information

MDPI

Article

Privacy Preserving Data Publishing for Multiple Sensitive Attributes Based on Security Level

Yuelei Xiao [1,2,*] and Haiqi Li [1]

[1] School of Modern Posts, Xi'an University of Post and Telecommunications, Xi'an 710061, China; lihaiqi1994@163.com
[2] Shaanxi Provincial Information Engineering Research Institute, Xi'an 710075, China
* Correspondence: xiaoyuelei@xupt.edu.cn; Tel.: +86-02987303602

Received: 24 January 2020; Accepted: 20 March 2020; Published: 22 March 2020

Abstract: Privacy preserving data publishing has received considerable attention for publishing useful information while preserving data privacy. The existing privacy preserving data publishing methods for multiple sensitive attributes do not consider the situation that different values of a sensitive attribute may have different sensitivity requirements. To solve this problem, we defined three security levels for different sensitive attribute values that have different sensitivity requirements, and given an L_{sl}-diversity model for multiple sensitive attributes. Following this, we proposed three specific greed algorithms based on the maximal-bucket first (MBF), maximal single-dimension-capacity first (MSDCF) and maximal multi-dimension-capacity first (MMDCF) algorithms and the maximal security-level first (MSLF) greed policy, named as MBF based on MSLF (MBF-MSLF), MSDCF based on MSLF (MSDCF-MSLF) and MMDCF based on MSLF (MMDCF-MSLF), to implement the L_{sl}-diversity model for multiple sensitive attributes. The experimental results show that the three algorithms can greatly reduce the information loss of the published microdata, but their runtime is only a small increase, and their information loss tends to be stable with the increasing of data volume. And they can solve the problem that the information loss of MBF, MSDCF and MMDCF increases greatly with the increasing of sensitive attribute number.

Keywords: privacy preserving data publishing; multiple sensitive attributes; sensitivity requirements; security level; maximal security-level first (MSLF)

1. Introduction

In recent years, different organizations such as governments, hospitals and other institutions have published more and more microdata. Microdata plays a key role in data analysis, data mining and scientific research. However, publishing microdata unavoidably exposes the privacy of the individual. To protect the privacy of the individual, Sweeney et al. proposed a k-anonymity model [1,2]. The model requires that the microdata is partitioned into a set of equivalence classes, each equivalence class contains at least k records, and all records within an equivalence class are assigned the same generalized value over each of their quasi-identifier attributes. Thus, each record in the k-anonymity model cannot be identified successfully with a probability greater than $1/k$. The l-diversity model in [3] extends the k-anonymity model. It requires that each equivalence class has at least l different "well-represented" values for a sensitive attribute, so it also implies l-anonymity. To address the limitations of the k-anonymity and l-diversity models, Li et al. [4] introduced the concept of t-closeness, which requires that the distribution of the sensitive attribute values within each equivalence class of indistinguishable records is similar to that of the sensitive attribute values in the entire microdata. Then, various enhanced anonymity methods were proposed, such as (α, k)-anonymity [5], p-sensitive

k-anonymity [6], anatomy [7], slicing [8], anatomy and generalization (ANGEL) [9], and permutation anonymization [10].

The above-mentioned works focus on anonymizing the microdata with only one sensitive attribute. They cannot be directly applied to the microdata with multiple sensitive attributes. Therefore, some extended *k*-anonymity, *l*-diversity, *p*-sensitive and *t*-closeness methods for multiple sensitive attributes [11–32] were proposed. And some extended anatomy methods combining multi-sensitive bucketization (MSB), clustering, generalization or permutation for multiple sensitive attributes [33–36] were proposed. To apply the slicing technique to the microdata with multiple sensitive attributes, some enhanced slicing techniques for multiple sensitive attributes were proposed [37–43]. Additionally, decomposition and decomposition plus were introduced to achieve *l*-diversity for multiple sensitive attributes [44,45]. The above methods for multiple sensitive attributes do not consider the sensitive requirements of various sensitive attributes. Different sensitive attributes may have different sensitivity requirements, so the rating techniques for multiple sensitive attributes were introduced [46,47]. These rating techniques not only protect privacy for multiple sensitive attributes, but also keep a large amount of correlations of the microdata. In real world, different values of a sensitive attribute may have different sensitivity requirements. It is not appropriate to apply the same sensitive requirement to all values of the sensitivity attribute. Hence, the above rating techniques for multiple sensitive attributes are not suitable for this situation.

To solve this problem, we defined three security levels for different sensitive attribute values that have different sensitivity requirements, and given an L_{sl}-diversity model for multiple sensitive attributes. Then, we proposed three specific greed algorithms based on the maximal-bucket first (MBF), maximal single-dimension-capacity first (MSDCF) and maximal multi-dimension-capacity first (MMDCF) algorithms [33] and the maximal security-level first (MSLF) greedy policy, named as MBF based on MSLF (MBF-MSLF), MSDCF based on MSLF (MSDCF-MSLF) and MMDCF based on MSLF (MMDCF-MSLF), to implement the L_{sl}-diversity model for multiple sensitive attributes. The experimental results show that the three algorithms can greatly reduce the information loss of published microdata, but their runtime is only a small increase, and their information loss tends to be stable with the increasing of data volume. Moreover, they can solve the problem that the information loss of the MBF, MSDCF and MMDCF algorithms increases greatly with the increasing of sensitive attribute number.

The remainder of this article is organized as follows. Section 2 provides an overview of the existing privacy preserving data publishing methods for multiple sensitive attributes. In Section 3, we provide some notations and definitions. Section 4 describes the three specific greed algorithms in detail. In Section 5, we present the experimental results and analysis, and concludes the paper in Section 6.

2. Related Works

A large variety of privacy preserving data publishing methods have been proposed for multiple sensitive attributes. In terms of the extended *k*-anonymity, *l*-diversity, *p*-sensitive and *t*-closeness methods for multiple sensitive attributes. Nidhi et al. [11] proposed a new *k*-anonymity model for multiple sensitive attributes, which realizes record suppression with minimum data distortion. Usha et al. [12] extended the *k*-anonymity model for multiple sensitive attributes, and provided several algorithms for implementation of the extended *k*-anonymity model. Liu et al. [13] proposed a new *k*-anonymity algorithm for multiple sensitive attributes, which uses the distribution of sensitive attribute values as a parameter to prevent association disclosure. Wang et al. [14] proposed a novel privacy preserving model for multiple sensitive attributes based on *k*-anonymity, called (α, β, k)-anonymity. They set a hierarchy sensitive attribute rule to achieve (α, β, k)-anonymity and developed a corresponding algorithm to anonymize the microdata by using generalization and hierarchy. Wang et al. [15] clustered multiple sensitive attributes based on a utility matrix, and then used a greedy strategy to partition records into equivalence classes. This method can guarantee that the size of each equivalence class is *k*

except the last one, and can also guarantee the diversity of each sensitive attribute value within an equivalence class.

Ahmed et al. [16] proposed a probabilistic model of multiple sensitive attribute diversity to prevent identification or non-membership attack that arises when the microdata with multiple sensitive attributes is published. In [17–19], a (α, l) model was applied to satisfy the diversity requirements for multiple sensitive attributes. Zhang et al. [17] used anatomization with generalization and suppression based on the (α, l) model. Guo et al. [18] proposed a personalized privacy preserving model for multiple sensitive attributes based on MSB, called personalized (α, l)-anonymity model. Li et al. [19] considered the associations between multiple sensitive attributes to prevent all chances of the positive and negative disclosure, and used a two-step greedy generalization algorithm to manage multiple sensitive attributes. Zhu et al. [20] proposed an addictive noise approach that publishes some anonymized tables after fulfilling the requirement of l-diversity. This approach replaces the multiple sensitive attribute values of each record by a value set and at least l-1 random selected noise values. Huang et al. [21] proposed a (v, l)-anonymity model which checks the differences of sensitive attribute values by incorporating the classification of sensitive attribute values. And (l_1, l_2)-diversity is used to validate the model. Jin et al. [22] proposed a l-coverage cluster grouping model which can handle multiple sensitive attributes. And this model is based on cluster algorithm.

Gal et al. [23] proposed a new model that extends k-anonymity and l-diversity to handle multiple sensitive attributes, and proposed a practical algorithm to implement this model. The algorithm used for this model contains two steps. In the first step, the microdata is divided into partitions, so that every partition contains at least k records and satisfies l-diversity. In the subsequent step, the microdata is anatomized. Wahyu et al. [24] proposed a distribution model to set sensitive attribute values when p-sensitive is applied to multiple sensitive attributes, minimizing their probability of disclosure. Wu et al. [25] proposed a p-cover k-anonymity model for protecting multiple sensitive attributes, and extended the incognito algorithm [26] to implement this model. Lin et al. [27] proposed a novel (k, p)-anonymity framework to solve the disclosure problem of sensitive attributes in the k-anonymity and l-diversity models. Anjum et al. [28] proposed an efficient approach for the anonymization of multiple sensitive attributes, called (p, k)-Angelization. The (p, k)-Angelization approach not only protects the privacy of the individual, but also improves the utility of the released information. Kanwala et al. [29] proposed a privacy-preserving model for 1:M records (i.e., an individual can have multiple records) dataset with multiple sensitive attributes, called (p, l)- Angelization.

Wang et al. [30] proposed two privacy-preserving algorithms for multiple sensitive attributes to satisfy the t-closeness model. The two algorithms use different methods to partition records into groups in terms of sensitive attributes. One uses a clustering method, while the other leverages a principal component analysis. Sowmyarani et al. [31] proposes a (p+)-sensitive, t-closeness model for multiple sensitive attributes. It combines the advantages of the t-closeness and the p-sensitive k-anonymity approaches to reduce the possibility of the similarity and skewness attacks of the anonymization techniques. Saraswathi et al. [32] proposed an enhanced t-closeness algorithm for multiple sensitive attributes. In the algorithm, t-closeness is applied over MSB k-anonymity clustering attribute hierarchy (MSB-KACA) algorithm. And they used earth mover distance (EMD) to avoid probabilistic inference attack due to bucketization.

In terms of the extended anatomy methods combining MSB, clustering, generalization or permutation for multiple sensitive attributes, Yang et al. [33] proposed an MSB approach. The main idea of the MSB approach is to partition the given table into a quasi-identifier attribute table and a sensitive attribute table, and to make that each sensitive attribute satisfies the l-diversity constraints. Lin et al. [34] proposed a technique to handle multiple numerical sensitive attributes and to eliminate the threat of proximity breach for multiple sensitive attributes. They applied clustering and MSB techniques to release the microdata with multiple numerical sensitive attributes. Luo et al. [35] proposed an improved framework for multiple sensitive attributes, named anatomy and generalization on multiple sensitive attributes (ANGELMS). This approach vertically partitions

the attributes into one quasi-identifier attribute table and several sensitive attribute tables. Each sensitive attribute table divides the records of the microdata into groups (i.e., buckets). Each bucket obeys the *l*-diversity requirement. In the quasi-identifier attribute table, each group generalizes the quasi-identifier attribute values by following the *k*-anonymity principle. Ye et al. [36] proposed an anonymization method combining anatomy and permutation for protecting privacy of the microdata with multiple sensitive attributes. This method includes two major steps: anatomizing microdata and permutating quasi-identifier attributes. To realize the anonymization method, they further proposed two algorithms, namely naive multi-sensitive bucketization permutation algorithm (NMBPA) and closest distance multi-sensitive bucketization permutation algorithm (CDMBPA).

In terms of the extended slicing methods for multiple sensitive attributes, Dhumal et al. [37] applied the slicing technique without permuting the values of multiple sensitive attributes and did not consider the quasi-identifier attributes while proposing this technique. Kiruthika et al. [38] proposed some enhanced slicing techniques like Mondrian and suppression slicing. Mondrian slicing randomly switches all the buckets whereas suppression slicing permutes the quasi-identifier attribute values of the records. Suppression slicing maintains the microdata's utility by guaranteeing the *l*-diversity principle in each quasi-identifier attribute group. Luo et al. [39] extended the slicing technique from single sensitive attribute to multiple sensitive attributes, which is called slicing on multiple sensitive (SLOMS). Further, they proposed an MSB-KACA algorithm to anonymize the microdata with multiple sensitive attributes by SLOMS. In [40], a dynamic data publishing technique for multiple sensitive attributes was proposed, named the KC slice. The proposed technique integrates the features of LKC-privacy and slicing techniques. Raju et al. [41] proposed a novel dynamic KCi-Slice publishing prototype for retaining the privacy and utility of multiple sensitive attributes, which is an improvement of KC-Slice. Reddy et al. [42] proposed a privacy preserving data publishing model that manages personalization for publishing the microdata with multiple sensitive attributes. The model uses the slicing technique supported by deterministic anonymization for quasi-identifier attribute, i.e., generalization for categorical sensitive attributes and fuzzy approach for numerical sensitive attributes based on diversity. Susan et al. [43] conducted a work which combined the anatomy and slicing techniques for multiple sensitive attributes, called anatomization with slicing for multiple sensitive attributes (SLAMSA). They used anatomization to reduce information loss and enhanced the slicing technique to improve attribute correlation.

In terms of the decomposition methods for multiple sensitive attributes, Ye et al. [44] proposed a decomposition technique to achieve *l*-diversity for multiple sensitive attributes. In the decomposition technique, vertical partitioning of multiple sensitive attributes is done that divides the original table into two tables, i.e., a sensitive table and a non-sensitive table. But adding noise in the decomposition technique causes distortion. Hence, Das et al. [45] extended the decomposition technique by optimizing the noise value selection (i.e., choosing the noise value closer to the original values), called decomposition plus.

The above methods for multiple sensitive attributes do not consider the sensitive requirements of sensitive attributes. Because different sensitive attributes may have different sensitivity requirements, Liu et al. [46] introduced a rating technique for multiple sensitive attributes, which is based on different sensitivity coefficients for different attributes. This approach not only protects privacy for multiple sensitive attributes, but also keeps a large amount of correlations of the microdata. But the rating technique can be attacked by applying association rules due to the relationship between sensitive attribute values. Yi et al. [47] removed the weaknesses of the rating technique and eliminated the threat of association attack.

3. Notations and Definitions

In the real world, different values of a sensitive attribute may have different sensitivity requirements. Some values of the sensitive attribute have no sensitivity requirement, i.e., these sensitive attribute values do not need to be protected because their leakage is not harmful to the individual. Some values

of the sensitive attribute have low sensitivity requirement, i.e., these sensitive attribute values need to be protected to some extent because their leakage cause certain harm to the individual. Furthermore, some values of the sensitive attribute have high sensitivity requirement, i.e., these sensitive attribute values need to be well protected because their leakage cause serious harm to the individual. Accordingly, three sensitive attribute security levels are defined as follows.

Definition 1 (sensitive attribute security Level 0). *Sensitive attribute security Level 0 is the security level of a sensitive attribute value with no sensitivity requirement, i.e., a sensitive attribute value with sensitive attribute security Level 0 have no sensitivity requirement.*

Definition 2 (sensitive attribute security Level 1). *Sensitive attribute security Level 1 is the security level of a sensitive attribute value with low sensitivity requirement, i.e., a sensitive attribute value with sensitive attribute security Level 1 have low sensitivity requirement.*

Definition 3 (sensitive attribute security Level 2). *Sensitive attribute security Level 2 is the security level of a sensitive attribute value with high sensitivity requirement, i.e., a sensitive attribute value with sensitive attribute security Level 2 have high sensitivity requirement.*

Let $T = \{A_1, A_2, \ldots, A_p, S_1, S_2, \ldots, S_d\}$ be the microdata, where A_i denotes the ith quasi-identifier attribute and $1 \leq i \leq p$, S_j denotes the jth sensitive attribute and $1 \leq j \leq d$, p denotes the number of quasi-identifier attributes and d denotes the number of sensitive attributes, n denotes the number of records of T(i.e., $n = |T|$), t_k denotes the kth record of T and $1 \leq k \leq n$, and $t_k[X]$ denotes the value of the attribute X of the kth record. An example of the microdata is shown in Table 1.

Table 1. An example of the microdata.

Records	Identifier Attributes		Quasi-identifier Attributes				Sensitive Attributes	
	SSN	Name	Age	Sex	Race	Zipcode	Physician	Disease
t_1	19200	Sam	21	M	White	11000	John	Flu
t_2	17720	Anne	60	F	Black	21000	John	Pneumonia
t_3	25000	Mike	56	M	White	11400	Mary	Cancer
t_4	14520	Lily	28	F	Black	65000	Bob	Flu
t_5	18010	Harry	60	M	White	41000	Bob	Pneumonia
t_6	23800	Mona	55	F	Black	41300	Anne	Gastritis
t_7	34000	Tony	43	M	White	39000	John	Gastritis
t_8	12000	Lucy	26	F	Black	15000	Sam	HIV
t_9	37080	Tim	37	M	White	19000	Mary	Flu

In Table 1, social security number (SSN) and name are two identifier attributes. Age, sex, race and zipcode are four quasi-identifier attributes. Further, physician and disease are two sensitive attributes. t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8 and t_9 are nine records of the microdata. The values of the sensitive attribute physician are John, Bob, Anne, Sam and Mary. Following this, the values of the sensitive attribute disease are flu, pneumonia, gastritis, human immunodeficiency virus (HIV) and cancer. For the former, the security levels of all sensitive attribute values can be set to sensitive attribute security Level 1 because these sensitive attribute values have the same and low sensitivity requirement. For the latter, the security level of flu can be set to sensitive attribute security Level 0 because the sensitive attribute value has no sensitivity requirement. The security levels of Pneumonia and Gastritis can be set to sensitive attribute security Level 1 because the two sensitive attribute values have the same and low sensitivity requirement. Further, the security levels of HIV and cancer can be set to sensitive attribute security Level 2 because the two sensitive attribute values have the same and high sensitivity requirement.

Definition 4 (composite sensitive attribute) [33]. *A composite sensitive attribute is the whole of all sensitive attributes of T, denoted by $S = \{S_1, S_2, \ldots, S_d\}$, where the ith sensitive attribute S_i $(1 \leq i \leq d)$ is the ith dimension of the composite sensitive attribute. $D(S_i)$ is the value field of S_i, and $|S_i|$ represents the number of $D(S_i)$.*

Definition 5 (composite sensitive attribute vector) [33]. *A composite sensitive attribute vector is a vector form of all sensitive attribute values of the kth record t_k in T, denoted by $< t_k[S_1], t_k[S_2], \ldots, t_k[S_d] >$.*

Definition 6 (group) [33]. *A group is a subset of records of T. each record of T belongs to only one group. All groups of T is denoted as $GT = \{G_1, G_2, \ldots, G_m\}$, where m denotes the number of all groups of T.*

For Table 1, the composite sensitive attribute of the microdata is {Physician, Disease}, and a composite sensitive attribute vector can be <John, Flu>. $G_1 = \{t_1, t_2, t_3\}$, $G_2 = \{t_3, t_4, t_5\}$ and $G_3 = \{t_7, t_8, t_9\}$ can be three groups of the microdata, and $G_1 \cap G_2 \cap G_3 = \phi$.

Definition 7 (l-diversity for single sensitive attribute) [33]. *For a group G with single sensitive attribute, if v is the sensitive attribute value with the maximum frequency and $c(v)/|G| \leq 1/l$, where $c(v)$ denotes the frequency of v, $|G|$ denotes the number of records of G, then G satisfies l-diversity for single sensitive attribute.*

Definition 8 (l-diversity for multiple sensitive attributes) [33]. *For a group G with multiple sensitive attributes, if each sensitive attribute of the composite sensitive attribute in G satisfies l-diversity for single sensitive attribute, then G satisfies l-diversity for multiple sensitive attributes.*

Definition 9 (l-diversity group for multiple sensitive attributes) [33]. *An l-diversity group for multiple sensitive attributes is a group of T and the group satisfies l-diversity for multiple sensitive attributes. All l-diversity groups for multiple sensitive attributes are denoted as $GTc = \{G_1, G_2, \ldots, G_{mc}\}$, where mc denotes the number of all l-diversity groups for multiple sensitive attributes.*

From Definitions 7, 8 and 9, all the sensitive attribute values of each group obey the same l-diversity requirement, i.e., the same sensitive requirement is applied to them. This is not appropriate and will cause extra information loss of the microdata. Because the maximal security level of sensitive attribute values in the microdata shown in Table 1 is sensitive attribute security Level 2, so l is set to 3 in this paper. Thus, only three diversity groups for multiple sensitive attributes can be formed. For different sensitive attribute values with different sensitive attribute security levels, they should have different l-diversity requirements because they have different sensitivity requirements. Hence, L_{sl}-diversity for single sensitive attribute and L_{sl}-diversity for multiple sensitive attributes are defined as follows, where $L_{sl} \subseteq \{l_0, l_1, l_2\}$, l_0 for sensitive attribute security level 0, l_1 for sensitive attribute security Level 1, l_2 for sensitive attribute security Level 2, and l_0, l_1, l_2 are set to 1, 2 and 3 in this paper, respectively.

Definition 10 (L_{sl}-diversity for single sensitive attribute). *For a group G with single sensitive attribute, if v_0 is a sensitive attribute value with sensitive attribute security level 0 of G, then $c(v_0)/|G| \leq 1/l_0$, where $c(v_0)$ denotes the frequency of v_0 in G, $|G|$ denotes the number of records in G. Similarly, if v_1 is a sensitive attribute value with sensitive attribute security level 1 of G, then $c(v_1)/|G| \leq 1/l_1$, where $c(v_1)$ denotes the frequency of v_1 in G. Further, if v_2 is a sensitive attribute value with sensitive attribute security level 2 of G, then $c(v_2)/|G| \leq 1/l_2$, where $c(v_2)$ denotes the frequency of v_2 in G. Then, G satisfies L_{sl}-diversity for single sensitive attribute.*

Definition 11 (L_{sl}-diversity for multiple sensitive attributes). *For a group G with multiple sensitive attributes, if each sensitive attribute of the composite sensitive attribute in G satisfies L_{sl}-diversity for single sensitive attribute, then G satisfies L_{sl}-diversity for multiple sensitive attributes.*

Definition 12 (L_{sl}-diversity group for multiple sensitive attributes). *An L_{sl}-diversity group for multiple sensitive attributes is a group of T and the group satisfies L_{sl}-diversity for multiple sensitive attributes. All L_{sl}-diversity groups of T is denoted as $GTs = \{G_1, G_2, \ldots, G_{ms}\}$, where ms denotes the number of all L_{sl}-diversity groups of T.*

For Table 1, as described above, the security levels of all values of the sensitive attribute Physician is sensitive attribute security Level 1. Further, the security level of Flu is sensitive attribute security Level 0, the security levels of pneumonia and gastritis are sensitive attribute security Level 1, and the security levels of HIV and Cancer are sensitive attribute security Level 2. Any record of the microdata shown in Table 1 consists of one value of the sensitive attribute physician and one value of the sensitive attribute disease. As a result, L_{sl} includes at least l_1, so $\{l_1\}$-diversity groups for multiple sensitive attributes, $\{l_0, l_1\}$-diversity groups for multiple sensitive attributes, $\{l_1, l_2\}$-diversity groups for multiple sensitive attributes and $\{l_0, l_1, l_2\}$-diversity groups for multiple sensitive attributes can be formed.

Definition 13 (multiple dimensional bucket) [33]. *A multiple dimensional bucket is a bucket that each dimension of the composite sensitive attribute is one of dimensions of the bucket. Therefore, the records of T can be mapped to corresponding buckets according to the sensitive attribute values of each dimension of their composite sensitive attribute vectors. If the number of dimensions of the composite sensitive attribute in T is d, then d dimensional buckets of T can be established, denoted as $Bucket(S_1, S_2, \ldots, S_d)$, where each d dimensional bucket is denoted as $buk < s^1, s^2, \ldots, s^d >$, $s^j \in D(S_j)$ and $1 \leq j \leq d$, and the size of each d dimensional bucket of is denoted as $size(buk < s^1, s^2, \ldots, s^d >)$, i.e., the number of records in the d dimensional bucket. Further, the dimension capacity of a certain value ${s_0}^j \in D(S_j)$ on the S_j dimension of the d dimensional bucket is the sum of all the bucket sizes with the certain value ${s_0}^j$ on this dimension, denoted as $Capa({s_0}^j) = \sum_{s^j = {s_0}^j} size(buk < s^1, s^2, \ldots, s^d >)$.*

According to Table 1, two-dimensional buckets of T can be established, as shown in Figure 1.

	Flu	Pneumonia	Gastritis	HIV	Cancer	
John	$\{t_1\}$	$\{t_2\}$	$\{t_7\}$			3
Bob	$\{t_4\}$	$\{t_5\}$				2
Anne			$\{t_6\}$			1
Sam				$\{t_8\}$		1
Mary	$\{t_9\}$				$\{t_3\}$	2
	3	2	2	1	1	

Figure 1. Two-dimensional buckets of T.

In Figure 1, the leftmost column is the values of the sensitive attribute physician, and the top row is the values of the sensitive attribute disease. The rightmost column is the dimension capacities of the values of the sensitive attribute physician, and the bottom row is the dimension capacities of the values of the sensitive attribute disease. Further, the five rows and five columns in the middle are 2-dimensional buckets of T. For example, when ${s_0}^1$ is Anne and ${s_0}^2$ is gastritis, $buk < {s_0}^1, {s_0}^2 >$ is a certain two-dimensional bucket, i.e., $\{t_6\}$ in Figure 1, and $size(buk < {s_0}^1, {s_0}^2 >)$ is the size of $buk < {s_0}^1, {s_0}^2 >$ and $size(buk < {s_0}^1, {s_0}^2 >) = 1$. $Capa({s_0}^1)$ is the dimension capacity of ${s_0}^1$ and $Capa({s_0}^1) = \sum_{s^1 = {s_0}^1} size(buk < s^1, s^2 >) = 0 + 0 + 1 + 0 + 0 = 1$, and $Capa({s_0}^2)$ is the dimension capacity of ${s_0}^2$ and $Capa({s_0}^2) = \sum_{s^2 = {s_0}^2} size(buk < s^1, s^2 >) = 1 + 0 + 1 + 0 + 0 = 2$.

In [33], the MSB method includes two stages: grouping phase and residual processing phase. In the first stage, according to a greedy strategy, l buckets with different values on each dimension are selected, and one record is extracted from each bucket to form an l-diversity group for multiple sensitive attributes, which circulates until it cannot form a new l-diversity group for multiple sensitive attributes

that meets the requirements. In the second stage, for the remaining records in the multi-dimensional buckets after grouping, add them to the existing l-diversity groups for multiple sensitive attributes as much as possible without destroying l-diversity for multiple sensitive attributes. Finally, records that do not belong to any l-diversity group for multiple sensitive attributes are suppressed from the published microdata. After the above steps, the quasi-identifier attributes of each l-diversity group for multiple sensitive attributes are published as a quasi-identifier attribute table, and the sensitive attributes of each l-diversity group for multiple sensitive attributes are published as a sensitive attribute table. Further, both the additional information loss and the suppression ratio are taken as the standard to measure the quality of the published microdata. The definition of additional information loss is extended as follows.

Definition 14 (additional information loss). *For $GTs = \{G_1, G_2, \ldots, G_{ms}\}$ of T, its additional information loss is $\sum_{1 \le i \le ms} (|G_i| - l_{G_i}) / \sum_{1 \le i \le ms} l_{G_i}$, where $|G_i|$ denotes the number of records in G_i, and l_{G_i} denotes the l value for the maximal sensitive attribute security level of sensitive attribute values in G_i.*

Definition 15 (suppression ratio) [33]. *After generating $GTs = \{G_1, G_2, \ldots, G_{ms}\}$ of T, the suppression ratio of T is $n_s/|T|$, where n_s denotes the number of suppressed records of T.*

Obviously, the smaller the suppression ratio is, the less records are lost. When the suppression ratio is the same, the smaller the additional information loss, the less information is lost.

4. Our Proposed Algorithms

In [33], three specific greed algorithms were proposed to implement the above MSB method, called MBF, MSDCF, and MMDCF. According to Definitions 10, 11 and 12, a record with a high sensitive attribute security level is more difficult to be used to form a group than a record with a low sensitive attribute security level, so the record with a higher sensitive attribute security level should be prioritized to form a group. In view of this idea, we also propose three specific greed algorithms based on the MBF, MSDCF and MMDCF algorithms and the MSLF greedy policy, named as MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF, to implement the L_{sl}-diversity model for multiple sensitive attributes.

4.1. MBF-MSLF

The basic idea of the MBF-MSLF algorithm is to first select an unshielded non-empty d dimensional bucket with the maximal sensitive attribute security level and the largest bucket size, and extract a record from the bucket to add to a group and delete the record from the bucket. For this record, all buckets of some certain dimensions of the record are shielded when adding any record of these buckets to the group will destroy L_{sl}-diversity for multiple sensitive attributes of the group. By repeating the above process, the L_{sl}-diversity group is formed. Following this, the shielding of each d dimensional bucket is removed, and the above grouping process is repeated until a complete group cannot be formed. For each remaining record, it is added to a formed group without destroying L_{sl}-diversity for multiple sensitive attributes of the group. Finally, the records that cannot be added to any formed group will be suppressed in the published microdata. The specific steps of the MBF-MSLF algorithm are shown in Algorithm 1.

Algorithm 1: MBF-MSLF

Input: the microdata $T = \{A_1, A_2, \cdots, A_p, S_1, S_2, \cdots, S_d\}$, and the sensitive attribute security levels for all sensitive attribute values of S_i $(1 \le i \le d)$.

Output: a quasi-identifier attribute table and a sensitive attributes table.

1. let $GTs = \varnothing$, and establish d dimensional buckets of T, i.e., $Bucket(S_1, S_2, \cdots, S_d)$;
2. calculate the sensitive attribute security level of each d dimensional bucket. i.e., the maximum of sensitive attribute security levels of all sensitive attribute values in one record of the d dimensional bucket;
3. while (the number of records in the d dimensional buckets > 0)
4. let $G = \varnothing$;
5. calculate the maximal sensitive attribute security level of the d dimensional buckets, and set l_G for the maximal sensitive attribute security level;
6. for $i = 1$ to l_G
7. if (the number of unshielded non-empty d dimensional buckets > 0)
8. calculate the maximal sensitive attribute security level of the unshielded non-empty d dimensional buckets;
9. select the bucket with the maximal sensitive attribute security level and the largest bucket size, and extract a record from the bucket to add to G;
10. delete this record from the bucket, and set $size(buk) = size(buk) - 1$;
11. shield all buckets of some certain dimensions of the record when adding any record of these buckets to G will destroy L_{sl}-diversity for multiple sensitive attributes of G;
12. else stop grouping process;
13. end if
14. end for
15. add G to GTs;
16. remove the shielding of each d dimensional bucket;
17. end while
18. for each record in the non-empty d dimensional buckets
19. add the record to an existing group $G_i \in GTs$ without destroying L_{sl}-diversity for multiple sensitive attributes of G_i;
20. end for
21. suppress the remaining records in the non-empty d dimensional buckets;
22. output a quasi-identifier attribute table and a sensitive attributes table according to GTs.

4.2. MSDCF-MSLF

The basic idea of the MSDCF-MSLF algorithm is to first select an unshielded non-empty d dimensional bucket with the maximal sensitive attribute security level and the largest bucket selectivity, and extract a record from the bucket to add to a group and delete the record from the bucket. The bucket selectivity in the MSDCF-MSLF algorithm is calculated as follows.

$$Select(buk < {s_0}^1, {s_0}^2, \ldots, {s_0}^d >) = Max_{1 \le j \le d} Capa({s_0}^j) + size(buk < {s_0}^1, {s_0}^2, \ldots, {s_0}^d >) \quad (1)$$

where $size(buk < {s_0}^1, {s_0}^2, \ldots, {s_0}^d >)$, $Max_{1 \le j \le d} Capa({s_0}^j)$ and $Select(buk < {s_0}^1, {s_0}^2, \ldots, {s_0}^d >)$ are the bucket size, the maximal single-dimensional capacity and the bucket selectivity of a certain bucket $buk < {s_0}^1, {s_0}^2, \ldots, {s_0}^d >$, respectively. For this record, all buckets of some certain dimensions of the record are shielded when adding any record of these buckets to the group will destroy L_{sl}-diversity for multiple sensitive attributes of the group. By repeating the above process, the L_{sl}-diversity group is formed. Following this, the shielding of each d dimensional bucket is removed, and the above grouping process is repeated until a complete group cannot be formed. For each remaining record, it is added to a formed group without destroying L_{sl}-diversity for multiple sensitive attributes of the group. Finally, the records that cannot be added to any formed group will be suppressed in the published microdata. The specific steps of the MSDCF-MSLF algorithm are shown in Algorithm 2.

4.3. MMDCF-MSLF

The basic idea of the MMDCF-MSLF algorithm is to first select an unshielded non-empty d dimensional bucket with the maximal sensitive attribute security level and the largest bucket selectivity, and extract a record from the bucket to add to a group and delete the record from the bucket. The bucket selectivity in the MMDCF-MSLF algorithm is calculated as follows.

$$Select(buk < {s_0}^1, {s_0}^2, \ldots, {s_0}^d >) = \sum_{1 \le j \le d} Capa({s_0}^j) + size(buk < {s_0}^1, {s_0}^2, \ldots, {s_0}^d >) \quad (2)$$

where $size(buk < {s_0}^1, {s_0}^2, \ldots, {s_0}^d >)$, $\sum_{1 \le j \le d} Capa({s_0}^j)$ and $Select(buk < {s_0}^1, {s_0}^2, \ldots, {s_0}^d >)$ are the bucket size, the sum of all dimension capacities and the bucket selectivity of a certain bucket $buk < {s_0}^1, {s_0}^2, \ldots, {s_0}^d >$, respectively. For this record, all buckets of some certain dimensions of the record are shielded when adding any record of these buckets to the group will destroy L_{sl}-diversity for multiple sensitive attributes of the group. By repeating the above process, the L_{sl}-diversity group is formed. Following this, the shielding of each d dimensional bucket is removed, and the above grouping process is repeated until a complete group cannot be formed. For each remaining record, it is added to a formed group without destroying L_{sl}-diversity for multiple sensitive attributes of the group. Finally, the records that cannot be added to any formed group will be suppressed in the published microdata. The specific steps of the MMDCF-MSLF algorithm are shown in Algorithm 3.

Algorithm 2: MSDCF-MSLF

Input: the microdata $T=\{A_1, A_2, \cdots, A_p, S_1, S_2, \cdots, S_d\}$, and the sensitive attribute security levels for all sensitive attribute values of S_i $(1 \le i \le d)$.

Output: a quasi-identifier attribute table and a sensitive attributes table.

1. let $GTs = \varnothing$, and establish d dimensional buckets of T, i.e., $Bucket(S_1, S_2, \cdots, S_d)$;
2. calculate the sensitive attribute security level of each d dimensional bucket. i.e., the maximum of sensitive attribute security levels of all sensitive attribute values in one record of the d dimensional bucket;
3. calculate all dimension capacities of each d dimensional bucket;
4. while (the number of records in the d dimensional buckets > 0)
5. | let $G = \varnothing$;
6. | calculate the maximal sensitive attribute security level of the d dimensional buckets, and set l_G for the maximal sensitive attribute security level;
7. | for $i = 1$ to l_G
8. | | if (the number of unshielded non-empty d dimensional buckets > 0)
9. | | | calculate the maximal sensitive attribute security level of the unshielded non-empty d dimensional buckets;
10. | | | calculate the maximal single-dimensional capacity of each unshielded non-empty d dimensional bucket;
11. | | | calculate the selectivity of each unshielded non-empty d dimensional bucket;
12. | | | select the bucket with the maximal sensitive attribute security level and the largest bucket selectivity, and extract a record from the bucket to add to G;
13. | | | delete this record from the bucket, and set $size(buk) = size(buk) - 1$ and $Capa(s_0^{\,j}) = Capa(s_0^{\,j}) - 1$, where $1 \le j \le d$;
14. | | | shield all buckets of some certain dimensions of the record when adding any record of these buckets to G will destroy L_{sl}-diversity for multiple sensitive attributes of G;
15. | | else stop grouping process;
16. | | end if
17. | end for
18. | add G to GTs;
19. | remove the shielding of each d dimensional bucket;
20. end while
21. for each record of the non-empty d dimensional buckets
22. | add the record to an existing group $G_i \in GTs$ without destroying L_{sl}-diversity for multiple sensitive attributes of G_i;
23. end for
24. suppress the remaining records of the non-empty d dimensional buckets;
25. output a quasi-identifier attribute table and a sensitive attributes table according to GTs.

Algorithm 3: MMDCF-MSLF

Input: the microdata $T=\{A_1,A_2,\cdots,A_p,S_1,S_2,\cdots,S_d\}$, and the sensitive attribute security levels for all sensitive attribute values of S_i $(1\le i\le d)$.

Output: a quasi-identifier attribute table and a sensitive attributes table.

1. let $GTs=\varnothing$, and establish d dimensional buckets of T, i.e., $Bucket(S_1,S_2,\cdots,S_d)$;
2. calculate the sensitive attribute security level of each d dimensional bucket. i.e., the maximum of sensitive attribute security levels of all sensitive attribute values in one record of the d dimensional bucket;
3. calculate all dimension capacities of each d dimensional bucket;
4. while (the number of records in the d dimensional buckets > 0)
5. let $G=\varnothing$;
6. calculate the maximal sensitive attribute security level of the d dimensional buckets, and set l_G for the maximal sensitive attribute security level;
7. for $i=1$ to l_G
8. if (the number of the unshielded non-empty d dimensional buckets > 0)
9. calculate the maximal sensitive attribute security level of the unshielded non-empty d dimensional buckets;
10. calculate the sum of all dimension capacities of each unshielded non-empty d dimensional bucket;
11. calculate the selectivity of each unshielded non-empty d dimensional bucket;
12. select the bucket with the maximal sensitive attribute security level and the largest bucket selectivity, and extract a record from the bucket to add to G;
13. delete this record from the bucket, and set $size(buk)=size(buk)-1$ and $Capa(s_0{}^j)=Capa(s_0{}^j)-1$, where $1\le j\le d$;
14. shield all buckets of some certain dimensions of the record when adding any record of these buckets to G will destroy L_{sl}-diversity for multiple sensitive attributes of G;
15. else stop grouping process;
16. end if
17. end for
18. add G to GTs;
19. remove the shielding of each d dimensional bucket;
20. end while
21. for each record of the non-empty d dimensional buckets
22. add the record to an existing group $G_i\in GTs$ without destroying L_{sl}-diversity for multiple sensitive attributes of G_i;
23. end for
24. suppress the remaining records of the non-empty d dimensional buckets;
25. output a quasi-identifier attribute table and a sensitive attributes table according to GTs.

5. Experimental Results and Analysis

The experimental environment in this paper is as follows: Intel (R) Core (TM) i5-7200U 2.5 GHz dual-core processor with 8 GB memory, Windows 10 64 bit operating system, and the programming language is C++. The experimental microdata is the demographic dataset of university of California Irvine (UCI) machine learning repository from http://kdd.ics.uci.edu. The microdata contains 30162

complete records, and each record has nine fields, where the Occupation field, the Education field, the Marital field, the Workclass field and the Race field are chosen as multiple sensitive attributes in this paper, as shown in Table 2.

Table 2. Multiple sensitive attributes of the microdata.

Multiple Sensitive Attributes	Occupation	Education	Marital	Workclass	Race
the number of sensitive attribute values	14	16	7	7	5

For the multiple sensitive attributes of the microdata, different sensitive attribute security levels and composite sensitive attributes are chosen in this paper, as shown as Tables 3 and 4 respectively.

Table 3. Different sensitive attribute security levels for the multiple sensitive attributes of the microdata.

Multiple Sensitive Attributes	Sensitive Attribute Security Level 0	Sensitive Attribute Security Level 1	Sensitive Attribute Security Level 2
Occupation	Other-service	Adm-clerical, Craft-repair, Exec-managerial, Farming-fishing, Handlers-cleaners, Machine-op-inspct, Priv-house-serv, Prof-specialty, Protective-serv, Sales, Tech-support, Transport-moving	Armed-Forces
Education	/	10th, 11th, 12th, 1st-4th, 5th-6th, 7th-8th, 9th, Assoc-acdm, Assoc-voc, Bachelors, Doctorate, HS-grad, Masters, Preschool, Prof-school, Some-college	/
Marital	/	Married-civ-spouse, Never-married	Divorced, Married-AF-spouse, Married-spouse-absent, Separated, Widowed,
Workclass	Private, Without-pay	Self-emp-inc, Self-emp-not-inc	Federal-gov, Local-gov, State-gov
Race	Other, White	Black	Amer-Indian-Eskimo, Asian-Pac-Islander

Table 4. Different composite sensitive attributes for the multiple sensitive attributes of the microdata.

The Number of Dimensions of a Composite Sensitive Attribute	Composite Sensitive Attributes
$d = 2$	{Occupation, Education}
$d = 3$	{Occupation, Education, Marital}
$d = 4$	{Occupation, Education, Marital, Workclass}
$d = 5$	{Occupation, Education, Marital, Workclass, Race}

The experiment mainly compares and analyzes the additional information loss, the suppression ratio and the central processing unit (CPU) runtime of the algorithms from the following aspects: (1) changing the number of records (i.e., the value of n is from 1000 to 10000) when the number of sensitive attributes is set to three; (2) changing the number of sensitive attributes (i.e., the value of d is from 2 to 5) when the number of records is set to 2000.

5.1. Comparative Analysis of MBF and MBF-MSLF

Figures 2–4 are the additional information loss, the suppression ratio and the CPU runtime of MBF and MBF-MSLF when the number of records is changed from 1000 to 10000 and the number of sensitive attributes is set to 3, respectively.

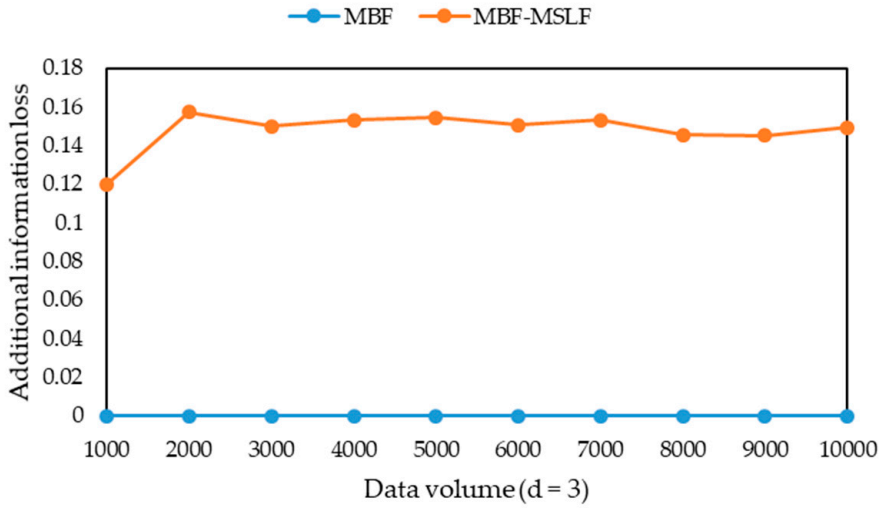

Figure 2. The additional information loss of maximal-bucket first (MBF) and MBF-maximal security-level first (MSLF) with different data volumes (d = 3).

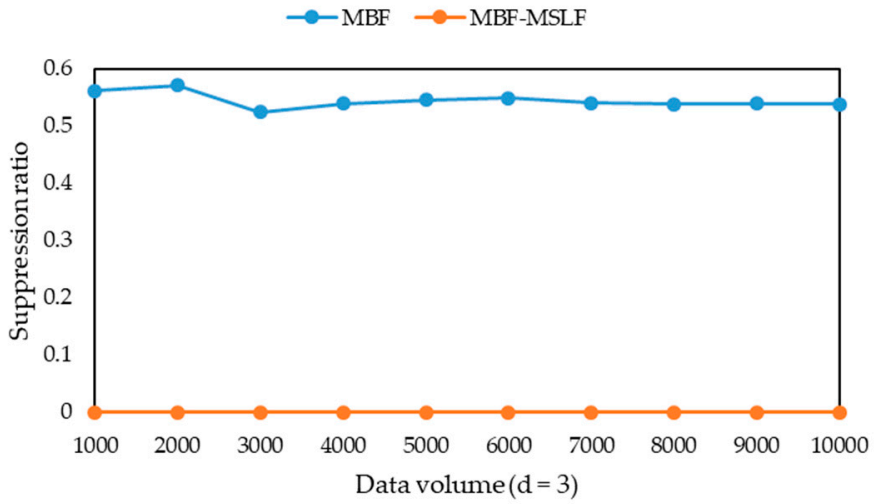

Figure 3. The suppression ratio of MBF and MBF-MSLF with different data volumes (d = 3).

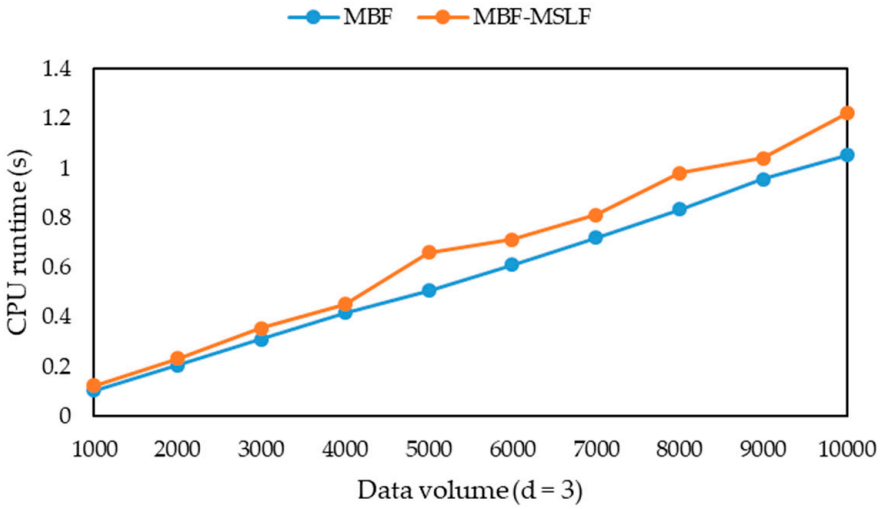

Figure 4. The CPU runtime of MBF and MBF-MSLF with different data volumes (d = 3).

According to Figures 2–4, compared with MBF, the additional information loss of MBF-MSLF increases a little, but the suppression ratio the MBF-MSLF directly decreases to 0, which greatly reduces the information loss of the published microdata. With the increasing of data volume, the additional information loss and the suppression ratio of MBF and MBF-MSLF tend to be stable because the distribution of the sensitive attribute values in the microdata becomes more and more stable. In addition, with the increasing of data volume, the CPU runtime of MBF and MBF-MSLF increase gradually, and the CPU runtime of MBF-MSLF increases faster than the that of MBF.

Figures 5–7 are the additional information loss, the suppression ratio and the CPU runtime of MBF and MBF-MSLF when the number of sensitive attributes is changed from 2 to 5 and the number of records is set to 2000, respectively.

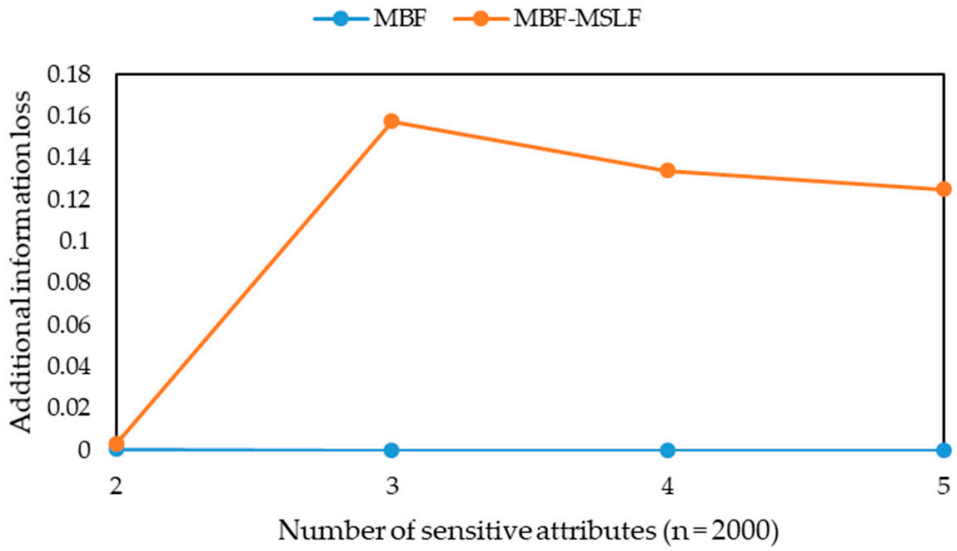

Figure 5. The additional information loss of MBF and MBF-MSLF with different sensitive attribute numbers (n = 2000).

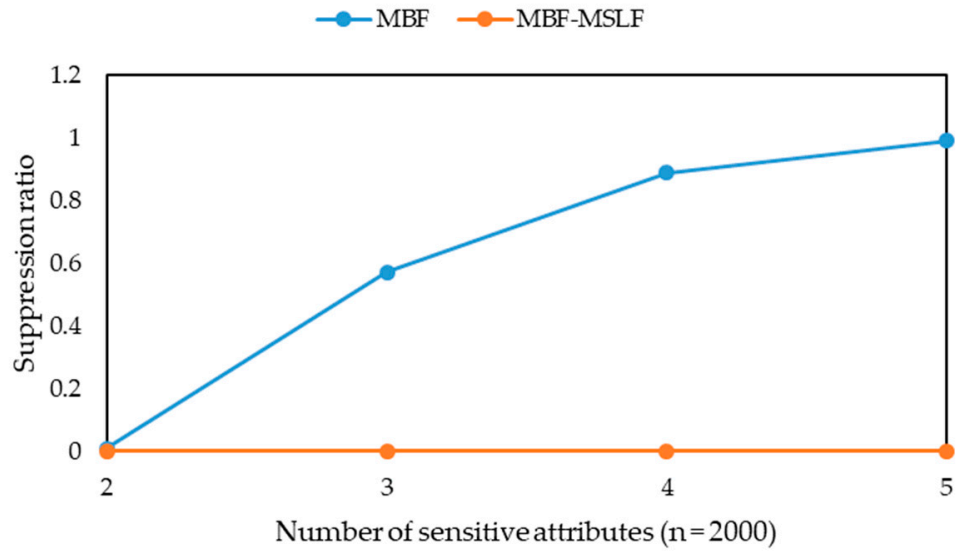

Figure 6. The suppression ratio of MBF and MBF-MSLF with different sensitive attribute numbers (n = 2000).

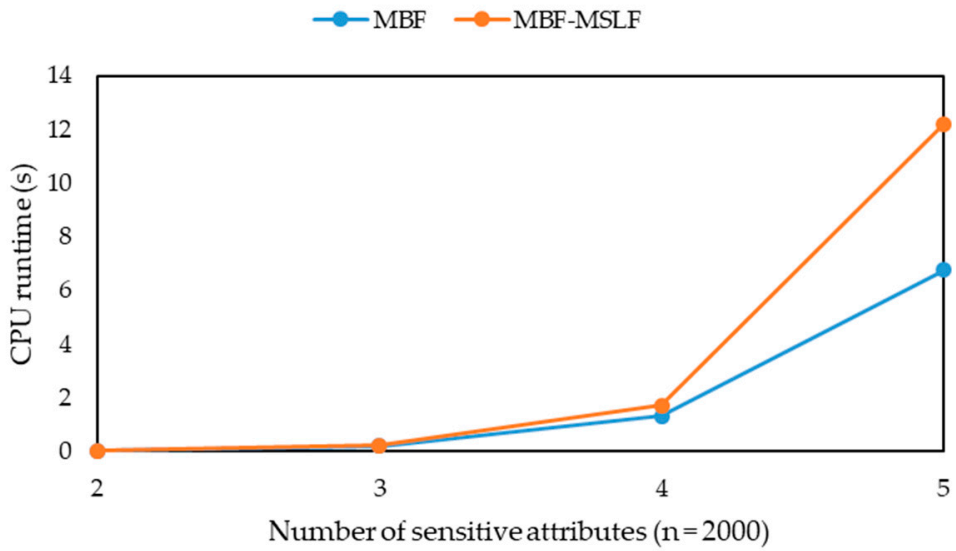

Figure 7. The CPU runtime of MBF and MBF-MSLF with different sensitive attribute numbers (n = 2000).

According to Figures 5–7, compared with MBF, the additional information loss of MBF-MSLF increases a little, but the suppression ratio of MBF-MSLF directly decreases to 0, which greatly reduces the information loss of the published microdata. With the increasing of sensitive attribute number, the additional information loss of MBF tends to be 0 quickly, and the suppression ratio of MBF increases very fast. This shows that fewer and fewer groups can be formed, and it is more and more difficult to add records to the formed groups with the increasing of sensitive attribute number. But for MBF-MSLF, with the increasing of sensitive attribute number, its additional information loss increases quickly and then decreases slowly, while its suppression ratio is always 0. This shows that all records of the microdata can be grouped, but at first it is easier to add records to the formed groups, and then it is more difficult to add records to the formed groups with the increasing of sensitive attribute number. In addition, with the increasing of sensitive attribute number, the CPU runtime of MBF and MBF-MSLF increase gradually, and the CPU runtime of MBF-MSLF increases faster than the that of MBF.

From the above comparative analysis of MBF and MBF-MSLF, compared with MBF, MBF-MSLF can greatly reduce the information loss of the published microdata, but its runtime is only a small increase. Like MBF, the information loss of MBF-MSLF tends to be stable with the increasing of data volume. And MBF-MSLF can solve the problem that the information loss of MBF increases greatly with the increasing of sensitive attribute number.

5.2. Comparative Analysis of MSDCF and MSDCF-MSLF

Figures 8–10 are the additional information loss, the suppression ratio and the CPU runtime of MSDCF and MSDCF-MSLF when the number of records is changed from 1000 to 10000 and the number of sensitive attributes is set to 3, respectively.

According to Figures 8–10, compared with MSDCF, the additional information loss of MSDCF-MSLF increases a little, but the suppression ratio of MSDCF-MSLF directly decreases to 0. Thus, MSDCF-MSLF can greatly reduce the information loss of the published microdata. With the increasing of data volume, the additional information loss and the suppression ratio of MSDCF and MSDCF-MSLF tend to be stable. This is because the distribution of the sensitive attribute values in the microdata becomes more and more stable. Moreover, with the increasing of data volume, the CPU runtime of MSDCF and MSDCF-MSLF increase gradually, and the CPU runtime of MSDCF-MSLF increases faster than the that of MSDCF.

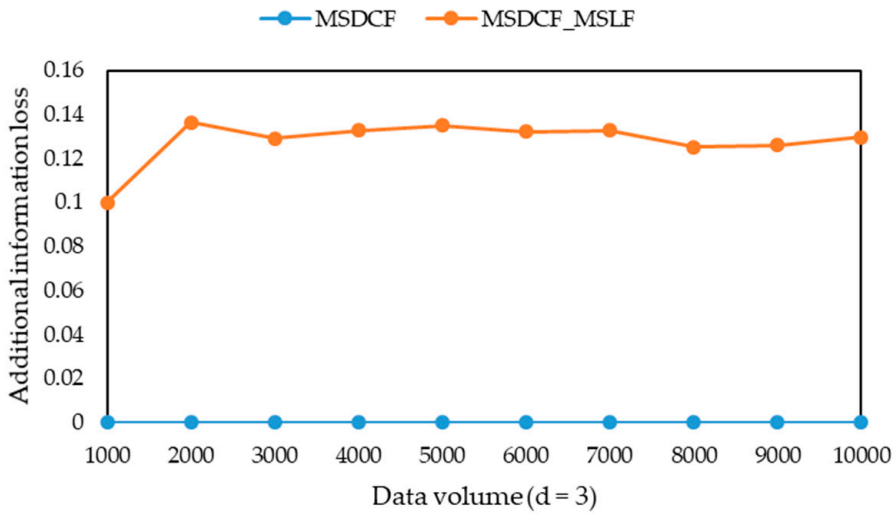

Figure 8. The additional information loss of maximal single-dimension-capacity first (MSDCF) and MSDCF-MSLF with different data volumes (d = 3).

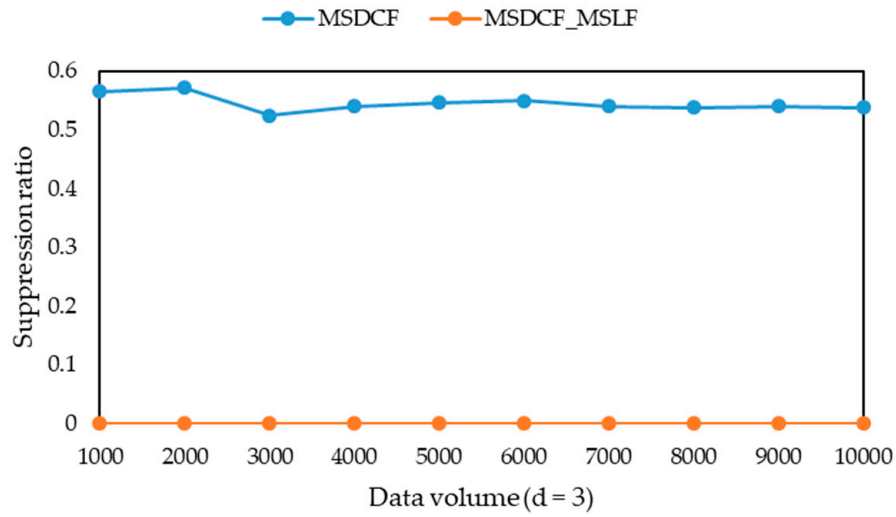

Figure 9. The suppression ratio of MSDCF and MSDCF-MSLF with different data volumes (d = 3).

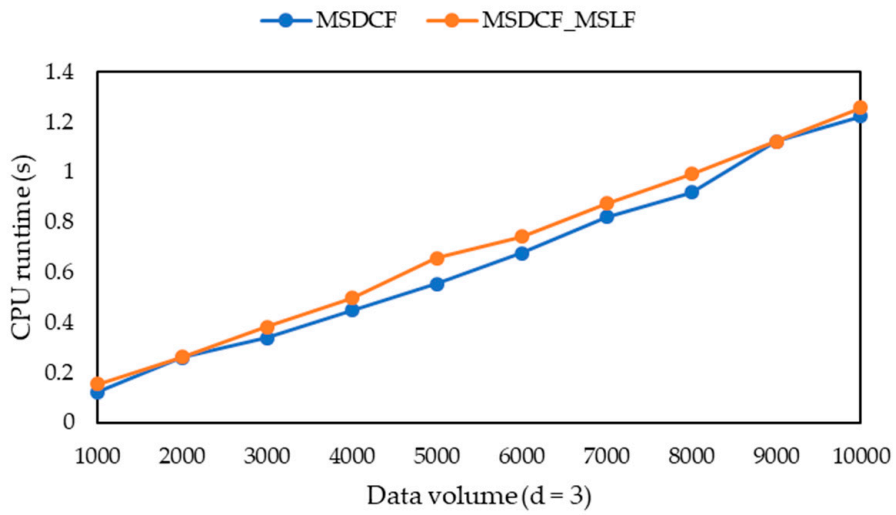

Figure 10. The CPU runtime of MSDCF and MSDCF-MSLF with different data volumes (d = 3).

Figures 11–13 are the additional information loss, the suppression ratio and the CPU runtime of MSDCF and MSDCF-MSLF when the number of sensitive attributes is changed from 2 to 5 and the number of records is set to 2000, respectively.

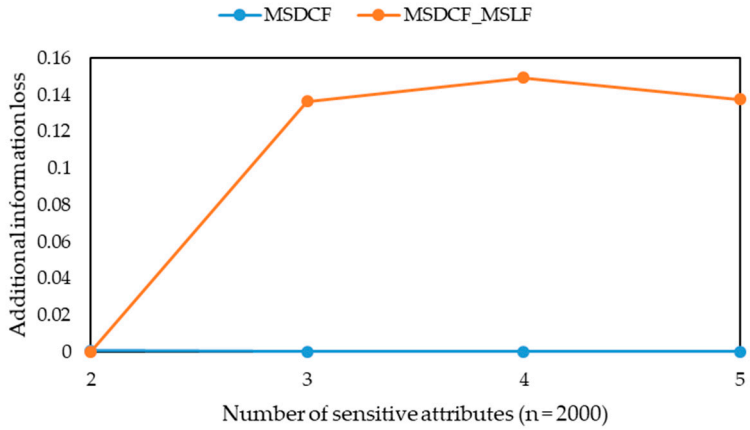

Figure 11. The additional information loss of MSDCF and MSDCF-MSLF with different sensitive attribute numbers (n = 2000).

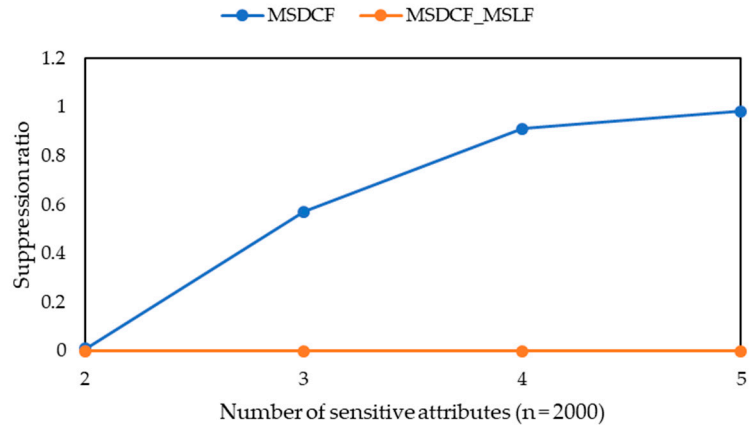

Figure 12. The suppression ratio of MSDCF and MSDCF-MSLF with different sensitive attribute numbers (n = 2000).

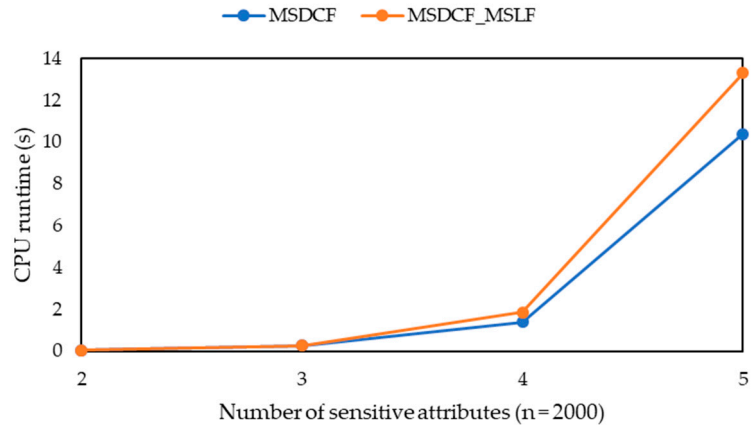

Figure 13. The CPU runtime of MSDCF and MSDCF-MSLF with different sensitive attribute numbers (n = 2000).

According to Figures 11–13, compared with MSDCF, the additional information loss of MSDCF-MSLF increases a little, but the suppression ratio of MSDCF-MSLF directly decreases to

0. Thus, MSDCF-MSLF can greatly reduce the information loss of the published microdata. With the increasing of sensitive attribute number, the additional information loss of MSDCF tends to be 0 quickly, and the suppression ratio of MSDCF increases very fast. This means that fewer and fewer groups can be formed, and it is more and more difficult to add records to the formed groups with the increasing of sensitive attribute number. However, with the increasing of sensitive attribute number, the additional information loss of MSDCF-MSLF increases quickly and then decreases slowly, while its suppression ratio is always 0. This means that all records of the microdata can be grouped, but at first it is easier to add records to the formed groups, and then it is more difficult to add records to the formed groups with the increasing of sensitive attribute number. Moreover, with the increasing of sensitive attribute number, the CPU runtime of MSDCF and MSDCF-MSLF increase gradually, and the CPU runtime of MSDCF-MSLF increases faster than the that of MSDCF.

From the above comparative analysis of MSDCF and MSDCF-MSLF, compared with MSDCF, MSDCF-MSLF can greatly reduce the information loss of the published microdata, but its runtime is only a small increase. the information loss of MSDCF-MSLF tends to be stable with the increasing of data volume, like MSDCF. Furthermore, MSDCF-MSLF can solve the problem that the information loss of MSDCF increases greatly with the increasing of sensitive attribute number.

5.3. Comparative Analysis of MMDCF and MMDCF-MSLF

Figures 14–16 are the additional information loss, the suppression ratio and the CPU runtime of MMDCF and MMDCF-MSLF when the number of records is changed from 1000 to 10000 and the number of sensitive attributes is set to 3, respectively.

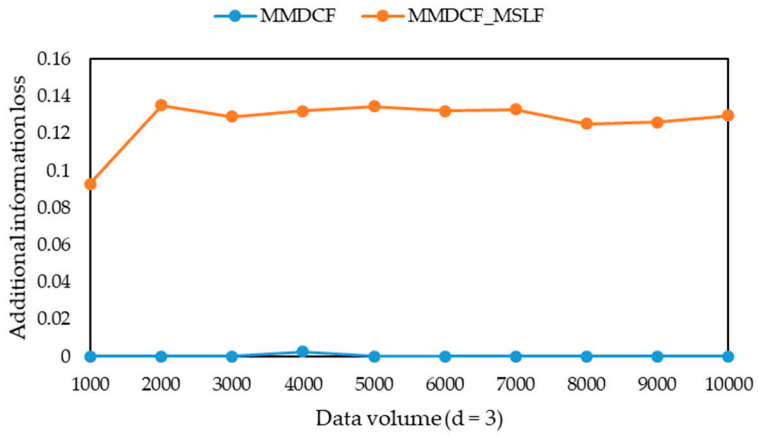

Figure 14. The additional information loss of MMDCF and MMDCF-MSLF with different data volumes (d = 3).

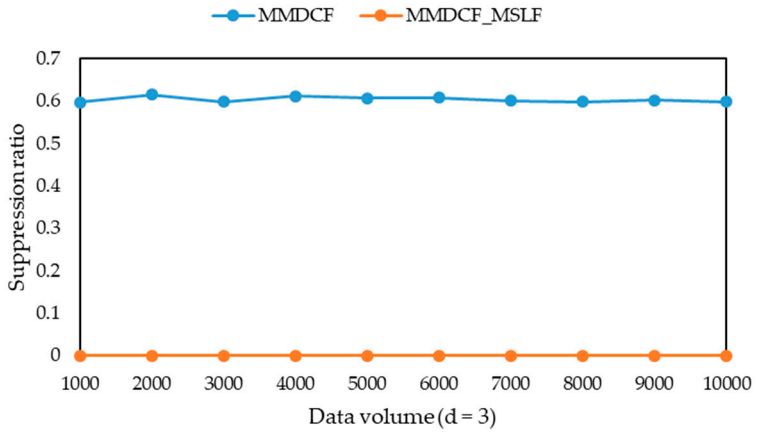

Figure 15. The suppression ratio of MMDCF and MMDCF-MSLF with different data volumes (d = 3).

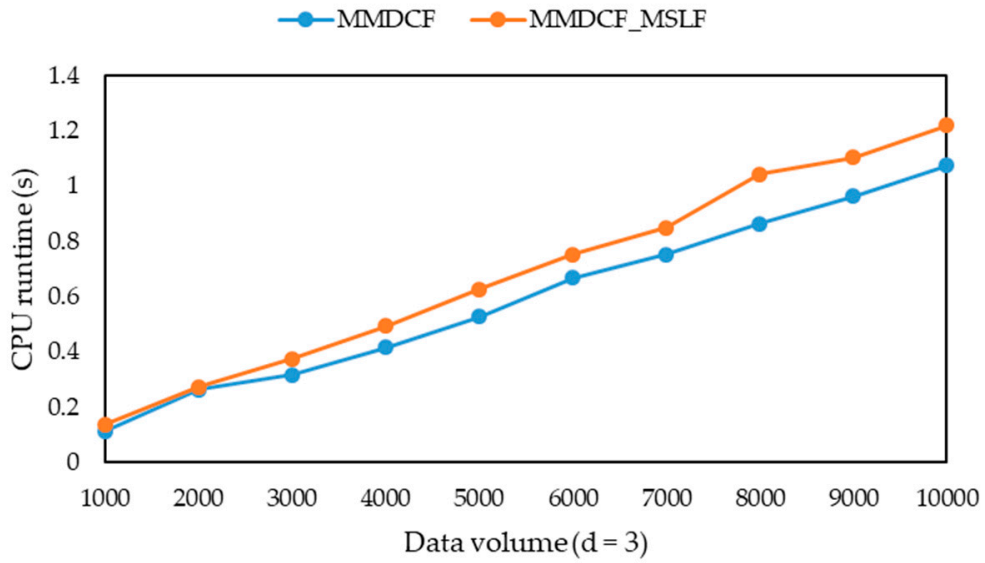

Figure 16. The CPU runtime of MMDCF and MMDCF-MSLF with different data volumes (d = 3).

Compared with MMDCF, the additional information loss of MMDCF-MSLF increases a little, but the suppression ratio of MMDCF-MSLF directly decreases to 0, as shown in Figures 14–16. Hence, MMDCF-MSLF can greatly reduce the information loss of the published microdata. With the increasing of data volume, the distribution of the sensitive attribute values in the microdata becomes more and more stable, so the additional information loss and the suppression ratio of MMDCF and MMDCF-MSLF tend to be stable. Additionally, with the increasing of data volume, the CPU runtime of MMDCF and MMDCF-MSLF increase gradually, and the CPU runtime of MMDCF-MSLF increases faster than the that of MMDCF.

Figures 17–19 are the additional information loss, the suppression ratio and the CPU runtime of MMDCF and MMDCF-MSLF when the number of sensitive attributes is changed from 2 to 5 and the number of records is set to 2000, respectively.

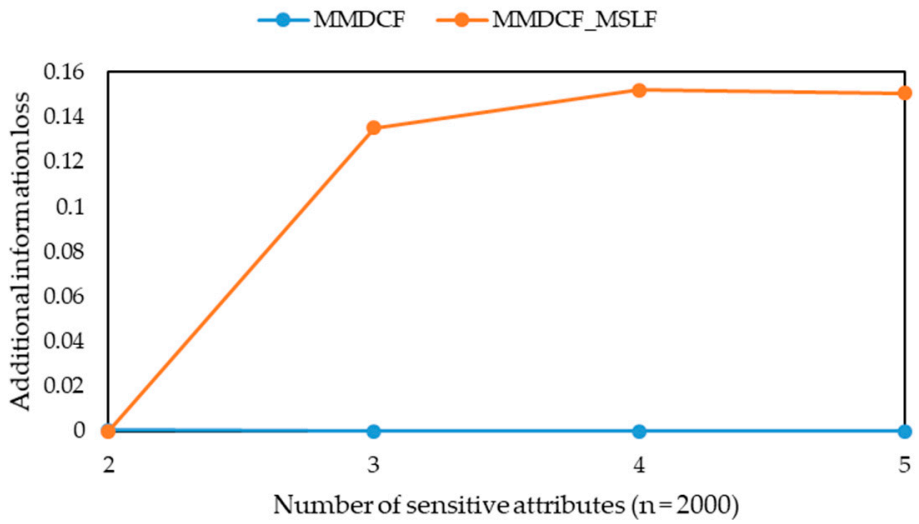

Figure 17. The additional information loss of MMDCF and MMDCF-MSLF with different sensitive attribute numbers (n = 2000).

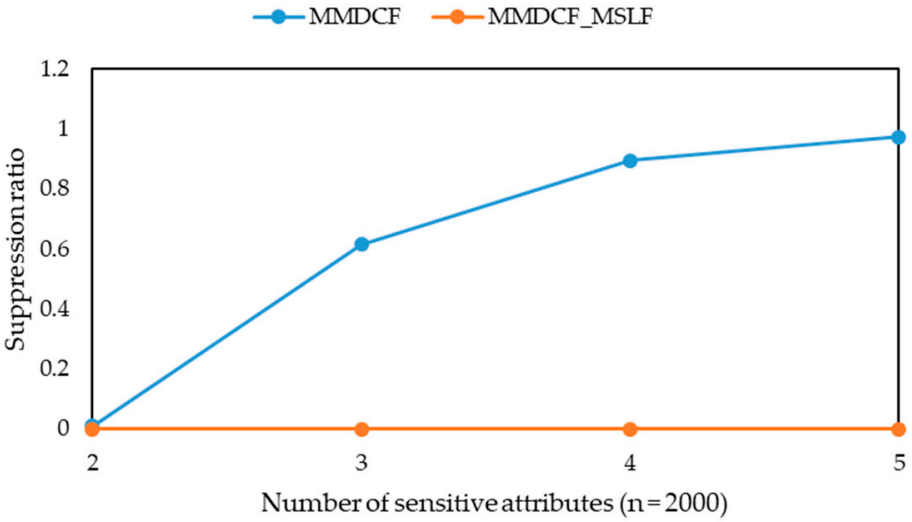

Figure 18. The suppression ratio of maximal multi-dimension-capacity first (MMDCF) and MMDCF-MSLF with different sensitive attribute numbers (n = 2000).

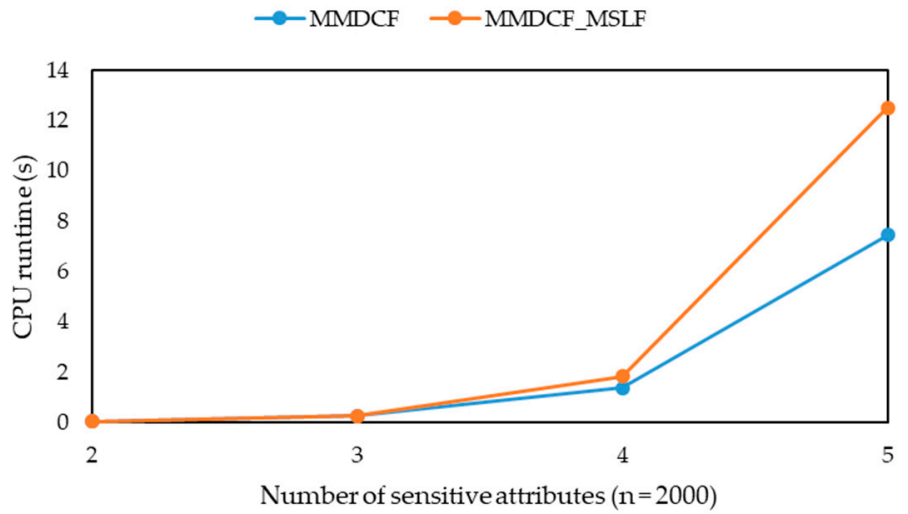

Figure 19. The CPU runtime of MMDCF and MMDCF-MSLF with different sensitive attribute numbers (n = 2000).

Compared with MMDCF, the additional information loss of MMDCF-MSLF increases a little, but the suppression ratio of MMDCF-MSLF directly decreases to 0 according to Figures 17–19. Hence, MMDCF-MSLF can greatly reduce the information loss of the published microdata. With the increasing of sensitive attribute number, the additional information loss of MMDCF tends to be 0 quickly, and the suppression ratio of MMDCF increases very fast. This illustrates that fewer and fewer groups can be formed, and it is more and more difficult to add records to the formed groups with the increasing of sensitive attribute number. With the increasing of sensitive attribute number, the additional information loss of MMDCF-MSLF increases quickly and then decreases slowly, while its suppression ratio is always 0. This illustrates that all records of the microdata can be grouped, but at first it is easier to add records to the formed groups, and then it is more difficult to add records to the formed groups with the increasing of sensitive attribute number. Additionally, the CPU runtime of MMDCF and MMDCF-MSLF increase gradually, and the CPU runtime of MMDCF-MSLF increases faster than the that of MMDCF with the increasing of sensitive attribute number.

From the above comparative analysis of MMDCF and MMDCF-MSLF, compared with MMDCF, MMDCF-MSLF can greatly reduce the information loss of the published microdata, but its runtime is

only a small increase. The information loss of MMDCF-MSLF tends to be stable with the increasing of data volume, similar to Like MMDCF. Furthermore, MMDCF-MSLF can solve the problem that the information loss of MMDCF increases greatly with the increasing of sensitive attribute number.

5.4. Comparative Analysis of MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF

The suppression ratio of MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF is 0 when the number of records is changed from 1000 to 10000 and the number of sensitive attributes is set to 3. Further, Figures 20 and 21 are the additional information loss and the CPU runtime of MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF when the number of records is changed from 1000 to 10000 and the number of sensitive attributes is set to 3, respectively.

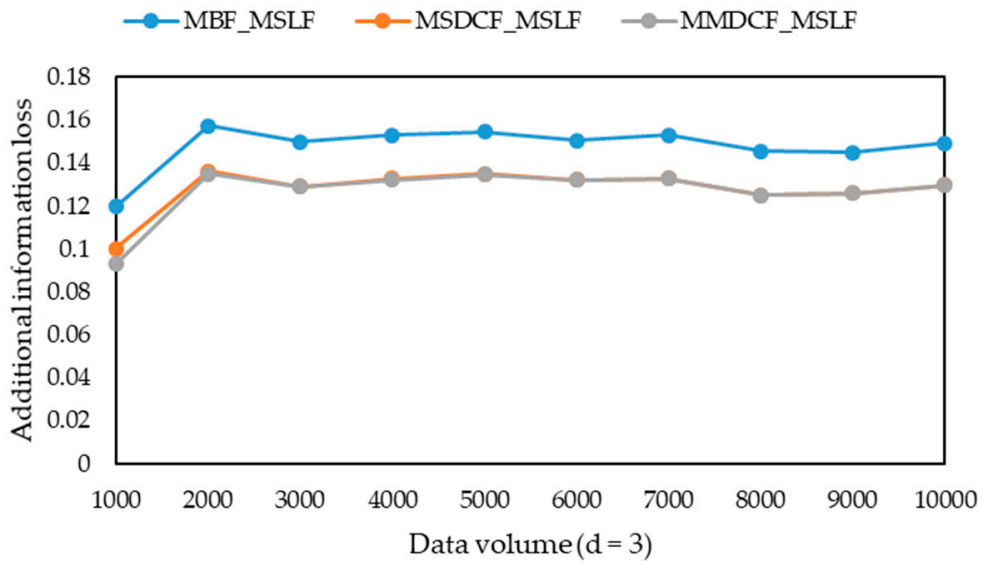

Figure 20. The additional information loss of MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF with different data volumes (d = 3).

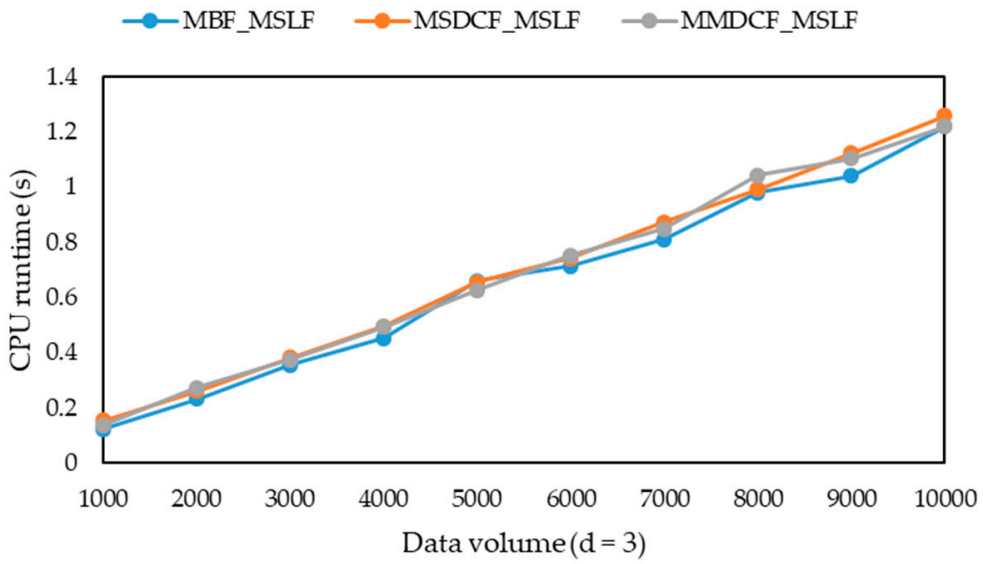

Figure 21. The CPU runtime of MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF with different data volumes (d = 3).

With the increasing of data volume, the suppression ratio of MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF are all 0, but the additional information loss of MSDCF-MSLF or MMDCF-MSLF is

smaller than that of MBF-MSLF according to Figure 20. This shows that all records can be grouped by using MBF-MSLF, MSDCF-MSLF or MMDCF-MSLF, but the added records of MSDCF-MSLF or MMDCF-MSLF is less than those of MBF-MSLF. In addition, with the increasing of data volume, the CPU runtime of MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF increase gradually, and the CPU runtime of MSDCF-MSLF or MMDCF-MSLF increases faster than the that of MBF-MSLF according to Figure 21.

The suppression ratio of MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF is 0 when the number of sensitive attributes is changed from 2 to 5 and the number of records is set to 2000. And Figures 22 and 23 are the additional information loss and the CPU runtime of MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF when the number of sensitive attributes is changed from 2 to 5 and the number of records is set to 2000, respectively.

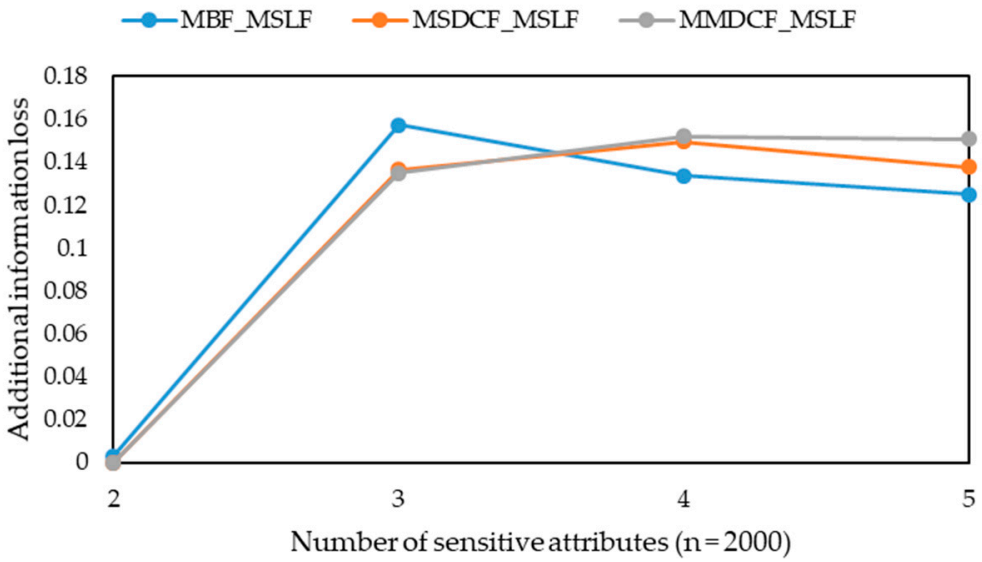

Figure 22. The additional information loss of MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF with different sensitive attribute numbers (n = 2000).

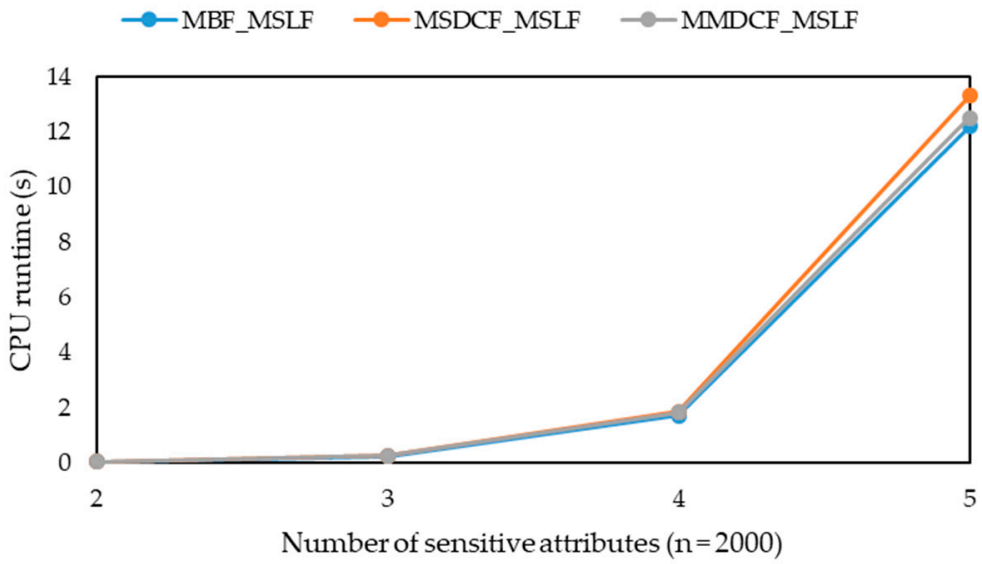

Figure 23. The CPU runtime of MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF with different sensitive attribute numbers (n = 2000).

With the increasing of sensitive attribute number, the suppression ratio of MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF are all 0, but the additional information loss of MSDCF-MSLF or MMDCF-MSLF is first smaller than that of MBF-MSLF, and then larger than that MBF-MSLF according to Figure 22. This shows that all records can be grouped by using MBF-MSLF, MSDCF-MSLF or MMDCF-MSLF, but the added records of MSDCF-MSLF or MMDCF-MSLF is first less than those of MBF-MSLF, and then more than those of MBF-MSLF. In addition, with the increasing of sensitive attribute number, the CPU runtime of MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF increase gradually, and the CPU runtime of MSDCF-MSLF or MMDCF-MSLF increases faster than the that of MBF-MSLF according to Figure 23.

From the above comparative analysis of MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF, their information loss tends to be stable with the increasing of data volume. Compared with MBF-MSLF, the runtime of MSDCF-MSLF or MMDCF-MSLF is only a small increase. Furthermore, when the number of sensitive attributes is small, the information loss of MSDCF-MSLF or MMDCF-MSLF is lower than that of MBF-MSLF, but the information loss of MSDCF-MSLF or MMDCF-MSLF is higher than that of MBF-MSLF with the increasing of sensitive attribute number.

6. Conclusions

In this paper, we first defined three security levels for different sensitive attribute values, and given an L_{sl}-diversity model for multiple sensitive attributes. Further, then we proposed three specific greed algorithms based on the MBF, MSDCF and MMDCF algorithms and the MSLF greedy policy, named as MBF-MSLF, MSDCF-MSLF and MMDCF-MSLF, to form L_{sl}-diversity groups for multiple sensitive attributes. When forming an L_{sl}-diversity group for multiple sensitive attributes, the algorithms is to first select an unshielded non-empty d dimensional bucket with the maximal sensitive attribute security level and the largest bucket size (or the largest bucket selectivity), and extract a record from the bucket to add to the group and delete the record from the bucket. For this record, all buckets of some certain dimensions of the record are shielded when adding any record of these buckets to the group will destroy L_{sl}-diversity for multiple sensitive attributes of the group. By repeating the above process, the L_{sl}-diversity group for multiple sensitive attributes is formed. Following this, the shielding of each d dimensional bucket is removed, and the above grouping process is repeated until a complete L_{sl}-diversity group for multiple sensitive attributes cannot be formed. For each remaining record, it is added to a formed L_{sl}-diversity group for multiple sensitive attributes without destroying L_{sl}-diversity for multiple sensitive attributes of the group. Finally, the records that cannot be added to any formed L_{sl}-diversity group for multiple sensitive attributes will be suppressed in the published microdata.

The experimental results show that the algorithms can greatly reduce the information loss of the published microdata, but its runtime is only a small increase, when comparing with MBF, MSDCF and MMDCF. Their information loss tends to be stable with the increasing of data volume, like MBF, MSDCF and MMDCF. Further, they can solve the problem that the information loss of MBF, MSDCF and MMDCF increases greatly with the increasing of sensitive attribute number. Compared with MBF-MSLF, the runtime of MSDCF-MSLF or MMDCF-MSLF is only a small increase. Further, when the number of sensitive attributes is small, the information loss of MSDCF-MSLF or MMDCF-MSLF is lower than that of MBF-MSLF, but the information loss of MSDCF-MSLF or MMDCF-MSLF is higher than that of MBF-MSLF with the increasing of sensitive attribute number. In this study, when there are more than two unshielded non-empty d dimensional buckets with the same maximal sensitive attribute security level and largest bucket size (or the largest bucket selectivity), we cannot know which bucket should be selected first, so we can only select one of these buckets at random. We will further introduce other security level greedy policies to solve this problem.

Author Contributions: Methodology, Y.X.; software, H.L. All authors have read and agree to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (61741216, 61402367), the Shaanxi Science and Technology Co-ordination and Innovation Project (2016KTTSGY01-03), the Special Scientific

Research Project of Education Department of Shaanxi Province (17JK0704) and the New Star Team Project of Xi'an University of Posts and Telecommunications.

Acknowledgments: The authors would like to thank the anonymous reviewers for their contribution to this paper.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Sweeney, L. *k*-anonymity: A model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.* **2002**, *10*, 557–570. [CrossRef]
2. Fung, B.C.M.; Wang, K.; Chen, R.; Yu, P.S. Privacy-preserving data publishing: A survey on recent developments. *ACM Comput. Surv.* **2010**, *42*, 12. [CrossRef]
3. Machanavajjhala, A.; Gehrke, J.; Kifer, D. *l*-diversity: Privacy beyond *k*-anonymity. *ACM Trans. Knowl. Discov. Data (Tkdd)* **2007**, *1*, 24–35. [CrossRef]
4. Li, N.; Li, T.; Venkatasubramanian, S. *t*-Closeness: Privacy beyond *k*-anonymity and 1-diversity. In Proceedings of the 23rd International Conference on Data Engineering (ICDE), Istanbul, Turkey, 15–20 April 2007; pp. 106–115.
5. Wong, C.R.W.; Li, J.; Fu, A.W.C. (*a*, *k*)-anonymity: An enhanced *k*-anonymity model for privacy preserving data publishing. In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, Philadelphia, PA, USA, 20–23 August 2006; pp. 754–759.
6. Truta, T.M.; Vinay, B. Privacy protection: P-sensitive *k*-anonymity property. In Proceedings of the 22nd International Conference on Data Engineering Workshops, Atlanta, GA, USA, 3–7 April 2006; p. 94.
7. Xiao, X.; Tao, Y. Anatomy: Simple and effective privacy preservation. In Proceedings of the 32nd international conference on Very large data bases, Seoul, Korea, 12–15 September 2006; pp. 139–150.
8. Li, T.; Li, N.; Zhang, J.; Molloy, I. Slicing: A new approach for privacy preserving data publishing. *IEEE Trans. Knowl. Discov. Data Eng.* **2012**, *24*, 561–574. [CrossRef]
9. Tao, Y.; Chen, H.; Xiao, X.; Zhou, S.; Zhang, D. ANGEL: Enhancing the utility of generalization for privacy preserving publication. *IEEE Trans. Knowl. Data Eng.* **2009**, *21*, 1073–1087.
10. He, X.; Xiao, Y.; Li, Y.; Wang, Q.; Wang, W.; Shi, B. Permutation anonymization: Improving anatomy for privacy preservation in data publication. *New Front. Appl. Data Min.* **2012**, 111–123, LNCS 7104.
11. Maheshwarkar, N.; Pathak, K.; Choudhari, N.S. *k*-anonymity model for multiple sensitive. *Int. J. Comput. Appl. Optim. -Chip Commun.* **2012**, *1*, 51–56.
12. Usha, P.; Shriram, R.; Sathishkumar, S. Multiple sensitive attributes based privacy preserving data mining using *k*-anonymity. *Int. J. Sci. Eng. Res.* **2014**, *5*, 122–126.
13. Liu, F.; Jia, Y.; Han, W. A new *k*-anonymity algorithm towards multiple sensitive attributes. In Proceedings of the 12th International Conference on Computer and Information Technology, Chengdu, China, 27–29 October 2012; pp. 768–772.
14. Wang, J. A novel anonymity algorithm for privacy preserving in publishing multiple sensitive attributes. *Res. J. Appl. Sci. Eng. Technol.* **2012**, *4*, 4923–4927.
15. Wang, L.; Zhu, Q. Utility-based anonymization for dataset with multiple sensitive attributes information. *Int. J. High. Perform. Comput. Netw.* **2016**, *9*, 401–415. [CrossRef]
16. Abdalaal, A.; Nergiz, M.E.; Saygin, Y. Privacy-preserving publishing of opinion polls. *Comput. Secur.* **2013**, *37*, 143–154. [CrossRef]
17. Zhang, L.; Xuan, J.; Si, R.; Wang, R. An improved algorithm of individuation *k*-anonymity for multiple sensitive attributes. *Wirel. Pers. Commun.* **2007**, *95*, 2003–2020. [CrossRef]
18. Guo, M.; Liu, Z.; Wang, H. Personalized privacy preserving approaches for multiple sensitive attributes in data publishing. In Proceedings of the 2016 International Conference on Information Science and Technology, Guilin, China, 13–14 August 2016; pp. 1–6.
19. Li, Z.; Ye, X. Privacy protection on multiple sensitive attributes. In Proceedings of the 9th International Conference on Information and Communications Security, Zhengzhou, China, 12–15 December 2007; pp. 141–152.
20. Zhu, H.; Tian, S.; Xie, M.; Yang, M. Preserving privacy for sensitive values of individuals in data publishing based on a new additive noise approach. In Proceedings of the 23rd International Conference on Computer Communication and Networks (ICCCN), Shanghai, China, 4–7 August 2014; pp. 1–6.

21. Huang, X.; Liu, J.; Han, Z.; Yang, J. Privacy beyond sensitive values. *Sci. China Inf. Sci.* **2015**, *58*, 1–15. [CrossRef]
22. Jin, H.; Liu, S.; Ju, S. Privacy preserving for multiple sensitive attributes based on l-coverage. *High. Perform. Netw. Comput. Commun. Syst.* **2011**, *163*, 319–326.
23. Gal, T.S.; Chen, Z.; Gangopadhyay, A. A privacy protection model for patient data with multiple sensitive attributes. *Int. J. Inf. Secur. Priv.* **2008**, *2*, 28–44. [CrossRef]
24. Widodo, W.; Wibowo, W.C. A distributional model of sensitive values on *p*-sensitive in multiple sensitive attributes. In Proceedings of the 2nd International Conference on Informatics and Computational Sciences (ICICoS), Semarang, Indonesia, 30–31 October 2018; pp. 1–5.
25. Wu, Y.; Ruan, X.; Liao, S.; Wang, X. P-cover *k*-anonymity model for protecting multiple sensitive attributes. In Proceedings of the 5th International Conference on Computer Science and Education (ICCSE), Hefei, China, 24–27 August 2010; pp. 179–183.
26. LeFevre, K.; DeWitt, D.J.; Ramakrishnan, R. Incognito: Efficient full-domain k-anonymity. In Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data, Baltimore, MD, USA, 14–16 June 2005; pp. 49–60.
27. Lin, J.C.W.; Fournier-Viger, P.; Liu, Q.; Djenouri, Y.; Zhang, J. Anonymization of multiple and personalized sensitive attributes. In Proceedings of the 20th International Conference on Big Data Analytics and Knowledge Discovery, Regensburg, Germany, 3–6 September 2018; pp. 204–215.
28. Anjum, A.; Ahmad, N.; Malik, S.U.R.; Zubair, S.; Shahzad, B. An efficient approach for publishing microdata for multiple sensitive attributes. *J. Super. Comput.* **2018**, *74*, 5127–5155. [CrossRef]
29. Kanwal, T.; Shaukat, S.A.A.S.; Anjum, A.; Malik, S.U.R.; Choo, K.K.R.; Khan, A.; Ahmad, N.; Ahmad, M.; Khan, S.U. Privacy-preserving model and generalization correlation attacks for 1:M data with multiple sensitive attributes. *Inf. Sci.* **2019**, *488*, 238–256. [CrossRef]
30. Wang, R.; Zhu, Y.; Chen, T.; Chang, C. Privacy-preserving algorithms for multiple sensitive attributes satisfying *t*-closeness. *J. Comput. Sci. Technol.* **2018**, *33*, 1231–1242. [CrossRef]
31. Sowmyarani, C.N.; Srinivasan, G.N. A robust privacy preserving model for data publishing. In Proceedings of the 2015 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 8–10 January 2015; pp. 1–6.
32. Saraswathi, S.; Thirukumar, K. Enhancing utility and privacy using *t* closeness for multiple sensitive attributes. *Adv. Nat. Appl. Sci.* **2016**, *10*, 6–13.
33. Yang, X.; Wang, Y.; Wang, B. Privacy preserving approaches for multiple sensitive attributes in data publishing. *Chin. J. Comput.* **2008**, *31*, 574–587. [CrossRef]
34. Liu, Q.; Shen, H.; Sang, Y. Privacy-preserving data publishing for multiple numerical sensitive attributes. *Tsinghua Sci. Technol.* **2015**, *20*, 246–254.
35. Luo, F.; Han, J.; Lu, J.; Peng, H. ANGELMS: A privacy preserving data publishing framework for microdata with multiple sensitive attributes. In Proceedings of the 3rd International Conference on Information Science and Technology, Yangzhou, China, 23–25 March 2013; pp. 393–398.
36. Ye, Y.; Wang, L.; Han, J.; Qin, S. An anonymization method combining anatomy and permutation for protecting privacy in microdata with multiple sensitive attributes. In Proceedings of the 2017 International Conference on Machine Learning and Cybernetics (ICMLC), Ningbo, China, 9–12 July 2017; pp. 1–13.
37. Dhumal, M.T.S.; Patil, M.Y.S. Implementation of slicing for multiple column multiple attributes: Privacy preserving data publishing. *Int. J. Recent Innov. Trends Comput. Commun.* **2015**, *3*, 4261–4266.
38. Kiruthika, S.; Raseen, M.M. Enhanced slicing models for preserving privacy in data publication. In Proceedings of the 2013 International Conference on Current Trends in Engineering and Technology (ICCTET), Coimbatore, India, 3 July 2013; pp. 406–409.
39. Han, J.; Luo, F.; Lu, J.; Peng, H. SLOMS: A privacy preserving data publishing method for multiple sensitive attributes microdata. *J. Softw.* **2013**, *8*, 3096–3104. [CrossRef]
40. Onashoga, S.A.; Bamiro, B.A.; Akinwale, A.T.; Oguntuase, J.A. KC-Slice: A dynamic privacy preserving data publishing technique for multisensitive attributes. *Inf. Secur. J. Glob. Perspect.* **2017**, *26*, 121–135. [CrossRef]
41. Raju, N.V.S.L.; Seetaramanath, M.N.; Rao, P.S. A novel dynamic KCi-Slice publishing prototype for retaining privacy and utility of multiple sensitive attributes. *Int. J. Inf. Technol. Comput. Sci.* **2019**, *11*, 18–32. [CrossRef]
42. Reddy, S.R.P.; Raju, K.V.; Kumari, V.V. A novel approach for personalized privacy preserving data publishing with multiple sensitive attributes. *Int. J. Eng. Technol.* **2018**, *7*, 197–206. [CrossRef]

43. Susan, V.S.; Christopher, T. Anatomisation with slicing: A new privacy preservation approach for multiple sensitive attributes. *Springerplus* **2016**, *5*, 964. [CrossRef]
44. Ye, Y.; Liu, Y.; Wang, C.; Lv, D.; Feng, J. Decomposition: Privacy preservation for multiple sensitive attributes. In Proceedings of the 14th Database Systems for Advanced Applications, Brisbane, Australia, 21–23 April 2009; pp. 486–490.
45. Das, D.; Bhattacharyya, D.K. Decomposition+: Improving *l*-diversity for multiple sensitive attributes. In Proceedings of the second International Conference on Computer Science and Information Technology, Bangalore, India, 2–4 January 2012; pp. 403–412.
46. Liu, J.; Luo, J.; Huang, J. Rating: Privacy preservation for multiple attributes with different sensitivity requirements. In Proceedings of the 11th International Conference on Data Mining Workshops (ICDMW), Vancouver, BC, Canada, 11 December 2011; pp. 666–673.
47. Yi, T.; Shi, M. Privacy protection method for multiple sensitive attributes based on strong rule. *Hindawi Publ. Corp. Math. Probl. Eng.* **2015**, 464731. [CrossRef]