Privacy Preserving Data Publishing for Heterogeneous Multiple Sensitive Attributes with Personalized Privacy and Enhanced Utility

Jayapradha. J*, Prakash. M, Yenumula Harshavardhan Reddy

Department of Computer Science and Engineering, College of Engineering and Technology, Faculty of Engineering and Technology, SRM Institute of Science and Technology, SRM Nagar, Kattankulathur, 603203, Kanchipuram, Chennai, TN, India,
jayapraj@srmist.edu.in
Department of Computer Science and Engineering, College of Engineering and Technology, Faculty of Engineering and Technology, SRM Institute of Science and Technology, SRM Nagar, Kattankulathur, 603203, Kanchipuram, Chennai, TN, India,
prakashm2@srmist.edu.in
Department of CSE in specialisation with Big Data Analytics, College of Engineering and Technology, Faculty of Engineering and Technology, SRM Institute of Science and Technology, SRM Nagar, Kattankulathur, 603203, Kanchipuram, Chennai, TN, India,
yv4543@srmist.edu.in

ABSTRACT

In recent years, personal data availability has become vast, which leads to the concept of Privacy-preserving. Privacy-Preserving is an essential issue in all research fields. Many privacy methods are available for privacy-preserving data publishing (PPDP); however, it suffers from a few drawbacks i) couldn't use on heterogeneous multiple sensitive attributes; ii) Customized sensitivity requirements are ignored. To make the model satisfy both criteria, we have proposed a Quasi-Identifier-Multiple heterogeneous sensitive attribute (QI-MHSA) generalization algorithm. Our first work in this paper is to apply vertical partitioning in the microdata and partitioning it into i) Quasi-identifier bucket (QIB) ii) Multiple heterogeneous sensitive attribute bucket (MHSAB). Second, we have applied k-anonymity in QIB to anonymize the quasi-identifiers and *l*-diversity in MHSAB to anonymize the different sensitive attributes (categorical and numerical). A Top-down generalization method is adopted to generalize the categorical and numerical attributes. Finally, a new approach has been implemented in the personalized privacy of sensitive attributes. A flag is set for both categorical and numerical sensitive attributes based on their sensitivity requirements in MHSAB. The generalization approaches differ according to the level of sensitivity requirement. Extensive implementation is done on two datasets to compare the algorithm's efficiency and prove that our model has a better balance between privacy and utility.

Keywords: Heterogeneous sensitive attributes, Privacy-preserving, Sensitivity requirement, Utility

Correspondence:
Jayapradha. J
Department of Computer Science and Engineering, College of Engineering and Technology, Faculty of Engineering and Technology, SRM Institute of Science and Technology, SRM Nagar, Kattankulathur, 603203, Kanchipuram, Chennai, TN, India
Email: jayapraj@srmist.edu.in

ABBREVIATIONS

EMD-Earth Mover Distance, MHSAB-Multiple Heterogeneous Sensitive Attribute Bucket, PPF-Personalized Privacy Flag, QI-Quasi-identifier, QIB-Quasi identifier Bucket, QI-MHSA- Quasi-identifier-Multiple Heterogeneous Sensitive Attribute, RT- Relational Table, RT*-Anonymized Relational Table, SA-Sensitive Attribute.

INTRODUCTION

Data publishing is the action of emancipating the scientific research data in published form for various purposes. Researchers need data from multiple applications for analysis for their research. To facilitate the analysis phase of research, the release of data is significant. The government sectors and private sectors publish microdata for scientific research. Though information is released for scientific research purposes, taking care of individual privacy is a significant concern. The term Privacy-preserving is associated with the collection of data and the broadcasting of data. Privacy issues arise in numerous applications such as health care, smart city, social network, Cloud agriculture, etc. The main goal of privacy-preserving is to shield the sensitive information of the individual. The dataset released is typically stored in the Relational Table (RT). Attackers try to gain an individual's information by linking it with the external source of records available. The relationship between the attributes in RT is linked to classify the individual's record. Such attributes are termed as "Quasi-identifiers" (QI). The Relational table also has a critical attribute known as sensitive attributes. The utmost care is taken for the sensitive attribute (SA) such that it should not be disclosed to anyone 1. The data collected and broadcasted from various sectors contains an abundant source of information and knowledge. However, due to privacy concerns, the original Relational table (RT) is anonymized by applying multiple techniques. Due to the anonymization of data, the utility is degraded as the privacy concerns increases. So, a balance between utility and privacy is always needed due to these issues. The trade-off between utility and privacy is still a fundamental problem in privacy-preserving data publishing. Relational table (RT) is anonymized as a relational table (RT*), which results in utility loss, which may result in inaccurate solutions during the extraction of knowledge 2,3.

Most of the existing studies do not consider the heterogeneous multiple sensitive attributes and its linking relation, leading to privacy leakage. Sweeney proposed a model k-anonymity with a set of protection policies for deployment. Sweeney stated that removing the names of the people in the microdata released for research does not protect the data from adversaries. In his investigation study, nearly 87% of the US people can be easily re-identified by linking the quasi-identifiers (e.g., age, gender, zip code) of an individual, leading to the disclosing of sensitive attribute 4. Though k-anonymity was popular for its privacy model, it may lead to a few attacks, such as background and homogeneity attacks. A concept l-diversity was proposed to overcome the flaws in *l*-diversity. *l*-diversity ensures that the even person releasing the microdata for research purpose are not aware of the individual information 5. *l*-diversity makes a concrete *l* "well represented" values for the attribute that need not be disclosed (i.e., Sensitive attribute) and the makes the quasi identifier unpredictable in each equivalence class. t-closeness was proposed to overcome the limitations in l-diversity.

The data representation granularity is reduced in t-closeness. A measure of Earth Mover Distance (EMD) is used to calculate the t-closeness in all the equivalence classes. EMD is used as a distance metric to calculate the distance between all the table attributes and sensitive attributes. EMD measures differ for both numeric and categorical attributes 6. The three de-identification models from 4, 5, 6 are the fundamental models for Privacy-preserving. Various methods and distributed frameworks have been developed and used for big data.

According to the survey, the scalable approaches developed for big data are based on the fundamental approach of k-anonymity and l-diversity. Even a lot of improved version of k-anonymity and l-diversity was developed. Data publishing privacy is achieved by weakening the link between the Quasi identifier and individual (data owner) 7, 8. The original values of the quasi identifier (QI) and the sensitive attribute (SA) in a relation table (RT) shouldn't be removed unswervingly. Instead, the QI and SA can be interchanged with a range of values or ambiguous entries such that QI attributes values cannot be distinguished within the equivalence group 4. To protect the individual's privacy, extensive anonymization techniques are needed, which results in unacceptable information loss. Due to massive information loss, the performance evaluation of anonymized data will not have high accuracy. This paper proposes an approach to anonymizing the data by dividing the RT into two subparts. The first subpart table includes a quasi-identifier, and the second subpart table is composed of heterogeneous sensitive attributes. The balancing of both privacy and utility in the data to be published is an NP-Hard Problem 9. Our main idea is to divide the RT vertically into different subsets; each subpart contains the needed attributes with the same number of records in both the tables. In our model, three algorithms, the QIB generalization algorithm, the MHSAB generalization algorithm, PPF generalization algorithm, are used.

PRELIMINARIES

To better understand the paper's concept, basic definitions and fundamental concepts are discussed briefly at the beginning of the paper, followed by the privacy-preserving algorithm and utility measurement.

The Relational Table attributes are grouped into three following categories.

Quasi-identifier – The Quasi identifier attribute term was coined in 1986 by Tore Dalenius. Few attributes combine to make the quasi-identifier. The individuals can be re-identified by linking the external source with the quasi-identifier. For example, in RT combination of the attributes age, gender, zip code can form a quasi-identifier, which may be linked with other external records to re-identify the individual.

Sensitive Attribute – It represents the individual's information, which the data owner is unwilling to disclose. E.g., Salary

Insensitive attribute – The non-private information that is considered to be not sensitive. E.g., Gender

Before the data publishing, the three categories, 1. Quasi-identifier 2. Sensitive attribute 3. The insensitive attribute in the original microdata needs to be anonymized to protect privacy 4.

Definition 1 (Equivalence Class). The microdata is divided into a subclass set with the generalized records that have the same value on the subclasses' QI attributes. i.e., the tuples inside the equivalence class could not be distinguishable by the QI attribute.{y ∈ Y: y RT c} where c is an element belonging to Y. {y RT c}indicates that y and c have an equivalence relation. {y RT c} iff y and c belong to the same equivalence class

Definition 2 (k-anonymity[4]) A table is said to be k-anonymity if each individual's record detail during the table's release should not be distinguished from at least k-1 individual provided the same individual information should be in the release table. In k-anonymity, the probability of identifying an individual should not be greater than 1/k.

Definition 3(L-Diversity [5]) A table is said to be l-diversity if all the equivalence class have at least *l" well* represented" record values for all the sensitive attributes. In l-diversity, the probability of privacy leakage should not be greater than 1/*l.*

Definition 4 (Multiple Heterogeneous Sensitive Attribute Bucket (MHSAB). For a given relational table (RT), the categorical sensitive attribute and numerical sensitive attribute satisfy *l*-diversity, RT confronts the multiple heterogeneous sensitive attribute *l*-diversity.

Definition 5 (Quasi identifier Bucket (QIB)). For a given relational table (original database), the quasi-identifiers are generalized so that table satisfies the k-anonymity.

Definition 6 (Data Generalization). For a given attribute in RT, the record's original values are replaced with fuzzy interval range values. The outcome of the generalized data is coarse-grained data.

MOTIVATIONAL AND CHALLENGES

Challenge1 (Multiple heterogeneous sensitive attribute anonymity)

Most of the papers have worked on multiple sensitive attributes. The existing research concentrates on multiple sensitive attributes but fails to protect an individual's privacy when there are multiple heterogeneous sensitive attributes. A novel k-anonymity was proposed for multiple sensitive attributes and achieved record suppression with minimum data distortion 13.

Most of the paper assumes that microdata will have a specified column of sensitive attributes, either categorical or numerical 10,11,12. The real-world data are mostly complicated than we assume, where an individual will have more than one sensitive attribute values in the relational table. The sensitive attribute can also be either categorical or numerical type. Unfortunately, most of the existing methods have not concentrated on the heterogeneous multiple sensitive attributes. The models available for microdata with multiple sensitive attributes will not fit the heterogonous multiple sensitive attributes 14. i.e., an individual may have multiple sensitive attributes with different data types, either categorical or numerical in his record. For example, in electronic health records, a patient might have age, gender, zip code, race, marital status, income, and disease. The sensitive attribute income is numerical, and the sensitive attribute disease is categorical. Although few researchers had good progress 15, 16, 17 on the heterogeneous multiple sensitive attributes, balancing between the utility and privacy remains open to challenge. In Table 1, the original database with heterogeneous multiple sensitive attributes is represented. Let RT be the original database to be published for various purposes. Let RT has n number of attributes AT={AT_1, AT_2, AT_3…….AT_n} and the domains of attributes are {D[AT_1],D[$AT_{2]}$,D[AT_3]……D[AT_n]}respectively. A tuple tp∈RT, defined as tp=(tp[AT_1],tp{AT_2],tp[AT_3]…..tp[AT_n]) where tp[AT_i] (1<i<n) represents the attribute value of tp.

Table 1: Original Medical Database RT

Personal Identifier	**Quasi Identifier**			**Heterogeneous Multiple Sensitive Attribute**	
Name	**Gender**	**Age**	**Zipcode**	**Income**	**Disease**
1(Ben)	M	22	250100	35000	Flu
2(Jack)	M	23	250100	42000	Diabetes
3(Mary)	F	25	202000	20000	HIV
4(Joe)	F	35	203200	25000	Pneumonia
5(Boly)	M	28	151000	45000	Ebola
6(Jim)	M	21	151001	51000	Hypertension
7(Anna)	F	47	160250	28500	Covid19
8(Hary)	M	42	160255	35000	Bronchitis
9(David)	M	57	180350	70000	Flu
10(Kathe)	F	57	180000	65000	Diabetes

Table 2: 2-Anonymity Table RT*

Personal Identifier	**Quasi Identifier**			**Heterogeneous Multiple Sensitive Attribute**	
Name	**Gender**	**Age**	**Zipcode**	**Income**	**Disease**
1(Ben)	*	2*	250100	35000	Flu
2(Jack)	*	2*	250100	42000	Diabetes
3(Mary)	*	[20,40]	20***	20000	HIV
4(Joe)	*	[20,40]	20***	25000	Pneumonia
5(Boly)	*	[20,30]	15100*	45000	Ebola
6(Jim)	*	[20,30]	15100*	51000	Hypertension
7(Anna)	*	[40,50]	[16000,17000]	28500	Covid19
8(Hary)	*	[40,50]	[16000,17000]	35000	Bronchitis
9(David)	*	5*	180000	70000	Flu
10(Kathe)	*	5*	180000	65000	Diabetes

Table 2 represents a 2-anonymity table for data publishing. In 2-anonymity, the original table is partitioned into different equivalence class subsets where the individual record of the Quasi identifier in each group of the class is indistinguishable. Let's assume the intruder have background knowledge about the individual in the released data, then the sensitive attribute of an individual can be exposed. According to the k-anonymity, the probability of the privacy leakage in the k-anonymized table is 1/k, so in Table 2, the likelihood of privacy leakage is ½.

Challenge 2 (Personalized Privacy Flag (PPF))

Most of the existing generalization technique fails when applied to heterogeneous multiple sensitive attributes dataset. Just using the anonymization in table 2 will not protect the sensitive attribute of an individual. As per the research done, the k-anonymity will not cover the sensitive attributes effectively against reverse attack. To solve this problem (α, k), anonymity 18 proposes that sensitive attribute frequency in the conforming equivalence class is not greater than 1/ α. To overcome the above problem, we have adopted an approach of vertical partitioning, that partition our original table into two tables consisting of the same no.of. Records. In the first scenario table QIB, the quasi-identifiers (i.e., age, gender, zip code) are anonymized through k-anonymity. In the second scenario table, the *MHSAB* is anonymized (i.e., income and disease) by l-diversity. In our paper, to prevent the exposure of heterogeneous multiple sensitive attributes, we have adopted an approach of QIB and *MHSAB (definition 4&5).*

On the other hand, applying standard generalization techniques on different sensitive attributes will lead to excessive information loss. The existing privacy-preserving techniques accomplish anonymization on the whole dataset without checking for the individual personalized privacy. In 19, personalized privacy preservation was attempted to consider individuals' level of privacy. Later, many researchers focused on personalized privacy for multiple sensitive attributes 20, 21, but still achieving privacy without information loss is not achieved.

Definition 7(Low-level security requirement) Low-level requirement of the sensitive attribute indicates that the sensitive attribute's generalization is not needed. i.e., The low-level sensitive attributes have no importance of sensitivity requirement.

Definition 8(Middle-level security requirement) Middle-level requirement of the sensitive attribute indicates that it needs a certain level of generalization.i.e the middle-level sensitive attribute has a low level of importance of sensitivity requirement.

Definition 9(High-level security requirement) High-level requirement of the sensitive attribute indicates that it needs a high level of generalization. i.e., the high-level sensitive attribute has a greater level of importance of sensitivity requirement.

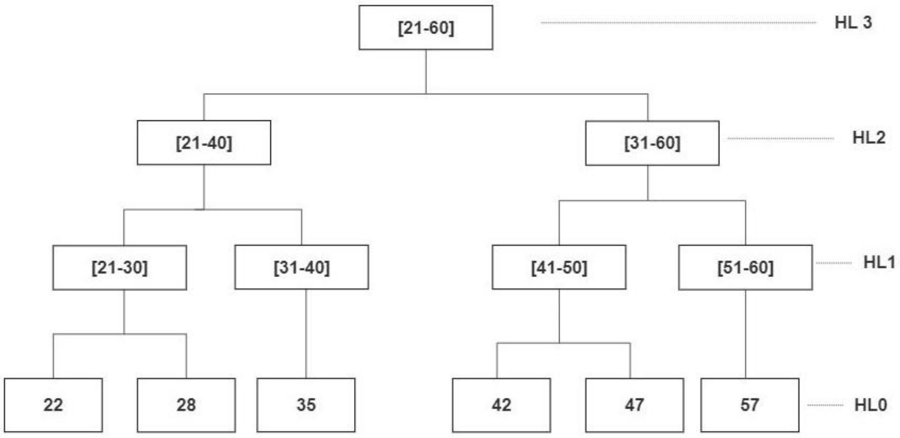


Figure 1: Generalization Hierarchy tree of attribute age

The sensitive attributes are categorized into three levels 1. Low 2. Middle 3. High. By applying the different levels of privacy according to the individual perspective, the overall generalization doesn't happen, which leads to high information loss.

PERSONALIZED PRIVACY FLAG SET

In PPS, we fix a flag for the different sensitivity requirements as sf∈ [0,1,2]. A column "Flag set" is introduced in our MHSAB. The Flag set column has two parameters 1. the sensitivity of the income, 2. the sensitivity of the disease. In our paper, we have set the flag for the categorical sensitive attribute as follows [Flu, Pneumonia, and Bronchitis] ∈0, [Diabetes, Hypertension] ∈1, [HIV, Ebola, Covid19] ∈2. The flag set for numerical sensitive attribute are [Income<=30,000] ∈0, [30,000< Income<50000] ∈1, [Income>50000] ∈2. Table 3 represents the quasi identifier bucket after generalization. Table 4 represents the Sensitive table with Flag for Income and Disease. The first parameter in the flag set column indicates Income sensitivity, and the second parameter indicates the disease sensitivity. The Flag setlist consists of n sensitive attributes $\{sf_1, sf_2, sfn\}$. In our example, we have only two sensitive attributes with three flag set for each where sf_1=0, sf_2=1, sf_2=2.

Table 3: QIB after generalization

Personal Identifier	**Heterogeneous Multiple Sensitive Attribute**		**Flag Set**
Name	**Income**	**Disease**	
1(Ben)	35000	Flu	(1,0)
2(Jack)	42000	Diabetes	(1,1)
3(Mary)	20000	HIV	(0,2)
4(Joe)	25000	Pneumonia	(0,0)
5(Boly)	45000	Ebola	(1,2)
6(Jim)	51000	Hypertension	(2,1)
7(Anna)	28500	Covid19	(0,2)
8(Hary)	35000	Bronchitis	(1,0)
9(David)	70000	Flu	(2,0)
10(Kathe)	65000	Diabetes	(2,1)

Table 4: MHSAB with a Flag set

Personal Identifier	**Quasi Identifier**		
Name	**Gender**	**Age**	**Zipcode**
1(Ben)	*	2*	250100
2(Jack)	*	2*	250100
3(Mary)	*	[20,40]	20***

4(Joe)	*	[20,40]	20***
5(Boly)	*	[20,30]	15100*
6(Jim)	*	[20,30]	15100*
7(Anna)	*	[40,50]	[16000,17000]
8(Hary)	*	[40,50]	[16000,17000]
9(David)	*	5*	180000
10(Kathe)	*	5*	180000

Here, a small survey is conducted among the people to categorize disease and income sensitivity requirements. As per the survey results, the user doesn't mind disclosing flu, bronchitis, Pneumonia, so we have set the sensitive flag sf1=0. Few users don't want to disclose the disease they have fully, so in that case, the sf_2=1. Few people strictly avoid disclosing the disease they persist, like HIV, Ebola so sf_2=2. For numerical attributes also, the same flag set is followed.

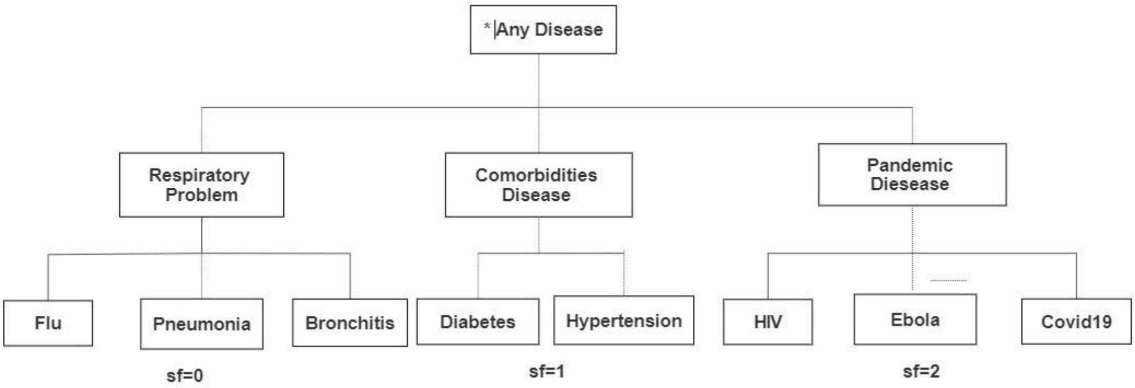


Figure 2: Generalization Hierarchy tree of attribute disease

Table 5: RT^a after QI- MHSA Generalization Algorithm

Personal Identifier	**Quasi Identifier**			**Heterogeneous Multiple Sensitive Attribute**	
Name	**Gender**	**Age**	**Zip code**	**Income**	**Disease**
1(Ben)	*	2*	250100	>=30k	Flu
2(Jack)	*	2*	250100	>=30k	comorbities
3(Mary)	*	[20,40]	20***	<30k	*
4(Joe)	*	[20,40]	20***	<30k	Pneumonia
5(Boly)	*	[20,30]	15100*	>=40k	*
6(Jim)	*	[20,30]	15100*	>=40k	comorbities
7(Anna)	*	[40,50]	[16000,17000]	<40k	*
8(Hary)	*	[40,50]	[16000,17000]	<40k	Bronchitis
9(David)	*	5*	180000	>50k	Flu
10(Kathe)	*	5*	180000	>50k	comorbities

GENERALIZATION HIERARCHY TECHNIQUE

We propose a different generalization hierarchy cut technique to generalize the sensitive attributes and the quasi identifier. We are dealing with both categorical and numerical attributes in the sensitive attribute. If the standard approach is applied for both categorical and numerical attributes, there will be a lot of data distortion, leading to high information loss. Earlier, individual personalized privacy was not accounted for, so the anonymization of attributes leads to unnecessary generalization even for the low-level sensitivity requirement individuals. In 19, personalized privacy was considered on the categorical attribute, which results in less data distortion. Our main focus in this paper is balancing privacy and utility. We have utilized the Top-down generalization 22 to generalize both numerical and categorical attributes. The data generalization in the anonymization process is the core part of the privacy-preserving algorithm. Most of the generalization techniques are categorized into two types of attributes i) Categorical ii) numerical, as shown in Figure 1 and Figure 2. A generalization hierarchy is established to prevent information loss and the structure of the data. To cut the hierarchy tree into different partitions, we have used the concept "hierarchy cut" 23. Our main motive is to generate the least common leaf node in a numerical generalization hierarchy tree that can cover all the different generalization levels' values. For example, 22 years and 28 years can be generalized in the interval [21, 30], as shown in Figure 1. The HL in figure 1 denotes the hierarchical level of the tree. In categorical attribute generalization, if the levels of hierarchy go higher, the loss of information is also increased so, there should be a limited number of levels in the generalization hierarchy tree. The categorical hierarchy tree's primary goal is to identify the least common cut to cover all the attribute values. For example, viral bronchitis and flu can be generalized to Respiratory problems, and the respiratory problem can be generalized to common attribute disease, as shown in figure 2.

The common strategy followed for numeric and categorical attribute generalization hierarchy are i) For numeric attribute, the original values of the records should be replaced with the range of values where the original value falls in ii) For categorical attribute, each value should be generalized to a least common type value which can cover the wider range of other values in the original table. Both categorical and numerical attribute hierarchy tree stores the sensitive values in the leaf node. As we follow the PPF strategy in our paper, the lowest sensitivity needs not to be generalized i.e. (people don't mind disclosing flu and bronchitis), which prevents the data distortion for that particular records. In general, the low sensitivity requirement can be eliminated, and the middle sensitivity requirement needs less focus (i.e., few people don't mind disclosing about diabetes, but few minds revealing). So, such a middle level can be generalized by the traditional generalization rules. The high sensitivity requirement needs to be concentrated much (i.e., people never disclose Ebola or HIV). This strategy of PPS helps to reduce the time as well the information loss.

COROLLARY

Given a QI- MHSA Generalized published dataset RT^a, the probability of privacy leakage in both QIB and MHSAB should not be more than RPL (tp)=max(1/k,1/l), where RPL is Risk of privacy-preserving

Proof In the publishing dataset $RT^{a,}$ the intruder can easily identify the individual record and the individual's sensitive value by linking the Quasi identifier. In k-anonymity, the probability of privacy leakage in the QIB is 1/k, and the probability of privacy leakage in MHSAB through the relational table (RT) is 1/*l.* The overall disclosure of the risk of privacy leakage should be composed of heterogonous multiple sensitive attribute disclosure and quasi-identifier disclosure. The overall risk privacy leakage(RPL) is the maximum value of both QIB and MHSAB : $RPL(tp)=max\{RPL(tp_{QIB}), RPL(tp_{MHSAB})= max(1/k,1/l)$.where $tp_{QIB} \in QIB$ and $tp_{MHSAB} \in MHSAB$.

For the corollary's better explanation, we can take table 1, the original database (RT), and table 5, the data of generalized data (RT^a) to be published where k=2 and l=2. There five equivalence classes with two records in each group. So, income is replaced with a range of values and disease with the generalized value if the sensitivity requirement is high or medium. In RT^a, let's take the record 1 and 2 of table 5 for the disease. Record 1 does not need generalization, and record 2 is replaced with the generalized value of diabetes, as shown in figure 2, so the probability of privacy leakage risk is 1/*l* (0.5). The likelihood of privacy leakage through QID is 1/k. So, the overall risk of privacy leakage is maximum (1/k, 1/*l)*

QI- MHSA GENERALIZATION ALGORITHM

In the previous section, we have seen proof of the proposed model. In this section, we are going to discuss the algorithms for QIB and MHSAB. We have undergone different generalization for the categorical and numerical attributes to minimize the information loss, as in Figures 1 & 2. Most of the existing algorithms apply the anonymization technique on the original database without partitioning the data. To deal with balancing both privacy and utility, we have proposed QI- MHSA Generalization Algorithm.

In this section, we have divided our work into three parts. Algorithm 1 adopts the concept of partitioning the table into two 1. QIB 2. MHSAB. We have adopted 21 method for partitioning the table vertically. In QIB generalization algorithm, Function Split divides the table into two parts vertically. Function k_division applies the k-anonymity and divides the records into equivalence class. Function _categorical and Function _numerical are fetching the numerical and categorical values from the QIB. For the categorical attribute, the unique attributes are listed, and an index is created [line 7-14]. The numerical attribute, range of values for the original data is listed, and an index is created [line15-21].

In MHSAB generalization algorithm, the MHSAB, is divided into equivalence groups by using Function_equivalence. The sensitivity level is checked, and the top-down approach 22 is used in the generalization hierarchy [line 4-10]. Before applying *l*-diversity, the length of the unique attribute is checked to proceed with the anonymization technique. [line 11-17]

Algorithm 1 QIB generalization algorithm (RT, k)

```
1.Input: RT = Table
2.Output: k anonymized QIB

3.QIB, MHSAB = Function_Split(RT)
4.QIBsub = Function k_division(QIB,k)
  #Using k_division function it splits into groups of tuples with size of k
number of tuples in each group for anonymization

#QIB TABLE
5.QIBcat = Function_categorical(QIBsub)
  #Fetching list of categorical attributes of QIBsub
6.QIBnum = Function_numerical(QIBsub)
  #Fetching list of numerical attributes of QIBsub

#Applying k-anonymity for categorical variables in Quasi attributes

 7.for each attribute in QIBcat
 8.   k_cat = list(unique(attribute))
```

```
end for

for each attribute in QIBcat
    for each value in the attribute
        value = k_cat[index(attribute)]
    end for
end for

#Applying k-anonymity for numerical variables in Quasi attributes

for each attribute in QIBnum
    k_num = list(range(attribute))

for each attribute in QIBnum
    for each value in attribute
        value = k_num[index(attribute)]
    end for
end for
```

Algorithm 2 MHSAB generalization algorithm (MHSAB, l)

```
Input: MHDSAB = Table
Output: l-diversity MHSAB
# l-diversity for sensitive attributes in MHSAB with the value of 'l'
MHSABsub = Function_equivalence (Function_partition (MHSAB))
#partition divides the MHSAB table to subsets to pass each subset into equivalence function to pull out equivalence class
#Equivalence classes of all our sensitive attributes into MHSABsub
#Level of sensitivity requirement are [HIGH, MEDIUM, LOW]
# If the level of sensitivity requirement is "LOW," then the record can be ignored.
#If the level of sensitivity requirement is "MEDIUM," traditional generalization technique is used for anonymization
#If the level of sensitivity requirement is high, generalization is made by the hierarchical tree method.
for each attribute in MHSABsub
    for each value in attribute
        if Function sensitivity(value) is HIGH
            value = Function Top_down(attribute)
            #Sensitivity returns the level of sensitivity requirement of value as per user choice
            #Top-down return the tree value of sensitivity ranges for the attribute in order to ensure more privacy to the value
            #Applying range for high sensitivity attribute values to give more privacy
        end if
    end for
end for
#l-diversity
for each attribute in MHSABsub
    if length(unique(attribute)) > l
        return MHSABsub
    else
        return Function_partition(MHSABsub)
        #This function adds on more tuples or groups more tuples in partition MHSABsub in order to increase unique variables in partition MHSABsub to get the length(unique(attribute))>l
    end if
end for
```

In the PPF generalization algorithm, the flag set is listed. Function assign flag is used to assign a flag for each attribute; the flag value is set according to sensitivity requirement, i.e., low=0, medium=1, high=2[lines 3-12]. Finally, the Function_join is used to join both QIB and MHSAB. The RT^a is the table to be released for publishing, which is composed of QIB, MHSABa s shown in table 5.

Algorithm 3 PPF generalization algorithm (MHSAB, l)

```
1.Input: MHDSAB = Table
2.Ouput: Privacy flag assigned.
#Personalised privacy flag set
#sensitivity requirement [0, 1 ,2]
3.flag_set = list ()
```

```
for each attribute in S
  for each value in attribute
    Function assign_flag(value)
    if value belongs to HIGH
      flag_set[index(value)] = assign_flag(2)
    else if value belongs to MEDIUM
        flag_set[index(value)] = assign_flag(1)
     else
       flag_set[index(value)] = assign_flag(0)
    end if
  end for
end for

RT^a = Function_join(QIBcat,QIBsub)
#RT^a table is ready for publishing as the publishing data set
return RT^a
```

EXPERIMENT AND RESULT ANALYSIS

In this section, we have evaluated the efficiency of the QI-MHSA Generalization Algorithm. We mainly focus on privacy-preserving and information loss in our experiment. We have compared our algorithm with Top-Down 22. In our work, we have used an Adult dataset 24; it contains 15 attributes. We removed few attributes and kept age, zip code, marital status, race, education as quasi identifier and income(numerical), and occupation(categorical) as a sensitive attribute. There are around 44,000 records. We have also worked with the Patients List Andhra Pradesh dataset, which has 13 attributes. Unfortunately, the records are very less around 200. So, we used the synthea tool to generate synthetic data. Around 50,000 records are generated for our work. We removed a few columns like sickness_id, pid, etc. The sensitive attribute is DiagnoseCode(categorical) and DateJoined(numerical). The sensitivity flag for diagnose code is set according to the disease they have diagnosed. We have also implemented our algorithm with different k and *l* values and found that our algorithm is effective. The privacy loss and utility loss for different k and *l* values are shown in the experiment section.

Information Loss

The information loss for the anonymized table (publishing table) RT^a can be calculated by the below equation. To measure the data distortion more effectively, we use a different method to calculate the numeric and categorical information loss. We have used a generalized information loss formula 25, 26, 27, 28 to evaluate the utility loss. Let the Q = {A$_1$, A2, A$_3$, An} where the set of quasi-identifiers in QIB. Let the numerical sensitive attribute be N_i^{sf}. Let RT be the original table and RT* is the anonymized table, so the information loss of RT* is

$$\mathrm{InfolossN}^{\mathrm{sf}} = \frac{\max(N^{sf}) - \min(N^{sf})}{\max - \min} \qquad (1)$$

The information loss of numerical attribute in the anonymized table is calculated using (1). The max (N^{sf}), min (N^{sf}) represents the maximum and minimum range of numerical attribute $N_i^{sf} \in$ QI. The categorical sensitive attribute is C_i^{sf}. Let RT be the original table and RT* is the anonymized table, so the information loss of RT* is

$$InfolossC^{sf} = \frac{\mathrm{NL}_{\mathrm{n}}(n_{root}) - 1}{NL_n(H)} \qquad (2)$$

The information loss of categorical attribute in the anonymized table is calculated using (2). $NL_n(n_{root})$ represents the leaf nodes in the primary root node's subtree, $NL_n(H)$ is the leaf node in the whole hierarchy tree H. The tuple tp in table T, tp ∈ RT*.

The general information loss of tuple tp in the anonymized table (RT*) is defined as

$$Infoloss(tp) = \sum_{i=1}^{d} InfolossA_i \qquad (3)$$

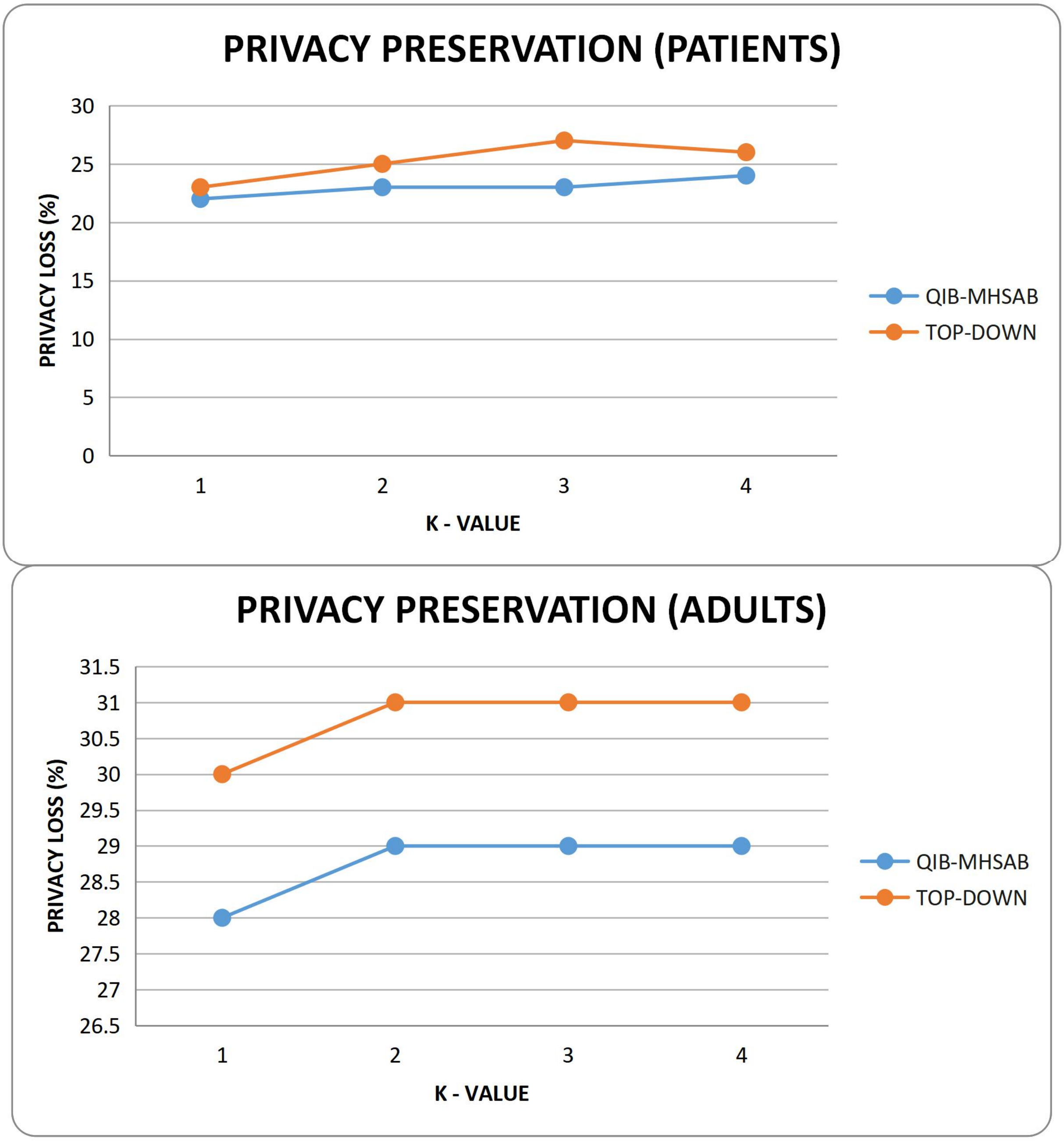
The above (3) is to calculate the single record information loss, whereas the information loss for the whole table RT* is

$$Infoloss_{(RT*)} = \frac{\sum_{tp \in RT*} InfolossN^{sf}}{Num.of.\mathrm{Re}\,c(RT*)} \qquad (4)$$

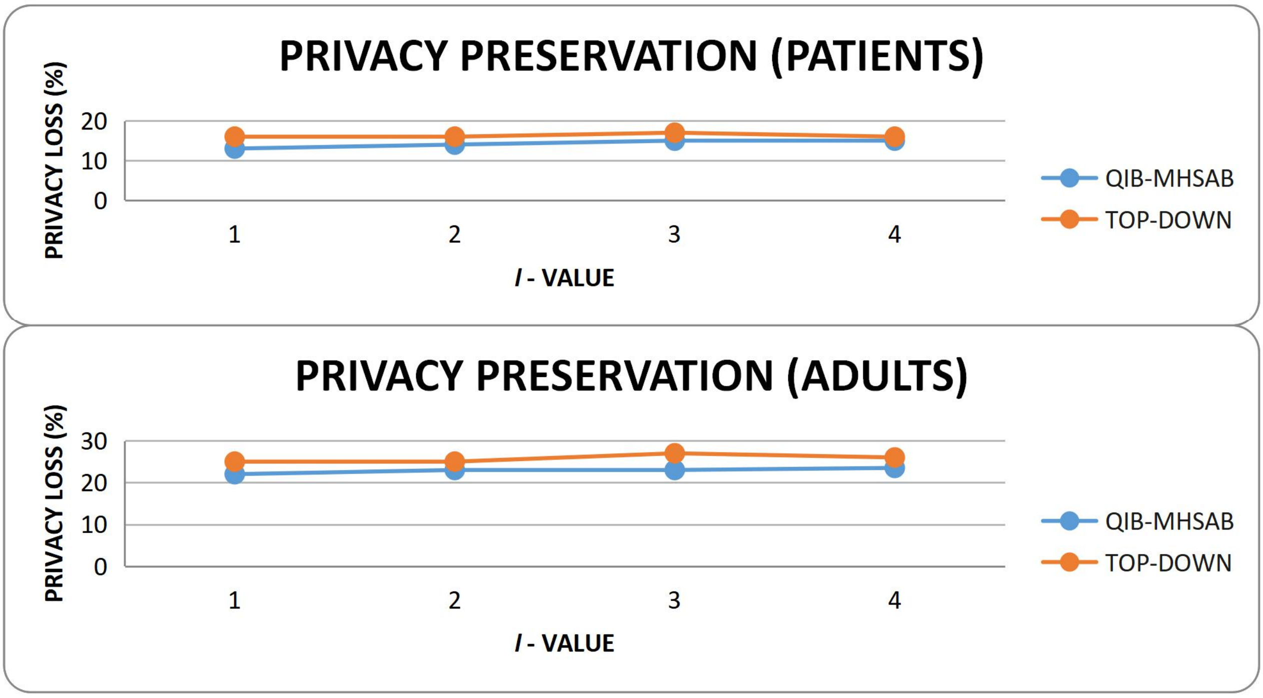
We use $\mathit{Infoloss}\mathrm{N}^{sf}$ $InfolossC^{sf}$ for measuring the information loss of categorical and numerical attributes. Infoloss (tp) is used to calculate the information loss in each record of the anonymized table. Infoloss $_{(RT*)}$ is the final formula to calculate the overall information loss in the generalized table.

Results of Experiment

We have conducted our experiment in two datasets, Adult (real-world dataset) and Patient (synthetic dataset). We analyzed the privacy and information loss by varying the k and l values. We can notice that our algorithm performed well than the Top-down algorithm. The Top-down algorithm is developed mainly for the relational dataset. We could see that our algorithm has less privacy loss in the Adult dataset than the top-down in Figures 3b and 3d for different k and l values. We could also notice that the real-world adult dataset is showing significant results than the patient dataset. Although our algorithm has lesser privacy loss in the patient dataset than top-down, our algorithm is a bit closer to the top-down algorithm. The proposed algorithm gives a less privacy loss in k-anonymity than the *l*-diversity, as shown in Figures 3a and 3b. So, we can also conclude that our QI-MHSA algorithm works well with real-world datasets.



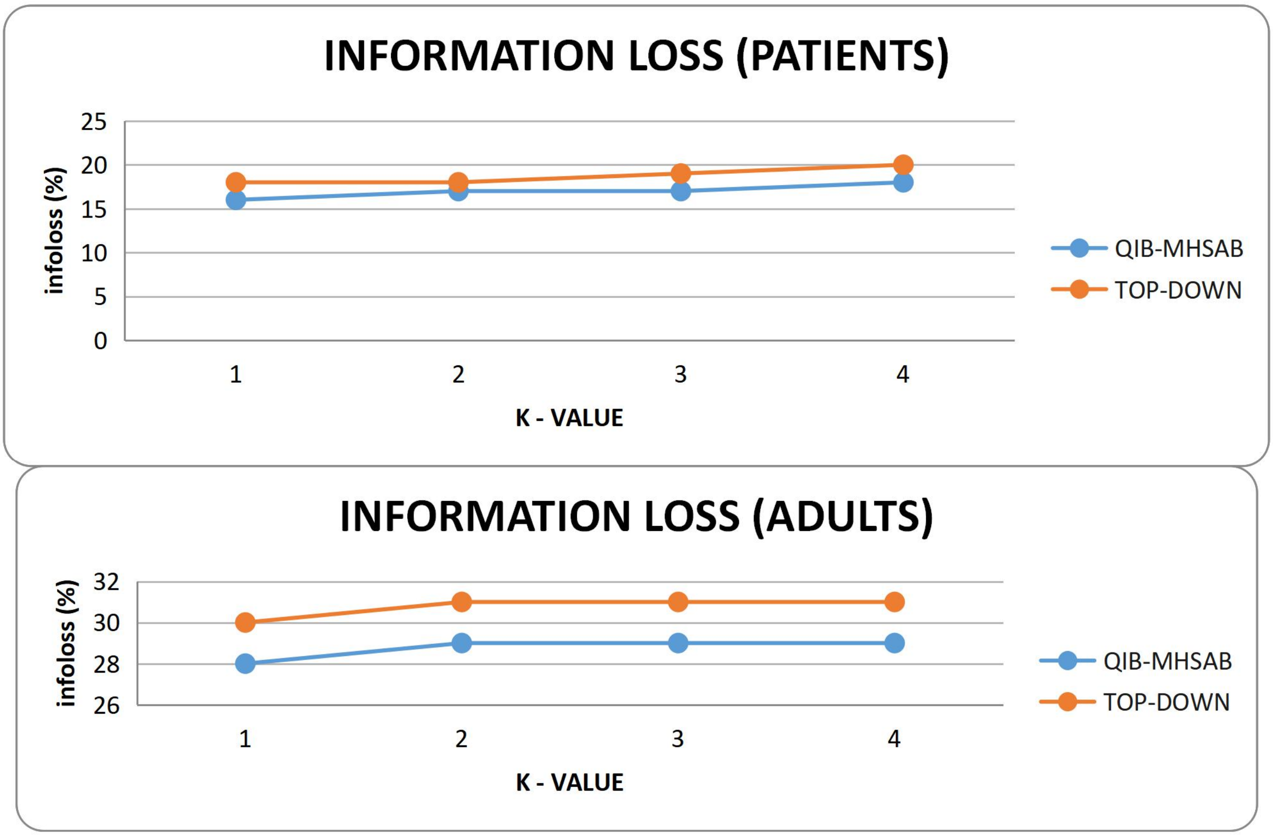
a) Privacy of Patient for different k values. b) Privacy of Adult for different k values.



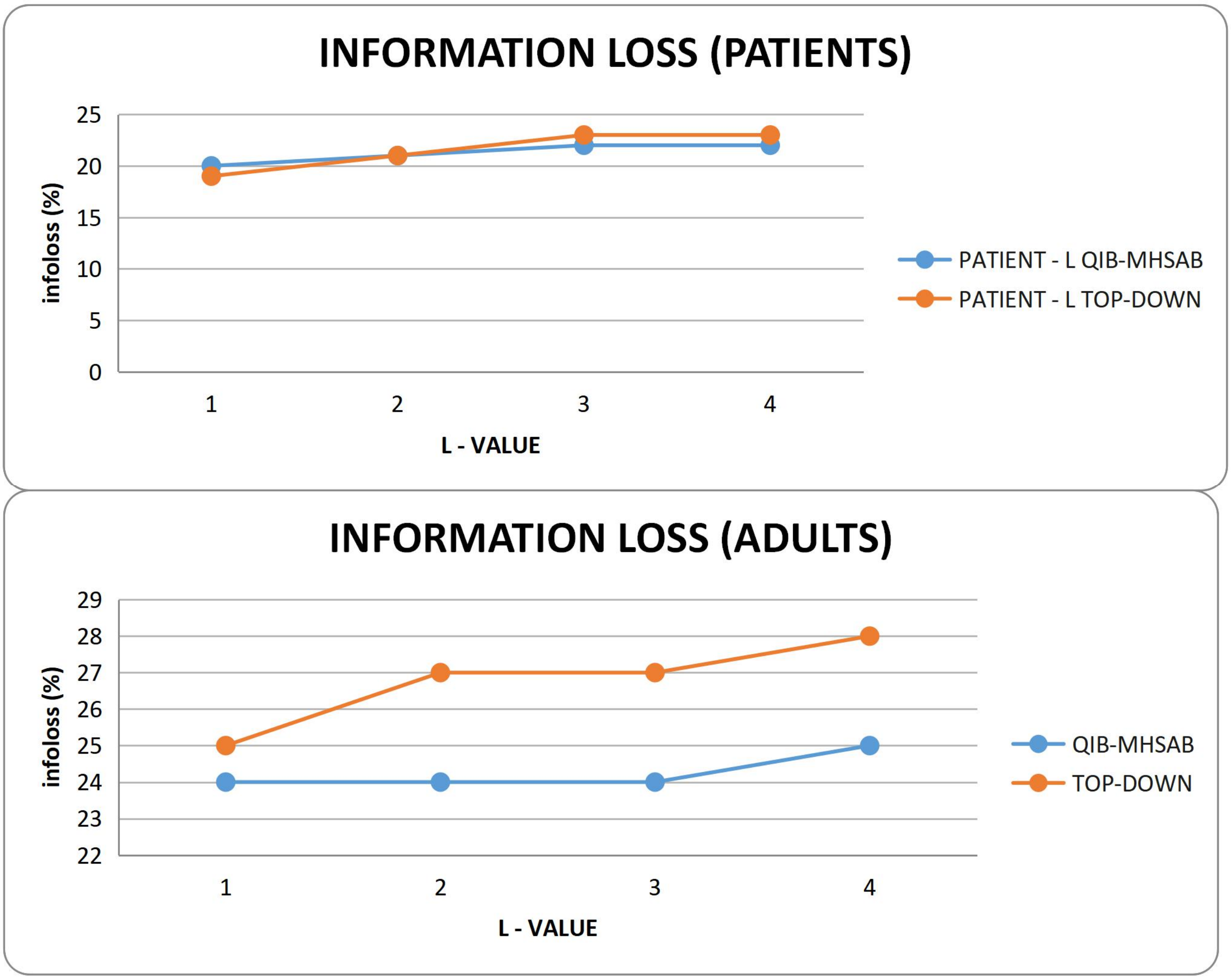
c) Privacy of Patient for different *l* values. d) Privacy of Adult for different *l* value.

Figure 3: Privacy loss when varying k and *l*.

The other parameter, information loss, is also measured. In Figures 4b and 4d, there is much difference in our algorithm and top-down algorithm's information loss. Our algorithm results in less information loss compared to top-down. In figures 4a and 4c, our algorithm values are a bit closer to top-down. We can also notice that the information loss measure works effectively with the real-world dataset than the synthesized data. Our algorithm result shows that information loss is less in the adult dataset with different k and l is less than top-down, as shown in figure 4b and 4d. Although our algorithm gives lesser information loss in the patient dataset than the top-down algorithm, as shown in Figures 4a and 4c, efficiency is a bit closer to the top-down algorithm. On real-world dataset, our QI- MHSA generalization algorithm performs better than top-down algorithm.



a) Infoloss in patient for different k values b) Infoloss in Adult for different k values.



b) Infoloss in the patient for different *l* values d) Infoloss in Adult for different *l* values

Figure 4: Infoloss when varying k and *l.*

CONCLUSION AND FUTURE WORK

This paper has proposed a QI-MHSA generalization model with a privacy-preserving flag to handle the challenge of multiple heterogeneous sensitive attribute generalization in the real-world dataset. Our paper's first work is to partition the microdata vertically into two buckets i) QIB ii) MHSAB. Second, we have focused on applying different anonymity techniques for QIB and MHSAB. As per approach, we have k-anonymity and *l*-diversity used for QIB and MHSAB, respectively. The main focus of our paper is to customized privacy for each individual. We have introduced a concept sensitive level flag, and the flag [0, 1, 2] is set according to the sensitivity requirement of sensitive attributes. Experiments conducted on the real-world dataset show that our proposed generalization algorithm is effective in both privacy-preserving and information loss. However, this is the beginning of our research; we have compared it with a top-down algorithm and proved our proposed algorithm gives a better balance in utility and privacy. Though we have generalized the sensitive attribute by introducing a concept of PPF and achieved good results, the flag set for the sensitivity level of records may give a clue for the adversary to distinguish between high, medium, and low sensitivity levels. Therefore, improvising customized personal privacy is our main focus of future research. We can also focus on privacy-preserving for stream data in the future as we focus only on the static data now.

REFERENCES

1. R.Mahesh, T. Meyyappan, "Anonymization Technique through Record Elimination to Preserve Privacy of Published Data," Proceedings of the International Conference on Pattern Recognition, Informatics and Mobile Engineering, 2013, 328-332.
2. Wensheng Gan, Jerry Chun-Wei, Han-Chieh Chao, Shyue-Liang Wang, and Philip S. Yu, "Privacy Preserving Utility Mining: A Survey", IEEE International Conference on Big Data (Big Data), 2018, 2617-2626.
3. Savitha Sam Abraham Deepak P, Sowmya S Sundaram, Fairness in Clustering with Multiple Sensitive Attributes, Proceedings of the 23rd International Conference on Extending Database Technology (EDBT), 2020, 1-15.
4. L. Sweeney, "k-anonymity: a model for protecting privacy", International Journal on Uncertainty, Fuzziness and Knowledge-based Systems,2002, 557-570.
5. A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. ℓ-diversity: Privacy beyond k-anonymity. Available at http://www.cs.cornell.edu/~mvnak, 2005.
6. Ninghui Li, Tiancheng Li, Suresh Venkatasubramanian, 2007. t-Closeness: Privacy Beyond k-Anonymity and l-Diversity, IEEE 23rd

International Conference on Data Engineering, 106-115,

7. Brijesh B. MehtaUdai Pratap Rao, Improved l-diversity: Scalable anonymization approach for PrivacyPreserving Big Data Publishing, Journal of King Saud University –Computer and Information Sciences, 2019, 1-8.
8. Loukides, G., & Shao, J, Preventing range disclosure in k-anonymized data. Expert Systems with Applications, 2011, 38(4), 4559–4574.
9. Razaullah Khan, Xiaofeng Tao, Adeel Anjum, Haider Sajjad, Saif ur Rehman Malik, Abid Khan, and Fatemeh Amiri, Privacy Preserving for Multiple Sensitive Attributes against Fingerprint Correlation Attack Satisfying c-Diversity, 2020,1-18.
10. Razaullah, Yan Jia, Weihong Han, A New k-anonymity Algorithm towards Multiple Sensitive Attributes, 2012 IEEE 12th International Conference on Computer and Information Technology,768-772
11. Tong Yi and Minyong Shi, Privacy Protection Method for Multiple Sensitive Attributes Based on Strong Rule, Hindawi Publishing Corporation, Mathematical Problems in Engineering, 2015, 1-4.
12. Yuelei Xiao and Haiqi Li, Privacy-Preserving Data Publishing for Multiple Sensitive Attributes Based on Security Level, MDPI/Information,2020,1-27.
13. Wang, J. A novel anonymity algorithm for privacy-preserving in publishing multiple sensitive attributes. Res. J. Appl. Sci. Eng. Technol. 2012, 4, 4923–4927.
14. Tehsin Kanwal, Sayed Ali Asjad Shaukat, Adeel Anjuma, Saif ur Rehman Malik , Kim-Kwang Raymond Chooc,, Abid Khana , Naveed Ahmade , Mansoor Ahmada , Samee U. Khanf, Privacy-preserving model and generalization correlation attacks for 1:M data with multiple sensitive attributes, Information Sciences 488, 2019, 238-256.
15. Yi,T. Shi M, "Privacy protection method for multiple sensitive attributes based on strong rule", Mathematical Problems in Engineering , 2015, 1-14.
16. Radha,D Valli Kumari, V., “Bucketize: protecting privacy on multiple numerical sensitive attributes”, Advances in Computational Sciences and Technology, 2017, 10(5), 991-1008.
17. S. A. Onashoga, B. A. Bamiro, A. T. Akinwale & J. A. Oguntuase , “KC-Slice: A dynamic privacy-preserving data publishing technique for multisensitive attributes”, Information Security Journal: A Global Perspective, 2017, 26(3), 121-135.
18. Liu X, Xie Q, Wang L Personalized extended (alpha, k)-anonymity model for privacy preserving data publishing, ConcurrComput Pract Exp 29(6), 2017.
19. M. Prakash and G. Singaravel, Haphazard, enhanced haphazard and personalised anonymisation for privacy preserving data mining on sensitive data sources, International Journal of Business Intelligence and Data Mining, Vol 13, 456-474, 2018.
20. QuinGhai L, Hang Shen and Yingpeng Shang, “A Privacy-Preserving Data Publishing Method for Multiple Numerical Sensitive Attributes via Clustering and Multisensitive Bucketization” IEEE, Beijing-2014.
21. Jianmin Hang, Fangwei Luo, Jianfengui L., and H. Peng, “SLOMS: A Privacy Preserving Data Pulishing Method for Multiple Sensitive Attributes Microdata”, JOURNAL OF SOFTWARE, 8(12),2013.
22. Sopaoglu U, Abul O, A top-down k-anonymization implementation for apache spark, IEEE international conference on big data (big data), 2017, 4513–4521.
23. Acs G, Achara JP, Castelluccia C, Probabilistic km-anonymity efficient anonymization of large set-valued datasets, IEEE international conference on big data (Big Data), 2015, 1164–1173.
24. https://archive.ics.uci.edu/ml/index.php
25. Cao J N, Karras P, Kalnis P, Tan K L. SABRE: A sensitive attribute bucketization and redistribution framework for t-closeness. The VLDB Journal, 2011, 59-81.
26. M Prakash, G Singaravel, An approach for prevention of privacy breach and information leakage in sensitive data mining, Computers & Electrical Engineering, 45, 134-140, 2015
27. Ghinita G, Karras P, Kalnis P, Mamoulis N. Fast data anonymization with low information loss. In Proc. the 33rd International Conference on Very Large Data Bases, 2007, 758-769.
28. Rong Wang, Student Member, CCF, Yan Zhu1, CCF, Tung-Shou Chen, and Chin-Chen Chang , Privacy-Preserving Algorithms for Multiple Sensitive Attributes Satisfying t-Closeness, JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 33(6), 2018 1231–1242 .