퍼지 및 지능시스템학회 논문지 2003, Vol. 13, No. 6, pp. 743-748

RDB-based Automatic Knowledge Acquisition and Forward Inference Mechanism for Self-Evolving Expert Systems

Jin Sung Kim

School of Business Administration, Jeonju University
Hyoja-Dong 3-1200, Wansan-Ku, Jeonju, Jeonbuk, 560-759, Korea

Abstract

In this research, we propose a mechanism to develop an inference engine and expert systems based on relational database (RDB) and SQL (structured query language). Generally, former researchers had tried to develop an expert systems based on text-oriented knowledge base and backward/forward (chaining) inference engine. In these researches, however, the speed of inference was remained as a tackling point in the development of agile expert systems. Especially, the forward inference needs more times than backward inference. In addition, the size of knowledge base, complicate knowledge expression method, expansibility of knowledge base, and hierarchies among rules are the critical limitations to develop an expert system. To overcome the limitations in speed of inference and expansibility of knowledge base, we proposed a relational database-oriented knowledge base and forward inference engine. Therefore, our proposed mechanism could manipulate the huge size of knowledge base efficiently, and inference with the large scaled knowledge base in a short time. To this purpose, we designed and developed an SQL-based forward inference engine using relational database. In the implementation process, we also developed a prototype expert system and presented a real-world validation data set collected from medical diagnosis field.

Key words : Database, SQL, Inference Engine, Forward Inference, Expert System.

1. Introduction

Over the past 20 years, expert systems had been widely used in domains where mathematical models could not be easily built, human experts were not available or the cost of querying an expert was high. Expert systems were commonly used when an inadequate algorithm or no algorithmic solution exists. Generally, to generate the explicit knowledge from incomplete domain, a knowledge engineer was needed to produce a dialog with a human expert. Therefore, the explicit (encoded) knowledge was elicited into a knowledge base to develop a domain expert system.

However, the whole development process of expert system was very time-consuming [2][7]. Therefore, shortening the time in developing was then the most important factor for the successful development of an expert system. In addition, automatic knowledge acquisition and development of knowledge base are remained as a development bottleneck [8].

As a result, many knowledge acquisition tools have been developed, e.g. MOLE [6], SALT [13], KCT [11], ITAKA [5], HCGT [17]. In effect, these tools vary according to its strategies and approaches for solving the knowledge acquisition problems [16].

Nevertheless, the reusability degree is still limited, since no of these tools integrates between task and domain [16]. In addition, the speed of inference and expandability of knowledge base were still remained as a tackling point to develop an expert system. Especially, the method of inference was particularly important in expert system development because inference mechanism was the basic technology by which expert systems can solve the problems. As a result of inference, expert system could offer several alternatives or a specific solution. To overcome these limitations, in this study, we propose a RDB-based expandable knowledge base construction and inference. To accomplish this purpose, during the experiment, we developed RDB & SQL-based high-speed forward inference mechanism. In addition, RDB-based knowledge base could improve the reusability of knowledge base effectively.

The remainder of this paper is organized as follows: The research background was briefly reviewed in Section 2. The methodology was proposed in Section 3. Our prototype system was presented in Section 4. Conclusion and future work are finally given in Section 5.

2. Research Background

2.1 Rule-based expert systems

The standard rule structure and knowledge expression method used in constructing a knowledge base was OAV (object-attribute-value) type IF-THEN rule. The structure of standard rule was

접수일자 : 2003년 8월 27일
완료일자 : 2003년 11월 26일

IF *condition* THEN *action*

To enrich the interpretability, Michalski and Winston (1986) proposed the censored production rule (CPR) of the form

<IF condition THEN action UNLESS censor>

as an underlying representational and computational mechanism to enable logic based systems to exhibit variable precision logic (VPL) in which certainty varies, while specificity stays constant. The form of CPR is as follows:

IF <premise> THEN <decision> UNLESS <censors>

And can be written P → DLC

Where P is the *premise*, D is the *decision*, and C is the *censor*.

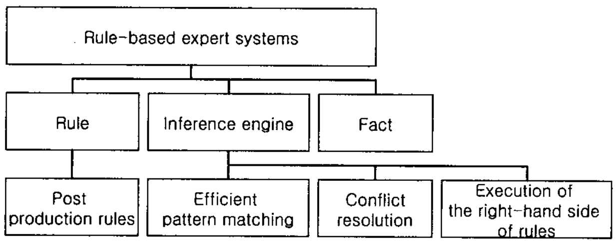


Figure 1. Foundations of modern rule-based expert systems.

The *premise* is a conjunction of literals; the *decision* is a single literal; and the *censor* is a disjunction of literal. CPRs embody both object level and control level information [8].

As shown above, most of expert systems were based on IF-THEN rule base. Figure 1 summarizes the foundations of modern rule-based expert system technologies [7].

2.2 Automatic knowledge acquisition

Knowledge acquisition is commonly regarded as a major obstacle and bottleneck in the process of designing and implementing knowledge-based expert system [16]. Knowledge typically could be acquired through one of two ways: either manual or automatic.

Recently, data mining was known as efficient rule mining tools. Data mining, also known as knowledge discovery in databases, is a rapidly emerging field. This technology is motivated by the need of new techniques to help analyze, understand or even visualize the huge amounts of stored data gathered from business and scientific applications [3]. This area can be defined as efficiently discovering interesting rules from large data set. One of attractive artificial intelligence (AI) technologies, machine-learning algorithm has been adopted to ease the knowledge acquisition bottleneck. Among proposed approaches, deriving rules from training examples was the most common [9] [10] [12] [19].

Recently, association rule mining technology was introduced by Srikant and Agrawal (1995). Given a large database of transactions, where each transaction consists of a set of items, and a taxonomy (is-a hierarchy) on the items, it could find associations between items at any level of the taxonomy.

The problem of mining association rules was introduced in Agrawal et al. (1993). Given a set of transactions, where each transaction is a set of items, an association rule is an expression X ➔ Y, where X and Y are sets of items. The intuitive meaning of such a rule is that transactions in the database which contain the items in X tend to also contain the items in Y. An example of such a rule might be that 98% customers who purchase tire and auto accessories also buy some automotive services; here 98% is called the *confidence* of the rule. The *support* of the rule X ➔ Y is the percentage of transactions that contain both X and Y. The problem of mining association rules is to find all rules that satisfy a user-specified *minimum support* and *minimum confidence*. Applications included cross-marketing, attached mailing, catalog design, loss-leader analysis, store layout, and customer segmentation based on buying patterns [18].

Our research methodology was also based on these data mining mechanisms.

2.3 Forward inference

Forward inference was well known inference method in the area of rule-based expert systems. Which begin with a set of known facts, derives new facts using rules whose premises match the known facts, and continues this process until a goal state is reached or until no further rules have premises that match the known or derived facts [4]. Figure 2 shows the whole forward inference process.

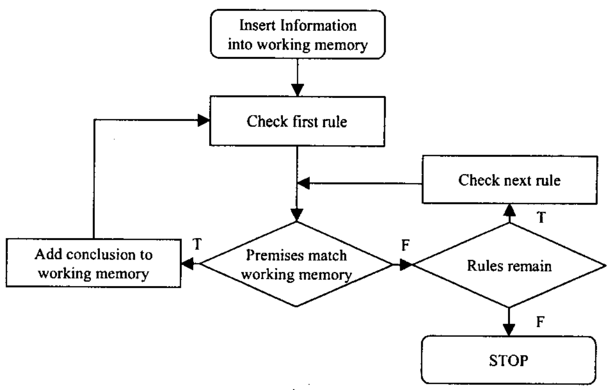


Figure 2. Forward inference process

3. Methodology

Our research methodology was graphically presented in Figure 3. This methodology 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). In this study, however, we expanded and revised Rafea et al. (2003)'s research architecture with other components as shown in Figure 3.

- *Library*: Library contains both reusable domain knowledge and control knowledge such as domain ontology, domain models, and control knowledge.
- *Knowledge Elicitation*: The main functions of knowledge elicitor are to create, maintain, and restore knowledge elicited from the external input, fetch the relevant knowledge components from the library, and transform this knowledge into appropriate knowledge structure.
- *ES Generator*: Automatically generates an executable knowledge, which corresponds to the intermediate knowledge generated above. It contains knowledge generator, knowledge transformer, and knowledge base generator. During the 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*: In this study, we developed SQL-based forward inference engine. Therefore, we could reduce the inference time effectively. This inference engine is different with Rafea et al. (2003)'s research mechanism. They didn't propose any specific inference engine.

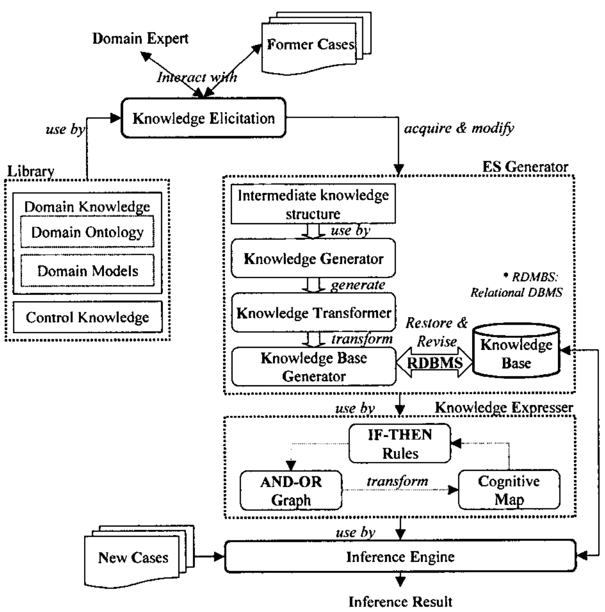


Figure 3. Research methodology

Figure 4 shows our proposed high-speed forward inference mechanism. Generally, most of traditional expert systems' inference engine use text-oriented pattern-matching mechanism to find a specific rule. When the more rules we get, therefore, the more time was spent. In the first phase, we removed all the rules, which didn't contain any facts matched with pre-inserted information. We called this phase as *Elimination of garbage rules*.

In the second phase, we developed the SQL-based revised forward inference mechanism. In contrast with traditional forward inference mechanism, we deleted matched rules and unmatched rules during inference because these rule has no more information available after the check. Therefore, we could reduce the forward inference time effectively.

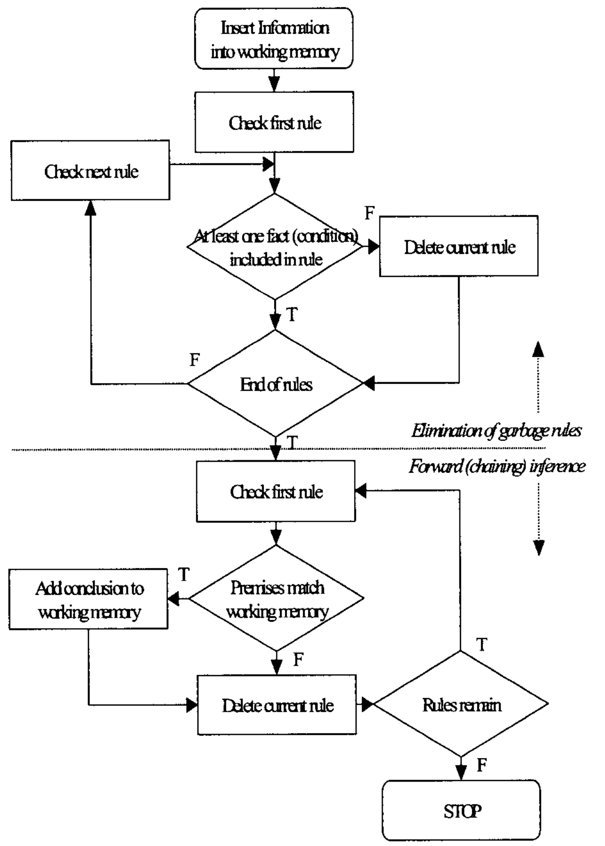


Figure 4. High-speed forward inference process

Table 1 shows the pseudo code for our proposed RDB-based forward inference algorithm.

Table 1. Pseudo code for RDB-based forward inference

```
Sub ForwardInference( )
Move to First Rule
Do While Not EOF of KB
  If Empty Rule Then Exit Do
  i = 0
  Do Until Is Null Rule
    For j = 0 To Number of Rules
      If Found Final Conclusion Then
        Exit Do
      End If
    Next j

    For j = 0 To Number of FACTs
      If FACTs(j) = Value of IFs in Rule Then
        FiredIF = True
        Exit For
      End If
    Next j

    If Not FiredIF Then
      For j = 0 To Number of Newly Inferred FACTs
        If Inferred FACTs(j) = IFs in Rule Then
          FiredIF = True
          Exit For
        End If
      Next j
    End If

    If FiredIF Then
      FiredRule = True
    Else
      FiredRule = False
      Exit Do
    End If
    i = i + 1
  Loop

  If FiredRule = True Then
    Add FACT in THENs i = IFsStart
    Do Until IsNull Fired Rule
      Save & Write Fired Rule (IFs and THEN)
    Loop
  End If
Move to Next Rule
Loop
End Sub
```

4. Implementation

To validate our proposed mechanism, we developed the prototype expert system shell SEES (Self Evolving Expert Systems) using the Visual Basic and Microsoft Access in a Windows-XP environment. Experimental data was collected from UCL Machine Learning group (2003). Which contains 366 dermatology analysis data. Database contains 34 attributes, 33 of which are linear valued and one of them is nominal. The diseases in this group are psoriasis, seboreic dermatitis, lichen planus, pityriasis rosea, cronic dermatitis, and pityriasis rubra pilaris.

To extract the rules from this data set we used preprocessed data set. Table 2 shows the preprocessed data set.

Table 2. Preprocessed data set

	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15	v16	v17	v18	v19	v20	v21	v22	v23	v24	v25	v26	v27	v28	v29	v30	v31	v32	v33	age	class
1	1	1	2	0	0	0	3	0	3	0	1	0	0	0	0	2	1	1	1	1	0	0	0	0	0	0	0	0	0	2	2	2	0	7	6
2	2	2	1	0	0	0	3	0	2	0	1	0	0	0	0	2	2	1	2	0	0	0	0	0	0	0	0	2	0	3	3	2	0	7	6
3	2	2	2	1	0	0	2	0	2	2	1	0	0	0	0	1	2	0	2	0	0	0	0	0	0	0	0	2	0	2	2	2	0	7	6
4	2	2	1	0	0	0	2	0	2	0	0	0	0	0	0	3	2	0	1	0	0	0	0	0	0	0	0	3	0	2	2	2	0	7	6
5	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

We used the Clementine 6.0.1 (SPSS, 2002) as a knowledge generator. The generated knowledge is encoded as production rules (Table 3). The tightly structured database form makes it possible for SEES to be designed to execute them in a form of SQL-based inference. The rules translated into a readable English format as shown in Table 3.

Table 3. SEES production rule

PREMISE: (AND (LESS_THAN_OR_EQUAL_TO CNTXT BAND-LIKE-INFILTRATE 1)
(GREATER_THAN CNTXT FIBROSIS-OF-THE-PAPILLARY-DERMIS 1))

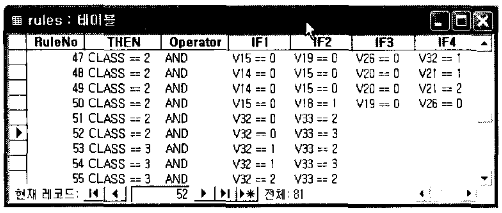
ACTION: (CONCLUDE CNTXT CLASS CHRONIC-DERMATITIS)

SEES's English translation:
IF 1) the band-like infiltrate is less than or equal to 1, and
2) fibrosis of the papillary dermis is greater than 1
THEN diagnosis is chronic dermatitis.

Each of rules is constructed from a predicate function with an associative triple OAV-(object, attribute, value)-as its argument. Each premise clause typically has the following four components:

<Predicate function> / <Object> / <Attribute> / <Value>
(GREATER_THAN/CNTXT/FIBROSIS-OF-THE-PAPILLARY-DERMIS/1)

Figure 5 shows the sample of knowledge base restored in relational database.


rules : 테이블

RuleNo	THEN	Operator	IF1	IF2	IF3	IF4
47	CLASS == 2	AND	V15 == 0	V19 == 0	V26 == 0	V32 == 1
48	CLASS == 2	AND	V14 == 0	V15 == 0	V20 == 0	V21 == 1
49	CLASS == 2	AND	V14 == 0	V15 == 0	V20 == 0	V21 == 2
50	CLASS == 2	AND	V15 == 0	V18 == 1	V19 == 0	V26 == 0
51	CLASS == 2	AND	V32 == 0	V33 == 2		
52	CLASS == 2	AND	V32 == 0	V33 == 3		
53	CLASS == 3	AND	V32 == 1	V33 == 2		
54	CLASS == 3	AND	V32 == 1	V33 == 3		
55	CLASS == 3	AND	V32 == 2	V33 == 2		

현재 레코드: 52 전체: 81

Figure 5. Knowledge base restored in RDB

As a result of data mining, totally 81 rules were extracted by using APRIORI and C5.0 algorithms which were developed by Agrawal (1993) and Quinlan (1993). Table 4 shows the example of IF-THEN rules.

Table 4. OAV type production rule (IF-THEN rules)

Rule #71
IF 1) the fibrosis of the papillary dermis is equal to 3, AND
2) band-like infiltrate is equal to 1
THEN diagnosis is cronic dermatitis

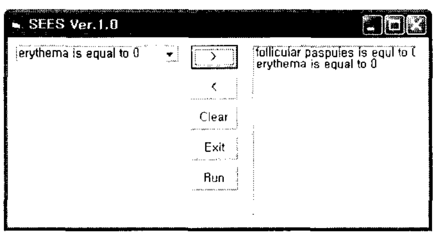
Rule #80
IF 1) the eosinophils in the infiltrate is equal to 2, AND
2) perifollicular parakeratosis is equal to 3
THEN diagnosis is pityriasis rubra pilaris

SQL-based forward inference algorithm was developed by Visual Basic. Therefore, rule consistency check and incompleteness check was easier than other traditional text-driven works. After the construction of knowledge base, SEES ready to execute inference. In this sense, the SEES use forward inference mechanism. Table 5 shows the randomly selected patients' brief clinical data and histopathological data to validate the ability of SEES's inference engine.

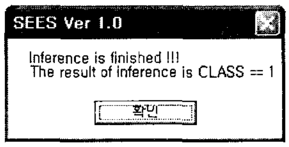
Table 5. Patients' clinical data set for validation

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30	V31	V32	V33	Age	Class
1	1	0	1	0	0	3	0	1	0	0	0	0	1	0	0	1	1	1	0	0	0	0	0	1	0	0	1	0	2	2	1	0	10	6
3	2	1	2	2	0	0	0	2	2	0	0	0	1	0	0	2	3	2	2	2	3	0	3	0	0	0	0	0	0	0	3	1	36	1
1	1	1	0	0	0	1	0	0	0	0	0	1	1	0	2	1	1	1	0	0	0	0	0	0	0	0	3	0	0	0	1	0	5[illegible]	2

Figure 6 shows the dialogue window for information insert, and the final inference result of the 2nd patient's data.



(a) Dialogue window for information insert



(b) Inference result (CLASS==1; psoriasis)

Figure 6. Inference result of SEES

The dialogue window shows the every information possible then the user could select the information for specific patient. As a result, the system showed the final inference result using dialog window (Figure 6(b)).

5. Conclusion

In this study, we proposed an RDB-based knowledge base construction mechanism and SQL-based forward inference algorithm. The proposed mechanism was consisted of the five main components *Library, Knowledge Elicitation, ES Generator, Knowledge Expresser, and Inference Engine*. Our mechanism was based on *data mining, RDB,* and high-speed *forward inference* algorithm, which were mainly aimed at the *reusability of knowledge base*, and *shorten the inference time*. It is expected that our proposed mechanism will have a significant impact on the research domain related to intelligent expert systems. Then, further research topics still remain. First, this expert system shell should be improved as a Web-based multiple decision support system to support the Internet user's flexible decision-making. Second, other intelligent decision support mechanism such as approximate reasoning, case-based reasoning, rough set, fuzzy logic, and etc. may improve the reasoning ability and adaptability of expert system dramatically.

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] Buchanan, B.G., & Shortliffe, E.H. (1984), *Rule-based expert system: the MYCIN experiments of the Stanford heuristic programming projects*, Reading, MA: Addison-Wesley.

[3] Chen, M.S., Han, J., & Yu, P.S. (1996), Data mining: An overview from a database perspective, *IEEE Transactions on Knowledge and Data Engineering*, 8(6), 866-883.

[4] Durkin, J. (1994), *Expert systems: Design and development*, Prentice Hall International, Inc.

[5] Edrees, S., & Rafea, A. (1997), ITAKA: An integrated tool for automatic knowledge

acquisition. Research and development in expert systems XIV, *Proceedings of Expert Systems 97, The Seventeenth SGES International Conference on Knowledge Based Systems and Applied Artificial Intelligence,* Cambridge, December.

[6] Eshelman, L. (1987), MOLE: A knowledge acquisition tool that buries certainty factors. In J. H. Books, & B.R. Gaines (Eds.), *The foundations of knowledge acquisition* (pp.203-217), New York: Academic press.

[7] Giarratano, J.C., & Riley, G.D. (1998), *Expert systems-principles and programming,* Boston: PWS publishing Company.

[8] 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.

[9] Hong, T.P., Wang, T.T., & Chien, B.C. (2001), Mining approximate coverage rules, *International Journal of Fuzzy Systems,* 3(2), 409-414.

[10] Hong, T.P., Wang, T.T., & Wang, S.L. (2000), Knowledge acquisition from quantitative data using the rough-set theory, *Intelligent Data Analysis,* 4, 289-304.

[11] Fujihara, H., Simmons, D.B., Ellis, N.C., & Shannon, R.S. (1997), Knowledge conceptualization tool, *IEEE Transactions on Knowledge and Data Engineering,* 9(2), 209-220.

[12] Kodratoff, Y., & Michalski, R.S. (1983), *Machine learning: An artificial intelligence approach,* Vol.3, California: Kaufmann Publishers.

[13] Marcus, S., & McDermott, J. (1989), SALT: A knowledge acquisition language for propose-and-revise systems. In B.G. Buchanan, & D.C., Wilkins (Eds.), *Reading in knowledge acquisition and learning automating the construction and improvement of expert systems* (pp.263-281), Los Altos, CA: Morgan Kaufmann Publishers.

[14] Michalski, R., & Winston, P. (1986), Variable precision logic, *Artificial Intelligence,* 29, 121-145.

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

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

[17] Rafea, M., & Rafea, A. (1998), Automatic expert system development tool based on hierarchical classification generic task, *The Fourth World Congress on Expert Systems,* Mexic, March 16-20.

[18] Srikant, R., & Agrawal, R, Mining generalized association rules, *Proceedings of the 21st VLDB Conference,* Zurich, Swizerland, 1995.

[19] Tsumoto, S. (1998), Extraction of experts' decision rules from clinical databases using rough set model, *Intelligent Data Analysis,* 2, 215-227.

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

저 자 소 개

김진성 (Kim, Jin Sung)
현재, 전주대학교 경영학부 전임강사로 재직 중이다. 주요 관심 분야는 퍼지와 인공지능 기법을 이용한 의사결정지원시스템, 인터넷 비즈니스, 웹 기반 의사결정지원시스템 등이다.

Phone : +82-63-220-2932
Fax : +82-63-220-2052/2787
E-mail : kimjs@jeonju.ac.kr