A DATA mining Procedure Using Neural Network- Self Organization Map and Rough Set to Discover Association Rules

S. Wesley Changchien and Tzu-Chuen Lu

Department of Information Management
Chaoyang University of Technology, Taichung, Taiwan, R.O.C
E-mail: {swc, s8814602}@cyut.edu.tw

Abstract

Modern enterprises are usually equipped with database and data warehouse for data storage and analysis, but they are of less use without insight and intellectual analysis. Data mining techniques are popular and often used to find knowledge from enterprise data store and thus support to make better decisions. Data mining by automatic or semiautomatic exploration and analysis on large amount of data item set in a database can discover significant patterns and rules underlying the database. However, for large non-homogeneous data item set, direct extraction of rules with traditional data mining techniques may be useless or meaningless.

The purpose of the paper is to propose a data mining procedure for database with large amount of transaction records. The data mining procedure consists of two methods. One is a neural network, Self-Organization Map (SOM), which performs an affinity grouping tasks on large amount of database records. The other is rough set theory, which extracts association rules for each homogeneous cluster of data records. An implemented system was applied to a sample from purchase records of a database for illustration.

Keywords: Data mining, neural network, SOM, Rough set, association rules

1. Introduction

In today's competitive business marketing, there is a number of reasons of utilizing data mining tools: ever-increasing amount of data, business reengineering and organizational decentralization, faster product cycles and globalization and enterprise topologies, etc. [7]. Enterprise managers usually use data mining tools to improve their services and make better decisions. Which customer is loyal? What Products are usually purchased together? What is the next product or service this customer will acquire? These questions interest managers very much. In general, OLTP (On-Line Transaction Process) and OLAP (On-line Analytical Processing) are often used to answer these questions. These tools although provide ways of analyzing data following the rules defined by users, but do not creatively derive empirical rules and conclude themselves. Therefore it is of little use for managers without intelligence. Intelligence allows us to devise rules, creating new ideas, and make predictions for the enterprise [4,7,8].

Data mining techniques can perform automatic or semiautomatic exploration and analysis of large quantity of data in order to discover meaningful patterns or rules. Data mining takes two forms: verification-driven data mining, which is a top-down approach that attempts to substantiate or disprove preconceived ideas, and discovery-driven data mining, which extracts information automatically from data [8,16]. Both data mining forms are to achieve some of the data mining tasks including finding association rules, clustering, classification and estimation, etc. [1,2,3,5,6,14,19,21,22]. When the managers try to understand more about their customers, their intuition is to separate the customers into smaller groups, each of which can be explained more specifically. In marketing terms, subdividing the population according to variables already known to be good discriminators is called "Clustering" [8]. There are numerous of clustering algorithms, one of which is Self-Organizing Map (SOM), a neural network that can recognize unknown patterns in the data [11,16]. When training data sets are fed into the network, SOM will compute and come up with a winner node. The network will adjust winner node and neighborhood weights accordingly. Training continues until it converges. Then SOM will then be capable of determining which cluster the new input belongs to.

Clustering techniques can separate data items into clusters of items, other methods then can be applied to figure out the underlying features for each cluster in terms of association rules. Knowledge is usually represented in the form of rules – rules deducting the degree of association between two variables, rules mapping data into predefined classes, rules identifying a finite set of categories or clusters to describe the data, etc. Mining association rules has attracted a significant amount of researchers [9,12,13,17,18,20]. Finding patterns in a database is the fundamental operation behind most of the common data mining tasks, including mining of association rules and sequential patterns.

Rough set theory is used for approximation of a concept that uses lower and upper sets of the concepts. When we inspect the data mining queries with respect to the rough set theory, dependency analysis and classification of data items are well investigated. The associations between values of an attribute can easily be solved by the rough set theory [10,15]. Pawlak proposed rough set theory in 1982. This theory is an extension of set theory for the study of intelligent systems with incomplete information [16]. Let U be a finite, nonempty set called the universe, and let I be an equivalence relation on U, called an indiscernibility relation. I(x) is an equivalence class of the relation I containing element x. The indiscernibility relation is meant to capture the consequence of inablility to discern in view of the available information. There are two basic operations on sets in the rough set theory, the I-lower and the I-upper approximations, defined respectively as follows:

$$I_*(X) = \{x \in U : I(x) \subseteq X\} \qquad (1)$$

$$I^*(X) = \{x \in U : I(x) \cap U \neq f\} \qquad (2)$$

Usually in order to define a set we use a confidence function. The confidence function is defined as follows:

$$CF(x) = \frac{Num(X \cap I(x))}{Num(I(x))} \quad \text{where} \quad CF(x) \in [0,1], \qquad (3)$$

Where $Num(X \cap I(x))$ is the number of objects that occur simultaneously in X and $I(x)$, and $Num(I(x))$ is the number of objects in $I(x)$.

Confidence function can be used to redefine equations (1) and (2) as follows:

$$I_*(X) = \{x \in U : CF(x) = 1\} \qquad (4)$$

$$I^*(X) = \{x \in U : CF(x) > 0\} \qquad (5)$$

The value of the confidence function denotes the degrees of how the element x belongs to the set X in view of the indiscernibility relation I.

The purpose of this paper is to discover association rules from a database with large number of data records. Before discovering association rules we first preprocess our database records into clusters. This paper proposes a procedure by applying a neural network SOM first for clustering, and then using rough set theory to discover association rules.

2. Proposed Data Mining Procedure

Data mining is an interactive and iterative process involving numerous steps. Fayyad [4] outlines a practical view of data mining process. Figure 1 outlines the data mining steps. Following the data mining steps, we propose a procedure of data mining for large amount of data set as follows.

2.1 Step 1-Selection and Sampling

Database consists of current detail data, history data, summarized data and metadata, etc. Step 1 selects target tables, dimensions, attributes, and records for data mining. Step 1 consists of four activities: creating fact table, selecting cause and result dimensions, selecting dimension attributes, and filtering data.

2.1.1 Creating fact table

There are a lot of tables in enterprise's database. The fact tables that interest the managers are created according to the mining purpose. For example, there may be vendor, product, sale, and customer tables in a database (Figure 2). User then selects dimensions from the fact table next.

2.1.2 Selecting dimensions

When managers want to analyze their company's general association rules, they can choose dimensions from the fact table. Our implemented system will analyze a certain dimension and find out the relationships among the attributes. For example, a manager can select sale table to find what relationships the attributes have in between.

2.1.3 Selecting dimension attributes

Sale table has multiple attributes. But not all of those attributes fit into the analysis and attributes are of different levels of importance to the manager. Therefore our implemented system allows user to choose some of the attributes and set different weights to selected attributes.

2.1.4 Filtering data

In order to get the right data, user can retrieve data with constraints such as the range of an attribute. This step also involves in removal of noises or handling of missing data fields. User can also take random sample or select part of records from database for analysis. For example, if a manager selects sale dimension then he can select dimension attributes including customer number, product number, kind of product, department, price, quality, etc. According to the corresponding attribute importance levels, we set their weights. Below we suppose that a manager selects *DEPT* and *AMT* two attributes (Table 1), where att_j indicates attribute j, $j \in [1,m]$, and $wegt_j$ indicates weight j, $j \in [1,m]$, where m is the total number of attributes selected.

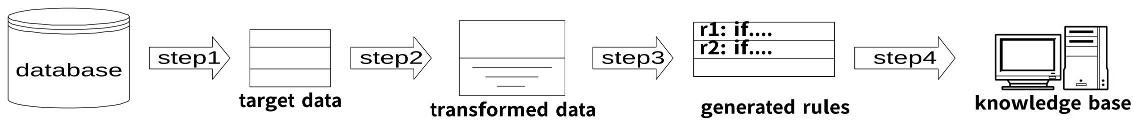

Figure 1. Data mining steps

Table 1. An example of selected attributes and weights

Attribute	AMT(att_1)	DEPT(att_2)
Weight	0.4 ($wegt_1$)	0.6 ($wegt_2$)

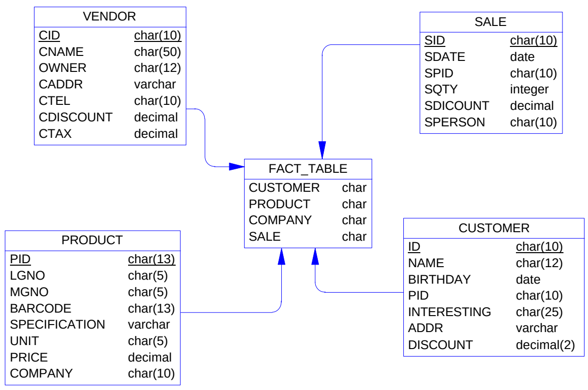

Figure 2. Fact table for a sales MIS

2.2 Step 2-Transformation and Normalization

Before proceeding data mining on data set we should normalize and scale data if it is necessary. Because this paper uses neural network for data mining preprocessing, we transfer data into values between 0 and 1. Here two different formulas can be used to transfer data.

(1) Numerical

If the attribute is numerical then use the following equation

$$response\_value_{jk} = \frac{(value_{jk} - \min(att_j))}{(\max(att_j) - \min(att_j))} * wegt_j, \quad (6)$$

where

$response\_value_{jk}$ is the normalized value for the j^{th} attribute of record k, $k \in [1, p]$,

$\min(att_j)$ is the minimum value of the j^{th} attribute,

$\max(att_j)$ is the maximum value of the j^{th} attribute, and

$val(att_{jk})$ is the original value of the j^{th} attribute of record k.

Since the attribute type of *AMT* in Table 1 is numerical, we can use equation (6) above for normalization.

(2) Non-numerical

If the attribute is non-numerical then user can design data transformation scale. For instance, attribute *DEPT* in Table 1 is not numerical. We thus need to transform it into normalized numerical response value. When all attributes have the same measurement and ranges, we then can proceed to the next data mining process.

2.3 Step 3 Data Mining Using SOM and Rough Set Theory

In this paper, we focus on clustering and association rule generation in data mining tasks. We use a neural network SOM for clustering. After the data are analyzed and clustered by SOM, rough set theory is used to discover association rules.

2.3.1 SOM

Kohonen proposed SOM in 1980. It is an unsupervised two-layer network that can recognize a topological map from a random starting point. By SOM we can cluster enterprise's customers, products, suppliers, etc. According to different clusters' characteristics, different marketing strategies may be adopted by making use of the corresponding discovered association rules. In SOM network, input nodes and output nodes are fully connected with each other. Each input node contributes to each output node with a weight. Figures 3 and 4 are the network structure and flow chart for SOM training procedure, respectively. In our developed system user can assign different numbers of output nodes (cluster number), learning rate, radius rate and converge error rate, etc. Users could also be compliant with default assignment of cluster number. In the simple example in Table 1, SOM has two input nodes: *DEPT* and *AMT*. Output nodes are set to be nine clusters. SOM network first determines the winning node using the same procedure as the competitive layer. Then the weight vectors for all neurons within a certain neighborhood of the winning neuron are updated. After the SOM network converges, we use those weights to split the data set in dimension table. According to the weights, data items can be assigned to their corresponding clusters.

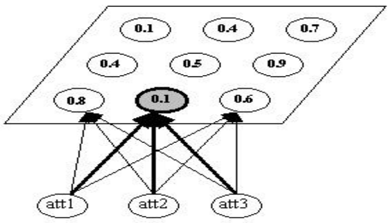

Figure 3. The neural network of SOM

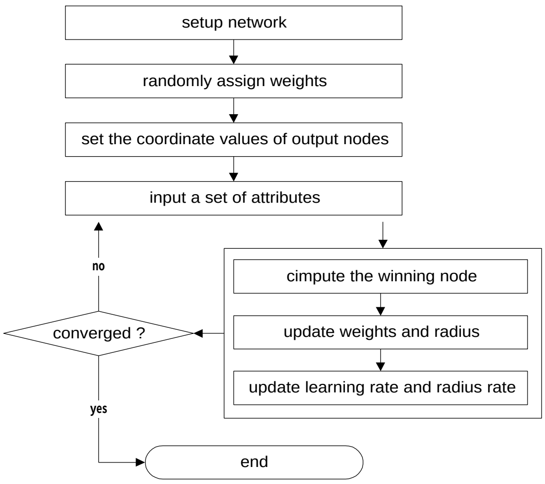

Figure 4. Flow chart for SOM training procedure

2.3.2 Rough set

Table 2 presents the records sampled from our purchase database. The attributes of the records were normalized and clustered, where *GID* is the cluster number and *TID* is the transaction identification number. Figure 5 shows the flow chart of implementing rough set. The steps are described as follows.

(1) Generate result equivalence classes

Let X_i denote the result equivalence class as a set of objects for cluster *i*. Table 3 lists some of the results equivalence classes.

(2) Generate cause equivalence classes

Let $Y_{i,j}$ denote the cause equivalence class, a set of objects, for a specific attribute *i* with value of *j*, $A_{i,j}$. Table 4 gives some examples of cause equivalence classes.

(3) Create lower approximation rules

Here equation (1) is used to create lower approximation rules. For example,

$GID = 1 \Rightarrow X_1 = \{Obj^{(2)}, Obj^{(4)}, Obj^{(5)}, Obj^{(9)}, Obj^{(10)}, Obj^{(19)}\}$

$A_{23*} = \{Obj^{(9)}\} \Rightarrow$ *"Insert this into the lower rule table: IF amt = 0.04 Then GID = 1."*

$A_{24*} = \{Obj^{(5)}\} \Rightarrow$ *"Insert this into the lower rule table: IF amt = 0.08 Then GID = 1."*

Delete objects of lower approximation rules from result equivalence and cause equivalence classes.

$A_{11*} = \{f\}$, …, $A_{23*} = \{Obj^{(9)}\}$

Delete $\{Obj^{(5)}, Obj^{(9)}\}$ from result equivalence class X_1 and cause equivalence class then we have:

$GID = 1 \Rightarrow X_1 = \{Obj^{(2)}, Obj^{(4)}, Obj^{(10)}, Obj^{(19)}\}$

$A_{11*} = \{f\}$,…, $A_{23*} = \{f\}$, $A_{24*} = \{f\}$ …

Since there are still some objects in X_i's, the next step is to find upper approximation rules.

(4) Create upper approximation rules and compute confidences

Let A_{ij}^* denote the set of objects that have the same attribute A_{ij} and some of the objects are the same as those in a cluster. Here equations (2) and (3) are applied to create upper approximation rules and to compute rules'

Table 2. Sample records and clustering results

ID	AMT (att_1)	DEPT (att_2)	GID (att_3)	TID	AMT (att_1)	DEPT (att_2)	GID (att_3)
1	0.02	0.3	2	11	1	0.6	9
2	0.22	0.6	1	12	0.79	0.1	8
3	0	0.3	2	13	0.65	0.1	8
4	0.12	0.6	1	14	0.44	0.1	6
5	0.08	0.6	1	15	0.12	0.1	3
6	0.14	0.1	3	16	0.14	0.4	2
7	0.67	0.1	8	17	0.44	0.2	6
8	0.39	0.4	7	18	0.18	0.1	3
9	0.04	0.6	1	19	0.18	0.6	1
10	0.12	0.6	1	…	…	…	…

Table 3. Examples of the results equivalence classes

GID	X_i
1	$\{Obj^{(2)}, Obj^{(4)}, Obj^{(5)}, Obj^{(9)}, Obj^{(10)}, Obj^{(19)}\}$
2	$\{Obj^{(1)}, Obj^{(3)}, Obj^{(16)}\}$
3	$\{Obj^{(6)}, Obj^{(15)}, Obj^{(18)}\}$
…	…

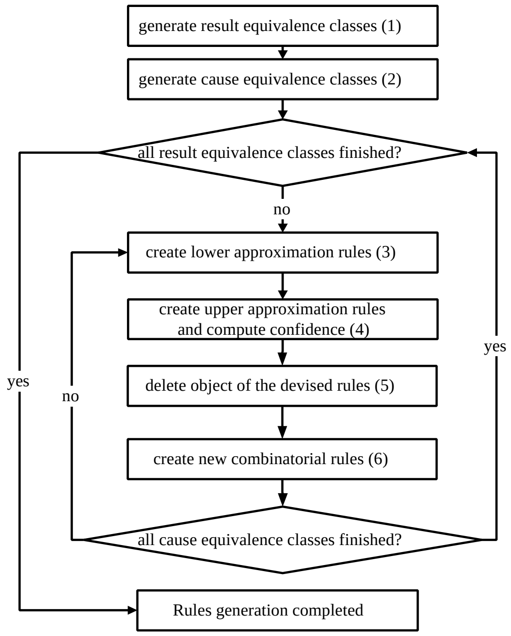

Figure 5. Flow chart of implementing Rough set

Table 4. Example of the cause equivalence classes

$A_{i,j}$		$Y_{i,j}$
i	j	
DEPT	0.1	$Y_{DEPT,0.1} = \{Obj^{(6)}, Obj^{(7)}, Obj^{(12)}, Obj^{(13)}, Obj^{(14)}, Obj^{(15)}, Obj^{(18)}\}$
DEPT	0.2	$Y_{DEPT,0.2} = \{Obj^{(17)}\}$
DEPT	0.3	$Y_{DEPT,0.3} = \{Obj^{(1)}, Obj^{(3)}\}$
DEPT	0.4	$Y_{DEPT,0.4} = \{Obj^{(8)}, Obj^{(16)}\}$
DEPT	0.6	$Y_{DEPT,0.6} = \{Obj^{(2)}, Obj^{(4)}, Obj^{(5)}, Obj^{(9)}, Obj^{(10)}, Obj^{(11)}\}$
AMT	0	$Y_{AMT,0} = \{Obj^{(3)}\}$
AMT	0.02	$Y_{AMT,0.02} = \{Obj^{(1)}\}$
AMT	0.04	$Y_{AMT,0.04} = \{Obj^{(9)}\}$
AMT	0.08	$Y_{AMT,0.08} = \{Obj^{(5)}\}$
AMT	0.12	$Y_{AMT,0.12} = \{Obj^{(4)}, Obj^{(10)}, Obj^{(15)}\}$
…	…	…

confidences.

(a) Create upper approximation rules

For example for

$GID=1 \Rightarrow X_1=\{Obj^{(2)},Obj^{(4)},Obj^{(10)},Obj^{(19)}\}$

$A_{11} \Rightarrow DEPT=0.1,\ A_{11}^*=\{f\},$

$A_{12} \Rightarrow DEPT=0.2,\ A_{12}^*=\{f\}, \ldots$

$A_{15} \Rightarrow DEPT=0.6,$

$A_{15}^*=\{Obj^{(2)},Obj^{(4)},Obj^{(5)},Obj^{(9)},Obj^{(10)},Obj^{(11)}\}, \ldots$

$A_{25} \Rightarrow AMT=0.12,\ A_{25}^*=\{Obj^{(4)}, Obj^{(10)},Obj^{(15)}\}, \ldots$

(b) Compute confidence for upper approximation rules

Use equation (3) to compute each rule's confidence.

Here we can assign a threshold value (minimum confidence) to determine which rules are acceptable.

For instance,

$$CF(A_{16}^*) = \left(\frac{Obj^{(2)}, Obj^{(4)}, Obj^{(10)}}{Obj^{(2)}, Obj^{(4)}, Obj^{(5)}, Obj^{(9)}, Obj^{(10)}, Obj^{(11)}}\right) = \frac{3}{6}$$

"Insert this into the upper rule table as R3: IF dept = 0.06 Then GID = 1 with CF= 1/2."

$$CF(A_{25}^*) = \left(\frac{Obj^{(4)}, Obj^{(10)}}{Obj^{(4)}, Obj^{(10)}, Obj^{(15)}}\right) = \frac{2}{3}$$

"Insert this into the upper rule table as R4: IF amt = 0.12 Then GID = 1 with CF=2/3."

(5) Create new combinatorial rules

Following the above procedure we can find single attribute association rules. Sometimes relationships exist between two or more attributes. In the following we consider two or more attributes to generate association rules. But how to combine and what attributes suits to be combined are quite challenging. To solve those questions, we apply a probability model for each cluster. Table 5 is the table of cluster hit ratios. Where

$p(GID_l)$: The probability of group l, $l \in [1, m]$,

$$p(GID_l) = \frac{no\_obj(GID_l)}{total\_\#\_of\_objects},$$

$no\_obj(GID_l)$: The number of objects in group l,

ATT_i: The attribute i.,

CLU_{ij}: For attribute i, the j^{th} cluster, $j \in [1, n]$,

$no\_obj(CLU_{ijl})$: The number of objects for attributes i, the j^{th} cluster, in group l,,

$HR(CLU_{ij})$: The hit ratio of attributes i, the j^{th} cluster.

$$HR(CLU_{ij}) = \sum_{l=1}^{m}\left[-\left(\frac{no\_obj(CLU_{ijl})}{total\_\#\_of\_objects}\right) * \log\left(\frac{no\_obj(CLU_{ijl})}{total\_\#\_of\_objects}\right) * p(GID_l)\right]$$

Table 5. Table of cluster hit ratios for attribute *I*

ATT_i	$p(GID_1)$	$p(GID_2)$	$\cdots$	$p(GID_m)$	Total
CLU_{i1}	$no\_obj(CLU_{i11})$	$no\_obj(CLU_{i12})$	$\cdots$	$no\_obj(CLU_{i1m})$	$HR(CLU_{i1})$
CLU_{i2}	$no\_obj(CLU_{i21})$	$no\_obj(CLU_{i22})$	$\cdots$	$no\_obj(CLU_{i2m})$	$HR(CLU_{i2})$
CLU_{i3}	$no\_obj(CLU_{i31})$	$no\_obj(CLU_{i32})$	$\cdots$	$no\_obj(CLU_{i3m})$	$HR(CLU_{i3})$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
CLU_{in}	$no\_obj(CLU_{in1})$	$no\_obj(CLU_{in2})$	$\dots$	$no\_obj(CLU_{inm})$	$HR(CLU_{in})$

Table 6. An example of hit ratio table for DEPT

Clusters	1	2	3	4	5	6	7	8	9	Hit ratio
	$13/102$	$0/102$	$24/102$	$2/102$	$20/102$	$15/102$	$13/102$	$12/102$	$12/102$	
0.1	13	0	0	2	0	0	9	0	0	0.027046
0.2	0	0	0	0	3	0	3	0	10	0.026206
0.3	0	0	0	0	9	1	1	1	2	0.028165
0.4	0	0	0	0	4	11	0	2	0	0.027139
0.5	0	0	0	0	4	3	0	0	0	0.017439
0.6	0	0	24	0	0	0	0	0	0	0.034790
Total hit ratio										0.16078

Table 6 is an example of cluster hit ratio table for attribute *DEPT*. Take on for example in Table 6, the hit ratio for cluster *DEPT=0.1* is calculated as follows:

$$HR(DEPT=0.1)=[-(\frac{13}{102})*\log(\frac{13}{102})*(\frac{13}{102})]+$$
$$[-(\frac{2}{102})*\log(\frac{2}{102})*(\frac{2}{102})]+[-(\frac{9}{102})*\log(\frac{9}{102})*(\frac{13}{102})]=0.02704$$

Attribute *DEPT* has a total hit ratio of 0.16078 that indicates the importance level of *DEPT*. The cluster *DEPT = 0.1* has an importance level of 0.027046. We can combine those attributes of various clusters with higher importance levels to find combinatorial rules. User can simply give a threshold of hit ratio to decide what attributes need to be combined for consideration.

After computing all the attributes of various clusters with higher hit ratios, we can combine upper approximation rules with higher hit ratio equivalence classes. For instance, in step 5 we have found an upper rule: $A_{25}^* \Rightarrow AMT=0.12, Y_{AMT,0.12}=\{Obj^{(4)},Obj^{(10)},Obj^{(15)}\}$

We combine them with all attributes which with higher hit ratios. For example, attribute *DEPT*, where *DEPT = 0.6* has higher hit ratios so we combine them. $A_{15} \Rightarrow DEPT=0.6, Y_{DEPT,0.6}=\{Obj^{(2)},Obj^{(4)},Obj^{(5)},Obj^{(9)},$
$Obj^{(10)},Obj^{(11)}\}$

$$COM\ (A_{25}^*, A_{15}^*) \Rightarrow (AMT\ =0.12)\cap(DEPT\ =0.6),$$

$$(Y_{AMT,0.12})\cap(Y_{DEPT,0.6})=\{Obj^{(4)},Obj^{(10)}\}$$

$$CF(COM(A_{25}^*, A_{15}^*))=(\frac{Obj^{(4)},Obj^{(10)}}{Obj^{(4)},Obj^{(10)}})=\frac{2}{2}$$

"Insert this into the variation rule table as R5: IF amt = 0.12 and dept=0.6 Then GID = 1 CF=2/2."

(6) Go back to (4) until all of the cause equivalence classes are completed.

(7) Go back to (3) until all of the result equivalence classes are completed.

2.4 Step 4-Knowledge Base

We summarize and store rules in lower rule table, upper rule table and variation rule table into knowledge base. Those rules can help managers make better decisions and find some unknown patterns in the database that are meaningful for them. Table 7 presents the rules found in the data mining example. For instance, rules 1 to 3 mean that if customers buy products from the store and the total amt levels are 0.04 or 0.08 or 0.22, respectively, then the purchase records or customers belong to cluster 1. These rules have 100 % confidence. Rules 17 and 18 mean that if customers buy products and the total amt level is 0.14 and the department is 0.4, respectively, then they belong

Table 7. Rules found in example

Group 1:	
R1: amt=0.04, CF= 1.00	R14: amt=0.12, CF= 0.67
R2: amt=0.08, CF= 1.00	R15: amt=0.18, CF= 0.50
R3: amt=0.22, CF= 1.00	R16: dept=0.60, CF= 0.43
Group 2:	
R4: amt=0.00, CF= 1.00	R17: amt=0.14, CF= 0.50
R5: amt=0.02, CF=1.00	R18: dept=0.40, CF= 0.50
R6: dept=0.3, CF=1.00	…
…	…

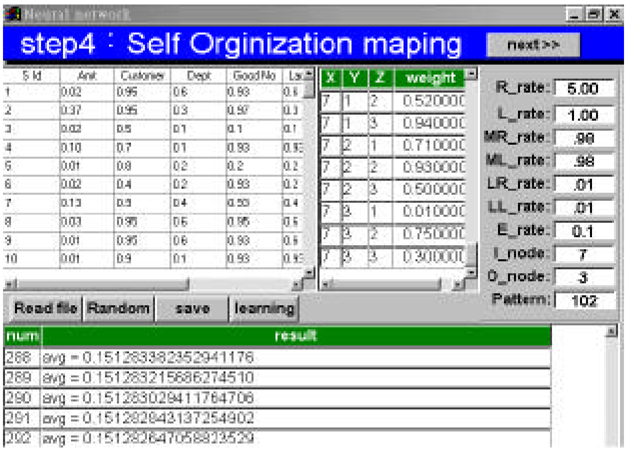

Figure 6. Self-Organization Mapping

to cluster 2. These rules have 0.5% confidence.

3. Results and Discussions

We follow the proposed data mining procedure and conduct data mining on purchase records of a store. Here we select one day of sales volume in a department store. There are 10,123 records in it and approximately memory is 1MB. First we choose sales table and amt, customer, department, product number (good_no), kind of products (Large_no), price and quality. And then we randomly sample one percept of all the 10,123 records from database and transform those data. After normalizing data we proceed to data mining step – SOM and rough set. Here we set radius equal to 5.0 and learning rate equal to 1.0. The change rate of radius and learning rate are both equal to 0.98. The lowest radius and learning rate are 0.01.When the error rate is lower than 0.1, network is then converged. Figure 6 shows the result of SOM. After the network converges we gain the final weights. Then we can compute the final score of each record with final weights. According to the final score we assign each record to its cluster. After SOM, we use rough set to generate association rules that explain what each cluster means. We create result equivalence classes, create lower approximation rules, create upper approximation rules, compute confidence, and create variation rules (Figure 7).

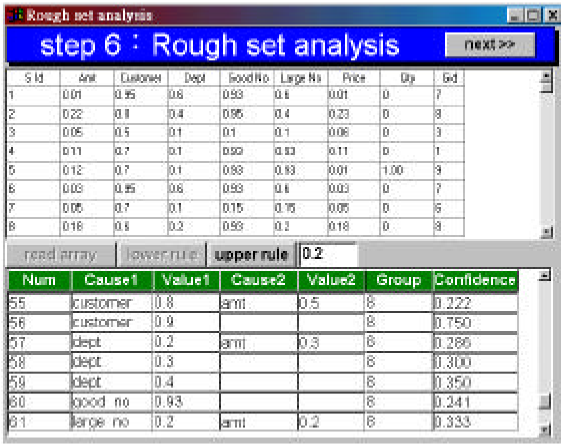

Figure 7. Rough set analysis

Here we set threshold equal to 0.2. We discovered 36 lower rule, 26 upper rule and 5 variation rule. For example, Lower rules: If amt = 0.24 or large_no = 0.1 or price = 0.24 then GID=3 (CF=1). Upper rules: If amt = 0.11 or customer =0.4 or price = 0.11 then GID=3 (CF>=0.5). Variation rules: If dept = 0.2 and amt = 0.3 then GID = 3 (CF=0.29).

Then we recuperate each attribute back to its original value. Total amount in range 0.24 means:

$$\frac{(value - \min(amt))}{(\max(amt) - \min(amt))} = 0.24$$, where max amount = 10,600 and min amount = 2. So current amount = $2,523.

We transfer upper rules to explain what cluster 3 means: total amount approximation is equal to 2,000,the kind of product is from 10 to 20, and the price of product approximation is equal to 1,000. Confidences of those characters equal 100%. And total amount approximation equal to 1,000, the kind of customer belongs to 4 and the price of product approximation is equal to 1,000. Confidences of those characters are more then 50%.

4. Conclusions

In this paper we use two methods to assist our proposed data mining procedure: a neural network –SOM and rough set to discover association rules. SOM clusters the transaction records for data mining preprocess purpose. Rough set theory is used to analyze what each cluster means and what relationships exist between attributes, so as to discovery enterprise rules. These rules can help managers make better decisions, for each cluster by employing corresponding strategy that realizes one by one on-line marketing. Using the proposed data mining system can help discover many of the enterprise rules. But there is still room for improvement. Future challenges include how to cluster customers and products meaningfully, and how to choose training set properly?

How to evaluate the rules' appropriately? Besides, in order to make system applicable, domain expertise is usually needed. Therefore, more intelligent systems will be the main focus of our future work.

5. REFERENCES

[1] Agrawal, R., Imielinski, T. and Swami, A., "Mining Association between Sets of Items in Massive Database," *In Proc. Of the ACM-SIGMOD Int'l Conf. On Management of Data*, pp. 207-216, 1993.

[2] Agrawal, R. and Srikant, R., "Fast algorithms for mining association rules," *In Proc. Of the International Conf. On Very Large Data Bases*, pp. 407-419, 1995.

[3] Apte, C. and Hong, S.J., "Predicting equity returns from securities data and minimal rule generation," *Advances in Knowledge Discovery (AAAI Press, Menlo Park, CA/ MIT Press, Cambridge, MA)*, pp. 541-560, 1995.

[4] Basu, A., "Theory and Methodology Perspectives on operations research in data and knowledge management," *European Journal of Operational Research*, Vol. 111, pp. 1-14, 1998.

[5] Bayardo, R.J., "Efficiently Mining Long Patterns from Databases," *In Proc. Of the ACM-SIGKDD Int'l Conf. On Management of Data*, pp. 85-93, 1998.

[6] Bayardo, R.J. and Agrawal, R., "Mining the Most Interesting Rules," *In Proc. Of the Fifth ACM-SIGKDD Int'l Conf. On Knowledge Discovery and Data Mining*, pp. 145-154, 1999.

[7] Bayardo, R.J., Agrawal, R. and Gunopulos, D., "Constriaint-Based Rule Mining in Large, Dense Databases," *In Proc. Of the 15th Int'l Conf. On Data Engineering*, pp. 188-197, 1999.

[8] Beery, J. A. and Linoff, G., *Data mining techniques: for marketing, sales, and customer support*, New York: Wiley, 1997.

[9] Chen, M.S., "Using Multi-Attribute Predicates for Mining Classification Rules," *IEEE*, pp. 636-641, 1998.

[10] Deogun, S., Raghavan, V., Sarkar, A. and Sever, H., "Data Mining: Trends in Research and Development," *Rough Sets and Data Mining – Analysis of Imprecise Data*, Kluwer Academic Publishers, 1997.

[11] Flexer, A., "On the Use of Self-Organizing Mpas for Clustering and Visualization," *In Proc. Of the 3rd Int'l European Conf. PKDD 99*, Prague, Czech Republic, 1999.

[12] Ha, S.H. and Park, S.C., "Application of data mining tools to hotel data mart on the Intranet for data base marketing," *Expert Systems with Applications*, Vol. 15, pp. 1-31, 1998.

[13] Han, J. and Fu, Y., "Mining multiple-level association rules in large databases," *IEEE Transactions on knowledge and data engineering*, Vol. 11, No.5, pp. 798-804, 1999.

[14] Houtsma, M. and Swami, A., "Set-oriented mining of association rules," *Research Report RJ 9567*, IBM Almaden Research Center, San Jose, California, 1993.

[15] Kleissner, C., "Data Mining for the Enterprise," *In Proc. Of the 31st Annual Hawaii International Conf. On System Sciences*, pp. 295-304, 1998.

[16] Lin, T.Y. and Cercone, N., *Rough Sets and Data Mining – Analysis of Imprecise Data*, Kluwer Academic Publishers, 1997.

[17] Mastsuzawa, H. and Fukuda, T., "Mining Structured Association Patterns from Databases," *In Proc. Of the 4th Pacific-Asia Conference, PAKDD 2000*, Kyoto, Japan, pp. 233-244, 2000.

[18] Pasquier, N., Bastide Y., Taouil, R. and Lakhal, L., "Efficient mining of association rules using closed itemset lattices," *Information system*, Vol. 24, No. 1, pp. 25-46, 1999.

[19] Toivonen, H., Klemettinen, M., Ronkainen, P., Hatonen, K. and Mannila, H., "Pruning and grouping of discovered association rules," I*n Workshop Notes of the ECML95 Workshop on Statistics*, Machine Learning, and Knowledge Discovery in Databases, pp. 47-52, 1995.

[20] Tsechansky, S., Pliskin, N., Rabinowitz, G. and Porath, A., "Mining relational patterns from multiple relational tables," *Decision Support Systems*, Vol. 27, pp. 177-195, 1999.

[21] Tsur, D., Ullman, J.D., Abiteboul, S., Clifton, C., Motwani, R. Nestorov, S. and Rozenthal, A., "Query flocks: A generalization of association –rule mining," *In Proc. Of the ACM-SIGMOD*, pp. 1-12, 1998.

[22] Zhang, T., "Association Rules," *In Proc. Of the 4th Pacific-Asia Conference, PAKDD 2000*, Kyoto, Japan, pp. 233-244, 2000.