Available online at www.sciencedirect.com
ScienceDirect

Computers and Mathematics with Applications 54 (2007) 1303–1318

An International Journal
computers & mathematics with applications

www.elsevier.com/locate/camwa

Association rule mining through the ant colony system for National Health Insurance Research Database in Taiwan

R.J. Kuo*, C.W. Shih

Department of Industrial Engineering and Management, National Taipei University of Technology, Taipei, Taiwan, ROC

Received 17 March 2006; accepted 21 March 2006

Abstract

In the field of data mining, an important issue for association rules generation is frequent itemset discovery, which is the key factor in implementing association rule mining. Therefore, this study considers the user's assigned constraints in the mining process. Constraint-based mining enables users to concentrate on mining itemsets that are interesting to themselves, which improves the efficiency of mining tasks. In addition, in the real world, users may prefer recording more than one attribute and setting multi-dimensional constraints. Thus, this study intends to solve the multi-dimensional constraints problem for association rules generation.

The ant colony system (ACS) is one of the newest meta-heuristics for combinatorial optimization problems, and this study uses the ant colony system to mine a large database to find the association rules effectively. If this system can consider multi-dimensional constraints, the association rules will be generated more effectively. Therefore, this study proposes a novel approach of applying the ant colony system for extracting the association rules from the database. In addition, the multi-dimensional constraints are taken into account. The results using a real case, the National Health Insurance Research Database, show that the proposed method is able to provide more condensed rules than the Apriori method. The computational time is also reduced.

Keywords: Data mining; Multiple dimensional constraints; Ant colony system; Apriori

1. Introduction

Mining association rules from a large database of business data, such as transaction records, has been an important issue in the field of data mining. The problem of association rule mining can be divided into two sub-problems: (1) frequent itemset discovery and (2) association rules generation. It has also been shown that the overall performance of mining is seriously determined by the first sub-problem.

Frequent itemset mining algorithms often generate a very large number of frequent itemsets and rules, which reduce both the efficiency and also the effectiveness of the mining algorithms since only the subset of the complete frequent itemsets and association rules is of interest to users. In addition, the users need an additional post-processing step to filter the large number of mined rules to determine the useful ones. Recent work [1–4] has highlighted the importance

* Corresponding author. Tel.: +886 2 27712171x4541; fax: +886 2 27317168.
E-mail address: rjkuo@ntut.edu.tw (R.J. Kuo).

0898-1221/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2006.03.043

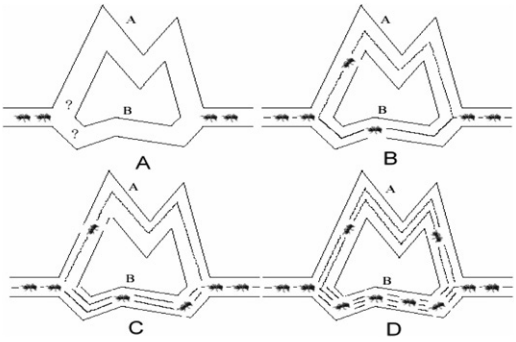


Fig. 1. The behavior of real ants.

of constraint-based mining. They exploit user-specific constraints in the mining process to improve performance, or efficiency. With multi-dimensional items, constraints can be imposed on multiple dimensional attributes. We classify multi-dimensional constraints into two cases according to the number of sub-constraints including: (1) single constraint against multiple dimensions, such as $\max(X, \text{cost}) \leq (X, \text{price})$, where X is an itemset and each item in X contains two attributes "cost" and "price", and (2) conjunction and/or disjunction of multiple sub-constraints, such as $(C_1 : X, \text{cost} \leq v_1) \wedge (C_2 : X, \text{price} \leq v_2)$, where v_1 and v_2 are constant values, respectively.

Therefore, this study intends to use the *ant colony system*, which has recently been shown to be very promising in the areas of the traveling salesman problem and scheduling [5,6], for multiple dimensional constraints mining association rules. Furthermore, since data mining has rarely been applied to solve questions in medical science, this study uses data from the National Health Insurance Research Database of Taiwan to find disease association rules. Here, an important issue is to find the potential disease and early prevention. The evaluation results show that the proposed method, using the ant colony system, really can provide more concise and accurate information than the conventional Apriori-based algorithm.

The rest of this paper is organized as follows. Section 2 summarizes some important background information, and the proposed method is described in Section 3. Section 4 presents the evaluation results and discussion. Finally, concluding remarks are made in Section 5.

2. Background

This section reviews three aspect of the literature. Two of these are the main component of the proposed method, namely Ant Algorithms, and Association Rule Mining, which is a technique for mining patterns. Finally, a related survey of Multi-Dimensional Constraints Mining is presented. Detailed information is presented below.

2.1. Ant algorithms

2.1.1. Concept of ant theorem

In the real world, ants communicate with other ants by a trail of chemicals called "*pheromones*" which are deposited by ants when they search for food. Then, the other ants encounter the previously laid pheromones and decide how many probabilities they will follow. As more and more ants pass by the same path, the pheromones on the shorter path would be increased, but the pheromone would evaporate on the other paths, as illustrated in Fig. 1.

2.1.2. The evolution and applications of ant algorithms

The first ant algorithm was introduced in Dorigo's dissertation, called *Ant System* [7], and was inspired by observation of real ant colonies. One of the first applications for the ant algorithm was for the traveling salesman problem (TSP). Recently, ACO has successfully been applied to several combinatorial optimization problems and yielded many promising developments. For instance, Gambardella and Dorigo [8] proposed *Ant-Q* station transition rules and used Q-learning to strengthen and renew pheromone trails. Bullnheimer et al. [9] proposed selecting a quantity of elite ants to renew the pheromone trail, this was called the $\mathbf{AS}_{\text{rank}}$ algorithm. Dorigo and Gambardella [10] presented a method to improve the ant system, called *Ant Colony System*. The current development of ant algorithms

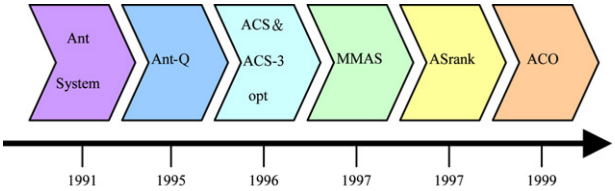


Fig. 2. Ant algorithm evolution.

can solve many kinds of optimal problems. Thereafter, Stuttzle and Hoos [11] proposed a pheromone trail that is controlled by the up-and-down limits in order to reduce stagnation, called *MAX–MIN Ant System*.

In the 1999, Dorigo et al. [12] concluded the ant system, ant colony system and other applications to propose the *Ant Colony Optimization* (ACO) Meta-Heuristic. The method is a concept for finding optimal solutions. In fact, ACO is an idea which can be employed for combinatorial optimization problems. Fig. 2 illustrates the ant algorithm evolution.

2.1.3. Ant colony system

The ant colony system (ACS) is an important part of this study, which is based on agents that simulate the natural behavior of ants, develop mechanisms of cooperation and learn from experiences [10]. The heuristics have been shown to be robust and versatile for different problems. In addition, ACS is a population-based heuristics that enables the exploration of the positive feedback between agents as a search mechanism.

There are some differences between real ants and artificial ants. Artificial ants have memory and are not completely blind. In addition to pheromone-based communication, a real ACS has another characteristic. An artificial ant has a probabilistic preference for paths with larger amounts of pheromone. Consequently, shorter paths tend to have a higher rate of growth in the amount of pheromone. The construction or modification of a solution is performed in a probabilistic way. The probability of adding a new item to the solution under construction is, in turn, a function of a problem dependent heuristic (η) and the amount of pheromone (τ) previously deposited in this trail. The pheromone trails are updated considering the evaporation rate and the quality of the current solution. Therefore, a practical implementation of an ACS includes the following typical definitions [13]:

- A heuristic function (η) that measures the quality of the items that can be added to the current partial solution.
- A method to enforce the construction of valid solutions.
- A rule that specifies how a pheromone trail (τ) should be updated.
- A probabilistic transition rule that uses the current value of the heuristic function (η) and the current amount of pheromone in the trail (τ).

ACS is a particular algorithm of ACO whereas real ants are able to communicate information concerning food sources via an aromatic essence. While searching for food, they secrete a pheromone to mark the path leading to a food source. When there are more pheromones on a path, there is larger probability that other ants will use that path, and therefore the pheromone trail on such a path will grow faster and attract more ants to follow. In ACS, the method whereby ants select the path is changed, called ACS state transition rule. When ant k in the city r will go to next city s, the selection rule is:

$$s = \begin{cases} \arg \max\limits_{u \in J_k(r)} \{\tau(r,u) \cdot \eta(r,u)^{\beta}\}, & \text{if } q \leq q_0 \\ S, & \text{otherwise} \end{cases} \tag{1}$$

where $0 \leq q \leq 1$ is randomly produced and q with $0 \leq q_0 \leq 1$ a random parameter of the system. S is the city by the *random-proportional* rule selection, which is defined as:

$$p_k(r,s) = \begin{cases} \dfrac{\tau(r,s) \cdot \eta(r,s)^{\beta}}{\sum\limits_{u \in J_k(r)} \tau(r,u) \cdot \eta(r,u)^{\beta}}, & \text{if } s \in J_k(r) \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where τ is called pheromone trials, and η is $1/d$ between the nodes. Thus, d represents distance, J_K means that ant k is non-passed city after ant k pass city r, and β is another system parameter in the ant colony system.

These two formulas are overall called *pseudo random proportional* rules, and they are according to the method of Ant-Q in the ant evolution. There are three models for the station transition rule: *pseudo random, pseudo random proportional*, and *random proportional*. Eq. (1) is called the act of exploitation as $q \leq q_0$; otherwise s is equal to S, which is called the act of biased exploration.

In ACS, the pheromone trials are divided into two parts, the ACS global and local updating rules, respectively. The ACS global updating rule is referred to the ANT-cycle method in the ant system. When ants have completed all their tours, the pheromone trial could be renewed, which is called the offline method. The ACS local updating rule is referred to as the ANT-density method in the ant system. When an ant is walking, each step renews the pheromone trail once, called online method.

The ACS global updating rule is presented as:

$$\tau(r, s) = (1 - \alpha) \cdot \tau(r, s) + \alpha \cdot \Delta\tau(r, s) \tag{3}$$

where

$$\Delta\tau(r, s) = \begin{cases} \dfrac{1}{L_{gb}}, & \text{if } (r, s) \in \text{global-best-tour} \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

In addition, $0 < \alpha < 1$ is called the pheromone decay parameter, and L_{gb} is the shortest path from the first point to current point (in the TSP problem). ACS is the method through which the ants find the shortest path from the start to the current point. Therefore, it can reach the optimal solution.

The ACS local updating rule is presented in the following equation:

$$\tau(r, s) = (1 - \rho) \cdot \tau(r, s) + \rho \cdot \Delta\tau(r, s) \tag{5}$$

where $0 < \rho < 1$ is called the pheromone evaporate parameter, and $\Delta\tau(r, s) = \tau_0$.

The ACS local updating rule is similar to the ACS global updating rule. It is increased by a fix quantity of pheromone trails every time. When the pheromone on the original path is bigger than τ_0, the pheromone value on the path is decreased after the local updating rule. This can prevent a larger number of ants using the same path, which causes pheromone trials to stagnate. When ants travel from one to another item, they could do local updating; and when ants finish their travel once, global updating is implemented.

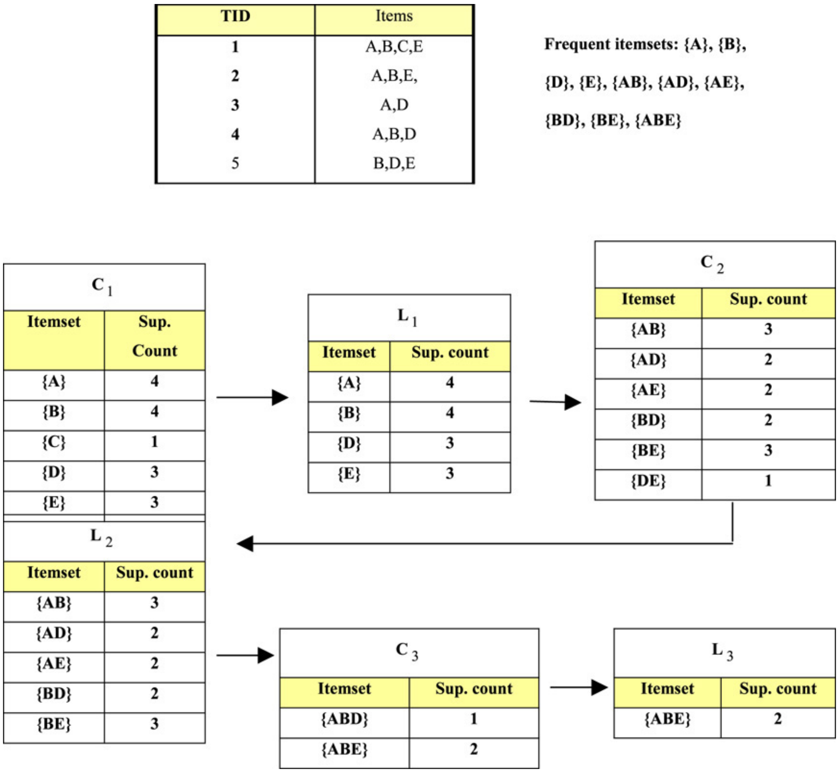
2.2. *Data mining*

Data mining includes several activities. It should acquire data from internal and external sources; and then the data needs to be translated, cleaned, and formatted for analysis, validation, and integration. As such, data mining encompasses computer-based methods that extract patterns or information for data yet requires only limited human involvement. Most of these methods are relatively recent and are based on the area of artificial intelligence.

The following five foundation-level analysis domains are the "reason why" of data mining: summarization, predictive modeling, clustering/segmentation, classification, and link analysis [14]. Link analysis refers to a family of methods that are employed to correlate pattern cross-sections over time. In marketing, a link analysis model can provide information about the buyers' behavior. Using the same idea to analyze medical behavior, link analysis can find patterns in patients' visits to doctors. This can be helpful for diagnosis and deciding on drugs. If a medical analyst can find out which groups of sets of items are most likely to be diagnosed in a particular group of patients, he can make several treating strategies, depending on the results of link analysis for their regular uses to make more effects. Because of its importance in medical science, in this study we will focus on this issue.

2.3. *Association rules mining*

Agrawal et al. [15] first addressed the issue of mining association rules in 1993. They pointed out that there are some hidden relationships among the purchased items in transactional databases. For example, there are associations or relationships between items such as bread and milk, which are often purchased together in a single basket transaction. The mining results can help understand the customer's purchase behavior, which might not have been previously perceived.



TID	Items
1	A,B,C,E
2	A,B,E,
3	A,D
4	A,B,D
5	B,D,E

Frequent itemsets: {A}, {B}, {D}, {E}, {AB}, {AD}, {AE}, {BD}, {BE}, {ABE}

C 1	
Itemset	Sup. Count
{A}	4
{B}	4
{C}	1
{D}	3
{E}	3

L 1	
Itemset	Sup. count
{A}	4
{B}	4
{D}	3
{E}	3

C 2	
Itemset	Sup. count
{AB}	3
{AD}	2
{AE}	2
{BD}	2
{BE}	3
{DE}	1

L 2	
Itemset	Sup. count
{AB}	3
{AD}	2
{AE}	2
{BD}	2
{BE}	3

C 3	
Itemset	Sup. count
{ABD}	1
{ABE}	2

L 3	
Itemset	Sup. count
{ABE}	2

Fig. 3. Frequent itemset mining by Apriori algorithm.

An association rule is of the form $X \Rightarrow Y$, where X and Y are both frequent itemsets in the given database and the intersection of X and Y is an empty set, i.e., $X \cap Y = \varnothing$. The support of the rule $X \Rightarrow Y$ is the percentage of transactions in the given database that contain both X and Y, i.e., $P(X \cup Y)$. The confidence of the rule $X \Rightarrow Y$ is the percentage of transactions in the given database containing X that also contains Y, i.e., $P(Y|X)$. Therefore, association rule mining is used to find all the association rules among itemsets in a given database, where the support and confidence of these association rules must satisfy the user-specified minimum support and minimum confidence. The problem of association rule mining can be divided into two sub-problems:

1. Finding frequent itemsets with their supports above the minimum support threshold.
2. Using frequent itemsets found in the step 1 to generate association rules that have a confidence level above the minimum confidence threshold.

Therefore, many studies of association rule mining concentrate on developing efficient algorithms for frequent itemset discovery. The following subsections summarize some of the most popular algorithms for frequent itemset mining.

2.3.1. Apriori-like algorithm

Agrawal et al. [16] proposed the well-known algorithm, Apriori, to mine large itemsets to find out the association rules among items. This algorithm employs a level-wise approach, which iteratively generates candidate k-itemsets from previously found frequent $(k-1)$-itemsets, and then checks the supports of candidates to form frequent k-itemsets. The algorithm scans multiple passes over the database. The efficiency and correctness of the level-wise generation of frequent itemsets are based on an important property, called the Apriori Property.

The algorithm is first pass counts item occurrences to find the set of frequent 1-itemsets, denoted as L_1. A subsequent pass, say pass k, consists of two steps; the join and prune steps. In the join step, a set of candidate k-itemsets (denoted as C_k) is generated by joining the frequent itemsets L_{k-1}found in the $(k-1)$th pass with itself. For example, Fig. 3 demonstrates how to find frequent itemsets in min_sup = 2.

2.3.2. FP-growth algorithm

Han and Pei [17] proposed a novel frequent pattern tree (FP-tree) structure, which contains all the compact information for mining frequent itemsets, and then proposed the FP-growth algorithm, which adopts a pattern segment growth approach to prevent generating a large number of candidate itemsets. Their mining method only scans the whole database twice and does not need to generate candidate itemsets, and so is very efficient.

2.3.3. Parallel mining

Parallel mining [18] is another technique used to improve the classic algorithm of mining association rules on the premise that there exist multiple processors in the computing environment. The core idea of parallel mining is to separate the mining tasks into several sub-tasks so that each sub-task can be performed simultaneously on various processors, which are embedded in the same computer system or even spread over the distributed systems. Thus; this improves the efficiency of the overall algorithm for mining association rules.

2.3.4. Sampling algorithm

A random sampling technique [19] was used to find association rules to reduce database activity. The sampling algorithm applies the level-based method on the sample with lower minimum support threshold to mine the superset of large itemsets. This method produces exact association rules, but in some cases it does not generate the entire association rules, that is, there might exist some missing association rules. Therefore, this approach requires only one full pass over the database in most cases, and only two passes in the worst case.

2.3.5. Lattice-based algorithm

Zaki [20] organized the items into a lattice structure and presented a set of algorithms including Eclat, MaxEclat, MaxClique, TopDown and AprClique for identifying maximal large itemsets. All of the algorithms attempt to look ahead and identify long large itemsets early to help prune the number of candidate itemsets considered. There are also another two approaches for mining long large itemsets. Lin and Kedem [21] proposed Pincer-Search algorithm for mining long large itemsets, whereas Bayardo [22] proposed the Max–Miner algorithm. Both algorithms attempt to discover the long and large scale patterns through the search effort. The greatest difference between the two methods is in the generation of candidate itemsets. The Max–Miner approach generates the candidate itemsets in polynomial time since it is an NP-hard problem in the Pincer-Search method to ensure that no long candidate itemsets contain any known infrequent itemset.

2.3.6. Partition algorithm

For mining association rules, Savasere et al. [23] introduced a partition algorithm that is fundamentally different from the classic algorithm. First, a partition algorithm scans the database once to generate a set of all potentially large itemsets, and then the supports for all the itemsets are measured in the second scan of the database. The key to correctness of the partition algorithm is that a potentially large itemset appears as a large itemset in at least one of the partitions. This algorithm logically divides the database into a number of non-overlapping partitions, which can be held in the main memory. The partitions are considered individually and all large itemsets for that partition are generated. These large itemsets are further merged to create a set of all potential large itemsets. Then these itemsets are generated.

2.4. Multiple-dimensional constraints mining

Regardless of the interests of users, the data mining process may disclose a large number of rules, but only some of the uncovered rules are relevant to users. The best rule according to any of a user's requests must reside along a support/confidence border. Even so, under the guidance of various kinds of constraints provided by the user, constraint-based mining can discover more significant rules.

In their study on the constraints that can be pushed deep into the mining process, Pei and Han [3,4] categorized all constraints into five classes: anti-monotone, monotone, succinct, convertible and inconvertible. Therefore, if the attributes or items in the association rules only refer to a single dimension, it is called a single dimensional association rule. For instance, buy diapers ⇒ buy beer is only from the point of view of "buy", so it is a single dimensional association rule. By contrast, if the attributes or items in the association rules refer to more than two

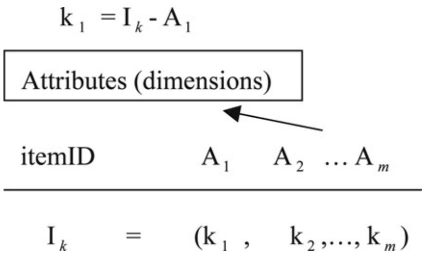


Fig. 4. Multi-dimensional items.

dimensions, it is called a multiple dimensional association rule. For instance, [age (X, "50…60") $\wedge$ salary (X, "60 K …70 K") $\Rightarrow$ buy (X, digital TV)] includes "age", "salary" and "buy items" is three dimensional, so it is a multiple dimensional association rule. Multiple dimensional association rules more accurately express the mining results than single dimensional association rules. Users can assign the mining dimensional conditions, and connect constraint conditions to find the rules efficiently.

2.5. *Applications of ant colony system for mining association rules*

The ant colony system employed for mining association rules is a very new application, although there have been a few applications in data mining. Parpinelli et al. [24] used the classification technique applied to unseen data as a decision aid. Wu and Shi [25] used the ants clustering algorithm to test the UCI machine learning data. Also, Teles et al. [26] used ant theory as a metaphor to predict user activity on a web site.

Regarding the application of the ant colony system for mining association rules, Su [27] adopted the technique and concept of Ant System to develop association rules. The developed algorithm is supported by quality data, quantity data, and mixed data. According to its results, the ant system must take more time in running the data in the assign cycle; and if the data is critical or has time constraints, it may not feasible. Furthermore, there are some parameters in the ant algorithm which need to be pre-determined, which may be time consuming. Therefore, this study aims at resolving the foregoing problems to improve the method. We employ the constraints concept to decrease the run time, and let almost all of the parameters be known before running the model.

3. Methodology

The proposed method of association rules for mining multiple dimensional constraints is described in this section. The following subsections will describe the problem definition and the proposed method, ant colony system, respectively.

3.1. *Problem definition*

Let $\Phi = \{I_1, I_2, \ldots, I_n\}$ be a set of all items, where an item is an object with m dimensional attributes $m \geq 1$ that are so-called dimensions (e.g., weight, high, cost, …etc.), as illustrated in Fig. 4. The value k_m is on dimension A_j [$j \in \{1, 2, \ldots, m\}$] of item $I_k - A_j$.

Definition 3.1 (*Association Rules Mining*). If $r(A_i, A_j) = \tau_{ij} \geq \tau_{\text{threshoid}}$, $(-1 \leq \tau_{ij} \leq 1)$, represents the degree of relations between $A_i, A_j, \forall i, j = 1, \ldots, n$ with $r(A_i, A_j) = r(A_j, A_i)$ and $r(A_i, A_i) = 1$, then an association rule is an expression of $A_i \Leftrightarrow A_j$, for any $A_i, A_j \in A$ when $\tau_{ij} \geq \tau_{\text{threshoid}}$, with $0 \leq \tau_{\text{threshoid}} \leq 1$.

Definition 3.2 (*Mining Frequent Itemsets with Multi-Dimensional Constraints*). Given a transaction database T and a set of multi-dimensional constraints C, the problem of mining itemsets with multi-dimensional constraints is to find the complete set [$\text{SAT}_c(\Phi)$] of itemsets satisfying C. For example, we want to find an itemset [$\text{SAT}_c(\Phi)$] for which the man buying a car satisfies the constraints of age between 30–40 and salary between 20 K–30 K, as illustrated in Fig. 5.

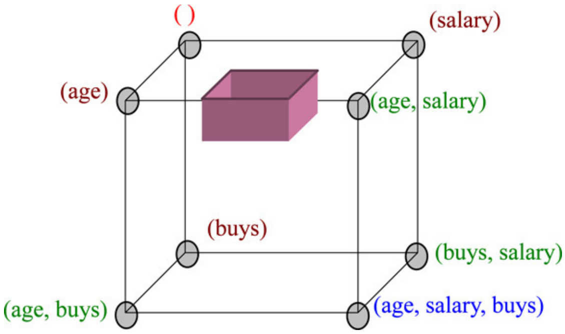


Fig. 5. An example of multi-dimensional constraints mining.

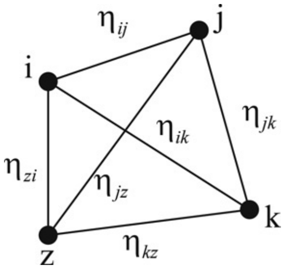


Fig. 6. An simply example.

As mentioned in frequent itemset discovery, support is a common measure of statistical significance, so it is generally accepted that the frequency is a necessary constraint for useful/interesting itemsets. Therefore, we automatically include the frequency constraint C_{freq} into our problem definition.

In the following discussion, although we do not specifically mention C_{freq}, in fact, the constraints of the problem include C_{freq} automatically. So when discussing constraints, we simply concentrate on constraints other than C_{freq}. However, it is critical to remember the existence of C_{freq}.

3.2. The proposed algorithm

Since the original database adopted the constraint conditions, we will apply an ant colony system to the association rules to conform to constraints. The association rules with n items construct a complete graph, where each pair of vertices is joined by an edge, as illustrated in Fig. 6. Let η_{ij} be the frequency between items i and j. The η_{ij} edges represent the frequency between items.

Let $b_i(t)$ $(i = 1, \ldots, n)$ be the number of ants at item i at time t and let $m = \sum_{i=1}^{n} b_i(t)$ be the total number of ants at time t. All ants will follow:

1. Ant chooses next item j to follow by the *state transition rule* that is defined by:

$$j = \begin{cases} \arg\max\limits_{u \in Z}\{\tau_{iu}(t) \cdot \eta_{iu}^{\beta}\}, & \text{if } q \le q_0 \\ S, & \text{otherwise} \end{cases} \tag{6}$$

   where Z is the set of the ant unaccomplished tour and β is a system parameter. In addition, q and q_0 are random numbers uniformly distributed in [0, 1] and the parameter $(0 \le q_0 \le 1)$ determines the relative importance of exploitation versus exploration, respectively. If $q \le q_0$, the item of unaccomplished tours j with maximum $\tau_{iu}(t)\eta_{iu}^{\beta}$ value is put at position (exploitation); otherwise the item is chosen according to S (biased exploration).
2. The random variable S is selected according to the probability distribution of the random-proportional rule as follows:

$$p_{ij}^{k}=\begin{cases}\dfrac{\tau_{ij}(t)\cdot\eta_{ij}^{\beta}}{\sum\limits_{u\in Z}\tau_{iu}(t)\cdot\eta_{iu}^{\beta}}, & \text{if } j\in Z\\ 0, & \text{otherwise.}\end{cases} \tag{7}$$

The resulting state transition rules refer to Eqs. (6) and (7), and are called the *pseudo-random-proportional* rule.

3. Ants can only choose a path that has never been used (increase *tabu*).
4. After traveling on a path, an ant will lay some pheromone on it (local updating).

Let $\tau_{ij}(t)$ be the intensity of pheromone trail on edge (i, j) at time t, therefore, we can consider an iteration to be when an ant completes the tour, and the next iteration will be started at $(t+1)$. Then the pheromone intensity is updated according to:

$$\tau_{ij}(t+1)=[(1-\alpha)\cdot\tau_{ij}(t)]+\alpha\cdot\Delta\tau_{ij} \tag{8}$$

where α $(0\le\alpha\le 1)$ is a coefficient for the remaining percentage of pheromone between time t to $t+1$. The association rules α is considered as a time series coefficient, which will decrease the impact levels of the old data and regulate the coefficient $\Delta\tau_{ij}$. Therefore, if the data is irrelevant to time, set $\alpha=0$.

For mining association rules, use Φ-correlation instead of correlation coefficient, as defined below:

$$\Delta\tau_{ij}^{k}(t)=\begin{cases}\dfrac{1}{L_{gb}}, & \text{if the } k\text{th ant in its tour is global-high-frequency}\\ 0, & \text{otherwise}\end{cases} \tag{9}$$

L_{gb} is proposed as the highest frequency from first ant to kth ant accumulate value. It is also the pheromone intensity among the shortest path ij, while each ant completes its trip at time t.

Now let us summarize our algorithm as follows:

Step 1: *Initialization*

Set $t=0$ $\{t$ is the time counter$\}$;
Set $NC=0$ $\{NC$ is the iteration counter$\}$;
Set $\tau_{ij}(t)=c$ and $\Delta\tau_{ij}=0$ and $\tau_0=c, \forall i, j=1,\ldots,n; i\neq j$;
Set $m=n$ Place the mth ant on the nth nodes (items)
Set $\beta=c$ and $q=c$ and $\rho=c$ and $\alpha=c\in[0,1]$; tabu$(s)=\varnothing$.

Step 2: *Multi-dimensional constraints test*

Scan the database once and find the complete set $[\mathrm{SAT}_c(\Phi)]$ of itemsets satisfying C.

Step 3: *Mining guided by ant colony system*

1. Calculate $\eta_{ij}(t)=\mathrm{support}_{ij}(t)$ from set $[\mathrm{SAT}_c(\Phi)]$.
2. Chooses next item j by the state transition rule,

$$j=\begin{cases}\arg\max\limits_{u\in Z}\{\tau_{iu}(t)\cdot\eta_{iu}^{\beta}\}, & \text{if } q\le q_0\\ S, & \text{otherwise.}\end{cases} \tag{10}$$

3. If $q\ge q_0$, then choose the next edge ij until a given step is selected to move to with the transition probability,

$$p_{ij}^{k}=\begin{cases}\dfrac{\tau_{ij}(t)\cdot\eta_{ij}^{\beta}}{\sum\limits_{u\in Z}\tau_{iu}(t)\cdot\eta_{iu}^{\beta}}, & \text{if } j\in Z\\ 0, & \text{otherwise.}\end{cases} \tag{11}$$

4. Move the kth ant from node i to the node j and insert that path into tabu (s).
5. Move the kth ant from node i to the node j and change the local pheromone trial by using

$$\tau_{ij}^{k}=(1-\rho)\cdot\tau_{ij}+\rho\cdot\Delta\tau_{ij} \tag{12}$$

where $0<\rho<1$ and $\Delta\tau_{ij}=\tau_0$.

6. After running a cycle, we update $\Delta\tau_{ij}^{k}(t)$ follows:

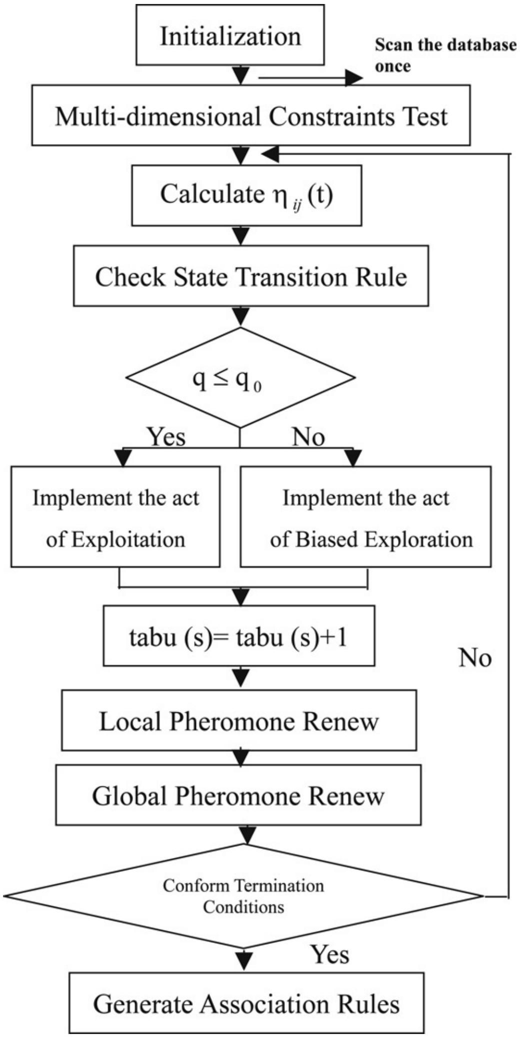

Initialization
Scan the database once
Multi-dimensional Constraints Test
Calculate η ij (t)
Check State Transition Rule
q ≤ q 0
Yes
No
Implement the act of Exploitation
Implement the act of Biased Exploration
tabu (s)= tabu (s)+1
No
Local Pheromone Renew
Global Pheromone Renew
Conform Termination Conditions
Yes
Generate Association Rules

Fig. 7. The flowchart of the proposed method.

$$\Delta\tau_{ij}^{k}(t) = \begin{cases} \dfrac{1}{L_{gb}}, & \text{if the } k\text{th ant in its tour is global-high-frequency} \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

Then, we calculate

$$\tau_{ij}(t+1) = [(1-\alpha)\cdot\tau_{ij}(t)] + \alpha\cdot\Delta\tau_{ij}. \tag{14}$$

7. Set $t = t + 1$ and $NC = NC + 1$, and then repeat Steps 1–5 until the termination iteration is met.
8. According to the mining results generate the association rules.

The flows and steps of the algorithm are shown in Fig. 7.

4. Model evaluation results and discussion

This section demonstrates how the proposed algorithm works by using the National Health Insurance Research Database of the Taiwan government to find disease association rules. Then, the results from an expert questionnaire are employed to demonstrate the reliability of the mining rules. They are also compared with the Apriori method.

4.1. Experimental setup

The National Health Insurance Plan of Taiwan Government has accumulated 12 million administrative and claims data, which is the largest database in the world. To rapidly and effectively respond to current and emerging health issues, The NHRI (National Health Research Institutes) cooperates with the National Health Insurance Bureau (NHIB) to establish a Nation Health Insurance research database. The NHRI will safeguard the privacy and confidentiality of subjects and routinely transfers the health insurance data from the NHIB to enable health researchers to analyze and improve the health of Taiwan's citizens.

The data used the systematic sampling method to randomly sample a representative database from the entire database. The size of the subset from each month is determined by the ratio of the amount of data in each month to that of the entire year. Then a systematic sampling is performed for each month to randomly choose a representative subset. This sampling database is obtained by combining the subsets for 12 months. The sampling database of the disease was 0.2% to the entire database respectively.

In a medical database, the most complete and detailed information are anamnesis data which contain disease name, prescription, patient's detail information, etc. Using this we aim to find the association rules between diseases, and between diseases and patients' information; and also to detect fake cases by data mining technology. This process should be able to increase medical quality, and decrease the cost and waste of medical resources.

In this research, the ACS algorithm is used find some hidden relationships among disease items in western medicine databases. There are 126,942 raw records in the medical database from 2001. The raw data which are incomplete and non-representative are deleted, leaving only 66,286 records. This study is based in part on data from the National Health Insurance Research Database provided by the Bureau of National Health Insurance, Department of Health and managed by National Health Research Institutes. The interpretation and conclusions contained herein do not represent those of Bureau of National Health Insurance, Department of Health or National Health Research Institutes.

In the data preparation stage, because this study is concerned with disease relationships, we must delete data in disease column whose value is invalid. Then the constraints are added to strengthen the data preparation stage, and make the mining process more efficient.

There are thirty-seven columns in the original medicine database, but this study is only concerned with the relationships of disease, so some columns in the database must be deleted. The remaining columns are "outpatient services", "outpatient services date", "patient's birthday", "international classification disease number 1–3 ICD code", and "patient's sex". It is necessary to normalize the ICD code before implementation, to a length of five digits. Those items whose codes are not five digits have a zero added for consistency. The experiment process flow is shown in Fig. 8.

According to the computational results for the proposed mining algorithm, the number of the association rules is much fewer than the conventional Apriori method. There are only ten association rules found and the number of the corresponding data is 585 when the pheromone is equal to 15. This data subset is 0.8825% of the total input data. To increase the number of rules, it is necessary to lower the pheromone threshold, although this may reduce the reliability since the data in the Medical Database are dispersed. Table 1 lists the extracted association rules.

Using the same method to implement the data in 2002 almost the same results are found. This reveals two important issues. The first is that our mining process is feasible and the second is that there are a lot of similar data in the CD-ROM of the National Health Insurance Research Database.

To determine the best pheromone value, sensitivity analysis is conducted, and according to Table 2, the pheromone setting of 15 is the best choice. If the pheromone is set to be 10, the association rules do not increase, but the data under the same rules is increased. This may cause many repeated data to appear and lower the efficiency of filtering the rules. On the other hand, if the pheromone is set to be 20, it will cause missing of some important mining rules. The missing rules are "Acute sinusitis → Acute bronchitis", "Acute tonsillitis → Acute bronchitis", and "Menopausal syndrome → Osteoporosis". These three rules are all important rules according to common sense. This claim can also be found from the expert questionnaire results in Table 4. Therefore, if the pheromone is 20, it does not mine all the potential knowledge and critical disease relationships.

The parameters setup for Ant Colony System is $\alpha = 0.1$, $\beta = 2$, $\rho = 0.1$, $t = 0$, $q_0 = 0.9$, $\tau_0 = (n * L_{nn})^{-1}$, where L_{nn} is the tour length produced by the nearest neighbor heuristic [28], and $m = 10$.

4.2. Analysis

It is difficult to measure the performance for the association rules miner. But, this study compares the proposed

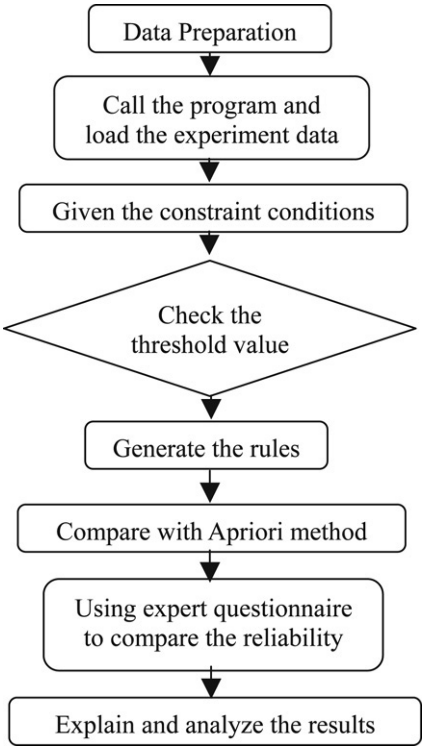

Data Preparation
Call the program and load the experiment data
Given the constraint conditions
Check the threshold value
Generate the rules
Compare with Apriori method
Using expert questionnaire to compare the reliability
Explain and analyze the results

Fig. 8. The experimental flowchart.

Table 1
The association rules of ACS algorithms

ICD CODE	Association rules	Pheromone
27240 → 41490	Hyperlipidemia → Ischemic heart disease	61.5
40190 → 25000	Hypertension → Diabetes mellitus	61.5
41490 → 40190	Ischemic heart disease → Hypertension	59.58
25000 → 53690	Diabetes mellitus → Functional gastrointestinal disorder	53.19
25000 → 78040	Diabetes mellitus → Dizziness and giddiness	52.82
25000 → 71590	Diabetes mellitus → Osteoarthritis	52.74
27490 → 40190	Gout → Hypertension	52.6
46190 → 46600	Acute sinusitis → Acute bronchitis	17.52
46300 → 46600	Acute tonsillitis → Acute bronchitis	17.52
62720 → 73300	Menopausal syndrome → Osteoporosis	15.2

Table 2
The sensitivity analysis of pheromone

Pheromone	10	15	20
The number of data	836	585	533
The number of rules	10	10	7

miner with the Apriori method under the same conditions. Thus, the base for demonstrating reliability is the expert questionnaire results.

Apriori which is coded by “Clementine” belonging to SPSS is used to discover association patterns by giving parameters of support = 0.3% and confidence = 20%, and the resulting top 19 association rules based on support are listed in Table 3. We select support = 0.3% because the data in this database are dispersed and various. Using “Clementine” can find 235 association rules and 13,685 data, which are 10.78% of the total input data. If the mining

Table 3
Some portion of the association rules using Apriori method

ICD CODE	Association rules	Support	Confidence
40190 → 25000	Hypertension → Diabetes mellitus	1.63	20
27240 → 41490	Hyperlipidemia → Ischemic heart disease	1.58	25
41490 → 40190	Ischemic heart disease → Hypertension	1.45	30
46190 → 46600	Acute sinusitis → Acute bronchitis	1.28	25
25000 → 71590	Diabetes mellitus → Osteoarthritis	1.15	28
62720 → 73300	Menopausal syndrome → Osteoporosis	0.89	39
36690 → 37200	Cataract → Acute conjunctivitis	0.81	46
25000 → 53390	Diabetes mellitus → Peptic ulcer	0.75	30
46300 → 46600	Acute tonsillitis → Acute bronchitis	0.73	20
36610 → 37210	Senile cataract → Chronic conjunctivitis	0.71	21
46190 → 47790	Acute sinusitis → Allergic rhinitis	0.71	33
49390 → 46600	Asthma → Acute bronchitis	0.7	31
46600 → 46590	Acute bronchitis → Upper respiratory infection	0.68	23
46600 → 46000	Acute bronchitis → Common cold	0.68	24
49390 → 47790	Asthma → Allergic rhinitis	0.65	35
46600 → 78060	Acute bronchitis → Fever	0.61	40
46600 → 53550	Acute bronchitis → Gastritis	0.6	28
46190 → 49390	Acute sinusitis → Asthma	0.57	25
47790 → 47390	Allergic rhinitis → Chronic sinusitis	0.57	20

results want to increase the rules, it must lower the support and the confidence threshold, although this may result in lower reliability. The reason to select only nineteen rules is that the supports for the other rules are all very small, say smaller than 0.3%. Of these nineteen association rules, there are seven rules which can also be found in the proposed method's result. Another three rules are found under support = 0.3%.

The three missing rules are "Diabetes mellitus → Functional gastrointestinal disorder", "Diabetes mellitus → Dizziness and giddiness", and "Gout → Hypertension". However, these three rules have higher pheromones, and based on common sense they should be extracted from the database. Therefore, it is considered that there may be missing values.

In order to show the reliability of the mining rules for these two methods, this study used the mining results to form an expert questionnaire. Interviewing 38 graduating students from China Medical University in Taiwan, the survey results are listed in Table 4.

There are three levels in the expert questionnaire. The level scale over 75% means the rule can be confirmed. The level about 50% means the rule may have some hidden knowledge in it (the probability of the rule occurring is 50%). The level less than 25% means that the rule has lower probability of occurring. Cronbach's α is equal to 0.9824, which means the reliability of the questionnaire is very high. And the validity of the questionnaire is based on expert's sense, so the validity is also confirmed.

Three classifications can be differentiated from the results of the questionnaire. First, the scale over 75% represents that the disease appears in clinical diagnosis. There are about 11 rules which conforms this result, and of these 11 rules, 7 are extracted from the proposed algorithm. And then the scale about 50% represents that the disease may or may not have appeared in clinical diagnosis and they are not found out. There are about 6 rules which conform to this result. And among these 6 rules, 3 rules are extracted from the proposed algorithm. Finally, the scale less than 25% represents the probability that the disease appears is very low. The rules may be wrong or not important. There are about 5 rules which conform to this result. But among these 5 rules, no rules are extracted by the proposed algorithm. This can prove that the proposed method is more able to find the accurate rules compared to the Apriori method.

According to the results of expert questionnaires, the reliability of the proposed method is slightly higher than Apriori. Besides, the proposed method has to scan less database than the Apriori method. This can save data scanning time in the process of generating rules. This makes the proposed method more efficient and valid. This finding is presented in Table 5. The proposed algorithm is more reliable than Apriori, and the computational time of the proposed method is much shorter than that of Apriori.

The algorithm structure can be discussed in two ways. In order to generate candidate itemsets, Apriori must scan multiple passes over the database, which requires a lot of time. And this method has to calculate a rule twice to obtain

Table 4
The results of expert questionnaire

Association rules	Over 75%	About 50%	Less than 25%
Ischemic heart disease → Hypertension	38	0	0
Hyperlipidemia → Ischemic heart disease	38	0	0
Hypertension → Diabetes mellitus	37	1	0
Menopausal syndrome → Osteoporosis	37	1	0
Diabetes mellitus → Osteoarthritis	35	3	0
Diabetes mellitus → Dizziness	35	3	0
Diabetes mellitus → Functional gastrointestinal disorder	30	8	0
Diabetes mellitus → Peptic ulcer	30	8	0
Senile cataract → Chronic conjunctivitis	29	5	4
Cataract → Acute conjunctivitis	27	8	3
Asthma → Acute bronchitis	25	11	2
Asthma → Allergic rhinitis	7	31	0
Acute sinusitis → Acute bronchitis	8	25	5
Gout → Hypertension	5	30	3
Acute bronchitis → Upper respiratory infection	4	30	4
Acute bronchitis → Common cold	4	30	4
Acute tonsillitis → Acute bronchitis	4	28	6
Acute bronchitis → Fever	2	20	16
Acute sinusitis → Asthma	0	17	21
Allergic rhinitis → Chronic sinusitis	0	8	30
Acute bronchitis → Gastritis	0	6	32
Acute sinusitis → Allergic rhinitis	0	6	32

Table 5
Comparison of efficiency and reliability between Apriori and ACS

Item	Apriori	ACS
Computational time	Eight hours and five min.	One hour and fifty-five min.
Generated rules	235	10
The corresponding data about the rules	13,685	585
The reliability of rules	Lower	Higher (refer to Table 3)
Threshold value	Support = 0.3%; Confidence = 20%	Pheromone = 15
Times for scanning the database	Several times	Less than twice

the confidence value (the confidence $A \rightarrow B \neq B \rightarrow A$). In contrast, ACS uses the pheromone to decide the next item, so it only scans the database once. If constraints are applied, it scans the database below once (refer to Fig. 7). The other advantage for using constraints is that the rules users want can be obtained within the constraints. This is closer to practical applications.

4.3. Scenario analysis

Here we assume some scenarios to simulate and use the program to implement the constraints. The study selects age and sex factors as the constraints to mine the relationships between the diseases. The scenario is designed for ten-year intervals and the mining results are presented in Table 6. It can be seen that the rules of mining in each ten-year level are different. The most frequently occurring rules for ages from zero to forty are “Acute tonsillitis → Acute bronchitis (46300 → 46600)” and “Acute sinusitis → Acute bronchitis (46190 → 46600)”. However, the most frequently occurring rule for ages forty to fifty is “Menopausal syndrome → Osteoporosis (62720 → 73300)”. For ages from fifty to sixty, the most frequently occurred rule are “Hyperlipidemia → Ischemic heart disease (27240 → 41490)” and “Hypertension → Diabetes mellitus (40190 → 25000)”. The most frequently occurring rule for ages sixty to seventy is “Diabetes mellitus → Functional gastrointestinal disorder (25000 → 53690)”, while the most frequently occurring rule for ages seventy to eighty is “Diabetes mellitus → Osteoarthritis (25000 → 71590)”. And the most frequently occurring rule for ages over eighty is “Hypertension → Diabetes mellitus (40190 → 25000)”.

Table 6
The mining results under the constraints of age

Age	Item		
	The rule of the most occur frequently	The number of data	Computational time (min)
0–10	Acute tonsillitis → Acute bronchitis & Acute sinusitis → Acute bronchitis	4164	12
10–20	Acute tonsillitis → Acute bronchitis & Acute sinusitis → Acute bronchitis	1850	3
20–30	Acute tonsillitis → Acute bronchitis & Acute sinusitis → Acute bronchitis	2942	5
30–40	Acute tonsillitis → Acute bronchitis & Acute sinusitis → Acute bronchitis	4144	12
40–50	Menopausal syndrome → Osteoporosis	6713	15
50–60	Hyperlipidemia → Ischemic heart disease & Hypertension → Diabetes mellitus	8981	20
60–70	Diabetes mellitus → Functional gastrointestinal disorder	11,298	23
70–80	Diabetes mellitus → Osteoarthritis	13,972	25
80 or greater	Hypertension → Diabetes mellitus	6098	15

By comparing Table 1 with Table 6 reveals that the rules are similar in the two tables. This indicates that the mining results are not influenced under different constraints. However, if there are some constraint conditions considered, the computational time can be reduced dramatically. The number of data in every age level can explain the distribution of the pheromone in every rule. For instance, the older age level has many more data records, which is the reason why the rules "Hyperlipidemia → Ischemic heart disease" and "Hypertension → Diabetes mellitus" have the highest pheromone values, as shown in Table 1.

The next step is to use Apriori to under the constraints condition. The constraint is set up as ages from seventy to eighty in order to run the Clementine given parameters of support = 0.3% and confidence = 20%. The reason to select this range is that the number of data for ages from seventy to eighty is the largest. There are 48 rules extracted and the corresponding number of data is 844. However, most of the rules are wrong or have low rates of appearance. For instance, the rules "Chronic obstructive pulmonary disease → Diabetes mellitus (49600 → 25000)", "Dysuria → Hypertension (78810 → 40190)", and "Diabetes mellitus → Upper respiratory infection (25000 → 46590)" are not correct based on experts' opinions. Furthermore, the computational time of ACS is still much faster than that of Apriori under the constraints. This is because Apriori must spend more time to filter the rules than ACS. In addition, the quality of rules generated by ACS is much better and contract than that of Apriori.

The Apriori method has totally different performance with and without constraints. The computational time for Apriori with constraints is much less, although this may result in losing some useful rules under the constraints. But, these missing rules are for other ages. Furthermore, if finding the rules for older people is the main objective, setting up the constraint can generate more feasible rules instead of the common rules. The case shows that the constraints are very important factors for extracting the rules in this study. By using the constraints to shorten the computational time and avoid losing rules, the results can be more useful and more efficient. Also, integrating constraints with mining techniques, the mining results can be much closer to the real world situation.

5. Conclusions

In this century, previously unknown diseases, like SARS, have created disasters for humanity. These may result from human beings' carelessness or environmental harm. Thus, developing a decision support system for medical workers becomes a very critical issue for patient treatments and extracting the important relationships or association rules between diseases is especially critical. This not only can save the medical costs, but also improve our health. This study has demonstrated that a novel approach, ACS, is able to mine the association rules from a health database, since it can deal with the discovery of hidden knowledge from the database. The results present some interacting relationships among the disease items. The proposed algorithm is much better than the Apriori both in efficiency and reliability according to expert questionnaire survey results. Since the proposed ACS scans the database only once, there can be a tremendous savings in computational time. This is especially important for large databases. In addition, the proposed method allows the user to define the search constraints, which makes the extracted rules more appropriate to users' needs and the computational speed will be faster.

Although this study has yielded very promising results, there are still some issues that to be resolved. In the data preparation stage, the proposed method uses constraint conditions to reduce the searching time. It is suggested to use another method to deal with the raw data, such as clustering methods. On the other hand, in the mining results, there are many similar rules to be generated, so it may feasible to apply another technology, like Fuzzy theory, to merge the similar rules together.

Acknowledgements

This study was financially supported by the National Science Council of Taiwan Government under contract number: NSC93-2213-E-027-015. Its support is greatly appreciated.

References

[1] R. Srikant, Q. Vu, R. Agrawal, Mining association rules with item constraints, in: Proc. Int. Conf. Knowledge Discovery and Data, KDD'97, Boston, MA, 1997, pp. 67–73.

[2] J.-F. Boulicaut, B. Jeudy, Using constraint for set mining: Should we prune or not? in: Proceedings Bases de Données Avançées, BDA'00, France, Oct. 2000, pp. 221–237.

[3] J. Pei, J. Han, Can we push more constraints into frequent pattern mining? in: Proc. Int. Conf. Knowledge Discovery and Data Mining, KDD'00, Boston, MA, Aug. 2000, pp. 350–354.

[4] J. Pei, J. Han, L.V.S. Lakshmanan, Mining frequent itemsets with convertible constraints, in: Proc. IEEE Int. Conf. Data Engineering, ICDE'01, Heidelberg, Germany, Feb. 2001, pp. 433–442.

[5] A. Colorni, M. Dorigo, V. Maniezzo, M. Trubian, Ant system for job-shop scheduling, Journal of Operations Research, Statistics and Computer Science (34) (1994) 39–53 (in Belgian).

[6] L.M. Gambardella, M. Dorigo, Solving symmetric and asymmetric TSPs by ant colonies, in: Proceedings of the IEEE Conference on Evolutionary Computation, ICEC'96, IEEE Press, New York, 1996, pp. 622–627.

[7] M. Dorigo, V. Maniezzo, A. Colorni, Positive feedback as a search strategy, Technical Report Dissertation, Dipartimento di Elettronica, Politecnico di Milano, Italy, 1991, pp. 91–116.

[8] L.M. Gambardella, M. Dorigo, Ant-Q: A reinforcement learning approach to the traveling salesman problem, in: Proceedings of the 12th International Conference on Machine Learning, ML-95, Morgan Kaufmann, Palo Alto, CA, 1995, pp. 252–260.

[9] B. Bullnheimer, R.F. Hartl, C. Strauss, A new rank-based version of the ant system a computational study, Technical Report POM-03/97, Institute of Management Science, University of Vienna, Austria, 1997, pp. 25–38.

[10] M. Dorigo, L.M. Gambardella, Ant colony system: A cooperative learning approach to the traveling salesman problem, IEEE Transactions on Evolutionary Computation (1) (1997) 53–66.

[11] T. Stuttzle, H.H. Hoos, The max–min ant system and local search for the traveling salesman problem, in: Proceedings of the 1997 IEEE International Conference on Evolutionary Computation, ICEC'97, IEEE Press, Piscataway, NJ, 1997, pp. 309–314.

[12] M. Dorigo, F. Glover, G.D. Caro, The Ant Colony Optimization Meta-Heuristic, in: New Ideas in Optimization, McGraw-Hill, 1999, pp. 11–32.

[13] B. Bonabeau, M. Dorigo, G. Thraulaz, Swarm Intelligence: From Natural to Artificial Systems, Oxford University Press, England, 1999.

[14] R. Peacock Peter, Data mining in marketing: Part 1, Marketing Management (1998) 9–18.

[15] R. Agrawal, T. Imielinski, A. Swami, Mining association rules between sets of items in large databases, in: Proc. ACM-SIGMOD Int. Conf. Management of Data, SIGMOD'93, Washington, USA, May 1993, pp. 207–216.

[16] R. Agrawal, T. Imielinski, A. Swami, Database mining: A performance perspective, (Learning and discovery in knowledge-based databases), IEEE Transactions on Knowledge and Data Engineering 5 (6) (1993) 914–925 (Special issue).

[17] J. Han, J. Pei, Mining frequent patterns by pattern-growth: Methodology and implications, ACM SIGKDD Explorations Newsletter, 2000, pp. 14–20.

[18] R. Agrawal, J.C. Shafer, Parallel mining of association rules, IEEE Transactions on Knowledge and Data Engineering 8 (6) (1996) 962–969.

[19] H. Toivonen, Sampling large databases for association rules, in: Proceedings of the International Conference on Very Large Data Bases, Mumbai (Bombay), India, 1996, pp. 134–145.

[20] M.J. Zaki, Scalable algorithms for association mining, IEEE Transactions on Knowledge and Data Engineering 12 (3) (2000) 372–390.

[21] D. Lin, Z. Kedem, Pincer-Search: An efficient algorithm for discovering the maximum frequent set, IEEE Transactions on Knowledge and Data Engineering 14 (3) (2002) 553–566.

[22] R.J. Bayardo, Efficiently mining long patterns from database, in: Proceedings of the ACM SIGMOD International Conference on Management of Data, Washington, USA, 1998, pp. 85–93.

[23] A. Savasere, E. Omiecinski, S. Navathe, An efficient algorithm for mining associate rules in large databases, in: Proceedings of the International Conference on Very Large Data Bases, Zurich, Switzerland, 1995, pp. 432–444.

[24] S. Rafael Parpinelli, S. Heitor Lopes, A. Alex Freitas, An ant colony based system for data mining: Applications to medical data, in: Proc. Int. Conf. Knowledge Discovery and Data, Boston, MA, 2000, pp. 55–62.

[25] B. Wu, Z. Shi, A clustering algorithm based on swarm intelligence, in: Proceedings of International Conference on Info-tech and Info-net, vol. 3, Beijing, Oct. 2001, pp. 58–66.

[26] W.M. Teles, L. Weigang, C.G. Ralha, AntWeb — the adaptive web server based on the ants' behavior, in: Proceedings of IEEE/WIC International Conference on Wed Intelligence, Oct. 2003, pp. 558–561.

[27] B.D. Su, Discovering association rules through ant systems, Master Thesis of National Chin-Hwa Univeristy, Taiwan, ROC, 2002.

[28] D.J. Rosenkrantz, R.E. Stearns, P.M. Lewis, An analysis of several heuristics for the traveling salesman problem, SIAM Journal of Computing 6 (3) (1997) 563–581.