

# 지식기반신경망에서 은닉노드삽입을 이용한 영역이론정련화

심 동 희<sup>†</sup>

## 요 약

인공지능의 기호적 방법과 수치적 방법을 결합한 지식기반신경망은 다른 기계학습모델보다 우수한 성능을 나타내고 있다. 그러나 지식기반신경망은 신경망으로 형성된 후 동적으로 그 구조를 변경할 수 없어서 영역이론정련화 기능을 갖추지 못하였다. 지식기반신경망의 이러한 단점을 보완하기 위하여 TopGen 알고리즘이 제안되었으나 삽입된 은닉노드를 모두 입력노드에 연결한 점, 빔탐색을 이용한 점 등의 문제를 안고 있다. 본 논문에서는 TopGen의 문제점을 해소하기 위하여 은닉노드를 다음 하위계층의 노드에 링크시켰으며, 역추적을 허용한 언덕오르기를 이용하는 알고리즘을 설계하였다.

## Theory Refinements in Knowledge-based Artificial Neural Networks by Adding Hidden Nodes

Donghee Shim<sup>†</sup>

### ABSTRACT

KBANN (knowledge-based artificial neural network) combining the symbolic approach and the numerical approach has been shown to be more effective than other machine learning models. However KBANN doesn't have the theory refinement ability because the topology of network can't be altered dynamically. Although TopGen was proposed to extend the ability of KBANN in this respect, it also had some defects due to the linking of hidden nodes to input nodes and the use of beam search. The algorithm which could solve this TopGen's defects, by adding the hidden nodes linked to next lower layer nodes and using hill-climbing search with backtracking, is designed.

### 1. 서 론

영역이론(Domain Theory)이란 어떤 문제영역에 적용될 수 있는 지식집합을 의미한다. 그런데 이 영역이론은 완전성(Completeness), 일관성(Consistency),

정확성(Correctness)이 있어야 한다[6]. 그러나 실제 세계에서 영역이론을 이러한 3가지 측면에서 모두 갖춘 경우는 드물다. 최근에 새로운 접근방법으로 등장한 학습방법에서는 불완전한 영역이론을 이용하여 예제를 해결하면서 추가적인 지식을 습득하여 영역이론을 갖추어 나간다. 불완전한 영역이론을 이와같이 이용하는 것은 연역적 학습에 해당하며, 예제를 해결하면서 추가지식을 습득하는 것은 귀납적 학습에 해당하기 때문에 이 새로운 방법을 통합적 학습(Hybrid

※ 이 논문은 1995년도 전주대학교 학술연구조성비에 의하여 연구되었음.

† 중신희원: 전주대학교 컴퓨터공학과 부교수

논문접수: 1996년 5월 21일, 심사완료: 1996년 8월 16일

Learning)이라 한다[5, 6]. 영역이론을 이와같이 점진적으로 확대하는 것을 영역이론정련화(Theory Refinements)라고 한다[1, 3, 9]. 이 방법에서는 바로 이론뿐만 아니라 데이터로부터 학습을 하는 것으로서 최근에 많은 연구가 이루어지고 있다[1, 2, 3].

한편 지식기반신경망(Knowledge-based Artificial Neural Network)은 인공지능에서의 기호적(Symbolic) 접근방법과 인공신경망에서 사용하는 수치적(Numeric) 접근방법을 결합한 구조다[7]. 이 모델에서는 어떤 문제영역에 대한 이론이 명제논리를 이용한 혼질(Horn-clause) 형태의 규칙집합으로 표현되어 있으면 이를 신경망으로 변환한다. 이때 문제영역에 대한 규칙은 바로 기호적 요소에 해당하며, 신경망이 수치적 요소에 해당한다. 이 규칙들을 신경망으로 변환한 후 예제에 의거하여 역전파 알고리즘[4]을 이용하여 신경망을 학습시킨다. 그러나 지식기반신경망에서는 규칙들을 신경망으로 변환한 후, 신경망 구조를 변경시킬 수 없다. 만약 지식기반신경망에서 다루는 최초의 규칙집합이 완벽하지 않은 경우 규칙집합이 갖고 있는 오류의 수정을 위하여 신경망 구조는 변경될 수 있어야 한다. 이 신경망 구조의 변경이 바로 영역이론정련화에 해당하는 것이다. 본 논문에서는 지식기반신경망에서 신경망 구조 변경방법을 통하여 이론정련화 방법을 제시하고자 한다.

## 2. 지식기반신경망

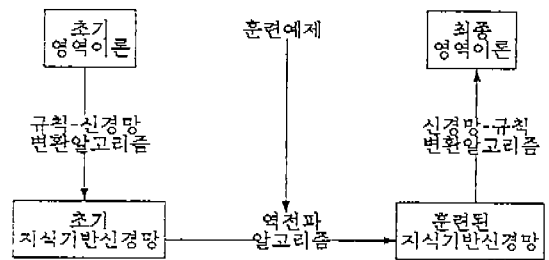
### 2.1 지식기반신경망 알고리즘과 그 한계

지식기반신경망은 규칙기반추론과 신경망을 결합한 접근방법으로서 3개의 알고리즘에 의하여 형성되는데, (그림1)에 나타난 바와 같은 처리절차를 거친다[7]. 먼저 사용자는 문제영역에 대한 초기영역이론과 훈련예제를 제공해야 한다. 여기서 초기영역이론이란 명제논리를 이용하여 표현된 규칙들로서 문제영역에 대하여 완벽하지 않을 수도 있다. 규칙-신경망 변환알고리즘이 이 초기영역이론을 신경망으로 변환하며, 신경망의 노드들의 위상과 연결가중치를 결정한다.

다음에는 역전파 알고리즘[4]을 이용하여 훈련예제로서 신경망을 훈련시킨다. 이 훈련과정에서 신경망의 위상은 변하지 않고 연결가중치를 변경시키게 된

다. 마지막으로 훈련된 지식기반신경망에서 규칙을 도출할 수 있다.

그러나 지식기반신경망에서는 규칙-신경망변환알고리즘에 의하여 지식기반신경망이 일단 수립되면 가중치나 바이어스의 변경만 가능하고 노드의 추가나 삭제는 이루어 질 수 없는 단점을 갖고 있다. 즉 영역이론정련화의 능력이 거의 없다고 볼 수 있다.



(그림1) 지식기반신경망에서의 처리절차  
(Fig. 1) Process in Knowledge-based Artificial Neural network

### 2.2 확장된 지식기반신경망

지식기반신경망의 단점인 영역이론정련화기능 부족을 보완할 수 있도록 지식기반신경망을 확장시키는 연구도 이루어졌다[1, 2]. 첫째 TopGen 알고리즘[1, 2]에서는 훈련된 지식기반신경망을 이용하여 노드의 오류율에 근거하여 노드의 추가위치를 N개 선정한다. 다음에는 빔탐색을 이용하여 신경망을 훈련시킨 후 가장 오류율이 적은 신경망을 도출한다. 그러나 이 알고리즘에서는 오류의 종류 즉 부(False)를 진(True)으로 분류, 진을 부로 분류 등에 관계없이 처리하였고, 추가노드를 모두 입력노드에 연결시켰으며, 빔탐색을 함으로써 지나치게 방대한 탐색이 소요된다는 단점을 안고 있다. 둘째 REGENT 알고리즘[1]에서는 유전자 알고리즘을 도입하여 신경망 노드들의 위상에 변화를 가할 수 있도록 하였다. 이 알고리즘에서 사용된 연산자는 Crossover와 Mutation이었다. 그러나 이 방법 역시 기존의 노드들간의 연결형태에만 변화를 가하기 때문에 영역이론정련 범위에 제약을 갖고 있다. TopGen이나 REGENT에 의하여 도출된 신경망은 제한된 범위이긴 하지만 훈련된 지식기반신경망을 다시 정련화시킨 결과이기 때문에

원래의 KBANN보다 좋은 성능을 보이고 있다.

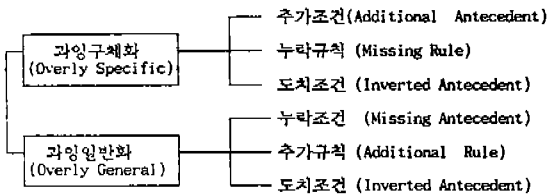
본 논문에서는 TopGen 알고리즘보다 효율적이 될 수 있도록 추가노드의 삽입 방법을 제안하였으며, 탐색시 언덕오르기를 채택하여 빙탐색으로 인한 복잡성부담을 완화시켰다.

### 3. 영역이론정련방법

#### 3.1 영역이론의 오류

예제는 진제와 부제로 분류될 수 있는데 진제란 양의 예제(진을 진으로 분류) 및 음의 예제(부를 부로 분류)를 가르키는 것으로 신경망에 의하여 올바르게 분류된 예제를 말한다. 부제란 진제가 아닌 것을 의미하는 데 양의 예제가 음으로, 음의 예제가 양으로 분류된 경우이다. 이러한 부제는 다시 양부제(false positive)와 음부제(false negative)로 나눌 수 있는데, 양부제는 진이 아닌 것이 진으로 분류된 경우이며, 음부제는 진이 부로 분류된 경우를 의미한다.

한편 명제논리 혼절(Propositional Horn-clause)을 이용하여 표현된 영역이론에서의 오류는 (그림2)에 나타낸 바와 같이 6종류가 있다.



(그림2) 영역이론에서의 오류 종류  
(Fig. 2) Type of Errors in Domain Theory

먼저 과잉구체화란 범주에 해당하는 예제를 부로 분류하는 경우로서 음부제에 해당한다. 이는 규칙의 조건이 추가됨으로 인하여 결론에 대한 and 조건이

강화되었을 때 발생할 수 있고, 또 정당한 규칙이 누락되어 or 조건이 부족하게 되었을 때 발생할 수 있고 또는 조건의 not이 잘못 표기되었을 때 발생할 수 있다. 과잉일반화란 과잉구체화의 반대로서 범주에 해당하지 않는 예제를 진으로 분류하는 경우로서 양부제에 해당한다. 이는 규칙의 조건이 누락됨으로 인하여 결론에 대한 and 조건이 완화되었을 때 발생할 수 있고, 또 잘못된 규칙이 추가되어 or 조건이 완화되었을 때 발생할 수 있고 또는 조건의 not이 잘못 표기되었을 때 발생할 수 있다.

#### 3.2 오류의 처리방법

과잉구체화나 과잉일반화를 수정하는 방법은 <표 1>에 나타낸 바와 같다.

현재 노드가 OR노드인 경우 음부제를 해결하기 위해서는 새로운 노드 N을 만들어 A에 OR로 연결하며, 양부제 해결시에는 새로운 노드 N1과 N2를 만들어 A와 N1을 N2에 AND로 연결시킨다. AND노드에서 음부제를 해결하기 위해서는 새로운 노드 N1과 N2를 만들어 A와 N1을 N2에 OR로 연결시키며, 양부제 해결시에는 새로운 노드 N을 만들어 A에 AND로 연결시킨다. 음부제의 해결은 과잉구체화를 해소시키는 것이며, 양부제의 해결은 과잉일반화를 해소시키는 것이다. 이 방법은 과잉구체화와 과잉일반화를 그 원인별로까지는 다루지 못하지만 전체적인 해결방법이 되는 것이다.

### 4. THRE-KBANN 알고리즘

THRE-KBANN (Theory REfinement for KBANN) 알고리즘은 KBANN에 적용할 수 있는 영역이론정련화를 위한 알고리즘으로서 아래에 나타낸 바와 같다. 이는 3장에서 설명한 6가지 종류의 영역이론 오류에 대처할 수 있는 것이다.

<표 1> 오류의 수정방법  
(Table 1) Methods for Error Correction

현재 노드	음부제(과잉구체화) 해결시	양부제(과잉일반화) 해결시
OR 노드 $A = B \vee C$	$A = B \vee C \vee N$	$A = B \vee C, N2 = A \wedge N1$
AND 노드 $A = B \wedge C$	$A = B \wedge C, N2 = A \vee N1$	$A = B \wedge C \wedge N$

- (1) 훈련사례를 시험집합, 조정집합1, 조정집합2로 임의로 분류한다.
- (2) 시험집합을 이용하여 훈련된 KBANN을 생성한다.
- (3) 조정집합1을 이용하되 다음의 절차에 의하여 신경망을 생성한다.
  - 1) 각 노드의 음부제와 양부제의 값을 0으로 초기화한다.
  - 2) 조정집합1에서 각 부제에 대하여 각 노드에서의 양부제, 음부제 여부를 판단하여 해당값을 증가시킨다.
  - 3) 양부제와 음부제의 합이 가장 큰 노드를 선정한다. 같은 경우는 양부제나 음부제의 비율이 편중된 노드, 입력계층에 가까운 노드 순으로 선정한다.
  - 4) 노드 추가방법에 의거하여 노드를 추가하여 신경망을 생성한다.
- (4) 새로운 신경망을 조정집합2를 이용하여 훈련시키고 오류율이 이전의 신경망보다 높으면 위의 3)단계로 되돌아간다.
- (5) 새로운 신경망의 오류율이 중지조건을 만족하

면 이를 출력하고 만족하지 않으면 3으로 간다.

4.1 노드 추가 위치

3장에서 설명한 양부제 및 음부제의 개념은 출력노드의 입장에서 정의된 것이다. 그런데 위 알고리즘에서 사용된 양부제 및 음부제는 신경망을 구성하는 각 노드 측면에서 정의되어야 한다. 즉 부제에 대하여 어떤 노드가 양인 경우 음으로 바꾸어 진제가 되면 양부제(false positive)라 하고, 노드가 음인 경우 양으로 바꾸어 진제가 되면 음부제(false negative)라고 볼 수 있다. 지식기반신경망에서 활성화값(activation value)은 1에 가깝거나 0에 가깝다. 음인 경우는 활성화값이 0에 가깝고 양인 경우는 활성화값이 1에 가까운 것이다. 이것은 Towell이 실험적으로 입증하여 주장한 “지식기반신경망에서 각 노드는 완전활성화(fully active)되거나 완전불활성화(fully inactive)된다”[7] 성질에 근거한다. 노드의 추가 위치를 결정하기 위하여 부제에 대하여 각 노드는 양부제수, 음부제수에 대한 자료를 유지한다. 그리하여 노드의 추가 위치는 양부제수와 음부제수의 합이 큰 노드를 선택하고, 같은 경우는

〈표 3〉 노드 추가시 링크가중치와 Bias  
(Table 3) Weight and Bias in Node Addition

현재 노드	음부제 해결시	양부제 해결시
OR 노드 $A = B \vee C$ A의 Bias = $-W/2$	$A = B \vee C \vee N$ A의 Bias = $-W/2$ N-A 링크가중치 = W	$A = B \vee C, N2 = A \wedge N1$ A의 Bias = $-W/2$ A-N2 링크가중치 = W N1-N2 링크가중치 = W N2의 Bias = $-3W/2$
	- B, C의 다음 하위계층에 연결 - B, C가 입력계층이면 입력계층과 연결	- B, C와 같은 계층의 다른 노드들과 연결 - 다른 노드가 없으면 다음 하위 계층에 연결
AND 노드 $A = B \wedge C$ A의 Bias = $-3W/2$	$A = B \wedge C, N2 = A \vee N1$ A의 Bias = $-3W/2$ A-N2 링크가중치 = W N1-N2 링크가중치 = W N2의 Bias = $-W/2$	$A = B \wedge C \wedge N$ A의 Bias = $-5W/2$ N-A 링크가중치 = W
	- B, C와 같은 계층의 다른 노드들과 연결 - 다른 노드가 없으면 다음 하위 계층에 연결	- B, C의 다음 하위계층에 연결 - B, C가 입력계층이면 입력계층과 연결

W: 지식기반신경망에서 링크의 기본가중치

양부제나 음부제의 비율이 편중된 노드를 선정한다.

#### 4.2 노드의 추가방법

새로운 노드의 추가위치가 결정되면 노드는 <표1>에 나타낸 바와 같이 추가가 되는 데 이때의 Bias, 링크가중치 등은 <표3>과 같이 한다.

##### (1) 새로운 노드로의 연결노드와 링크 가중치

새로운 노드는 하위계층의 노드들로부터 링크를 갖도록 한다. 이때 하위계층의 노드들은 다음과 같이 결정한다. 먼저 OR노드에서 음부제 해결시와 AND노드에서 양부제 해결시에는 B, C의 다음 하위계층에 연결하되 만일 B, C가 입력계층이면 입력계층과 연결한다. 한편 AND노드에서 음부제 해결시와 OR노드에서 양부제 해결시에는 B, C와 같은 계층의 다른 노드들과 연결하되 B, C와 같은 계층의 다른 노드가 없으면 다음 하위계층의 노드들과 연결한다.

이때 링크 가중치는 노드의 추가위치를 결정하기 위하여 수집한 음부제, 양부제의 자료에 의거하여 음부제와 양부제의 비율에 의거하여 결정한다. 그리하여 양부제나 음부제의 비율이 유사하면 +1 아니면 -1의 가중치를 갖도록 한다. 즉 노드가 추가되는 위치가 양부제의 비율이 높았다면 하위계층의 노드중 양부제의 비율이 높은 노드로부터의 링크 가중치는 W로 하며, 하위계층의 노드중 양부제의 비율이 낮은 노드로부터의 링크 가중치는 -W로 하며, 양부제의 비율이 음부제와 대등한 노드로부터의 링크 가중치는 0으로 한다. 이와 반대로 노드가 추가되는 위치가 음부제의 비율이 높았다면 하위계층의 노드중 음부제의 비율이 높은 노드로부터의 링크 가중치는 W로 하며, 하위계층의 노드중 음부제의 비율이 낮은 노드로부터의 링크 가중치는 -W로 하며, 음부제의 비율이 양부제와 대등한 노드로부터의 링크 가중치는 0으로 한다. 한편 노드가 추가되는 위치가 양부제의 비율과 음부제의 비율이 유사한 경우는 하위계층의 노드중 양부제의 비율이 높은 노드로부터의 링크 가중치는 -W로 하며, 하위계층의 노드중 양부제의 비율이 낮은 노드로부터의 링크 가중치는 -W로 하며, 양부제의 비율이 음부제와 대등한 노드로부터의 링크 가중치는 W로 한다.

하위계층의 노드중 양부제나 음부제가 발생하지

않는 경우는 링크가중치를 0으로 한다.

##### (2) 새로운 노드의 Bias

OR노드에서 음부제 해결시와 AND노드에서 양부제 해결시 노드 N의 Bias 및 AND노드에서 음부제 해결시와 OR노드에서 양부제 해결시 노드 N1의 Bias 모두  $-(2N-1)W/2$  (N는 하위계층 노드로부터의 링크수)로 한다. 또한 OR노드에서 음부제 해결시 A의 Bias는  $-W/2$ 로, 양부제 해결시 N2의 Bias는  $-3W/2$ 로, AND노드에서 음부제 해결시 노드 N2의 Bias는  $-W/2$ 로, 양부제 해결시 A의 Bias는  $-5W/2$ 로 한다.

##### (3) 새로운 노드로부터 상위계층 노드로의 링크 가중치

새로운 노드로부터 상위계층 노드로의 링크 가중치는 본래의 노드가 갖고 있던 가중치를 따르도록 한다.

## 5. THRE-KBANN의 성능평가

### 5.1 평가대상 알고리즘

본 연구에서 제시한 THRE-KBANN의 성능을 TopGen[1, 2]과 비교하고, 또한 간단한 확장신경망[2]과 비교한다. 간단한 확장신경망이란 노드들을 하나씩 추가하여 이 노드들을 모든 입력노드 및 출력노드와 연결한 것이다.

### 5.2 평가대상 문제영역

유전학에서 나타나는 두가지 문제 즉 프로모터(Promoter) 인식문제와 접목점(Splice-Junction) 결정문제를 평가에 이용하였다. 분자생물학에서 DNA는 뉴클레오티드라고 불리는 {A, G, T, C} 문자집합에서 선택된 문자들의 선형배열이다. 이 DNA는 RNA로, RNA는 다시 단백질(Protein)으로 복제가 된다. 이 단백질은 세포의 활성적 요소에 해당한다. 그런데 DNA에 있는 명령에 근거하여 단백질이 생성되기 때문에 DNA에서의 오류는 잘못된 단백질을 유발하며 선천적 질병으로 나타나게 된다. 따라서 인간의 DNA배열을 아는 것은 선천적 질병을 치료하기 위한 첫 단계가 된다. DNA배열에서는 위치를 명시하는데 특별한 표기법을 이용한다. 즉 생물학적으로 의미있

는 고정된 점(기준점)을 기준으로 위치를 숫자화한다. 음수는 기준점에 앞서는 위치를 나타내고 양수는 기준점 뒤의 위치를 나타낸다. 예를 들어 @3 'AGTC'는 기준점에서 오른쪽으로 3번째의 뉴클레오티드는 A이며 다음에는 GTC임을 의미한다.

(1) 프로모터 인식문제

DNA의 배열중 프로테인으로 복제되는 부분을 유전인자(Genes)라고 한다. DNA배열의 이해를 위해서는 먼저 이 유전인자의 시작부분과 끝을 인식해야 한다. 유전인자의 끝부분은 3개의 뉴클레오티드에 의하여 표시가 되기 때문에 쉽게 인식이 되는 데 시작부분의 인식은 쉽지가 않다. 프로모터란 유전인자(Genes) 앞에 선행하는 짧은 DNA배열을 의미한다. 따라서 프로모터를 인식하면 유전인자의 시작부분을 알게 된다. 프로모터 DNA배열은 뉴클레오티드 57개로 구성되는데 이것이 프로모터인지 아닌지를 판단할 수 있는 <표4>에 나타난 바와 같은 일부의 규칙[9]들이 알려져 있는데 이 규칙들이 완벽하지는 않다.

프로모터 인식문제에 대한 훈련예제는 프로모터에 해당하는 53개의 DNA배열과 프로모터가 아닌 53개의 DNA배열로 구성되어 있다. 최초의 지식기반신경

망은 106개의 예제를 제대로 인식하지 못하지만 훈련을 거쳐 인식율이 높아진다.

(2) 접목점 결정(Splice-Junction Determination) 문제

접목점(Splice junction)은 프로테인 생성과정중 DNA 배열에서 불필요한 DNA가 제거되는 지점들이다. 이 DNA 배열에는 접목후에도 유지되는 부분에 해당하는 엑손(Exons), 접목후에 없어지는 부분에 해당하는 인트론(Introns)이 있다. 그래서 E/I 부분이라 불리우는 엑손/인트론 경계와 I/E 부분이라 불리우는 인트론/엑손 경계를 인식해야 하는 데 이러한 경계인식문제를 접목점 결정문제라 한다. <표5>에는 접목점 결정에 대한 불완전한 규칙들을 나타냈다.

5.3 평가결과

프로모터 인식문제와 접목점 결정문제에 THRE-KBANN을 적용한 결과는 <표6>에 나타난 바와 같다. 이 결과 본 논문에서 제안한 THRE-KBANN의 프로모터 인식문제와 접목점 결정문제에 대한 오류율은 각각 1.98%, 4.05%로서 TopGen의 오류율 2.06%, 4.17%보다 더 낮은 성능이 우수함을 보이고 있다.

<표 4> 프로모터 인식 규칙  
<Table 4> Rules for Promoter Recognition

- promoter :- contact, conformation.
- contact :- minus-35, minus-10.
- minus-35 :- @-37 ' CTTGAC-'.    minus-35 :- @-37 '-TTG-CA'.
- minus-35 :- @-37 ' -TTGACA'.    minus-35 :- @-37 '-TTGAC-'.
- minus-10 :- @-14 'TATAAT--'.    minus-10 :- @-14 '-TA-A-T-'.
- minus-10 :- @-14 '-TATAAT-'.    minus-10 :- @-14 '--TA---T'.
- conformation :- @-45 'AA--A'.
- conformation :- @-45 'A---A', @-28 'T---T-AA--T-', @-04 'T'.
- conformation :- @-45 'A----T', @-27 'T----A--T-TG', @-01 'A'.
- conformation :- @-47 'CAA-TT-AC', @-22 'G---T-C', @-08 'GCGCC-CC'.

〈표 5〉 접목점 인식규칙  
 〈Table 5〉 Rules for Splice-Junction Determination

E/I : -@3 'MAGGTRAGT', 'not(E/I-stop).  
 E/I-stop :: -@-3 'TAA'.    E/I-stop :: -@-4 'TAA'.    E/I-stop :: -@-5 'TAA'.  
 E/I-stop :: -@-3 'TAG'.    E/I-stop :: -@-4 'TAG'.    E/I-stop :: -@-5 'TAG'.  
 E/I-stop :: -@-3 'TGA'.    E/I-stop :: -@-4 'TGA'.    E/I-stop :: -@-5 'TGA'.  
 I/E : -pyrimidine-rich, @-3 'YAGG', not(I/E-stop).  
 pyrimidine-rich : -6 of' (@-15 'YYYYYYYYY').  
 For i from ((-30 to -1) and (+1 to +30))  
     {@<i> 'Y' :: - @<i> 'C'.    @<i> 'Y' :: - @<i> 'T'. }  
 I/E-stop :: -@1 'TAA'.    I/E-stop :: -@2 'TAA'.    I/E-stop :: -@3 'TAA'.  
 I/E-stop :: -@1 'TAG'.    I/E-stop :: -@2 'TAG'.    I/E-stop :: -@3 'TAG'.  
 I/E-stop :: -@1 'TGA'.    I/E-stop :: -@2 'TGA'.    I/E-stop :: -@3 'TGA'.

〈표 6〉 인식오류율 결과  
 〈Table 6〉 Error Rate Comparison

알고리즘	오류율	
	프로모터 인식문제	접목점 결정문제
THRE-KBANN	1.98	4.05
TopGen	2.06	4.17
간단한 확장신경망	2.12	4.53
KBANN	2.31	4.58

6. 결 론

KBANN은 인공지능의 대표적인 접근방법인 기호적 방법과 수치적 방법의 장점을 결합한 학습구조를 보유하여 그 성능이 이미 입증되었다. 그러나 실세계의 대부분의 초기지식은 근사적으로 옳바르기 때문에 영역이론정련화가 사실 필요하다. 그런데 KBANN은 영역이론정련화 능력을 보유하고 있지 않다. KBANN의 이러한 단점을 보완하기 위하여 TopGen에서는 오류의 원인이 되는 노드를 찾아 노드를 추가하되 이 노드를 입력노드들 모두와 연결시켰다. 또한 노드를 추가할 때 빔탐색을 이용함으로써 시간복잡성의 부

담을 안고 있다. 본 논문에서 제안한 THRE-KBANN은 TopGen의 이러한 단점을 보완하였다. 그리하여 추가된 노드를 입력노드 대신 하위노드와 연결하고 링크의 가중치를 노드간의 성격에 따라 차별성을 부여함으로써 그동안의 학습효과가 반영되도록 하였다. 또한 노드를 삽입할 때 역추적을 허용한 언덕오르기 탐색을 이용하여 빔탐색의 부담을 완화하였다. 유전학에서 발생하는 두 문제영역에 이 알고리즘을 적용하여 본 결과 이러한 개선으로 인하여 인식율에서 KBANN, TopGen보다 효율적인 성능을 보였다.

참 고 문 헌

[1] D. W. Opitz, 'An Anytime Approach to Connectionist Theory Refinement: Refining the Topologies of Knowledge-Based Neural Networks', PhD thesis, University of Wisconsin-Madison, 1995.  
 [2] D. W. Opitz, and J. W. Shavlik, "Heuristically Expanding Knowledge-Based Neural Networks", In Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence, 1993,

pp. 1360-1365, 1993.

[3] D. Ourston and R. Mooney, "Theory Refinement Combining Analytical and Empirical Methods", *Artificial Intelligence*, Vol. 66, pp. 273-309, 1994.

[4] D. E. Rumelhart, G. E. Hinton, and J. R. Williams, "Learning Internal Representations by Error Propagation", Vol. 1, pp. 318-363, MIT press, 1986.

[5] J. W. Shavlik and G. G. Towell, "An Approach to Combining Explanation-based and Neural Learning Algorithms", *Connection Science*, Vol. 1, No. 3, pp. 233-255, 1989.

[6] S. B. Thrun and T. M. Mitchell, "Integrating Inductive Neural Network Learning and Explanation-Based Learning", In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, pp. 930-936, 1993.

[7] G. G. Towell, "Symbolic Knowledge and Neural Networks: Insertion, Refinement, and Extraction", PhD thesis, University of Wisconsin-Madison, 1991.

[8] G. G. Towell, & J. W. Shavlik, "Using Symbolic Learning to Improve Knowledge-Based Neural Networks", *Proceedings of AAAI*, pp. 177-182, 1992.

[9] G. G. Towell, & J. W. Shavlik and M. Noordewier, "Refinement of Approximate domain theories by Knowledge-based Neural Networks", In *Proc. of the 8th National Conference on Artificial Intelligence*, pp. 861-866, Boston, MA, 1990.



**심 동 희**

1980년 서울대학교 산업공학과 졸업(학사)  
 1982년 서울대학교 대학원 산업공학과 졸업(공학석사)  
 1994년 고려대학교 대학원 전산과학과 졸업(이학박사)  
 1982년~1985년 국토개발연구원  
 1985년~1990년 해운산업연구원  
 1990년~현재 전주대학교 컴퓨터공학과 부교수  
 관심분야: 기계학습, 전문가 시스템