

SymCSN : 유연한 지식 표현 및 추론을 위한 기호-연결주의 모델

SymCSN : a Neuro-Symbolic Model for Flexible Knowledge Representation and Inference

노 희 섭* 안 흥 섭** 김 명 원*
(He-sub Rho) (Hong-sub Ahn) (Myung-Won Kim)

요 약 기존의 기호주의적 추론 시스템은 경직성 문제로 인하여 유연성을 결여하고 있다. 이는 기호주의적 지식표현 체계가 지식의 유연한 의미구조를 충분히 반영하고 있지 못할 뿐 아니라 추론 방법도 논리를 바탕으로 하기 때문이다. 이러한 문제를 해결하기 위하여, 우리는 최근 인공지능망에 기반한 유연한 지식표현과 추론을 위한 연결주의적 의미망(CSN)을 제안한 바 있다. CSN은 인간의 유사성과 연관성에 기반하여 근사추론과 상식추론을 수행할 수 있다.

그러나 CSN 모델에서는 상위개념간의 관계를 표현하는 데 있어서 단순한 전향 신경망을 이용함으로써 상위개념간의 일반적이고 구조화된 관계를 표현하거나 변수의 표현 및 바인딩의 어려움과 같은 문제점이 있었다. CSN모델의 이런 문제점을 해결하기 위해 본 논문에서는 상위개념간의 일반적이고 구조화된 지식표현을 가능하게 하고 추론이 용이한 기호주의 표현 체계와 이 표현 체계 안에서 의미구조를 표현하고 학습할 수 있는 연결주의 학습 모델인 CSN을 결합한 기호-연결주의 통합 시스템 SymCSN(Symbolic CSN)을 제안하고, 실험을 통하여 제안한 시스템이 인간과 유사한 유연한 지식표현과 추론을 위한 모델임을 보인다.

주제어 지식 표현, 유연한 추론, 유사성, 연관성, 신경망, 기호논리

Abstract Conventional symbolic inference systems lack flexibility because they do not well reflect flexible semantic structure of knowledge and use symbolic logic for their basic inference mechanism. For solving this problem, we have recently proposed the 'Connectionist Semantic Network(CSN)' as a model for flexible knowledge representation and inference based on neural networks. The CSN is capable of carrying out both approximate reasoning and commonsense reasoning based on similarity and association.

However, we have difficulties in representing general and structured high-level knowledge and variable binding using the connectionist framework of the CSN. In this paper, we propose a hybrid system called SymCSN(Symbolic CSN) that combines a symbolic module for representing general and structured high-level knowledge and a connectionist module for representing and learning low-level semantic structure. Simulation results show that the SymCSN is a plausible model for human-like flexible knowledge representation and inference.

Keywords knowledge representation, flexible inference, similarity, association, neural networks, symbolic logic

* 숭실대학교 정보과학대학 컴퓨터학부
School of Computing, College of Information Science,
Soongsil Univ.

** (주)zoi comm. (zoi communication inc.)
연구세부분야 : 인공지능

주소 : 서울특별시 동작구 상도5동 1-1 숭실대학교 컴퓨터학부
우편번호 : 156-743
전화 : 02)813-8755
Fax : 02)825-1087
E-mail : syrhee@synapse.soongsil.ac.kr
mkim@computing.soongsil.ac.kr

1. 서론

기존의 기호주의적 추론시스템은 추론 과정에 있어서 지식표현 체계가 표현하고자 하는 지식의 의미구조를 잘 반영하지 못하고 또 기호논리를 기반으로 한 추론 방법을 사용함으로써 발생하는 경직성 문제(brittleness problem)로 유연성이 크게 결여되어 있다. 경직성의 의미는 일반적으로 지식기반 시스템에서 주어진 문제에 대하여 적용 가능한 지식이 부족하거나 그 문제에 꼭 맞는 지식이 없을 경우 문제에 대해서 시스템이 추론을 하지 못하거나 잘못된 결론을 생성하는 것을 의미한다. 이와 같은 경직성 문제는 기호주의적 지식기반 시스템의 응용성을 크게 감소시키는 원인이 된다. 반면에 인간의 추론기능은 이러한 경직성 문제가 없으며, 본질적으로 상황에 의존적이며 매우 유연하다.

기존의 기호주의적 접근방법의 문제점을 해결하기 위해 신경망에 기반한 추론 시스템들이 연구되고 있고(1, 2, 3, 4), 신경망을 이용하여 개념이나 관계 등을 표현하고 그 안에 의미구조를 반영함으로써 보다 유연한 추론 시스템 구성을 하고 있다. 이러한 연구들은 기호 처리를 위해 사용되는 신경망의 형태에 따라서 크게 지역적 방법과 분산적 방법으로 구분된다. 지역적 방법은 하나의 개념을 표현하기 위해 하나의 노드를 사용하고 노드간의 연결은 직접적으로 상응하는 개념간의 연결을 나타낸다. 전향 신경망(feedforward neural network)을 이용하여 학습과 근사 추론이 가능한 전문가 시스템인 MACIE(1)가 이에 해당한다. 분산적 방법은 하나의 개념이 분산된 여러 개념들의 활성화 패턴에 의해 나타내어지며 각 활성화 패턴은 하나 이상의 개념에 속할 수 있다. 서로 상호 작용하는 다층 신경망에 의하여 문맥 의존적인 문자인식을 할 수 있는 IAC 모델(2), 유사성(similarity)에 기반한 상식 추론(common sense reasoning)을 위해 제안한 CONSYDERR 모델(3), 본 연구팀에서 제안한 CSN(Connectionist Semantic Network) 모델(4)이 분산적 방법에 해당한다.

CONSYDERR와 CSN 모델은 구조적인 면에서는 지식 표현을 위해 상위개념과 각 상위개념에 대한 하위개념을 표현하는 두 개의 층으로 이루어진 점은 유사하다. CONSYDERR 모델이 상위개념간의 유사성에만 의존하는 데 비하여 CSN 모델은 개념간의 유사성뿐만 아니라 연관성을 이용해 보다 유연한 추론을

수행한다. 그러나 위 두 모델은 상위개념 간의 관계를 단순한 전향 신경망을 이용해 규칙 형태로 구현함으로써 일반적이고 구조화된 지식을 표현하기 어렵고 또 변수의 표현 및 변수 바인딩 문제를 해결하기 어려운 문제점을 가지고 있다.

따라서 본 논문에서는 CSN처럼 신경망에 기반한 유사성과 연관성을 이용한 유연한 추론 메커니즘과 상위개념간의 복잡하고 구조적인 지식 표현과 추론이 용이한 기호주의적 접근방식을 결합한 통합 지식표현 구조를 제안한다. 또한 본 논문에서 제안한 지식표현 구조가 기호주의적 접근방식이 가지는 경직성 문제와 연결주의적 접근방식이 상위개념간의 구조화되고 복잡한 관계를 표현하기 어려웠던 문제점을 해결할 수 있음을 보인다.

본 논문의 2절에서는 유연한 추론을 위한 기존의 대표적 모델인 CSN, 3절에서는 유연한 추론을 위해 본 논문에서 제안한 지식표현 구조 및 추론 과정, 4절에서는 추론 예제를 통한 실험에 대해서 기술한다. 마지막으로 5절에서는 결론과 향후의 연구과제를 기술한다.

2. SymCSN의 구조

2.1 CSN

CSN은 프레이밍(5)과 실어증(6,7)에 관한 연구에서 볼 수 있듯이 인간의 기억 및 회상 메커니즘의 근본적 특징인 개념간의 유사성과 연관성을 이용해 인간과 같은 유연한 추론을 수행할 수 있는 지식표현 구조이다.

CSN은 상위개념을 나타내기 위한 CL(Concept Level)과 각 상위개념에 대한 하위개념을 나타내기 위한 SL(Subconcept Level)의 두 층으로 구성된다(그림 1). CL층은 상위개념들 간의 관계를 규칙의 형태로 표현할 수 있도록 전향 신경망으로 구성되어 있다. 규칙의 조건부가 될 수 있는 것이 신경망의 입력 노드가 되고 결론부가 신경망의 출력 노드가 된다. 그리고 입력 노드와 출력 노드 사이에는 다수 개의 은닉 노드를 가질 수 있다. CSN에서 상위개념과 하위개념들과의 관계는 CL과 SL간의 연결 가중치로 표현된다. 따라서 어떤 상위개념의 중요한 하위개념일수록 가중치가 크고, 무관한 하위개념일수록 가중치의 값이 0에 가깝게 된다. SL은 CL에 속해 있는 상위개념들에 대한 하위개념들을 노드로 하는 연속적 홉필드 네트워크(continuous Hopfield Network)로

구성되어 있어서, 각 하위개념간의 연관 정도가 노드간의 연결 가중치로 표현된다. 연속적 홉필드 네트워크를 통한 지식의 분산 표현 구조는 동적으로 변하는 정보의 확실도를 표현할 수 있고, 적은 수의 개념들 만으로도 고용량의 정보나 지식을 나타낼 수 있으며, 개념들 사이의 유사성이나 연관성 정도를 보다 효율적으로 구현할 수 있는 장점을 가진다(8).

CSN의 전체 수행단계는 ①하향단계(Top-down phase), ②안정단계(Settling phase), ③상향단계(Bottom-up phase), ④CL-전파단계(CL-propagate phase)의 4단계로 구성된다(그림 1). 각 단계의 보다 자세한 수행과정은 3절에서 설명한다.

CSN에서의 학습은 두 단계에 걸쳐 이루어진다. 첫 번째 단계는 각 상위개념에 대한 하위개념들의 관계를 학습하는 단계로 CL과 SL노드간의 연결 가중치를 학습한다. 두 번째 단계는 활성화된 상위개념에 의해서 발화되는 하위개념들의 값을 입력으로 서로 연관 있는 개념들을 학습하는 단계로 SL내의 노드간의 연결 가중치를 학습한다. 이 단계의 학습은 기본적으로 Hebbian 학습법칙을 통하여 이루어진다. 보다 자세한 학습과정은 [4]를 참조한다.

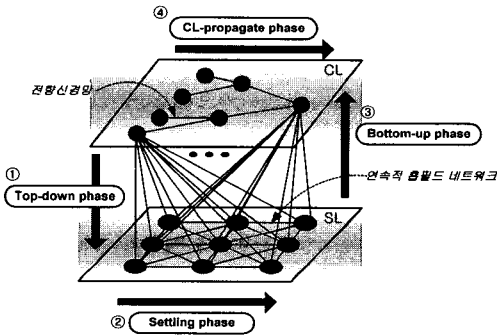


그림 1. CSN의 구조 및 수행단계

CSN 모델은 상위단계의 지식 표현을 위한 이질적 메커니즘인 기호주의적 접근방법과 하위단계의 지식 표현을 위한 동질적 메커니즘인 연결주의적 접근방법간의 유기적인 상호작용을 통해 인간의 유연한 추론 과정을 모델링 하고 있다. 그러나 CSN은 상위단계의 지식 표현을 위해 전향 신경망을 사용한다. 예를 들어 '깃털이 있고(has-feathers), 날 수 있고(can-fly), 알을 낳으면(lay-eggs) 새이다'라는 규칙을 CSN의 CL에서 표현한다면 그림 2와 같다. 규칙은 변수(variable)를 포함할 수 없는 명제논리(propositional

logic) 정형식(well-formed formula)으로 표현되기 때문에 일반적이고 구조화된 개념간의 관계를 표현하고 추론하기에 어려운 점이 있다. 특히 변수의 바인딩 처리가 곤란하며 또한 추론 과정 중 동적으로 새로운 지식의 추가나 삭제의 경우 전향 신경망의 노드나 연결 가중치를 다시 설정해야 하는 어려움이 있다. 이러한 문제점의 근본적인 원인은 상위단계 지식 표현과 추론을 위한 기호 처리과정이 연결주의적 틀 안에서 이루어지고 있기 때문이다. 따라서 본 논문에서는 CSN의 상위단계의 지식표현 및 추론 과정을 기호주의적 방법으로 처리함으로써 이러한 문제점을 해결할 수 있는 확장된 지식표현 구조를 제안한다.

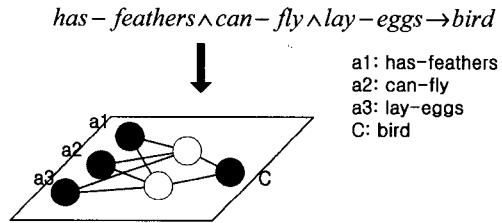


그림 2. CSN의 CL에서 지식 표현

2.2 논리적 추론 구조

본 논문에서는 기존의 CSN이 가지고 있는 문제점을 보완하기 위하여 논리를 기반으로 한 기호주의적 추론 구조(SLM: Symbolic Logic Module)를 CSN과 통합함으로써 SymCSN을 구현한 것에 대하여 기술한다. 제안하는 지식표현 구조는 CSN과 유사하게 상위개념을 표현하는 CL과 각 상위개념에 대한 하위개념들을 표현하고 있는 SL 두 층으로 구성되어 있다(그림 3)

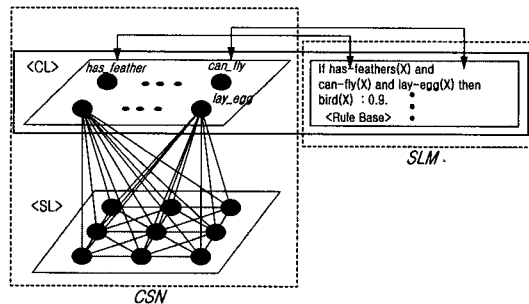


그림 3. SymCSN의 지식표현 구조

기존의 CSN이 CL에서 전향 신경망을 이용하여 상위개념간의 관계를 규칙의 형태로 표현한 반면, SymCSN에서는 CL의 전향신경망 대신 SLM과 연결될 수 있는 개념 노드로 이루어진다. SLM에서는 술어논리의 정형식으로 표현함으로써 보다 확장된 표현력을 갖는다. 또 신경망으로 해결하기 어려운 변수의 바인딩 및 함수의 표현 등을 효율적으로 수행할 수 있다. 그림 2에서 전향 신경망으로 표현된 규칙을 SLM에서 기호적으로 표현하면 식 (1)과 같다.

$$\text{if } \text{has-feathers}(X) \wedge \text{can-fly}(X) \wedge \text{lay-egg}(X) \text{ then } \text{bird}(X) : 0.9 \quad (1)$$

조건부와 결론부에서 사용되는 각 술어는 CL에서 하나의 노드에 해당되며, 식 (1)에서 사용된 ‘:’는 규칙에 대한 확신도(certainty factor)를 표현하기 위해 사용한 연산자이다. 추론 과정에서 확신도 계산은 최소-대수적 합(Min-Algebraic Summation) 방법을 사용한다. 이 방법은 조건부의 각 CL 노드의 활성화 값 중 최소값과 규칙의 확신도를 곱함으로써 계산된다. 이와 같이 SLM을 사용하면 불확실한 사실이나 규칙에 대한 표현 및 이를 이용한 추론을 쉽게 처리할 수 있다.

SL은 CL에 속해있는 상위개념들에 대한 하위개념들의 노드로 하는 연속적 홉필드 네트워크로 구성되어 있는 점은 CSN과 동일하다. 보다 자세한 SL의 구성은 [4]를 참조하기 바란다.

3. SymCSN의 추론

3.1 추론 과정

본 논문에서 제안하는 SymCSN의 추론 과정은 SLM에 의한 논리적 추론 단계, 하향단계, 안정단계, 상향단계의 4단계로 구성된다(그림 4).

(1) 논리적 추론 단계

사용자의 입력 또는 질의를 수행해야 할 목표로 설정하고, 역방향 추론에 기반한 SLM으로 추론을 수행하는 단계이다. SLM은 논리적 완전 정합 연산을 통해 지식베이스 안의 사실이나 규칙을 이용해 추론을 수행한다. 논리적 완전 정합이 이루어지지 않으면 더 이상 추론이 되지 않는 경직성 문제를 가지고 있다.

예를 들면 지식베이스에 ‘monkey(adam)’이라는 사실과 ‘if man(X) then mortal(X)’라는 규칙이 있고, ‘mortal(adam)’이란 목표를 얻기 위해 논리적 추

론을 수행하면 추론에 실패하게 된다. 규칙의 조건부인 ‘man(X)’와 정합 되는 사실이 지식베이스 내에 없기 때문이다. 그러나 사람의 경우 ‘monkey’는 ‘man’과 유사하므로 어느 정도의 확신도로 ‘mortal(adam)’이라는 결론을 내릴 것이다. 이와 같이 사람은 명확히 모든 사실과 규칙을 가지고 있지 않더라도 유연하게 추론을 수행할 수 있는 반면에, 기존의 기호주의적 추론 시스템의 경우 그렇지 못하다. 본 논문에서 제시하는 지식표현 구조가 가지는 추론 과정은 기호주의적 규칙기반 시스템이 완전 정합에 의한 추론 과정으로 발생하는 경직성 문제를 해결할 수 있으며, 유연하게 추론이 되는 과정은 그림 5와 같다.

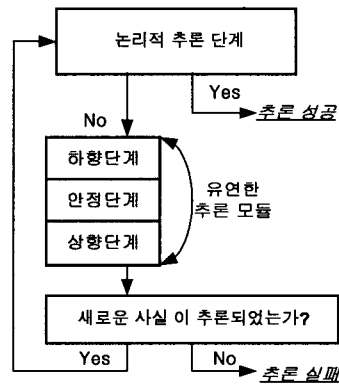


그림 4. SymCSN의 추론 체계

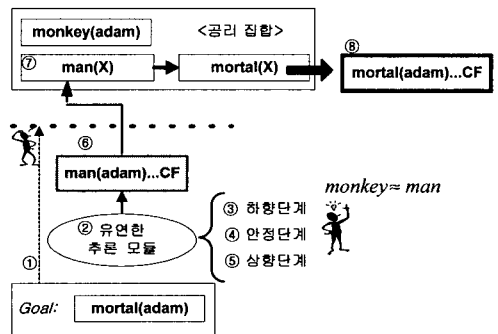


그림 5. 유연한 추론 과정

먼저 기존의 규칙기반 시스템처럼 논리적 완전 정합에 의한 추론을 시도하고(①), 실패하면 유연한 추론 모듈(②)을 수행한다. 이 모듈에서는 지식베이스 안에 ‘adam’이 ‘monkey’라는 사실을 이용하여 CL의

'monkey'라는 노드를 활성화시킨다. 활성화된 'monkey'라는 노드는 하향(③), 안정(④), 상향(⑤) 단계를 통해 'monkey'와 가장 유사한 상위개념인 'man'을 활성화시킨다. 이 때 'man'이라는 개념의 활성화 정도는 'monkey'와 'man'간의 유사한 정도(CF)를 나타낸다. 이렇게 새로이 활성화된 개념 'man'을 이용해 'man(adam)'이라는 새로운 사실을 도출해 낸다(⑥). 그리고 새로 도출된 이 사실로 인해 'if man(X) then mortal(X)'라는 규칙이 적용 가능하게 된다(⑦). 규칙의 적용 결과 'mortal(adam)'이라는 결론을 도출한다(⑧).

(2) 하향단계

이 단계는 활성화된 상위개념에 대한 하위개념들을 활성화시키는 단계이다. 하위개념의 활성화 정도는 상위개념들의 활성화 정도와 하위개념의 중요도에 따라 달라질 수 있도록 하였다. 이것은 식 (2)와 같이 하위개념의 활성화 정도가 각각의 상위개념의 출력값과 상위개념과 하위개념 사이의 연결가중치의 가중합을 시그모이드 함수를 통하여 비선형 변환함으로써 결정된다.

$$y_i^{SL}(t+1) = 2 \times \left(\frac{1}{1 + e^{-\lambda \sum_j y_j^{CL}(t) W_{iSLjCL}}} - 0.5 \right) \quad (2)$$

y_j^{CL} : j 번째 CL노드의 출력 값

y_i^{SL} : i 번째 SL노드의 출력값

W_{iSLjCL} : j 번째 CL와 i 번째 SL노드사이의 출력값

λ : 상수

(3) 안정단계

안정단계에서는 하향단계를 통하여 활성화된 하위개념들을 입력 값으로 하여 식 (3)에 따라 각각의 노드의 출력값을 계산하며, 이 과정을 수렴할 때까지 반복한다.

$$y_i^{SL}(t+1) = 2 \times \left(\frac{1}{1 + e^{-\lambda \sum_j y_j^{CL}(t) W_{iSLjSL} + \theta_i^{SL}}} - 0.5 \right) \quad (3)$$

θ_i^{SL} : i 번째 SL노드의 출력 값

λ : 상수

홉필드 네트워크에서 가중치는 학습 자료의 통계적 관계로부터 얻어진 제약사항으로 볼 수 있으며, 홉필드 네트워크가 수렴하는 과정은 현재의 입력 값에 대

하여 제약사항을 가장 잘 만족시킬 때까지 각각의 노드가 자신의 출력 값을 변화시켜서 입력 값과 가장 가까운 학습 자료를 회상하는 과정으로 볼 수 있다 [11]. 이 과정을 통하여 서로 연관된 상위개념의 하위개념들이 동시에 활성화된다.

(4) 상향단계

안정단계를 통하여 연관된 하위개념들을 활성화시킨 이후에는, 효율적인 추론을 위하여 SL에 분산적으로 표현된 자료를 상위개념의 수준에서 재해석하는 과정이 필요하다. 이것은 하향단계에서 식 (4)를 통하여 하위개념의 영향을 상위개념으로 전파함으로써 이루어진다. 식 (4)는 하위개념의 활성화 값과 상위개념의 활성화 값을 결합하는 식으로 MYCIN의 CF (certainty factor)결합방법을 적용한다[12].

$$y_i^{CL}(t+1) = \begin{cases} y_i^{BU}(t+1) + y_i^{CL}(t) \times (1 - y_i^{BU}(t+1)) \\ \frac{y_i^{BU}(t+1) + y_i^{CL}(t)}{1 - \min(|y_i^{BU}(t+1)|, |y_i^{CL}(t)|)} \\ y_i^{BU}(t+1) + y_i^{CL}(t) \times (1 + y_i^{BU}(t+1)) \end{cases}$$

$$\begin{aligned} & y_i^{BU}(t+1) > 0, \quad y_i^{CL}(t) \\ & y_i^{BU}(t+1) > y_i^{CL}(t) \\ & y_i^{BU}(t+1) < 0, \quad y_i^{CL}(t) < 0 \end{aligned} \quad (4)$$

식 (5)를 통하여 계산되어지는 N_i^{CL} 은 상위개념에 속한 하위개념을 학습하기 위한 학습 자료에서 특정 상위개념과 무관한 관계(0)가 아니라 양의 관계(+1) 혹은 음의 관계(-1)를 가지는 하위개념의 수를 의미한다. 식 (5)에서 가중합을 상위개념에 대한 유효 노드 수로 나누는 것은 유효 노드 수가 다른 상위개념의 가중합을 학습 자료에 근거하여 정규화하기 위한 것이다.

$$y_i^{BU}(t+1) = 2 \times \left(\frac{1}{\sum_j y_j^{SL}(t) W_{iBUjSL} - \lambda \frac{j}{N_i^{CL}}} - 0.5 \right)$$

$$N_i^{CL} = \frac{\sum_{k=1}^n \sum_j y_{kj}^{CL} \| y_{kj}^{SL} \|}{\sum_{k=1}^n |y_{kj}^{CL}|} \quad (5)$$

y_{kj}^{CL} : k 번째 학습 자료의 j 번째 CL노드의 출력값

y_{kj}^{SL} : k 번째 학습 자료의 j 번째 SL노드의 출력값

n : 학습자료수

λ : 상수

그림 6은 각 단계의 효과를 잘 보여준다. 그림 6-a에서 볼 수 있듯이 '고릴라'와 '침팬지'는 의미적으로 유사하므로 둘은 많은 하위개념들을 공유한다. 따라서 '고릴라'에 의해 활성화된 하위개념들 중 '침팬지'와 공통인 하위개념들이 상향단계를 통하여 '침팬지'를 활성화시킨다. 이처럼 하향단계를 통하여 하위개념을 공유하고 있는 상위개념들이 그 공유정도에 따라 활성화됨으로써 유사한 관계에 있는 상위개념들이 활성화된다. 또한 그림 6-b에서 볼 수 있는 바와 같이 '꽃'이 하향단계를 통하여 하위개념들을 활성화시키고, 안정단계를 통하여 '꽃'의 하위개념들과 연관되는 '나비'의 하위개념들을 활성화시킨다. 이렇게 활성화된 '나비'의 하위개념들은 상향단계를 통하여 '나비'를 활성화시킨다.

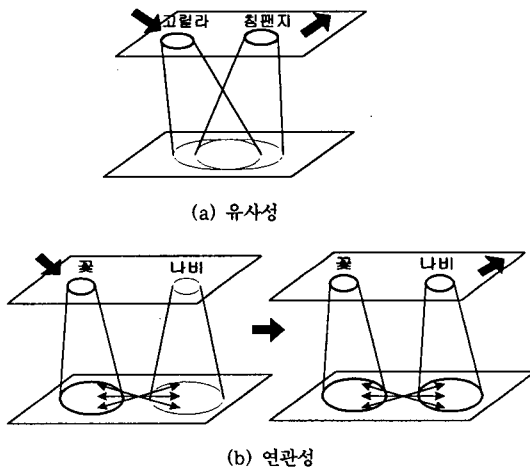


그림 6. 유사성과 연관성에 의한 유연한 추론

3.2 학습 알고리즘

제안된 지식표현 구조의 학습은 CSN과 동일하게 각 상위개념에 대한 하위개념을 학습하는 단계와 하위개념간의 연관성을 학습하는 단계로 이루어진다.

(1) 상위개념에 대한 하위개념을 학습하는 단계

이 단계는 CL에 있는 각각의 상위개념과 SL의 하위개념간의 상호 관계를 학습하는 단계이다. 이 단계에서의 학습은 기본적으로 Hebbian 학습 규칙을 통하여 학습이 이루어진다.

먼저 CL에 학습하려는 상위개념의 출력 값과 SL의 하위개념들의 출력 값을 입력한다. 출력 값은 -1에서 1사이의 실수 값을 가지며, 상위개념과 하위개

념의 출력 값의 부호가 같을 때는 양의 상관관계를, 부호가 반대일 때는 음의 상관관계를 가진다. 또한 출력 값의 곱이 0에 가까울수록 서로 무관한 관계를 나타낸다. 학습율은 함수 $\eta(x)$ 로 한 상위개념에 대한 학습이 진행될수록 학습율을 낮추어 습관화(habituation) 되도록 하였다.

(2) 상위개념에 의해 발화하는 하위개념의 값을 가지고 하위개념간의 연관성을 학습하는 단계

이 단계는 하위개념간의 연관 관계를 학습하는 단계이다. 이 단계에서는 서로 연관관계가 높은 상위개념들을 동시에 활성화시키면 하향단계를 통하여 활성화된 상위개념들의 하위개념들이 활성화된다. 이렇게 활성화된 값을 입력으로 하여 Hebbian 학습 규칙을 통하여 학습이 이루어진다.

학습 알고리즘에 대한 자세한 내용은 [4]를 참조한다.

4. 실험

현재 SymCSN은 Microsoft VC++ 5.0과 Amzi! Logic Server 4.1을 사용하여 PC상에 구현되었다.

SymCSN의 성능을 분석하고 타당성을 검증하기 위하여 크게 두 가지 유형의 실험을 하였다. 첫째, 추론의 유연성을 위하여 CSN에 사용한 같은 문제 즉, 동물에 대한 일상적 담화에 대하여 유연성과 연관성을 이용한 추론에 대하여 실험하였다. 둘째, 문맥 정보를 이용한 추론을 검증하기 위하여 'hiding pot' 문제와 'store' 문제에 대하여 실험하였다. 본 논문에서는 문맥의존 추론에 대한 실험에 대하여 더 중점을 두어 기술하고자 한다.

4.1 유연한 추론: 동물에 관한 일상적 담화

여기서는 동물에 관한 일상적 담화에 관한 예제에 대하여 실험하였다. 다음은 실험에서 사용한 추론 예제이다.

예 ①

- A: 원숭이(monkey)도 새끼를 낳을까(viviparous)?
B: 원숭이는 사람(human)하고 비슷하니까 새끼를 낳겠지.

예 ②

- A: 말(horse)은 무리생활을 하니(crowd)?
B: 발굽동물(ungulate)은 무리생활을 하니까 말도 무리생활을 하겠지.

예 ③

- A: 호랑이(tiger)는 잘 숨니?

B: 호랑이는 줄무늬(has_stripe)가 있으니까 잘 숨을 쉰다.

예 ④

A: 날려면(fly) 날개(wing)가 필요하니?

B: 새(bird)는 날개가 있으니까 그런 것 같다.

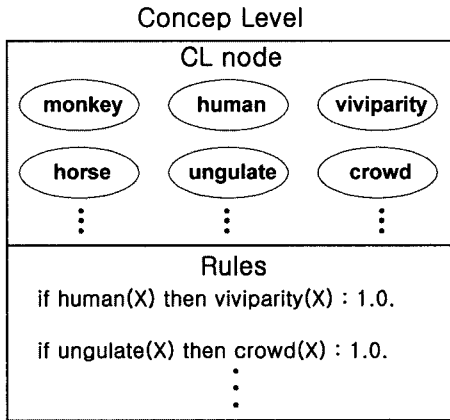


그림 7. CL의 상위개념과 SLM로 표현된 규칙

CL은 위 예에서 발견되는 개념들과 그 개념들로 이루어지는 규칙들로 구성된다. 표현된 규칙의 문법은 'if'와 'then' 사이에 규칙의 조건부, 'then'과 ':' 사이에 규칙의 결론부 그리고 ':' 뒤에는 규칙에 대한 확신도를 나타내었다. 표현된 규칙은 '사람은 태생이다', '발굽동물은 무리생활을 한다', '줄무늬가 있으면 잘 숨는다', '새는 난다', '방패가 있으면 몸을 보호할 수 있다'는 5개의 규칙이다. 각 규칙에 대한 확신도는 보는 바와 같이 모두 1.0으로 되어 있다. 각 상위개념에 대한 하위개념은 Britannica 백과사전과 온라인 어휘 영문 데이터베이스인 WordNet을 참조하여 구성하였다. CL과 SL간의 노드들은 완전 연결되어 있고, 초기 연결 가중치는 모두 0으로 하여 학습시켰다.

4.1.1 유사성에 의한 유연한 추론

유사성을 이용한 추론 예는 예 ①과 예 ②이다. 예 ①의 경우, 지식베이스 내에 'monkey(adam)'이라는 사실을 추가하고, 'viviparous(adam)'이라는 목표를 주면 'adam'과 관련된 사실들을 지식베이스에서 찾아서 활성화시킨다. 'monkey(adam)'이라는 사실에 의해서 'monkey' 노드가 활성화되고, 하향단계를 거쳐 'monkey'의 하위개념들이 활성화된다. 안정단계에서는 'monkey'의 하위개념들과 연관된 다른 하위개념이 없으므로 별다른 변화가 없게 된다. 마지막으로

상향단계를 거쳐 'monkey'의 하위개념을 공유하고 있는 다른 상위개념들을 활성화시킨다. 이 과정을 통하여 'monkey'와 가장 유사한 개념인 'human'이 두 개념간의 유사성 정도만큼 활성화되고, 'human(adam)'이라는 새로운 사실이 지식베이스 안에 추가된다. 이렇게 새로 도출된 사실로 규칙 ① 'if human(X) then viviparous(X)'라는 규칙이 적용될 수 있고, 'viviparous(adam)'이라는 목표를 추론할 수 있게 된다. 이러한 추론 과정은 '원숭이는 태생이다'라는 규칙이 없어도 'monkey'와 'human'이 유사하다는 사실을 이용하여 유연하게 추론을 할 수 있다. 예 ②의 경우도 ①의 경우와 추론 과정은 비슷하다.

4.1.2 연관성에 의한 유연한 추론

위의 예에서 개념간의 연관성을 이용한 추론 예는 예 ③, ④이다. 예 ③의 경우, 먼저 지식베이스 안에 'tiger(my_tiger)'라는 사실을 추가하고, 'hide(my_tiger)'라는 목표를 주면 'my_tiger'와 관련된 사실들을 지식베이스에서 찾아서 활성화시킨다. 'tiger(my_tiger)'라는 사실에 의해서 'tiger'노드가 활성화되고, 하향단계를 거쳐 'tiger'의 하위개념들이 활성화된다. 'tiger'의 주요 하위개념의 하나인 'cat_family'와 다른 하위개념간의 연결 가중치가 학습되거나 사용자에게 의하여 주어진다. 'stripe'의 주요 하위개념의 하나인 'line'과 다른 하위개념간의 가중치로 같은 방법으로 결정된다. 'cat_family'의 높은 가중치의 하위개념과 'line'의 중요한 하위개념이 공통적이므로 'tiger'와 'stripe'는 서로 연관성이 큼을 알 수 있다. 따라서 안정단계를 통하여 'tiger'와 연관 있는 'stripe'의 하위개념들이 활성화된다. 안정단계를 통하여 'stripe'의 하위개념들이 활성화되었으므로 상향단계를 거친 후에는 'tiger'와 'has_stripe'가 모두 활성화되고, 'stripe(my_tiger)'라는 새로운 사실이 지식베이스에 추가된다. 이렇게 새로 도출된 사실에 규칙 ③ 'if has_stripe(X) then hide(X)'라는 규칙이 적용될 수 있고, 'hide(my_tiger)'라는 사실을 추론하게 된다. 이러한 추론 과정은 '호랑이는 잘 숨는다'라는 규칙이 없어도 'tiger'와 'stripe'간의 연관성을 이용하여 유연하게 추론할 수 있음을 볼 수 있다. 예 ④의 경우도 마찬가지로 'wing'과 'bird'의 높은 연관성을 이용하여 유연한 추론을 한다. 유사성과 연관성에 의한 유연한 추론에 대한 상세한 내용은 [4]를 참조할 수 있다.

4.2 문맥의존 추론(I): Hiding Pot 문제

여기서는 SymCSN의 유연한 추론 실험을 위해 어의 모호성을 해결하는 문제에 대해서 실험하였다. 어의 모호성 해결 문제는 자연어 이해에 있어서 매우 중요한 문제중의 하나로 문맥과 인간의 명시적인 배경 지식에 의존적인 특성이 있다[13]. 이러한 어의 모호성 문제를 위해 동적으로 변하는 문맥정보 생성은 연상메모리 역할을 하는 CSN 모듈로, 인간의 명시적인 배경지식의 표현 및 추론은 SLM로 구현함으로써 해결하고자 했다. SLM은 상위개념간의 관계를 표현한 규칙(rules)과 사실(facts)로 이루어진 지식 베이스와 역방향 추론(backward reasoning)을 기본으로 하는 추론 엔진으로 구성된다. SLM은 Amzi!에서 제공하는 프롤로그 로직 서버(Prolog logic server)와 API를 사용하여 구현하였다(그림 8).

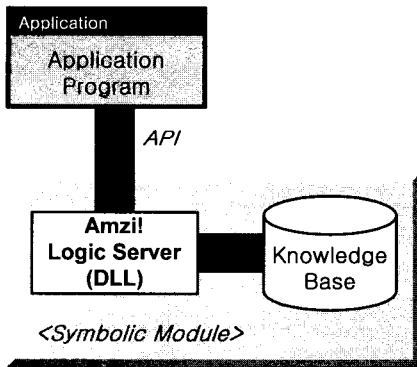


그림 8. SLM 구현

문맥 정보를 이용한 추론 실험에서 상위 개념에 대한 하위 개념들은 인간의 어휘 메모리의 심리언어학적 이론을 바탕으로 구성된 온라인 어휘 영문 데이터 베이스인 WordNet(ver 1.5)의 단어 의미 네트워크(Word Sense Network)에서 IS-A(hypernym/hyponym) 관계를 이용해 구성하였다(그림 9).

Hiding Pot 문제는 'pot'이라는 단어가 상황에 따라서 'cookware'나 'marijuana'의 의미를 가지게 되므로 각 상황에 따라서 'pot'이 가지는 적절한 의미를 찾는 예제이다. 예를 들면 아래 문장 (3), (5)의 경우 일반적으로 'John이 cookware를 씻고자 한다'라는 뜻으로 해석될 수 있는 반면 (4)의 경우 'John이

marijuana를 숨기고자 한다'라는 뜻으로 재해석될 수 있다.

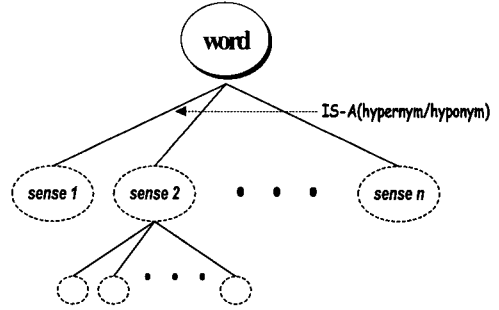


그림 9. WordNet의 Word Sense Network

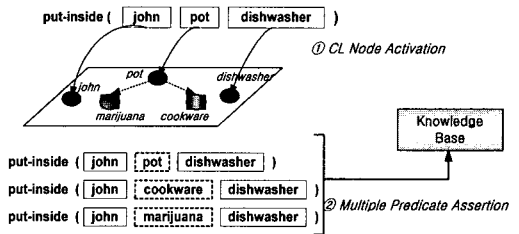


그림 10. ① CL 노드 활성화 단계와 ② 다중 술어 선언 단계

- John put the pot inside the dishwasher. (3)
- John put the pot inside the dishwasher because the police were coming. (4)
- John put the pot inside the dishwasher because friends were coming over for dinner. (5)

인간의 경우 이와 같은 문맥이나 상황에 따른 재해석은 입력에 대한 복잡한 계획이나 목표 분석에 의해 이루어지지 않으며 인간이 가진 일반적인 지식을 가지고 추론한 결과로 보여진다[14]. 위 예제에 대한 실험에서는 'pot'이 가지는 어의의 모호성을 해결하고 유연한 추론을 하는 예를 보이기 위해 지식 베이스에 다음과 같은 지식을 추가하고(표 1), 문장 (3), (4), (5)을 술어논리 정형식으로 변형하여 입력으로 준다(표 2). 상위개념의 수는 8개이고 하위개념의 수는 63개이며 'cookware'와 'marijuana'의 경쟁 관계와 'cookware'와 'dinner' 그리고 'marijuana'와 'police' 개념간의 연관성을 학습한다. 개념간의 경쟁 관계나 연관성의 학습의 개념의 하위 개념간의 가중치로 표현되며 하위 개념간의 가중치가 1에 가까울수록 연관성이 큰 것이며 -1에 가까울수록 경쟁 관계가 큼을

나타낸다.

표 1: Hiding Pot 문제 지식 베이스 구성

사실	· illegal(marijuana), cooker(cookware), opaque(dishwasher)
규칙	· if put-inside(X, Y, dishwasher) and cooker(Y) then clean(X, Y) : 1.0 · if put-inside(X, Y, Z) and illegal(Y) and opaque(Z) then hide(X, Y) : 1.0

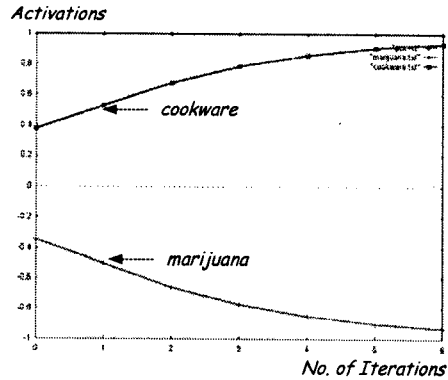
표 2: Hiding Pot 문제에서 입력되는 술어 논리 정형식

문장 번호	술어 논리 정형식
(4)	put-inside(john, pot, dishwasher)
(5)	put-inside(john, pot, dishwasher), come(police, _)
(6)	put-inside(john, pot, dishwasher), come(friends, dinner)

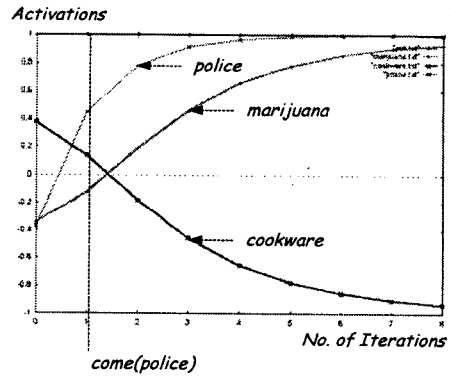
① "John put the pot inside the dishwasher"
위 문장이 술어 정형식 형태로 입력되면 'pot'은 'cookware'의 의미를 가진다. 이것의 의미는 'cookware'가 'marijuana'보다 더 문맥적으로 지원을 받음을 의미한다. 그림 11-a는 CL에서 문맥적으로 지원 받는 'cookware' 노드의 활성화 값은 계속 증가 되면서 경쟁 관계에 있는 'marijuana' 노드의 활성화 값을 계속 감소시키는 것을 보여준다.

② "John put the pot inside the dishwasher because the police were coming"

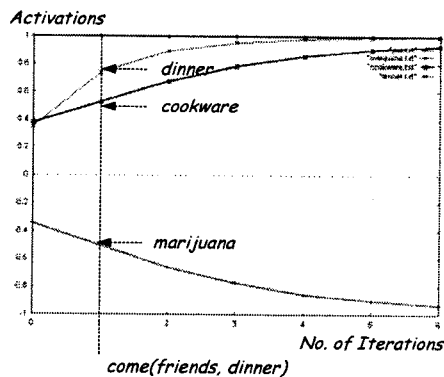
반면에 위 문장이 술어 정형식 형태로 입력되면 초반에는 ①과 유사하게 'cookware'의 활성화 값이 더 크지만, 'come(police, _)'라는 술어가 입력으로 들어오면, 'police'라는 단어가 'cookware'보다는 'marijuana'라는 단어와 더 연관성이 있으므로 'marijuana'의 활성화 값을 계속 증가시켜 주고 'marijuana'와 경쟁 관계에 있는 'cookware'의 활성화 값을 계속 작아지게 된다(그림 11-b). 이는 새로 들어온 입력이 생성한 문맥정보가 'cookware'보다는 'marijuana'를 더 지원함을 의미하며 이렇게 새로 들어오는 입력에 따라 변화하는 활성화 값을 지식베이스에 계속 전파시켜 준다. SLM에서는 추론할 때 동적으로 변화하는 활성화 값을 반영하며 'hide(john, marijuana)'라는 결론이 'clean(john, cookware)'보다 더 큰 확신도를 가지게 된다.



(a) put-inside(john, pot, dishwasher) 입력



(b) put-inside(john, pot, dishwasher),
come(police, _) 입력



(c) put-inside(john, pot, dishwasher),
come(friends, dinner) 입력

그림 11. Hiding Pot 문제의 추론

③ "John put the pot inside the dishwasher because the company was coming over dinner"

이 예제에서는 ①과 유사하게 'cookware'의 활성화 값이 증가하다가, 'come(company, dinner)'라는 술어가 새로 입력으로 들어와도 'dinner'라는 단어가 'marijuana'보다는 'cookware'와 연관성이 더 크므로 'cookware'의 활성화 값을 계속 증가 시켜주는 반면에 'cookware'와 경쟁 관계에 있는 'marijuana'의 활성화 값은 계속 감소하게 된다(그림 11-c). 이런 활성화 값을 가지고 추론을 하면 'clean(john, cookware)'라는 결론이 'hide(john, marijuana)'보다 더 큰 확신도를 가진다.

Hiding Pot 문제에서는 SymCSN이 모호성을 가지는 단어 'pot'의 의미를 동적으로 문맥이 변하는 상황에서 'cookware'나 'marijuana'로의 올바른 의미를 찾아서 추론함을 알 수 있다. 또한 SLM을 통해서 상위개념 간의 일반적인고 구조화된 지식표현과 추론을 수행함을 알 수 있다.

4.3 문맥의존 추론 (II): Store 문제

Store 문제는 'store'라는 단어가 'shop'이나 'storehouse'의 의미를, 'bank'라는 단어는 'financial bank'나 'sloping land'의 의미를 가지므로 상황에 따라서 각 단어가 가지는 적절한 의미를 찾는 문제이다. 이 실험을 위해 사용된 시나리오는 표 3과 같다.

표 3 : Store 문제 시나리오

Rosalind walked to a *store*. She wanted to give her mother a present. Rosalind found out that everything was too expensive. She went to a *bank* to withdraw more money. After she bought the gift, she returned to home.

위 예제를 위해서 지식 베이스에 다음과 같은 지식을 추가하고(표 4), 시나리오를 술어 논리 정형식으로 변형하여 입력으로 준다(표 5). 상위 개념의 수는 15개이고 하위 개념의 수는 112개이다. 또한 'shop'과 'storehouse' 그리고 'financial bank'와 'sloping land'간의 경쟁 관계를 학습하고 'storehouse'와 'present'와 'financial bank'와 'money'간의 연관성을 학습한다.

표 4 : Store 문제 지식 베이스 구성

사실 (facts)	<ul style="list-style-type: none"> · sale_inst(shop) · goods_dep(storehouse) · finance_inst(finance_bank) · waterside_area(sloping_land)
규칙 (rules)	<ul style="list-style-type: none"> · if go(X, Y) and sale_inst(Y) then buy(X, goods) : 1.0 · if go(X, Y) and goods_dep(Y) then store(X, goods) : 1.0 · if go(X, Y) and finance_inst(Y) then withdraw(X, money) : 1.0 · if go(X, Y) and waterside_area(Y) then get_into(X, boat) : 1.0

표 5 : Store 문제에서 입력되는 술어 논리 정형식

술어 논리 정형식
1. go(rosalind, store)
2. give(rosalind, mother, present)
3. expensive(goods)
4. go(rosalind, bank)
5. withdraw(rosalind, money)
6. buy(rosalind, gift)
7. go(rosalind, home)

그림 12, 13에서는 표 5의 술어 논리 정형식이 입력이 되면서 각 CL 노드의 활성화 값의 추이를 보여 준다.

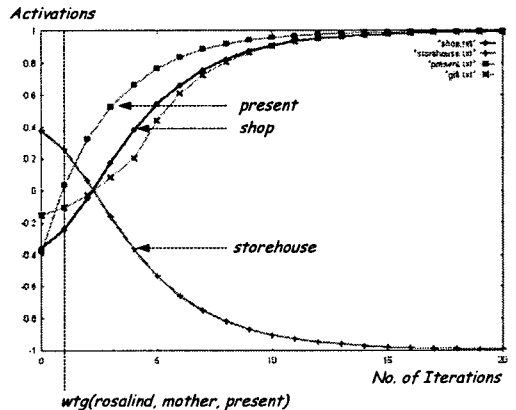


그림 12. 추론 과정 중 CL 노드 활성화 값의 변화(1)

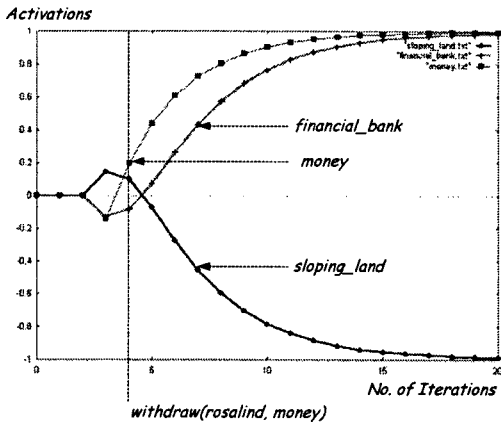


그림 13. 추론 과정 중 CL 노드 활성화 값의 변화(2)

그림 12에서는 'store'의 'shop'과 'storehouse'의 의미가 서로 경쟁하다가 'give(rosalind, mother, present)'라는 입력이 들어오면서 'present'에 의해서 문맥적으로 지원을 더 받는 'shop'의 활성화 값은 계속 커지는 반면에 'shop'과 경쟁 관계에 있는 'storehouse'의 활성화 값은 계속 줄어들게 된다. 이런 활성화 값의 변화는 SLM의 지식 베이스에 반영되어 'buy(rosalind, goods)'라는 결론이 'store(rosalind, goods)'보다 더 큰 확신도를 가지게 된다.

또한 이와 유사하게 그림 13에서는 'bank'의 'financial bank'와 'sloping land'의 의미가 서로 경쟁하다가 'withdraw(rosalind, money)'라는 입력이 들어오면서 'money'에 의해서 문맥적으로 지원을 더 받는 'financial bank'의 활성화 값이 계속 커지는 반면에 경쟁 관계에 있는 'sloping land'의 활성화 값은 계속 작아진다. 이런 활성화 값의 변화는 SLM의 지식 베이스에 반영되며 'withdraw(rosalind, money)'라는 결론이 'get_into(rosalind, boat)'라는 결론보다 더 큰 확신도를 가지게 된다.

5. 결론

본 논문에서는 기호주의적 추론 시스템이 갖는 경직성 문제를 해결하기 위하여 본 연구팀이 제안한 바 있는 CSN 모델의 결점을 보완하는 새로운 추론 시스템 SymCSN을 제안하였다. SymCSN은 신경망을 사용함으로써 상위개념간의 일반적이고 구조적인 지식을 표현하기 어렵고 또 변수의 바인딩이 곤란한 문제점을 해결하기 위하여 기호주의적 방법과 연결주의적

방법을 통합한 지식표현 및 추론 시스템이다. SymCSN은 CSN의 상위수준의 지식표현 및 추론 구조인 신경망을 확신도가 추가된 논리에 바탕을 둔 논리적 추론 모듈로 대체한 것이다. 그렇게 함으로써 SymCSN은 기존의 CSN과 같은 추론의 유연성을 유지하되 보다 더 강력한 지식표현과 추론 기능을 가진다. 또 SymCSN은 추론에 있어서 논리적 추론을 일차적으로 수행하고 논리적 추론이 실패할 경우 유사성이나 연관성에 의한 유연한 추론을 수행함으로써 인간의 계층적 추론 메카니즘의 특성을 보인다. 여러 가지 문제에 대하여 실험한 결과 SymCSN은 인간의 유연한 추론 메카니즘이 갖는 중요한 특성을 나타냄으로써 인간의 추론 메카니즘을 설명할 수 있는 모델로서의 높은 가능성을 제시하였다. 특히 본 논문에서는 SymCSN이 추론이나 자연어 이해 등에 있어서 중요한 문맥정보를 이용한 어의의 모호성 해결과 같은 문제를 용이하게 해결할 수 있음으로써 그 기본 모델의 유용성이 크다고 할 수 있다. 본 논문에서 기술한 SymCSN은 아직 초보적인 단계이나 기호주의적 방법과 연결주의적 방법이 상호 보완적으로 통합됨으로써 보다 효율적인 추론 시스템이 구현될 수 있는 가능성을 제시한다.

향후의 연구과제로는 속성 상속(property inheritance) 등을 통한 지식표현 및 추론 메카니즘의 확장과 확신도를 가지는 지식베이스의 진리 보존 시스템(Truth Maintenance System) 등 지식베이스의 일관성 관리에 대한 연구가 필요하다. 또 SymCSN을 보다 실질적인 추론 문제에 적용함으로써 SymCSN의 유효성을 입증하고 유연한 추론, 지식 표현을 위한 다른 모델이나 시스템과의 비교 평가가 수행되어야 한다. 특히 퍼지이론에 기반을 둔 퍼지추론 방법과의 비교 및 통합에 대한 연구도 필요하다.

참고문헌

- [1] S. Gallant, Connectionist Expert Systems, communication of ACM, VOL. 31(2), pp. 152-169, 1988
- [2] McClelland, J. L., and Rumelhart, D. E., An interactive activation model of context effects in letter perception: part 1. An account of basic findings, Psychological Review 88, pp. 375-407, 1981
- [3] Ron Sun, "A Two-Level Hybrid Architecture for Structuring Knowledge for Commonsense

- Reasoning". Computational architecture integrating neural and symbolic process A Perspective on the State of the Art, pp. 247-281, 1995.
- [4] 김영분, 민창우, 김명원, "유연한 추론을 위한 연결주의적 지식표현 구조", 한국정보과학회 논문지, 제 24권 6호, pp. 650-662, 1997.
- [5] Kintsch, W., "The Role of Knowledge in Discourse Comprehension: A Construction-Integration Model", Psychological Review, Vol. 95, No. 2, pp. 163-182, 1988
- [6] Roman Jakobson, 일반언어학 이론, pp. 45-72, 민음사, 1989
- [7] Martindale, C., 인지심리학: 신경회로망적 접근, pp. 328-332, 교육과학사, 1994
- [8] Fukumi Kozato, "A Hybrid Rule-based System : How Variables are Involved in Connectionist Rule-based System", IJCAI, 1995
- [9] <http://www.amzi.com/catprls.html>
- [10] Dennis Merritt, "Prolog Under the Hood: An Honest Look", PC AI magazine Sep/Oct, 1992
- [11] Sharkey, A. J. C., and Sharkey, N. E., "Weak Contextual Constraints in Text and Word Priming", Journal of Memory and Language, Vol. 31, pp. 543-572, 1992
- [12] Buchanan, B. G., Shortliffe, E. H., Rule-Based Expert Systems, pp. 209-232, Addison-Wesley, 1984
- [13] Samuel W.K.Chan, J. Franklin, "Symbolic Connectionism in Natural Language Disambiguation", IEEE Transactions on Neural Networks, Vol. 9, No. 5, September, pp. 739-755, 1998
- [14] Lange, T, "A Structured Connectionist Approach to Inferencing and Retrieval", Computational Architectures Integrating Neural and Symbolic Processes, Kluwer Academic Publishers, pp. 69-115, 1995

부 록

A. Hiding Pot 문제에서 주요 상위개념에 대한 하위개념들

상위 개념	하위 개념들
pot	1. caldron 2. coffeepot 3. dixie 4. dutch_oven 5. kettle 6. marmite 7. pottle 8. stockpot 9. teapot 10. coffee_urn 11. samovar 12. tea_urn 13. cannabis 14. ganja 15. grass 16. marihuana 17. dope 18. weed 19. gage 20. sess 21. sens 22. skunk 23. joint 24. cigarette 25. reefer 26. stick 27. acapulco_gold 28. bhang
marijuana	1. cannabis 2. ganja 3. grass 4. marihuana 5. dope 6. weed 7. gage 8. sess 9. sens 10. skunk 11. joint 12. cigarette 13. reefer 14. stick 15. acapulco_gold 16. bhang
cookware	1. caldron 2. coffeepot 3. dixie 4. dutch_oven 5. kettle 6. marmite 7. pottle 8. stockpot 9. teapot 10. coffee_urn 11. samovar 12. tea_urn 13. chafing_dish 14. cooker 15. enamelware 16. grid 17. griddle 18. pan
police	1. gendarmerie 2. gendarmery 3. rcmp 4. scotland_yard 5. new_scotland_yard 6. vice_squad 7. secret_police 8. gestapo 9. posse 10. posse-comitatus
friend	1. brother 2. buddy 3. companion 4. confidant 5. glatmate 6. girlfriend 7. light 8. mate 9. playmate 10. roommate 11. schoolfriend
dinner	1. supper 2. last_supper 3. high_tea 4. dinner_party 5. banquet 6. feast 7. gaudy 8. beanfeast