

# 인공 신경망과 사례기반추론을 혼합한 지능형 진단 시스템

이길재<sup>†</sup> · 김창주<sup>\*\*</sup> · 안병렬<sup>\*\*\*</sup> · 김문현<sup>\*\*\*\*</sup>

## 요 약

최근 IT 서비스 발달과 함께 고장제어, 고장의 원인분석 등의 복잡한 문제에 대하여 적합한 해결책을 제시할 수 있는 효과적인 진단시스템의 필요성이 커지고 있다. 따라서 본 논문에서는 지능형 진단 시스템분야에서의 시스템의 성능을 향상시키고, 최적의 진단을 수행하고자 사례기반추론과 인공신경망을 혼합한 지능형 진단 시스템을 제안 한다. 사례기반추론은 과거의 사례(경험)를 통해 현재의 제시된 문제를 해결하는 추론방식으로, 지식 획득이 덜 복잡하고, 정형화되기 어려운 규칙이나 문제영역이 불분명한 분야를 효율적으로 추론할 수 있다. 하지만 사례기반추론만을 이용해 추론된 사례는 증상에 대해 다수의 원인을 추론하게 된다. 이때 추론된 증상에 따른 다수의 원인은 동일한 가중치를 가져 불필요한 원인까지 진단해야 하는 문제점이 있다. 이러한 문제를 해결하고자 인공신경망의 오류역전파 학습 알고리즘을 이용하여 증상에 대한 원인들의 쌍을 학습 시킨 후 각각의 증상에 대한 원인의 가중치를 구해 제시된 증상에 대해 가장 발생 가능성이 높은 원인을 찾아내어, 보다 명확하고 신뢰성 있는 진단을 하는 데 그 목적이 있다.

키워드 : 사례기반 추론, 인공신경망, 진단시스템, 전문가 시스템

## The hybrid of artificial neural networks and case-based reasoning for intelligent diagnosis system

Lee Gil jae<sup>†</sup> · Kim Chang Joo<sup>\*\*</sup> · Ahn Byung Ryul<sup>\*\*\*</sup> · Kim Moon Hyun<sup>\*\*\*\*</sup>

## ABSTRACT

As the recent development of the IT services, there is a urgent need of effective diagnosis system to present appropriate solution for the complicated problems of breakdown control, a cause analysis of breakdown and others. So we propose an intelligent diagnosis system that integrates the case-based reasoning and the artificial neural network to improve the system performance and to achieve optimal diagnosis. The case-based reasoning is a reasoning method that resolves the problems presented in current time through the past cases (experience). And it enables to make efficient reasoning by means of less complicated knowledge acquisition process, especially in the domain where it is difficult to extract formal rules. However, reasoning by using the case-based reasoning alone in diagnosis problem domain causes a problem of suggesting multiple causes on a given symptom. Since the suggested multiple causes of given symptom has the same weight, the unnecessary causes are also examined as well. In order to resolve such problems, the back-propagation learning algorithm of the artificial neural network is used to train the pairs of the causes and associated symptoms and find out the cause with the highest weight for occurrence to make more clarified and reliable diagnosis.

Key Words : Case-based reasoning, Artificial neural network, Diagnosis System, Expert system

### 1. 서 론

오늘날 인공지능의 한 분야로 연구 되어왔던 진단시스템의 대표적인 적용 시스템은 전문가 시스템(Expert System)이다. 전문가 시스템은 어떤 특정 분야의 인간 전문가의 전문 지식을 저장하여 필요한 지식을 사람이 응용해서 보여주거나 사용될 수 있도록 구성된 시스템으로써, 진단을 위해

전문가의 전문 지식을 표현하는 방법으로는 규칙(Rules), 의미론적 네트워크(Semantic Network), 프레임(Frame), 사실(Facts), 사례(Case)들로 표현되어진다[1]. 대부분의 전문가 시스템은 IF-THEN 형태의 규칙(Rule)들로 전문 지식을 표현하는 규칙기반 시스템(Rule-Based System)과 과거의 사례(Case)들로 표현된 사례기반 시스템(Case-Based System)이 주를 이룬다. 사례기반추론을 이용한 사례기반 시스템은 과거의 사례(경험)로부터 문제를 해결하는 방법으로, Case Library에 저장된 과거사례(Previous cases)를 검색하여 유사한 사례를 재사용(Reuse)함으로써 제시된 문제를 해결하는 것이다[2][3]. 이처럼 사례기반추론은 인간의 추론 과정과

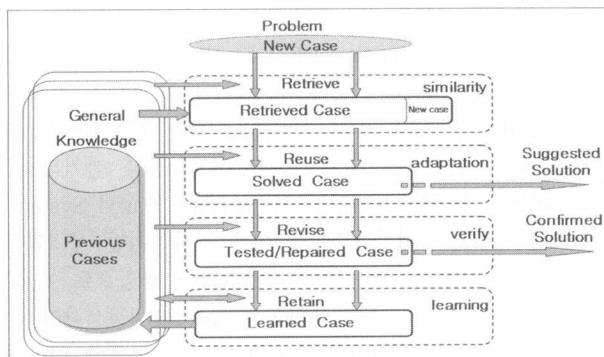
† 준 회원 : 성균관대학교 컴퓨터공학과 석사수료  
 \*\* 준 회원 : 성균관대학교 전기전자 및 컴퓨터공학과 박사수료  
 \*\*\* 정 회원 : 성균관대학교 컴퓨터공학과 박사수료  
 \*\*\*\* 종신회원 : 성균관대학교 전자전기컴퓨터공학부 교수  
 논문접수 : 2006년 10월 2일, 심사완료 : 2007년 11월 16일

매우 유사하다는 점 때문에 정형화되기 어려운 규칙이나, 문제 영역이 불명확한 분야에 효율적으로 활용된다. 하지만 사례기반추론은 유사도를 근거로 추론을 하기 때문에 주어진 문제보다는 많은 Case Library를 유지 하여야한다[4][5]. 그래서 Case Library양이 방대해지고, 그로 인해 검색되는 시간이 지연될 뿐 아니라 도출된 해가 광범위하여 주어진 증상에 대해 정확하고 신뢰성 있는 진단을 하는데 어려움이 있다. 따라서 사례기반추론에 의해 측정된 증상에 유사도와 더불어 좀 더 정확하고 효율적인 진단을 수행하기 위해 인공신경망(Artificial Neural Networks)의 오류역전파(back propagation) 학습 알고리즘을 이용한 학습을 제안하게 되었다. 인공신경망은 인간의 뇌 조직을 모방해 인간처럼 생각할 수 있게 하는 수학적 모델로써 분산된 여러 요소간의 연결로 지식을 표현한다. 신경망의 특징을 살펴보면 다음과 같다. 첫째, 경험하는 환경에 따라 자신의 상태를 스스로 변화 시켜 지식을 만들어 낸다. 둘째, 학습을 통하여 얻은 지식을 일반화 시켜 제시된 문제에 대하여 적절한 해를 도출해 낸다. 셋째, 전형적이지 못한 자료로부터 특정한 특징을 도출해 낸다. 넷째, 주어진 자료들의 특징을 찾아내 각 특징에 따라 분류한다[12]. 따라서 본 논문에서는 사례기반추론을 이용하여 증상에 대해 높은 유사도를 가지는 사례를 검색하고, 검색된 사례의 증상과 원인을 오류역전파 알고리즘을 통해 학습시켜 양질의 Ann library를 생성하여, 제시된 문제에 대해 보다 정확하고, 신뢰성이 있는 진단을 하고자 한다. 2장에서는 사례기반추론과 인공신경망의 기술적 측면에 대해 서술하였으며, 3장에서는 제안시스템 및 적용방안을 4장에서는 실험 및 분석을 5장에서는 결론 및 향후 연구과제를 기술하고자 한다.

## 2. 사례기반 추론과 인공신경망

### 2.1 사례기반 추론

사례기반추론은 과거의 사례를 현재의 비슷한 문제에 적용하여 문제를 해결하는 방법으로, 현재의 주어진 문제를 해결하기 위하여 과거에 동일하거나 유사하게 수행된 적이 있는 문제를 추론하여 도출된 해를 수정하거나 재사용 하여 현재의 문제를 해결하는 인공지능 기법이다[6][15]. (그림 1)



(그림 1) Case-based reasoning architecture

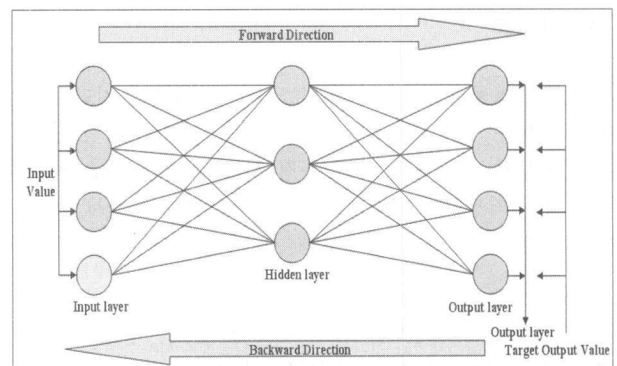
은 Case-based Reasoning Architecture를 보여주고 있다.

사례기반추론은 4단계(검색, 재사용, 수정, 저장)를 통해 제시된 문제를 해결한다. 각각의 Module의 기능은 아래와 같다.

- 1) **Retrieved Module** : 새로운 문제에 대해 기존의 사례와 비교하여 동일하거나 유사한 사례를 찾아내서 이용할 수 있도록 하는 일을 한다.(유사도 평가) Case Library에 저장되어 있는 사례를 빠르게 검색하기 위해서는 Indexing이 필요하다.
- 2) **Reuse Module** : 검색된 유사한 사례들을 수정단계로 거치지 않고, 문제해결에 적용하는 일을 한다.
- 3) **Revise Module** : 검색된 사례가 직면한 문제를 해결하는데 적절하지 못하였을 경우 다양한 변수와 변화에 의해 수정된다.
- 4) **Retain Module** : 확인된 해결안이 Case Library에 저장되는 단계로 지식을 습득하고 학습하는 일을 한다. 이때 검색을 용이하기 위해 색인(Indexing)을 부여하여 저장한다.

### 2.2 인공신경망

인공신경망은 인간의 뇌 조직을 모방해 인간처럼 생각할 수 있게 하는 수학적 모델로써 분산된 여러 요소간의 연결로 지식을 표현하는 것으로, 인간의 신경계를 구성하는 단위세포인 뉴런(neuron)의 집단인 신경조직을 모방하여 간단한 계산 능력을 가진 노드(node) 또는 처리장치(PE: processing element)들 사이를 대규모 연결망으로 연결시킨 네트워크 구조로써 인간 두뇌 모델에 입각하여 병렬적이고 연속적인 정보를 처리하는 모델을 말한다. 따라서 신경망 모델은 인간의 뉴런과 같은 많은 처리단위가 서로 연결(connection)되어 외부로부터 입력되는 여러 정보를 동적인 상황에서 처리할 수 있는 지능형 시스템이다[7]. 신경망의 일반적인 구조는 (그림2)와같이 입력(Input)층, 은닉(Hidden)층, 출력(Output)층의 3계층으로 이루어져 있으며, 하나



(그림 2) Artificial neural networks architecture

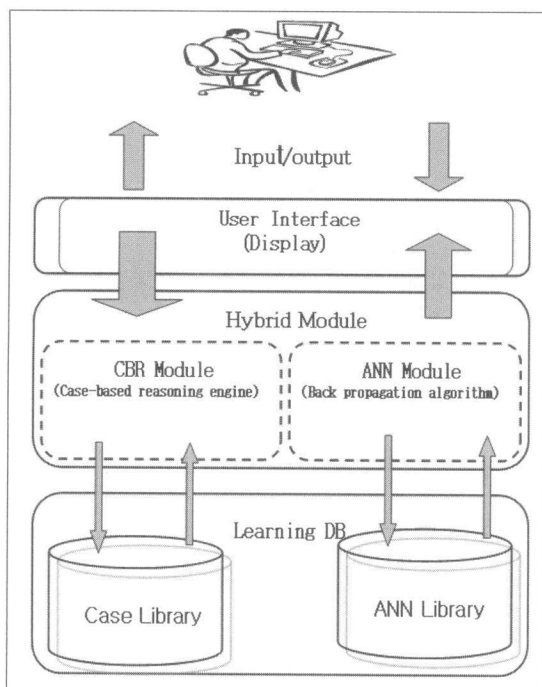
이상의 은닉층을 가져야 한다[8][9].

### 3. 제안시스템 및 적용방안

#### 3.1 제안시스템

본 논문에서는 사례기반추론을 이용해 증상에 대해 가장 높은 유사도 값을 갖는 증상의 사례를 검색할 수 있다. 하지만 사례기반추론을 이용해 검색된 사례의 증상은 다수 또는 하나의 원인을 가지게 된다. 이때 증상과 원인은 1대1의 관계를 가지게 되므로 검색된 증상에 따른 다수의 원인들에 대해 하나하나 진단해야하는 문제점이 발생한다. 이러한 문제를 보완하고자 검색된 사례의 증상과 원인을 신경망의 오류역전과 학습알고리즘을 이용하여 학습시킨 후 각 증상에 따른 원인들의 가중치를 구하였다. 이때 원인의 가중치가 높을수록 검색된 증상에 대해 발생 가능성이 큰 원인이라고 판단할 수 있어, 검색된 증상에 따른 다수의 원인들 중 불필요한 원인까지 진단해야 하는 문제점을 보완하고, 각각의 다수의 원인들 중에 우선순위를 두어 보다 효과적인 진단을 수행 할 수 있다. 따라서 사례기반추론에 의해 검색된 증상의 유사도와 신경망의 학습을 통해 얻어진 원인의 가중치를 혼합하여, 보다 정확하고 신뢰성 있는 진단을 하고자 사례기반추론과 인공신경망을 혼합한 시스템을 제안한다. (그림 3)은 제안하고자 하는 혼합시스템의 구조를 보여주고 있으며, 각 모듈의 기능은 아래와 같다.

- (1) **User Interface Module** : 사용자가 증상의 특징정보를 입력하여 Hybrid Module 엔진과의 상호작용을 통해서 원하는 결과를 Display하는 역할을 한다.



(그림 3) Hybrid system architecture

- (2) **Hybrid Module** : 사례기반추론과 인공 신경망 학습 모듈로써, user Interface Module을 통해 얻어진 사례의 증상의 특징을 Case Library에 저장해 각 사례의 원인과 증상의 특징을 비교/분석하여 각각의 사례에 대한 유사도를 계산한다. 그리고 추출된 각 사례의 원인과 증상의 특징을 신경망의 오류역전과 학습 알고리즘을 이용하여 학습시킨 후 ANN Library를 생성한다. 세부적으로는 CBR Module과 ANN Module로 나뉜다.

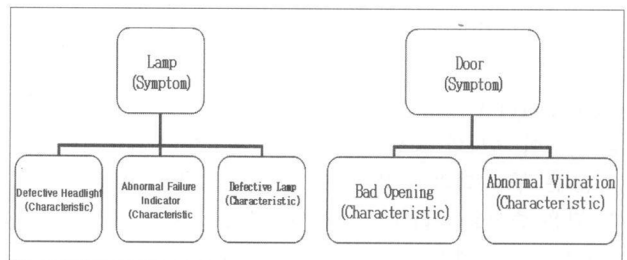
- **Case-Based Reasoning Module** : Case Library에 저장되는 사례는 원인과 증상의 특징으로 이루어져 있으며, Case Library에 저장된 사례는 추론엔진(검색, 재사용, 수정, 저장)을 통해 각 사례의 원인과 증상의 특징에 고유의 key 값을 주어 유사도를 측정 한다.

- **Artificial Neural Networks Module** : 사례기반추론을 통해 유추해낸 각 사례의 증상과 원인의 쌍을 인공신경망을 통해 학습하는 일을 한다. 본 논문에서 사용한 신경망 학습 규칙은 오류역전과 학습알고리즘으로 출력층의 오차 신호를 이용하여 은닉층과 출력층 간의 연결강도와 입력층과 은닉층 간의 연결강도를 조절하여 목표 출력값과 실제 신경망이 산출해낸 출력값의 차이를 최소화하는 방향으로 신경망의 가중치를 조절하는 알고리즘이다. 여기서 사용하는 학습 규칙은 일반화된 델타 규칙을 사용하였다.

- (3) **Learning DB** : Case Library와 ANN Library로 구성되어 있다. Case Library는 증상과 원인의 특징들로 이루어져 있는 각 사례들이 저장되어 있으며, ANN Library는 Hybrid Module을 통해 학습된 증상과 원인의 관계를 나타낼 수 있는 가중치로 이루어져 있다.

#### 3.2 사례기반추론 적용방안 및 분석

각 사례를 이루고 있는 증상의 특징 key값을 유사도를 측정하는 기준으로 선택하였으며, 증상을 이루고 있는 각각의 특징들은 (그림 4)와 같이 계층화구조로 구성하였다. 이와 같은 구성은 불필요한 노드에 대한 검색을 실행하지 않아서 검색시간을 단축할 수 있으며, 유사도 측정에 있어서



(그림 4) 사례 증상의 계층화 구조

$$\begin{aligned}
 & C: \text{입력사례}, C^m: \text{저장사례} \\
 & Case_i = (S_i, C_i), \quad S_i: \text{Symptom} \quad C_i: \text{Cause} \\
 & Sim(C, C^m) = \sum_a W(a) sim_a(C, C^m) \\
 & Case_r = \max_n Sim(C, C^m)
 \end{aligned}$$

(그림 5) K-NN 알고리즘을 이용한 사례 검색

$$\begin{aligned}
 & V_a = \emptyset \wedge V_a^n = \emptyset \rightarrow sim_a(C, C^m) = 0 \quad \dots\dots(1) \\
 & (V_a \neq \emptyset \wedge V_a^n = \emptyset) \vee (V_a = \emptyset \wedge V_a^n \neq \emptyset) \\
 & \rightarrow sim_a(C, C^m) = \frac{1}{cardinality(a)} \quad \dots\dots(2) \\
 & V_a = V_a^n \rightarrow sim_a(C, C^m) = 1 \quad \dots(3) \\
 & V_a \neq V_a^n \rightarrow sim_a(C, C^m) = 0 \quad \dots(4)
 \end{aligned}$$

(그림 6) 증상에 따른 유사도 측정

부모 노드와 자식노드에 걸쳐 측정하기 때문에 보다 정확한 유사도를 구할 수 있다. 사례기반추론을 이용해 유사도를 측정하기 위해서는 먼저 사용자들이 각각의 증상의 특징들에 대한 정보를 문자열로 입력하고, 입력된 증상의 특징과 저장된 증상의 특징을 비교하여 검색하였다. (그림 5)는 입력된 증상의 특징을 저장된 증상의 특징과 비교 하여 가장 근접한 사례를 검색한다.

$C^m$ 은 저장사례를 C는 입력사례를 나타내며, 각 사례는 증상,  $S_i$  와 원인  $C_i$ 로 구성된다. 각 저장사례는 입력사례와의 유사도가 계산되며,  $Sim(C, C^m)$ 은 저장사례  $C^m$ 과 입력사례 C와의 유사도를 나타낸다. 유사도는  $C^m$ 과 C의 증상에서, 각각의 특징 a에 대한 유사도  $sim_a(C, C^m)$ 를 구한 후, 특징의 가중치  $W(a)$ 를 곱하고 이 값을 모두 더하여 구한다. 최대 유사도를 갖는 저장된 사례가 결과로서 검색되며 이를  $Case_r$ 로 부른다. 특징 a에 대한 유사도  $sim_a(C, C^m)$ 은 (그림6)과 같이 구해진다. (그림 6)은 증상의 특징단위로 특징값의 명시여부와 특징값의 일치여부에 따른 유사도의 계산방식을 보이고 있다. 입력된 사례의 특징 a의 값을  $V_a$ , 저장사례  $C^m$ 의 특징 a의 값을  $V_a^n$ 으로 둔다.

(1)은 두 사례 모두 특징 값이 명시되지 않은 경우로서 유사도를 0으로 둔다.

(2)는 두 사례 중 하나의 사례만 특징 값이 명시된 경우로서 유사도는 특징 a가 가질 수 있는 값 집합의 농도(cardinality)의 역수로 둔다. 이는 값이 명시되지 않은 사례의 특징이 하나의 특정한 값을 가질 확률이기 때문이다.

(3)은 두 사례의 특징값이 모두 명시되어 있고 같을 때로서 같은 증상을 나타내므로 유사도를 1로 둔다.

(4)는 두 사례의 특징값이 모두 명시되어 있으나 값이 다른 경우로서 이는 두 사례가 각각 다른 증상을 표현하고 있

음으로 유사도를 0으로 둔다.

### 3.3 인공신경망의 적용방안 및 분석

사례기반추론만을 이용해 검색된 사례는 증상에 대해 다수의 원인을 검색하게 된다. 이때 검색된 증상에 따른 다수의 원인은 동일한 가중치를 가지게 된다. 따라서 검색된 증상에 대한 다수의 불필요한 원인까지 진단해야 하는 문제점이 있다. 이러한 문제를 해결하고자 사례기반추론을 통해 검색된 증상과 원인의 집합을 본 논문에서는 인공지능의 학습 알고리즘인 오류역전과 학습 알고리즘을 사용하여 학습시켰다. 오류역전과 학습 알고리즘은 출력층의 오차 신호를 이용하여 은닉층과 출력층간의 연결강도와 입력층과 은닉층간의 연결강도를 조절하여 목표 출력값과 실제 신경망이 산출해낸 출력값의 차이를 최소화하는 방향으로 신경망의 가중치를 조절하는 알고리즘이고, 일반화된 델타 규칙을 사용한다[10][11]. 즉 총 오류의 합이 오류 기준치 이하가 될 때까지 반복 실행한다. 이러한 학습 규칙은 결국 오류의 최소화를 추구하는 규칙으로서, 오류 척도용으로 잘 알려진 총 에러 제곱의 합인 [식 3.1]을 사용하여 오류 E가 감소하는 방향으로 가중치 변화량을 구하게 된다.

$$E = \frac{1}{2} \sum_k (t_k - y_k)^2 \quad \text{[식 3.1]}$$

$t_k$  : 출력층 뉴런의 k의 목표 출력값  
 $y_k$  : 출력층 뉴런의 k의 실제 출력값

오류역전과 학습 알고리즘의 학습 방법을 (그림 2)의 구조를 이용하여 단계별로 그 과정을 설명하면 다음과 같다 [13][14].

- ① 단계 : 신경망의 연결된 가중치를 임의의 실수로 초기화 한다.  
 m : 입력층의 유닛의 개수  
 n : 출력층의 유닛의 개수  
 h : 은닉층의 노드의 개수
- ② 단계 : 임계치에 대응하는 유닛의 출력값을 -1로 초기화하며, 이 값은 변하지 않는다.
- ③ 단계 : 입력데이터  $x^i$ 와 목표 출력값  $t^i$ 을 입력한다.
- ④ 단계 : 은닉층의 출력값을 구한다.

$$h_j = \frac{1}{1 + \exp\left\{-\sum_{i=0}^m W_{ij}^1 \cdot x_i\right\}} \quad \text{[식 3.2]}$$

$h_j$  : 은닉층의 j번째 출력값  
 $x_i$  : 입력층의 i번째 출력값

⑤ 단계 : 출력층의 값을 구한다.

$$y_j = \frac{1}{1 + \exp\left(-\sum_{i=0}^h W_{ij}^2 \cdot h_j\right)} \quad \text{[식 3.3]}$$

$y_j$  : 출력층의  $j$  번째 노드의 출력값

⑥ 단계 : 출력층의 오차의 값을 계산한다.

$$\delta_j^2 = y_j(1-y_j)(t_j - y_j) \quad \forall j = 1, \dots, N \quad \text{[식 3.4]}$$

⑦ 단계 : 은닉층의 오차의 값을 계산한다.

$$\delta_j = h_j(1-h_j) \sum_{i=0}^n \delta_i^2 \cdot W^{2j} \quad \forall j = 1, \dots, H \quad \text{[식 3.5]}$$

⑧ 은닉층과 출력층의 연결 강도를 변경한다.

$$\Delta W_{ij}^2 = lr \cdot \delta_j^2 \cdot h_i \quad \forall i = 0, \dots, H, j=1, \dots, N$$

$lr$  : 학습률 [식 3.6]

⑨ 단계 : 입력층과 출력층의 연결 강도를 변경한다.

$$\Delta W_{ij}^1 = lr \cdot \delta_j^1 \cdot x_i \quad \forall i = 0, \dots, M, j=1, \dots, H$$

[식 3.7]

⑩ 단계 : 입력 데이터와 목표 출력값을 입력하고, 연결 강도를 변경한다.

⑪ 단계 : 오차의 값이 원하는 범위까지 작아지면 학습을 종료한다.

각 사례의 증상과 원인의 집합의 특징을 학습시키기 위해 증상의 특징값은 0 ~ 1 사이의 실수 값을 가지며, 이 값들은 증상의 원인에 대한 입력패턴 값이다. 각 사례의 증상과 원인의 training data를 살펴보면 <표 1>과 같다.

<표 1> The Training data from the Artificial Neural Network

case \ symptom	ICT	RS	IES	CD	ITM	JCHL	AL	PG
Emergency braking (No characteristics)	0.1	0.0	0.0	0.0	0.2	0.9	0.0	0.3
Power failure (No characteristics)	0.0	0.0	0.2	0.1	0.0	0.0	0.3	0.8
Immobility (No characteristics)	0.0	0.0	0.0	0.8	0.0	0.1	0.0	0.1

[CD: Clogged Drain, Air Leakage: AL, Inoperable Traction Motor : ITM, JCHL : Jumper Cable Heat Loss, PG : Poor Grounding, ICM : Inferior CAM Terminal , RS :Refrigerant Shortage, IEE: Inferior Electronic Equipment]

<표 2> The Learning Results from the Error Back Propagation Learning Algorithm

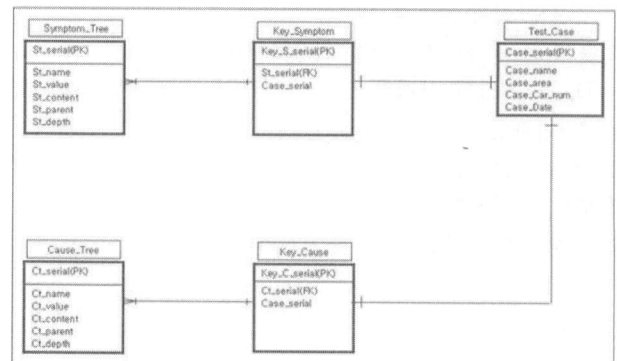
Symptom		Cause	Weight
Immobility	No characteristics	Clogged Drain	0.87
		Poor Grounding	0.11
Power failure	No characteristics	Inferior Electronic Equipment	0.23
		Air Leakage	0.12
Emergency braking	No characteristics	Poor Grounding	0.83
		Jumper Cable Heat Loss	0.94

<표 1>의 training data 즉 증상과 원인의 집합을 오류역전과 학습 알고리즘으로 학습시키기 위해 입력층의 신경세포수를 20으로 하고, 출력층의 세포의 수를 10으로 설계하였다. <표 2>는 증상과 원인의 집합을 5000회 반복하여 학습 시킨 결과 즉, 증상에 따른 원인의 가중치를 보여주고 있다.

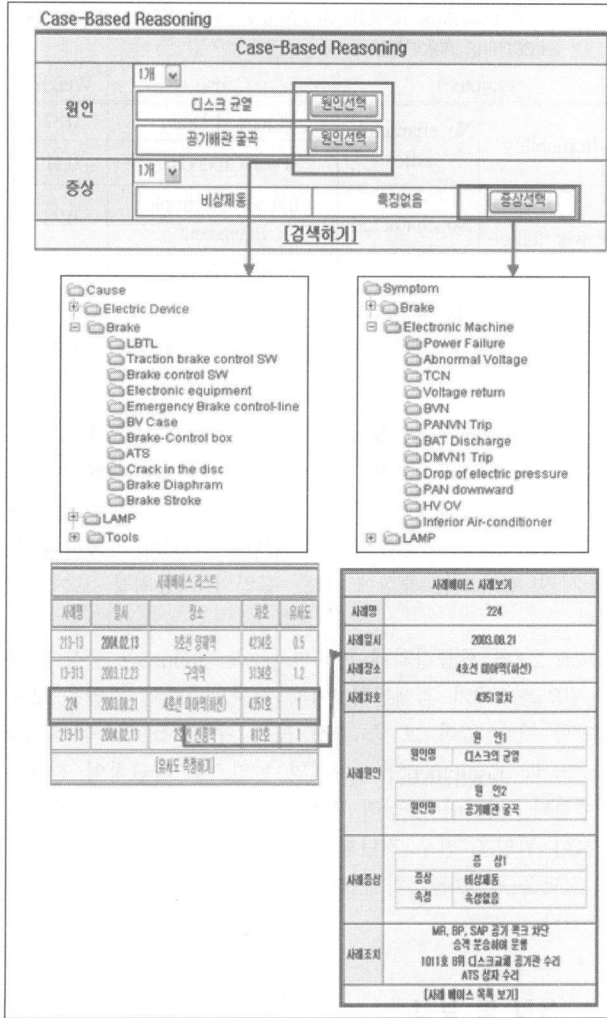
<표 2>의 학습결과에 따르면 power failure의 증상에 따른 원인은 2개가 검색된다. 이때 검색된 2개의 원인을 모두 진단할 필요 없이 그 중 가장 높은 가중치를 갖는 Inferior Electronic Equipment을 1차적인 원인으로 판단하여 진단을 수행한다. 이처럼 증상과 원인의 집합을 신경망의 학습을 통하여 원인을 전부 진단할 필요 없이 직접적인 원인 위주로 진단을 수행하게 하여 진단 수행 속도 및 효율성을 높일 수 있다.

### 4. 실험 및 분석

본 논문에서는 Oracle\_9로 데이터베이스를 구축하였다. 각 테이블은 사례 테이블, 증상 테이블, 원인 테이블, 사례 테이블의 증상과 원인의 특징을 연결해 주는 key\_symptom, key\_cause 테이블을 구성하였다. key\_symptom, key\_cause 테이블에는 각각의 증상과 원인의 특징의 고유 key 값이 저장되어 있다. 이러한 구조는 증상과 원인을 색인화 하여 사례 간의 연관성을 향상시켜 검색 및 평가가 용이하고 사례에 관련 정보를 추가로 저장할 수 있어 이해력을 높일 수 있다. (그림 7)은 데이터베이스의 구조관계를 보여주고 있다.



(그림 7) 데이터베이스의 구조



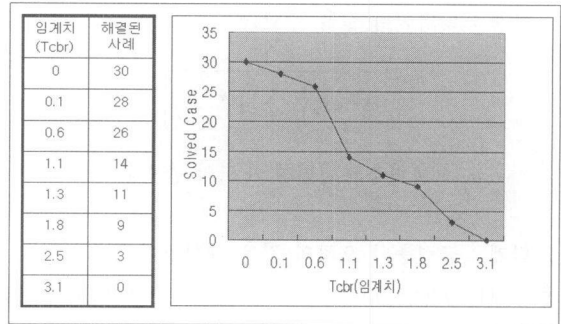
(그림 8) 증상을 이용한 유사도 측정

각 사례의 증상과 원인의 특징을 문자열로 입력하여 hybrid module의 엔진을 단계를 거쳐 유사도와 증상에 따른 가중치를 구할 수 있다. user interface를 통해 증상과 다수의 원인의 특징이 입력 가능하며, 각각의 특징은 트리를 선택하여 입력이 가능하다. 사례기반추론에서의 증상 특징은 유사도를 측정하는데 원인의 특징은 정확도를 측정하는 기준이 된다. 또한 증상과 원인의 집합은 신경망의 학습을 하는 Training data로 사용된다. (그림 8)은 user interface를 통해 증상과 원인을 입력 후 유사도를 측정하는 결과 화면이다.

증상에 의해 저장된 사례를 검색한 후 검색된 사례의 원인을 통해 정확도를 측정을 다음과 같이 하였다.

$D_C$ 는 저장된 사례의 원인이고  $D_C$ 는 입력 사례의 원인이다. 정확도를 검사하기 위해서는 (그림 6)과 같이 유사도를 측정 후 유사한 사례의 원인과 비교 검사를 통해서 정확도를 검사하게 된다. [식 4.1]은 원인이 같을 경우 해당 사례에 대해서 측정값에 1을 증가시킨다.

$$\text{if } D_{C'} = D_C \\ \text{then Count} = \text{Count} + 1 \quad \text{[식 4.1]}$$



(그림 9) 정확도에 따른 문제점 해결 사례

$$\text{error}_p = \sqrt{(o_{p0} - t_{p0})^2 + (o_{p1} - t_{p1})^2 + \dots + (o_{pn} - t_{pn})^2} \\ = \sqrt{\sum_j (o_{p,j} - t_{p,j})^2}$$

$$\text{ERROR} = \sum_p \text{error}_p$$

$\text{error}_p$  : 입력패턴 p에 대한 신경망의 에러

$o_{p,j}$  : 입력패턴 p에 대한 출력층 신경세포 j의 실제 출력

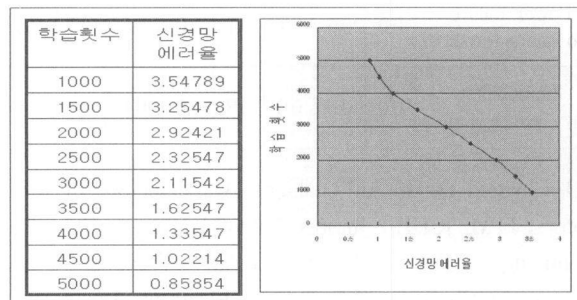
$t_{p,j}$  : 목적패턴 p의 j번째 성분

$\text{ERROR}$  : 목적패턴 p의 j번째 성분

[식 4.2]

사례의 원인에 대한 정확도가 임계치보다 높다고 판단되었을 경우, 추론에 성공했다고 판단하여 문제에 대한 정확도 측정을 하였다. (그림 9)는 입력된 30개의 사례와 저장된 사례간의 유사도 검사 후 각각의 정확도를 계산한 값과 임계치 값을 정한 후, 그 임계치보다 정확도가 높은 경우 해결된 사례(Solved Case)라고 정의한다. 이에 임계치가 증가됨에 따라서 해결된 사례가 반비례함을 알 수 있다. 즉 증상에 의해 검색된 유사도를 바탕으로 좀 더 유사한 사례를 검색하고자 원인에 대한 정확도를 계산하여 보다 효율적인 사례를 검색하게 된다.

신경망의 에러율을 [식 4.2]와 같이 정의한다[14]. 사례기반추론을 통해 검색된 원인과 증상의 집합을 오류역전과 학습 알고리즘을 이용하여 학습 시킨 횟수가 증가 할수록 신경망의 에러율은 반비례함을 볼 수 있다. 다음 (그림 10)은 학습 횟수가 증가 할수록 에러율이 감소 것을 나타내는 그래프이다.



(그림 10) 신경망의 학습 횟수에 따른 에러율

## 5. 결론 및 향후 연구과제

본 논문에서는 지능형 진단 분야의 시스템 성능의 향상과 효율적인 진단을 수행하고자 사례기반추론과 인공지능망의 학습을 혼합한 지능형 진단시스템을 제안하였다. 사례기반추론은 Case Library에 저장된 과거의 사례(Previous cases)를 검색하여 유사한 사례를 재사용(Reuse)함으로써 제시된 문제를 해결하는 것으로 인간의 추론 과정과 매우 유사하다는 점 때문에 정형화되기는 어려운 규칙이나, 문제 영역이 불명확한 분야에서는 효율적으로 활용 될 수 있다. 따라서 사례기반추론을 이용한 진단시스템은 새로 발생하는 문제에 대해서는 가장 근접한 지식을 추출할 수 있다. 또한 신경망의 오류역전과 학습 알고리즘을 통해 증상과 원인의 특징을 학습함으로써, 광범위하게 진단된 사례의 원인과 증상의 관계를 보다 정확하고, 효율적인 진단을 수행할 수 있다.

향후 연구 과제로는 진단분야 뿐만 아니라 다양한 분야에서의 선형 회귀라든지 데이터 마이닝 기법을 통한 예측 시스템, CRM 기법을 이용한 고객 분석 시스템과 자가 학습을 통해 지식의 확장이 가능한 시스템 설계 등 사례기반추론 및 인공지능망의 활용이 가능한 다양한 분야에서의 시스템 설계가 이루어져야 할 것이다.

## 참 고 문 헌

[1] Balakrishnan, K. and Honvar, V., "Intelligent Diagnosis System." Journal of Intelligent System, vol. 8, No.3/4 pp239-290. 1998.

[2] G. Morcou, H. Rivard, A. M. Hanna, "Case-Based Reasoning System for Modeling Infrastructure Deterioration", Journal of Computing in Civil Engineering, vol. 16, No. 2, pp104-114, 2002.

[3] Gavin Finnie, Zhaohao Sun, "Similarity and metrics in case-based reasoning", International Journal of Intelligent System", vol. 17, pp273-287, 2002.

[4] Jieh-Haur Chen and S.C. Hsu, "Hybrid ANN-CBR model for disputed change orders in construction projects", Automation in Construction, vol. 17, pp56-64, 2007.

[5] Peter Wriggers, Marina Siplivaya, Irina Joukova and Roman Slivin, "Intelligent support of engineering analysis using ontology and case-based reasoning" Engineering Applications of Artificial Intelligence, vol. 20, pp709-720, 2007.

[6] 이재식, 전용준, "사례기반추론에 근거한 설비 이상 진단시스템", 한국전문가시스템 학회지, 제1권 1호, pp85-102, 1995.

[7] 고영배, 이재식, "인공신경망을 이용한 고장진단 시스템 개발", 한국지능정보시스템 학회, 제2권, pp475-484, 1994.

[8] Yong Pan, Juncheng Jiang and Zhirong Wang, "Quantitative structure - property relationship studies for predicting flash points of alkanes using group bond contribution method with back-propagation neural network", Journal of Hazardous Materials, vol. 147, pp424-430, 2007.

[9] Daniel J. Fonseca and Daniel Navarrese, "Artificial neural networks for job shop simulation", Advanced Engineering Informatics, vol. 16, pp241-246, 2002.

[10] Ayşegül Güven and Sadık Kara, "Diagnosis of the macular diseases from pattern electroretinography signals using artificial neural networks", Expert Systems with Applications, vol. 30, pp361-366, 2006.

[11] Liang Liang and Desheng Wu, "An application of pattern recognition on scoring Chinese corporations financial conditions based on backpropagation neural network", Computers & Operations Research, vol. 32, pp1115-1129, 2005.

[12] 김대수, "신경망 이론과 응용 [I][II]", 한M&B, 2005.

[13] 김문현, "인공지능", 생능출판사, 2001.

[14] 이상원, "학습하는 기계 신경망", ohm사, 1993.

[15] <http://www.aistudy.co.kr>



### 이길재

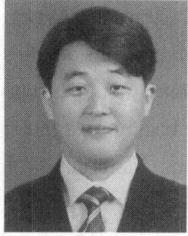
e-mail : utp99@ece.skku.ac.kr  
 2004년 한신대학교 정보시스템공학과 (학사)  
 2006년 성균관대학교 대학원 컴퓨터공학과 수료(공학석사)  
 2007년~현재 한국철도기술연구원 파견근무

관심분야 : 사례기반추론, 인공신경망, 패턴인식, 상형인식, 센서네트워크 등



### 김창주

e-mail : zerocool@ece.skku.ac.kr  
 1997년 성균관대학교 정보공학과 (학사)  
 2000년 성균관대학교 전기전자 및 컴퓨터공학과(석사)  
 2002년~현재 성균관대학교 전기전자 및 컴퓨터공학과(박사수료)  
 관심분야 : 유비쿼터스, 센서네트워크, 추론엔진, 진단시스템등



### 안 병 렬

e-mail : anbr0305@skku.edu

2003년 한신대학교 정보시스템공학과 (학사)

2005년 성균관대학교 대학원 컴퓨터학과  
(공학석사)

2006년~현재 성균관대학교 대학원 컴퓨터  
공학과(박사수료)

관심분야: DRM, 소프트웨어 표절, 인공신경망, 사례기반추론 등



### 김 문 현

e-mail : mhkim@ece.skku.ac.kr

1997년 서울대학교 전자공학과 (학사)

1980년 과학기술원 전기 및 전자공학과  
(공학석사)

1988년 University of Southern

California 컴퓨터공학(공학박사)

1988년~현재 성균관대학교 전자전기 컴퓨터공학부 교수

관심분야: 인공신경망, 패턴인식, 지문인식, 상황인식,  
센서네트워크, 추론엔진 등