

C-mixture and Multi-constraints Based Genetic Algorithm for Collaborative Data Publishing

Yogesh Kulkarni¹, Senthil Murugan¹

¹Vel Tech University

India

(yogeshrk2015, senthilmuruganme)@gmail.com



ABSTRACT: Due to the increasing need of using distributed databases, a high demand on sharing data to easily update and access the useful information without any interruption. The sharing of distributed databases cause a serious issue of securing information since the databases consists of sensitive personal information. To preserve the sensitive information and at the same time, releasing the useful information, a significant effort is made by the researchers under privacy preserving data publishing that have been received considerable attention in recent years. In this work, a new privacy measures, called c-mixture is introduced to maintain the privacy constraint without affecting utility of the database. In order to apply the proposed privacy measure to privacy preserving data publishing, a new algorithm called, CPGEN is developed using genetic algorithm and multi-objective constraints. The proposed multi-objective optimization considered the multiple privacy constraints along with the utility measurement to measure the importance. Also, the proposed CPGEN is adapted to handle the cold-start problem which is commonly happened in distributed databases. The proposed algorithm is experimented with adult dataset and quantitative performance is analyzed using generalized information loss and average equivalence class size metric. From the experimentation, we proved that the proposed algorithm maintained the privacy and utility as compared with the existing algorithm.

Keywords: Privacy, Utility, Distributed Databases, Data Publishing, Optimization, Sensitive Information

Received: 11 March 2017, Revised 9 April 2017, Accepted 19 April 2017

©2017 DLINE. All Rights Reserved.

1. Introduction

Many organizations bring out microdata-tables, which contain unaggregated information about persons. Medical, voter registration, census, and customer data can be included in these microdata tables. Microdata are important source of information for the allotment of public funds, medical research, and trend analysis. Still, if the persons can be individually identified in the microdata, then their private information such as their medical condition would be revealed, and

it is undesirable [16]. As an example, in the healthcare field, a national agenda is to form the Nationwide Health Information Network (NHIN) to share information among hospitals and other providers, and support proper use of health information further than direct patient care with privacy protection [1]. It is important that the sensitive information about the individuals should not be revealed when the microdata is released. Two kinds of information disclosures have been found out in the literature are identity disclosure and attribute disclosure. Identity disclosure happens when a person is connected to a particular record in the released table. Attribute disclosure occurs when new information about some individuals is revealed, i.e., the released data makes it feasible to understand the uniqueness of an individual more precisely than it would be possible before releasing the data [19-25].

The most important concepts for privacy is anonymity. Anonymity refers to a state where one's identity is completely concealed, and anonymity is sometimes used as a synonym for privacy [8]. Eventhough k -anonymity prevents from identity disclosure, it is inadequate to avoid attribute disclosure. Anonymous data can protect individuals in two ways. One way is to protect identity privacy, for example by making it not possible to learn to whom a data record is associated. The second way is through attribute privacy, for example making it not possible to know about a specific property of individuals. When collecting databases such as health records, collected by hospitals or government organizations, anonymity has a major role to protect privacy as the information connected to individuals are very sensitive [4]. There are two methods to achieve in k -anonymizing a dataset. First one is suppression, which involves not releasing an entire tuple or a value at all to the third party. Second one is generalization which involves replacing the value or tuple with a less definite but semantically constant value.

The researchers are paying attention towards the research on privacy-preserving data publishing [12-13]. Anonymization techniques includes (1) hiding the identities by making each record indistinguishable from at least $k-1$ other records [9] (k -anonymity), (2) making sure that the distance between the distribution of sensitive attributes in a class of records and the distribution of them in the whole table is not more than ' t ' [10] (t -closeness), and (3) making sure that there are at least l different values for a given sensitive attribute in each indistinguishable group of records [11] (l -diversity). The m -invariance is one of the representative models [14]. The basic idea of these techniques is unchanging the set of sensitive attribute values in the group that a tuple belongs to, even if the tuple may be put into different groups in different versions of the microdata. With these restrictions, the present privacy standard is proposed to make them to maintain the better tradeoff between data quality and privacy [15].

In this paper, a new algorithm called, CPGEN (C-mixture based privacy genetic algorithm) is developed by combining the genetic algorithm with c -mixture theory which is newly developed here for privacy measurements. C -mixture is a new privacy measure developed in this work by integrating the multiple privacy constrains such as, k -anonymity, l -diversity and m -privacy. In addition with, cold-start attack is defined based on the real time fact that one provider provides large number of records but other one share less number of data records with more distribution among every group. In order to alleviate this attack, noisy records are inserted into the databases. At first, the input data is directly given to the genetic algorithm which considered the unique encoding of chromosomes for doing anonymization process and the fitness is evaluated using the proposed multi-objective function which considered both utility through generalized information loss and privacy through average equivalence class size metric. Based on the objective function, the better chromosome is selected and it is used further to construct the anonymized data.

The paper is organized as follows: Section 2 explains literature review and section 3 provides C -mixture principle and its definitions. Section 4 presents the proposed C -mixture based privacy genetic algorithm for collaborative data publishing. Section 5 discusses the experimentation and outcome of the proposed method. Finally, conclusion is given in section 6.

2. Literature Review

Table 1 presents the review of recent work available for privacy enabled data publishing using different methods like, k -anonymization, l -diversity and so on.

Existing Challenges: Data publishing requires preservation of privacy to protect the sensitive information hidden in the database. When doing privacy preservation within the database contributed by n -different contributors, three

Authors	Contribution	Advantages	Disadvantages
Slawomir Goryczka et al. [1]	m -privacy and a data provider-aware anonymization algorithm	Horizontally partitioned data are anonymized at multiple data providers.	When data are distributed in a vertical or ad-hoc manner, it is difficult to handle.
Mohamed R. Fouad et al. [2]	A personalized anonymization technique based on an aggregate formulation	Effectiveness of data disclosure is maintained while keeping its risk below an acceptable threshold.	It is susceptible to bound on the estimated utility.
Slawomir Goryczka et al. [3]	Secure Distributed Data Anonymization and Integration with m -Privacy	Privacy constraint against any group of up to 'm' colluding data is satisfied by the anonymized data.	Managing set-value data is difficult.
Md. Enamul Kabir et al. [4]	systematic clustering problem for k -anonymization	Usability for incremental datasets.	Suitability of l -diversity using systematic clustering algorithm is a problem.
Tiancheng Li et al. [5]	Slicing model for privacy in data publishing	Slicing conserves superior data utility than generalization and can be used for membership disclosure protection.	It considers randomly generated links between column values of a bucket
Rashid Hussain Khokhar et al. [6]	analytical cost model to measure trade-off between privacy and utility	Used for perturbative and non-perturbative anonymization techniques.	Miss the trade-off between privacy protection and information utility.
Xiaoxun Sun et al. [7]	distinct (l, a) -diversity	Improve the present privacy standards to make them maintain the better trade-off between data quality and privacy.	Not fit for multiple sensitive attributes.

Table 1. Literature review

important considerations should be handled based on the attributes. All the quasi-identifiers should have at-least k -duplicate records of every groups. All the sensitive attributes should have l -diverse set of values in every group of data. Also, for dealing with collaborative data publishing, one important attack proposed in [1], insider attack which explains about the way of obtaining the sensitive information by colluding with the different data providers need to be handled. These three challenges should be handled before collaborative data publishing.

Cold Start Attack: The additional challenge considered here is that if one of the data providers provides large number of records but other one share less number of data records with more distribution among every groups. For this case, data provider who shares more number of data can easily track out the sensitive information of a data provider who shares less number of records. This new attack, we named as, cold start attack which should be also taken into consideration to provide more privacy before data publishing.

3. C-mixture: A Practical Privacy Definition

This section discusses the c-mixture principle about how to instantiate it with specific definitions of privacy and how to handle with sensitive attributes and number of data providers. Along with, the definitions associated with the c -mixture and example of the proposed constraints is also discussed.

C-mixture: A new privacy measure called c-mixture which includes three different privacy measures. Let $T(P, B_i, B_2, B_o, Q_1, Q_q, E_1, E_2, E_s)$ be a table and Q_i be a quasi identifier and E_i be sensitive attribute and is index of data providers

associated with it. T is said to satisfy c -mixture then, i) (1) QIs should have at-least ' c '% of duplicate records in every groups, (2) ' c '% of well defined values in sensitive attributes of every groups, (3) every group should have ' c '% of data providers.

Relative Strength: Relative strength is a parameter newly devised here for bringing the privacy constraint from the user input (c) to k -anonymity, l -diversity and m -privacy. For example, suppose, a user want to c % maintain of the mixture constraint in the table T , then k , l and m value will be directly found out from the c based on the following equation which is defined as relative strength here.

$$k = \lfloor N * c \rfloor; l = \lceil r * c \rceil; m = \lceil D * c \rceil \quad (1)$$

c is user input ranging from 0 to 1.

C-mixture Principle: The principle of c -mixture property is explained with a running example given below. Let assume the table 2 is an input table where, zip code, gender, age, education and disease are quasi identifiers. Disease is sensitive attribute and Provider is provider's name of the data records. Table 3 is known to be a c -mixture table for $c = 0.6$. If an input of c is 0.6, k value is computed by taking floor function after multiplying number of record (N) and c value. Here, number of record is eight and c is 0.6. So, the value of k is four. Now, if we check the 4-anonymity of table 3, the condition is satisfied for all the quasi identifiers which have four numbers of duplicate records for every unique attributes. To find the value of l , ceiling function is taken after multiplying of number of classes in sensitive attributes and c . The number of classes in sensitive attribute is two and the multiplication provides the value of 1.2. The final l -value after ceiling function is two. If we examine the sensitive attribute, we found that it have the 2-well represented sensitive values in every groups. Now, m value is found out by taking ceiling function after multiplying number of data providers with c value. Here, number of data providers is two and input c value is 0.6 and the final m -value is two. If we check the m -privacy constraint in table 3, every group has m number of data providers. All the three constraints based on the relative strength formula are satisfied for the c value of 0.6 so we can say that the table 3 is 0.6 mixture data.

Provider	Zip	Gender	Age	Education	Disease	Expense
P1	4351	M	25	8 th	HIV	2000
P1	4353	M	28	4 th	HIV	3000
P1	4362	F	32	8 th	HIV	2500
P1	4362	F	33	8 th	HIV	6000
P2	4354	M	26	8 th	Diabetes	3500
P2	4353	M	28	5 th	Diabetes	4000
P2	4362	M	34	4 th	Diabetes	5000
P2	4361	M	37	6 th	Diabetes	1500

Table 2. Sample input data

4. Proposed C-mixture based Privacy Genetic Algorithm for Collaborative Data Publishing

This section presents the proposed C-mixture based privacy genetic algorithm for collaborative data publishing. Here, a new algorithm called, CPGEN (C-mixture based privacy genetic algorithm) is developed by combining the genetic algorithm which is one of the optimization algorithms with c-mixture theory which is newly developed here for privacy measurements. This proposed algorithm utilizes the generalization concept for anonymization purpose by doing exhaustive search. Figure 1 shows the block diagram of the proposed collaborative data publishing.

4.1 C-mixture based Privacy Genetic Algorithm

Let assume that a trusted third party (TTP) receive a data from the multiple data providers P_i , each contributing a subset of records T_i . Each data record coming from the data provider contains provider name, set of quasi identifier,

Provider	Zip	Gender	Age	Education	Disease	Expense
P1	435*	Gender	[20-30]	Primary	HIV	2000
P1	435*	Gender	[20-30]	Primary	HIV	3000
P1	435*	Gender	[20-30]	Primary	Diabetes	3500
P1	435*	Gender	[20-30]	Primary	Diabetes	4000
P2	436*	Gender	[30-40]	Primary	Diabetes	5000
P2	436*	Gender	[30-40]	Primary	Diabetes	1500
P2	436*	Gender	[30-40]	Primary	HIV	2500
P2	436*	Gender	[30-40]	Primary	HIV	6000

Table 3. 0.6-mixture input data

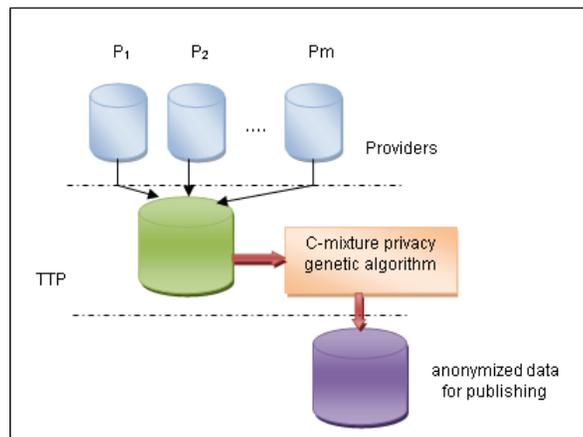


Figure 1. Block diagram of the proposed collaborative data publishing

sensitive attributes and other attributes.

$$T = \{ T_i \in P_i; 1 \leq i \leq D \} \quad (2)$$

$$T = \{ P, B_i, B_2, B_o, Q_i, Q_2, Q_q, E_i, E_2, E_s \} \quad (3)$$

Where, P is provider's name, B is common attributes, Q is quasi identifiers and E is sensitive attributes. The criteria to be fulfilled is that the TTP should publish the data T which can be viewable by any service provider so the data to be anonymous T^* in a better way to avoid the inference of founding information from the anonymous data. The final goal is to make a database T^* from the data T by considering all the attacks and should guarantee there is no disclosure of intermediate information during the anonymization.

a) Anonymization using Solution Coding

The first step of the c -mixture privacy genetic algorithm is how to encode the process of anonymization into a single vector to do the exhaustive search. The solution can be indicated as, S which contains the q number of elements. q is the number of quasi identifier. Every element in solution S may vary between the 1 to L . L is the number of levels of the quasi attributes based on the taxonomy tree. For example, the taxonomy tree of the example given in table 2 is shown in figure 2. Here, every quasi attributes are generalized with the parent values based on the L level of generalization. The solution coding for the taken record is given in figure 3. Here, four elements are presented as the number of quasi attributes is four for the example. Every element may range between 1 to L . For example, if you consider age attribute,

level of the age in taxonomy tree is three. So, the values to be placed in the solution may vary between 1 to L.

The solution encoding is then utilized to do anonymization for data publishing. Every levels indicated in the solution S is used to convert the original data T into anonymized data T^* . For example, if the zip code pointing solution element is indicated as 2 (as per figure 3), so, we can convert the zip code values presented in the table T to the second level of codes presented in the taxonomy tree. This means that the 4351, 4353 and 4354 can be converted to 435* and 4361 and 4362 can be converted to 436*. Similarity, gender pointing solution element is encoded as one. This means that ‘M’ and ‘F’ can be converted to gender as the first level of information for gender is gender. This mechanism is applied for all the quasi identifier according to the solution encoding procedure. If the solution representation given in figure 3 is applied to the original data given in table 2, then the anonymized table T^* can be as like the table given in table 3.

b) Multi-objective Optimization Formulation for Privacy Fitness Score

The objective evaluation of every solution is performed using the proposed multi-objective criteria which are newly proposed here based on generalized information loss (*GenLoss*) and average equivalence class size metric (C_{AVG}) [18]. The proposed multi-objective optimization framework considered three constrains such as, k -anonymity, l -diversity and m -privacy. These three constrains should be satisfied by every solution. Even if the solution satisfied these constraints, the objective is to minimize the utility and maximize the privacy. The three constraints can provide the privacy after anonymization but, utility can be preserved through *GenLoss* which gives the minimum value if the original table is not generalized. This means that the utility should be high for the lower values of *GenLoss*. C_{AVG} is a metric used to measure the privacy through equivalence class. The lower values of the C_{AVG} is better for preserving the privacy. So, these two tradeoffs are effectively integrated into a single function to maintain the privacy and utility along with three privacy constraints. The proposed fitness function is given as follows:

$$F(S) = \alpha * GenLoss(S) + \beta * C_{AVG}(S) \quad (4)$$

subject to the following constraints:

- i) $k \geq f_{ano}(A, T)$
- ii) $l \geq f_{div}(A, T)$
- iii) $m \geq f_{pri}(A, T)$

Where,

$$GenLoss(s) = \frac{1}{N * q} * \sum_{i=1}^q \sum_{j=1}^N \frac{P_{ij} - L_{ij}}{P_i - L_i} \quad (5)$$

$$C_{AVG}(s) = \frac{N}{|EQ_s| * k} \quad (6)$$

Here, P_i and L_i are lower and upper bounds of an i^{th} quasi identifier. P_{ij} and L_{ij} is the upper and lower bound of the generalized interval. $f_{ano}(A, T)$ is a function to compute the number of duplicate records presented in table T for every sequence value after applying anonymization A . $f_{pri}(A, T)$ is a function to compute the number of data providers for every groups in the provider after applying anonymization A . $f_{div}(A, T)$ is a function to compute the number of well represented sensitive values in every group of sensitive attributes.

c) Genetic Algorithm

Genetic algorithm (GA) [17] is one of the traditional and popular search algorithm widely applied for optimization problems. GA is developed by taking the genetic process of natural selection, inheritance, crossover, mutation and evolution. This work aim to utilize the GA for privacy-enabled data publishing as it requires heuristic search to find the optimal database to publish to third party without violating the privacy and utility. The adapted genetic algorithm for the privacy data publishing consists of following steps:

Population: The first step of the GA is initialization of population which has a n number of solutions. Every solution is represented as like the procedure discussed above. Solution is otherwise, called as chromosomes which, are

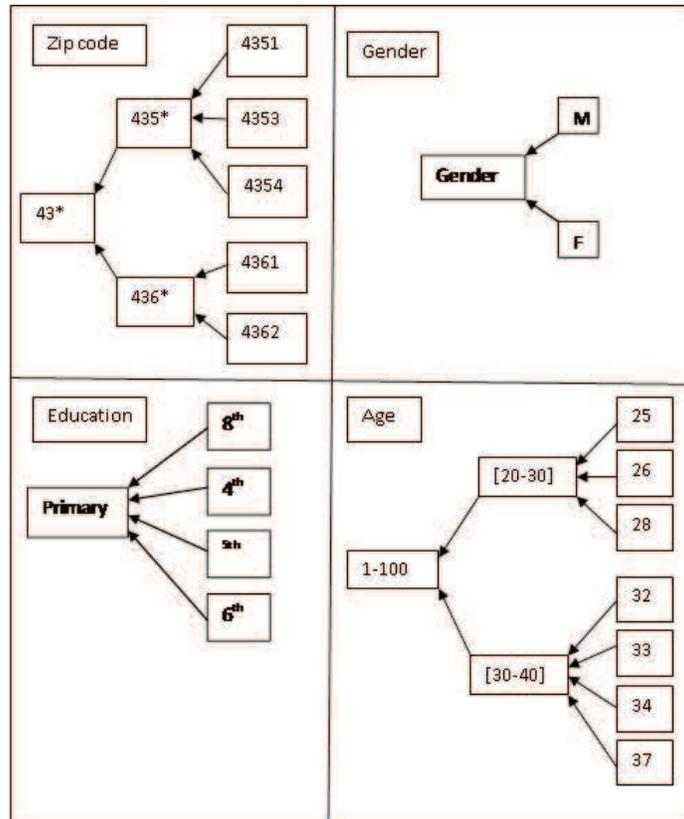


Figure 2. Taxonomy tree

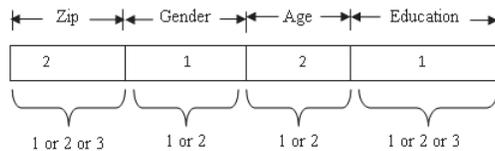


Figure 3. Solution coding

vector representations of solutions to a particular problem. Population can be indicated as follows.

$$I = \{ S_i \ ; \ i \leq i \leq n \} \quad (7)$$

Where, I is population, S is solution or chromosomes, n number of chromosomes in the population.

Once the population is initialized, privacy fitness is computed as per the privacy fitness score developed newly in this work. Once the privacy fitness is computed for all the chromosomes, it is then undergone selection process.

Selection: Selection is an important step to bring new type of solution into the population according to the natural evolution process. From the population, two chromosomes are to be selected for the evolution process. The selection of two chromosomes is purely based on the probability criteria of fitness score. The fitness score computed for all the chromosomes are then utilized to find the probability of selection based on the following formulae.

$$P_i = \frac{F(S_i)}{\sum_{i=1}^n F_i(S_i)} \quad (8)$$

Where, $F(S_i)$ is the fitness score of the i^{th} chromosomes. This means that a probability of chromosome being selected is proportional to its relative fitness.

Crossover: The two chromosomes selected from the previous step are used here to do cross over operation which produces two child chromosomes. Here, one point cross over operation is utilized. According to, a random number γ is generated in the range of 1 to q which is the length of the chromosome. Random number, γ points the location in the selected chromosomes and the solutions are interchanged accordingly. The parent two chromosomes are interchanged their values based on the random number to obtain new child chromosomes.

Mutation: The two new child chromosomes obtained from the previous steps is then given to mutation operators

1	Input T, c
2	Output S_b
3	Start
4	Initialize population I randomly
5	Perform anonymization of every chromosomes
6	Find privacy fitness of every chromosomes, $F(S)$
7	$iteration=0;$
8	While ($iteration < t$)
9	Select two chromosomes based on probability p_i
10	Apply cross over operator
11	Apply mutation operator
12	Perform anonymization of new chromosomes
13	Find the privacy fitness of new chromosomes
14	Update population I
15	Store the best chromosome S_b
16	$t = t + 1$
17	EndWhile
18	Return S_b
19	End

Figure 4. Pseudo code of CPGEN algorithm

which act on an every child chromosomes to flip one or more allele values. In order to accomplish this task, random number γ is again generated within the range of 1 to q and the solution value which points based on γ is flipped to some other values which should be in the range of level of taxonomy. Now again, two new child chromosomes are generated.

Fitness Assignment and Updating Population: For the four child chromosomes generated newly from the above

	<i>GenLoss</i>		<i>C_{AVG}</i>	
	$\alpha = 0.4, \beta = 0.6$	$\alpha = 0.6, \beta = 0.4$	$\alpha = 0.4, \beta = 0.6$	$\alpha = 0.6, \beta = 0.4$
C = 0.2	0.4529	0.4529	1.25	0.3125
C = 0.25	0.4529	0.4529	1.002	0.2505
C = 0.3	0.4529	0.4529	0.98	0.2104
C = 0.35	0.6734	0.6734	0.625	0.2104

Table 4. Analysis of weightage constants

	<i>GenLoss</i>		<i>C_{AVG}</i>	
	<i>n</i> = 4	<i>n</i> = 8	<i>n</i> = 4	<i>n</i> = 8
C = 0.2	0.4529	0.4529	1.25	1.25
C = 0.25	0.4529	0.4529	1.002	1.002
C = 0.3	0.4529	0.4529	0.98	0.98
C = 0.35	0.6804	0.6804	0.625	0.625

Table 5. Analysis of number of population

steps is then utilized to find the privacy fitness score of the solution. Once the fitness is found out for the four chromosomes, the best chromosome having minimum fitness is replaced with the worst chromosome in the population and the process is continued.

Stopping Criteria: The above steps are executed for the number of iterations, t which is the user input for terminating the algorithm. Once t iterations are reached, the best chromosome, S_b is taken out and the anonymization is performed based on the solution representation. Figure 4 shows the pseudo code of CPGEN algorithm.

4.2 Adapting CPGEN to Cold-start Problem for Data Publishing

This step aims to adapt the CPGEN to handle the cold start problem to be considered in data publishing. In real time scenario, the data records of every data provider may not be same. The distribution characteristics of data records have great fluctuation because one provider gets the updates of data frequently but, other one seem less updates. In this scenario, the maintenance of m -privacy seems very challenging. If m -privacy is not maintained in the published data records, the data provider which has more data distribution can easily tracked out the information belonging to other data records (having less data distribution) since it is very less in frequency among the data groups. Also, maintaining m -privacy is also not possible. On the other hand, if two data providers colludes each other, the data property of other providers can be easily tracked if the data distribution is not uniform among the data providers. So, in order to handle colluding attack and cold start attack, we randomly add the noisy data into the original database with the name of the provider having minimum distribution. This process of adding noisy records can easily overcome the colluding and cold start attack.

5. Results and Discussion

This section presents the experimentation and the quantitative results of the proposed CPGEN algorithm. The performance and comparative analysis is also performed with the existing algorithm [1].

5.1 Experimental Set Up

The proposed CPGEN algorithm is implemented using Java 1.7 with netbeans IDE 7.3. The experimentation is conducted on Windows 8.1 machines with Intel Core i5 processors and 4 GB of main memory.

	<i>GenLoss</i>	C_{AVG}
C = 0.001	0.4529	104.45
C = 0.002	0.4529	101.76
C = 0.003	0.4529	96.15
C = 0.004	0.4529	76.49

Table 6. Low values of c

	<i>GenLoss</i>		C_{AVG}	
	Proposed	Existing	Proposed	Existing
C = 0.2	0.4529	0.6804	1.25	5
C = 0.25	0.4529	0.6804	1.002	4.008
C = 0.3	0.4529	0.6804	0.98	3.3374
C = 0.35	0.6734	0.6804	0.625	2.5

Table 7. Effectiveness analysis

Dataset Description: Adult dataset is otherwise called as, “Census Income” dataset [26]. This data was extracted from the census bureau database. This database consists of 48842 instances and 14 attributes including both categorical and integer attributes namely, age, workclass, fnlwtg, education, education-num, marital-status, occupation, relationship, race, sex, capital-gain, capital-loss, hours-per-week, native-country.

Evaluation Metrics: The performance of the proposed algorithm through utility is evaluated through generalized information loss and average equivalence class size metric [18] is used to ensure the privacy of the proposed method.

5.2 Performance Analysis

The performance of the proposed algorithm is analyzed with the help of generalized information loss and average equivalence class size metric. Table 4 shows the performance analysis for various values of alpha and beta. The better performance is to obtain the minimum value for both the metrics to minimize the information loss and maximize the gain. For two different set of alpha and beta values, *GenLoss* is same by achieving the value of 0.4529 and 0.6734 for c value of 0.2 and 0.35 when alpha and beta is fixed as 0.4, and 0.6. But, in the case of C_{AVG} , the better value of 0.2104 is achieved when $\alpha=0.6$ and $\beta=0.4$.

Table 5 shows the analysis of number of population versus c value. For n value of four, *GenLoss* and C_{AVG} behave similarly. This shows that the number of chromosomes does not affect the performance of the proposed algorithm. When analyzing the table 5, the better performance is achieved when c is set to 0.2 by reaching the value of 0.4529 and 1.25 for *GenLoss* and C_{AVG} respectively. From the above analysis, we set $\alpha=0.6$, $\beta=0.4$ and n value as four and the experimentation is done with the lower values of c. table 6 shows the values reached by the proposed algorithm for lower values of c. Here, *GenLoss* is almost constant but the C_{AVG} value is changing frequently. For C=0.001, the proposed algorithm obtained the value of 104.45 as C_{AVG} .

5.3 Comparative Analysis

The comparative analysis is also performed with the existing algorithm [1] and the values are plotted in table 7. From table 7, we understand that the proposed algorithm obtained the minimum value of 0.4529 for *GenLoss* when c value is from 0.2 to 0.3. When the value of c is fixed as 0.35, the proposed algorithm obtained the value of 0.6734 as *GenLoss* but the existing algorithm reached the higher value of 0.6734. When analyzing the performance of both algorithms through C_{AVG} , the proposed algorithm obtained the value of 1.25 and existing algorithm achieved the value of 5 when c is fixed as 0.2. So, the proposed algorithm obtained the better performance for all the values of c parameters.

6. Conclusion

This paper presented a new privacy measure, called c-mixture for collaborative data publishing problem. Here, we considered a new kind of “cold start problem” which occur when the data distribution is not same for multiple data providers. In order to handle this new type of problem in collaborative data publishing, a new measure called, c-mixture is proposed. This new measure considered the multiple privacy constraints into a single formula by considering the relative strength. Then, an algorithm called, CPGEN is developed using genetic algorithm and multi-objective constraints. The multi-objective optimization function considered the multiple privacy constraints along with the utility measurement to ensure high utility and privacy. The proposed algorithm is extensively analyzed using generalized information loss and average equivalence class size metric and the performance is compared with existing algorithm to prove the better or comparable utility and privacy than previous algorithms. The proposed can be further enhanced to reduce the constraints defined in the objective function without affecting the privacy and utility.

References

- [1] Goryczka, Slawomir., Xiong, Li., Fung, Benjamin C. M (2014). m-Privacy for Collaborative Data Publishing, *IEEE transactions on knowledge and data engineering*, 26 (10) 1-14.
- [2] Fouad, Mohamed R., Elbassioni, Khaled., Bertino, Elisa (2014). A Supermodularity-Based Differential Privacy Preserving Algorithm for Data Anonymization, *IEEE transactions on knowledge and data engineering*, 26 (7) 1591-1601.
- [3] Goryczka, Slawomir., Xiong, Li., Fung, Benjamin C. M. (2013). Secure Distributed Data Anonymization and Integration with m-Privacy.
- [4] Kabir, Md. Enamul., Wang, Hua., Bertino, Elisa (2011). Efficient systematic clustering method for k-anonymization, *Acta Informatica*, 48 (1) 51-66, February 2011.
- [5] Li, Tiancheng., Li, Ninghui., Zhang., Jian Molloy, Ian (2012). Slicing: A New Approach for Privacy Preserving Data Publishing, *IEEE transactions on knowledge and data engineering*, 24 (3) 561- 574, March.
- [6] Khokhar, Rashid Hussain., Chen, Rui., Benjamin, C.M. Fung c, Lui, Siu Man (2014). Quantifying the costs and benefits of privacy-preserving health data publishing, *Journal of Biomedical Informatics*, 50, p. 107-121, August.
- [7] XSun, Xiaoxun., Li, Min., Wang, Hua (2011). A family of enhanced (L,a)-diversity models for privacy preserving data publishing, *Future Generation Computer Systems*, 27 (3) 348-356, March 2011.
- [8] Byun, J.W., Kamra, A., Bertino, E., Li, N. (2007). Efficient k-anonymization using clustering techniques. *In: International Conference on Database Systems for Advanced Applications (DASFAA) (2007)*.
- [9] Samarati, P., Sweeney, L. (1998). Data to provide anonymity when disclosing information. *In: Proceedings of the ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS)*, p.188.
- [10] Li,T., Li, N. (2007). t-closeness: Privacy beyond k-anonymity and l-diversity. *In: Proceedings of the IEEE International Conference on Data Engineering (ICDE)*, p.106-115.
- [11] Machanavajjhala, A., Gehrke, J., Kifer, D., Venkatasubramaniam, M. (2006). l-diversity:Privacy beyond k-anonymity. *In: Proceedings of the IEEE International Conference on Data Engineering (ICDE)*, p. 24.
- [12] Li, T., Li, N. (2008). Injector: mining background knowledge for data anonymization, *In: Proc. International Conference on Data Engineering, ICDE*.
- [13] Byun, J.W., Sohn, Y., Bertino, E., Li, N. (2006). Secure anonymization for incremental datasets, *Secure Data Management, SDM*, p. 4863.
- [14] Xiao, X., Tao, Y. (2007). m-invariance: towards privacy preserving republication of dynamic datasets, *In: Proceedings of the ACM SIGMOD Int'l Conf. on Management of Data, SIGMOD*, p. 689-700.
- [15] Fu, Ada Wai-Chee., Wang, Ke., Wong, Raymond Chi-Wing ., Wang, Jia., Jiang, Minhao (2014). Small sum privacy and large sum utility in data publishing, *Journal of Biomedical Informatics*, 50, p. 20-31.
- [16] Machanavajjhala, Ashwin., Gehrke, Johannes., Kifer, Daniel., Venkatasubramaniam, Muthuramakrishnam. (2007). l-Diversity: Privacy Beyond k-Anonymity, *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1 (1),

March 2007.

- [17] McCall, John (2005). Genetic algorithms for modelling and optimisation, *Journal of Computational and Applied Mathematics*, 184 (1), 205-222, December 2005.
- [18] Ayala-Rivera., McDonagh, Patrick., Cerqueus, Thomas., Murphy, Liam (2014). A Systematic Comparison and Evaluation of k-Anonymization Algorithms for Practitioners, *Transactions on Data Privacy*, 7 (3), 337-370.
- [19] Goryczka, S., Xiong, L., Fung, B. C. M. (2011). m-Privacy for collaborative data publishing, *In: Proc. of the 7th Intl. Conf. on Collaborative Computing: Networking, Applications and Worksharing*.
- [20] Chaytor, R., Wang, K. (2010). Small domain randomization: same privacy, more utility, *In: VLDB*, p. 608-618.
- [21] Soria-Comas J, Domingo-Ferrer J, Sanchez D, Martinez S. (2013). Improving the utility of differentially private data releases via k-anonymity, *In: 12th IEEE international conference on trust, security and privacy in computing and communications*, p. 371-379
- [22] Clifton, C., Tassa, T. (2013). On syntactic anonymity and differential privacy, *Trans Data Privacy*, 6 (2) 161-183.
- [23] Wang, K., Xu, Y., Fu, A. W. C., Wong, R. C. W. (2009). ff-anonymity: When quasi-identifiers are missing, *In: Proceedings of the 25th IEEE International Conference on Data Engineering (ICDE)*.
- [24] Ghasemzadeh M, Fung BCM, Chen R, Awasthi A. (2014). Anonymizing trajectory data for passenger flow analysis, *Transp Res Part C: Emerging Technol (TRC)*, 39, p. 63-79.
- [25] Fung, BCM., Wang, K., Yu, PS. (2007). Anonymizing classification data for privacy preservation, *IEEE Trans Knowledge Data Eng (TKDE)*, 19 (5) 711-725.
- [26] Adult Data Set from <https://archive.ics.uci.edu/ml/datasets/Adult>.