass flow that transforms B_1 to Q is *1/9*. The probability mass is moved across the following pairs:
$(5k \rightarrow 11k)$, $(5k \rightarrow 10k)$, $(5k \rightarrow 9k)$,
$(4k \rightarrow 8k)$, $(4k \rightarrow 7k)$, $(4k \rightarrow 6k)$,
$(3k \rightarrow 5k)$, $(3k \rightarrow 4k)$.
We notice that the pair $(3k \rightarrow 3k)$ could be add even if it equals "0" in order to better understand the transformations.

The cost is then: $1/9 \times 1/8 \times (6 + 5 + 4 + 4 + 3 + 2 + 2 + 1) = 27/72 = 0.375$. When we tried to apply the equation 3.3 we did not find the same cost as mentioned through the experiment. Thus, we had tried to go throughout the cost result in order to find the right equation as shown below.

$$
\begin{aligned}
Cost[B_1, Q] = Distance[B_1, Q] =& 1/9 \times 1/8 \times (6+5+4+4+3+2+2+1) \\
=& 1/9 \times 1/8 \times (|w_3 - v_9| + |w_3 - v_8| + |w_3 - v_7| + \\
& |w_2 - v_6| + |w_2 - v_5| + |w_2 - v_4| + |w_1 - v_3| + \\
& |w_1 - v_2| + |w_1 - v_1|) \\
\approx& 1/9 \times 1/8 \times |(3w_3 + 3w_2 + 3w_1) - (v_9 + v_8 + \\
& v_7 + v_6 + v_5 + v_4 + v_3 + v_2 + v_1)| \\
\approx& \frac{1}{9} \times \frac{1}{8} |3 \sum_{i=1}^{3} w_i - \sum_{i=1}^{9} v_i|
\end{aligned}
\tag{3.4}
$$

From where we concluded that the Equation 3.2 is the most suitable since we have detected that several values are missing when trying to apply Equation 3.3. In the following, we present our proposed algorithm called *T-MSN*.

The *T-MSN* algorithm represents an extended version of our algorithm called "Variable *T-closeness* for one sensitive numerical attribute" proposed in [26]. The extended version contains modifications to be able to process a data set with multiple sensitive numerical attributes. We have used a variable threshold T in *T-closeness* throughout the algorithm until all buckets are processed. Our algorithm highlights two cases depending on the existence or not of the correlation between numerical attributes. If there is a high correlation, the algorithm is only applied on one of these attributes. In the other case, the algorithm is applied on all the numerical sensitive attributes. Besides, the resulting time complexity of *T-MSN* algorithm is $O(q \times N^2)$ where q represents the number of the treated sensitive numerical attributes and N represents the number of values within each column in the data set.

3.4.2 Discussion

We have presented in this section two algorithms ensuring privacy. The first one is a preliminary phase as long as it deals only with one sensitive numerical attribute. In this algorithm, we tried to both ensure privacy and resist against the Similarity attack. We used the principle of *T-closeness* as a solution to address the *L-diversity* limitation which does not care about the notion of semantics within buckets in the data set. Thus, we used the EMD to calculate the distance between values within each bucket and values existing in the whole treated sensitive numerical attribute's column. The EMD allowed us to distinguish between buckets that include values that are close to each other and those including divergent values. Then, a permutation process is applied on buckets containing values that are semantically near to each other to be able by the end of the algorithm to resist against the Similarity attack since an adversary would not find any semantic relation between values within each bucket in the data set. However, if the sensitive numerical attributes on which the *T-MSN* algorithm will be applied are highly correlated, the preliminary algorithm previously presented which deals with just one sensitive numerical attribute will be sufficient. This is justified by the fact that the processing carried out on one of the highly sensitive correlated attributes will be assigned to the other ones. On the other side, the *T-MSN* algorithm will take place when the sensitive numerical attributes existing in the data set are not highly correlated. In this case, every sensitive numerical attribute's column will be anonymized through our proposed algorithm *T-MSN* separately.

Algorithm 4 Variable *T-closeness* for multiple sensitive numerical attributes (*T-MSN*)

Require: Test table including *QI* and *Sensitive Attributes*
Ensure: Sliced *T-closeness* table
D : Distance table
T : Original table
TN : Original table containing only numerical attributes
RT : Rest of table
SB : Sliced *T*-close table
$RT = \{\emptyset\}$
$SB = \{\emptyset\}$
if there is a high correlation between *TN* attributes **then**
Apply algorithm 3
else
$i = 0$;
while $i <$ size(TN) **do**
1. Calculate distance for all buckets existing in Table *T* by using the mathematical expression in Equation 3.2
2. Insert the calculated distances in *D*;
3. Insert into *SB* the bucket which has the minimum distance as *d* in *D*;
4. Insert into *RT* the rest of buckets which have a distance bigger than *d*;
5. Update *RT* by permuting the lines that have the minimum value of $TN[i]$ of every two consecutive buckets;
6. Truncate *D* table;
while *RT* is not empty **do**
7. Calculate distance for all buckets existing in Table *RT* by using the mathematical expression in Equation 3.2
8. Insert the calculated distances in *D*;
9. Insert into *SB* the bucket which has the minimum distance as *d* in *D*;
10. Delete the bucket existing in *D* from *RT* table;
11. Update *RT* by permuting the lines that have the minimum value of $TN[i]$ of every two consecutive buckets;
12. Truncate *D* table;
end while
end while
end if
return *SB*

3.5 Our algorithm *PM-HCA* based on *T-closeness* for Sensitive Categorical Attributes

In the previous section, we proposed an algorithm that treats sensitive numerical attributes called *T-MSN*. This algorithm is able to resist against the Similarity attack when the data set contains sensitive numerical attributes. However, when the attributes existing in the data set are categorical, then the processing becomes more difficult. In this section, we present an algorithm called *PM-HCA* which means that we will ensure privacy by measuring the distance between hierarchical categorical attributes. The proposed algorithm *PM-HCA* resists against the Similarity attack by removing the semantic relationship within every bucket in the data set. In addition we will evaluate the resulting anonymized table through the *NCP* criterion which measures the amount of information loss before and after anonymization.

3.5.1 Problem position

Let's take a look at the following case based on the test Table 3.2. It contains three buckets where distinct *L-diversity* technique is applied since the values within each bucket are distinct with respect to "Disease" attribute.

Table 3.2: Table containing sensitive categorical attribute

ID	Age	Disease	Bucket
1	[20,26]	Concussion injury of brain	1
2	[20,26]	Alzheimer	1
3	[20,26]	Stroke	1
4	[27,30]	Asthma	2
5	[27,30]	Stroke	2
6	[27,30]	Pulmonary emphysema	2
7	[31,35]	Asthma	3
8	[31,35]	Pulmonary emphysema	3
9	[31,35]	Chronic obstructive bronchitis	3

Although the values within each bucket are distinct, they may correspond to a specific category. Consequently, the adversary could easily break the identity of individuals for example based on the content of the hierarchy presented in Figure 3.1 which is specific to respiratory and Gut-brain diseases. Actually, if an adversary has access to the cited hierarchy he or she may easily deduce that a person of *22* years age is belonging to *bucket1* and certainly has a brain disease. In fact, based on the hierarchy in Figure 3.1, the category "brain diseases" includes "Concussion injury of brain", "Alzheimer" and "stroke" which all of them correspond to the *bucket1*. The fact that *bucket1* contains semantically similar values makes the *3*-diverse test Table 3.2 suffering from the Similarity attack.

Then, making the distinction between values related to the same category and those corresponding to different categories is essential but in the same time a difficult task. Thus,

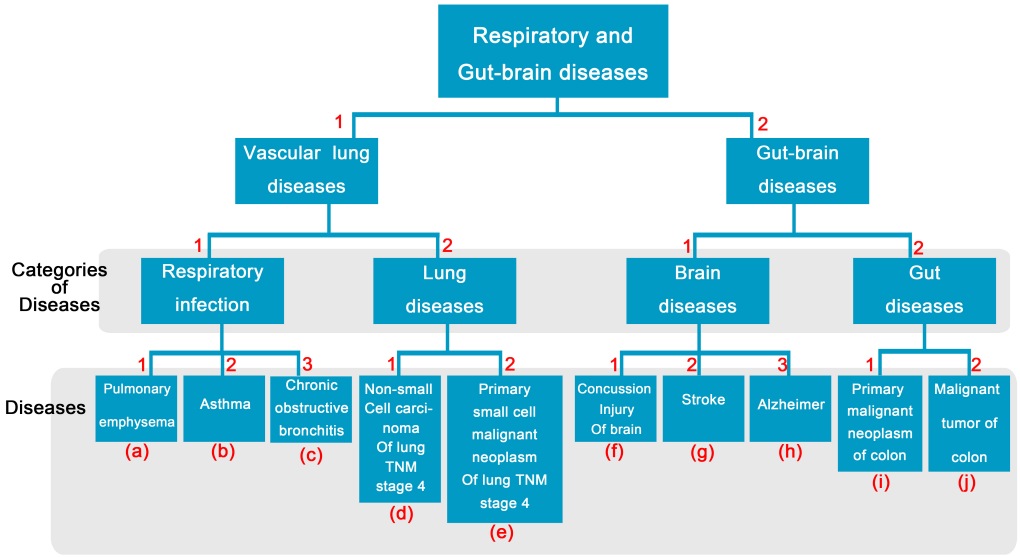

Figure 3.1: Hierarchy for disease categorical attribute.

in order to address the problem of semantics, we had thought about assigning weights to every node in the hierarchy as seen in Figure 3.1 to be able to handle numerical values instead of categorical ones.

After that, we tried to find a way to deduce whether the values existing in each bucket are belonging to the same category or not. As a solution, we have thought about testing if the values within each bucket are consecutive or not since the values related to a specific category must necessarily be consecutive. Then, the anonymization process will only be applied on buckets belonging to the same category.

3.5.2 Algorithm presentation

Based on clustering idea, there exist several anonymization techniques such as *K-anonymity* and *L-diversity* that ensure privacy while preserving data utility. However, even if *L-diversity* technique treats both numerical and non-numerical sensitive attributes, it suffers from various limitations [26]. Among *L-diversity* technique limitations, it does not care about semantics. In other words, after the anonymization process, this technique is unable to specify whether the values within each bucket are corresponding to a specific category or not [26].

That is why, we thought about developing an algorithm that treats categorical sensitive attributes and ensures that the records within each bucket in the data set are not corresponding to a specific category. First of all, we have assigned weights to every node in the hierarchy as mentioned in Figure 3.1 in such a way each disease corresponds to a single code. Based on the hierarchy, we thought about doing a kind of conversion from categorical attributes to numerical ones to be able to make computation on numerical values. Thus, every disease is corresponding to a numerical value different from the other diseases as presented in Table 3.3.

The weights are assigned to each disease manually as seen in Table 3.3 by using a deep

Table 3.3: Weight assignments to diseases.

Disease	Diseases' relating letters	Disease Code
Pulmonary emphysema	a	111
Asthma	b	112
Chronic obstructive bronchitis	c	113
Non-small cell carcinoma of lung TNM stage 4	d	121
Primary small cell malignant neoplasm of lung TNM stage 4	e	122
Concussion injury of brain	f	211
Stroke	g	212
Alzheimer	h	213
Primary malignant neoplasm of colon	i	221
Malignant tumor of colon	j	222

assignment starting from the top of the hierarchy. The assignment is done in a specific order especially when treating the last level of the hierarchy, because our proposed proximity test concerning categorical values will be based on consecutive values within each bucket. In the following, we present our proposed algorithm dealing with sensitive hierarchical categorical attributes called *PM-HCA*.

The algorithm has two inputs, the original table with *L-diversity* property and the hierarchy as presented in Figure 3.1. We start the algorithm by creating a *vector* as mentioned in line *1* of the algorithm to store the number of buckets which need anonymization. After that, and from line *4* (While loop) until line *15* (End of While loop), we calculate the difference *L* between every two consecutive "Disease Code" values within each bucket until treating all buckets in the table. If the difference between two consecutive values in a bucket is different from *1*, then the current bucket does not need anonymization. Else, we store the index of the treated bucket in the variable *vector*.

Next, in line *16*, we make a test to know whether the number of buckets in *vector* is even or odd. If the number of buckets needing anonymization (*decision*) is even, then, we apply the even permutation process where every two consecutive buckets existing in *vector* will be permuted. In that case, we gain time processing because we use every bucket only once. Else, if the *decision* variable is odd, then every two consecutive buckets in *vector* will be permuted in such a way all intermediate buckets in the data set will be used twice except the first and the last buckets. By the end of the algorithm, we make sure to generalize the *QI* attributes' values because the permutation process used in the proposed algorithm will break the *K-anonymity* already applied on *QI* attributes' columns in the data set. In addition, the resulting time complexity of *PM-HCA* algorithm is $O(N)$. In the following, we present the formula of measuring the amount of Information Loss caused by the application of the anonymization process.

3.5.3 Measuring the amount of Information Loss through *NCP*

Before publishing the anonymized data set, it is essential to focus on preserving the data utility. There are various metrics to quantify the usefulness of the data such as Frequent

Algorithm 5 Proximity Measurement for Hierarchical Categorical Attributes algorithm (*PM-HCA*

Require: *TableBuckets* and *Hierarchy*
{*TableBuckets*: The initial table with different buckets}
{*Hierarchy*: the hierarchy of categorical attributes}
Ensure: Anonymized Table
1: *vector*[*NB*]
{*vector*: array of integer} {*NB*: the number of buckets in *TableBuckets*}
2: $z \leftarrow 0$
3: $decision \leftarrow 0$
4: $i \leftarrow 0$
4: **while** $i < NB$ **do**
5: $j \leftarrow 0$
5: **while** $j <$ *TableBuckets*[i].*length* **do**
{For every tuple in *TableBuckets*[i]}
6: $L \leftarrow$ *TableBuckets*[i][j+1].*DiseaseCode*−*TableBuckets*[i][j].*DiseaseCode*
{Measure the difference between two consecutive tuples in *TableBuckets*[i]}
7: **if** $L \neq 1$ **then**
8: *TableBuckets*[i] does not need anonymization
9: goto line 4 (while)
10: **else**
11: $vector[z] \leftarrow i$
{*vector*: stores the indexes of buckets needing anonymization}
12: $z++$
13: $j++$
14: **end if**
14: **end while**
15: $i++$
15: **end while**
16: $decision \leftarrow z \pmod 2$
{z: begins from 0}
17: **if** $decision \neq 0$ **then**
18: Apply odd permutation
19: **else**
20: Apply even permutation
21: **end if**
22: Generalize *QI* attributes ensuring same values within each bucket

Itemset, ILoss, Error rate, *NCP* and other utility measurement as mentioned in [46], [83]. In this thesis, we will focus on the *NCP* utility measurement in order to evaluate the results of our proposed algorithm *PM-HCA*. In the literature, several studies have used the *NCP* to measure the information loss [143], [46], [83], [60], [140]. The utility is mostly measured by computing the information loss and the *NCP* is considered a very popular measurement [71]. The *NCP* cost is principally used to measure the degree of generalization of values in the data set [143], [63]. Moreover, the *NCP* can be calculated during the anonymization process, thus, it can be considered as a distance in clustering based anonymization algorithms [43].

The *NCP* is defined in [40], [67] with respect to the taxonomy tree of the sensitive categorical attribute as shown in the Equation 3.5.

$$NCP(i) = \begin{cases} 0, & subtr(\tilde{i}) = 1; \\ \frac{subtr(\tilde{i})}{|x|}, & \text{otherwise} \end{cases} \tag{3.5}$$

where «$\tilde{i}$» indicates the generalized set of elements that is declared as a non-leaf level node in the hierarchy H and to which the set of elements «i» is mapped. And subtr: $\tilde{I} \rightarrow [1, |i|]$ is a function that counts the number of descending values of the generalized set of elements «$\tilde{I}$» based on the whole hierarchical generalization tree H [40], [67]. The *NCP* for a data set D is defined in Equation 3.6:

$$NCP(D) = \frac{\sum_{\forall i \in \mathbf{I}} (sup(i, D) \times NCP(i))}{\sum_{\forall i \in \mathbf{I}} (sup(i, D))} \tag{3.6}$$

The mapping $i \rightarrow Sup(i, D)$ represents the number of times the set of elements i is repeated in the data set D. The *NCP* is a utility measurement intended for computing the amount of information loss, which is powerful and simple to use [143], [140]. In other words, the *NCP* defines the level of generalization related to the anonymized data set [143], [42]. Notice that more generalization leads to an important loss of information [97]. Then, *NCP* works according to how elements in the data set are generalized. Moreover, it allocates significant penalties to elements with the highest generalization in the original data set [67].

3.5.4 Discussion

We have presented in this section our proposed algorithm called *PM-HCA* dealing with sensitive hierarchical categorical attributes. This algorithm ensures privacy while resisting against the Similarity attack. Since the values are categorical, we assigned a number to each node in the hierarchy in such a way we can calculate the distance between every two consecutive values within each equivalence class in the data set. We notice that the distance is calculated after giving an ascending order to the values within each equivalence class. However, even if we did not calculate for example the similarity between the diseases with codes *111* and *113* in *bucket1* of test Table 3.2, the result is the same. We can deduce that *bucket1* containing *111*, *112* and *113* has to be anonymized.

Then, if a bucket contains *3* values, we simply make two computations (difference between value1 and value2) and (difference between value2 and value3). Thus, we do not need to calculate the difference between value1 and value3. With the same reasoning, if a bucket contains *4* values, we make *3* computations. So, we do not need to calculate the difference between value1 and value3, the difference between value1 and value4, and also the difference between value2 and value4. In this case, our proposed algorithm makes the decision whether the bucket needs anonymization or not with less computations. Nevertheless, our proposed

algorithm *PM-HCA* suffers from a limitation. Suppose that "Respiratory infection" category in Figure 3.1 has for example *10* diseases. Then, in a certain step we will have to calculate the difference between diseases with *119* and *1110* as codes and normally, these codes have to be consecutive but this is not the case. In fact, we made several researches and we rely on the fact that a category of diseases cannot include more than *9* nodes. Otherwise, in most cases we could have more categories than nodes. Certainly, there exist several utility measurements; in this section, we used the most common metric called *NCP* to measure the amount of information loss before and after the anonymization process.

3.6 Conclusion

In this chapter, we had presented several algorithms dealing with *QI* and sensitive attributes. The first one treats *QI* attributes and called *V-KAN*. It ensures privacy as much as possible since it does not fix the value of the threshold K of *K-anonymity* beforehand. Besides, it has the ability to treat huge data sets since determining the number of rows in the data set is not required before applying our proposed algorithm. The second algorithm is entitled *V-COLD* and treats sensitive attributes. We first make a correlation analysis in order to specify the highly correlated attributes among all sensitive attributes existing in the test table. This step allows us to preserve the data utility since the relationship between highly correlated attributes is preserved. Then *V-COLD* algorithm ensures privacy by applying the distinct *L-diversity* technique on the small generated data sets containing only highly correlated attributes. Nevertheless, even if the second algorithm gives good results in terms of anonymization, it cannot resist against the Similarity attack. That is why we thought about addressing our algorithm *V-COLD* limitation by proposing two algorithms using *T-closeness* principle. The *T-MSN* and *PM-HCA* algorithms deal with sensitive numerical and categorical attributes respectively. In the *PM-HCA* algorithm, we specify the buckets needing anonymization among all the existing buckets in the data set. The decision is realized with less computations. In fact, if we have a bucket with n values, we have to make C_n^2 computations. However, with our proposed algorithm, the decision is made only with *n-1* computations. However, *PM-HCA* has a limitation in the way it assigns numerical values to the nodes in the hierarchy as seen in Figure 3.1. In order to tackle this issue, we plan to develop an algorithm dealing with sensitive categorical attributes while focusing on a threshold representing the distance between categorical values within each bucket. Therefore, we would not be obliged to set the number of nodes to a maximum number of *9*. Otherwise, instead of calculating the difference between for example *119* and *1110* disease codes, we can only limit ourselves to calculate the difference between *9* and *10*, that is to say we remove the digits corresponding to categories and we just settle for digits corresponding to the last leaf codes of the hierarchy. In this case, we will not have the problem of fixing a maximum value to diseases in the hierarchy. In the next chapter, we validate our new hybrid technique *V-KLT* in several steps.

Chapter 4

Validation of our New Hybrid Technique *V-KLT* in e-Health Context

4.1 Introduction

In this chapter, we will validate our proposed algorithms already presented in chapter3. The test Table 4.1 will be our original Table containing both *QI* and sensitive attributes. First, we treat the *QI* attributes existing in Table 4.1 by applying our proposed algorithm *V-KAN*. Then, in order to understand how our *V-COLD* algorithm works, it will be applied on an extract of a real data set called "careplans" [130]. Based on the application of "Pearson" correlation tool on Table 4.8 containing *4* sensitive attributes, we generated *2* Tables including only highly correlated attributes. The *V-COLD* algorithm is then applied on the generated tables in order to ensure privacy and to preserve the data utility. However, we found that the resulting table after applying *V-COLD* algorithm cannot resist against the Similarity attack. That is why, we thought about proposing two main algorithms called *T-MSN* and *PM-HCA*. Lately, we will apply the utility measurement *NCP* on both the final anonymized Table and a real fairly huge data set by measuring the information loss before and after the generalization process.

4.2 Step1: Dealing with *QI* attributes through *V-KAN* algorithm

The *K-anonymity* is reached when all the records in a set of *QI* are indistinguishable from at least *K-1* other records in the data set [107]. The *K-anonymity* is a technique that belongs to both *Generalization-based* and *Randomization-based* approaches depending on the type of the *QI* attribute. For example, if we have "Age" as an attribute, then the most suitable is to apply the *Generalization* on "Age" values by putting them into intervals. In this case, we are dealing with *K-anonymity based on Generalization* where values are replaced by more general ones based on Value Graph Hierarchy (VGH), either on Taxonomy tree as mentioned in [54].

Table 4.1: Original table

ID	Gender	Age	Zip code	Salary	Loan	Disease
1	M	24	67540	3k	900	Concussion injury of brain
2	F	28	68333	7k	2100	Asthma
3	M	24	67001	4k	1200	Alzheimer
4	M	32	75201	9k	2700	Asthma
5	F	29	68301	9k	2700	Stroke
6	M	31	75012	11k	3300	Pulmonary emphysema
7	M	34	75111	8k	2400	Chronic obstructive bronchitis
8	F	30	68032	10k	3000	Pulmonary emphysema
9	M	21	67299	5k	1500	Stroke

In addition, when we have an attribute such as "Zip code", then the most suitable is to apply *K-anonymity based on Suppression* where we hide parts of the values by an asterisk "*" [85]. We can substitute the whole values by the special character "*"; however, we will totally lose the utility.

4.2.1 Experiment results

The experiment will be applied on Table 4.1 which represents a test table that includes both *QI* and sensitive attributes. The following Table 4.2 contains the generalized form of "Gender", "Age" and "Zip code" attributes.

Table 4.2: Generalized table of *QI* attributes

ID	Gender	Age	Zip code	Salary	Loan	Disease
1	M	[20,26]	67***	3k	900	Concussion injury of brain
2	F	[27,30]	68***	7k	2100	Asthma
3	M	[20,26]	67***	4k	1200	Alzheimer
4	M	[31,35]	75***	9k	2700	Asthma
5	F	[27,30]	68***	9k	2700	Stroke
6	M	[31,35]	75***	11k	3300	Pulmonary emphysema
7	M	[31,35]	75***	8k	2400	Chronic obstructive bronchitis
8	F	[27,30]	68***	10k	3000	Pulmonary emphysema
9	M	[20,26]	67***	5k	1500	Stroke

We suppose that the values that correspond to the "Age" attribute are generalized referring to the taxonomy tree shown in Figure 4.1.

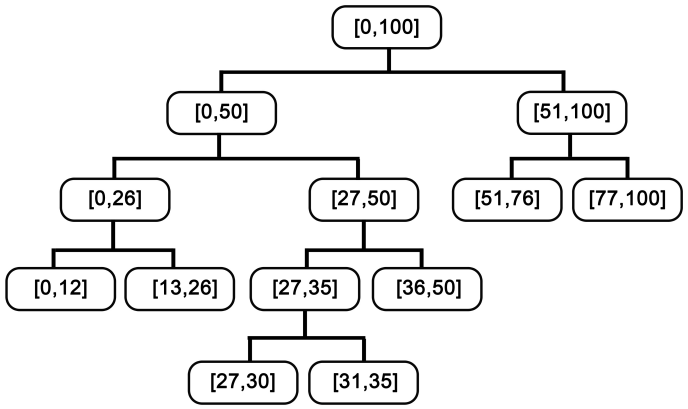

Figure 4.1: Taxonomy Tree for continuous "Age" attribute [27]

The taxonomy tree is defined as needed. The wider the interval of "Age" attribute is, more the privacy is assured, but in return the utility is lost. For this reason, we reduce the "Age" intervals to settle the trade-off between privacy and data utility. For the "Age" attribute, we can use the multiset of exact values. For example, in [64], the multiset 22, 22, 25, 26 is suggested rather than using the generalized interval such as [20, 26]. The multiset of specific values provides more information about the distribution of values in each column than the generalized interval. Thus, using multisets of specific values preserves data utility more than *Generalization*.

The proposed algorithm in this section begins with creating a table called *QI bucket1* based on Table 4.2. We ensure that all combinations of the chosen attributes are identical in all the rows of the created Table 4.3. Thus, even if an adversary accesses to Table 4.2, he or she could not be sure about the true identity of a certain person because he or she will find *3* individuals with the same information with respect to the treated *QI* attributes.

Table 4.3: *QI bucket1*

ID	Gender	Age	Zip code	Salary	Loan	Disease	Bucket
1	M	[20,26]	67***	3k	900	Concussion injury of brain	1
3	M	[20,26]	67***	4k	1200	Alzheimer	1
9	M	[20,26]	67***	5k	1500	Stroke	1

Then, we continue the process of the algorithm by creating another table called *QIRT1* as shown in Table 4.4 which includes lines other than those existing in Table 4.3.

After that, we create another table which we call *QI bucket2* as mentioned in Table 4.5. It is based on Table 4.4 instead of Table 4.2. We ensure that *QI bucket2* includes only identical rows with respect to the treated *QI* attributes.

We can see that our proposed algorithm treats the test table rows in the order. After ensuring that Table 4.5 includes only identical rows wrt to *QI* attributes, we put the remaining

Table 4.4: *QI* rest of table *QIRT1*

ID	Gender	Age	Zip code	Salary	Loan	Disease
2	F	[27,30]	68***	7k	2100	Asthma
4	M	[31,35]	75***	9k	2700	Asthma
5	F	[27,30]	68***	9k	2700	Stroke
6	M	[31,35]	75***	11k	3300	Pulmonary emphysema
7	M	[31,35]	75***	8k	2400	Chronic obstructive bronchitis
8	F	[27,30]	68***	10k	3000	Pulmonary emphysema

Table 4.5: *QI bucket2*

ID	Gender	Age	Zip code	Salary	Loan	Disease	Bucket
2	F	[27,30]	68***	7k	2100	Asthma	2
5	F	[27,30]	68***	9k	2700	Stroke	2
8	F	[27,30]	68***	10k	3000	Pulmonary emphysema	2

lines from Table 4.4 in another table called *QIRT2* as shown in Table 4.6 which itself represents the *QI bucket3*.

Table 4.6: *QI bucket3*

ID	Gender	Age	Zip code	Salary	Loan	Disease	Bucket
4	M	[31,35]	75***	9k	2700	Asthma	3
6	M	[31,35]	75***	11k	3300	Pulmonary emphysema	3
7	M	[31,35]	75***	8k	2400	Chronic obstructive bronchitis	3

Table 4.6 shows the *QI bucket3* representing at the same time the *QI* rest of table *QIRT2* including the remaining lines from Table 4.4 except those belonging to *QI bucket2*. Then, the algorithm ends because we have achieved all buckets so that the *QI* rest of Table 4.4 will be empty. Thus, we do not have to reapply the *K-anonymity* procedure. The final anonymized Table 4.7 represents a table where we had grouped Tables 4.3, 4.5 and 4.6. In other words, we group all the returned *QI buckets* after applying our proposed algorithm.

As shown in Table 4.7, each bucket includes three identical lines with respect to the combination "Gender", "Age" and "Zip code" *QI* attributes. Thus, we can conclude that the resulting threshold K is equal to *3*.

Table 4.7: Anonymized *QI* table

ID	Gender	Age	Zip code	Salary	Loan	Disease	Bucket
1	M	[20,26]	67***	3k	900	Concussion injury of brain	1
3	M	[20,26]	67***	4k	1200	Alzheimer	1
9	M	[20,26]	67***	5k	1500	Stroke	1
2	F	[27,30]	68***	7k	2100	Asthma	2
5	F	[27,30]	68***	9k	2700	Stroke	2
8	F	[27,30]	68***	10k	3000	Pulmonary emphysema	2
4	M	[31,35]	75***	9k	2700	Asthma	3
6	M	[31,35]	75***	11k	3300	Pulmonary emphysema	3
7	M	[31,35]	75***	8k	2400	Chronic obstructive bronchitis	3

4.2.2 Discussion

As shown in the implementation, the value of the privacy parameter K of *K-anonymity* is deduced and not fixed beforehand. Let's see the following example. Suppose that we have *28* rows in our data set, and we have *4* distinct combinations which are repeated *7* times. Thus, after applying the algorithm, necessarily we are going to have *4* buckets with *7* identical rows in each of them. In this case, we have a value of K equals to *7*, so if we had set the value of K to a value that is lower than *7*, then, we will not benefit from the most possible value of K and consequently we will not have the best possible privacy because, as mentioned in [102], the confidentiality of the published data is better ensured when the value of the threshold K is high. Another thing, if we set the threshold K to a fixed value; then, we are obliged to fill in the buckets with fake values. Consequently, as mentioned in [56], the utility of the data set is lost by generating fictitious values. In the following, we present the validation of our proposed algorithm entitled *V-COLD*.

4.3 Step2: Dealing with sensitive attributes through *V-COLD* algorithm

In this section, we will implement our proposed algorithm based on data set related to health sector which includes only sensitive attributes. Besides, the *V-COLD* algorithm is divided into two main parts in its processing. The first one concerns the detection of highly correlated attributes among the existing sensitive attributes in the data set. The second one applies the principle of *L-diversity* only on data sets containing highly correlated attributes.

4.3.1 Experiment results

In this implementation, we will work on Table 4.8 which is a part of a fairly huge real data set called "careplans" [130] containing "Disease", "Treatment", "Date of diagnosis" and "Cure

date" sensitive attributes. In addition, we have randomly selected *9* tuples from the "care-plans" real data set.

Table 4.8: Table of sensitive attributes

ID	Disease	Treatment	Date of diagnosis	Cure date
1	Whiplash injury to neck	Recommendation to rest	04/09/2015	27/09/2015
2	Whiplash injury to neck	Musculoskeletal care	15/02/2008	17/03/2008
3	Fracture of forearm	Recommendation to rest	18/12/2007	04/02/2008
4	Gout	Healthy diet	18/01/1968	24/09/1975
5	Gout	Musculoskeletal care	18/01/1968	24/09/1975
6	Rheumatoid arthritis	Ice therapy	16/12/2005	13/08/2010
7	Whiplash injury to neck	Recommendation to rest	28/12/1942	05/02/1943
8	Gout	Healthy diet	18/01/1968	24/09/1975
9	Rheumatoid arthritis	Healthy diet	16/12/2005	13/08/2010

Before highlighting the different steps of implementing our proposed algorithm using distinct *L-diversity* principle, we need to convert the non-numerical values contained in Table 4.8 into numerical ones as shown in Table 4.9. We substitute non-numerical values (String and Date types) by numerical ones.

After substituting identical non-numerical values by the same numerical value, we calculate the correlation between every two sensitive attributes in the data set. We start the process by calculating the correlation between "Disease" attribute and the other attributes in the data set. In the following, we give the corresponding "Pearson" correlation coefficients:

$$r(Disease,\ Treatment) = 0.8431$$

$$r(Disease,\ Date\ of\ diagnosis) = 0.5103$$

$$r(Disease,\ Cure\ date) = 0.5103$$

The correlation between "Disease" and "Treatment" attributes is strong and positive. However, there is a moderate positive correlation between "Disease" and "Date of diagnosis", the same moderate correlation is between "Disease" and "Cure date" attributes.

The "Pearson" correlation coefficient between "Disease" and "Treatment" equals *0.8431*, which is the highest value among the three correlation values calculated between "Disease" attribute and the other attributes in the data set. The distinct *L-diversity* principle will be applied on a part of Table 4.8, which only contains "Disease" and "Treatment" sensitive attributes.

Table 4.9: Table 4.8 after applying the conversion process

ID	Disease	Treatment	Date of diagnosis	Cure date
1	1	5	9	15
2	1	6	10	16
3	2	5	11	17
4	3	7	12	18
5	3	6	12	18
6	4	8	13	19
7	1	5	14	20
8	3	7	12	18
9	4	7	13	19

Second, we calculate the correlation between "Treatment", "Date of diagnosis" and "Cure date" attributes. As follows, the values of the calculated "Pearson" correlation coefficients:

$$r(\textit{Treatment, Date of diagnosis}) = 0.3983$$

$$r(\textit{Treatment, Cure date}) = 0.3983$$

We remark that the correlation value between "Treatment" and "Date of diagnosis" attributes equals the correlation value between "Treatment" and "Cure date". The relationship between the attributes is weak since the correlation value is near zero value. Finally, we calculate the correlation between the last two attributes "Date of diagnosis" and "Cure date".

$$r(\textit{Date of diagnosis, Cure date}) = 1$$

The computation of "Pearson" correlation coefficient between "Date of diagnosis" and "Cure date" gives a value of *1*, which means that there is a strong positive correlation between these two attributes.

Now, we will process by applying distinct *L-diversity* on Table 4.8 with respect to "Disease" and "Treatment" attributes corresponding to the highest value of Pearson correlation coefficient (*0.8431*).

We are going to highlight through different tables the whole steps until we obtain an anonymized table satisfying distinct *L-diversity*. The Table 4.10 represents *bucket1* where all the tuples are distinct with respect to both "Disease" and "Treatment" attributes.

Table 4.10: Sensitive *bucket1*

Id	Disease	Treatment	Bucket
1	Whiplash injury to neck	Recommendation to rest	1
4	Gout	Healthy diet	1
6	Rheumatoid arthritis	Ice therapy	1

In the first step, we collect the distinct values from "Treatment" attribute column, which are "Recommendation to rest", "Musculoskeletal care", "Healthy diet" and "Ice therapy". Then, we put in *bucket1* the tuples corresponding to the already mentioned distinct values with an ascendant order. We can see that both "Recommendation to rest" and "Musculoskeletal care" values correspond to "Whiplash injury to neck" value; then, we will only retain "Recommendation to rest" value because it is the first one in the order. However, "Healthy diet" and "Ice therapy" correspond to distinct values, which are "Gout" and "Rheumatoid arthritis" values respectively. Then, we will obtain the first bucket satisfying distinct *L-diversity* as mentioned in Table 4.10. In the next step, we put the remaining tuples from Table 4.8 with respect to "Disease" and "Treatment" attributes in another table called Rest of table *RT1*.

Table 4.11: Sensitive Rest of Table *RT1*

Id	Disease	Treatment
2	Whiplash injury to neck	Musculoskeletal care
3	Fracture of forearm	Recommendation to rest
5	Gout	Musculoskeletal care
7	Whiplash injury to neck	Recommendation to rest
8	Gout	Healthy diet
9	Rheumatoid arthritis	Healthy diet

Table 4.11 is called *RT1* containing tuples other than those existing in *bucket1*. This table will take the place of the Original Table 4.8 in the remaining of the implementation. The Table 4.12 corresponds to *bucket2*.

Table 4.12: Sensitive *bucket2*

Id	Disease	Treatment	Bucket
2	Whiplash injury to neck	Musculoskeletal care	2
3	Fracture of forearm	Recommendation to rest	2
8	Gout	Healthy diet	2

Table 4.12 includes three tuples containing distinct values with respect to "Disease" and "Treatment" attributes satisfying distinct *L-diversity*.

Table 4.13: Sensitive *bucket3* and Rest of table *RT2*

Id	Disease	Treatment	Bucket
5	Gout	Musculoskeletal care	3
7	Whiplash injury to neck	Recommendation to rest	3
9	Rheumatoid arthritis	Healthy diet	3

The Table 4.13 represents the Rest of table *RT2* and in the same time *bucket3* since all the tuples existing in this table are all of them containing distinct values. And here we obtain *3* buckets satisfying distinct *L-diversity*.

Table 4.14: Table 4.8 after anonymization.

Disease	Treatment	Date of diagnosis	Cure date	Bucket
Whiplash injury to neck	Recommendation to rest	04/09/2015	27/09/2015	1
Gout	Healthy diet	18/01/1968	24/09/1975	1
Rheumatoid arthritis	Ice therapy	16/12/2005	13/08/2010	1
Whiplash injury to neck	Musculoskeletal care	15/02/2008	17/03/2008	2
Fracture of forearm	Recommendation to rest	18/12/2007	04/02/2008	2
Gout	Healthy diet	18/01/1968	24/09/1975	2
Gout	Musculoskeletal care	18/01/1968	24/09/1975	3
Whiplash injury to neck	Recommendation to rest	28/12/1942	05/02/1943	3
Rheumatoid arthritis	Healthy diet	16/12/2005	13/08/2010	3

We notice that we will reapply all the steps of distinct *L-diversity* on a table containing "Date of diagnosis" and "Cure date" sensitive attributes since there is a strong correlation between them.

Since Tables 4.10, 4.12 and 4.13 satisfy the principle of distinct *L-diversity*, we could say that Table 4.14 satisfies distinct *L-diversity* too. Besides, we remark that at least there exist three tuples within each bucket in Table 4.14. Consequently, the resulting table after the anonymization process is called distinct *L*-diverse table.

4.3.2 Discussion

The aim of this experimental part is to preserve the data utility while ensuring privacy when dealing with sensitive attributes. In order to achieve this, we used "Pearson" correlation tool to be able to define the highly correlated attributes among the existing sensitive attributes in the data set. Then, we generated in our case two tables where every one of them contains two highly correlated attributes. Thus, the data utility is well preserved since the correlation between values within each one of the generated tables still exist. We notice that "Pearson" correlation tool deals with numerical values; that is why, we added to *V-COLD* algorithm an intermediate algorithm to convert non-numerical values to numerical ones in order to be able to deal with both numerical and non-numerical attributes when using "Pearson" tool. Afterwards, variable distinct *L-diversity* technique is applied on the tables containing highly correlated attributes in order to ensure privacy. The threshold L of *L-diversity* is variable in

our proposed algorithm *V-COLD* to be able to ensure privacy as much as possible. In addition, by using a variable threshold L, we are not supposed to achieve distinct *L-diversity* principle by adding fake values to the treated data set. In the following, we present the validation of our proposed algorithm entitled *T-MSN* able to treat sensitive numerical attributes.

4.4 Step3: Resistance against the Similarity attack based on sensitive numerical attributes

In this section, we will present the implementation of our proposed algorithm entitled variable *T-closeness* for sensitive numerical attributes. The algorithm is applied on a modified version of the anonymized *QI* Table 4.7 which satisfies *3*-diversity since the four buckets contain *3* distinct values with respect to "Salary" and "Disease" sensitive attributes. We made a little modification on Table 4.7 in order to better show the importance of the proposed algorithm as shown in Table 4.15 by adding another bucket to the table and also making an ascending order to "ID" attribute.

Table 4.15: Modified Anonymized *QI* table

ID	Gender	Age	Zip code	Salary	Loan	Disease	Bucket
1	M	[20,26]	67***	3k	900	Concussion injury of brain	1
2	M	[20,26]	67***	4k	1200	Alzheimer	1
3	M	[20,26]	67***	5k	1500	Stroke	1
4	F	[27,30]	68***	7k	2100	Asthma	2
5	F	[27,30]	68***	9k	2700	Stroke	2
6	F	[27,30]	68***	10k	3000	Pulmonary emphysema	2
7	M	[31,35]	75***	9k	2700	Asthma	3
8	M	[31,35]	75***	11k	3300	Pulmonary emphysema	3
9	M	[31,35]	75***	8k	2400	Chronic obstructive bronchitis	3
10	F	[36,50]	48***	8k	2400	Pulmonary emphysema	4
11	F	[36,50]	48***	13k	3900	Alzheimer	4
12	F	[36,50]	48***	11k	3300	Asthma	4

Moreover, the Table 4.15 satisfies *3*-anonymity since every bucket contains *3* similar values with respect to "Gender", "Age" and "Zip code" *QI* attributes.

4.4.1 Experiment results

Our proposed algorithm will treat "Salary" attribute. Although *bucket1* contains three distinct values *3K*, *4K* and *5K*, those values are close to each other. That is why we proceed by calculating the distance between the distribution of "Salary" attribute in every bucket and the distribution of the same attribute in the whole table. The Table 4.16 shows calculated distances for all buckets existing in the test Table 4.15.

Table 4.16: Calculated distances based on Table 4.15

Bucket	Bucket's distance
1	0.4696
2	0.1515
3	0.1060
4	0.0151

We notice that *bucket4* corresponds to the lowest distance. Therefore, we have to put it in another table that we call sliced T-close table *SB*. After that, we permute between the lines that have the minimum value of the "Salary" attribute in every two consecutive buckets except *bucket4*. Table 4.17 represents sliced T-close table *SB* which was empty before executing our algorithm.

Table 4.17: Sliced T-close Table *SB* containing *bucket4*

ID	Gender	Age	Zip code	Salary	Loan	Disease	Bucket
10	F	[36,50]	48***	8k	2400	Pulmonary emphysema	4
11	F	[36,50]	48***	13k	3900	Alzheimer	4
12	F	[36,50]	48***	11k	3300	Asthma	4

After selecting *bucket4* which corresponds to the minimum distance compared to the other calculated distances in Table 4.16, we put the remaining buckets including *bucket1*, *bucket2* and *bucket3* in Table 4.18 called rest of table *RT*.

Table 4.18: Rest of table *RT* including *bucket1*, *bucket2* and *bucket3*

ID	Gender	Age	Zip code	Salary	Loan	Disease	Bucket
1	M	[20,26]	67***	3k	900	Concussion injury of brain	1
2	M	[20,26]	67***	4k	1200	Alzheimer	1
3	M	[20,26]	67***	5k	1500	Stroke	1
4	F	[27,30]	68***	7k	2100	Asthma	2
5	F	[27,30]	68***	9k	2700	Stroke	2
6	F	[27,30]	68***	10k	3000	Pulmonary emphysema	2
7	M	[31,35]	75***	9k	2700	Asthma	3
8	M	[31,35]	75***	11k	3300	Pulmonary emphysema	3
9	M	[31,35]	75***	8k	2400	Chronic obstructive bronchitis	3

In this step we have chosen to permute between the lines that have the minimum value of the "Salary" attribute in every two consecutive buckets existing in Table 4.18. We could

also choose to permute between lines having the maximum value of the "Salary" attribute in every two consecutive buckets because the idea is to get after permutation a table that does not contain closed values in all buckets. The Table 4.19 represents the updated version of Table 4.18 after executing the permutation procedure.

Table 4.19: Rest of table *RT* including *bucket1*, *bucket2* and *bucket3* after permutation

ID	Gender	Age	Zip code	Salary	Loan	Disease	Bucket
4	F	[27,30]	68***	7k	2100	Asthma	1
2	M	[20,26]	67***	4k	1200	Alzheimer	1
3	M	[20,26]	67***	5k	1500	Stroke	1
9	M	[31,35]	75***	8k	2400	Chronic obstructive bronchitis	2
5	F	[27,30]	68***	9k	2700	Stroke	2
6	F	[27,30]	68***	10k	3000	Pulmonary emphysema	2
7	M	[31,35]	75***	9k	2700	Asthma	3
8	M	[31,35]	75***	11k	3300	Pulmonary emphysema	3
1	M	[20,26]	67***	3k	900	Concussion injury of brain	3

Most of *T-closeness* algorithms will stop at this level and consider that we attain the desired threshold T which corresponds in this example to *0.015*. However, we notice that the values of "Salary" attribute corresponding to *bucket2* (*8k*, *9k* and *10k*) are close to each other as seen in *bucket1* before executing the first iteration of our proposed algorithm. That is why, we thought about calculating the distance several times until the rest of table *RT* is empty. The Table 4.20 represents the calculated distances of every bucket based this time on Table 4.19.

Table 4.20: Calculated distances based on Table 4.19

Bucket	Bucket's distance
1	0.25
2	0.2083
3	0.0416

After calculating the distances based on rest of table *RT* including *bucket1*, *bucket2* and *bucket3* after permutation, we proceed by joining *bucket3* to sliced T-close table *SB*. We mention that *bucket3* corresponds to the minimum distance compared to the other calculated distances in Table 4.20. The following Table 4.21 represents sliced T-close table *SB* containing *bucket4* and *bucket3*.

After joining *bucket3* to sliced T-close table *SB* which already contained *bucket4*, we remove *bucket4* from Table 4.19 which represents the rest of table *RT* including *bucket1*,

Table 4.21: Sliced T-close table *SB* containing *bucket4* $\bigcup$ *bucket3*

ID	Gender	Age	Zip code	Salary	Loan	Disease	Bucket
10	F	[36,50]	48***	8k	2400	Pulmonary emphysema	4
11	F	[36,50]	48***	13k	3900	Alzheimer	4
12	F	[36,50]	48***	11k	3300	Asthma	4
7	M	[31,35]	75***	9k	2700	Asthma	3
8	M	[31,35]	75***	11k	3300	Pulmonary emphysema	3
1	M	[20,26]	67***	3k	900	Concussion injury of brain	3

bucket2 and *bucket3* after permutation. The Table 4.22 includes all buckets existing in Table 4.19 except *bucket3* which has just been added to sliced T-close table *SB*.

Table 4.22: Rest of table *RT* including *bucket1* and *bucket2*

ID	Gender	Age	Zip code	Salary	Loan	Disease	Bucket
4	F	[27,30]	68***	7k	2100	Asthma	1
2	M	[20,26]	67***	4k	1200	Alzheimer	1
3	M	[20,26]	67***	5k	1500	Stroke	1
9	M	[31,35]	75***	8k	2400	Chronic obstructive bronchitis	2
5	F	[27,30]	68***	9k	2700	Stroke	2
6	F	[27,30]	68***	10k	3000	Pulmonary emphysema	2

Now, we will proceed by permuting twice between the lines corresponding to the lowest value of "Salary" attribute in both *bucket1* and *bucket2*. In this case we will have as result Table 4.23 which represents Table 4.22 after permutation.

Now, only *bucket1* and *bucket2* are remaining. We will proceed by calculating the distances of the two buckets in order to join the bucket which corresponds to the minimum distance to sliced T-close table *SB*. Table 4.24 represents the calculated distance of resulting buckets from Table 4.23.

After calculating the distances, we will join *bucket1* which corresponds to the minimum distance from Table 4.24 to sliced T-close table *SB*. Table 4.25 represents *SB* which already contains *bucket4* and *bucket3*.

After joining *bucket1* to the previous sliced T-close table *SB*. We remove *bucket1* from Table 4.23 which represents the rest of table *RT* including *bucket1* and *bucket2* after permutation. The Table 4.26 includes the remaining bucket which is *bucket2*.

Normally, we have to proceed by permuting between the lines that have the minimum value of the "Salary" attribute in two consecutive buckets. However, it remains only one bucket, and then we have to remove it and joining it to the sliced T-close table *SB* which already contains *bucket4*, *bucket3* and *bucket1*. Table 4.27 represents the final table grouping all the buckets existing in sliced T-closeness table *SB* with an ascending order.

Table 4.23: Rest of table *RT* including *bucket1* and *bucket2* after permutation

ID	Gender	Age	Zip code	Salary	Loan	Disease	Bucket
4	F	[27,30]	68***	7k	2100	Asthma	1
9	M	[31,35]	75***	8k	2400	Chronic obstructive bronchitis	1
2	M	[20,26]	67***	4k	1200	Alzheimer	1
3	M	[20,26]	67***	5k	1500	Stroke	2
5	F	[27,30]	68***	9k	2700	Stroke	2
6	F	[27,30]	68***	10k	3000	Pulmonary emphysema	2

Table 4.24: Calculated distances based on Table 4.23

Bucket	Bucket's distance
1	0.4666
2	0.9666

Table 4.25: Sliced T-close table *SB* containing *bucket4* $\bigcup$ *bucket3* $\bigcup$ *bucket1*

ID	Gender	Age	Zip code	Salary	Loan	Disease	Bucket
10	F	[36,50]	48***	8k	2400	Pulmonary emphysema	4
11	F	[36,50]	48***	13k	3900	Alzheimer	4
12	F	[36,50]	48***	11k	3300	Asthma	4
7	M	[31,35]	75***	9k	2700	Asthma	3
8	M	[31,35]	75***	11k	3300	Pulmonary emphysema	3
1	M	[20,26]	67***	3k	900	Concussion injury of brain	3
4	F	[27,30]	68***	7k	2100	Asthma	1
9	M	[31,35]	75***	8k	2400	Chronic obstructive bronchitis	1
2	M	[20,26]	67***	4k	1200	Alzheimer	1

Table 4.26: Rest of table *RT* including *bucket2*

ID	Gender	Age	Zip code	Salary	Loan	Disease	Bucket
3	M	[20,26]	67***	5k	1500	Stroke	2
5	F	[27,30]	68***	9k	2700	Stroke	2
6	F	[27,30]	68***	10k	3000	Pulmonary emphysema	2

Table 4.27: Final sliced T-close table SB w.r.t Salary

ID	Gender	Age	Zip code	Salary	Loan	Disease	Bucket
4	F	[27,30]	68***	7k	2100	Asthma	1
9	M	[31,35]	75***	8k	2400	Chronic obstructive bronchitis	1
2	M	[20,26]	67***	4k	1200	Alzheimer	1
3	M	[20,26]	67***	5k	1500	Stroke	2
5	F	[27,30]	68***	9k	2700	Stroke	2
6	F	[27,30]	68***	10k	3000	Pulmonary emphysema	2
7	M	[31,35]	75***	9k	2700	Asthma	3
8	M	[31,35]	75***	11k	3300	Pulmonary emphysema	3
1	M	[20,26]	67***	3k	900	Concussion injury of brain	3
10	F	[36,50]	48***	8k	2400	Pulmonary emphysema	4
11	F	[36,50]	48***	13k	3900	Alzheimer	4
12	F	[36,50]	48***	11k	3300	Asthma	4

As shown in Table 4.27, no bucket contains values that are close to each other. So the goal of our proposed algorithm is achieved with success. We note that the use of a variable privacy parameter T is mandatory so as to avoid some cases that can occur during the permutation procedure as mentioned in Table 4.19.

Our proposed variable *T-closeness* algorithm for multiple sensitive numerical attributes highlights two cases depending on the existence or not of correlation between numerical attributes.Thus, if there is a high correlation between numerical attributes, the algorithm is applied only on one of them. And consequently even if we have another sensitive numerical attributes such as "Loan" we will proceed as done before in the experimental part by applying the algorithm on "Salary" or "Loan" attributes. In the other case, the algorithm is applied on all the numerical sensitive attributes until verifying that the resulting table is T-close.

4.4.2 Discussion

The aim of this section is proposing an algorithm able to resist against the Similarity attack when dealing with sensitive numerical attributes. We have first validate the case we have one sensitive numerical attribute in the data set. Then the work was extended to treat multiple sensitive numerical attributes through our proposed algorithm entitled *T-MSN*. This algorithm is divided into two main parts according to the correlation between attributes in the data set. If two attributes are highly correlated, we apply *T-closeness* principle only on one of the sensitive numerical attributes existing in the data set. Then, we will gain time processing since the anonymization process which is applied on one of the attributes will affect the other one immediately. Otherwise, the *T-MSN* algorithm will be applied on every sensitive numerical attribute separately until anonymizing our data set. The *T-MSN* algorithm resists against the Similarity attack by using an adapted mathematical equation able to measure the distance between values within each bucket in the data set. According to the distances measured by EMD, we ensure that the final anonymized data set will contain

divergent values where no semantic relationship exists between values within buckets. In the following, we present the validation of our proposed algorithm entitled *PM-HCA* that is able to treat sensitive hierarchical categorical attributes.

4.5 Step4: Resistance against the Similarity attack based on sensitive categorical attributes

In this section, we present the results of the implementation of our algorithm *PM-HCA*. Moreover, we evaluate the algorithm through *NCP* analysis and we discuss the obtained results. Our proposed algorithm is applied on an extract of a real fairly huge data set called "Careplans" related to the health sector. The "careplans" data set belongs to the "SyntheticMass" database which contains one million synthetic patient medical record.

4.5.1 Experiment results

The experiments in this section will be applied on Table 4.7. However, we have tried to add some modification in order to better understand the processing of our proposed algorithm. In Table 4.28, *K-anonymity* is already applied with respect to "Gender", "Age" and "Zip code" *QI* attributes. The *K-anonymity based on Suppression* is applied on the "Zip code" attribute and *K-anonymity based on Generalization* is applied on the "Age" attribute. Besides, a distinct *L-diversity* technique is also applied with respect to the sensitive attributes "Salary" and "Disease" where their corresponding values into each bucket are distinct.

The test Table 4.28 contains four buckets with different number of tuples. It satisfies *2*-diversity with respect to "Salary" and "Disease" attributes because every bucket includes at least two distinct values. The values corresponding to "Disease" attribute will be converted into codes by using the hierarchy in Figure 3.1. The Table 4.29 represents Table 4.28 where tuples into each bucket are sorted in an ascending order according to "Disease Code" attribute.

Since the "Disease Code" column is sorted in Table 4.29 within each bucket, we make a test to know whether the values within each bucket are consecutive or not. Based on Table 4.29, *bucket1*, *bucket3* and *bucket4* contain consecutive values with respect to "Disease Code" attribute which means that these buckets must be permuted in order to avoid the Similarity attack. The even permutation is applied when the number of buckets needing anonymization is even. In this case, the permutation process will be applied on every two consecutive buckets without repeating a bucket another time until processing all the buckets. For example if we have *4* buckets needing anonymization; then, the algorithm will apply two permutations, the first one between *bucket1* and *bucket2*, and the second one between *bucket3* and *bucket4*. However, the odd permutation is applied when the number of buckets needing anonymization is odd. In this case, the permutation process will be applied in a way that just the first bucket and the last one will be used only once, but all the intermediate buckets will be used twice. For example, if we have *5* buckets needing anonymization; then, the algorithm will apply 4 permutations, the first one between *bucket1* and *bucket2*, the second one between *bucket2* and *bucket3*, the third one between *bucket3* and *bucket4* and the last permutation will be between *bucket4* and *bucket5*.

In our case, it is quite difficult because we have an odd number of buckets requiring anonymization. Actually, *bucket1* belongs to two categories which are "Brain diseases" and "Respiratory infection". Thus, based on Table 4.28, we opted to make the permutation between the maximum value of *bucket1* (Alzheimer disease in line *2*) and the minimum value

Table 4.28: Modified Anonymized *QI* table

ID	Gender	Age	Zip code	Salary	Loan	Disease	Disease code	Bucket
1	M	[20,26]	67***	3k	900	Concussion injury of brain	211	1
2	M	[20,26]	67***	4k	1200	Alzheimer	213	1
3	M	[20,26]	67***	5k	1500	Stroke	212	1
4	F	[27,30]	68***	5k	1500	Asthma	112	2
5	F	[27,30]	68***	9k	2700	Stroke	212	2
6	F	[27,30]	68***	10k	3000	Pulmonary emphysema	111	2
7	F	[27,30]	68***	7k	2100	Malignant tumor of colon	222	2
8	M	[31,35]	75***	11k	3300	Asthma	112	3
9	M	[31,35]	75***	9k	2700	Pulmonary emphysema	111	3
10	M	[31,35]	75***	10k	3000	Chronic obstructive bronchitis	113	3
11	F	[36,50]	48***	14k	4200	Non-small cell carcinoma of lung TNM stage 4	121	4
12	F	[36,50]	48***	13k	3900	Primary small cell malignant neoplasm of lung TNM stage 4	122	4

Table 4.29: Table 4.28 in an ascending order wrt "Disease Code".

ID	Gender	Age	Zip code	Salary	Loan	Disease	Disease code	Bucket
1	M	[20,26]	67***	3k	900	Concussion injury of brain	211	1
2	M	[20,26]	67***	5k	1500	Stroke	212	1
3	M	[20,26]	67***	4k	1200	Alzheimer	213	1
4	F	[27,30]	68***	10k	3000	Pulmonary emphysema	111	2
5	F	[27,30]	68***	5k	1500	Asthma	112	2
6	F	[27,30]	68***	9k	2700	Stroke	212	2
7	F	[27,30]	68***	7k	2100	Malignant tumor of colon	222	2
8	M	[31,35]	75***	9k	2700	Pulmonary emphysema	111	3
9	M	[31,35]	75***	11k	3300	Asthma	112	3
10	M	[31,35]	75***	10k	3000	Chronic obstructive bronchitis	113	3
11	F	[36,50]	48***	14k	4200	Non-small cell carcinoma of lung TNM stage 4	121	4
12	F	[36,50]	48***	13k	3900	Primary small cell malignant neoplasm of lung TNM stage 4	122	4

of *bucket3* (Pulmonary emphysema in line *9*). Then, we applied the permutation process between the new line in *bucket3* which is already swapped containing "Alzheimer" disease and line *11* "Non-small cell carcinoma of lung TNM stage 4" disease in *bucket4*. We notice that *bucket3* has been involved twice in the process of permutation. The Table 4.30 shows the result after applying the permutation process.

Table 4.30: The result Table after applying permutation.

ID	Gender	Age	Zip code	Salary	Loan	Disease	Disease code	Bucket
1	M	[20,26]	67***	3k	900	Concussion injury of brain	211	1
2	M	[20,26]	67***	5k	1500	Stroke	212	1
8	M	[31,35]	75***	9k	2700	Pulmonary emphysema	111	1
4	F	[27,30]	68***	10k	3000	Pulmonary emphysema	111	2
5	F	[27,30]	68***	5k	1500	Asthma	112	2
6	F	[27,30]	68***	9k	2700	Stroke	212	2
7	F	[27,30]	68***	7k	2100	Malignant tumor of colon	222	2
9	M	[31,35]	75***	11k	3300	Asthma	112	3
10	M	[31,35]	75***	10k	3000	Chronic obstructive bronchitis	113	3
11	F	[36,50]	48***	14k	4200	Non-small cell carcinoma of lung TNM stage 4	121	3
3	M	[20,26]	67***	4k	1200	Alzheimer	213	4
12	F	[36,50]	48***	13k	3900	Primary small cell malignant neoplasm of lung TNM stage 4	122	4

Now, even if an adversary has access to the resulting Table 4.30, he or she could not recognize the category of disease belonging to a certain *bucketx* because this bucket after the anonymization process contains at least two categories of diseases. In other terms, there is no correlation between tuples within the same bucket. Consequently, we can see that the resulting buckets in Table 4.30 contain distinct values corresponding at least to two different categories of diseases. However, Table 4.30 still does not resist against the Similarity attack since an adversary may know for example that a person called Bob suffers from Lung diseases if he had access to Table 4.31.

Table 4.31: Information of the individual Bob.

Zip code	Age
48685	48

Based on Table 4.31, the adversary will know that Bob's "Zip code" and "Age" are *48685*

and *48* respectively. Thus, by making a link between Tables 4.30 and 4.31, the adversary will find that an individual who is 48 years old certainly suffers from a disease with codes *121* or *122* and consequently Bob suffers from a disease belonging to Lung diseases category. So, we have to more generalize the *QI* attributes related to buckets needing anonymization to prevent the adversary from knowing the exact category of diseases related to a person and hence resisting against the Similarity attack. Table 4.32 present the final anonymized Table.

Table 4.32: The final anonymized Table.

ID	Gender	Age	Zip code	Salary	Loan	Disease	Disease code	Bucket
1	M	[20,35]	67***	3k	900	Concussion injury of brain	211	1
2	M	[20,35]	67***	5k	1500	Stroke	212	1
8	M	[20,35]	75***	9k	2700	Pulmonary emphysema	111	1
4	F	[27,30]	68***	10k	3000	Pulmonary emphysema	111	2
5	F	[27,30]	68***	5k	1500	Asthma	112	2
6	F	[27,30]	68***	9k	2700	Stroke	212	2
7	F	[27,30]	68***	7k	2100	Malignant tumor of colon	222	2
9	M	[31,50]	75***	11k	3300	Asthma	112	3
10	M	[31,50]	75***	10k	3000	Chronic obstructive bronchitis	113	3
11	F	[31,50]	48***	14k	4200	Non-small cell carcinoma of lung TNM stage 4	121	3
3	M	[20,50]	67***	4k	1200	Alzheimer	213	4
12	F	[20,50]	48***	13k	3900	Primary small cell malignant neoplasm of lung TNM stage 4	122	4

The *Generalization* of *QI* attributes must be applied by creating a new interval in every bucket that was already needing anonymization (In our case, they are *bucket1*, *bucket3* and *bucket4*). Then, we take the minimum and the maximum value of all the intervals existing in each bucket and we put them in the new created interval. If we take for example *bucket4* of Table 4.30, we will transform the intervals [20;26] and [36;50] to a new interval. We will have the values *20* and *50* as the minimum and maximum values respectively of the new created interval. Besides the *QI* attributes' values within each bucket have to be the same to satisfy the *K-anonymity* constraint in the resulting anonymized table.

After ensuring that the *K-anonymity* constraint is satisfied in all the buckets of Table 4.32 with respect to *QI* attributes, the adversary would not be able to deduce the real disease of Bob even if he/she knows the values of Bob's "Zip code" and "Age". Since the information that the adversary has about Bob exist in both *bucket3* and *bucket4* by referring to Table 4.32, Bob's disease corresponds to "Respiratory infection", "Lung diseases" and "Brain diseases" categories. Consequently, the proposed algorithm resists well against the Similarity attack.

4.5.2 Discussion

In this section, we validate our proposed algorithm entitled *PM-HCA* dealing with sensitive hierarchical categorical attributes. There is still an untreated case when applying this algorithm, especially when permuting between buckets. The problem occurs when the resulting table after applying the permutation process is still including similar values. For instance, the two buckets needing anonymization are the same and then necessarily the result after the permutation process will not give a protected table. However, we rely on two things; the first one consists on the fact that the original table is an *L*-diverse table which means that values within each bucket are different form the other buckets in the table. The second one assumes that the original table includes various values and the probability that the permutation will be processed on two identical buckets is close to zero. In addition, we could have limited ourselves to Table 4.30 and considering it as final anonymized table able to resit against the Similarity attack since the values within all buckets in the Table are divergent. However, we found that generalizing the *QI* attributes in Table 4.30 is mandatory in order to prevent any identity leakage.

4.6 Final Step: Evaluating the final anonymized table with *NCP*

Our result Table 4.32 gives good results in terms of anonymization since the *L-diversity* principle is still applied and in the same time Table 4.32 resists against the Similarity attack. There exist several ways to measure the amount of information loss such as Utility loss criterion [50] and *NCP* [42]. Here, we are going to focus on the *NCP* privacy measurement. In the following, we will present the application of *NCP* criterion on both Table 4.28 and Table 4.32. In other words, the *NCP* criterion will be applied before and after the anonymization process. Moreover, the application of NCP criterion will be applied on the real fairly huge data set "Careplans" with respect to Respiratory and Gut-brain diseases category as shown in the hierarchy in Figure 3.1.

4.6.1 Applying the *NCP* on a test table

As mentioned in Figure 3.1, the diseases in the hierarchy refer to letters in order to facilitate the handling of diseases. The Tables 4.33 and 4.35 represent the content of the original and the anonymized buckets after applying the *PM-HCA* algorithm. Besides, Tables 4.34 and 4.36 are the generalized forms of Tables 4.33 and 4.35 respectively based on the hierarchy in Figure 3.1. The generalized form is made by replacing the existing values in each bucket by all the descendent values related to the minimum root which encompasses the previous values before the *Generalization*.

Table 4.33: The original Buckets.

Disease	Bucket
{f, h, g}	1
{j, a, g, b}	2
{b, a, c}	3
{d, e}	4

Table 4.34: The generalized form of Table 4.33 .

Disease	Bucket
$i_1 = \{f, h, g\}$	1
$i_2 = \{a, b, c, d, e, f, g, h, i, j\}$	2
$i_3 = \{b, a, c\}$	3
$i_4 = \{d, e\}$	4

Table 4.35: The anonymized buckets.

Disease	Bucket
{f, g, a}	1
{j, g, b, a}	2
{b, c, d}	3
{h, e}	4

Table 4.36: The generalized form of Table 4.35.

Disease	Bucket
$i_1 = \{a, b, c, d, e, f, g, h, i, j\}$	1
$i_2 = \{a, b, c, d, e, f, g, h, i, j\}$	2
$i_3 = \{a, b, c, d, e\}$	3
$i_4 = \{a, b, c, d, e, f, g, h, i, j\}$	4

The Table 4.34 represents the generalized form of Table 4.33, we recognize that only the values of *bucket2* are changed to their generalized form since the original values belongs to three different categories of diseases which are "Respiratory Infection", "Lung diseases" and "Brain diseases". Thus, the *bucket2* of Table 4.34 will include all the descendent values belonging to the three categories of diseases mentioned before. Based on Equation 3.5, we calculate the *NCP* for the four buckets in Table 4.34.

$$NCP(i_1) = NCP(i_3) = 3/10; NCP(i_2) = 10/10; NCP(i_4) = 2/10$$

$NCP(i_1) = NCP(i_3) = 3/10$ because the lowest common ancestors of i_1 and i_3 are "Brain diseases" and "Respiratory infection" respectively which have 3 leaves. The value of the denominator 10 refers to the number of leaves in the entire Disease domain hierarchy. Moreover, $NCP(i_2) = 10/10 = 1$ because "Respiratory Gut-brain diseases" is the lowest common ancestor of i_2 (which has 10 leaves). Besides, $NCP(i_4) = 2/10$ since the lowest common ancestor of i_4 is "Lung diseases" category which has *2* leaves. And based on Equation 3.6, we calculate the *NCP* for the whole original generalized data set represented in Table 4.34. In this case, the *NCP* equals *0.37*.

$$NCP(D) = [(2 \times 3/10) + (1 \times 10/10) + (2 \times 3/10) + (2 \times 2/10)]/(2 + 1 + 2 + 2) = 0.37$$

In our case, i_1, i_3 and i_4 are repeated twice, however the set of elements i_2 is repeated only once.

We repeat the same process to calculate the *NCP* for Table 4.36 as done before for Table 4.34.

$$NCP(i_1) = NCP(i_2) = NCP(i_4) = 10/10; NCP(i_3) = 5/10$$
$$NCP(D) = [(3 \times 10/10) + (3 \times 10/10) + (4 \times 5/10) + (3 \times 10/10)]/(3 + 3 + 4 + 3) = 0.84$$

The *NCP* is used to evaluate the information loss. Its value is between *0* and *1*. Besides, the less the *NCP* is, the higher data utility is [140], [43]. If *NCP* equals *0*, that means there is no information loss, else if *NCP* equals *1*, that means there is a maximum information loss. In this case, we remark that the *NCP* of the anonymized and generalized test Table 4.36 equals *0.84* and the value of *NCP* belonging to the original generalized test Table 4.34 equals *0.37*.

4.6.2 Applying the *NCP* on a real fairly huge data set

It is interesting to show the performances of our algorithm on a real fairly huge data set called "Careplans" related to the health sector. Therefore, after applying our proposed algorithm, we find *141* buckets including the combination of diseases (a, b, c) corresponding to "Respiratory

infection" category and *105* buckets including the combination of diseases (g, h) corresponding to "Brain diseases" category. Thus, *246* buckets among *397* buckets must be anonymized in order to address the *L-diversity* limitation. According to the proposed algorithm, *210* (*105*2*) buckets will be permuted taking into consideration both "Respiratory infection" and "Brain diseases" categories. And the *36* remaining buckets will be permuted with other buckets even if they do not need to be anonymized. For instance, we permute *36* buckets including (a, b, c) with *36* buckets including (g, h, j) among *37* buckets as shown in Table 4.37. Now, we will show the application of *NCP* criterion on both the generalized form of original data set and the anonymized one.

Tables 4.37 and 4.39 represent the original and the anonymized buckets respectively. Tables 4.38 and 4.40 are the generalized forms of Tables 4.37 and 4.39 respectively based on the hierarchy in Figure 3.1.

Table 4.37: The original data set with buckets.

Number of occurrence of buckets	buckets	Blocs
141	{a, b, c}	i_1
18	{b, c, d}	i_2
28	{b, d, e}	i_3
13	{b, d, f}	i_4
5	{d, f, g}	i_5
24	{d, g, h}	i_6
26	{g, h, i}	i_7
37	{g, h, j}	i_8
105	{g, h}	i_9

Table 4.38: The generalized form of Table 4.37 .

Number of occurrence of buckets	buckets	Blocs
141	{a, b, c}	i_1
18	{a, b, c, d, e}	i_2
28	{a, b, c, d, e}	i_3
13	{a, b, c, d, e, f, g, h, i, j}	i_4
5	{a, b, c, d, e, f, g, h, i, j}	i_5
24	{a, b, c, d, e, f, g, h, i, j}	i_6
26	{f, g, h, i, j}	i_7
37	{f, g, h, i, j}	i_8
105	{f, g, h}	i_9

Table 4.39: The anonymized table of Table 4.37.

Number of occurrence of buckets	buckets	Blocs
141	{g, b, c}	i_1
18	{b, c, d}	i_2
28	{b, d, e}	i_3
13	{b, d, f}	i_4
5	{d, f, g}	i_5
24	{d, g, h}	i_6
26	{g, h, i}	i_7
36	{a, h, j}	i_8
1	{g, h, j}	i_9
105	{a, h}	i_{10}

Table 4.40: The generalized form of Table 4.39.

Number of occurrence of buckets	buckets	Blocs
141	{a, b, c, d, e, f, g, h, i, j}	i_1
18	{a, b, c, d, e}	i_2
28	{a, b, c, d, e}	i_3
13	{a, b, c, d, e, f, g, h, i, j}	i_4
5	{a, b, c, d, e, f, g, h, i, j}	i_5
24	{a, b, c, d, e, f, g, h, i, j}	i_6
26	{f, g, h, i, j}	i_7
36	{a, b, c, d, e, f, g, h, i, j}	i_8
1	{f, g, h, i, j}	i_9
105	{a, b, c, d, e, f, g, h, i, j}	i_{10}

Based on Equation 3.5, we calculate the *NCP* for the *9* buckets in Table 4.38.

$$NCP(i_1) = NCP(i_9) = 3/10$$
$$NCP(i_2) = NCP(i_3) = NCP(i_7) = NCP(i_8) = 5/10$$
$$NCP(i_4) = NCP(i_5) = NCP(i_6) = 10/10$$

And based on Equation 3.6, we calculate the *NCP* for the whole original generalized data set represented in Table 4.38. In this case, the *NCP* equals *0.39*.

$$\begin{aligned} NCP(D) = &[(6 \times (3/10) \times 141) + (5 \times (5/10) \times 18) + (5 \times (5/10) \times 28) \\ &+ (3 \times (10/10) \times 13) + (3 \times (10/10) \times 5) + (3 \times (10/10) \times 24) \\ &+ (5 \times (5/10) \times 26) + (5 \times (5/10) \times 37) + (6 \times (3/10) \times 105)]/[(6 \times 141) + \\ &(5 \times 18) + (5 \times 28) + (3 \times 13) + (3 \times 5) + (3 \times 24) + (5 \times 26) + (5 \times 37) + (6 \times 105)] \\ &= 8413/(10 \times 2147) \\ &= 0.39. \end{aligned}$$

We repeat the same process to calculate the *NCP* for Table 4.40 as done before for Table 4.38.

$$NCP(i_1) = NCP(i_4) = NCP(i_5) = NCP(i_6) = NCP(i_8) = NCP(i10) = 10/10$$
$$NCP(i_2) = NCP(i_3) = NCP(i_7) = NCP(i_9) = 5/10$$

$$\begin{aligned} NCP(D) = &[(6 \times (10/10) \times 141) + (8 \times (5/10) \times 18) + (8 \times (5/10) \times 28) \\ &+ (6 \times (10/10) \times 13) + (6 \times (10/10) \times 5) + (6 \times (10/10) \times 24) \\ &+ (8 \times (5/10) \times 26) + (6 \times (10/10) \times 36) + (8 \times (5/10) \times 1) + (6 \times (10/10) \times 105)] \\ &/[(6 \times 141) + (8 \times 18) + (8 \times 28) + (6 \times 13) + (6 \times 5) + (6 \times 24) + (8 \times 26) \\ &+ (6 \times 36) + (8 \times 1) + (6 \times 105)] \\ &= 22360/(10 \times 2528) \\ &= 0.88 \end{aligned}$$

Figures 4.2 and 4.3 show values of *NCP* before and after anonymization respectively.

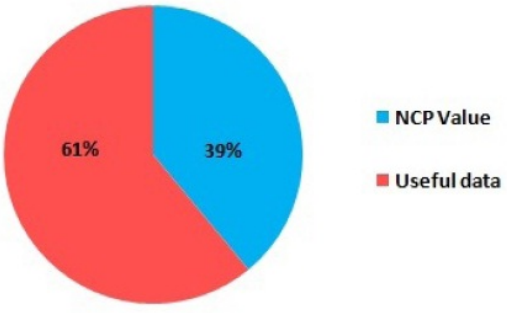

Figure 4.2: *NCP* before anonymization.

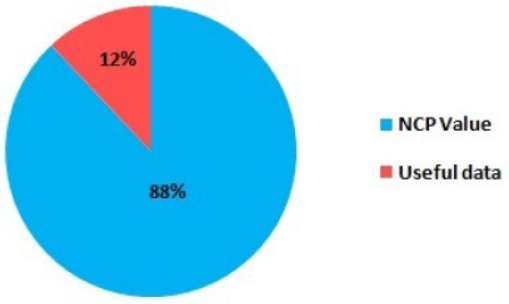

Figure 4.3: *NCP* after anonymization.

Based on Figures 4.2 and 4.3, the amount of information loss had increased after applying the anonymization process however, the amount of useful data had decreased.

The comparison between applying the *NCP* criterion on small data set and fairly big one is presented in the following Table 4.41.

Table 4.41: Comparison between applying the *NCP* criterion on small data set and fairly big one.

	***NCP* on a small data set**	***NCP* on Careplans (*1086* lines)**
Before anonymization	0.37	0.39
After anonymization	0.84	0.88

In the following, we will discus the results given when applying the utility measurement called *NCP* before and after the implementation of our proposed algorithm called *PM-HCA*.

4.6.3 Discussion

As shown in Figures 4.2 and 4.3, we remark that the value of *NCP* has increased after anonymization because the rang of disease values within each bucket has increased, in other words, the value of *NCP* tends to value *1*, which corresponds to the total amount of information loss.

We have calculated the *NCP* value before the anonymization process to know the amount of information loss caused by the application of *K-anonymity* and *L-diversity* techniques (Table 4.28). As mentioned in [40], the amount of *NCP* should equal the value 0 for the original data set, since we have not yet applied the anonymization process and consequently we should not have information loss. Then, the exact amount of information loss for the anonymized data set should equal ($0.88-0.39=0.49$). Figure 4.4 shows the exact amount of information loss.

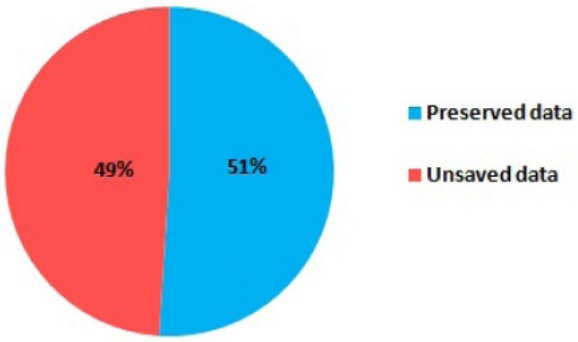

Figure 4.4: The preserved and the unsaved data after anonymization.

We remark that the value of unsaved data after anonymization (0.49) did not even reach 50% of the entire information in the data set.

4.7 Conclusion

In this chapter, we had presented the validation of several algorithms dealing with *QI* and sensitive attributes constituting *V-KLT* algorithm. In the first step, we applied our algorithm *V-KAN* on a test Table where we did not set the threshold *K* of *K-anonymity* to a fixed value.

Thus, by applying our proposed algorithm, we ensure privacy as much as possible since we will get after the anonymization process the highest possible privacy. Otherwise, if we set the privacy parameter K to a fixed value; then, necessarily fictitious data will be added within buckets in the anonymized Table in order to achieve the *K-anonymity* principle. Consequently, although the privacy is ensured, the data utility is lost by adding fake values as mentioned in [56]. In the second step, we validated our proposed algorithm entitled *V-COLD*. The goal behind this algorithm is to preserve the data utility while ensuring privacy when dealing with sensitive attributes. We specified the highly correlated attributes among the existing sensitive attributes in the data set by using a correlation tool called "Pearson" in order to preserve the data utility. Then, variable distinct *L-diversity* technique is applied only on tables including highly correlated attributes in order to ensure privacy. Besides, the threshold L of *L-diversity* is variable to be able to ensure privacy as much as possible same as the processing of *V-KAN* algorithm when we did not fix the value of the threshold K of *K-anonymity*. However, we found that our proposed algorithm *V-COLD* cannot resist against the Similarity attack. That is why we thought about tackling this issue by proposing *T-MSN* and *PM-HCA* algorithms.

The validation of *T-MSN* algorithm is divided into two main parts based on the result of the correlation test between the different sensitive numerical attributes existing in the treated data set. The high correlation between attributes will give us the opportunity to gain time processing since the anonymization process will affect all the remaining sensitive numerical attributes. If not, *T-MSN* algorithm will be applied on every sensitive numerical attribute separately until anonymizing the treated data set. Our proposed algorithm is able to resist against the Similarity attack by measuring the distance between values within each bucket through EMD. The anonymized data set is protected by making divergent values within buckets in such a way no semantic relation exists between the numerical sensitive values. In the other side, we validated our proposed algorithm entitled *PM-HCA*. The resistance against the Similarity attack is based on assigning numerical values to nodes in the hierarchy in such a way each node has a unique numerical value. Then, by calculating the difference between each two consecutive values within each bucket, we can deduce whether the values within each bucket in the data set are convergent or divergent. By using *PM-HCA* algorithm, we make less computations when deciding the buckets needing anonymization since we only process the buckets containing convergent values. In addition, the fact of specifying if the number of buckets needing anonymization is even or odd will allow us to gain time processing in the case where the amount of buckets needing anonymization is even. Besides, the *PM-HCA* algorithm shows a good balance between privacy and data utility when we had applied *NCP* criterion on both a test table and a real fairly huge data set by measuring the information loss before and after the anonymization process.

Conclusion and Perspectives

In this Digital Era, different sectors such as government, business, education, health, etc. are generating large amount of data that could be used by both internal and external sources for analysis and research purposes. However, data gathered from e-health sites and applications may leak personal or sensitive information about patients. All over the world, people are facing an unusual global health emergency due to the COVID-19 pandemic and have witnessed how technology and research based on data analysis play an important role in saving lives. Nevertheless, all of these benefits come with many risks, mainly patient's privacy invasion. Even with international and national data protection laws and standards, it still difficult to prove the compliance of e-health applications and data sets with these regulations.

In this thesis, we essentially focused on preserving patient's privacy. Our main research's goal is to find the best way of ensuring data privacy through anonymization techniques while preserving data utility. At this purpose, we have first made a wide classification of anonymization techniques divided into two main categories including cryptographic and non-cryptographic techniques. This study allows us to highlight the main existing anonymization techniques in the literature that could be used to ensure our main goal. Thus, we proposed new algorithms related to *Generalization-based* approaches. At this level, our second contribution is an algorithm entitled *V-KAN* which aims to ensure privacy as much as possible by using *K-anonymity* technique without setting a prior value for the threshold K at the contrary of the majority of works using this technique. In addition, we are not supposed to browse the whole data set to know the number of existing rows and thus we decrease the time processing.

Unlike the second contribution that deals with QI attributes, the second contribution treats sensitive attributes. We proposed an algorithm called *V-COLD* that apply variable distinct *L-diversity* only on highly sensitive correlated attributes. We found that the principle of *L-diversity* has to be applied on highly correlated attributes to be able to ensure privacy while preserving the data utility. The correlation analysis is done in this thesis through a specific tool and even if the sensitive attributes could be non-numerical, we made an intermediate algorithm able to convert non-numerical values to numerical one.

The third contribution concerns a new issue: Although our algorithm *V-COLD* correctly diversifies the values within each bucket, the semantic notion within values may still exists and thus the anonymization suffers from the Similarity attack. Trying to tackle this issue, we proposed in the first place an algorithm treating one sensitive numerical attribute to be extended later to treat multiple sensitive numerical attributes. Our proposed algorithm entitled *T-MSN* resists against the Similarity attack by calculating the distance between values within each bucket through a method called EMD [105] which allows us to know the resemblance between distributions. Many methods to calculate the desired distance such as VD or KL distance [113]; however, the EMD method still the most common solution used to calculate such distance.

The *T-MSN* algorithm resists against the Similarity attack when the data set includes

sensitive numerical attributes. Nevertheless, when the attributes are categorical, then a special process must be done. For this, we proposed an algorithm entitled *PM-HCA*. The idea is different from what exist in the literature; we have assigned weights to every node in the hierarchy by using a deep assignment starting from the top of the hierarchy in such a way each disease corresponds to a single value. This step allows us to calculate the distance between each two consecutive values within each bucket after giving an ascending order to the values within each bucket in the data set. In addition, we calculated the amount of utility loss through NCP measurement before and after the anonymization process. We notice that the value of unsaved data after applying *PM-HCA* algorithm did not even reach 50% of the entire information existing in the data set. Thus, the result shows a good balance between ensuring privacy and preserving data utility. The combination of our four main proposed algorithms is called *V-KLT* and it enables to treat both *QI* and sensitive attributes.

During the thesis period different aspects of privacy protection were treated. As perspectives, we plan to develop an algorithm dealing with sensitive categorical attributes while focusing this time on a threshold representing the distance between the categorical values within each bucket. In addition, we intend to validate all our proposed algorithms on various huge real data sets related to health sector. Furthermore, it will be interesting to try to propose more efficient and resistant anonymization techniques that preserve data utility in order to enable to open more health data sets to feed AI algorithms and allow more research innovation in the health sector.

List of Publications

1. Z. El Ouazzani and H. El Bakkali. A Classification of non-Cryptographic Anonymization Techniques Ensuring Privacy in Big Data. **International Journal** of Communication Networks And Information Security (IJCNIS), Vol. 12, No. 1, pp. 142-152, **Scopus**, 2020.

2. Z. El Ouazzani, H. El Bakkali and S. Sadki. Privacy Preserving in Digital Health: Main Issues, Technologies, and Solutions. **Book Chapter**, Social, Legal, and Ethical Implications of IoT, Cloud, and Edge Computing Technologies (IGI Global), USA, Chapter 12, pp. 253-276, 2020. doi: 10.4018/978-1-7998-3817-3.ch012.

3. Z. El Ouazzani and H. El Bakkali. Privacy in Big Data through Variable t-Closeness for MSN Attributes. **Book Chapter**, In: Zbakh M., Essaaidi M., Manneback P., Rong C. (eds) Cloud Computing and Big Data: Technologies, Applications and Security. CloudTech 2017. Lecture Notes in Networks and Systems, vol. 49. Springer, Cham, PP 125-141, **Scopus**, 2019. doi: 10.1007/978-3-319-97719-5-9.

4. Z. El Ouazzani and H. El Bakkali. Variable Distinct L-diversity Algorithm Applied on Highly Sensitive Correlated Attributes. The Fifteenth **International Conference** on Wireless and Mobile Communications (ICWMC 2019), Rome, Italy, pp. 47-52, **ThinkMind**, 2019.

5. Z. El Ouazzani and H. El Bakkali. Proximity Test for Sensitive Categorical Attributes in Big Data. The 4th **International Conference** on Cloud Computing Technologies and Applications (CloudTech'18), IEEE, Brussels, Belgium, pp. 1-7, **Scopus**, 2018. doi: 10.1109/CloudTech.2018.8713359.

6. Z. El Ouazzani and H. El Bakkali. A new technique ensuring privacy in big data: k-anonymity without prior value of the threshold k. The First **International Conference** on Intelligent Computing in Data Sciences (ICDS), ELSEVIER, Vol. 127, pp. 52-59, **Scopus**, 2018. doi: 10.1016/j.procs.2018.01.097.

7. Z. El Ouazzani and H. El Bakkali. A New Technique Ensuring Privacy in Big Data: Variable t-Closeness for Sensitive Numerical Attributes". The 3rd **International Conference** on Cloud Computing Technologies and Applications (CloudTech'17), IEEE, pp. 1-6, **Scopus**, 2017. doi: 10.1109/CloudTech.2017.8284733.

8. Z. El Ouazzani, A. Braeken and H. El Bakkali. Proximity Measurement for Hierarchical Categorical Attributes in Big Data. Security and Communication Networks **International Journal**, (Accepted).

Bibliography

[1] K. Abouelmehdi, A. Beni-Hessane, and H. Khaloufi. Big healthcare data: preserving security and privacy. *Springer Journal of big data*, 5(1):1–18, 2018. doi: 10.1186/s40537-017-0110-7.

[2] S. Alpert. Protecting medical privacy: Challenges in the age of genetic information. *Journal of Social Issues*, 59(2):301–322, June 2003. doi: 10.1111/1540-4560.00066.

[3] A. Anjum, N. Ahmad, S. Malik, S. Zubair, and B. Shahzad. An efficient approach for publishing microdata for multiple sensitive attributes. *Springer The Journal of Supercomputing*, 74(10):5127–5155, 2018. doi: 10.1007/s11227-018-2390-x.

[4] J. Anthony and A. S. Thanamani. Comparison and analysis of anonymization techniques for preserving privacy in big data. *Advances in Computational Sciences and Technology*, 10(2):247–253, 2017. doi: 10.37622/ACST/10.2.2017.247-253.

[5] K. Arava and S. Lingamgunta. Adaptive k-anonymity approach for privacy preserving in cloud. *Arabian Journal for Science and Engineering*, 45:2425–2432, 2020. doi: 10.1007/s13369-019-03999-0.

[6] S. Arfaoui, A. Belmekki, and A. Mezrioui. Privacy enhancement of telecom processes interacting with charging data records. *Springer Proceeding of the Tenth International Conference on Soft Computing and Pattern Recognition (SoCPaR)*, 942:268–277, 2019. doi: 10.1007/978-3-030-17065-3_27.

[7] D. Bachlechner, K. La Fors, and A. M. Sears. The role of privacy-preserving technologies in the age of big data. *Proceeding of the 13th Pre-ICIS Workshop on Information Security and Privacy, San Francisco*, pages 1–15, 2018.

[8] A. Barth, J. C. Mitchell, and J. Rosenstein. Conflict and combination in privacy policy languages. *ACM Proceeding of the workshop on Privacy in the electronic society (WPES)*, pages 45–46, October 2004. doi: 10.1145/1029179.1029195.

[9] T. Basso, R. Matsunaga, R. Moraes, and N. Antunes. Challenges on anonymity, privacy and big data. *IEEE Proceeding of the Seventh Latin-American Symposium on Dependable Computing (LADC), Colombia*, pages 164–171, 2016. doi: 10.1109/LADC.2016.34.

[10] D. Baumer, J. Earp, and J. Poindexter. Internet privacy law: a comparison between the united states and the european union. *Journal of Computer and Security*, 23(5):400–412, July 2004. doi: 10.1016/j.cose.2003.11.001.

[11] J. Borking and C. Raab. Laws, pets and other technologies for privacy protection. *Journal of Information, Law and Technology (JILT)*, pages 1–16, November 2001. URL `https://warwick.ac.uk/fac/soc/law/elj/jilt/2001_1/borking/`.

[12] M. Burmester, Y. Desmedt, R. Wright, and A. Yasinsac. Security or privacy, must we choose?. *Proceeding of the Symposium on Critical Infrastructure Protection and the Law*, pages 1–8, 2002.

[13] Y. Canbay, Y. Vural, and S. Sagiroglu. Privacy preserving big data publishing. *IEEE Proceeding of The International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism, Turkey*, pages 24–29, 2018. doi: 10.1109/IBIGDELFT.2018.8625358.

[14] A. Cavoukian and D. Tapscott. Who knows: Safeguarding your privacy in networked world. *Proceeding of Cavouvian 1995 WhoK*, pages 197–208, 1995.

[15] S. Chakraborty and B. Tripathy. Alpha-anonymization techniques for privacy preservation in social networks. *Social Network Analysis and Mining Journal*, 6(29):1–11, December 2016. doi: 10.1007/s13278-016-0337-x.

[16] R. Conejar, H. K. Kim, and J. Rosenstein. A study for home and mobile u-healthcare system ip-based. *International Journal of Software Engineering and its Applications*, 9(5):255–260, 2015. doi: 0.14257/ijseia.2015.9.5.24.

[17] L. Cui, Y. Qu, S. Yu, L. Gao, and G. Xie. A trust-grained personalized privacy-preserving scheme for big social data. *IEEE Proceeding of the International Conference on Communications (ICC), USA*, pages 1–6, 2018. doi: 10.1109/ICC.2018.8422439.

[18] P. Dabas and S. Sharma. Privacy and security issues in social networks with prevailing privacy preserving techniques. *Journal of Network Communications and Emerging Technologies (JNCET)*, 8(2):54–56, February 2018. URL `https://www.jncet.org/Manuscripts/Volume-8/Issue-2/Vol-8-issue-2-M-10.pdf`.

[19] M. Damrich. Ehr adoption rate facts, importance, and comparison to other technology. *Vincari Now part of Nuance*, January 2016. URL `https://vincari.com/media/ehr-adoption-rate-facts-importance-comparison-technology/`.

[20] M. Delgado. The evolution of health care it: Are current u.s. privacy policies ready for the clouds?. *IEEE World Congress on Services, Washington, DC*, pages 371–378, 2011. doi: 10.1109/SERVICES.2011.70.

[21] K. Dhivakar and S. Mohana. A survey on privacy preservation recent approaches and techniques. *International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)*, 2(11):6559–6566, 2014. URL `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1059.3427&rep=rep1&type=pdf`.

[22] T. DLA PIPER. Data protection laws of the world. *Technical report*, 2020. URL `https://www.dlapiperdataprotection.com`.

[23] J. Domingo-Ferrer. On the connection between t-closeness and differential privacy for data releases. *IEEE International Conference on Security and Cryptography (SECRYPT)*, page 478–481, 2013.

[24] M. Du, K. Wang, Z. Xia, and Y. Zhang. Differential privacy preserving of training model in wireless big data with edge computing. *IEEE Proceeding of The Transactions on Big Data*, 6(2), 2018. doi: 10.1109/TBDATA.2018.2829886.

[25] L. El Haourani, A. A. Elkalam, and A. A. Ouahman. Knowledge based access control a model for security and privacy in the big data. *ACM Proceeding of the 3rd International Conference on Smart City Applications (SCA), Morocco*, pages 1–8, 2019. doi: 10.1007/978-3-030-11196-0_60.

[26] Z. El Ouazzani and H. El Bakkali. A new technique ensuring privacy in big data: Variable t-closeness for sensitive numerical attributes. *IEEE, the 3rd International Conference on Cloud Computing and Technology Application (CloudTech'17), Rabat, Morocco*, pages 1–6, 2017. doi: 10.1109/CloudTech.2017.8284733.

[27] Z. El Ouazzani and H. El Bakkali. A new technique ensuring privacy in big data: K-anonymity without prior value of the threshold k. *Elsevier Proceeding of The First International Conference on Intelligent Computing in Data Sciences*, pages 52–59, 2018. doi: 10.1016/j.procs.2018.01.097.

[28] Z. El Ouazzani and H. El Bakkali. Proximity test for sensitive categorical attributes in big data. *IEEE Proceeding of The 4th International Conference on Cloud Computing Technologies and Applications (Cloud'tech), Brussels, Belgium*, pages 1–7, 2018. doi: 10.1109/CloudTech.2018.8713359.

[29] Z. El Ouazzani and H. El Bakkali. Privacy in big data through variable t-closeness for msn attributes. *Elsevier, Lecture Notes in Networks and Systems, Zbakh M., Essaaidi M., Manneback P., Rong C. (eds) Cloud Computing and Big Data: Technologies, Applications and Security (CloudTech 2017)*, 49:125–141, 2019. doi: 10.1007/978-3-319-97719-5-9.

[30] Z. El Ouazzani and H. El Bakkali. Variable distinct l-diversity algorithm applied on highly sensitive correlated attributes. *ThinkMind, The Fifteenth International Conference on Wireless and Mobile Communications (ICWMC), Rome, Italy*, pages 47–52, 2019.

[31] Z. El Ouazzani and H. El Bakkali. A classification of non-cryptographic anonymization techniques ensuring privacy in big data. *International Journal of Communication Networks And Information Security (IJCNIS)*, 12(1):142–152, April 2020.

[32] Z. El Ouazzani, H. El Bakkali, and S. Sadki. Privacy preserving in digital health: Main issues, technologies, and solutions. *Social, Legal, and Ethical Implications of IoT, Cloud, and Edge Computing Technologies (IGI Global), USA, Chapter 12*, pages 253–276, 2020. doi: 10.4018/978-1-7998-3817-3.ch012.

[33] E. Elabd, H. Abdulkader, and A. Mubark. L–diversity based semantic anonymaztion for data publishing. *International Journal of Information Technology and Computer Science (IJITCS)*, 7:1–7, September 2015. doi: 10.5815/ijitcs.2015.10.01. URL `https://pdfs.semanticscholar.org/fc03/95551f4ba7be074b13745db3268195524b05.pdf`.

[34] T. Eytan. 6 reasons why mhealth is different than ehealth. *health Diversity Washington, DC*, 2010. URL `https://www.tedeytan.com/2010/02/18/4731`.

[35] F. Fei, S. Li, H. Dai, C. Hu, W. Dou, and Q. Ni. A k-anonymity based schema for location privacy preservation. *IEEE Transactions on Sustainable Computing*, 4(2):156–167, April 2019. doi: 10.1109/TSUSC.2017.2733018.

[36] W. Feng, Q. Zhu, J. Zhuang, and Y. Shimin. An expert recommendation algorithm based on pearson correlation coefficient and fp-growth. *Cluster Computing*, 22: 7401–7412, 2019. doi: 10.1007/s10586-017-1576-y.

[37] Y. Feruza and T. H. Kim. It security review: Privacy, protection, access control, assurance and system security. *Proceedings Feruza 2007 ITSR*, 2(2):17–32, April 2007.

[38] D. Forster, H. Lohr, and F. Kargl. Decentralized enforcement of k-anonymity for location privacy using secret sharing. *IEEE Vehicular Networking Conference (VNC), Kyoto, Japan*, pages 279–286, 2015. doi: 10.1109/VNC.2015.7385589.

[39] Y. Gaoming, L. Jingzhao, Z. Shunxiang, and Y. Li. An enhanced l-diversity privacy preservation. *The 10th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2013), Shenyang*, pages 1115–1120, 2013. doi: 10.1109/FSKD.2013.6816364.

[40] G. Ghinita, P. Karras, P. Kalnis, and N. Mamoulis. A framework for efficient data anonymization under privacy and accuracy constraints. *ACM Transactions on Database Systems (TODS)*, 34(2):1–47, 2013. doi: 10.1145/1538909.1538911.

[41] A. G. Giannopoulos and D. I. Mouris. Privacy preserving medical data analytics using secure multi party computation. an end-to-end use case. *Master thesis*, pages 1–112, 2018. URL `https://jimouris.github.io/publications/giannopoulosMouris2018thesis.pdf`.

[42] Q. Gong, M. Yang, Z. Chen, and J. Luo. Utility enhanced anonymization for incomplete microdata. *IEEE 20th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Nanchang*, pages 74–79, 2016. doi: 10.1109/CSCWD.2016.7565966.

[43] Q. Gong, M. Yang, Z. Chen, W. Wu, and J. Luo. A framework for utility enhanced in complete microdata anonymization. *Springer International Journal on Cluster Computing*, 20(2):1749–1764, 2017. doi: 10.1007/s10586-017-0795-6.

[44] R. Grant, E. Campbell, R. Gruen, T. Ferris, and D. Blumenthal. Prevalence of basic information technology use by u.s. physicians. *Springer Journal of General Internal Medicine*, 21(11):1150–1155, November 2006. doi: 10.1111/j.1525-1497.2006.00571.x.

[45] N. Gruschka, V. Mavroeidis, K. Vishi, and M. Jensen. Privacy issues and data protection in big data: A case study analysis under gdpr. *IEEE Proceeding of the International Conference on Big Data, USA*, pages 5027–5033, 2018. doi: 10.1109/BigData.2018.8622621.

[46] D. Gunawan and M. Mambo. Set-valued data anonymization maintaining data utility and data property. *12th International Conference on Ubiquitous Information Management and Communication, Langkawi, Malaysia*, pages 1–8, 2018. doi: 10.1145/3164541.3164583.

[47] A. Hamlin, N. Schear, E. Shen, M. Varia, S. Yakoubov, and A. Yerukhimovich. Cryptography for big data security. *Book Chapter for Big Data. Sponsored by the Assistant Secretary of Defense for Research and Engineering under Air Force Contract. Boston, United States: MACS project under NSF Frontier grant*, pages 1–50, December 2015.

[48] O. Hasan, B. Habegger, L. Brunie, N. Bennani, and E. Damiani. A discussion of privacy challenges in user profiling with big data techniques: The eexcess use case. *IEEE Proceeding of the International Congress on Big Data, CA, USA*, pages 25–30, 2013. doi: 10.1109/BigData.Congress.2013.13.

[49] A. Hatem Rashid and N. Binti Mohd Yasin. Sharing healthcare information based on privacy preservation. *Journal of Scientific Research and Essays*, 10(5):184–195, March 2015. doi: 10.5897/SRE11.862.

[50] C. Hebert, D. Bernau, and A. Lahouel. Anonymization techniques to protect data. *United States Patent Application Publication, US 2018 / 0004978 A1, Walldorf, Germany, Patent Application*, pages 1–14, 2018.

[51] T. P. International. The right to privacy in singapore. *Technical report*, 1999.

[52] P. Jain, M. Gyanchandani, and N. Khare. Big data privacy: a technological perspective and review. *Springer Journal of Big Data*, 3(25):1–25, 2016.

[53] H. Jiang, K. Wang, Y. Wang, M. Gao, and Y. Zhang. Energy big data: A survey. *IEEE Access*, 4:3844–3861, 2016. doi: 10.1109/ACCESS.2016.2580581.

[54] S. Kavitha, S. Yamini, and P. Raja Vadhana. An evaluation on big data generalization using k-anonymity algorithm on cloud. *IEEE Proceeding of the 9th International Conference on Intelligent Systems and Control (ISCO), Coimbatore, India*, pages 1–5, 2015. doi: 10.1109/ISCO.2015.7282237.

[55] K. Kiruthika, M. Kavitha, and S. Gayathiri. Publishing high dimensional micro data using anonymization technique. *Imperial Journal of Interdisciplinary Research (IJIR)*, 2(8):86–96, 2016.

[56] S. Kiruthika and M. Mohamed Raseen. Enhanced slicing models for preserving privacy in data publication. *International Conference on Current Trends in Engineering and Technology (ICCTET), COIMBATORE, India*, pages 406–409, December 2013. doi: 10.1109/ICCTET.2013.6675998.

[57] T. Krizan, M. Brakus, and D. Vukelic. In-situ anonymization of big data. *IEEE Proceeding of the 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia*, pages 292–298, 2015. doi: 10.1109/MIPRO.2015.7160282.

[58] Y. Kulkarni and T. Murugan. C-mixture and multi-constraints based genetic algorithm for collaborative data publishing. *Journal of King Saud University-Computer and Information Sciences*, 30(2):175–184, April 2018. doi: 10.1016/j.jksuci.2016.06.001.

[59] M. K. Kundalwal, K. Chatterjee, and A. Singh. An improved privacy preservation technique in health-cloud. *ICT Express*, 5(3):167–172, 2018. doi: 10.1016/j.icte.2018.10.002.

[60] D. Li, X. He, L. Cao, and H. Chen. Permutation anonymization. *Springer International Journal of Intelligent Information Systems*, 47(3):427–445, 2016. doi: 10.1007/s10844-015-0373-4.

[61] F. Li, X. Zou, P. Liu, and J. Chen. New threats to health data privacy. *BMC Bioinformatics*, 12:1–7, November 2011. doi: 10.1186/1471-2105-12-S12-S7.

[62] N. Li, T. Li, and S. Venkatasubramanian. T-closeness: Privacy beyond k-anonymity and l-diversity. *IEEE 23rd International Conference on Data Engineering (ICDE), Istanbul, Turkey*, page 106–115, 2007. doi: 10.1109/ICDE.2007.367856.

[63] R. Li, S. An, D. Li, J. Dong, W. Bai, H. Li, Z. Zhang, and Q. Lin. K-anonymity model for privacy-preserving soccer fitness data publishing. *2nd International Conference on Material Engineering and Advanced Manufacturing Technology (MEAMT)*, pages 1–6, 2018. doi: 10.1051/matecconf/201818903007.

[64] T. Li, N. Li, J. Zhang, and I. Molloy. Slicing: A new approach for privacy preserving data publishing. *IEEE Transactions on Knowledge and Data Engineering*, 24(3):561–574, March 2012. doi: 10.1109/TKDE.2010.236.

[65] X. Li and Z. Zhang. Exploit the scale of big data for data privacy: An efficient scheme based on distance-preserving artificial noise and secret matrix transform. *IEEE Proceeding of the China Summit International Conference on Signal and Information Processing, Xi'an, China*, pages 500–504, 2014. doi: 10.1109/ChinaSIP.2014.6889293.

[66] C. Lin, Q. Liu, P. Fournier Viger, Y. Djenouri, and J. Zhang. Anonymization of multiple and personalized sensitive attributes. *SPRINGER 20th International Conference on Big Data Analytics and Knowledge Discovery (DaWaK)*, pages 204–215, August 2017. doi: 10.1007/978-3-319-98539-8_16.

[67] G. Loukides and A. Gkoulalas-Divanis. Utility-preserving transaction data anonymization with low information loss. *ELSEVIER International Journal on Expert Systems with Applications, Zurich*, 39(10):9764–9777, 2012. doi: 10.1016/j.eswa.2012.02.179.

[68] A. Mehmood, I. Natgunanathan, Y. Xiang, G. Hua, and S. Guo. Protection of big data privacy. *Special section on theoretical foundations for big data applications: Challenges and opportunities*, 4:1821–1834, 2016.

[69] B. Mehta and U. Rao. Improved l-diversity: Scalable anonymization approach for privacy preserving big data publishing. *Journal of King Saud University-Computer and Information Sciences*, pages 1–8, August 2019. doi: 10.1016/j.jksuci.2019.08.006.

[70] W. Mingzheng, J. Zhengrui, Z. Yu, and Y. Haifang. T-closeness slicing a new privacy preserving approach for transactional data publishing. *ACM INFORMS Journal on Computing*, 30(3):1–42, 2018. doi: 10.1287/ijoc.2017.0791.

[71] K. Murakami and T. Uno. Optimization algorithm for k-anonymization of datasets with low information loss. *Springer International Journal of Information Security*, 17(6): 631–644, 2018. doi: 10.1007/s10207-017-0392-y.

[72] S. Murthy, A. Bakar, F. Rahim, and R. Ramli. A comparative study of data anonymization techniques. *5th International Conference on Big Data Security on Cloud (Big Data Security), IEEE International Conference on High Performance and Smart Computing, (HPSC) and IEEE International Conference on Intelligent Data and Security (IDS)*, pages 306–309, 2019. doi: 10.1109/BigDataSecurity-HPSC-IDS.2019.00063.

[73] M. Naish. Implementing public key infrastructure (pki) using microsoft windows server 2012 certificate services. *Digital Certificates*, pages 1–38, September 2014.

[74] M. Naish. Implementing public key infrastructure (pki) using microsoft windows server 2012 certificate services. *SysAdmin, Audit, Network, Security (SANS) Institute Information Security Reading Room*, pages 1–38, September 2014.

[75] S. Nass, L. Levit, and L. Gostin. The value and importance of health information privacy. *Institute of Medicine (US) Committee on Health Research and the Privacy of Health Information: The HIPAA Privacy Rule; Nass SJ, Levit LA, Gostin LO, editors. Beyond the HIPAA Privacy Rule: Enhancing Privacy, Improving Health Through Research. Washington (DC): National Academies Press (US)*, pages 75–110, 2009. URL `https://www.ncbi.nlm.nih.gov/books/NBK9579/`.

[76] G. Natesan and J. Liu. An adaptive learning model for k-anonymity location privacy protection. *IEEE 39th Annual Computer Software and Applications Conference (COMPSAC), Taichung, Taiwan*, pages 10–16, 2015. doi: 10.1109/COMPSAC.2015.281.

[77] S. Nazir, S. Khan, H. Khan, S. Ali, I. García-Magariño, R. Atan, and M. Nawaz. A comparative study of data anonymization techniquesa comprehensive analysis of healthcare big data management, analytics and scientific programming. *IEEE Access*, 8:95714–95733, 2020. doi: 10.1109/ACCESS.2020.2995572.

[78] A. Net 2000 Ltd. Data masking: What you need to know, what you really need to know before you begin. *White paper*, pages 1–26, 2016. URL `https://assets.red-gate.com/products/dba/data-masker/data-masking-whitepaper.pdf`.

[79] P. Nithya and V. Karpagam. Improving privacy and data utility for high- dimensional data by using anonymization technique. *International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)*, 2(1):2874–2881, Mars 2014.

[80] T. Office of the National Coordinator for Health Information Technology. Office-based physician electronic health record adoption. *Health IT Dashboard*, January 2019. URL `https://dashboard.healthit.gov/quickstats/pages/physician-ehr-adoption-trends.php`.

[81] K. Oishi, Y. Tahara, Y. Sei, and A. Ohsuga. Proposal of l-diversity algorithm considering distance between sensitive attribute values. *IEEE Symposium Series on Computational Intelligence (SSCI 2017)*, pages 1–8, 2017. doi: 10.1109/SSCI.2017.8280973.

[82] N. C. on Vital and H. Statistics. Functional requirements needed for the initial definition of a nationwide health information network (nhin). *Report to the Secretary of the U.S. Department of Health and Human Services*, 2006. URL `https://ncvhs.hhs.gov/wp-content/uploads/2014/05/061030lt.pdf`.

[83] M. Orooji and G. Knapp. Improving suppression to reduce disclosure risk and enhance data utility. *Conference on Institute of Industrial and Systems Engineers (IISE), Langkawi, Malaysia*, page 1415–1420, 2019.

[84] S. Parimala. A survey on security and privacy issues of big data in healthcare industry and implication of predictive analytics. *International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)*, 5(4):8130–8134, 2017. doi: 10.15680/IJIRCCE.2017.0504098.

[85] M. Patil and S. Ingale. Privacy control methods for anonymous confidential database using advance encryption standard. *International Journal of Computer Science and Mobile Computing (IJCSMC)*, 2(8):224–229, August 2013.

[86] G. Peddapunnaiah and Y. Kiran. Anonymizing tree structure with privacy preserving data. *International Journal of Emerging Technology in Computer Science and Electronics((IJETCSE))*, 23(8):1–3, November 2016.

[87] M. I. Pramanik, R. Y. K. Lau, and W. Zhang. K-anonymity through the enhanced clustering method. *IEEE 13th International Conference on e-Business Engineering (ICEBE), Macau, China*, pages 85–91, 2016. doi: 10.1109/ICEBE.2016.024.

[88] R. Praveena Priyadarsini, S. Sivakumari, and P. Amudha. Enhanced l–diversity algorithm for privacy preserving data mining. *Springer Annual Convention of the Computer Society of India (CSI)*, 679:14–23, 2016. doi: 10.1007/978-981-10-3274-5-2.

[89] L. Qinghai, S. Hong, and S. Yingpeng. Privacy-preserving data publishing for multiple numerical sensitive attributes. *IEEE Tsinghua Science and Technology*, 20(3):246–254, June 2015. doi: 10.1109/TST.2015.7128936.

[90] D. Radha and V. Vatsavayi. Bucketize: Protecting privacy on multiple numerical sensitive attribute. *Advances in Computational Sciences and Technology*, 10(5):991–1008, 2017.

[91] A. Rahmani, A. Amine, and R. M. Hamou. Combination of access control and de-identification for privacy preserving in big data. *IGI Global International Journal of Information Security and Privacy*, 10(1), 2016.

[92] A. Raj and R. G. L. D'Souza. Big data anonymization in cloud using k-anonymity algorithm using map reduce framework. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT)*, 5(1):50–56, 2019.

[93] P. R. M. Rao, S. M. Krishna, and A. P. S. Kumar. Privacy preservation techniques in big data analytics: a survey. *Springer Journal of Big Data*, 5(33):1–12, 2018. doi: 10.1186/s40537-018-0141-8.

[94] U. P. Rao, B. B. Mehta, and N. Kumar. Scalable l-diversity: An extension to scalable k-anonymity for privacy preserving big data publishing. *IGI Global International Journal of Information Technology and Web Engineering (IJITWE)*, 14(2):27–40, 2019.

[95] D. Roy and S. Jena. Determining t in t-closeness using multiple sensitive attributes. *International Journal of Computer Applications*, 70(19):47–51, May 2013.

[96] S. Sadki and H. El Bakkali. An approach for privacy policies negotiation in mobile health-cloud environments. *IEEE Proceedings of 2015 International Conference on Cloud Computing Technologies and Applications, CloudTech 2015*, pages 1–6, June 2015. doi: 10.1109/CloudTech.2015.7336983.

[97] R. Saeed and A. Rauf. Anatomization through generalization (ag): A hybrid privacy-preserving approachto prevent membership, identity and semantic similarity disclosure attacks. *IEEE International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), Sukkur, Pakistan*, pages 1–7, 2018. doi: 10.1109/ICOMET.2018.8346323.

[98] S. Sangeetha and G. S. Sadasivam. Privacy of big data a review. *Springer International Publishing A. Dehghantanha, K. K. R. Choo (eds.), Book Handbook of Big Data and IoT Security, India*, pages 5–23, 2019.

[99] S. Sangeetha and G. S. Sadasivam. Privacy of big data: A review. *Springer International Publishing A. Dehghantanha, K. K. R. Choo (eds.), Book: Handbook of Big Data and IoT Security, India*, pages 5–23, 2019. doi: 10.1007/978-3-030-10543-3_2.

[100] S. Saraswathi and K. Thirukumar. Enhancing utility and privacy using t-closeness for multiple sensitive attributes. *Open access Journal advances in natural and applied sciences*, 10(5):6–13, 2016.

[101] M. Seastrom. Data stewardship: Managing personally identifiable information in electronic student education records. *Technical/Methodological Report. National Center for Education Statistics, (NCES 2011-602)*, 2011.

[102] J. Sedayao, R. Bhardwaj, and N. Gorade. Making big data, privacy, and anonymization work together in the enterprise: Experiences and issues. *IEEE Proceeding of the International Congress on Big Data (BigData Congress), Alaska, USA*, pages 601–607, 2014. doi: 10.1109/BigData.Congress.2014.92.

[103] Y. Sei and A. Ohsuga. Randomized addition of sensitive attributes for l-diversity. *IEEE The 11th International Conference on Security and Cryptography (SECRYPT)*, pages 1–11, August 2014.

[104] Y. Sei, T. Takenouchi, and A. Ohsuga. (l1,...,lq)-diversity for anonymizing sensitive quasi-identifiers. *IEEE Journal of Trustcom/BigDataSE/ISPA*, 1:596–603, 2015. doi: 10.1109/Trustcom.2015.424.

[105] Y. Sei, H. Okumura, T. Takenouchi, and A. Ohsuga. Anonymization of sensitive quasi-identifiers for l-diversity and t-closeness. *IEEE Transactions on Dependable and Secure Computing*, 16(4):580–593, July 2017. doi: 10.1109/TDSC.2017.2698472.

[106] A. Shah, H. Abbas, W. Iqbal, and R. Latif. Enhancing e-healthcare privacy preservation framework through l-diversity. *IEEE The 14th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 394–399, June 2018. doi: 10.1109/IWCMC.2018.8450306.

[107] K. Sharma, A. Jayashankar, K. SharmilaBanu, and B. Tripathy. Data anonymization through slicing based on graph-based vertical partitioning. *Springer Proceeding of the 3rd International Conference on Advanced Computing, Networking and Informatics (ICACNI), India*, 44:569–576, September 2015. doi: 10.1007/978-81-322-2529-4_59.

[108] K. Sharmila, A. C. S. Borgia, and V. Sreeja. A comprehensive study of data masking techniques on cloud. *International Journal of Pure and Applied Mathematics*, 119(15): 3719–3728, 2018.

[109] K. M. P. Shrivastva, M. Rizvi, and S. Singh. Big data privacy based on differential privacy a hope for big data. *IEEE Proceeding of the 6th International Conference on Computational Intelligence and Communication Networks (CICN), Bhopal, India*, pages 776–781, 2014. doi: 10.1109/CICN.2014.167.

[110] M. Snehal and P. Rashmi. Data de-identification tool for privacy preserving data mining. *International Journal of Computer Science Engineering and Information Technology Research (IJCSEITR)*, 3(1):267–276, 2013.

[111] D. J. Solove. Conceptualizing privacy. *California Law Review*, 90:1087–1155, 2002. doi: 10.2307/3481326.

[112] L. Sondeck, M. Laurent-Maknavicius, and V. Frey. The semantic discrimination rate metric for privacy measurements which questions the benefit of t-closeness over l-diversity. *The 14th International Conference on Security and Cryptography (ICSC)*, pages 285–294, January 2017. doi: 10.5220/0006418002850294.

[113] Y. Song, L. Li, J. Zhang, and J. Yang. A method of enhanced t-closeness for privacy protection. *IEEE Proceedings of 2013 2nd International Conference on Measurement, Information and Control, Harbin*, pages 1491–1494, 2013. doi: 10.1109/MIC.2013.6758241.

[114] J. Soria-Comas and J. Domingo-Ferrer. Differential privacy via t-closeness in data publishing. *IEEE Eleventh Annual Conference on Privacy, Security and Trust, Tarragona*, pages 27–35, 2013. doi: 10.1109/PST.2013.6596033.

[115] J. Soria-Comas, J. Domingo-Ferrer, D. Sanchez, and S. Martinez. T-closeness through microaggregation: Strict privacy with enhanced utility preservation. *IEEE Transactions on Knowledge and Data Engineering*, 27(11):3098–3110, 2016. doi: 10.1109/TKDE.2015.2435777.

[116] P. Sreevani, P. Niranjan, and P. Shireesha. A novel data anonymization technique for privacy preservation of data publishing. *International Journal of Engineering sciences and Research Technology (IJESRT)*, 3(11):201–205, November 2014.

[117] S. B. Sriramoju. Analysis and comparison of anonymous techniques for privacy preserving in big data. *International Journal of Advanced Research in Computer and Communication Engineering*, 6(12):64–67, December 2017. doi: 10.17148/IJARCCE.2017.61212.

[118] W. Stallings. Cryptography and network security: Principles and practice. *Global Edition*, pages 1–767, 2014. URL `https://dl.hiva-network.com/Library/security/Cryptography-and-network-security-principles-and-practice.pdf`.

[119] U. States Department of Justice. Privacy technology focus group. *The Integrated Justice Information Systems (IJIS) Institute*, pages 1–83, 2006. URL `https://www.it.ojp.gov/documents/privacy_technology_focus_group_full_report.pdf`.

[120] C. Stergiou, A. Plageras, K. Psannis, and B. Gupta. Secure machine learning scenario from big data in cloud computing via internet of things network. *Gupta B., Perez G., Agrawal D., Gupta D. (eds) Handbook of Computer Networks and Cyber Security*, 2020. doi: 10.1007/978-3-030-22277-2_21.

[121] V. Susan and T. Christopher. Anatomisation with slicing: a new privacy preservation approach for multiple sensitive attributes. *SpringerPlus*, 5(1):1–21, July 2016. doi: 10.1186/s40064-016-2490-0.

[122] L. A. Tawalbeh and G. Saldamli. Reconsidering big data security and privacy in cloud and mobile cloud systems. *Journal of King Saud University–Computer and Information Sciences*, May 2019. doi: 10.1016/j.jksuci.2019.05.007.

[123] A. Tejero and I. de la Torre. Advances and current state of the security and privacy in electronic health records: Survey from a social perspective. *Journal of Medical Systems*, 36(5):3019–3027, 2012. doi: 10.1007/s10916-011-9779-x.

[124] Z. Tu, K. Zhao, F. Xu, Y. Li, L. Su, and D. Jin. Beyond k-anonymity: Protect your trajectory from semantic attack. *IEEE 14th International Conference on Sensing, Communication, and Networking (SECON), San Diego, CA, USA*, pages 1–9, 2017. doi: 10.1109/SAHCN.2017.7964921.

[125] Z. Tu, K. Zhao, F. Xu, Y. Li, L. Su, and D. Jin. Protecting trajectory from semantic attack considering k-anonymity, l-diversity and t-closeness. *IEEE Transactions on Network and Service Management*, 16(1):264–278, March 2019. doi: 10.1109/TNSM.2018.2877790.

[126] D. Veena. Data anonymization approaches for data sets using map reduce on cloud: A survey. *International Journal of Science and Research (IJSR)*, 3(4):308–311, 2014.

[127] M. Veeningen, S. Chatterjea, A. Zsófia, G. Spindler, E. Boersma, P. Van der Spek, O. van der Galiën, J. Gutteling, W. Kraaij, and T. Veugen. Enabling analytics on sensitive medical data with secure multi-party computation. *Proceeding of the Studies in health technology and informatics*, pages 76–80, 2018.

[128] G. Venifa Mini and K. S. Angel Viji. A comprehensive cloud security model with enhanced key management, access control and data anonymization features. *International Journal of Communication Networks and Information Security (IJCNIS)*, 9(2):263–273, August 2017.

[129] N. Volgushev, M. Schwarzkopf, B. Getchell, M. Varia, A. Lapets, and A. Bestavros. Conclave: secure multi-party computation on big data. *ACM Proceedings of the Fourteenth EuroSys Conference (EuroSys '19)*, pages 1–18, March 2019. doi: 10.1145/3302424.3303982. URL `https://dl.acm.org/doi/pdf/10.1145/3302424.3303982`.

[130] J. Walonoski, M. Kramer, J. Nichols, A. Quina, C. Moesel, D. Hall, C. Duffett, K. Dube, T. Gallagher, and S. McLachlan. Synthea: An approach, method, and software mechanism for generating synthetic patients and the synthetic electronic health care record. *Journal of the American Medical Informatics Association*, 25(3):230–238, March 2018. doi: 10.1093/jamia/ocx079. URL `https://storage.googleapis.com/synthea-public/synthea_sample_data_csv_apr2020.zip`.

[131] D. Wang, B. Guo, Y. Shen, S. Cheng, and Y. Lin. A faster fully homomorphic encryption scheme in big data. *IEEE Proceeding of the 2nd International Conference on Big Data Analysis, India*, pages 345–349, 2017. doi: 10.1109/ICBDA.2017.8078836.

[132] H. Wang, J. Han, J. Wang, and L. Wang. (l, e)-diversity–a privacy preserving model to resist semantic similarity attack. *Journal of Computers*, 9(1):59–65, 2014. doi: 10.4304/jcp.9.1.59-64.

[133] L. Wang and X. Li. A hybrid optimization approach for anonymizing transactional data. *ACM International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP)*, 9532:120–132, December 2015. doi: 10.1007/978-3-319-27161-3_11.

[134] T. Wang, Z. Zheng, M. H. Rehmani, S. Yao, and Z. Huo. Privacy preservation in big data from the communication perspective-a survey. *IEEE Communications Surveys and Tutorials*, 21(1), 2019. doi: 10.1109/COMST.2018.2865107.

[135] W. Wang, Y. Jiang, Q. Shen, W. Huang, H. Chen, S. Wang, X. Wang, H. Tang, K. Chen, K. E. Lauter, and D. Lin. Toward scalable fully homomorphic encryption through light trusted computing assistance. *ArXiv:1905.07766*, May 2019.

[136] T. Working party. Opinion 05/2014 on anonymisation techniques. *Article 29 Data Protection Working Party, White paper*, pages 1–37, April 2014. URL `https://ec.europa.eu/justice/article-29/documentation/opinion-recommendation/files/2014/wp216_en.pdf`.

[137] Y. Xie, Q. He, D. Zhang, and X. Hu. Medical ethics privacy protection based on combining distributed randomization with k-anonymity. *IEEE 8th International Congress on Image and Signal Processing (CISP), Shenyang, China*, pages 1577–1582, 2015. doi: 10.1109/CISP.2015.7408136.

[138] W. Xumeng, C. Jia-Kai, C. Wei, G. Huihua, C. Wenlong, L. Tianyi, and M. Kwan-Liu. A utility-aware visual approach for anonymizing multi-attribute tabular data. *IEEE International Journal Transactions on Visualization and Computer Graphics*, 24(1): 351–360, January 2018. doi: 10.1109/TVCG.2017.2745139.

[139] S. Yakoubov, V. Gadepally, N. Schear, E. Shen, and A. Yerukhimovich. A survey of cryptographic approaches to securing big-data analytics in the cloud. *IEEE Proceeding of the High Performance Extreme Computing Conference (HPEC), Waltham, MA United States*, pages 1–6, 2014. doi: 10.1109/HPEC.2014.7040943.

[140] Y. Ye, L. Wang, J. Han, S. Qiu, and F. Luo. An anonymization method combining anatomy and permutation for protecting privacy in microdata with multiple sensitive attributes. *IEEE International Conference on Machine Learning and Cybernetics (ICMLC)*, pages 404–411, 2017. doi: 10.1109/ICMLC.2017.8108955.

[141] F. Zhang, V. E. Lee, R. Jin, S. Garg, K. Raymond Choo, M. Maasberg, L. Dong, and C. Cheng. Privacy-aware smart city: A case study in collaborative filtering recommender systems. *Journal of Parallel and Distributed Computing*, 127:147–159, 2019. doi: 10.1016/j.jpdc.2017.12.015.

[142] J. Y. Zhang, P. Wu, J. Zhu, H. Hu, and F. Bonomi. Privacy-preserved mobile sensing through hybrid cloud trust framework. *IEEE Sixth International Conference on Cloud Computing*, pages 952–953, June 2013. doi: 10.1109/CLOUD.2013.108.

[143] W. Zheng, Z. Wang, T. Lv, Y. Ma, and C. Jia. K-anonymity algorithm based on improved clustering. *Proceeding of the International Conference on Algorithms and Architectures for Parallel Processing, China*, 11335:462–476, December 2018. doi: 10.1007/978-3-030-05054-2_36.

[144] W. Zheng, Y. Ma, Z. Wang, C. Jia, and P. Li. Effective l-diversity anonymization algorithm based on improved clustering. *Springer The 11th International Symposium on Cyberspace Safety and Security (CSS), Guangzhou, China*, 11983:318–329, 2019. doi: 10.1007/978-3-030-37352-8_29.

[145] H. Zhu, S. Tian, and M. Xie. Anonymization on refining partition: Same privacy, more utility. *The 2nd International Conference on Systems and Informatics (ICSAI), Shanghai*, pages 998–1005, 2014. doi: 10.1109/ICSAI.2014.7009431.

[146] H. Zhu, H. Liang, L. Zhao, D. Peng, and L. Xiong. ζ-safe (l,k)-diversity privacy model for sequential publication with high utility. *IEEE Access*, 7:687–701, 2019. doi: 10.1109/ACCESS.2018.2885618.