Enhancing the Utility of Generalization for Privacy Preserving Re-publication of Dynamic Datasets

Leela Rani. P
PG Scholar, Sri Venkateswara College of Engineering, Pennalur, Chennai-602105,Tamil Nadu

Revathi.N
Asst.Prof, Sri Venkateswara College of Engineering, Pennalur, Chennai-602105,Tamil Nadu

ABSTRACT
Anonymized publication on static micro data can be achieved with heavy information loss by Generalization. An enhanced utility of Generalization known as Angelization produces the same level of anonymization but with minimal information loss. In reality, there may be a need to publish another version of micro data, after insertions and deletions. Anonymization is applicable to any generalization principles like k-Anonymity, l-diversity and t-closeness. Incremental m-invariance with Angelization preserves privacy in re-publication of dynamic micro data after insertions and deletions. Mondrian algorithm is used for the partitioning in Angelization. m-invariance also supports publication of marginals from the generalized micro data. KL-divergence is employed for quantifying the discrepancy of two distributions. The reconstruction error will be measured as the KL-divergence between the reconstructed distribution and the original distribution. Data reconstruction error is minimal in m-invariance with enhanced utility of Generalization.

General Terms
Retrieval Model, Information Search and Retrieval.

Keywords
Privacy, Generalization, ANGEL, m-invariance, dynamic data set.

1. INTRODUCTION
The amount of data being collected every day by private and public organizations is quickly increasing. In such a scenario, data mining techniques are becoming more and more important for assisting decision making processes and to extract hidden knowledge from massive data in the form of patterns, models and trends. While not explicitly containing the original actual data, data mining results could potentially be exploited to infer information from the original data and not intended for release, which may breach the privacy of the parties to whom the data refer. Effective application of data mining can take place only if the privacy of the underlying data is not compromised.

The concept of privacy preserving data mining has been proposed for providing privacy. Privacy preserving data mining aims at providing a trade-off between sharing information for data mining analysis, on the one side, and protecting information to preserve the privacy of the involved parties on the other side. Several privacy preserving data mining approaches have been proposed, which usually protect data by modifying them to mask or erase the original sensitive data that should not be revealed. These approaches typically are based on the concepts of: loss of privacy, measuring the capacity of estimating the original data from the modified data and loss of information, measuring the loss of accuracy in the data. In general, the more the privacy of the respondents to which the data refer, the less accurate the result obtained by the miner and vice versa.

2. GENERALIZATION
Privacy preserving publication has received considerable attention from the database community. Let T be a table containing sensitive information. Table T is called as micro data. The aim is to release a modified version of T such that the modified version prevents opponents from inferring the sensitive data of T, but allows researchers to understand useful correlations in T. Consider that a hospital wants to publish Table 1. Attribute Disease is sensitive, that is, the publication must prevent the disease of any patient from being discovered.

Just removing the names is insufficient due to the possibility of linking attacks [9]. Consider an adversary that knows the age 21 and gender M of Alan. Given Table 1 (without the names), he is still able to assert that the first tuple must belong to Alan and thus find out his disease pneumonia. As the combination of Age and Sex can be used to recover a patient's identity, they are referred to as quasi-identifier (QI)-attributes.

A popular method to overcome linking attacks is Generalization. It is done by replacing QI-values in the micro data with fuzzier, less specific values. Table 2 is a generalized version of Table 1. The age 21 of the first tuple in Table 1 has been replaced with an interval [21, 40] in Table 2. Generalization creates QI-groups, each of which consists of tuples with identical and generalized QI-values. Table 2 has two QI-groups, including the first and last four tuples, respectively. Consider an opponent who knows Alan's age and gender. Given Table 2, he cannot infer exactly which of the first four tuples belongs to Alan. With a random guess, the opponent can find out Alan's disease as pneumonia only with 50 percent probability.

The E-M Generalization Modeling [14] can be formalized as:

Let T be a micro data table, which contains d quasi-identifier (QI) attributes and a sensitive attribute (SA) X. The core of generalization is to

1. Divide the tuples of T into a set E of disjoint equivalence classes (ECs), and
2. Transform the QI-values of the tuples in each EC to the same format.

QI-values in all ECs follow strict global recoding [5].Generalization follows anonymization principle, which helps in deciding whether a table has been adequately anonymized to guarantee privacy.

Table 1. Sample Micro data

NAME	AGE	SEX	DISEASE
Alan	21	M	Pneumonia
Clara	23	M	Pneumonia
Carrie	38	F	Bronchitis
Daisy	40	F	Bronchitis
Eddy	41	M	Pneumonia
Frank	43	M	Pneumonia
Gloria	58	F	Bronchitis
Helena	60	F	Bronchitis

Table 2. Generalization

AGE	SEX	DISEASE
[21,40]	*	Pneumonia
[21,40]	*	Pneumonia
[21,40]	*	Bronchitis
[21,40]	*	Bronchitis
[41,60]	*	Pneumonia
[41,60]	*	Pneumonia
[41,60]	*	Bronchitis
[41,60]	*	Bronchitis

Three commonly used anonymization principles are *k*-anonymity [8], [10], [11],*l*-diversity [7], and *t*-closeness [6]. In order to reduce the risk of identification, the *k*-anonymity approach requires that every tuple in the table be indistinguishably related to no fewer than k respondents. Formalization of *k*-anonymity is as follows: "Each publication of the data must be such that every combination of values of quasi-identifiers can be indistinguishably matched to at least k respondents". *K*-anonymity is not effective in preventing the inference of the sensitive values of the attributes of a record. *l*-diversity, in addition to maintaining the minimum group size of *k*, also maintains the diversity of the sensitive attributes. *l*-diversity model for privacy is formalized as follows: Let a q*-block be a set of tuples such that its non-sensitive values generalize to q*. A q*-block is *l*-diverse if it contains *l* "well represented" values for the sensitive attribute S. The *t*-closeness model is an enhancement of *l*-diversity. *t*-closeness model uses the property that the distance between the distribution of the sensitive attribute within an anonymized group should not be different from the global distribution by more than a threshold *t*. The Earth Mover distance metric is used in order to quantify the distance between the two distributions. An anonymization principle is monotonic if the following is true: given any two multi sets of sensitive values S1 and S2 whose SA-distributions obey the principle, the SA-distribution of the union S1US2 also should obey the principle. Monotonicity is important for performing a top-down pruning strategy in forming a generalization [2] where as Space-filling curves are used for generalization in [3].

3. OVERVIEW OF THE PROJECT

This paper deals with re- publication of dynamic data using m-invariance after performing angelization on micro data. The first step in angelization is batch partitioning. In batch partitioning, micro data is partitioned into batches. The second step is bucket partitioning. Here, micro data is partitioned into buckets. The final step is generalization of tuples in each bucket to the same form. This results in production of Generalized Table (GT).GT will not contain the sensitive attribute. It contains the Batch ID of the batch to which the tuple belongs. Releasing versions of generalized micro data is called as Marginal publication. Marginal publication is supported by Angelization. It is capable of releasing one Batch Table (BT) and any number of marginals without privacy breach. Dynamic data set can be modified by any sequence of insertions and deletions. The process of re-publication is incremental, where in, the publisher can publish n-th release, by consulting only the data of the last release using m-invariance[12].The data reconstruction error is less in m-invariance with Angelization when compared to m-invariance with Generalization,thus concluding that it offers great degree of privacy and less loss of information.

4. ANGELIZATION FOR PUBLICATION OF STATIC DATA

4.1 Angel Technique

Let T be the micro data and P be the objective anonymization principle (e.g., 2-diversity). Definition of Batch is as follows:

A batch partitioning of the micro data T consists of batches $B_1, B_2, \ldots, B_b$ such that

1. Each batch is a set of tuples in T;
2. Union of all batches form T and, for any $i \neq j$, $B_i \cap B_j = \emptyset$;
3. The SA-distribution in each batch B_i ($1 \leq i \leq b$) satisfies principle P.

The subscript i of batch B_i is the Batch-ID of B_i. Definition of Bucket is as follows: A bucket partitioning consists of buckets $C_1, C_2, \ldots C_e$ such that

1. Each bucket is a set of tuples in T;
2. Each bucket contains at least k tuples, where k is a parameter controlling the degree of protection against presence attacks;
3. Union of all buckets form T and, for any $i \neq j$, $C_i \cap C_j = \emptyset$;

In ANGEL, any pair of bucket and batch partitioning determines an anonymized publication called Angelization [14].

To publish the micro data of Table 1, conforming to 2-diversity [7], Batch Partitioning is done. In this partitioning, the table is divided into batches as follows:

Batch1: {Alan, Carrie},
Batch2: {Clara, Daisy},
Batch3: {Eddy, Gloria},
Batch4: {Frank, Helena}.

Each batch obeys 2-diversity: it contains one pneumonia and one bronchitis tuple. ANGEL creates a batch table (BT), as in Table 3. For example, the first row of Table 3 states that exactly one tuple in Batch 1 carries pneumonia. Then, ANGEL creates bucket partitioning of Table 4 (which do not have to be 2-diverse).

Bucket1: {Alan, Clara},
Bucket2: {Carrie, Daisy},
Bucket3: {Eddy, Frank},
Bucket4: {Gloria, Helena}.

ANGEL generalizes the tuples of each bucket into the same form producing a generalized table (GT).Table 4 shows the GT. GT has the ID of the batch containing each tuple. The final output of Angelization has two tables which are obtained as a result of batch partitioning and bucket partitioning. Tables 3 and 4 are the final relations published by ANGEL.

Table 3. ANGEL Publication – Batch Table

Batch ID	DISEASE	Count
1	Pneumonia	1
1	Bronchitis	1
2	Pneumonia	1
2	Bronchitis	1
3	Pneumonia	1
3	Bronchitis	1
4	Pneumonia	1
4	Bronchitis	1

Table 4. ANGEL Publication – Generalized Table

AGE	SEX	Batch ID
[21,23]	M	1
[21,23]	M	2
[38,40]	F	1
[38,40]	F	2
[41,43]	M	3
[41,43]	M	4
[58,60]	F	3
[58,60]	F	4

4.2 Method of Finding Angelization

This method leverages 2 algorithms AL_P and AL_k that compute a simple generalization following principles P and k-anonymity, respectively.

There are three steps:

1. AL_P is implemented to get a simple generalization T^*_p of micro data T. The set of QI-groups in T^*_p is batch partitioning.
2. AL_k is executed to acquire a simple generalization T^*_k of micro data T and the QI-groups obtained constitute a bucket partitioning.
3. Discarding T^*_p and T^*_k, angelization is derived from the batch and bucket partitioning.

Given a batch partitioning $\{B_1;B_2; \ldots ;B_b\}$ of the micro data T and a bucket partitioning $\{C_1; C_2; \ldots ; C_e\}$, an angelization of T is a pair of a BT and a GT such that

1 BT has three columns: {Batch-ID, X, Count}, where X is the sensitive attribute of T. For every batch B_i ($1 \leq i \leq m$) and every sensitive value $x \in X$ that appears in B_i, BT has a row (i, x, y), where y is the number of occurrences of x in B_i.

2 GT has all the QI-attributes of T, together with an extra column Batch-ID. Every tuple $t \in T$ defines a row in GT, which stores the generalized QI-values of t, and the ID of the batch containing t. All tuples in the same bucket C_i ($1 \leq i \leq e$) have equivalent generalized QI-values.

4.3 Marginal Publication

Generalization loses less information when the number of QI-attributes is smaller [1]. In addition to publishing a large table that has all the QI-attributes, the publisher may also release certain projections as in Tables 5, 6 and 7 to enhance the understanding on the underlying correlations. This approach is called marginal publication [13].

ANGEL accomplishes the task by releasing one BT and s GTs. BT is shared by all marginals and every marginal has a GT of its own. s+1 tables are obtained through s+1 simple generalizations including a k-anonymous generalization (for obtaining the BT) and s generalizations conforming to P (one for each GT).Let AL_k be any algorithm for computing a k-anonymous generalization, and AL_P any algorithm for computing a generalization under P.

The following procedure is used for marginal publication by ANGEL:

1. AL_P is implemented on T to obtain a simple generalization T^*_p .The batch partitioning of T is decided according to the equivalence classes in T^*_p. The batch partitioning is denoted by E_P. E_P is a set, where each element is a batch.
2. For each marginal G_i ($1 \leq i \leq s$),AL_k is executed on Π_{Gi} (T) to obtain a simple generalization T^*_{ki}. The bucket partitioning of T is decided according to the equivalence classes in T^*_{ki}. The bucket partitioning is denoted by E_i, which is a set where each element is a bucket.
3. Having E_P and $E_1; \ldots ; E_s$, the BT and GTs can be formed. The batch partitioning E(.) alone uniquely decides a BT as follows. For every batch B in E(.) and every sensitive value x appearing in B, create a row (i, x, y) in the BT, where i is the batch-ID of B and y the number of occurrences of x in B.

Let T be a micro data table with a sensitive attribute X and P the anonymization principle selected by the publisher. It is needed to publish any marginals of T using ANGEL, while ensuring the guarantee of P. Without loss of generality, suppose that we need to release s marginals as in Tables 6 and 7, denoted by $G_1, G_2, \ldots ,G_s$, respectively. Each G_i ($1 \leq i \leq s$) is a set of attributes in T. Consider X appears in all of $G_1, G_2, \ldots ,G_s$. It is unnecessary to publish a marginal G that does not contain X—It is possible to release a k-anonymous generalization of G, without worrying about privacy breach.

Let the privacy principle P be 2-diversity and the parameter k of ANGEL be 2. To apply ANGEL, first choose two algorithms AL_P and AL_k for computing 2-diverse and 2-anonymous generalizations, respectively e.g., both AL_P and AL_k can be the Mondrian algorithm in [4]. ANGEL computes a BT and two GTs as follows. First, it applies AL_P to obtain a 2-diverse generalization T^*_P of T and derives a batch partitioning E_P from T^*_P. Let us assume that E_P has these three batches:

B_1 = {Alan; Clara} ; B_2 = {Carrie; Daisy}; B_3= {Eddy; Frank}

Second, ANGEL determines two bucket partitionings E_1 and E_2 for marginals G_1 and G_2, respectively. Specifically, E_1 comes from a 2-anonymous generalization of T (returned by algorithm AL_k). In the example, assume that E_1 includes three buckets:

C_1 = {Alan; Clara}; C_2 = {Carrie; Daisy}; C_3 = {Eddy; Frank}

Similarly, to obtain E_2, ANGEL takes the projection of T onto G_2, invokes AL_k to find a 2-anonymous generalization of the projection, and produces E_2 from the generalization. Assume that E_2 has these buckets C'_1={Alan;Carrie}; C'_2={Clara; Daisy};C'_3={Eddy; Frank}.

ANGEL determines the contents of a BT and two GTs from E_P, E_1, and E_2.

Table 5. Marginal Publication- Batch Table

Batch ID	DISEASE	Count
1	Pneumonia	1
1	Flu	1
2	Bronchitis	1
2	Pneumonia	1
3	Flu	1
3	Bronchitis	1

Table 6. Marginal Publication- First GT

AGE	SEX	Zipcode	Batch ID
[21,23]	M	[10k,58k]	1
[21,23]	M	[10k,58k]	1
[58,60]	F	[12k,60k]	2
[58,60]	F	[12k,60k]	2
[70,72]	M	[78k,80k]	3
[70,72]	M	[78k,80k]	3

Table 7. Marginal Publication- Second GT

Zipcode	Batch ID
[10k,12k]	1
[10k,12k]	2
[58k,60k]	1
[58k,60k]	2
[78k,80k]	3
[78k,80k]	3

5. EMPLOYING M-INVARIANCE WITH ANGELIZATION FOR DYNAMIC DATA

5.1 Proposed Architecture

Figure 1 shows the proposed architecture which aims to preserve privacy of dynamic micro data using m-invariance with angelization, an enhanced utility of generalization. Privacy preserving publication of dynamic data after sequence of insertions and deletions is referred to as re-publication. This system also supports publication of marginals. After Angelization is done for the current data set T1 as shown in Table 8, the release may be published as in Table 9. If the data set is modified by any sequence of additions and deletions, then angelization is performed on the modified data set T2 as depicted in Table 10.Then m-invariance is applied on the angelized tables of T1 and T2 and a generalized table is obtained. The generalized table also supports marginal releases. This is a generalization principle whose satisfaction ensures strong protection of sensitive information in re-publication. This solution includes the integration of two concepts: m-invariance and counterfeited generalization.

Let QI* be a QI group in T*(j) for any j ϵ [1, n]. The signature of QI* is the set of distinct sensitive values in QI*. If a sensitive value is present in multiple tuples in QI*, the value appears only once in the signature.

m-Invariance is formalized as follows:
A generalized table T*(j) ($1 \leq j \leq n$) is m-unique, if each QI group in T*(j) contains at least m tuples, and all the tuples in the group have different sensitive values.
A sequence of published relations T*(1), ..., T*(n) (where $n \leq 1$) is m-invariant if the following conditions hold:

1. T*(j) is m-unique for all j ϵ [1, n].
2. For any tuple t ϵ U(n) with lifespan [x, y], t.QI*(x),t.QI*(x + 1), ..., t.QI*(y) have the same signature.

m-uniqueness demands that each sensitive value should appear at most once in every QI-group. m-invariance algorithm has 4 phases:

1. Division
2. Balancing
3. Assignment
4. Split

The calculation of T*(n) requires only micro data tables T(n − 1), T(n), and the last published relation T*(n − 1).The tuples in T(n) are divided into two disjoint sets $S_{\cap} = T(n) \cap T(n-1)$ and $S_{-} = T(n) - T(n-1)$.
m-invariance algorithm ensures two properties:

1. For any tuple t ϵ $S_{\cap}$, its generalized hosting groups t.QI*(n − 1) and t.QI*(n) have the same signature.
2. For any tuple t ϵ S_{-}, its generalized tuple t* in T*(n) is in a QI group which has at least m tuples and all the tuples have distinct sensitive values.

Consider that a hospital releases patients' records once in 3 months, but each publication includes only the results of diagnosis done in the past 6 months before the publication. Table 8 shows the micro data for the first release, at which time the hospital publishes the Angelized relation in Table 9.
The micro data at the second release is presented in Table 10. The tuples of Alice, Anna, Helen, Kelly, and Paul have been deleted, while 5 new tuples have been inserted. Since each tuple in the micro data may be involved in any number of subsequent publications until it is deleted, there are too many correlations among various micro data that may be utilized to derive sensitive values. A new generalization principle is needed, which provides privacy, in spite of inferences that may lead to any possible correlations. Such a principle may not exist. This is called as critical absence.
Accordingly, the hospital publishes the Angelized relation in Table 11. Both the published relations (Tables 9 and 11) are 2-anonymous and 2-diverse. An adversary can determine the disease of a patient by exploiting the correlation between the two releases. Assume an opponent who knows Clara's age and Zip code and knows that Clara has a record in both Tables 9 and 11 i.e., Clara was admitted for treatment within 6 months. Based on Table 10, the opponent is certain that Clara must have contracted either dyspepsia or bronchitis. From Table 11, he finds out that Clara's disease must be either dyspepsia or gastritis. Combining the above information, the opponent identifies Clara's actual disease as dyspepsia.

This can be explained by re-examining the first release from the hospital. Given Table 9, an opponent is sure that Clara

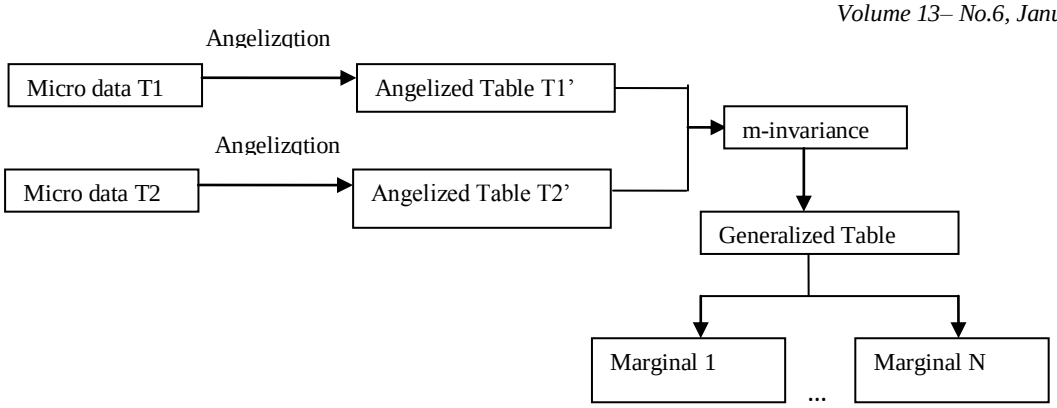

Figure 1. Proposed Architecture

contracted dyspepsia or bronchitis, provided he has some background information about Clara. Bronchitis is absent in the micro data Table 10 at the second release. So, publishing the generalized version of Table 10 always enables the adversary to eliminate the possibility that Clara contracted bronchitis. There is a possibility that Clara's privacy will be breached after the second release.

Table 8. Micro data T1 of first release

Name	Age	Zip.	Disease
Clara	21	12000	dyspepsia
Alice	22	14000	bronchitis
Anna	24	18000	flu
Dorothy	23	25000	gastritis
Gary	41	20000	flu
Helen	36	27000	gastritis
Jane	37	33000	dyspepsia
Kelly	40	35000	flu
Linda	43	26000	gastritis
Paul	52	33000	dyspepsia
Steve	56	34000	gastritis

Table 9. Generalization of T1 at first release

G. ID	Age	Zip.	Disease
1	[21, 22]	[12k, 14k]	dyspepsia
1	[21, 22]	[12k, 14k]	bronchitis
2	[23, 24]	[18k, 25k]	flu
2	[23, 24]	[18k, 25k]	gastritis
3	[36, 41]	[20k, 27k]	flu
3	[36, 41]	[20k, 27k]	gastritis
4	[37, 43]	[26k, 35k]	dyspepsia
4	[37, 43]	[26k, 35k]	flu
4	[37, 43]	[26k, 35k]	gastritis
5	[52, 56]	[33k, 34k]	dyspepsia
5	[52, 56]	[33k, 34k]	gastritis

5.1.1 Division Phase

For each tuple t ϵ $S_{\cap}$, signature is defined as the signature of its generalized hosting group in T_(n − 1). This phase simply partitions $S_{\cap}$ into several buckets, such that each bucket contains only the tuples with the same signature. In the example, $S_{\cap}$ contains the tuples of Clara, Dorothy, Jane, Linda, Gary, and Steve. Figure 2 shows the contents of the buckets after this phase. The tuple of Clara has a signature {dyspepsia, bronchitis} (i.e., the sensitive values in Group 1 of Table 11). It is the only element in bucket BUC3. A bucket can have multiple tuples. For example, BUC1 contains Gary and David, since they share an equivalent signature {flu, gastritis}.

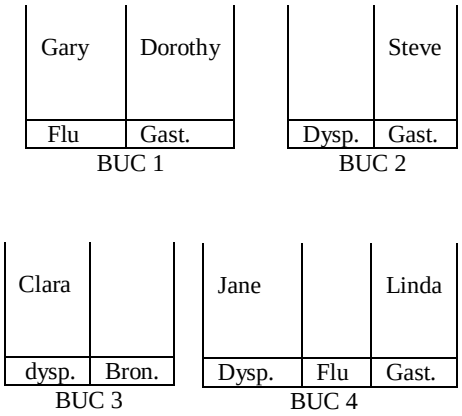

Figure 2. Bucket contents after Division phase

5.1.2 Balancing Phase

This phase considers tuples' sensitive values. A bucket BUC is balanced if every sensitive value in its signature is owned by the same number of tuples in BUC. BUC1 is balanced, since its signature has two values flu and gastritis, each of which is possessed by a tuple. The objective of this phase is to balance all buckets.

Each bucket BUC is inspected. If BUC is not balanced, there is a shortage of some sensitive values in BUC. The shortage is rectified by moving the tuples in S_{-} into BUC, as long as the resulting S_{-} is still m-eligible.BUC2 is unbalanced, because there is one gastritis but no dyspepsia. S_{-} equals {Emily, Mary, Ray, Tom, Vince}. Ray, whose disease value is dyspepsia, can be moved to BUC2, because there are 2 flu and 2 gastritis in S_{-}, which is still 2-eligible. The updated BUC2, shown in Figure 3, becomes balanced.

If S_{-} cannot be used to fix an unbalanced bucket BUC, there are two possibilities:

1. no tuple in S_{-} carries the required sensitive value, or
2. S_{-} is no longer m-eligible after a tuple removal.

In both cases, counterfeits are inserted to balance BUC. In Figure 2, both BUC3 and BUC4 are unbalanced, but neither of them can be remedied with S_-.BUC3 needs a bronchitis, which is absent in S_-. BUC4 needs a flu; although there are tuples with flu in S_-, removing any of them leaves 2 gastritis and 1 flu in S_-, violating the 2-eligibility constraint. Therefore, as in Figure 3, two counterfeits c1 and c2 (with sensitive values bronchitis and flu) are added to BUC3 and BUC4 respectively, both of which are now balanced. Each counterfeit has a value null on every QI attribute.

Table 10. Micro data T2 of Second release

Name	Age	Zip.	Disease
Clara	21	12000	dyspepsia
Dorothy	23	25000	gastritis
Emily	25	21000	flu
Jane	37	33000	dyspepsia
Linda	43	26000	gastritis
Gary	41	20000	flu
Mary	46	30000	gastritis
Ray	54	31000	dyspepsia
Steve	56	34000	gastritis
Tom	60	44000	gastritis
Vince	65	36000	flu

Table 11. Generalization of T2 at Second release

G. ID	Age	Zip.	Disease
1	[21, 23]	[12k, 25k]	dyspepsia
1	[21, 23]	[12k, 25k]	gastritis
2	[25, 43]	[21k, 33k]	flu
2	[25, 43]	[21k, 33k]	dyspepsia
2	[25, 43]	[21k, 33k]	gastritis
3	[41, 46]	[20k, 30k]	flu
3	[41, 46]	[20k, 30k]	gastritis
4	[54, 56]	[31k, 34k]	dyspepsia
4	[54, 56]	[31k, 34k]	gastritis
5	[60, 65]	[36k, 44k]	gastritis
5	[60, 65]	[36k, 44k]	flu

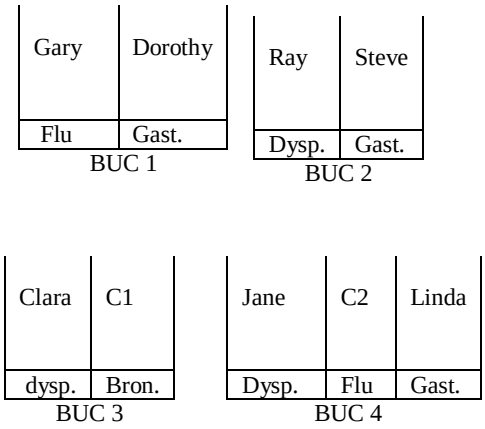

Figure 3. Bucket contents after Balancing phase

5.1.3 Assignment Phase

In this phase, the remaining tuples in S_- are assigned to buckets, subject to two rules.

1) Each tuple $t \in S_\cap$ can be placed only in a bucket whose signature includes $t[A^s]$.
2) At the end of the phase, all buckets are still balanced. If necessary, new buckets (each bucket's signature contains at least m values) may be generated, and they also obey these rules.

An assignment scheme always exists, as long as S_- is m-eligible. In the example,
S_- = {Emily, Mary, Tom, Vince} after the balancing phase. Figure 4 illustrates the buckets after all assignments. The 4 tuples in S_- are all placed in BUC1, which remains balanced.

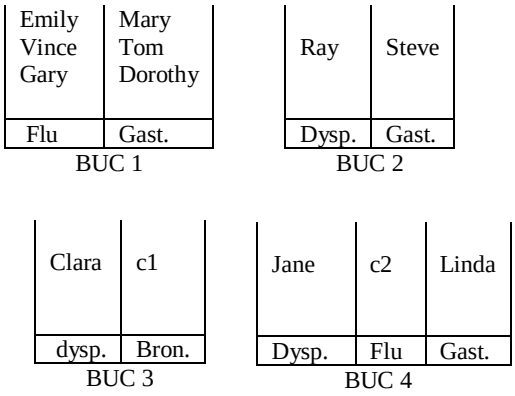

Figure 4. Bucket contents after Assignment phase

5.1.4 Split Phase

This last phase processes each bucket individually. It splits BUC into |BUC|/s QI groups, where s (≥m) is the number of values in the signature of BUC. Each group has s tuples, taking the s sensitive values in the signature, respectively.

Splitting optimizes the quality of generalization. Let $t_1, t_2, ..., t_s$ be the tuples in a group. Their generalized tuples form a QI group QI* in the published T*(n). On each QI attribute A_i^{qi} ($1 \le i \le d$), $QI^*[A_i^{qi}]$ is the minimum interval enclosing all $t_1[A_i^{qi}], ..., t_s[A_i^{qi}]$.Therefore, splitting aims at minimizing the length sum of intervals $QI^*[A_1^{qi}], ..., QI^*[A_d^{qi}]$. Given BUC1, the split algorithm creates three QI groups: {David, Emily}, {Gary, Mary}, and {Vince, Tom}. They lead to QI groups 2, 4, and 6 in Table 13. Similarly, BUC2, BUC3, and BUC4 result in QI groups 5, 1, 3 respectively. Considering the age [21, 22] of QI group 1 in Table 13, this group covers the tuple of Clara (age 21) and a counterfeit (age ø). Hence, it is required that each QI value in T*(n) should be an interval whose length is at least a threshold (e.g., 2 for Age). This threshold may vary for different QI attributes (e.g., 2k for Zipcode).The Counterfeit statistics is also released to the researchers as in Table 12.

Let BUC be a balanced bucket output by the assignment phase, whose signature has s≥m sensitive values v1, v2, ..., vs. The split phase starts by initiating a set $S_{buc} = \{BUC\}$. If BUC includes more than s tuples, it will be removed from S_{buc}, and split BUC into two balanced buckets BUC1 and BUC2 with the same signature as BUC. BUC1 and BUC2 are then added to S_{buc}.

Table 12. Published Counterfeit Statistics

Group-ID	Count
1	1
3	1

If any bucket in S_{buc} still has a size over s, BUC is set to that bucket, and the above procedures are repeated. The phase terminates when all the buckets in S_{buc} contain exactly s tuples. They are returned as the QI groups for generalization. Totally |BUC| / (s −1) bucket splits are performed.

Table 13. Remedying Critical absence with Counterfeits

Name	G. ID	Age	Zip.	Disease
Clara	1	[21, 22]	[12k, 14k]	dyspepsia
c1	1	[21, 22]	[12k, 14k]	bronchitis
Dorothy	2	[23, 25]	[21k, 25k]	gastritis
Emily	2	[23, 25]	[21k, 25k]	flu
Jane	3	[37, 43]	[26k, 33k]	dyspepsia
c2	3	[37, 43]	[26k, 33k]	flu
Linda	3	[37, 43]	[26k, 33k]	gastritis
Gary	4	[41, 46]	[20k, 30k]	flu
Mary	4	[41, 46]	[20k, 30k]	gastritis
Ray	5	[54, 56]	[31k, 34k]	dyspepsia
Steve	5	[54, 56]	[31k, 34k]	gastritis
Tom	6	[60, 65]	[36k, 44k]	gastritis
Vince	6	[60, 65]	[36k, 44k]	flu

BUC is organized into s groups, such that the j-th ($1 \leq j \leq s$) group contains only the tuples with the sensitive value vj . Clearly, every group has size |BUC|/s. Then, the tuples in each group are sorted in ascending order of their values (Ø precedes all non-empty values in sorting). Lj denotes the sorted list of the j-th ($1 \leq j \leq s$) group.

5.2 Mondrian Algorithm

The algorithm used in batch and bucket partitioning in Angelization is Mondrian algorithm[4].The algorithm is summarized as follows:

A Private Table PT as shown in Table 14 is represented as a set of points in a multidimensional space, where each attribute represents one dimension. For computing a k-anonymous table, the multidimensional space is partitioned in regions that have to contain at least k points as depicted in Figure 5. All points in a given region are then generalized to the same value for QI. Given a space region r, at each iteration the algorithm chooses a dimension d and splits the region at the median value x of d: all points such that d>x will belong to one of the resulting regions, while all points with d≤x will belong to the other region. The algorithm terminates when no more splitting operations are allowed. The tuples with in a given region are then generalized to a unique tuple of summary statistics for the considered region.

In Figure 5, the attributes Marital_status and Gender are represented as the y-axis and x-axis respectively. The number of tuples are plotted accordingly.

Table 14. Example Table for Mondrian Algorithm

Marital_status	Sex	Hours	#tuples
Divorced	M	35	2
Divorced	M	40	17
Divorced	F	35	2
Married	M	35	10
Married	F	50	9
Single	M	40	26

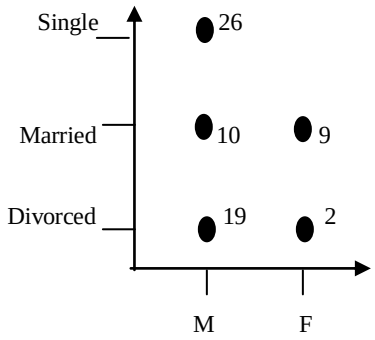

Figure 5. Spatial Representation

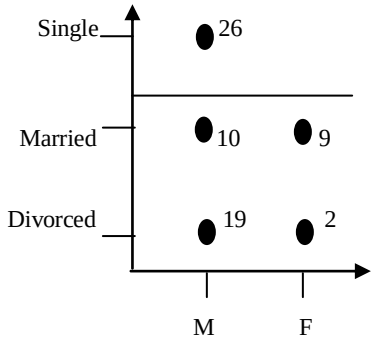

Figure 6. Possible Partitioning-1

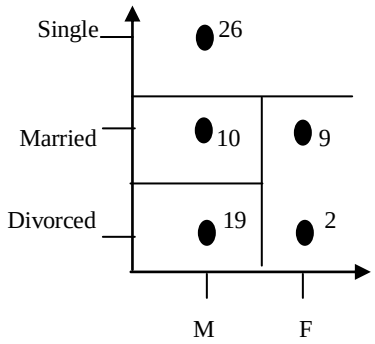

Figure 7. Possible Partitioning-2

The first partitioning is formed with the tuple with value for number of tuples as 26 and the remaining tuples form the second partitioning as in Figure 6. With further application of

Mondrian Algorithm, Since the value of k=10, two more partitionings as shown in Figure 7 is possible. The algorithm terminates when there are no further possible partitioning.

5.3 Marginal Publication on Dynamic Data

The steps are the same as described in section 4.3. Using this approach, any number of versions of the private table can be released as marginals without privacy breach after performing Angelization and m-invariance on dynamic data.

6. CONCLUSION

Experimental Results have shown that re-publication of dynamic data with angelization has minimal data reconstruction error when compared to re-publication with generalization. This is depicted in Figure 8.

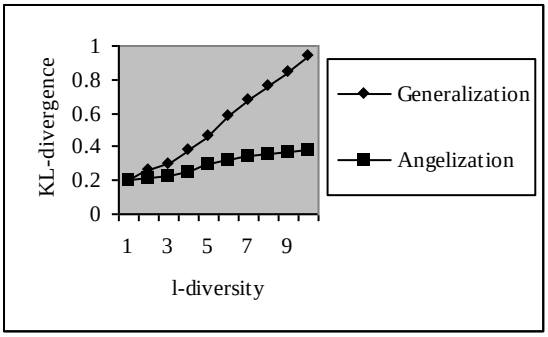

Figure 8. Data Reconstruction error comparing m-invariance with Generalization and m-invariance with Angelization

KL-divergence is used for quantifying the data reconstruction. m-invariance, with enhanced utility of Generalization, i.e., Angelization ensures privacy preservation in re-publication of dynamic data and also in publishing marginals. The data reconstruction error is less in m-invariance with Angelization when compared to m-invariance with Generalization, thus concluding that it offers greater degree of privacy and less loss of information. The distribution involves all the QI- and sensitive attributes. So, the accuracy indicates the captured correlation in the original micro data by Angelization and m-invariance. Figure 8 demonstrates the KL-divergence as a function of l, where l-diversity is the underlying anonymization principle. Angelization incurs lower error than generalization and the gap increases, when more stringent privacy protection is enforced with a larger l. It would be challenging to determine the ways to use the distribution reconstructed from angelization to perform advanced data mining techniques such as decision tree classification, association rule mining etc. Another scope of study in this would be to ascertain whether it is possible to obtain better angelization directly, without using generalization and then employ m-invariance for dynamic data sets.

7. ACKNOWLEDGMENTS

My sincere thanks to our Principal Dr. R. Ramachandran, Sri Venkateswara College of Engineering for being a source of inspiration throughout my study in this college. I express my sincere thanks to the Head of the Department, Dr. M. Gopala Krishnan and Ms.N. Revathi, Ms.S. Pushpa, Ms.P. Raja Lakshmi Asst. Professors, for their valuable guidance and suggestions.

8. REFERENCES

[1] C.C. Aggarwal 2005. On K-Anonymity and the Curse of Dimensionality.Proc.Int'l.Conf.Very Large Data Bases (VLDB). pp.901-909.

[2] R. Bayardo and R. Agrawal 2005. Data Privacy through Optimal k-Anonymization. Proc. Int'l Conf. Data Eng. (ICDE). Pp. 217-218.

[3] G. Ghinita, P. Karras, P. Kalnis and N. Mamoulis 2007 Fast Data Anonymization with Low Information Loss. Proc. Int'l Conf. Very Large Data Bases (VLDB).pp. 758-769.

[4] K. LeFevre, D. J. Dewitt and R.Ramakrishnan. 2006. Mondrian Multidimensional k-Anonymity. Proc. Int'l Conf. Data Eng. (ICDE). pp.277-286.

[5] K. LeFevre, D. J. Dewitt and R.Ramakrishnan. 2005. Incognito:Efficient Full-Domain k-Anonymity. Proc. ACM SIGMOD Int'l Conf. Management of Data. pp. 49-60.

[6] N. Li, T. Li and S. Venkatasubramanian. 2007. t-closeness: Privacy beyond k-Anonymity and l-diversity. Proc. Int'l Conf Data Eng. (ICDE). pp. 106-115.

[7] A. Machanavajjhala, J. Gehrke, D. Kifer and M. Venkitasubramaniam. 2006. l-diversity:Privacy beyond k-Anonymity. Proc.Int'l Conf. Data Eng.(ICDE). pp. 24.

[8] H.Park and K.Shim. 2007. Approximate Algorithms for k-Anonymity. Proc. ACM SIGMOD Int'l Conf. Management of Data. Pp-67-78.

[9] V. Rastogi, S.Hong and D.Suciu. 2007. The Boundary between Privacy and Utility in Data Publishing. Proc. Int'l Conf. Very Large Data bases (VLDB) .pp. 531-542.

[10] P.Samarati. 2001.Protecting Respondents' Identities in Micro data Release. IEEE Trans. Knowledge and Data Eng. Vol. 13.no. 6.pp. 1010-1027.

[11] L.Sweeny. 2002. K-Anonymity: A Model for protecting privacy. Int'l J. Uncertainty,Fuzziness and Kowledge-based Systems.vol.10.no.5. pp. 557-570.

[12] X. Xiao and Y. Tao. 2007. m-Invariance: Towards Privacy Preserving Re-Publication of Dynamic Data Sets. Proc. ACM SIGMOD Int'l Conf. Management of Data. pp. 689-700.

[13] C. Yao, X.S. Wang, and S. Jajodia 2005. Checking for k-Anonymity Violation by Views. Proc. Int'l Conf. Very Large Data Bases (VLDB), pp. 910-921.

[14] Yufei Tao, Hekang Chen, Xiaokui Xiao, Shuigeng Zhou, Member, IEEE Computer Society, and Donghui Zhang 2009, ANGEL: Enhancing the Utility of Generalization for Privacy Preserving Publication. IEEE Transaction on Knowledge and Data Engineering.Vol 21.No.7.pp.1073-1087.