Data Mining and FNN-Driven Knowledge Acquisition and Inference Mechanism for Developing A Self-Evolving Expert Systems

Jin Sung Kim

School of Business Administration, Jeonju University
Hyoja-Dong 3-1200, Wansan-Ku, Jeonju, Jeonbuk, 560-759, South Korea
Tel: +82-63-220-2932, Fax: +82-63-220-20502, E-mail: kimjs@jeonju.ac.kr

Abstract:
In this research, we proposed the mechanism to develop self-evolving expert systems (SEES) based on data mining (DM), fuzzy neural networks (FNN), and relational database (RDB)-driven forward/backward inference engine. Most former researchers tried to develop a text-oriented knowledge base (KB) and inference engine (IE). However, they have some limitations such as 1) automatic rule extraction, 2) manipulation of ambiguousness in knowledge, 3) expandability of knowledge base, and 4) speed of inference. To overcome these limitations, many of researchers had tried to develop an automatic knowledge extraction and refining mechanisms. As a result, the adaptability of the expert systems was improved. Nonetheless, they didn't suggest a hybrid and generalized solution to develop self-evolving expert systems. To this purpose, in this study, we propose an automatic knowledge acquisition and composite inference mechanism based on DM, FNN, and RDB-driven inference. Our proposed mechanism has five advantages empirically. First, it could extract and reduce the specific domain knowledge from incomplete database by using data mining algorithm. Second, our proposed mechanism could manipulate the ambiguousness in knowledge by using fuzzy membership functions. Third, it could construct the relational knowledge base and expand the knowledge base unlimitedly with RDBMS (relational database management systems). Fourth, our proposed hybrid data mining mechanism can reflect both association rule-based logical inference and complicate fuzzy logic. Fifth, RDB-driven forward and backward inference is faster than the traditional text-oriented inference.

Keywords:
Knowledge base; Data mining; Fuzzy neural networks; Expert systems; Relational database; Inference engine; Self-evolving expert systems.

1. Introduction

The purpose of this study is to develop self-evolving expert system shell based on hybrid knowledge acquisition (DM and FNN) and RDB-driven high-speed inference mechanism. Expert system (ES) has been emerged as a new area of human knowledge management field during several decades. ES is a collection of emerging technologies inspired by the intelligent processing of expert knowledge in the human reasoning, decision-making, software engineering, process scheduling, medical diagnosis, and etc. Therefore, most researchers tried to extract the knowledge from human expert or database. It was commonly regarded as a major obstacle and bottleneck in the process of designing and implementing expert systems. Through the former researches automated knowledge acquisition tools were developed to help the knowledge engineer or even the expert himself to build and maintain the required knowledge systems (Eriksson, 1991; Gruber, 1987; Hong et al., 2002; Rafea et al., 2003).

Data mining is one of interested topics in the field of knowledge discovery (or extraction) in database (Bonchi, et al., 2001; Chakrabarti et al., 1999; Changchien & Lu, 2001; Hui & Jha, 2000; Lee et al., 2002; Song et al., 2001), and has been recognized as a new area for database research. The area can be defined as efficiently discovering interesting rules from large collections of data. Especially, association rule extraction mechanism, which was proposed by Agrawal et al.(1993), was a most popular tools to execute the data mining. Given a set of transactions, where each transaction is a set of item, an association rule is an expression of the form $X \rightarrow Y$. X and Y means the sets of items. An example of an association rule is: "20% of transactions that contain beer also contain diapers; 10% of all transactions contain both these items." Here 20% is called the *confidence* of the rule, and 10% the *support* of the rule.

However, association rules couldn't represent the fuzzy logic embedded in real world knowledge. Therefore, combination of fuzzy logic with data mining was very difficult for general decision makers because they require high expertise in knowledge discovery, artificial intelligence and fuzzy logic (Lee et al., 2002). In this sense, we propose a hybrid data mining mechanism based on association rule mining, fuzzy neural network, and fuzzy rule extraction algorithm.

Fuzzy neural networks and fuzzy rule extraction algorithm were used to extract the implicit fuzzy knowledge from database. Unfortunately, however, expandability and reusability degree are still remained as tackling points since no one of this tool integrates between task and domain (Allsopp et al., 2002). These issues are critical motivations in developing our mechanism.

The remainder of this paper is organized as follows. The research methodology is briefly proposed in Section 2. In Section 3, prototype system and its performance with an illustrative example are presented. Conclusion and some future works are finally suggested in Section 4.

2. Methodology

To develop the self-evolving expert system shell, we adopted and revised Rafea et al.'s (2003) research architecture. The architecture of our research was graphically shown in Figure 1.

The mechanism includes five main components namely: knowledge elicitation, library, ES (expert systems) generator, knowledge expresser, and inference engine. These components are similar with the research architecture of Rafea et al. (2003). However, we expanded the Rafea et al.'s (2003) research architecture with knowledge expresser and inference engine as shown in Figure 1. Detailed description for this architecture was shown in as follows.

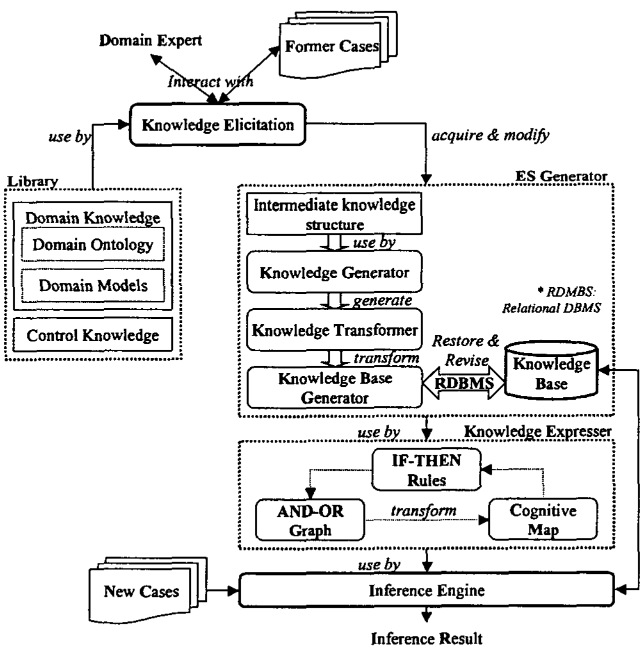


Figure 1 Research Methodology

- *Library*: Contains reusable domain knowledge, domain ontology, domain models, and control knowledge.
- *Knowledge Elicitation*: Its main functions are to create, maintain, and restore knowledge elicited from the various external inputs, fetch the relevant knowledge components from the library, and transform this knowledge into appropriate knowledge structure.
- *ES Generator*: Automatically generate an executable knowledge, which corresponds to the intermediate knowledge generated above. It contains knowledge generator, knowledge transformer, and knowledge base generator. During the process of knowledge transformation, ES Generator uses the RDBMS to restore and revise her knowledge bases.
- *Knowledge Expresser*: Support the three knowledge expression methods such as, IF-THEN rules, AND-OR graph, and Relationship matrix. It could help users to understand the knowledge base efficiently.
- *Inference Engine*: It contains the SQL-based bi-directional (forward and backward) inference engine. Therefore, its inference speed is faster than other text-oriented inference.

Especially, in Figure 1, our proposed ES generator and knowledge generator were based on fuzzy membership function, association rule mining and fuzzy neural networks. Knowledge generator could enrich the adaptability of knowledge base. The proposed knowledge generator consists of the four phases-association rule extraction, fuzzy neural networks, and fuzzy rule extractions. Figure 2 shows our knowledge generator.

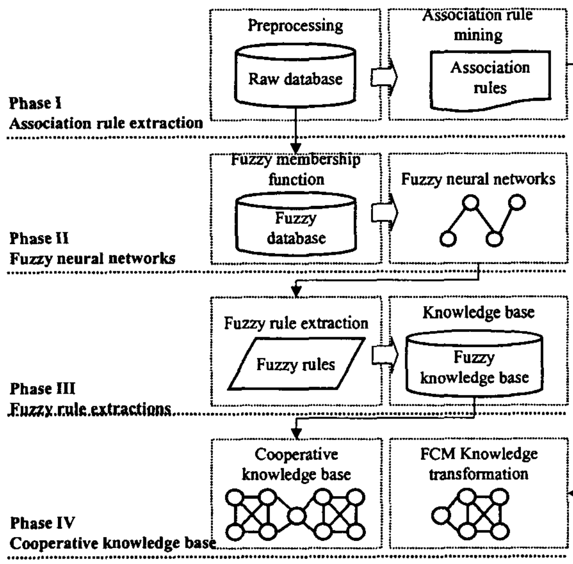


Figure 2 Knowledge generator

- *Phase I: Association rule extractions*

  The first phase is to preprocess the raw database and association rule mining. In this phase, we adopted the association rules mining technique to extract the relationships among items and attributes.
- *Phase II: Fuzzy neural networks*

  The second phase is to adapt the fuzzy membership function to traditional databases. As a result, raw database was transformed into fuzzy database. Then, we used the fuzzy neural networks to learn the implicit knowledge from the fuzzy database.
- *Phase III: Fuzzy rule extractions*

  The fourth stage of the proposed hybrid data mining mechanism is to apply the fuzzy rule extraction algorithm the fuzzy neural networks. Then, initial knowledge base was extended by these fuzzy rules.

- *Phase IV: Cooperative knowledge base*

The final stage of our proposed mechanism starts with the transformation of association rules into knowledge base. Then, association rule-based knowledge base was combined and with fuzzy rules extracted from fuzzy neural networks.

3. Implementation

To validate our proposed mechanism, we developed the prototype expert system shell SEES (Self-Evolving Expert System Shell) using Visual Basic (VB) and MS Access in a Windows-XP environment. The prototype system SEES has five components 1) Knowledge Elicitor, 2) Library, 3) ES Generator, 4) Knowledge Expresser, and 5) Inference Engine. As a validation set, in the application process, hepatitis data stored in University of California Irvine's machine learning data repository was used (UCI ML Group, 2003).

3.1 Library

Table 1 shows the library of hepatitis. Where, the library contains 20 attributes, 6 of which are linear valued and 14 of them is nominal. The diseases in this group are DIE and LIVE. This kind of knowledge for disease, attributes, and other control value of hepatitis were regarded as domain ontology, domain knowledge, domain model, and control knowledge simultaneously.

Table 1 - Domain knowledge of hepatitis

No	Attribute	Value
1	Class	DIE, LIVE
2	AGE	10, 20, 30, 40, 50, 60, 70, 80
3	SEX	male, female
4	STEROID	no, yes
5	ANTIVIRALS	no, yes
6	FATIGUE	no, yes
7	MALAISE	no, yes
8	ANOREXIA	no, yes
9	LIVER BIG	no, yes
10	LIVER FIRM	no, yes
11	SPLEEN PALPABLE	no, yes
12	SPIDERS	no, yes
13	ASCITES	no, yes
14	VARICES	no, yes
15	BILIRUBIN	0.39, 0.80, 1.20, 2.00, 3.00, 4.00
16	ALK PHOSPHATE	33, 80, 120, 160, 200, 250
17	SGOT	13, 100, 200, 300, 400, 500
18	ALBUMIN	2.1, 3.0, 3.8, 4.5, 5.0, 6.0
19	PROTIME	10,20, 30, 40, 50, 60, 70, 80, 90
20	HISTOLOGY	no, yes

First, totally 155 data was selected. After the pre-processing such as missing data elimination, however, totally 80 data was used for validation. Which was composed of 19 input variables and 1 output variable (two classes 1:die, 2:live).

3.2 Knowledge Elicitor

To acquire and modify a meaningful set of knowledge from the database, the first step to be done is to cleanse the original data so that the preprocessed data may become more traceable (Lee et al., 2002). Then, preprocessed data set was transformed into a table format for efficient knowledge elicitation. Table 2 shows the raw database and preprocessed data set of hepatitis check. SPSS and Clementine 6.0.1 were also used to preprocess the raw-data and extract the association rules.

Table 2 - Example of raw database and preprocessed data set

(a) Raw dataset of hepatitis check

2,2,0,3,0,0,0,0,1,0,0,0,0,0,0,3,2,0,0,0,0,0,0,0,0,0,0,3,0,0,0,1,0,55,2
3,3,3,2,1,0,0,0,1,1,1,0,0,1,0,1,2,0,2,2,2,2,2,1,0,0,0,0,0,0,0,1,0,8,1
2,1,2,3,1,3,0,3,0,0,0,1,0,0,0,1,2,0,2,0,0,0,0,0,2,0,2,3,2,0,0,2,3,26,3
2,2,2,0,0,0,0,0,3,2,0,0,0,3,0,0,2,0,3,2,2,2,2,0,0,3,0,0,0,0,0,3,0,40,1
2,3,2,2,2,2,0,2,0,0,0,1,0,0,0,1,2,0,0,0,0,0,0,0,2,2,3,2,3,0,0,2,3,45,3

(b) Preprocessed database of hepatitis check

No	Class	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20
1	2	30	2	1	2	2	2	2	1	2	2	2	2	2	1	85	18	4	?	1
2	2	50	1	1	2	1	2	2	1	2	2	2	2	2	0.9	135	42	3.5	?	1
3	2	78	1	2	2	1	2	2	2	2	2	2	2	2	0.7	96	32	4	?	1
4	2	31	1	?	1	2	2	2	2	2	2	2	2	2	0.7	46	52	4	80	1
5	2	34	1	2	2	2	2	2	2	2	2	2	2	2	1	?	200	4	?	1
6	2	34	1	2	2	2	2	2	2	2	2	2	2	2	0.9	95	28	4	75	1
7	1	51	1	1	2	1	2	1	2	2	1	1	2	2	?	?	?	?	?	1
8	2	23	1	2	2	2	2	2	2	2	2	2	2	2	1	?	?	?	?	1
9	2	39	1	2	2	1	2	2	2	1	2	2	2	2	0.7	?	48	4.4	?	1
10	2	30	1	2	2	2	2	2	2	2	2	2	2	2	1	?	120	3.9	?	1
11	2	39	1	1	1	2	2	2	1	1	2	2	2	2	1.3	78	30	4.4	85	1
12	2	32	1	2	1	1	2	2	2	1	2	1	2	2	1	59	249	3.7	54	1
13	2	41	1	2	1	1	2	2	2	1	2	2	2	2	0.9	81	60	3.9	52	1
14	2	30	1	2	2	1	2	2	2	1	2	2	2	2	2.2	57	144	4.9	78	1
15	2	47	1	1	1	2	2	2	2	2	2	2	2	2	?	?	60	?	?	1
16	2	38	1	1	2	1	1	1	2	2	2	2	1	2	2	72	89	2.9	46	1
17	2	66	1	2	2	1	2	2	2	2	2	2	2	2	1.2	102	53	4.3	?	1
18	2	40	1	1	2	1	2	2	2	1	2	2	2	2	0.6	62	166	4	63	1
19	2	38	1	2	2	2	2	2	2	2	2	2	2	2	0.7	53	42	4.1	85	2
20	2	38	1	1	1	2	2	2	1	1	2	2	2	2	0.7	70	28	4.2	62	1

3.3 ES Generator

Phase I: Association rule extraction

The association rule mining algorithm we adopted here is an APRIORI algorithm (Agrawal et al., 1993), which was known to yield a set of association rules. Based on the hepatitis data in Table 2, the corresponding association rules were extracted with a threshold of 80% confidence. Figure 3 shows the association rule extraction process using Clementine.

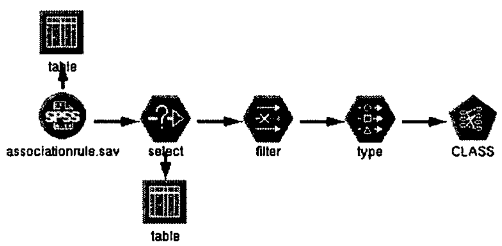


Figure 3 - Association rule extraction process using Clementine

Table 3 shows an excerpt of the derived association rules. The association rules shown in Table 3 are straightforward and easy to understand and interpret.

Table 3 - Example of association rules from the database

CLASS = 1 <= V9 = 2 & V8 = 2 & V3 = 1 & V11 = 2 & V20 = 2 (9:22.5%, 0.889)
CLASS = 1 <= V9 = 2 & V8 = 2 & V3 = 1 & V5 = 2 & V20 = 2 (13:32.5%, 0.846)
CLASS = 1 <= V9 = 2 & V8 = 2 & V3 = 1 & V6 = 1 & V20 = 2 (11:27.5%, 0.909)
CLASS = 1 <= V9 = 2 & V8 = 2 & V3 = 1 & V4 = 1 & V20 = 2 (7:17.5%, 1.0)
CLASS = 1 <= V9 = 2 & V8 = 2 & V3 = 1 & V7 = 1 & V20 = 2 (9:22.5%, 0.889)
CLASS = 1 <= V9 = 2 & V8 = 2 & V3 = 1 & V20 = 2 & V12 = 1 (9:22.5%, 0.889)
CLASS = 2 <= V9 = 2 & V8 = 2 & V14 = 2 & V20 = 1 & V7 = 2 (16:40.0%, 1.0)
CLASS = 2 <= V9 = 2 & V8 = 2 & V11 = 2 & V20 = 1 & V7 = 2 (16:40.0%, 1.0)
CLASS = 2 <= V9 = 2 & V8 = 2 & V13 = 2 & V20 = 1 & V7 = 2 (16:40.0%, 1.0)

Phase II: Fuzzy membership functions and fuzzy neural networks

In this phase, we adapted the fuzzy membership functions to transform the real data into fuzzy sets. Fuzzy membership functions used in this phase was as follows (Mitra & Pal, 1994):

$$\pi(F_j : c, \lambda) = \begin{bmatrix} 2\left(1-\frac{|F_j - c|}{\lambda}\right)^2, & \text{for } \frac{\lambda}{2} \le |F_j - c| \le \lambda \\ 1-2\left(\frac{|F_j - c|}{\lambda}\right)^2, & \text{for } 0 \le |F_j - c| \le \frac{\lambda}{2} \\ 0, & \text{otherwise} \end{bmatrix}$$

$$\lambda_{medium} = \frac{1}{2}(F_{\max} - F_{\min})$$

$$C_{medium} = F_{\min} + \lambda_{medium}$$

$$\lambda_{low} = \frac{1}{fdenom}(c_{medium} - F_{\min})$$

$$C_{low} = C_{medium} + 0.5 * \lambda_{low}$$

$$\lambda_{high} = \frac{1}{fdenom}(F_{\max} - C_{medium})$$

$$C_{high} = C_{medium} + 0.5 * \lambda_{high}$$

Table 4 shows the fuzzified database transformed by fuzzy membership functions.

Table 4 - Fuzzified database

No	V2_L	V2_M	V2_H	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15_L	V15_M	V15_H	Class	Class
1	0.96	0.61	0.04	0.10	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.98	0.14	0.00	0.90	0.10
2	0.78	0.91	0.22	0.10	0.10	0.10	0.90	0.90	0.90	0.10	0.10	0.90	0.90	0.90	0.90	0.99	0.40	0.01	0.90	0.10
3	0.99	0.43	0.01	0.10	0.90	0.10	0.10	0.90	0.90	0.90	0.10	0.90	0.10	0.90	0.90	0.99	0.19	0.00	0.90	0.10
4	0.67	0.97	0.33	0.10	0.90	0.10	0.10	0.90	0.90	0.90	0.10	0.90	0.90	0.90	0.90	0.98	0.14	0.00	0.90	0.10
5	1.00	0.28	0.00	0.10	0.90	0.90	0.10	0.90	0.90	0.90	0.10	0.90	0.90	0.90	0.90	0.72	0.95	0.28	0.90	0.10
6	0.83	0.87	0.17	0.10	0.10	0.90	0.10	0.10	0.10	0.90	0.90	0.90	0.90	0.10	0.90	0.81	0.88	0.19	0.90	0.10
7	0.73	0.95	0.27	0.10	0.10	0.90	0.10	0.90	0.90	0.90	0.10	0.90	0.90	0.90	0.90	0.93	0.04	0.00	0.90	0.10
8	0.83	0.87	0.17	0.10	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.95	0.06	0.00	0.90	0.10
9	0.83	0.87	0.17	0.10	0.10	0.10	0.90	0.90	0.90	0.10	0.10	0.90	0.90	0.90	0.90	0.95	0.06	0.00	0.90	0.10
10	0.82	0.00	0.00	0.90	0.90	0.10	0.10	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.90	0.98	0.14	0.00	0.90	0.10
...																				

Phase III: Fuzzy rule extractions

After the learning of fuzzy neural networks, we adopted the fuzzy rule extraction algorithm proposed by Mitra & Pal (1994) to fuzzy neural network. Fuzzy rule extraction algorithm was shown in Table 5.

Table 5 - Fuzzy rule extraction algorithm

Step I: Path generation by backtracking
Step 1.1: Find the intermediate node i which has a positive effect on output node j in H(output) layer. If $w_{ji}^{H-1} > 0$, *Then select node i in H-l layer*
Step 1.2: Select the connection weights between i and j.
Step 1.3: Select the input node, which has an output value more than 0.5. Then, find the connection weight from the lower layer until there's no connection weight.
Step 1.4: Sort the selected connection weight list.

Step 2: Sentence generation
Adapt two conditions as follows:
Condition 1: Define the conditions for sorting. Then, generate the If-Then rules.
Condition 2: Select the linguistic hedge or real values.

Table 6 shows the fuzzy rules extracted from fuzzy neural networks. Where, each real value means the fuzzy membership value.

Table 6 - Sample of fuzzy rules extracted from fuzzy neural networks

CLASSS = 1<=V2_L=0.96 & V2_M=0.61 & V3=0.90 & V4=0.90 & V5=0.90 & V6=0.90 & V7=0.90 & V8=0.90 & V9=0.90 & V10=0.90 & V11=0.90 & V12=0.90 & V13=0.90 & V14=0.90 & V15_L=0.98 & V15_M=0.68 & V16_L=1.00 & V16_H=1.00 & V17_=0.92 & V17_H=0.82 & V18_L=0.98 & V18_M=0.81 & V18_H=0.87 & V19_L=0.62 & V19_M=0.73 & V19_H=0.91 (95%)
CLASSS = 1<=V2_L=0.78 & V2_M=0.91 & V5=0.90 & V6=0.90 & V7=0.90 & V8=0.90 & V9=0.90 & V10=0.90 & V11=0.90 & V12=0.90 & V13=0.90 & V14=0.90 & V15_L=0.99 & V16_L=0.99 & V16_M=0.64 & V17_=0.93 & V18_H=1.00 & V19_H=1.00 (95%)
CLASSS = 2<=V2_M=0.95 & V2_H=0.99 & V4=0.90 & V5=0.90 & V6=0.90 & V7=0.90 & V8=0.90 & V9=0.90 & V10=0.90 & V11=0.90 & V12=0.90 & V13=0.90 & V14=0.90 & V15_L=0.97 & V15_M=0.56 & V16_L=0.97 & V16_M=0.55 & V16_H=0.66 & V17_=1.00 & V17_M=0.91 & V17_H=0.79 & V18_M=1.00 & V18_H=0.55 & V19_L=1.00 & V19_M=0.88 & V19_H=0.81 & V20=0.90 (90%)
CLASSS = 2<=V2_M=0.95 & V2_H=0.73 & V4=0.90 & V5=0.90 & V6=0.90 & V7=0.90 & V8=0.90 & V9=0.90 & V10=0.90 & V11=0.90 & V13=0.90 & V15_L=0.81 & V15_M=0.88 & V16_L=1.00 & V17_=0.89 & V18_M=0.60 & V18_H=0.96 & V19_M=0.96 & V19_H=0.70 & V20=0.90 (90%)

Phase IV: Cooperative knowledge base

After the extraction of association rules and fuzzy rules, we combined two different kinds of knowledge bases into cooperative knowledge base. Table 7 shows the cooperative knowledge base.

Table 7 - Example of cooperative knowledge base

CLASS = 1 <= V9 = 2 & V8 = 2 & V3 = 1 & V11 = 2 & V20 = 2 (9:22.5%, 0.889)
CLASS = 1 <= V9 = 2 & V8 = 2 & V3 = 1 & V5 = 2 & V20 = 2 (13:32.5%, 0.846)
CLASS = 1 <= V9 = 2 & V8 = 2 & V3 = 1 & V6 = 1 & V20 = 2 (11:27.5%, 0.909)
CLASS = 2 <= V9 = 2 & V8 = 2 & V14 = 2 & V20 = 1 & V7 = 2 (16:40.0%, 1.0)
CLASS = 2 <= V9 = 2 & V8 = 2 & V11 = 2 & V20 = 1 & V7 = 2 (16:40.0%, 1.0)
CLASS = 2 <= V9 = 2 & V8 = 2 & V13 = 2 & V20 = 1 & V7 = 2 (16:40.0%, 1.0)

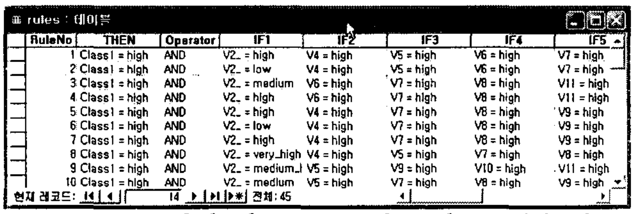
CLASSS = 1<=V2_L=0.96 & V2_M=0.61 & V3=0.90 & V4=0.90 & V5=0.90 & V6=0.90 & V7=0.90 & V8=0.90 & V9=0.90 & V10=0.90 & V11=0.90 & V12=0.90 & V13=0.90 & V14=0.90 & V15_L=0.98 & V15_M=0.68 & V16_L=1.00 & V16_H=1.00 & V17_=0.92 & V17_H=0.82 & V18_L=0.98 & V18_M=0.81 & V18_H=0.87 & V19_L=0.62 & V19_M=0.73 & V19_H=0.91 (95%)

CLASSS = 1<=V2_L=0.78 & V2_M=0.91 & V5=0.90 & V6=0.90 & V7=0.90 & V8=0.90 & V9=0.90 & V10=0.90 & V11=0.90 & V12=0.90 & V13=0.90 & V14=0.90 & V15_L=0.99 & V16_L=0.99 & V16_M=0.64 & V17_=0.93 & V18_H=1.00 & V19_H=1.00 (95%)

CLASSS = 2<=V2_M=0.95 & V2_H=0.99 & V4=0.90 & V5=0.90 & V6=0.90 & V7=0.90 & V8=0.90 & V9=0.90 & V10=0.90 & V11=0.90 & V12=0.90 & V13=0.90 & V14=0.90 & V15_L=0.97 & V15_M=0.56 & V16_L=0.97 & V16_M=0.55 & V16_H=0.66 & V17_=1.00 & V17_M=0.91 & V17_H=0.79 & V18_M=1.00 & V18_H=0.55 & V19_L=1.00 & V19_M=0.88 & V19_H=0.81 & V20=0.90 (90%)

CLASSS = 2<=V2_M=0.95 & V2_H=0.73 & V4=0.90 & V5=0.90 & V6=0.90 & V7=0.90 & V8=0.90 & V9=0.90 & V10=0.90 & V11=0.90 & V13=0.90 & V15_L=0.81 & V15_M=0.88 & V16_L=1.00 & V17_=0.89 & V18_M=0.60 & V18_H=0.96 & V19_M=0.96 & V19_H=0.70 & V20=0.90 (90%)

The rule premise part may contain arbitrarily complex conjunctions or disjunctions nested within each clause. Otherwise, a separate rule is written for each clause, instead of writing rules whose premise would be a disjunction of clauses. The ACTION part or THEN part indicates one or more conclusions that can be drawn if the premises are satisfied making the rules purely inferential. Each rule is highly stylized- with an IF-THEN format and a specified set of admissible primitives. These rules transformed into relational database. Therefore, this tightly structured database form makes it possible for SEES to be designed to execute them in a form of SQL-based inference. Furthermore, it could be represented as an AND-OR graph or Relationship matrix. Figure 4 shows the knowledge base (fuzzy rules) restored in relational database.


rules : 테이블

RuleNo	THEN	Operator	IF1	IF2	IF3	IF4	IF5
1	Class1 = high	AND	V2_ = high	V4 = high	V5 = high	V6 = high	V7 = high
2	Class1 = high	AND	V2_ = low	V4 = high	V5 = high	V6 = high	V7 = high
3	Class1 = high	AND	V2_ = medium	V6 = high	V7 = high	V8 = high	V11 = high
4	Class1 = high	AND	V2_ = high	V6 = high	V7 = high	V8 = high	V11 = high
5	Class1 = high	AND	V2_ = high	V4 = high	V7 = high	V8 = high	V9 = high
6	Class1 = high	AND	V2_ = low	V4 = high	V7 = high	V8 = high	V9 = high
7	Class1 = high	AND	V2_ = high	V4 = high	V7 = high	V8 = high	V9 = high
8	Class1 = high	AND	V2_ = very_high	V4 = high	V5 = high	V7 = high	V8 = high
9	Class1 = high	AND	V2_ = medium_l	V5 = high	V9 = high	V10 = high	V11 = high
10	Class1 = high	AND	V2_ = medium	V5 = high	V7 = high	V8 = high	V9 = high

현재 레코드: 14 전체: 45

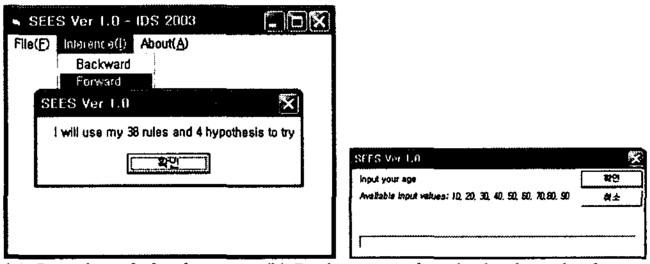
Figure 4 - Knowledge base restored in relational database

3.4 Knowledge Expresser

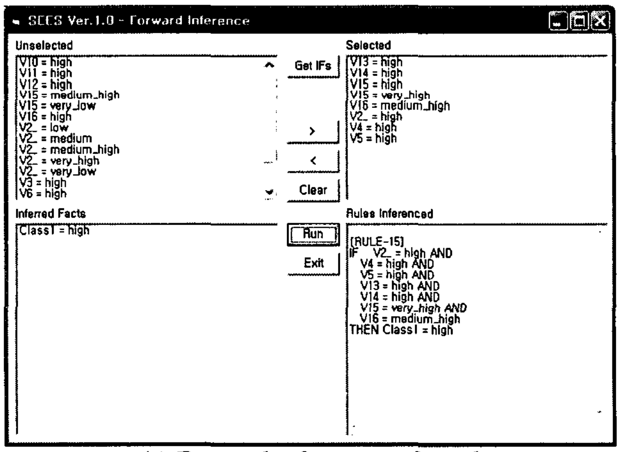
Totally 38 production rules were extracted by using APRIORI algorithms which are highly efficient techniques developed by Agrawal (1993). SEES could express the knowledge base as OAV type production rules (IF-THEN rules), AND-OR graph, and Relationship matrix (Kim, 2003).

3.5 Inference Engine

Inference engine depends on RDB and SQL-driven forward/backward inference mechanism. Therefore, rule consistency check and incompleteness check was easier than other traditional text-driven works (Kim, 2003). Figure 5 shows the example of inference process and final inference result for a patient's data set.



(a) Open knowledge base *(b) Dialogue window for backward inference*



(c) Forward inference and result
Figure 5 - Inference results of SEES

In the first step, user opened the knowledge base. Then, he might confirm the total number of rules and hypotheses (Figure 5(a)). In the third step, SEES shows the dialog window for backward/forward inference (Figure 5(b), 5(c)) requiring user's response. Especially, Figure 5(c) shows the unselected and selected IFs, inferred facts, and final conclusions. Then the final conclusion was translated into OAV-typed IF-THEN rules as shown in Figure 5(c).

4. Conclusion

In this study, we introduced the problems of traditional data mining and construction mechanism for ES. Therefore, we suggested an automatic expert systems shell construction and maintaining mechanism. The proposed mechanism consisted of the five main components Library, Knowledge Elicitation, ES Generator, Knowledge Expresser, and Inference Engine. In the implementation process, we developed a prototype expert system shell SEES and proved the inference ability using hepatitis data set. This mechanism and prototype systems were based on data mining, fuzzy membership functions, fuzzy neural networks, and RDB-driven forward/backward inference algorithm, which were mainly aimed at expand the adaptability, expandability, and reusability of knowledge base. In addition, this study has shown how the knowledge base could be transformed into IF-THEN rules, AND-OR graph, and Relationship matrix to help the user's correct recognition in knowledge base. Nonetheless, our study has some limitations. First, we developed simple prototype expert system shell. Second, effective rule refinement process (such as pruning, conflict resolution, and etc.) was omitted, which was enable to improve the inference ability of that expert systems. Then, further research topics still remains. First, this expert system shell should be improved as an Internet-based system to support the Web-based user's decision-making. Second, other AI (artificial intelligence) technologies (such as fuzzy logic, neural networks, rough set, and etc.) may improve the inference ability of our expert system shell. Third, multiple-decision problem and concurrent decision-making should be supported using other concurrent engineering mechanism. Fourth, the basic technology of association rule mining used for this study needs to be improved so that more fuzzy knowledge can be analyzed. Fifth, fuzzy membership functions need to be integrated with other rule refining and reasoning mechanism.

References

[1] Agrawal, R., Imielinski, T., & Swami, A. (1993), Database mining: A performance perspective, *IEEE Transactions on Knowledge and Data Engineering,* 5(6), 914-925.

[2] Allsopp, D.J., harrison, A., & Sheppard, C. (2002), A database architecture for reusable CommonKADS agents specification components, *Knowledge-Based Systems*, 15, 275-283.

[3] Bonchi, F., Giannotti, F., Gozzi, C., Manco, G., Nanni, M., Pedreschi, D., Renso, C., & Ruggieri, S. (2001), "Web Log Data Warehousing and Mining for Intelligent Web Caching," *Data & Knowledge Engineering,* 39, pp.165-189.

[4] Brasil, L.M., Azevedo, F.M., & Barreto, J.M. (2001), Hybrid expert system for decision supporting in the medical area: complexity and cognitive computing, *International Journal of Medical Informatics*, 63, 19-30.

[5] Chakrabarti, S., Dom, B.E., Kumar, S. R., Raghavan, P., Rajagopalan, S., Tomkins, A., Gibson, D., & Kleinberg, J.M. (1999), "Mining the Web's Link Structure," *Computer,* 32, pp.60-67.

[6] Changchien, S.W., & Lu, T.C. (2001), "Mining Association Rule Procedure to Support On-Line Recommendation by Customers and Products Fragmentation," *Expert Systems with Applications*, 20, pp.325-335.

[7] Davis, R. (1980), Meta-rules: Reasoning about control, *AI Journal*, 15, 179-222.

[8] Davis, R., & Buchanan, B. (1977), Meta-level knowledge: Overview and applications, *International Joint Conference on Artificial Intelligence (IJCAI 1977)*, 5, 920-928.

[9] Eriksson, E. (1991), *Meta-tool support for knowledge acquisition*, PhD Thesis, Linköping University, Sweden.

[10] Gruber, T.R. (1987), Acquiring strategic knowledge from experts, In J.H. Boose, & Gaines, B.R. (Eds.), *The foundations of knowledge acquisition*, New York: Academic Press.

[11] Hong, T.P., Tseng, L.H., & Wang, S.L. (2002), Learning rules from incomplete training examples by rough sets, *Expert Systems with Applications*, 22, 285-293.

[12] Hui, S.C. & Jha, G. (2000), "Data Mining for Customer Service Support," *Information & Management*, 38, pp.1-13.

[13] Kim, J.S. (2003), "Development of Expert Systems using Automatic Knowledge Acquisition and Composite Knowledge Expression Mechanism," *Proceedings of the 4th International Symposium on Advanced Intelligent Systems (ISIS)* 2003, Jeju, Korea, pp.447-450, pp.25-28.

[14] Lee, K.C., Kim, J.S., Chung, N.H., & Kwon, S.J. (2002), Fuzzy cognitive map approach to Web-mining inference amplification, *Expert Systems with Applications*, 22, 197-211.

[15] Mitra, S. & Pal, S.K. (1994), "Logical Operation Based Fuzzy MLP for Classification and Rule Generation", *Neural Networks,* 7(2), pp.353-373. 1994.

[16] Quinlan, J.R. (1993), *C4.5 - Programs for Machine Learning,* The Morgan Kaufmann series in machine learning, Morgan Kaufman Publishers, San Mateo, CA, USA.

[17] Rafea, A., Hassen, H., & Hazman, M. (2003), Automatic knowledge acquisition tool for irrigation and fertilization expert systems, *Expert Systems with Applications*, 24, 49-57.

[18] Song, H.S., Kim, J.K., & Kim, S.H. (2001), "Mining the Change of Customer Behavior in an Internet Shopping Mall," *Expert Systems with Applications,* 21, pp.157-168.

[19] UC Irvine Machine Learning Group (2003), http://www.ics.uci.edu/~mlearn/MLSummary.html.