Journal of Theoretical and Applied Information Technology
15th August 2017. Vol.95. No.15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

PROTECT SENSITIVE KNOWLEDGE IN DATA MINING CLUSTERING ALGORITHM

[1]ALAA KHALIL JUMAA, [2]AYSAR A. ABUDALRAHMAN,

[3]REBWAR RASHID AZIZ, [4]ABDUSALAM ABDULLA SHALTOOKI

[1]Lecturer, Sulaimani Polytechnic University, Technical College of Informatics, Iraq

[2]Lecturer. University of Sulaimani, College of Science, Iraq

[3] MSc Student. Sulaimani Polytechnic University, Computer Science Institute, Iraq

[4]Assistant Lecturer. University of Human Development, College of Science and Technology, Iraq

E-mail: [1]Alaa.alhadithy@spu.edu.iq, [2]aysser2005@yahoo.com,
[3]rebwar_cs@hotmail.com, [4]salam.abdulla@ uhd.edu.iq

ABSTRACT

Privacy preserving knowledge discovery is a new and very important topic in data mining that is perfectly talked about the privacy of data and information. This paper focuses on protecting the knowledge in the clustering data mining techniques (K-Mean Clustering). Moreover, a new algorithm is suggested for protecting sensitive clusters, which uses the Adaptive noise techniques for protecting process. In the proposed algorithm, the adaptive noise values that are used for protecting sensitive clusters are evaluated depending on the original database values. In deep, the evaluated noise values depend on the distances (Euclidian Distance) between Sensitive Cluster and the rest of the other clusters (Non-Sensitive Clusters) for the original database. The proposed algorithm use three different techniques for protecting sensitive cluster. The prototype system was used to perform the proposed algorithm. For the three different datasets that are used in a prototype system implementation, the experimental results show that the proposed algorithm is protecting Sensitive Clusters with High Privacy Ratio and Low Information Loss Ratio. Hence, the proposed system provides a good accuracy with a low ratio of side effects, and it supports high level of privacy.

Keywords: *Privacy preserving, knowledge discovery, K-Mean Clustering, sensitive clusters Euclidian Distance, Privacy Ratio.*

1. INTRODUCTION

Since the increasing use of data mining, huge volumes of detailed personal data are commonly collected and analyzed, such as: shopping obsession, criminal records, medical history, and credit records. Such data are critical asset to business organizations and governments, and they also are important to decision making processes and to afford social benefit, such as medical research, crime cutback, and national security. Beside that, analyzing such data may open new threats to privacy and autonomy of the original [1]. Thus, an interesting new direction of data mining research along with considering the above factors, is the main motivation of this work.

Privacy preserving data mining is a latest research area in the field of data mining. It is defined as "protecting user's information". Protection of privacy has become an important in data mining research because of the adding talent to store personal data about users and the development of data mining algorithms to infer this information. The main goal in privacy preserving data mining is to evolve a system for modifying the original data in some way, so that the private data and knowledge remain private even after the mining process [2].

Clustering is one of the most common tasks in data mining. It partitions the data records into groups (clusters) that are internally homogeneous and heterogeneous from class to class. In the business world, data clustering has been used

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

extensively to find optimal customer object, better profitability, and market more effectively, and maximize return on investment supporting business collaboration. Achieving privacy preservation when sharing data for clustering is a challenging problem. To address this problem, the system must not only meet privacy essential of the data owners but also agreement valid clustering results [3].

There are a number of techniques that are used in protect sensitive knowledge with clustering data mining, some of these technique are [4]:

- Additive Noise Technique: The basic idea for this technique is to add random noise to the actual data. The noise being added is typically continuous and with mean zero, which suits well continuous original data.
- Data Swapping Technique: Data swapping preserves the privacy of original impressionable information available at record level. If the records are picked at random for each swap then it is called random swaps. It is arduous for an intruder to identify particular person or entity in database, because all the records are altered to the maximum level. In order to increase the privacy result by choosing more than one value in the selected cluster (the values that are caused of founding this record in that cluster) to swap across different records in other clusters, in order to shift this record to another cluster.
- Data Copying Technique: This is a new perturbative technique that it's suggested for protecting the sensitive numerical attributes in the selected cluster. It is very similar to Data Swapping technique because it is simple and can be used only on sensitive data without disturbing non sensitive data.

The remaining of this paper is organized as follows: Section 2 present the problem description, section 3 briefly discusses some of the related works, section 4 concentrates on the K-mean clustering and Euclidean Distance techniques. Next, Section 5 discuss the proposed protecting system, then section 6 show, discuss, and analysis the excremental results, and finally section 7 presents the conclusion for this paper.

2. PROBLEM DESCRIPTION

Clustering is one of the most common tasks in data mining. It partitions the data records into groups (clusters) that are internally homogeneous and heterogeneous from class to class. In the business world, data clustering has been used extensively to find optimal customer object, better profitability, and market more effectively, and maximize return on investment supporting business collaboration. Achieving privacy preservation for clustering is a challenging problem. To address this problem, the system must not only meet privacy essential of the data owners but also agreement valid clustering results. The problem of privacy preservation in clustering can be stated [5]:

Let D be a relational database which is include number of tuples (t_j) where $t_j \in t_1, t_2 \ldots t_n$ and n = number of tuples in D, and C a set of clusters generated from D where $C_1, C_2 \ldots C_m \in C$ and m= number of generated clusters. The goal is to transform D into D` so that the following restrictions grasp [5]:

1. A transformation T when applied to D must protect the privacy of individual records, so that the released database D` conceals the values of secret attributes, such as salary, disease diagnosis, credit grade, and others.
2. The likeness between objects in D` must be the identical as that one in D, or just a little altered by the transformation process. Although the mutation database D` looks very different from D, the clusters in D and D' should be as close as possible since the distances between objects are preserved or marginally changed.

In this paper, a new algorithm for protecting sensitive cluster in data mining was presented. This algorithm depends on the Adaptive noise techniques. The evaluation processes for the adaptive noise value depends on the Euclidian Distance between Sensitive cluster and other Remain Clusters.

3. RELATED WORKS

There are a large number of researches that are focused on privacy preserved in clustering data miming. In [2010] C. Ming et al, are proposed a method of Privacy-Preserving Clustering of Data Streams (PPCDS) to improve data stream mining procedures while together preserving privacy with a high degree of mining accuracy. PPCDS is mainly calm of two phases: Rotation-Based Perturbation and cluster mining. This method used two processes for protect privacy and reduces the repeated calculation time in order to build up mining efficiency without losing mining accuracy [6]. In [2011] B. Karthikeyan et al, proposed a new way to preserve sensitive information using fuzzy logic. First they performed clustering on the original data set then they add noise to the numeric data using a fuzzy participation function that

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

conclusion in distorted data. The main benefit of this method is that it maintains the privacy and at the same time preserves the relativity between the data values [7]. In [2013] V. Rajalakshmi et al, have analyzed the use of normalization techniques like Min-Max normalization, Z-Score normalization and Decimal Scaling methods with esteem to privacy and accuracy. K-means Clustering algorithm is applied to the original and the tailored data to verify the effectiveness and the correctness of the planned approach. Min-max normalization performs a linear modify on the original data. The values are normalized within the given scope. The advantages of Min-Max normalization is that all the values are annealed within certain scope [8]. In [2013] M. Naga et al, proposed a hybrid method for privacy preserving clustering in centralized database habitat. In this method, the dataset is perturbed using Non adverse Matrix Factorization (NMF) and Principle Component Analysis (PCA) methods to protect the impressionable attribute values. This method preserves privacy of original and maintains data utility for clustering analysis [9]. In [2014] H. Adnan et al, proposed a new technique (PAM Clustering) for protecting the sensitive cluster that is done by using privacy techniques through of modifying the data values (attributes) in the dataset. They used Additive Noise, Data Swapping and Data copying techniques to prevent attacker from concluding user's privacy information in the sensitive cluster. Experimental results shows that the proposed techniques is efficient for clustering in all data sets and the sensitive cluster are protected efficiently [10]. In [2016] Z. Gheid et al, proposed a novel privacy-preserving k-means algorithm based on a simple yet secure and efficient multi-party additive scheme that is cryptography-free. Through different evaluations, it proved the security for the proposed techniques as well as their simplicity and efficiency compared to other propositions. These preliminary results demonstrate that the author's solution suites better to big data properties and scales to large data sets where that is demonstrate that performance (computation overheads) of the solution is independent from data entity sizes [11].

4. K-MEAN CLUSTERING & EUCLIDEAN DISTANCE

The k-means algorithm is a clear iterative clustering algorithm that partitions a given dataset into a user-specified number of clusters, k. The algorithm is clear to implement and run, relatively fast, easy to adapt, and average in practice. It is historically one of the most important algorithms in data mining. The k-means algorithm applies to objects that are depicted by points in a d-dimensional vector space. Thus, it clusters a set of d-dimensional vectors, $D = \{X_i \mid i = 1 \ldots N\}$, *where* $X_i \in d$ designate the i^{th}object or "data point". The k-means is a clustering approach that partitions D into k clusters of points. That is, the k-means approach clusters all of the data points in D such that each point X_i descent in one and only one of the k partitions. One can keep track of which point is in which cluster by attaching each point a cluster ID. Points with the identical cluster ID are in the identical cluster, while points with unlike cluster IDs are in unlike clusters. The value of k is an input to the base algorithm. Commonly, the value for k is based on criteria such as prior knowledge of how many clusters absolutely appear in D, how many clusters are wanted for the current application, or the types of clusters found by exploring/experimenting with unlike values of k [12].

In k-means, each of the k clusters is represented by an alone point in. Let it denote this set of cluster delegate as the set $C = \{C_j \mid j = 1 \ldots k\}$. These k cluster representatives are also called the cluster means or cluster centroids. In clustering algorithms, points are grouped by some belief of "closeness" or "similarity." In k-means, the default measure of proximity is the Euclidean distance. In other words, k-means attempts to minimize the total squared Euclidean distance between each point X_i and its closest cluster representative C_j. The K-mean algorithm works as follows. First, the cluster representatives are initialized by picking k points in d. Techniques for selecting these initial seeds include sampling at haphazard from the dataset, setting them as the solution of clustering a slight subset of the data, or perturbing the universal mean of the data k times. The algorithm then iterates between two steps until convergence [12]:

Step 1: Data assignment. Each data point is assigned to its nearby centroid with ties broken arbitrarily. This results in a division of the data.

Step 2: Relocation of "means." Each cluster representative is relocated to the center (i.e., arithmetic mean) of all data points assigned to it. The rationale of this step is based on the attention that, given a set of points, the single best representative for this set (in the sense of minimizing the sum of the squared Euclidean distances middle each point and the representative) is nothing but the mean of the data points.

Journal of Theoretical and Applied Information Technology
15th August 2017. Vol.95. No.15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The Euclidean distance is a standard metric for geometrical problems. It is the ordinary distance between two points and can be simply measured with a ruler in two- or three-dimensional space. Euclidean distance is broadly used in clustering problems, including clustering text. It is also the default distance measure used with the K-means algorithm. Measuring distance between text documents d_a and d_b depicted by their term vectors $\vec{t_a}$ and $\vec{t_b}$ respectively, the Euclidean distance of the two documents is defined as [13]:

$$D_E(\vec{t_a}, \vec{t_b}) = \left(\sum_{i=1}^{m} | w_{i,a} - w_{i,b}|^2\right)^{1/2}$$

Where the term set is T = $\{t_1 \ldots, t_m\}$.

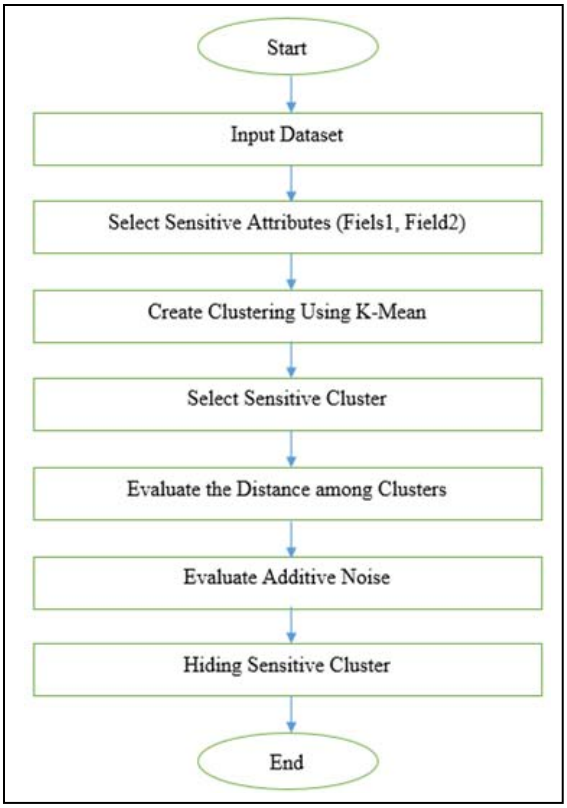

Figure 1 : General Flowchart For The Proposed System

5. THE PROPOSED PROTECTING SYSTEM

There are number of techniques that are used for protecting or hiding sensitive clusters such as: additive noise, data swapping, data copy … etc. In this paper, the proposed system, which used for hiding sensitive clusters, depends on adding adaptive noise to the original database.
The adaptive noise techniques can be applied by adding the evaluated noise to original database according to the specific purpose. In the proposed algorithm, the adaptive noise value which used for hiding sensitive cluster can be evaluated depending on the original database values. In another words, the evaluated values depend on the distances between sensitive cluster and the other clusters of the original database.

The first step to perform the proposed system needs to perform the k-mean clustering algorithm for the original database to extract the sensitive cluster. Then evaluate the Euclidian distance between sensitive cluster and the other non-sensitive clusters. According to the values of the adaptive noise, the sensitive cluster could be hid or protect by adding this noise to the original database values for the sensitive cluster. Figure (1) shows the general flowchart for the proposed system.
The proposed protected system can be performed according to the following steps:-

A. Generate Database Clusters

To generate database clusters, the Weka package is used. Weka is a collection of machine learning algorithms for data mining tasks. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is also well-suited for developing new machine learning schemes. K-means algorithm is one of the clustering algorithms which used in this work. At the beginning, all the sensitive items in the original database need to be selected. (In the proposed system we are used two items). Then, the clusters evaluated according to those two items.

B. Evaluate Distance between Clusters

The proposed system for protecting or hiding sensitive cluster depends on the distance between the sensitive cluster and the rest of the other clusters. First, the sensitive cluster needs to be chosen. Then the distance between this cluster and the other clusters are evaluated. In this proposed system, the Euclidian Distance algorithm is used to evaluate the distance among the clusters.

The following steps show how the distances between clusters are calculated:-

1. Input Sensitive Cluster (S_c) with its attributes;
   - Input S_c tuples (t_j), where $t_j \in t_1, t_2 \ldots t_n$, j = 1 to n, and n = number of tuples within sensitive cluster
   - For each tuple (t_j) there are two sensitive attributes ($S_c.A_{1j}$, $S_c.A_{2j}$), where j = 1 to n,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A_1 and A_2 represent Attribure1 and Attribute2 respectively, and n = number of tuples within sensitive cluster

- Input the Center point for Sensitive Cluster ($S_c A_{1\,(ctr)}$, $S_c A_{2\,(ctr)}$)

2. Input Remain Clusters (R_{ci}) with its attributes
   - Where $R_{ci} \in R_{c1}, R_{c2}, \ldots R_{cm}$, i = 1 to m, and m = number of remain clusters.
   - Input the Center Point of the Remain Clusters ($R_{ci}A_{1\,(ctr)}$, $R_{ci}A_{2\,(ctr)}$)
3. Evaluate Euclidian Distance between sensitive clusters (S_c) and other Remain Clusters (R_{ci}).
   - $d_i = ED\,(S_c\ , R_{ci})$ (1)

   Where d_i is the evaluated distance, i =1 to m, and m = number of Remain Clusters.

C. Evaluated Adaptive Noise

In this work, three techniques for protecting or hiding sensitive clusters are proposed. All these techniques depend on the Euclidian Distance between the sensitive cluster and other Clusters. For hiding process, it needs to evaluate the Adaptive Noise which is used in the protecting or hiding techniques. Because there are three techniques for the proposed system; so, it will need three processes for evaluating different Adaptive Noise for each technique. Figure (2) shows the general flowchart for Adaptive Noise Evaluation.
The processes of evaluating adaptive noise for all three techniques are explained in below.

• First Technique

In this technique, hiding process is achieved by moving the sensitive cluster to the nearest cluster. The nearest cluster ($R_{c\,(nst)}$) is the cluster that has a minimum Euclidian Distance with the sensitive cluster.
The process of evaluating the adaptive noise in this technique is explained by the following steps:

1. Select the Nearest Cluster ($R_{c\,(nst)}$).
2. Evaluate the Adaptive Noise for the Sensitive Attributes (A_1Noise, A_2Noise) by:
   - $A_1Noise = [(R_{c(nst)}\ A_{1(ctr)}) - (S_c A_{1(ctr)})]$ (2)
   - $A_2Noise = [(R_{c\,(nst)}\ A_{2(ctr)}) - (S_c A_{2(ctr)})]$ (3)

   Where $(R_{c(nst)}\ A_{1(ctr)})$ and $(R_{c\,(nst)}\ A_{2(ctr)})$ represent the center points of the Nearest Cluster.

• Second Technique

In this technique, hiding process is achieved by moving the Sensitive Cluster to the Farthest Cluster. The Farthest Cluster ($R_{c\,(far)}$) is the cluster that has a maximum Euclidian Distance with the sensitive cluster.

The process of evaluating the adaptive noise in this technique is explained by these following steps:

1. Select the Farthest Cluster ($R_{c\,(far)}$).
2. Evaluate the Adaptive Noise for the sensitive attributes (A_1Noise, A_2Noise) by:
   - $A_1Noise = (R_{c(far)}\ A_{1(ctr)}) - (S_c A_{1\,(ctr)})]$ (4)
   - $A_2Noise = [(R_{c(far)}\ A_{2(ctr)}) - (S_c A_{2(ctr)})]$ (5)

   Where $R_{c\,(far)}\ A_{1\,(ctr)}$ and $R_{c\,(far)}\ A_{2\,(ctr)}$ represent the center points for the Farthest Cluster.

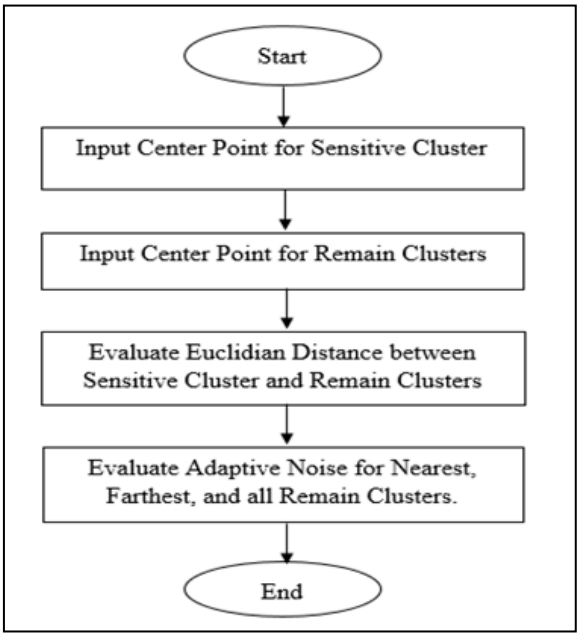

Figure 2: Flowchart For Adaptive Noise Evaluation Process

ISSN: 1992-8645 www.jatit.o

- **Third Technique**

In this technique, hiding process is achieved by distributed the Sensitive Cluster points (attribute values) to All Remaining Clusters with equal ratio for each one. So, it needs to evaluate different Adaptive Noise for each Remain Cluster. This can be done by the following steps:

1. Select all Remain Clusters.
2. For each Remain Cluster (Rc_i).

$$A_1Noise = [(R_{ci}.A_{1(ctr)}) - (S_c.A_{1(ctr)})] \quad (6)$$

$$A_2Noise = [(R_{ci}.A_{2(ctr)}) - (S_c.A_{2(ctr)})] \quad (7)$$

D. Protecting or hiding Sensitive Cluster

After calculating the additive Noise of the all protecting technique types, the process for protecting or hiding sensitive cluster is performed. All proposed hiding techniques use the same hiding process. This process achieved by adding the Adaptive Noise value to the original database value (sensitive attributes) for the Sensitive Cluster. And to avoid the outlier values that can be appears after adding the Adaptive Noise, the result values after adding process must be neither greater than maximum attribute value nor less than minimum attribute value for the original database. So, it needs to assign the maximum and minimum values for the sensitive attribute in the original database. Then, if the new value of the sensitive attribute (after adding noise) is greater than the maximum value in original database, this new value must be assigned to maximum value. Otherwise, if the new value is less than the minimum value for original database, it must be assign to the minimum value.

Figure (3) shows the flowchart of protecting or hiding sensitive cluster process, and the steps for this process are explained in the following:

1. For each tuple (t_j) in the S_c that include ($S_c.A_{1j}$) and ($S_c.A_{2j}$), Where *j = 1* to *n*.

$$(S_c.A_{1j})_{new} = (S_c.A_{1j}) + A_1Noise \quad (8)$$

$$(S_c.A_{2j})_{new} = (S_c.A_{2j}) + A_2Noise \quad (9)$$

Where $(S_c.A_{1j})_{new}$ represent a new value for the first sensitive attributes within the sensitive cluster, and $(S_c.A_{2j})_{new}$ represent a new value for the second sensitive attributes within the sensitive cluster.

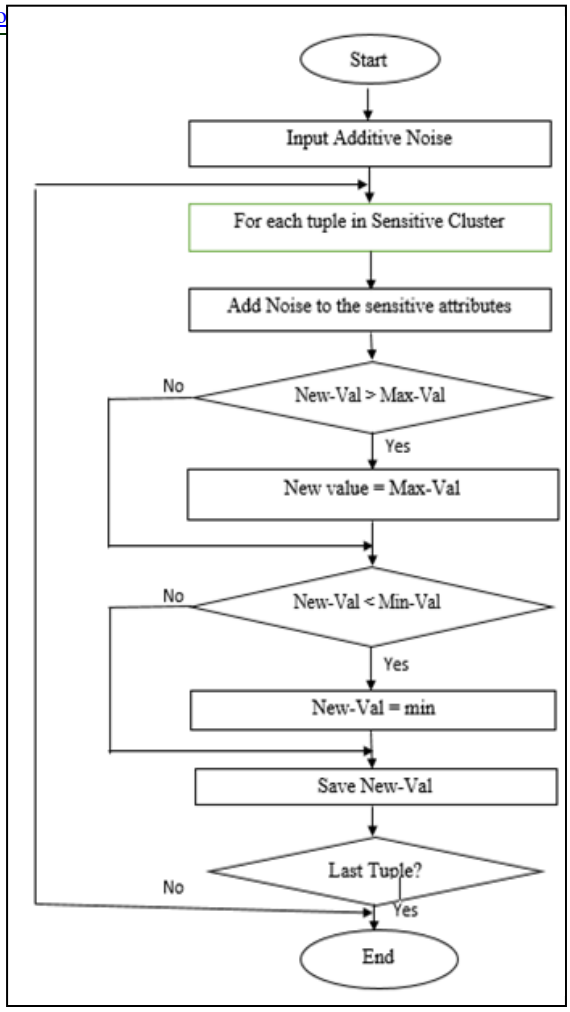

Figure 3: Flowchart Of Protecting Or Hiding Sensitive Cluster Process

2. Assign the maximum and minimum values for the sensitive attributes in original database (A_{1Max} and A_{1Min} , A_{2Max} and A_{2Min})

If $(S_c.A_{1j})_{new} > A_{1Max}$ then $(S_c.A_{1j})_{new} = A_{1Max}$

If $(S_c.A_{1j})_{new} < A_{1Min}$ then $(S_c.A_{1j})_{new} = A_{1Min}$

If $(S_c.A_{2j})_{new} > A_{2Max}$ then $(S_c.A_{2j})_{new} = A_{2Max}$

If $(S_c.A_{2j})_{new} < A_{2Min}$ then $(S_c.A_{2j})_{new} = A_{2Min}$

Where A_{1Max} = Maximum value for the first sensitive attribute in original DB,

A_{1Min} = Minimum value for the first sensitive attribute in original DB,

A_{2Max} = Maximum value for the second sensitive attribute in original DB,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A_{min2} = Minimum value for the second sensitive attribute in original DB

3. For each tuples (t_j) in the S_c, the original value for (S_c A_{1j}) and (S_c A_{2j}), need to be replaced by $(S_c\ A_{1j})_{new}$ and $(S_c\ A_{2j})_{new}$.

6. EXPERIMENTAL RESULTS AND ANALYSIS

The implementation of the proposed system done using Weka package for performing data mining techniques and Java programming language to perform protecting or hiding process and build the system interface. Three types of datasets are used for experimenting and testing the proposed system.

- **Datasets**

Three types of datasets are used for implementing and testing the proposed algorithm. The three datasets are Train, Vehicle, and Airlines datasets.
The train datasets contains 21 attributes and 700 records, and the chosen sensitive attributes are the **Age and Duration**. The vehicle datasets contains 19 attributes and 864 records, and the chosen sensitive attributes are **Radius Ratio**, **Scatter Ratio.** The Airline datasets contains 7 attributes and 539383 records, and the chosen sensitive attributes is **Time and Length.**

- **Clusters Generation**

To generate clusters from the three datasets, K-mean algorithm is used. This algorithm performed easily by using Weka package. To increase the accuracy for the experimental results in the proposed system, different numbers of clusters are generated from the different datasets. Table (1) shows the details about datasets information and Figure 4, 5 and 6 shows the output clusters for each dataset.

Table (1) Details of Dataset Information

Dataset Names	No. of Attributes.	DB Size (Tuples)	Generated Clusters	Sensitive Cluster Size (Tuples)
Train	21	700	4	195
Vehicle	19	846	4	322
Airlines	8	539	6	32963

- **Performance Evaluation Factors**

Three main factors have been considered to evaluate the performance for the proposed algorithm, which are: Privacy ratio, Information Loss Ratio, and execution Time. These three factors are explained below [9]:-

1. **Privacy Ratio(PR):** The privacy ratio measured by calculating the percentage between the number of records in the original cluster before privacy, and the numbers of records that remained near to the original cluster after privacy. The privacy ratio is calculated by Equation (10).

$$PR= \left(1 - \frac{\bar{R}(C)}{R(C)}\right) * 100\ \% \quad (10)$$

Where $\bar{R}(C)$ is the number of records that remained near to the original cluster after hiding process, $R(C)$ is the number of records in the original cluster before hiding process.

2. **Information Loss Ratio (ILR):** This performance factor is used to measure the percentage of distortion the information of all data set after applying the privacy technique. It is the percentage of the summation of the difference between original value and modified value (the sensitive attribute in each record of the selected cluster) and the summation of original values (all dataset). The Information Loss Ratio (ILR) is calculated by

$$ILR= \frac{(\sum|\ orginal\ value - new\ value\ |)}{(\sum|\ orginal\ values\ |)} * 100 \quad (11)$$

3. **Run Time (RT):** it represents the running time required by the proposed algorithm for hiding sensitive cluster. This time represents the CPU time that are needed to perform the hiding process.

- **Performance Evaluation for the Proposed Algorithm**

The experiments of the proposed hiding techniques have been implemented on a Notebook with Intel(R) Core (TM) 2 Duo CPU 7.0 GHZ processor and 8 GB memory under windows 7 Ultimate operating system.

The experimental results are analyzed based on the Privacy ratio, Information loss ratio, and Running time factors.

As it mentioned in the previous sections, the cluster hiding algorithm process is performed using three techniques: hide to nearest cluster, hide to farthest cluster, and hide to the rest of the other

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

clusters. So, the performance measurements factors need to be evaluated for all these techniques.

Moreover, because there are three types of datasets are used in this evaluation; datasets need to be discussed in details.

To analysis and discuss the experimental results for the proposed algorithm, Table 2 and Figure 4. present the Privacy Ratio of the three techniques (Nearest Cluster, Farthest Cluster, and All Remain Clusters) for the used three Datasets (Train, Vehicle, and Airlines). It can be observed that the Farthest Cluster technique has the highest Privacy Ratio when it comparing with the other two techniques; beside that, the other two techniques (Nearest and All Remain Cluster) also have a good Privacy ratio, as shown below.

Table (2) Privacy Ratio for the Three Datasets

Dataset	Nearest cluster	Farthest cluster	All Remain Clusters
Train	90%	100%	96%
Vehicle	86%	100%	95%
Airlines	84%	99%	88%

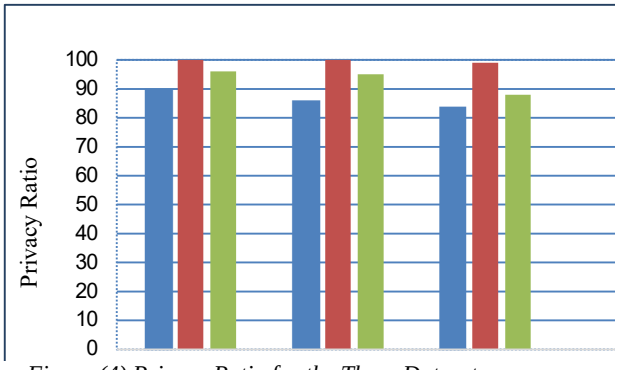

Figure (4) Privacy Ratio for the Three Datasets

Table (3) and Figure (5) present the Information Loss Ratio of the three techniques for the all three Datasets. It can be observed that the Nearest Cluster technique has the lowest value of Information Loss Ratio, and the Farthest Cluster technique has the highest ratio of Information Loss Ratio with the acceptable Information Loss Ratio in the All Remain Clusters Technique.

Table (3) Information Loss Ratio for the three Datasets

Dataset	Nearest cluster	Farthest cluster	All Remain Clusters
Train	6.58%	19.21%	6.89%
Vehicle	4.82%	13.16%	5.37%
Airlines	4.20%	6.17%	4.34%

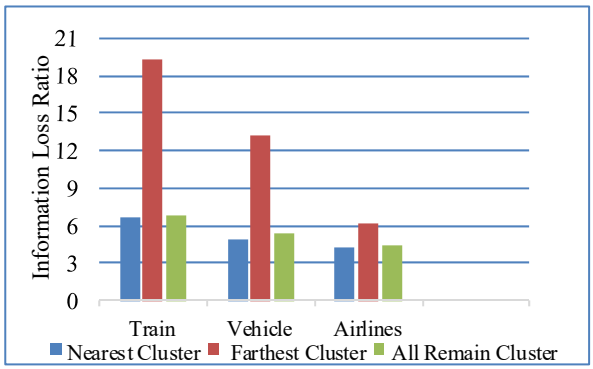

Figure 5: Information Loss Ratio for the Three Datasets

Table 4 and Figure 6 present the Run Time in milliseconds of all the three datasets with the three hiding techniques. It can be observed that the third dataset (Airplane Dataset) has the highest Running Time. The reason behind that is having a large number of records (large size database) comparing with the other Datasets. Also, it can be observed that the three hiding techniques have the same running type for each Dataset, and the reason is their trying to hide the same number of records that are exists in the sensitive cluster.

Table (4) Run Time for the Three Datasets

Dataset	Nearest cluster	Farthest cluster	All Remain Clusters
Train	20 (ms)	18 (ms)	18 (ms)
Vehicle	25 (ms)	23 (ms)	25 (ms)
Airlines	1669 (ms)	1673 (ms)	1621 (ms)

Journal of Theoretical and Applied Information Technology
15th August 2017. Vol.95. No.15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

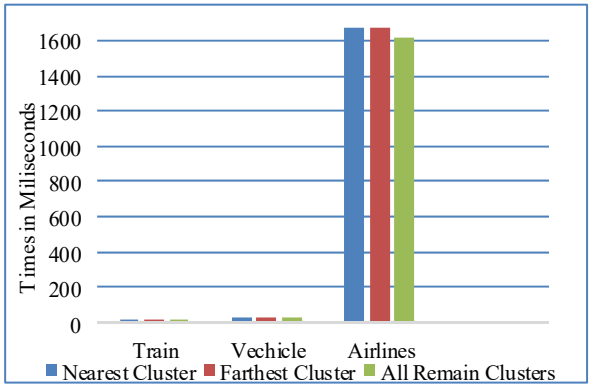

Figure 6: Run Time for the Three Datasets

Figures 7, 8, 9, and 10 show the Original Output Clusters, New Clusters after applied Nearest Cluster, New Clusters after applied Farthest Cluster, and All Remain Clusters techniques from the Dataset1.

According to the last three previous figures, it can be observed that the sensitive cluster with (Cyan Color); which appeared in the output clusters for the original database is; disappeared in the new output clusters after applied the hiding techniques. Which means the sensitive cluster is hided and not appears after applying the proposed hiding algorithm.

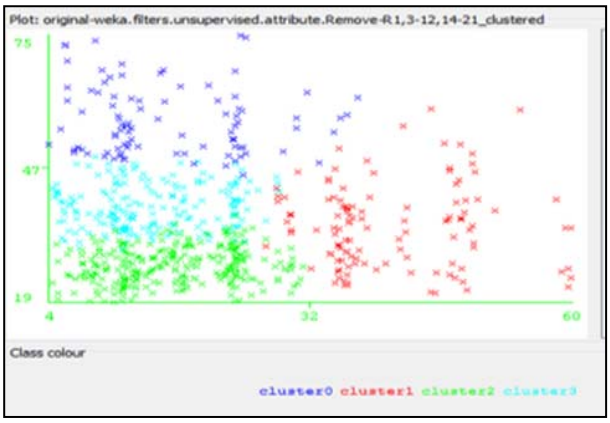

Figure 7: Original Output Clusters in Dataset1

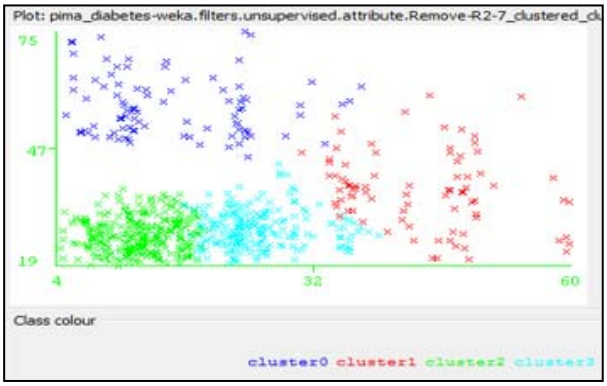

Figure (8) New Output Clusters after applied Nearest Cluster Technique in Dataset1

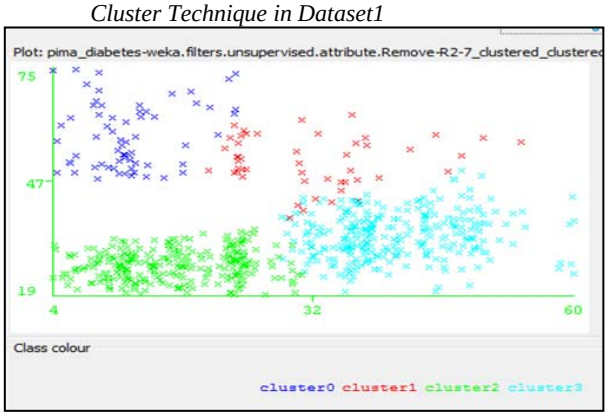

Figure 9: New Output Clusters after applied Farthest Cluster Technique in Dataset1

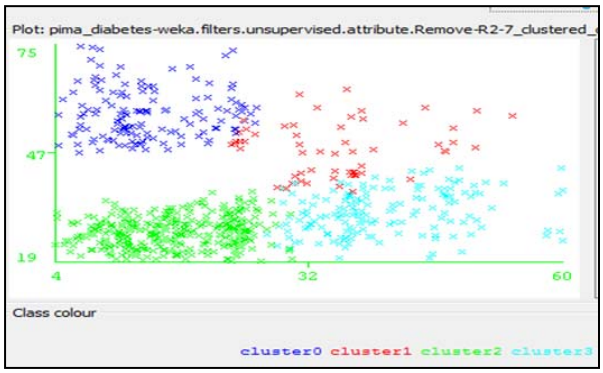

Figure 10: New Output Clusters after applied All Remain Clusters Technique in Dataset1

7. CONCLUSION

In this paper, a new algorithm for protect sensitive cluster in K-mean clustering techniques is proposed and implemented. This algorithm uses the Adaptive Noise techniques for protecting sensitive cluster. The values for the adaptive noise that are used for protecting or hiding process is evaluated

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

depending on the original database values. The prototype system that are used for performing the proposed algorithm is built using Java programming language, and Weka package, which is used for extracting K-Mean clusters before and after hiding Sensitive Cluster.

The proposed algorithms consist of three different techniques (Nearest Cluster, Farthest Cluster, and All Remain Clusters). Those techniques have well performed for three different Datasets. Three main factors have been considered to evaluate the performance for the proposed algorithm, which are: Privacy ratio, Information Loss Ratio, and execution Time. The experimental results show that the proposed hiding algorithm can protect a Sensitive Cluster with High Privacy Ratio and Low Information Loss Ratio.

One of the main merits the value of loss ratio; which calculated in this work; is independent for the size of database. On the other hand, it is still depends on the nature of the database and the Euclidean distance between the sensitive cluster and other remain clusters. So the loss ratio will be increased when the distance between the sensitive cluster and other remain cluster are increased.

. About the running time, the three hiding techniques have the same running type (less run time in Milliseconds) for each Dataset, and the reason is their trying to hide the same number of records that are exists in the sensitive cluster.

REFERENCES

[1] D. Agrawal, C. Aggarwal, "design and quantification of privacy preserving data mining algorithms", Proceedings of the 20th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, Santa Barbara, California, pp. 247-255, May 2001.

[2] K. Das, "Privacy Preserving Distributed Data Mining based on Multi-objective Optimization and Algorithmic Game Theory", PhD Thesis, University of Maryland, Baltimore County, Maryland, USA, 2009.

[3] L.Li, Q.Zhang, "A Privacy Preserving Clustering Technique Using Hybrid Data Transformation Method", IEEE, International Conference on Grey Systems and Intelligent Services, Nanjing, China. PP. (1502-1506), 2009.

[4] H. Raheem, S. Al-Mamory,"Privacy Preserving in Data Mining Using PAM Clustering Algorithm", Journal of Babylon University/Pure and Applied Sciences, Vol (22), No (9), pp (2266-2276), 2014.

[5] D. Aruna, R. Roa, M. Suman," Vector Quantization for Privacy preserving Clustering in Data", Advanced Computing: An International Journal (ACIJ) DOI: (10.5121), Vol (3), No (6), pp. (69-74), 2012.

[6] C. Ming, P. Zung, C. Hao," Privacy-Preserving Clustering of Data Streams", Tamkang Journal of Science and Engineering, Vol. (13), No.(3), pp. (349-358), 2010.

[7] B. Karthikeyan, G. Manikandan, V. Vaithiyanathan," A Fuzzy Based Approach for privacy Preserving Clustering", Journal of Theoretical and Applied Information Technology , ISSN : (1817-3195), Vol. (32) and No.(2), pp. (118-122), 2011.

[8] V. Rajalakshmi, G. Anandha," Anonymization Based on Nested Clustering for Privacy Preservation in Data", Indian Journal of Computer Science and Engineering (IJCSE), ISSN: (0976-5166), Vol. (4), No. (3), pp. (216-224), 2013.

[9] M. Naga, K. Sandhya," Privacy Preserving Clustering by Hybrid Data Transformation Approach", International Journal of Emerging Technology and Advanced Engineering, ISSN: (2250-2459), ISO: (9001:2008), Vol. (3), pp. (696-700), 2013.

[10] H. Adnan, S. Al-Mamory," Privacy Preserving in Data Mining", Journal of Karbala University, ISSN: (18130410), Vol. (12), No. (3), pp. (179-195), 2014.

[11] Z. Gheid, Y. Challal," Efficient and Privacy-Preserving k-means clustering For Big Data Mining", IEEE TristCom, Aug 2016, Tianjin, China. pp.791 – 798, 2016

[12] Prasad B, P. V. Ramana, A. Naidu," K Means Clustering Algorithm", International Journal of Computer Science and Information Technology, ISSN: (0974-8385), Vol (5), No (1) PP. (19-28), 2012.

[13] A. Singh, A. Yadav, A. Rana," K Means with Three Different Distance Metrics", International Journal of Computer Applications, (IJCA), ISSN: (0975 – 8887), Vol. (7), No. (10), 2013.