Association Rule Mining from Large and Heterogeneous Databases with Uncertain Data using Genetic Network Programming

Eloy Gonzales* and Koji Zettsu*
* *Information Services Platform Laboratory*
Universal Communication Research Institute
National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0289, Japan
Tel: +81-774-98-6866, Fax: +81-774-98-6960
e-mail: {egonzales, zettsu}@nict.go.jp

***Abstract*—Association Rule Mining is one of the most important tasks in data mining and it has been deeply studied during last years. Recently several rule mining algorithms have been developed due to many real-world applications. Most of these studies have generally considered only precise data, which means that items within each datum or transaction are definitely known and precise. However, there are also many real life situations where the data is uncertain, which means that items are expressed in terms of existential probabilities. In this paper, a method for association rule mining from large, heterogeneous and uncertain databases is proposed using an evolutionary method named Genetic Network Programming (GNP). Some other association rule mining methods can not handle uncertain data directly, they are inapplicable or computational inefficient under such a model. GNP uses direct graph structure and is able to extract rules without generating frequent itemsets to improve mining efficiency. The performance of the method is evaluated through extensive experiments using real scientific large-scale heterogeneous databases that show its effectiveness and efficiency.**

***Keywords*-Association rule mining; heterogeneous databases; uncertain data; evolutionary computation.**

I. Introduction

The continuously growing in the size and number of databases in a variety of domains has boosted the develop of several data mining methods during the last decade. There is an increasing need to discover associations and relations among large and heterogeneous databases, which may be tackled by association rule mining. Actually, several association rule mining algorithms have been proposed. Most of them assume a data model, which *transactions* capture the doubtless facts about items contained in each transaction, that is, they handle *precise* data, such as databases of market basket transactions, web logs and click streams where the user definitely knows whether an item is present in, or is absent from, a transaction in the databases. However, in many other applications, the existence of an item in a transaction is best captured by a likelihood measure or probability, for example, a medical dataset may contain a list of patients as records (tuples) and for each record a set of symptoms or illnesses that a patient suffers as the items. Applying association analysis on such dataset allows to discover any potential correlation among the symptoms and illnesses, a physician may highly suspect (but cannot guarantee) that a patient suffers some disease. The uncertainty of such suspicion can be expressed in terms of *existential probability*. Other examples of uncertain datasets are pattern recognition databases where image processing techniques are applied on a picture to extract features that indicate the presence or absence of certain objects in an area, but due to noises and limited resolution, the presence of an object is ofter uncertain and expressed as probability.

Many of the developed algorithms for uncertain mining have been focused on data mining tasks like clustering and classification of uncertain data [1]. With respect to association rule mining of uncertain data, Chui et al.[2] proposed an Apriori-based algorithm called *U-Apriori* and introduced a trimming strategy to reduce the number of candidates that need to be counted by the algorithm. To speed up the mining process, they also proposed a decremental pruning technique, however as an Apriori-based algorithm, U-Apriori relies on the candidate generate-and-test paradigm. Kai-San Leung et al. [3] have tried to reduce the searching space by adding constraints given by users, but the scalability issues have not been described.

In this paper, a method for mining association rules from uncertain data is proposed using an evolutionary optimization algorithm named Genetic Network Programming (GNP). There have been some proposals of association rule mining using GNP [4][5], however all of them use certain data. The advantages of the proposed method are as follows: (1). It is widely known that the search space of frequent patterns from precise data is very huge, and from uncertain data is even much bigger. Thus, the proposed method extracts rules directly without generating the frequent patterns. (2). The support and confidence are the most used framework to evaluate the association rules. However, this measurements are not longer valid for probabilistic datasets. GNP provides the flexibility to use any correlation measure either independently or combined. Thus, in this paper *hyper-lift* and *hyper-confidence* proposed in [6] are used. (3). The scalability issue is not an important concern in other algorithms since most of them deal with mining frequent patterns, which is computationally expensive and therefore use relatively small datasets. In this paper, large-scale and heterogeneous databases are mainly focused.

The following sections of this paper are organized as follows: In Section II, the uncertain data model is presented, the concepts and explanations of general association rules are introduced in Section III, the outline of GNP is briefly reviewed in Section IV where also the method for rule extraction from uncertain data is presented. Simulation results are described in Section V, and finally, conclusion and future work are given in Section VI.

II. Uncertain Data Model

Because of the uncertainty in various real-life situations, users may not be certain about the presence or absence of an item x in a transaction t_i. They may suspect the presence of x in t_i, but cannot guarantee. The uncertainty of such suspicion can be expressed in terms of *existential probability* $P(x, t_i)$, which indicates the likelihood of x being present in t_i in a probabilistic database D of uncertain data. The existential probability $P(x, t_i)$ ranges from a positive value close to 0, which indicates that x has an insignificantly low chance to be present in D, to a value of 1, which indicates that x is definitely present. Applying this notion to the traditional databases of precise data, each item of any transaction can be viewed as an item with a 100% likelihood of being present in such a transaction.

III. Association Rules

A transaction database consist of a series of transactions, each of them containing a subset of available items[7]. Let $I = \{A_1, A_2, \ldots A_l\}$ be a set of attributes. Let G be a set of transactions,where each transaction T is a set of attributes such that $T \subseteq I$. Associated with each transaction is a unique identifier whose set is called TID. A transaction T contains X, a set of some attributes in I, if $X \subseteq I$. An association rule is an implication of the form of $X \Rightarrow Y$, where $X \subset I, Y \subset I$, and $X \cap Y = \emptyset$. X is called antecedent and Y is called consequent of the rule. Both are called **itemsets**. In general, an itemset is a non-empty subset of I. There are some assumptions in our model, 1) transactions occur randomly following a homogeneous Poisson process. 2) all items occur independently of each other and for each item exist a probability of being contained in a transaction.

Looking at the observed co-occurrences counts for all pairs of two items A_i and A_j, in a dataset with N transactions, it is possible to form an $n \times n$ contingency table. Each cell can be modeled by a random variable C_{ij}, which given fixed marginal counts c_i and c_j, follows a *hyper-geometric distribution*. In the case of two itemsets X and Y, the random variable C_{XY} follows a hyper-geometric distribution with the counts of the itemsets as its parameter [6], that is, the probability of counting exactly k transactions, which contain the two independent itemsets X and Y is given by:

$$P(C_{XY} = k) = \frac{\binom{C_X}{k}\binom{N-C_X}{C_Y-k}}{\binom{N}{C_Y}} \tag{1}$$

The probability of counting more than k transactions is:

$$P(C_{XY} > k) = 1 - \sum_{i=0}^{k} P(C_{XY} = i) \tag{2}$$

The contingency table is shown in Table I.

Table I
The contingency table of X and Y.

	Y	$\neg Y$	$\sum_{row}$
X	C_{XY}	$C_X - C_{XY}$	C_X
$\neg X$	$C_Y - C_{XY}$	$(N - C_Y) - (C_X - C_{XY})$	$N - C_X$
$\sum_{col}$	C_Y	$N - C_Y$	N

(N: the number of transactions $(= |TID|)$)

A. Hyper-Lift

The expected value of a random variable C_{XY} for the co-occurrence counts for two itemsets X and Y is:

$$E(C_{XY}) = \frac{C_X C_Y}{N} \tag{3}$$

Therefore, lift can be written as:

$$\textit{lift}(X \Rightarrow Y) = \frac{C_{XY}}{E(C_{XY})} \tag{4}$$

However, it works well for items with a relatively high occurrence frequency. Thus, for relatively infrequent itemsets the hyper-lift is defined as:

$$\textit{hyper-lift}_\delta(X \Rightarrow Y) = \frac{C_{XY}}{Q_\delta(C_{XY})} \tag{5}$$

where, $Q_\delta(C_{XY})$ is the quantile distribution. The minimal value of the δ quantile of the distribution of C_{XY} is defined by the following inequalities:

$$\begin{aligned} &P(C_{XY} < Q_\delta(C_{XY})) \leq \delta \ \text{, and} \\ &P(C_{XY} > Q_\delta(C_{XY})) \leq 1 - \delta \end{aligned} \tag{6}$$

B. Hyper-confidence

The hyper-confidence is defined directly by the probability of realizing a count smaller that the observed co-occurrence count c_{XY} given the marginal counts c_X and c_Y as follows:

$$\begin{aligned} \textit{hyper-confidence}(X \Rightarrow Y) &= P(C_{XY} < c_{XY}) \\ &= \sum_{i=0}^{c_{XY}-1} P(C_{XY} = i) \end{aligned} \tag{7}$$

where $P(C_{XY} = i)$ is calculated using Equation 1.

Note that hyper-confidence is equivalent to a special case of Fisher's exact test, the one-sided test on 2 × 2 contingency tables. In this case, the p-value is directly obtained from the hyper-geometric distribution, which is computationally negligible compared to the effort of counting support and finding frequent itemsets. Furthermore, for the application of mining association rules where only rules with positively correlated elements are of interest, a one-side test as used here is much more appropriate.

Therefore, the problem of mining probabilistic association rules from uncertain data is to find all rules that are highly likely to be interesting, that is, satisfying the minimum hyper-confidence threshold.

$$\begin{aligned} &\textit{hyper-confidence}(X \Rightarrow Y) \geq min_{hyper-conf} \text{ , and} \\ &\qquad \textit{hyper-lift}(X \Rightarrow Y) \geq 1 \end{aligned} \tag{8}$$

IV. Genetic Network Programming

Genetic Network Programming (GNP) is one of the evolutionary optimization algorithms, which evolves directed graph structures as solutions instead of strings (Genetic Algorithms) or trees (Genetic Programming) [8], [9], [10]. The main aim of developing GNP was to deal with dynamic environments efficiently by using the higher expression ability of graph structures.

The basic structure of GNP is shown in Fig. 1. The graph structure is composed of three types of nodes that are connected on a network structure: a start node, judgment nodes (diamonds), and processing nodes (circles). Judgment nodes are the set of J_1, J_2, $\ldots$, J_p, which work as *if-then* conditional decision functions and they return judgment results for assigned inputs and determine the next node to be executed. Processing nodes are the set of P_1, P_2, $\ldots$, P_q, which work as action/processing functions. The start node determines the first node to be executed. The nodes transition begins from the start node, however there are no terminal nodes. After the start node is executed, the next node is determined according to the node's connections and judgment results.

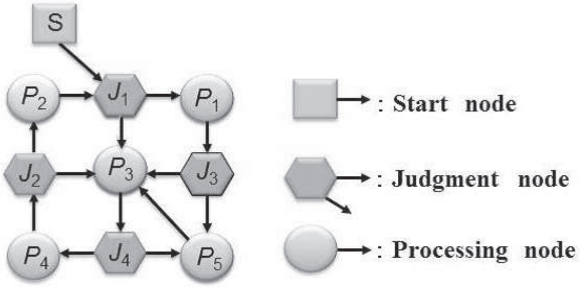
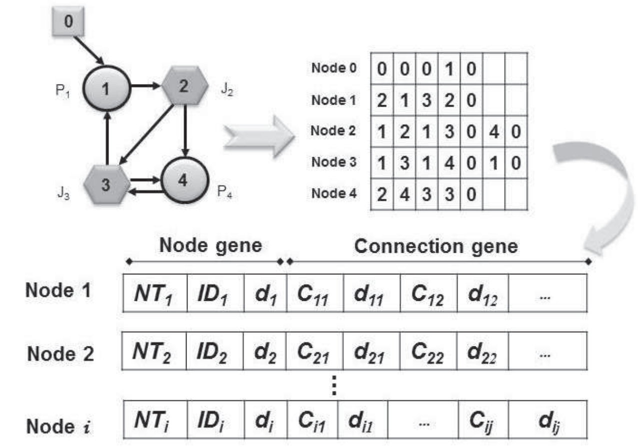


Figure 1. Basic structure of GNP

The gene structure of GNP (node i) is shown in Fig. 2. The set of these genes represents the genotype of GNP-individuals. NT_i describes the node type, $NT_i = 0$ when node i is the start node, $NT_i = 1$ when node i is a judgment node and $NT_i = 2$ when node i is a processing node. ID_i is an identification number, for example, $NT_i = 1$ and $ID_i = 1$ mean node i is J_1. C_{i1}, C_{i2}, $\ldots$, denote the nodes, which are connected from node i firstly, secondly, $\ldots$, and so on depending on the arguments of node i. d_i and d_{ij} are the delay time, which are the time required to execute the judgment or processing of node i and the delay time of transition from node i to node j, respectively.



Node 0	0	0	0	1	0		
Node 1	2	1	3	2	0		
Node 2	1	2	1	3	0	4	0
Node 3	1	3	1	4	0	1	0
Node 4	2	4	3	3	0		

Node 1	NT_1	ID_1	d_1	C_{11}	d_{11}	C_{12}	d_{12}	...
Node 2	NT_2	ID_2	d_2	C_{21}	d_{21}	C_{22}	d_{22}	...
⋮								
Node i	NT_i	ID_i	d_i	C_{i1}	d_{i1}	...	C_{ij}	d_{ij}

NT_i : node type (Start node=0; Judgment node=1; Processing node=2)
ID_i : identification number; d_i, d_{ij}: delay time; C_{ij} : connected node

Figure 2. Gene structure of GNP (node i)

In this paper, the execution time delay d_i and the transition time delay d_{ij} are not considered. All GNP-individuals in a population have the same number of nodes.

The characteristics of GNP are described as follows. (1) The judgment and processing nodes are repeatedly used in GNP, therefore the structure becomes compact and an efficient evolution of GNP is obtained. (2) Since the number of nodes is defined in advance, GNP can find the solutions of the problems without bloating, which can be sometimes found in Genetic Programming (GP). (3) Nodes that are not used at the current program execution will be used for future evolution. (4) GNP is able to cope with partially observable Markov processes. (5) The node transition in GNP individual is executed according to its node connections without any terminal nodes.

In the conventional GNP-based mining method, the attributes of the database correspond to the judgment nodes in GNP. Association rules are represented by the connections of nodes. Candidate rules are obtained by genetic operations. Rule extraction using GNP is done without identifying frequent itemsets used in Apriori-like methods [11]. Therefore, this method extracts important rules sufficient enough for user's purpose in a short time. The association rules extracted are stored in a pool through generations. The fundamental difference with other evolutionary methods is that GNP evolves in order to store new interesting rules in the pool, not to obtain the individual with the highest fitness value. GNP method has also advantages over other evolutionary methods such as Genetic Algorithms (GA) and Genetic Programming (GP). For GA-based methods [12], there are limitations in the number of association rules extracted because they are represented in individuals. In GP-base methods [13], an individual is usually represented by a tree with attribute values in the functions (e.g., logical, relational or mathematical operators) of the internal nodes. An individual's tree can grow in size and shape in a very dynamical way making it very difficult to understand for real applications.

A. GNP for rule extraction in a uncertain database

In this section, a general association rule mining method for uncertain databases is proposed using GNP. Let A_i be an attribute in an uncertain database and its value an existential probability. Each attribute A_i is associated with a_i, which is a threshold value. One of the features of the proposed method is to evolve the threshold a_i along with the evolution of GNP in order to obtain as many rules as possible [14]. The initial threshold a_i is determined as follows: (1). The mean μ_i and standard deviation σ_i of every attribute A_i is calculated. (2). The initial threshold is selected randomly from the interval $[\mu_i - \alpha_i\sigma_i, \mu_i + \alpha_i\sigma_i]$, where α_i is a parameter that determines the size of the interval. Then, the initial threshold is evolved by mutation in every generation of GNP. Once the threshold a_i is selected for all attributes, each value of the attribute A_i is checked to verify whether it is greater than the threshold a_i in the judgment nodes of the GNP individuals. The evolution of the thresholds is carried out by introducing an additional parameter that determines the mutation rate t_r. In this paper, the mutation rate t_r is gradually adjusted as it is described in [14].

1) Rule Representation: Attributes and its values correspond to the functions of judgment nodes in GNP. Association rules are represented as the connections of nodes .

Fig. 3 shows a sample of the connection of nodes in GNP for probabilistic association rule mining. P_1 is a processing node and is a starting point of association rules. "$A_1 \geq a_1$", "$A_2 \geq a_2$", "$A_3 \geq a_3$" and "$A_4 \geq a_4$" in Fig. 3 denote the functions of judgment nodes. Association rules are represented by the connections of these nodes, for example, $(A_1 \geq a_1) \Rightarrow (A_2 \geq a_2)$, $(A_1 \geq a_1) \wedge (A_2 \geq a_2) \Rightarrow (A_3 \geq a_3)$, $(A_1 \geq a_1) \wedge (A_2 \geq a_2) \wedge (A_3 \geq a_3) \Rightarrow (A_4 \geq a_4)$ and $(A_1 \geq a_1) \wedge (A_2 \geq a_2) \Rightarrow (A_3 \geq a_3) \wedge (A_4 \geq a_4)$.

Judgment nodes in GNP are used to examine the attribute values of database tuples and processing nodes calculate the measurements of association rules. Judgment nodes determine the next node by a judgment result. Each judgment node has two connections Continue-side and Skip-side. The Continue-side of the judgment node is connected to another judgment node. Skip-side of the judgment node is connected to the next numbered processing node. If the attribute value is greater or equal to a_i, then move to the Continue-side. If the attribute value is less than a_i, then the transition goes for the Skip-side.

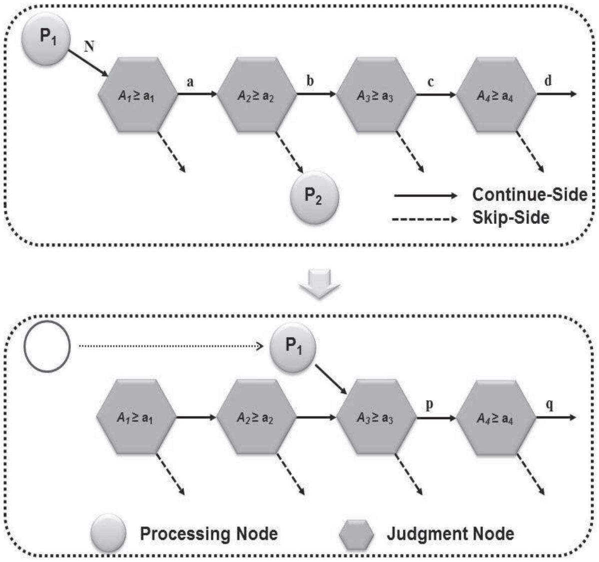


Figure 3. A connection of nodes in GNP for probabilistic association rule mining

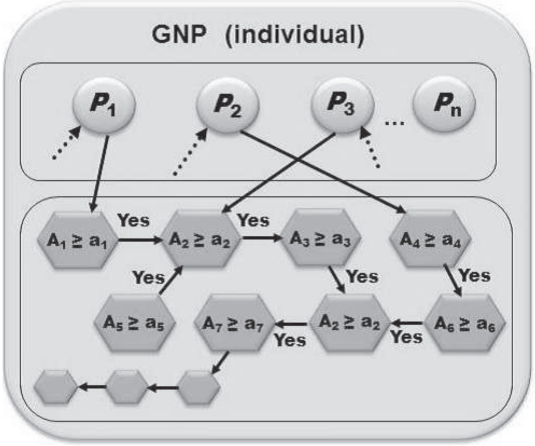


Figure 4. Basic structure of GNP for uncertain association rule mining

A basic structure of GNP-individual for association rule mining is shown in Fig. 4. In Fig. 4, the Skip-side of judgment nodes is abbreviated. Each processing node has an inherent numeric order (P_1, P_2, ..., P_s) and is connected to a judgment node. Start node connects to P_1. For each judgment node, the examinations of attribute values start and in case to move to the Continue-side continuously, the connection is obligatorily transfered to the next processing node using the Skip-node when the maximum number of attributes ($MaxLength$) in the rule is reached. When the examination of the attribute values of tuple $TID = 1$ from the starting point P_s ends, then GNP examines the next tuple $TID = 2$ from P_1 likewise. Therefore, all tuples in the database are examined.

2) Rule Measurements: In GNP the number of tuples moving to the Continue-side are counted up and they are used for calculation of the measurements In upper side of Fig. 3, a, b, c and d are the number of tuples moving to the Continue-side at each judgment node when the attribute values are greater or equal to a_1, a_2, a_3 and a_4, respectively.

In the proposed method the significance of the associations are measured via the hyper-geometric distribution used in classical statistics. For example in lower side of Fig. 3 it is possible to calculate the number of tuples going to the Continue-Side of A_3 and $A_3 \wedge A_4$ if the connection of node P_1 is changed from node $A_1 \geq a_1$ to node $A_3 \geq a_3$. This procedure is repeated like a chain operation in each generation. The extracted important association rules are stored in a local pool all together through generations. When an important rule is extracted by GNP, the redundancy of the attributes is checked and it is also checked whether the important rule is new or not, that is, whether the rule is already in the local pool or not.

3) Genetic Operations: Changing an attribute to another one or adding some attributes in the rules would be considered as candidates of important rules. These rules can be obtained effectively by GNP genetic operations, because mutation and crossover will change the connections or contents of the nodes.

Three kinds of genetic operators are used for judgment nodes: GNP-crossover, GNP-mutation-1 (change the connections) and GNP-mutation-2 (change the function of nodes).

- GNP-Crossover: uniform crossover is used. Judgment nodes are selected as the crossover nodes with the probability of P_c. Two parents exchange the gene of the corresponding crossover nodes.
- GNP-Mutation-1: Mutation-1 operator affects one individual. The connection of the judgment nodes is changed randomly by mutation rate of P_{m1}.
- GNP-Mutation-2: Mutation-2 operator also affects one individual. This operator changes the functions of the judgment nodes by a given mutation rate P_{m2}.

On the other hand, all the connections of the processing nodes are changed randomly. At each generation, all GNP-individuals are replaced with the new ones by the following criteria: The GNP-individuals are ranked by their fitness values and the best one-third GNP-individuals are selected. After that, these GNP-individuals are reproduced three times for the next generation using the genetic operators described before.

If the probabilities of crossover (P_c) and mutation (P_{m1}, P_{m2}) are set at small values, then the same rules in the pool may be extracted repeatedly and GNP tends to converge prematurely at an early stage. If the probability of mutation is set at high values, then some genetic characteristics of the individuals might be lost. These parameter values are chosen experimentally avoiding these issues.

4) Heterogeneity Level: The heterogeneity level of rule r, $hl(r)$, is defined as follows:

$$hl(r) = \frac{\prod_{k}^{T}[na_k(r)/NA_k]}{T}; \; k = 1, 2, \ldots, T \tag{9}$$

where,

$na_k(r)$ is the number of attributes in rule r (antecedent and consequent), which belong to database k.

NA_k is the number of attributes of database k.

T is the number of heterogeneous databases.

The heterogeneity level of rule r measures the ratio of attributes that exist in the rules, which belong to one or another database. $hl(r) \geq \gamma$, where γ is a threshold value for the heterogeneity level. It ensures that every rule contains at least one attribute per every heterogeneous database. γ is defined experimentally and its value decreases when the number of databases increases.

5) Fitness of GNP: The number of processing nodes and judgment nodes in each GNP-individual is determined based on experimentation depending on the number of attributes processed. All GNP-individuals in a population have the same number of nodes. The connections of the nodes and the functions of the judgment nodes at an initial generation are determined randomly for each GNP-individual.

Fitness of GNP is defined by:

$$\begin{aligned} F = \sum_{r \in Q} \{ &\textit{hc(r)} + \alpha_{new}(r) + hl(r)(NA_A(r) - 1) + \\ &hl(r)(NA_C(r) - 1)\} \end{aligned} \tag{10}$$

The terms in Eq. (10) are as follows:

Q: set of suffixes of extracted important association rules satisfying (8)

hc(r): value of *hyper-confidence(r)* of rule r

$\alpha_{new}(r)$: additional constant defined by

$$\alpha_{new}(r) = \begin{cases} \alpha_{new} & \text{(rule } r \text{ is new)} \\ 0 & \text{(rule } r \text{ has been already extracted)} \end{cases} \tag{11}$$

$hl(r)$: heterogeneity level of rule r.

$NA_A(r)$: the number of attributes in the antecedent of rule r.

$NA_C(r)$: the number of attributes in the consequent of rule r.

Constant $\alpha_{new}(r)$ in Eq. 10 is defined empirically based on the values of *hyper-confidence(r)*. Thus, $\alpha_{new}(r) = 0.3$.

$NA_A(r) \leq MaxLength$ and $NA_C(r) \leq MaxLength$. $MaxLength = 2T + 1$, where T is the number of heterogeneous databases.

hc(r), $NA_A(r)$ and $NA_C(r)$, and $\alpha_{new}(r)$ are concerned with the importance, complexity and novelty of rule r, respectively. The fitness represents the potential to extract new rules.

V. Simulation Results

In order to test and validate the effectiveness of the proposed method, two real-time scientific databases from UCI ML Repository [15] and World Data System (WDS) [16] were taken to conduct the experiments, which are frequently used in data mining community. Both of them contains heterogeneous spatial-temporal data and they are suitable for mining general association rules. The first one ("A" dataset) is El Nino dataset and contains oceanographic and surface meteorological readings taken from a series of buoys positioned throughout the equatorial Pacific. The second one ("B" dataset) correspond to the weather information of the Pacific Ocean taken by sensors of World Ocean Circulation Experiment (WOCE).

A. Experiment Setting

Both datasets are combined taken into account the date and each attribute is separated into two corresponding attributes according to their values. For instance, if $Latitude \leq 0$ correspond to the $Latitude = South$. In this experiment, data only from one year (1993) is considered. Thus, one large database is generated (36 attributes × 20609 records), then the data is normalized between the interval [0, 1] and these values are used as existential probabilities.

1) Parameters of GNP: The population size of GNP is 120. The number of processing nodes and judgment nodes in each GNP individual are 10 and 75, respectively. The maximum number of changing the connections of the processing nodes (*MaxLenght*) in each generation is $2(2) + 1 = 5$. The conditions of crossover and mutation are $P_c = 1/5$, $P_{m1} = 1/3$ and $P_{m2} = 1/5$. The termination condition is 100 generations. 10 runs were executed and the results are given as an average. These parameters were selected through experimentation. All algorithms were coded in Java language. Experiments were performed on a 3.2GHz Intel Xeon PC with 12G of main memory, running Windows 7 Ultimate 64bits.

Table II shows some examples of the rules extracted by GNP. The termination "A" or "B" of each attribute means the correspondence to its dataset. From Table II, the rules extracted by GNP are simple due to the small number of itemsets, which contribute to their understandability. Although the GNP-based data mining method extracts significant number of rules in a short period of time, it does not extract all the possible rules. Instead, it extracts rules with higher quality as it is shown in Table II.

Fig. 5 shows the number of extracted rules according to the number of generations using the complete database and $min_{hyper-conf} \geq 0.9$. It can be seen that most of the association rules are extracted at earlier generations becoming stable at later generations, which improves the performance of the method.

Fig. 6 shows the number of extracted rules for different values of minimum hyper-confidence. Fig. 6 shows that when the minimum hyper-confidence increases, the number of association rules decreases because the conditions become more strict and fewer rules are able to satisfy them.

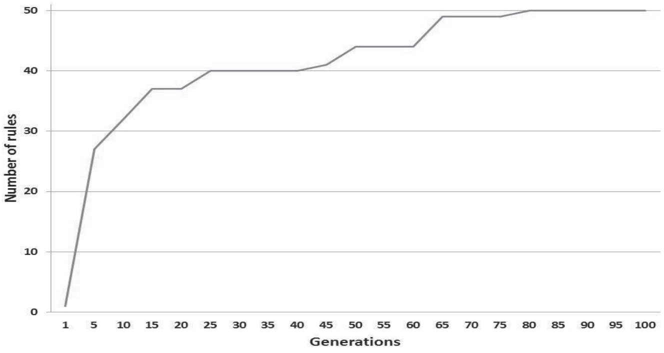


Figure 5. Number of extracted rules vs. number of generations

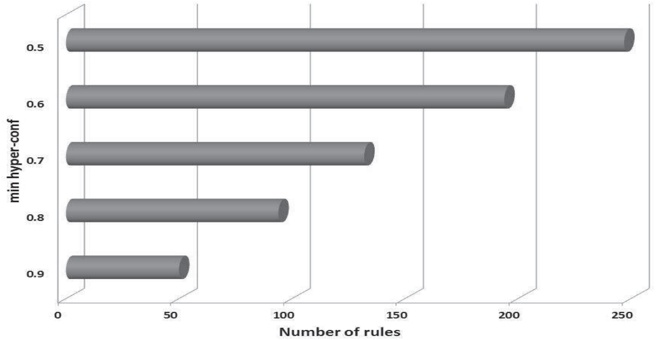


Figure 6. Number of extracted rules vs. min hyper-confidence

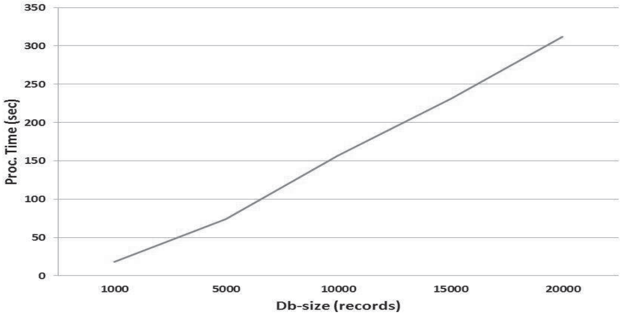


Figure 7. Processing Time vs. database size

Fig. 7 shows the processing time for extraction of association rules when the database size varies. Fig. 7 shows that the processing time increases linearly when the database size increases.

Fig. 8 shows the processing time for extraction of association for different values of hyper-confidence. Fig. 8 shows that the processing time does not vary significantly when hyper-confidence changes.

Table II
EXAMPLES OF RULES EXTRACTED BY GNP. GENERATIONS=100, $min_{hyper-conf} \geq 0.9$

Association Rules	*Hyper-Conf.*
IF Sea_Surf_Temp = High_A ∧ Latitude = South_B, THEN Longitude = West_B ∧ Rel_Hum = High_B ∧ Precip = High_B	1.0
IF Longitude=West_A ∧ Zon_Winds=West_A ∧ Humidity=Low_A, THEN Precip = High_B ∧ Temp_Water = Low_B	0.9871
IF Temp_Air=High_A ∧ Speed=High_B, THEN Meridional_Winds= South_A ∧ Rel_Hum = High_B	0.9962
IF Pressure_Atm=High_B ∧ Temp_Air=Low_A ∧ Sea_Surf_Temp = High_A, THEN Longitude = West_B ∧ Temp_Water = High_B	1.0
IF Temp_Air=Low_B ∧ Zon_Winds=West_A ∧ Latitude=South_B, THEN Rel_Hum = High_B ∧ Precip = High_B	1.0

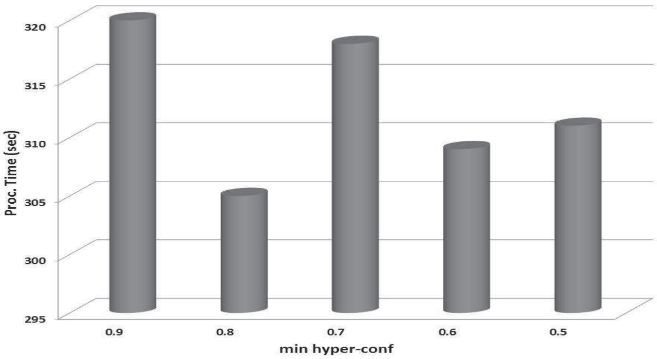


Figure 8. Processing Time vs. min hyper-confidence

VI. CONCLUSION AND FUTURE WORK

A method for association rule mining from uncertain databases has been proposed using GNP. An uncertain database includes the existential probability of every item in a transaction. Traditional approaches count the frequency of the itemsets. The proposed method can extract directly important rules without calculating the frequency and the conditions of important association rules can be flexibly defined by users. The performance of the rule extraction has been evaluated using real data sets. The results shows that the proposed method has the potential to realize associations considering heterogeneous databases and may be applied for rule discovery from probabilistic databases in several other fields. For future work, the method will be extended to deal with large and heterogeneous scientific databases combined with web data.

REFERENCES

[1] R. Cheng et al., “Probabilistic verifiers: evaluating constrained nearest-neighbor queries over uncertain data”. In *Proc. of the IEEE ICDE 2008*, pp. 973-982, 2008.

[2] C.K. Chui, B. Kao, and E. Hung, “Mining Frequent Itemsets from Uncertain Data”. In *Proc. of the 11th Pacific-Asia Conference on Knowledge Discovery and Data Mining, PAKDD 2007*, pp. 47-58, 2007.

[3] C.K.S. Leung and D. A. Brajczuk, “Efficient Algorithms for the Mining of Constrained Frequent Patterns from Uncertain Data”, *SIGKDD Explorations*, Vol.11, Issue 2, pp. 123-130, 2009.

[4] K. Shimada, K. Hirasawa, and T. Furuzuki, “Genetic Network Programming with Acquisition Mechanisms of Association Rules”, *Journal of Advanced Computational Intelligence and Intelligent Informatics*, Vol. 10, No. 1, pp. 102-111, 2006.

[5] E. Gonzales, K. Taboada, K. Shimada, S. Mabu, and K. Hirasawa, “Combination of Two Evolutionary Methods for Mining Association Rules in Large and Dense Databases”, *Journal of Advanced Computational Intelligence and Intelligent Informatics*, Vol.13, No.5, pp. 561-572, 2009.

[6] M. Hahsler and K. Hornik. “New probabilistic interest measures for association rules” in *Journal of Intelligent Data Analysis*, Vol. 11, No. 5, pp.437-455, 2007.

[7] C. Zhang and S. Zhang, *Association Rule Mining: models and algorithms*, Springer, 2002.

[8] S. Mabu, K. Hirasawa, and J. Hu, “A Graph-Based Evolutionary Algorithm: Genetic Network Programming (GNP) and Its Extension Using Reinforcement Learning“, Evolutionary Computation, *MIT Press* , Vol 15, No. 3, pp. 369-398, 2007.

[9] K. Hirasawa, T. Eguchi, J. Zhou, L. Yu, J. Hu, and S. Markon, “A Double-deck Elevator Group Supervisory Control System using Genetic Network Programming”,*IEEE Trans. on System, Man and Cybernetics, Part C*, Vol.38, No.4, pp. 535-550, 2008.

[10] T. Eguchi, K. Hirasawa, J. Hu, and N. Ota, “A study of Evolutionary Multiagent Models Based on Symbiosis”,*IEEE Trans. on System, Man and Cybernetics, Part B*, Vol.36, No.1, pp. 179-193, 2006.

[11] R. Agrawal and R. Srikant, “Fast Algorithms for Mining Association Rules”, in *Proc. of the 20th VLDB Conf.*, pp. 487-499, 1994.

[12] C.Z. Janikow, “A knowledge-intensive genetic algorithm for supervised learning”, *Machine Learning 13*, pp. 189-228, 1993.

[13] C.C. Bojarczuk, H.S. Lopes, and A.A. Freitas, “Genetic programming for knowledge discovery in chest pain diagnosis”, *IEEE Trans. on Engineering in Medicine and Biology Magazine*, Vol. 19, No.4, pp. 38-44, 2000.

[14] K. Taboada, E. Gonzales, K. Shimada, S. Mabu, K. Hirasawa, and J. Hu, “Association Rule Mining for Continuous Attributes using Genetic Network Programming”, *IEEJ Transactions on Electrical and Electronic Engineering*, Vol. 3, No. 2, pp. 199-211 March 2008.

[15] Frank, A. Asuncion, A. (2010). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science. [Last Access: Jun 14th, 2011]

[16] Walden, B; WOCE Surface Meteorology Data, WOCEMET (2006): Continuous meteorological surface measurement during KNORR cruise 316N138_12. Woods Hole Oceanographic Institution, Physical Oceanography Department.