A Novel Association Rule Hiding Approach in OLAP Data Cubes

Mohammad Naderi Dehkordi
Computer Engineering Department,Najafabad branch, Islamic Azad University, Esfahan, Iran
naderi@iaun.ac.ir

Abstract

Data mining services require exact input data for their outcomes to be significant, but privacy concerns may influence users to provide fake information. We study here, with respect to mining association rules, whether or not users can be confident to provide correct information by ensuring that the mining process cannot, with any reasonable degree of certainty, breach their privacy. A data warehouse stores current and historical records consolidated from multiple transactional systems. Protecting data warehouses is of rising interest, particularly in view of areas where data are sold in pieces to third parties for data mining studies. In this case, current normal data warehouse security techniques, like data access control, may not be easy to impose and can be in effective. As an alternative, this paper proposes a data perturbation based approach, to provide privacy preserving in association rule mining on data cubes in a data warehouse. In order to conceal association rules and save the utility of transactions in data cubes, we select Genetic Algorithm to find optimum state of modification. In our approach various hiding styles are applied in different multi-objective fitness functions. To cope with the multi-objective functions, Pareto-front ranking strategy has been applied for obtaining the non-dominated solutions front. First objective of these functions is hiding sensitive rules and the second one is keeping the accuracy of transactions in data cube. After sanitization process we test the sanitization performance by evaluation of various criterions. The major feature is that the proposed strategy does not affect the functionality of the On-Line Analytical Processing system. Finally our experimental results show its effectiveness and feasibility.

Keywords: OLAP, Data Cube, Data Mining, Association Rule Hiding

1. Introduction

A *data warehouse* defined in (Chaudhuri & Dayal ,1997) is a collection of data from multiple sources, integrated into a common repository and extended by summary information (such as aggregate views). This technology has been used by organizational decision makers. Some sort of queries that typically involves group-by and aggregation operators is called an online analytical processing or OLAP. OLAP application enables data analysts and organizational managers to acquire the ability to understand the performance of an enterprise and help to make appropriate decisions. OLAP applications are managed by ad hoc, complex queries. There are two strategies used in its implementation. The first strategy uses the relational feature of conventional databases, which is called the ROLAP. The other strategy uses a data cube, known as MOLAP. Data cubes in a data warehouse are used to support data analysis. Particularly, a data cube is used to represent data along some measures of interest. In general, a data cube can be 2-dimensional, 3-dimensional, or higher dimensional. Each dimension represents an attribute in the data warehouse and the cells in the cube stand for the criterion of interest. In the data cube model, a data cube is built from a subset of attributes in the database.

Most information systems contain private information, such as social security numbers, income and disease type. So this information should be correctly protected and concealed from unauthorized access. While the security of data has been a perpetual goal in database management systems and data warehouses, mining of knowledge and avoiding of sensitive knowledge disclosure becomes the most important and highest priority target in the data mining process. Basically, the sharing of data between businesses for the purpose of gaining valuable information is useful but it can bring plenty of disadvantages. Recent advances in data mining algorithms increased the risk of information leakage and disclosure possibility (Verykios *et al.*, 2004). Because of this progress, the parallel research area has been started to overcome the information leakage risks and immunization of mining environment**.**

Privacy preserving data mining is a hot research area that investigates the side-effects of data mining methods that come from the privacy distribution of persons or organizations. A considerable amount of work on privacy preserving data mining (Rizvi & Haritsa, 2002), (Verykios *et al.*, 2004), (Goldberg, 1989) has been investigated recently. Among them, a randomization technique has been a main tool to conceal sensitive data in data mining regarding to preserve the privacy. On-line Analytical Processing system is category of software programs designed to enable many types of analyses of data stored in a database, which requires the open nature of data warehouse. Today the wide range of data warehouse users leads to more privacy concern and necessity of enforcing accurate policies to access control.

The approach which is addressed in this paper uses some special kind of transactional data cube in binary format as an input and finds appropriate solution based on the concept of Genetic Algorithms to how to modify the original data cube to hide all the sensitive rules and minimum modification performed in the original dataset. The most well-known method for transaction modification is distortion of the original database by toggling one and zero. We involve balancing some issues in sanitization of the original dataset. First, we will make some changes to hiding sensitive association rules. Second, we will not make so many changes to the lost non-sensitive association rules. Third, we will make changes in such a way that no spurious association rules will be extracted. We will try to satisfy all the objectives simultaneously.

In this paper we propose a novel framework based on genetic algorithms for privacy preserving of association rule to find the best solution for sanitizing original data cube based on multi-objective optimization. We involve balancing some critical factors in database sanitization; Starting from some changes to hide sensitive association rules and do not so many changes to loss non-sensitive association rules and finally some changes in such a way that no spurious association rules would be extracted. We try to satisfy all of these objectives simultaneously. There are so many methods to solve multi-objective problems. Some of the most well-known methods are: weighted sum strategy, ε -constraint method, set of non-inferior solutions (Pareto frontier) and goal attainment method. In our framework we try to solve this optimization problem by Pareto ranking strategy in genetic algorithm.

The rest of paper is organized as follows: Section 2 gives a summary of the high-tech methodologies and related works for privacy preserving in data mining and association rule hiding with data cube sanitization. In Section 3 we describe problem formulation and enlighten the major concepts upon which we base the proposal for the new privacy preserving framework. Section 4 describes our proposed solution for data cube sanitization against association rule mining. Section 5 presents the experiments we performed first in case study and second in large scale datasets to introduce our approach and to prove the effectiveness of our method. Finally the conclusion will be given in Section 6.

2. Related Works

Privacy issue of data management has been focused for long time. For example one of earlier papers in this research area was by Atallah (Atallah & Bertino, 1999). In this work proved that many of underlying problem in privacy preserving are NP-Hard. Therefore, most of researches have done in heuristic approach. Some of these works are stated as follows. In one of the latest papers by Verykios et al. (Verykios *et al.*, 2004), has addressed the problem of privacy preserving in association rules as "hiding association rules" and they have done by heuristic approaches. Because of many underlying NP-hard problems (Atallah & Bertino, 1999), using heuristic approaches is not astonishing. Some of most important works have done on hiding of frequent itemsets (Oliveira & Zaiane, 2002). Although they proposed four approaches to preserve privacy in datasets, these approaches are relatively limited as like as other related heuristic based works and do not warranty global optimality of their solutions in sanitization problem (this is a major drawback of heuristic approaches). Wang et al. (Wang *et al.*, 2005) propose a heuristic approach that achieves to fully eliminate all the sensitive inferences, while effectively handling overlapping rules. Their proposed algorithm identifies the set of attributes that influence the existence of each sensitive rule the most and removes them from those supporting transactions that affect the non-sensitive rules the least. Wang and Jafari (Wang & Jafari, 2005) propose two modification schemes that incorporate "unknowns" and aim at the hiding of predictive association rules, i.e. rules containing the sensitive items on their LHS. Both algorithms rely on the distortion of a portion of the database transactions to lower the confidence of the association rules. Amiri (Amiri, 2007) proposes three effective, multiple rule hiding heuristics that surpass SWA by offering higher data utility and lower distortion, at the rate of computational cost. Although there is similarity between these approaches, the proposed schemes do a better job in modeling the overall objective of a rule hiding algorithm. The work of Abul et al. (Abul *et al.*, 2006) is the first to concentrate on the NP-hardness issue involving the optimal hiding of sequences and to provide a heuristic, polynomial time algorithm that carries out the sanitization task. A different research direction concerns the use of database reconstruction approaches. Prominent research efforts towards this direction include the work of several researchers in the field of inverse frequent itemset mining (Wu *et al.*, 2005), (Wu & Chiang, 2007), (Jagannathan *et al.*, 2006). Inan (Inan & Saygin, 2006) extends the protocol-based approaches to capture the clustering of spatio-temporal data. The proposed protocol is in compliance with a series of trajectory comparison functions and allows for secure similarity computations through the use of a trusted third party. Gkoulalas and Verykios (Gkoulalas-Divanis & Verykios, 2006) propose an exact approach for hiding sensitive rules that uses the itemsets belonging in the revised positive and the revised negative borders to identify the candidate itemsets for sanitization.

Heretofore, inference control and privacy preserving for OLAP systems received less attention. Nevertheless, Lingyu Wang et al. have analytically studied this issue: (Wang *et al.*, 2003) extracts sufficient conditions for non-compromisability in sum-only data cubes; (Wang *et al.*, 2003) studies the different inference aspects caused by the multi-dimensional range queries; (Wang *et al.*, 2004) proposes a method to eliminate both unauthorized accesses and malicious inferences.

In this article we have tried to find optimal solutions for sanitization problem in data cubes. One of the most important issues in sanitization problem is that there are different criterions in privacy preserving and it is not realistic that all of these criterions are satisfied at the same time. On the other hand, in sanitization of data cube tried to keep all of these measurements at best level. In this paper we have tried to solve this multi-objective optimization problem by appropriate Genetic Algorithm approach with proper Pareto frontier fitness function. Indeed we have supposed that there are no specified priorities or costs as weights for the objectives and finally showed a set of *non-dominated* solutions (Pareto frontier) as result.

3. Problem Formulation

Let $I=\{i_1,i_2,...,i_m\}$ be a set of items and let D is the data cube of transactions that contains sensitive information and it should be sanitized before publishing. Item sets denoted as $X\subseteq I$. Each data cube which contains *k* items called *k*- data cube. Let $D=\{T_1,T_2,...,T_n\}$ be a set of transactions. The well known measure in frequent data cube mining is *support* of data cube. The *support* measure of an item $X\subseteq I$ in database D, is the count of transactions contain X and denoted as *Support_count(X)*. An itemset *X* has *support* measure *s* in data cube C if *s%* of transactions support *X* in data cube *C*. *Support* measure of *X* is denoted as *Support(X)*.

$$\text{Support}(X)=\frac{Support\_count(X)}{n}\times 100$$

where *n* is number of transactions in data cube *C*.

Itemset *X* is called frequent data cube when Support(X) $\geq MST$, where *MST* is an acronym for "Minimum Support Threshold" that is predefined threshold. After mining frequent itemsets, the association rule is an implication of the form $X\rightarrow Y$, where $X,Y\subset I$ and $X\bigcap Y=\phi$. The *Confidence* measure for rule $X\rightarrow Y$ in data cube *C* is evaluated as follows:

$$Confidence(X\rightarrow Y)=\frac{Support(XY)}{Support(X)}\times 100\cdot$$

Note while the *support* is a measure of the frequency of a rule, the *confidence* is a measure of the strength of the relation between sets of items. Association rule mining algorithms scan the data cube of transactions and evaluate the *support* and *confidence* of candidate rules to determine if they are considerable or not. A rule is considerable if its *support* and *confidence* is higher than the user specified minimum support and minimum confidence threshold. In this way, algorithms do not retrieve all possible association rules that can be derivable from a dataset, but only a very small subset that satisfies the minimum support and minimum confidence requirements set by the users. An association rule-mining algorithm works as follows. It finds all the sets of items that appear frequently enough to be considered relevant and then it derives from them the association rules that are strong enough to be considered interesting. The major goal here is to prevent some of these rules that we refer to as "sensitive rules", from being revealed. The problem of privacy preserving in association rule mining (so called association rule hiding) focused on this paper can be formulated as follows:

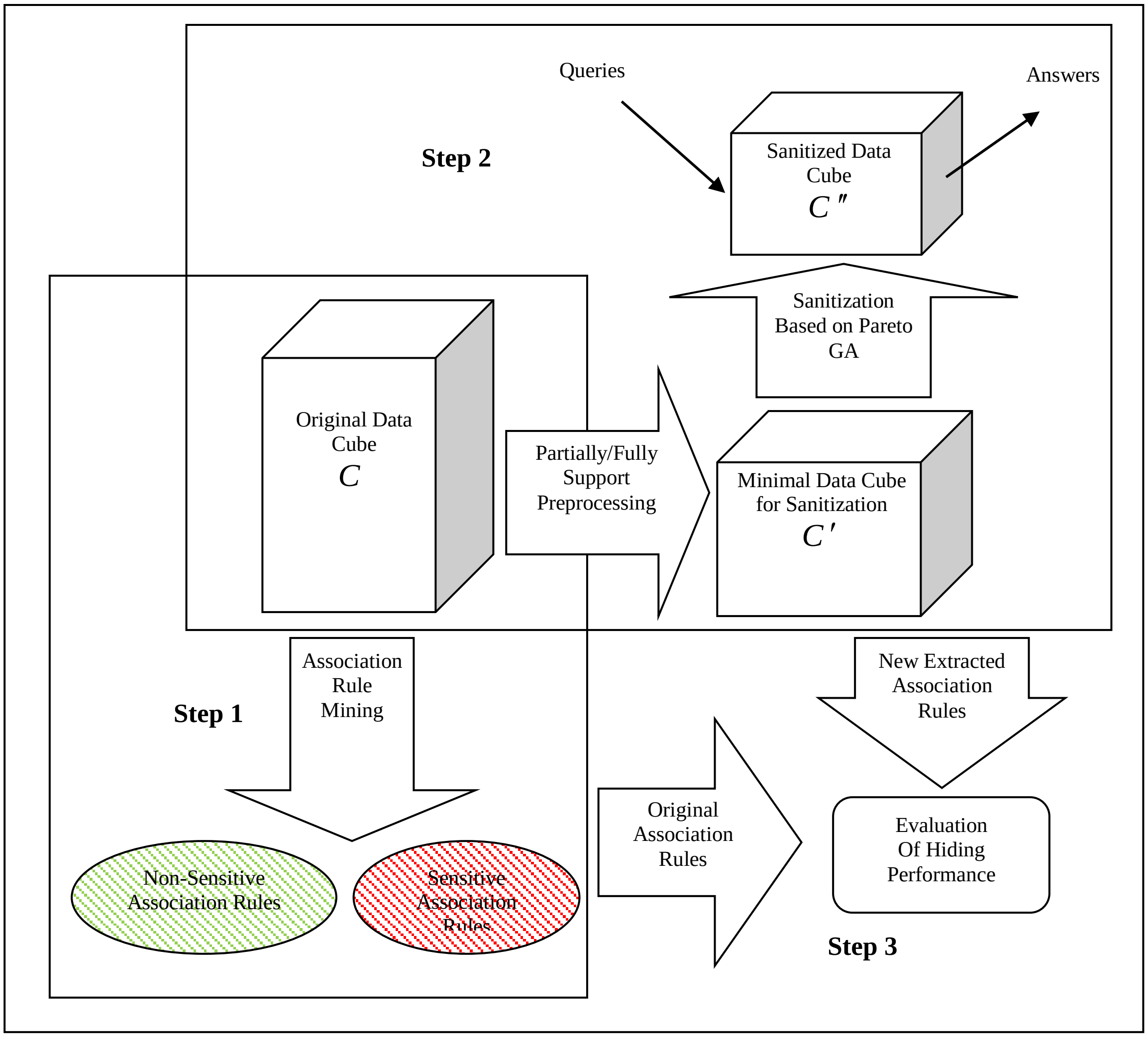
Given a transaction data cube C, minimum support threshold “MST”, minimum confidence threshold “MCT”, a set of significant association rules R mined from D and a set of sensitive rules $R_{Sen}\subseteq R$ to be hided, generate a new data cube C'', such that the rules in $R_{non-Sen}=R-R_{Sen}$ can be mind from data cube C'' under the same values of “MST” and “MCT”. Further, no normal rules in $R_{non-Sen}$ are falsely hidden (lost rules), and no extra spurious rules (ghost rules) are mistakenly will mined after the rule hiding process.

In (Atallah & Bertino, 1999) proved that solving above problem by sinking the support of the large itemsets via removing items from transactions or adding fake item into the transactions (also referred to as “sanitization” problem) are an NP-hard problem. Therefore, we are looking for a special modification of *C* (the original Data cube) in C'' (sanitized data cube which is going to be released) that *maximizes* the number of rules in $R_{non-Sen}$(*minimizing* number of lost rules) that can still be mined. Therefore we involve specific optimization problem. In one side we must conceal the sensitive association rule, thus it is necessary to modify the data cube and in the other side we should keep the utility of modified data cube to extract useful information and rules. In order to solve this optimization problem we have developed a framework and some criterion to evaluate our sanitization performance.

4. The Proposed Solution

In the following section we will explain our approach in depth. The critical phase in this work is "preprocessing step" and the related specifications of the fitness function which is to be used for our Genetic Algorithm method.

Fig.1. *Main steps in our proposed framework*


Queries
Answers
Step 2
Sanitized Data Cube
C''
Sanitization Based on Pareto GA
Original Data Cube
C
Partially/Fully Support Preprocessing
Minimal Data Cube for Sanitization
C'
Association Rule Mining
Step 1
New Extracted Association Rules
Original Association Rules
Evaluation Of Hiding Performance
Non-Sensitive Association Rules
Sensitive Association Rules
Step 3

Our proposed method is depicted in figure 1. The main steps in the framework are explained as follows:

Step 1- Initial Association Rule Mining:

- Consider a data cube with a set of items and transactions
- Apriori algorithm is used to find the frequent item sets based on the minimum support threshold.
- From the frequent item sets, the set of association rules can be generated based on the minimum support and confidence thresholds.
- Select the sensitive rules from the set of association rules.

Step 2- Preprocessing Data Cube:

- Preprocessing original data cube regarding sensitive association rules to prepare the minimal data cube for sanitization.
- Genetic algorithm is used for modifying the items based on the fitness function by Pareto ranking strategy.

Step 3- Hiding Strategy Evaluation:

- Mining the frequent item sets and the set of association rules similar to the routine in step 1, from sanitized data cube.
- Evaluate the sanitization for all measures.

4.1 Preprocess of Original Data cube

The overall workflow in our approach is depicted in Figure 1. The whole approach is divided into two steps: 1-Preprocessing of original data cube 2-Searching for the best sanitization solution based on Genetic Algorithm and according to appropriate fitness strategy in minimal dataset.

The first phase is to preprocess of original data cube and address minimal itemsets that need modification. We propose two strategies in preprocessing of the dataset. First, we can select all transactions that support sensitive itemsets. In this strategy a common item(s) between the transaction and sensitive rule is required to select the transaction. Therefore in this strategy each transaction that has sensitive items is addressed to change. So we should have amount of locations that possibly changed either fully support or partially support sensitive association rule. As a result, we need more space to generate longer chromosomes and manipulation of these chromosomes needs more time. Further, we may have so many candidate locations for modifications in original dataset, and the utility of data cube may be affected more.

Fig.2. *Preprocessing algorithm for partially supported transactions strategy.*

```
INPUT: a set of sensitive association rules to hide $R_{sen}$ and original data cube$C$

OUTPUT: the minimal data Cube $C'$ for sanitization

Begin
$C' \leftarrow \phi$    // Minimal Data Cube is empty at the start
for each sensitive association rule $r \in R_{sen}$ do {
  for each transaction $t \in C$ do {
    $common\_items \leftarrow (r_{items} \cap t_{items})$
    if $common\_items \neq \phi$ then        // Step 1: Select transaction t that partially supports sensitive rule r
      append_to_dataCube ($C', t_{common\_items}$);   // Step2: Select sensitive items form transaction t
  }
}
End
```

Fig.3. *Preprocessing algorithm based on selection of fully supported transactions strategy.*

```
INPUT: a set of sensitive association rules to hide $R_{sen}$ and original data cube$C$

OUTPUT: the minima data cube $C'$ for sanitization
Begin
$C' \leftarrow \phi$                    // Minimal Data Cube is empty at the start
for each sensitive association rule $r \in R_{sen}$ do {
  for each transaction $t \in C$ do {
    $common\_items \leftarrow (r_{items} \cap t_{items})$
    if $common\_items = r_{items}$ then        //Step1: Select transaction t that fully supports sensitive rule r
        append_to_dataCube ($C', t_{common\_items}$);    // Step 2: Select sensitive items form transaction t
  }
}
End
```

On the other hand, in this strategy we make more changes and the sensitive items will be concealed by more scrupulosity. The algorithm of first preprocessing strategy is depicted in Figure 2. Second, we can use minimum confidence threshold to select all transactions that support sensitive association rules. In this strategy each transaction that fully supports the sensitive association rule are addressed to change. In comparison with the first strategy, the strategy candidates a fewer number of transaction to change. Because many of the transactions do not support the whole typical sensitive association rule. On the other hand, in this strategy we modify a smaller number of transactions. Hence, accuracy and usefulness of data cube is also maintained. The algorithm of the second preprocessing strategy is depicted in Figure 3.

Therefore, if the high priority goal is to fully preserve sensitive items, we should select first preprocess strategy and if we are going to maintain utility of data cube more than before; the second strategy is a better choice for preprocessing. The overall view of preprocessing phase is depicted in Figure 4.

4.2 GA Proposed Solution for Privacy Preserving

1) Genetic Algorithm Background

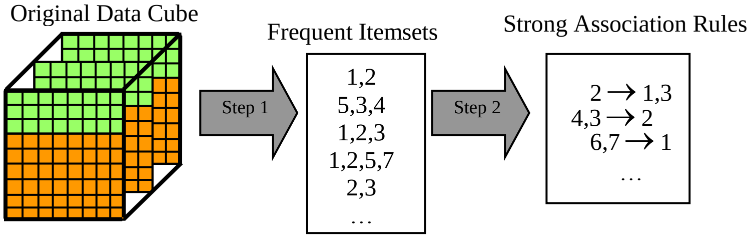
A Genetic Algorithm performs fitness tests on new structures to select the best population. Fitness determines the quality of the individual on the basis of the defined cost function. Genetic Algorithms are meta-heuristic search methods that have been developed by John Holland in 1975. GA's applied natural selection and natural genetics in artificial intelligence to find the globally optimal solution to the optimization problem from the feasible solutions (David, 1991), (Goldberg, 1989). In nature, an individual's fitness is its ability to pass on its genetic material. The fortune of an individual chromosome depends on the fitness value; the better the fitness value, the better the chance of survival. Genetic Algorithms solve design problems similar to that of natural solutions for biological design problems (Goldberg *et al.*, 1989).

2) Population Generation and Chromosome Presentation

In Genetic Algorithms, a population consists of a group of individuals called chromosomes that represent a complete solution to a defined problem. Each chromosome is a sequence of 0s or 1s. The initial set of the population is a randomly generated set of individuals. A new population is generated by two methods: steady-state Genetic Algorithm and generational Genetic Algorithm. The steady-state Genetic Algorithm replaces one or two members of the population; whereas the generational Genetic Algorithm replaces all of them at each generation of evolution. In this work a generational Genetic Algorithm is adopted as population replacement method. In this method tried to keep a certain number of the best individuals from each generation and copies them to the new generation (this approach known as elitism).

Each transaction is represented as a chromosome and presence of an ith item in transaction showed by 1 and absence of the item by 0 in ith bit of transaction. The fitness of a chromosome is determined by several factors and different strategies. Each population consists of several chromosomes and the best chromosome is used to generate the next population. For the initial population, a large number of random transactions are chosen. Based on the survival fitness, the population will transform into the future generation.

Fig.4. *The steps of Association Rule Mining from Data Cube*



3) Fitness Strategies

The dynamic area in this research is multi-objective optimization. The idea is quite simple. In these strategies fitness measurements happen in two stages. In stage one, each objective is measured with its natural fitness measurement (as like as weighted sum approach). However, these scores are not merged at all, but are kept separate for each population member within a *vector* of scores. A Genetic Algorithm would therefore evaluate each individual according to all the multi-objective evaluation tests as are necessary for the problem. Stage two involves finding overall rankings for the population. Recall that ranked fitness measurements discard absolute fitness scores, and instead replace them with integer numbers (1, 2, 3,..., with 1 being the most fit, 2 being 2nd fittest, etc.). The ranking done here uses the Pareto ranking strategy. The idea behind Pareto ranking is that it will never try to compare quantity of two objectives in different types: each dimension of the problem is always kept independent of the other dimensions, and an individual is better than, or *dominates*, another individual if it is shown to be at least as good in all dimensions, and better in at least one dimension. For a minimization problem (one in which we are trying to minimize scores), then for two individuals *U(u(1), u(2), ..., u(k))* and *V (v(1), v(2), ..., v(k))*, we say that:

$$\textbf{\textit{U} dominates \textit{V}} \quad \textit{iff: } \forall i : u(i) \le v(i) \wedge \exists i : u(i) < v(i)$$

The first expression with "for all" says that there is *U* is at least as good as *V* is in all aspects. And the second expression ("there exists") says that there is at least one aspect of *U* that is definitely better than *V*. Therefore it is so clear that *U* is superior to *V*, because it is better in at least one aspect, and not worse in any aspect.

The Pareto ranking algorithm relies on the idea of domination. It first goes through the entire population (all sanitization solutions for this problem) to find the non-dominated individuals (superior sanitization solutions). These are the individuals in which nothing dominates them. These will be assigned rank one (first one in ranking), the fittest individuals in the population. The ranking algorithms takes an individual A, and then looks through the rest of the population to see if any individual B dominates A. If so, then A cannot be in rank one, and it is skipped. If however, it is found that there is no B that dominates A, then A is assigned rank one. Once the entire population is evaluated for the rank one members, these rank one individuals are marked as "processed", and the whole procedure is repeated on the remaining population to find the rank two individuals... those that are *non-dominated* by any yet unranked individuals. This repeats until the entire population is assigned a rank.

The end result of the Pareto ranking is that each member of the population has a single Pareto rank value assigned to it. The lower rank, the better individual. These ranks can then be converted to a Roulette wheel or used within a tournament selection to create the next generation (Kim & Weck, 2005).

There will usually be sets of individuals in each rank as well. The individuals in a rank dominate all the individuals with higher rank numbers, and are in turn dominated by the sets with lower ranks. However, individuals in the same rank set are incomparable, in the sense that none of them is clearly better or worse than any other member of that set. Each individual will be better in some dimensions of the problem, but worse in others.

Based on Pareto ranking strategy, we have conducted four fitness evaluation strategies in this paper. We will discuss these strategies in following sections.

a) Confidence-based Fitness Strategy

First fitness strategy relies on both hiding all sensitive rules and minimum number of modification in original dataset. We design this fitness strategy based on Pareto ranking strategy as follows:

minimize *objective_1=Rules Hiding Distances*
AND
minimize *objective_2=Number of Modifications*

where:

- $Rules\ Hiding\ Distances = \sum_{i=1}^{Number\ of\ sensitive\ Rules} Rule_i\ \ Hiding\ Distance$
- $$Rule_i\ Hiding\ Distance = \begin{cases} 0 & if\ Confidence(Rule_i) \leq MCT \\ Confidence(Rule_i) - MCT & otherwise \end{cases}$$
- $Number\ of\ Modifications = \sum_{j=1}^{|Critical\ Transactions| \times |I|} D'_j \oplus D_j$

Where: $|CriticalTransactions|$ is number of critical transactions (in Figures 2 colored by orange) and $|I|$ is number of items in original database (denoted by *D*). And finally D'_j and D_j are jth item of each data cube after and before sanitization respectively.

Association rule mining process depicted in Figure 2. In this fitness strategy we are trying to filter sensitive rules in 2nd step of mining process. Further, this strategy tried to apply minimum modifications in original dataset.

b) Support-based Fitness Strategy

Second fitness strategy relies on both hiding all sensitive itemsets and minimum number of modification in original dataset. We design this fitness strategy based on Pareto ranking strategy as follows:

minimize *objective_1= Itemsets Hiding Distances*
AND
minimize *objective_2= Number of Modifications*

where:

- $Itemset\ \ Hiding\ Distances = \sum_{i=1}^{Number\ of\ sensitive\ Itemsets} Itemset_i\ Hiding\ Distance$
- $$Itemset_i\ Hiding\ Distance = \begin{cases} 0 & if\ Support(Itemset_i) \leq MST \\ Support(Itemset_i) - MST & otherwise \end{cases}$$
- $Number\ of\ Modifications = \sum_{j=1}^{|Critical\ Transactions| \times |I|} D'_j \oplus D_j$

Where: $|CriticalTransactions|$ is number of critical transactions (in Figure 2 colored by orange) and $|I|$ is number of items in original database (denoted by *D*). And finally D'_j and D_j are j^th item of each data cube after and before sanitization respectively.

In this fitness strategy we are trying to filter sensitive itemsets in 1st step of mining process (showed in Figure 2). Further, this strategy tried to apply minimum modifications in original dataset.

c) *Hybrid Fitness Strategy*

Third fitness strategy relies on hiding all sensitive rules and items. Further, minimum number of modification in original data cube is applied. We design this fitness strategy as hybrid of first and second strategies.

minimize *objective_1= Total Hiding Distances*
AND
minimize *objective_2= Number of Modifications*

where:

- *Total Hiding Distances=* $\sum_{i=1}^{Number\ of\ sensitive\ Itemsets / Rules} Itemset_i$ *Hiding Distance+* $Rule_i$ *Hiding Distance*

- $Itemset_i$ *Hiding Distance* $= \begin{cases} 0 & if\ Support(Itemset_i) \le MST \\ Support(Itemset_i) - MST & otherwise \end{cases}$

- $Rule_i$ *Hiding Distance* $= \begin{cases} 0 & if\ Confidence(Rule_i) \le MCT \\ Confidence(Rule_i) - MCT & otherwise \end{cases}$

- *Number of Modifications* $= \sum_{j=1}^{|Critical\ Transactions| \times |I|} D'_j \oplus D_j$

Where: $|CriticalTransactions|$ is number of critical transactions and $|I|$ is number of items in original database (denoted by *D*). And finally D'_j and D_j are j^th item of each data cube after and before sanitization respectively.

In this fitness strategy we are trying to filter sensitive itemsets/rules both in 1st and 2nd steps of mining process (showed in Figure 2). Further, this strategy tried to apply minimum modifications in original dataset.

d) *Min-Max Fitness Strategy*

Fourth fitness strategy relies on minimizing number of sensitive rules and maximizing number of non-sensitive association rules that can be extracted from sanitized dataset. (See Figures 1 to 4 again). We design this fitness strategy as follows:

minimize *objective_1=* $|R' \cap R_{Sen}|$
AND
maximize *objective_2=* $|R' \cap R_{non-Sen}|$
or

minimize *objective_1=* $|R' \cap R_{Sen}|$
AND
minimize *objective_2=* $-|R' \cap R_{non-Sen}|$

where: $|R' \cap R_{Sen}|$ is number of sensitive association rules that is mined from sanitized data cube and $|R' \cap R_{non-Sen}|$ is number of non-sensitive association rules that is mined from sanitized dataset. In this strategy tried to balance hiding all sensitive rules and keeping non-sensitive information. In other words, we have tried to preserve the privacy and accuracy of original dataset, simultaneously.

4) *Selection*

After evaluation of population's fitness, the next step is chromosome selection. Selection embodies the principle of "survival of the fittest". Satisfied fitness chromosomes are selected for reproduction. Poor chromosomes or lower fitness chromosomes may be selected a few or not at all. In this paper we have used Pareto ranking strategy. The end result of the *Pareto ranking* is that each member of the population has a single Pareto rank value assigned to it. The lower the rank, the better the individual. These ranks can then be converted to a "*Roulette-wheel*" or used within a "*Tournament*" selection to create the next generation. In *Tournament* selection, which is used in this paper, two chromosomes are chosen randomly from the population. First, for a predefined probability *p*, the more fit of these two is selected and with the probability *(1-p)* the other chromosome with less fitness is selected (Gkoulalas-Divanis & Verykios, 2006).

5) *Crossover*

Main function of crossover operation in Genetic Algorithms is combination two chromosomes together to generate new offspring (child). Crossover occurs only with some probability (crossover probability). Chromosomes are not subjected to crossover remain unmodified. The intuition behind crossover is exploration of new solutions and exploitation of old solutions. Better fitness chromosomes have a prospect to be selected more than the worse ones, so good solution always alive to the next generation. There are different crossover operators that have been developed for various purposes. Single-point crossover and multi-point are the most famous operators. In this paper single-point crossover has been applied to make new offspring. Normally high value of crossover probability is used (between 0.80 and 0.90).

6) Mutation

After performing crossover operation, the new introduced generation will only have the character of the parents. This behavior can lead to a problem where no new genetic material is introduced in the offspring and finding better population has been stopped. Mutation operator permits new genetic patterns to be introduced in the new chromosomes (random changed in random gene of chromosome). Mutation introduces a new sequence of genes into a chromosome but there is no guarantee that mutation will produce desirable features in the new chromosome. The selection process will keep it if the fitness of the mutated chromosome is better than the general population, otherwise, selection will ensure that the chromosome does not live to mate in future. Same as crossover operator, the mutation rate (mutation probability) is defined to manage how often mutation is applied. Contrasting crossover, the mutation rate is very low, about 0.005 to 0.01.

4.3 Performance Evaluation

To illustrate our proposed approach for the association rule hiding problem, validation of its feasibility and discussion about sanitization performance, let us consider an example.

B. Case Study

In this example we have original data cube and some sensitive association rule (See tables 1 to 3). A slice of data cube has shown in table 1 and the sensitive association rule in table 2. Before any modification in original data cube and with MST=0.33 and MCT=0.7, we can extract some association rules that are depicted in table 3.

Table 1. *Original dataset*

T1	1 2 3
T2	1 2 3
T3	1 2 3
T4	1 2
T5	1
T6	1 3

Table 2. *Sensitive rule*

R1	1,3 → 2

Table 3. *Association rules extracted from original data cube with MCT=0.70 and MST=0.33*

Rule	Confidence	Support
2 → 1	1	0.66
2 → 3	0.75	0.50
3 → 1	1	0.66
3 → 2	0.75	0.50
2,1 → 3	0.75	0.50
3 → 1,2	0.75	0.50
1,2 → 3	0.75	0.50
1 → 3	0.66	0.66
1,3 → 2	0.75	0.50
2,3 → 1	1	0.50

In this case study as we can see in table 1, there are six transactions in original data cube and assumed one sensitive association rule in this sanitization problem. The rule is strong (Its value of *Support* and *Confidence* measurement is greater than corresponding thresholds). So problem is what are the best solutions for modification of data cube in order to conceal the sensitive rule and keeping the accuracy of data cube (our first and second objectives)? We introduced four sanitization strategies to solve this problem in previous sections. We will show first two strategy results for this example.

Fig.5. *Pareto Front for first Fitness Function (MCT=0.70)*

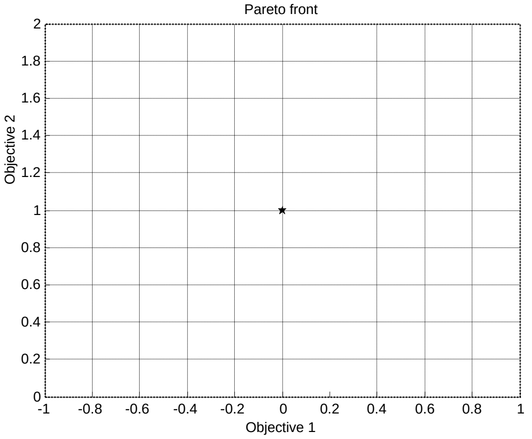


Fig.6. *Average Pareto Spread for first Fitness Function (MCT=0.70)*

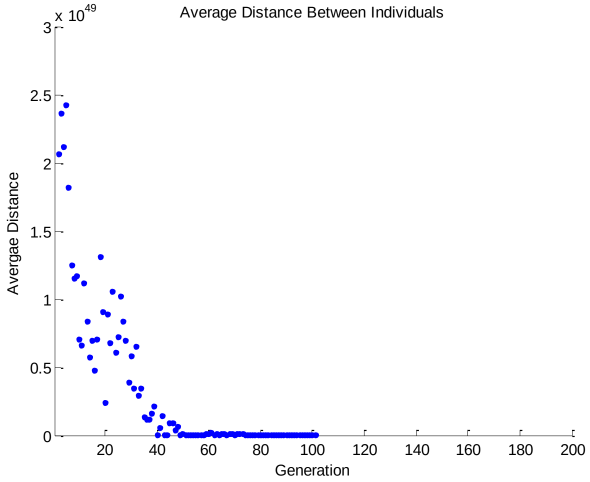
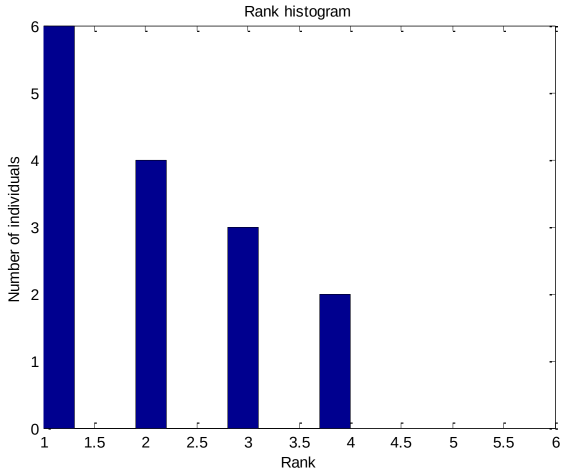


Fig.7. *Ranking per number of individuals (MCT=0.70)*



As we can see in figure 5, after running our method for first fitness function with Pareto ranking strategy, there is only one superior solution suggested for MCT=0.70. It means that this a best point that satisfy both objectives. In this case we should modify just one itemset to conceal the sensitive association rule. We can see the average Pareto spread for first fitness function for MCT=0.70 in figure 6. In figure 6 we can see that the average distance between individuals that the average is zero from generation 50 to 100. Ranking of individuals is depicted in figure 7.

C. Computational Experiments and Results on Large Datasets

The extensive computational testing was conducted, both on real and synthetic data cubes. This section describes the data used for computational testing, discusses the parameters used, and analyzes the results.

We have chosen *chess* and *mushroom* datasets as real-world data cube and *T2* data cube as synthetic data cubes. Characteristics of these data cubes are presented in table 4.

Table 4. *Characteristics of the experimental data cubes*

Data cube name	Number of transactions	Number of items
chess	3196	75
mushroom	8124	119
T2	19714	194

We will present the comparison between our approach and Algorithm 1.a (Clifton & Marks, 1996) by results obtained both on real and synthetic data cubes. In our three experiments minimum confidence threshold is 5%, minimum support threshold is 7% and number of sensitive rules is chosen randomly between 5 and 10.

Our major experimental measures are "Number of modification", "Dissimilarity" and "Execution time". The results of three experiments are shown in figure 17-19. As we can see there is almost less number of modifications needed in our approach. This difference is more significant for T2 data cube based on min-max approach. On the other hand, Alog1a, Algo1b often have better executions time than our methods, because of their simplicity and less computational complexity. Our most methods have an equal performance in execution time than Algo1a and Algo1b, especially when it used for more heavy data cubes. The main reason for this matter is our preprocessing phase and its good performance in preparing minimal data cube to association rule hiding. The main factor for better execution time in light data cubes is that Algo1a is designed based on greedy algorithm but our approach has meta-heuristic algorithm which greedy algorithms in small solution space has better performance that other exact algorithms. Although in these three experiments greedy algorithms often have less execution time but their final solution can be non-optimal in contrast with meta-heuristic algorithm. The problem of "number of modifications" is an important issue in privacy preserving approaches. In our approach all sensitive association rule are concealed completely with modifying less number of transactions in comparison with Algo1a approach. So the accuracy of our approach is higher and we loss less number of non-sensitive rules than the other method.

To have overall conclusion we integrate our experiments in each data cube for different aspects. There are three key evolution factors in our sanitization research: Number of modifications, dissimilarity between original and modified data cube and execution time of sanitization approach. We present the results of experiments for our experimental data cubes in figures 8 to 10. According the number of sensitive rules in sanitization process, these experiments are done there times for each method. Using this approach, we have managed to optimally solve problem that are many magnitude larger than those previously presented in the literature in terms of number of modifications, dissimilarity and execution time.

Fig.8. *Number of modifications in experimantal data cubes for Pareto Ranking Strategy (in four fitness function) vs. base algorithms*

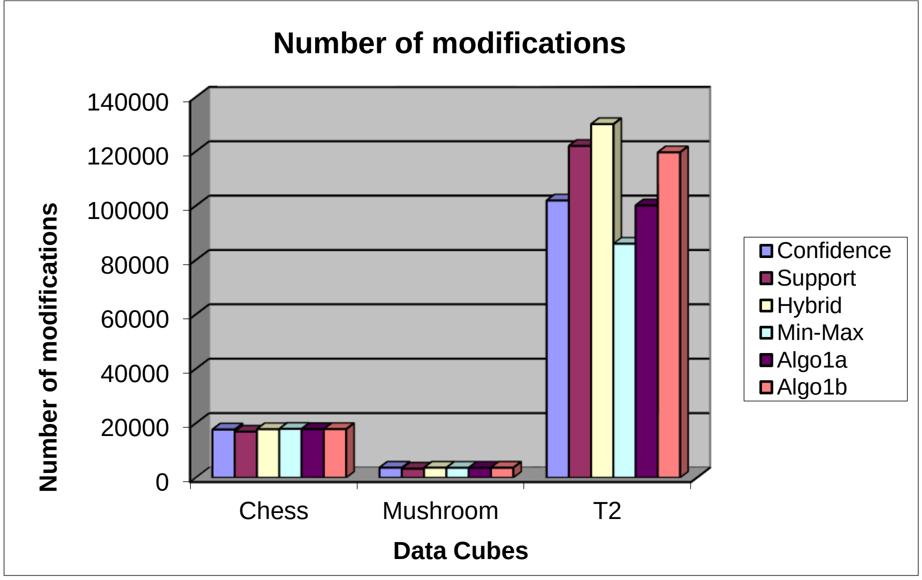


Fig.9. *Dissimilarity in experimantal data cubes for Pareto Ranking Strategy (in four fitness function) vs. base algorithms*

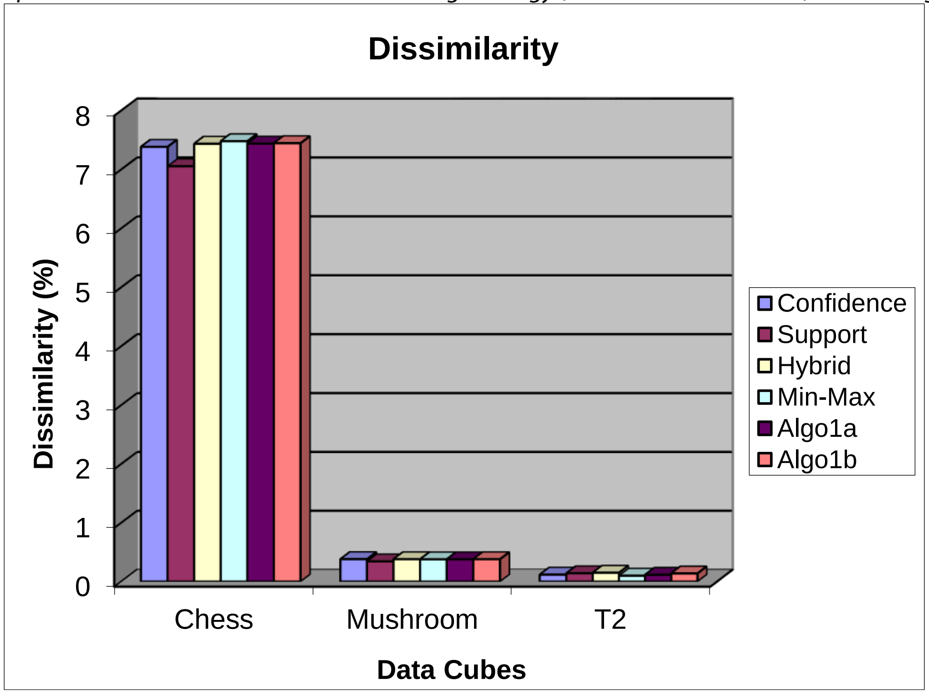
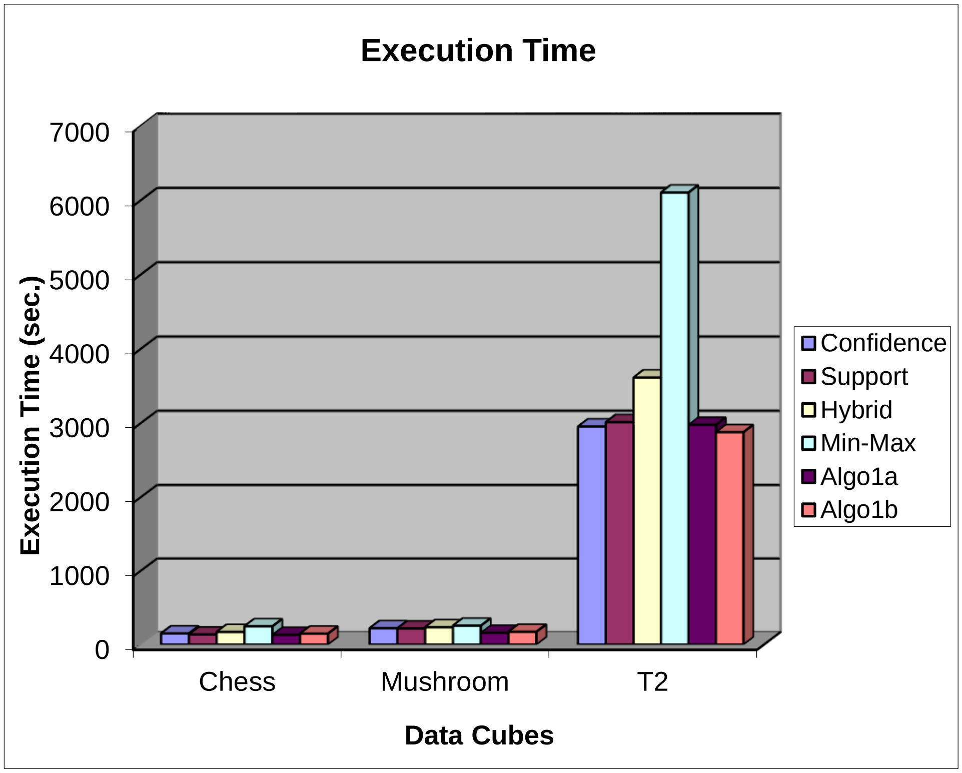


Fig.10. *Execution time in experimantal data cubes for Pareto Ranking Strategy (in four fitness function) vs. base algorithms*



5. Conclusion

This paper addresses the problem of concealing sensitive association rules in data cubes. This is an important issue that arises when data cubes are shared between firms. In this paper, a new multi-objective optimization algorithm is proposed for privacy preserving of association rule mining. To deal with the multi-objective functions, Pareto-front ranking strategy has been used for obtaining the non-dominated solutions front. In this method not only provides the efficient solution(s), but also exposes better diversity along the Pareto-optimal front. Hence more solution choices become available for designers. Actually in this work, end-user (e.g. an individual or security administrator) is free to choose more appropriate solutions based on her/his multi-objective priorities. The proposing method is more useful when proper fitness function is selected for hiding and appropriate preprocessing strategy is used for concealing frequent item sets or association rules. Because of its rapid convergence capability, the proposed fitness functions have the advantage of shortening the computational time to gain the necessary results, especially by applying proper preprocessing approach in large data cubes. The key contributions in this paper can be summarized as follows: first, two initial preprocesses are designed. These methods select which transaction(s) and which item(s) in each transaction should be changed in order to all frequent item sets/association rules concealed safely and minimum side effect accrues. Second, several sanitization strategies proposed that comprise the core of our approach. Different criteria were also introduced in these sanitization strategies. The novelties of our approach are summarized in applying meta-heuristic approach for finding best solution(s), and suggesting a variety of best solutions for all objectives. Finally the work presented here introduces the idea of rule and itemset sanitization, which complements the old idea behind data sanitization. At present, we are looking for new aspects of sanitization and proposing new fitness functions according to new types of sanitization. The final aim in this area is keeping privacy and accuracy of data cube as more as possible.

6. References

1. Chaudhuri S, Dayal U (1997) An Overview of Data Warehousing and OLAP Technology. Sigmod Record.
2. Rizvi S J and Haritsa J R (2002) Maintaining Data Privacy in Association Rule Mining. *In proceedings 28th VLDB Conference*, Hong Kong, China.
3. Verykios V, Elmagarmid A and Bertino E (2004) Association rule hiding. *IEEE Transactions on Knowledge and Data Engineering*, 16(4):434–447.

4. Clifton C and Marks D (1996) Security and privacy implications of data mining. *SIGMOD '96: Proceedings of the 2000 ACM IGMOD International Conference on Management of Data*, pages 15–20.
5. Oliveira S and Zaiane O (2002) Privacy preserving frequent itemset mining. *RPITS'14: Proceedings of the IEEE International Conference on Privacy, Security, and DataMining*, pages 43–54.
6. Sun X and Yu P S (2005) A border-based approach for hiding sensitive frequent itemsets. ICDM '05: Proceedings of the 5th IEEE International Conference on Data Mining, pages 426-433.
7. Atallah M, Bertino E (1999) A. Elmagarmid,M. Ibrahim and V. Verykios. Disclosure limitation of sensitive rules. *Proc. of IEEE Knowledge and Data Engineering Exchange Workshop (KDEX)*.
8. Verykios V, Elmagarmid A, Bertino E, Saygin Y and Dasseni E (2004) Association Rule Hiding. *IEEE Trans. on Knowledge and Data Engineering*, 16(4).
9. Oliveira S and Zaiane O (2002) Privacy preserving frequent itemset mining. *CRPITS'14: Proceedings of the IEEE International Conference on Privacy, Security, and Data Mining*, pages 43–54.
10. David L (1991) *Handbook of Genetic Algorithms*. New York : Van Nostrand Reinhold.
11. Goldberg D E (1989) *Genetic Algorithms: in Search, Optimization, and Machine Learning*. New York : Addison-Wesley Publishing Co. Inc.
12. Goldberg D, Karp B, Ke Y, Nath S, and Seshan S (1989) *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley.
13. Kim I Y and Weck O L de (2005) Adaptive weighted-sum method for bi-objective optimization: Pareto front generation. *Struct Multidisc Optim. 29*, 149–158, *Springer*.
14. Amiri (2007) Dare to share: Protecting sensitive knowledge with data sanitization. *Decision Support Systems*, 43(1):181–191.
15. Wang K, Fung B C M, and Yu P S (2005) Template-based privacy preservation in classification problems. In *Proceedings of the Fifth IEEE International Conference on Data Mining (ICDM 2005)*, pages 466–473.
16. Wang S L and Jafari A (2005) Using unknowns for hiding sensitive predictive association rules. In *Proceedings of the 2005 IEEE International Conference on Information Reuse and Integration (IRI 2005)*, pages 223–228.
17. Wu X, Wu Y, Wang Y, and Li Y (2005) Privacy aware market basket data set generation: A feasible approach for inverse frequent set mining. In *Proceedings of the 2005 SIAM International Conference on Data Mining (SDM 2005)*.
18. Wu Y H, Chiang C M, and Chen A L P (2007) Hiding sensitive association rules with limited side effects. *IEEE Transactions on Knowledge and Data Engineering*, 19(1):29–42.
19. Abul O, Atzori M, Bonchi F, and Giannotti F (2006) Hiding sequences. Technical report, Pisa KDD Laboratory, ISTI-CNR, Area della Ricerca di Pisa.
20. Gkoulalas-Divanis and Verykios V (2006) An integer programming approach for frequent itemset hiding. In *Proceedings of the 2006 ACM Conference on Information and Knowledge Management (CIKM 2006)*, pages 748–757.
21. Inan and Saygin Y (2006) Privacy preserving spatio-temporal clustering on horizontally partitioned data. In *Proceedings of the 8th International Conference on Data Warehousing and Knowledge Discovery (DaWaK 2006)*, pages 459–468.
22. Jagannathan G, Pillaipakkamnatt K, and Wright R N (2006) A new privacy preserving distributed k-clustering algorithm. In *Proceedings of the 2006 SIAM International Conference on Data Mining (SDM 2006)*, 2006.
23. Wang L, Wijesekera D (2002) Cardinality-based Inference Control in Sum-only Data Cubes. Proc. of the 7th European Symp. on Research in Computer Security.
24. Wang L, Li Y, Wijesekera D and Jajodia S (2003) Precisely Answering Multi-dimensional Range Queries without Privacy Breaches. ESORICS 2003, pages 100-115.
25. Wang L, Jajodia S and Wijesekera D (2004) Securing OLAP data cubes against privacy breaches. Proc. IEEE Symp. on Security and Privacy, pages 161-175