

人工知能의 理解

韓 成 國
〈圓光大學校電子計算工學科長·工博〉

1. 서 론

인간은 오감을 통하여 감각계의 각종정보를 받아들여 이에 대한 적절한 응답을 할 뿐만 아니라 생각하고 연상하며 여러 가지의 판단을 하는 등 고도의 지적 능력을 수행하고 있다. 이러한 인간의 지적 능력을 기계로 실현하고자 하는 노력은 오래전부터 지속되어 왔다.

18 세기 후반 헝가리의 발명가 쾨펠렌(Baroy Woffgang von Kewplem)이 고안한 자동체스 기계(automatic chess machine)나 바베지(Babbage)의 해석기(analytic engine)도 이러한 능력을 실현해 보려는 욕망에서 이루어진 것들이다. 컴퓨터가 출현한 이후 이러한 노력은 가속화되어 1947년 계산 이론(Computational theory)으로 유명한 튜링(Turing)은 컴퓨터의 지능을 테스트하는 Turing test를 제안하였으며 1956년엔 J. McCarthy를 중심으로 한 일단의 과학자들이 모여 "기계는 생각할 수 있을까"란 주제로 진지한 토의와 연구를 한 결과 인공지능(Artificial Intelligence)의 새 시대를 개척하였고 최초의 전문가시스템(expert system)이라 할 수 있는 Logical Theorist라는 프로그램을 개발하였다. 이 프로그램은 Whitehead와 Russell이 저술한 Principia Mathematica에 수록된 일부 공식을 증명하였는데 그중 어떤 것은 논리학자보다 더 나은 방법으로 증명하였으나 컴퓨터가 증명하였다는 이유로 학술지에 게재되

지 못하였다.

1950 년경부터 본격적인 연구가 시작된 기계 번역(machine translation)도 인간의 지적 능력을 실현해 보고자한 좋은 예라고 할 것이다.

그러나 이러한 초창기의 연구는 컴퓨터의 능력을 과신한 나머지 인간의 지적 능력에 대한 충분한 연구와 통찰없이 시작되었기 때문에 대부분 미약한 결과를 얻는데 그치고 말았다.

70년에 들어와 반도체 집적기술의 비약적인 발전으로 컴퓨터의 새로운 가능성이 열리기 시작하였다. 초당 수백만개의 명령(instruction)을 실행하는 고속연산기능을 갖는 컴퓨터나 병렬처리 구조를 갖는 슈퍼 컴퓨터가 등장하였으며 logic programming machine, data flow machine 등 종래와는 전혀 다른 구조를 갖는 컴퓨터의 출현으로 지적 처리능력을 갖는 컴퓨터의 개발을 눈앞에 두게 되었다. 즉 인공지능과 결합한 고도의 지적 능력을 갖는 지식처리형의 제5세대 컴퓨터는 새로운 시대의 전개를 예고하는 제3의 물결로 우리에게 접근하고 있다.

각 국가는 이러한 필연성과 시대적 요청에 따라 막대한 연구비를 투자하여 제5세대 인공지능형 컴퓨터개발에 박차를 가하고 있다. 미래에는 정보와 지식처리를 가속화할 수 있는 국가만이 강대국이 될 수 있다는 판단아래 국운을 걸고 경쟁적으로 연구개발에 주력하고 있다.

본고에서는 아직 우리에게는 생소하지만 미래의 중추적 학문으로 등장할 인공지능 분야에 대한 이해를 돕고

자 평이한 설명으로 인공지능분야를 총괄 조감하고 인공지능의 주요 연구대상과 방법, 현재의 상황과 앞으로의 추세 등에 대하여 전반적으로 서술한다. 또한 제 5세대 컴퓨터의 기본구상과 실현방법 등에 대해서 자세히 설명하고자 한다.

우리는 컴퓨터과학과 그 주변 학문의 발전추세를 통하여 급변하는 학문과 과학기술에 대응하는 우리의 자세와 역할에 대해서도 진지하게 생각하여야 할 때가 왔다.

2. 인공지능과 제 분야

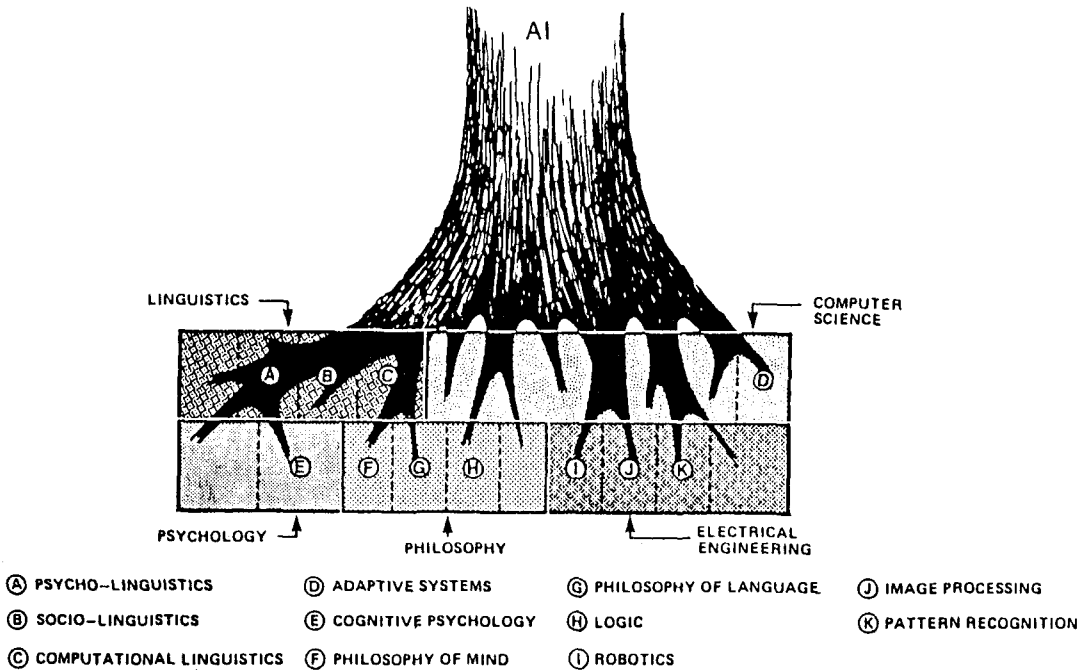
지능(intelligence)은 인간 고유의 기능이다. 지능이 있기 때문에 인간은 지식을 이해하고 배울 수 있으며 창조의 능력을 갖게 되는 것이다. 인간은 축적된 지식을 이용하여 장기와 바둑을 두거나 어려운 기하학 등을 자유자재로 이해할 수 있으며 언어를 이용해 상호의 의사전달과 정보교환을 한다. 그러나 간단한 규칙으로 이루어진 장거나 바둑조차도 아직 인간만큼 능숙하게 두거나 기하학 문제를 증명해 내는 기계는 없다. 그렇다면 인공지능이란 무엇인가? 간단히 말하자면 컴퓨터

로 하여금 인간과 유사한 지능을 갖게하는 학문이다. 다시 말하면 인간의 행동중에서 지능과 관계되는 일의 처리를 수행하는 컴퓨터시스템을 개발하는데 있다. 이러한 인공지능의 연구는 50년대 이후 본격적으로 시작되었다. 초기에는 인공지능게임(game)을 하거나 논리증명 등 컴퓨터 내적인 문제에 치중하였으나 최근에는 질의응답(Question-answering), 전문가시스템(expert system) 등 실용적 문제에 초점을 두고 있다. 앞서 설명한 바와 같이 인간 고유의 지적 능력을 컴퓨터에 부여하는 것은 간단한 문제가 아니다. 따라서 인공지능은 지능과 관련된 모든 학문을 총합에 의해 형성된 새로운 작품이라 할 것이다. 인공지능은 <그림-1> 과 같이 제반요소의 결합에 의한 결정체인 것이다. 그러므로 인공지능은 방대한 분야와 학문이 밀접한 관계를 형성하기 때문에 그만큼 난해하게 된다.

이 절에서는 인공지능의 중요분야에 대하여 개괄적으로 설명한다.

1) 문제해결과 수색

바둑을 두는 경우를 생각해 보자. 바둑은 간단한 규칙하에 이 규칙이 허락하는 장소에 한 수씩 두어



<그림-1> 인공지능과 연관분야

나가면 된다. 한수를 놓는 경우의 수를 생각하면 대략 평균적으로 10^2 (실제로 제1수는 19^2 , 제2수는 $19^2 - 1, \dots$ 등으로 된다.)의 경우를 고려해야 한다. 100 수로서 바둑이 끝났다면 $(10^2)^{100} = 10^{200}$ 의 경우를 고려해야 한다. 현존의 최고속 컴퓨터가 이 바둑을 두는 경우 1초에 10^8 수를 놓는다면 1년간에는 약 10^8 수/초 $\times 3 \times 10^7$ 초/년 $= 3 \times 10^{15}$ 수/년이 되어 최소한 10^{180} 년이 걸리게 된다. 그러나 인간처럼 수십 수 앞을 내다보고 두는 경우에는 더 많은 시간이 소요될 것이다.

이 바둑 등 인간의 지적 능력이 필요로 하는 일반문제는 경우의 수가 지수함수적으로 증가하기 때문에 컴퓨터에서는 combinatorial explosion이 발생한다. 그러므로 컴퓨터가 이러한 문제를 해결하기 위해서는 인간의 경험적 지식을 포함한 지적문제 해결과정을 처리할 수 있는 능력을 갖지 않으면 안된다. 먼저 문제와 문제의 해결방법 또는 규칙 등을 컴퓨터에 명확히 부여해 주어야 한다.

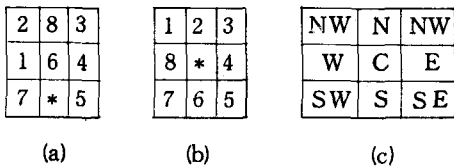
문제를 표현(problem representation)하는 방법으로서는

- 상태공간법(state-space representation)
- 문제축소법(problem-reduction representation)
- game tree

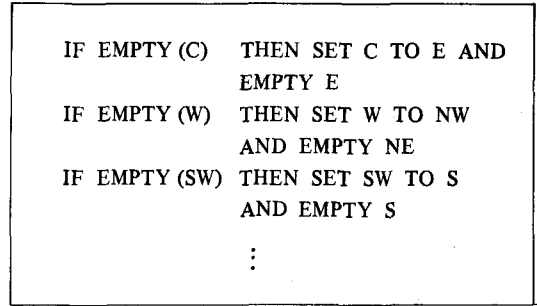
등의 방법이 사용되고 있다.

규칙의 표현방법으로는 주로 논리식에 기초한 production system이 사용되고 있다.

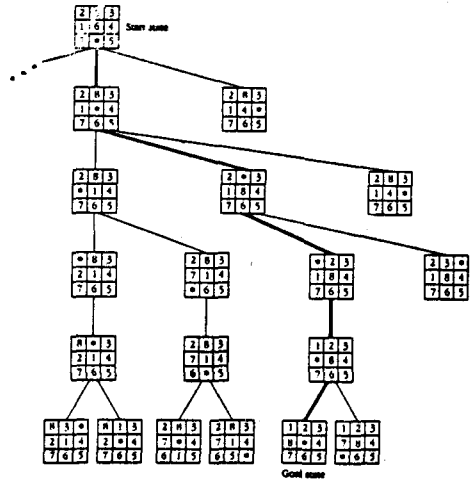
다음의 8-puzzle 문제를 고찰해 보자. <그림-2>의 (a)의 초기상태를 (b)의 최종상태가 되도록 각 숫자를 재배치하는 문제이다. 이 문제를 풀기 위한 규칙을 생각해 보면 <그림-3>의 형식과 같은 24개의 규칙을 얻어낼 수 있다. 이 production rule중 적용가능한 규칙을 이용하여 최초 상태에서부터 최종 상태에 이르게 함으로써 문제가 해결된다. 이 과정의 일부를 보면



<그림-2> 8-puzzle 문제



<그림-3> 문제풀이 규칙의 서술 예



<그림-4> 문제 해답의 탐색과정

<그림-4>와 같다.

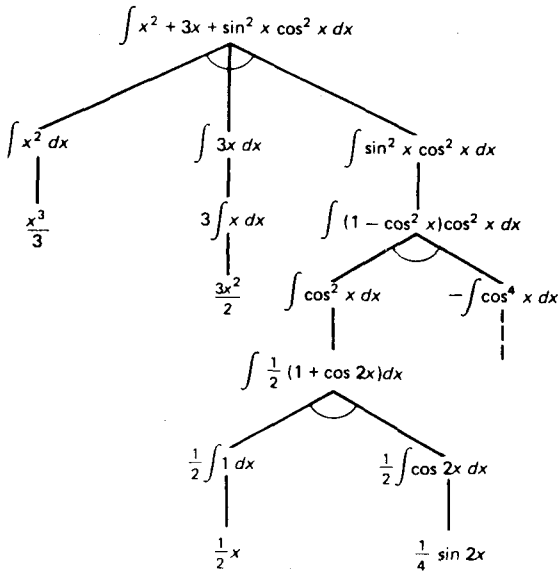
이 문제풀이 과정을 일반화하면 문제해결이란 결국 상태그래프(state graph)에 표시된 각 상태를 여행하는 것, 즉 현재 상태에서부터 다음의 최적 상태를 탐색(search)하는 것이 된다. 따라서 다음 상태를 어떠한 것으로 택하느냐가 문제해결의 중요한 요소가 된다.

문제가 분할(decomposition)이 가능하다면 <그림-5>와 같이 분할된 각각의 문제를 해결함으로써 전체 문제를 해결할 수도 있다.

그러나 어느 경우에도 탐색은 문제를 해결하는 과정에서 중요하게 된다.

탐색방법으로는

- depth-first
- breath-first



〈그림-5〉 분할 가능한 문제

- best-first
- hill-climbing
- branch and bound
- Waltz labeling method
- minimax
- alpha-beta pruning
- means-ends analysis

등 수 많은 방법이 개발되어 있다. 그러나 일반적으로 상태의 수는 지수함수적으로 증가하기 때문에 수색은 많은 시간이 소요된다. 불필요한 수색을 피하기 위해서는 최종상태를 예측하거나 상태전이시 heuristic function을 구하는 방법 등이 연구되기도 하였다. 일반적인 문제의 경우는 최종상태의 예측이나 heuristic function을 구하기 어렵기 때문에 보편적인 응용성은 더욱 연구되어야 할 것이다.

문제는 인간이 갖고 있는 forward reasoning과 backward reasoning을 포함한 추론능력과 경험적인 지식을 어떠한 방법으로 구조화 또는 모델화 시키느냐에 달려있다. 문제해결과 수색은 인공지능의 기본문제이기 때문에 조속히 좋은 결과가 나와야 할 과제중의 하나이다.

2) 지식표현

인공지능은 인간의 고차적인 지적 활동능력을 갖는 컴퓨터의 실현이다. 따라서 인공지능의 1차적인 지식(knowledge)이 된다. 지식에 관한 문제는 언어학, 심리학, 철학, 사회학 등에서 주된 과제로 지능과 관련하여 상당한 연구가 있었지만 아직 해결되지 않은 부분이 산재하여 있다.

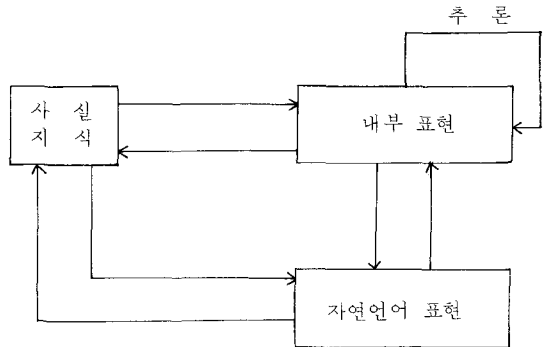
인간의 지적 행위의 대상으로서의 지식은 수학·물리학 등의 재방법칙 뿐만 아니라 음성, 신호, 도형, 화상, 언어 등 시·공간적인 구상과 추상의 모든 정보를 포함하기 때문에 인간이 갖는 이러한 지식을 컴퓨터 내부에서 기술하는 지식표현(knowledge representation)은 난해한 과제중의 하나이다. 여기서 지식표현이란 종래의 구조화된 데이터의 모임이 아니라 의미를 표현하는 실체로서 인간의 지식처럼 추론과 연상이 가능한 형태로서의 표현을 의미하는 것이다.

특히 지식은 언어 등의 가시적인 형태로 표현될 수 있다는 관점에서 지식표현은 자연언어 이해 등을 포함한 언어학적 문제와 밀접한 관계를 형성한다. 그러므로 컴퓨터 내부에서 지식표현은 외적으로 적어도〈그림-6〉과 같은 상호관계를 형성해야 한다. 즉 지식과 컴퓨터 내부표현, 언어표현 사이에는 상호변화와 연관관계가 형성되어야 한다.

지식표현은 내적으로는 지식축적(knowledge acquisition), 추출(retrieval)과 추론(inference 또는 reasoning) 등을 위한 지식조작(knowledge manipulation)이 가능한 형태로 구조화 되어야 한다.

지식표현 방법으로는

- logic representation
- semantic network



〈그림-6〉 지식표현의 상호관계

- procedural representation
- frame
- script

등 수 많은 방법이 제안되어 있다. 그러나 이들 방법은 특정 영역(specified domain)의 지식을 표현하는데는 이용될 수 있어도 포괄적인 응용을 위해서는 아직도 많은 점이 보완되어야 한다.

논리식 표현의 가장 전통적인 지식표현 방법으로 명제논리, 술어논리 등의 발전을 거쳐 양상논리(modal logic), 내포논리(intentional logic), non-monotonic logic, fuzzy logic 등의 발전을 이루었다. 이 논리식 표현방법은 논리학의 연구 결과를 그대로 이용할 수 있으며 추론 등에 적합한 구조를 갖는 장점이 있다. 예를 들어

All birds have wings.

A robin is a bird

의 경우 논리식으로 표현하면

$\forall x \text{ Bird}(X) \rightarrow \text{HasWings}(X)$

$\forall x \text{ Robin}(X) \rightarrow \text{Birds}(X)$

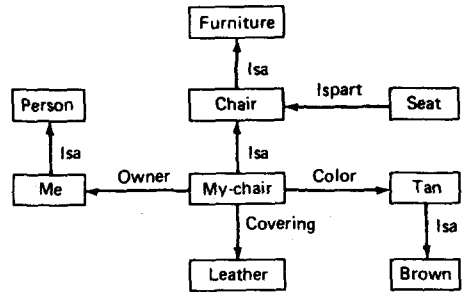
가 된다. 위 두 논리식으로부터

$\forall x \text{ Robbin}(X) \rightarrow \text{HasWings}(X)$

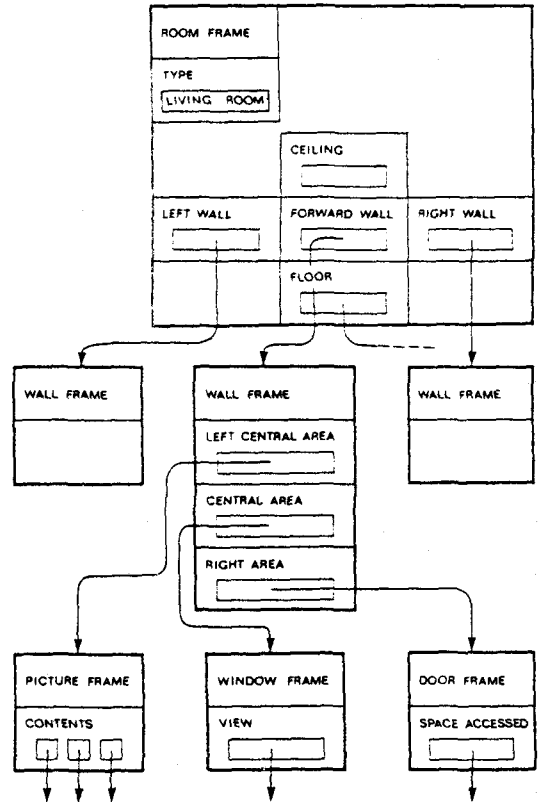
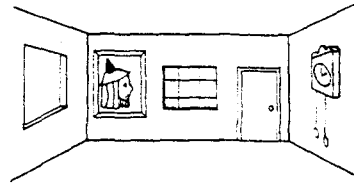
을 유도하는 것은 지극히 간단한 일이다. 이러한 추론을 위해 resolution principle이 정립되어 있기 때문에 효과적인 지식처리를 할 수 있다. 그러나 지식의 의미와 상호연관성을 형성하는데는 문제점이 있다.

지식의 의미와 상호관계를 중시한 것이 Quillian의 semantic network이다. semantic network은 개념 또는 object를 표현하는 node와 이를 연결하는 link로서 <그림-7>과 같은 형태로 구성된다. 이 표현방법은 지식의 외부 연결관계를 명확히 표현하는 연상(association)기능을 모델화한 것으로 지식표현의 일반적인 방법으로 널리 이용되고 있다. 그러나 간단한 지식의 표현에도 대용량의 기억장치가 필요하고 시간의 흐름과 개념표현이 모호한 단점이 있어 확장발전 연구가 진행중이다.

Minsky에 의해 제안된 frame 이론은 어떤 상황하에서 이와 연관된 지식정보를 구조화한 것으로 지식을 slot 형식의 계층적인 network을 만든 것이다. 예를 들어 어떤 방(room)이라는 지식을 표현할 때 <그림-8>의 표현과 같이 상위 frame에 대한 각 slot로서 하위 frame을 확장함으로써 계층적인 network을



<그림-7> Semantic Net의 예



<그림-8> frame 표현의 예

형성해 간다.

frame표현은 논리식이나 semantic network의 선언적 표현(declarative representation)과 절차적 표현(procedural representation)의 장점을 수용할 수 있으며 상위 frame의 지식의 특성을 하위 frame에 유전(inheritance)할 수 있으므로 상당히 유연한 표현을 기대할 수 있다.

Script는 개념의존이론(conceptual dependency theory)과 연관하여 연속적으로 변화하는 상황의 지식을 표현하는데 적합하다.

procedural representation은 의미의 실용적 처리를 강조한 표현으로 "age"의 의미는 [AGE (TO-ESTABLISH (SUBTRACT NOW YEAR-OF-BIRTHDAY))]와 같이 표현된다. 이처럼 의미를 procedure로 나타냄으로써 시스템에 유연한 제어능력을 부여할 수 있게 된다.

이상의 대표적인 지식 표현의 고찰을 통해서 알 수 있는 바와 같이 지식 표현은 복합적이고 추상적인 문제이기 때문에 더 많은 연구가 있어야 할 것이다. 그러나 지식에 대한 상당 부분이 규명되었으므로 컴퓨터의 측면에서 이들의 유용성을 어떠한 방법으로 결합하느냐가 중요하게 된다.

3) 자연언어이해

인간만이 언어를 사용하여 정보와 상호의 의사를 전달할 수 있다. 이러한 인간의 언어생성과 이해능력을 기계에 부여하려는 노력은 컴퓨터의 탄생과 더불어 지금까지 계속되어 왔다. 컴퓨터의 자연언어(natural language) 이해는 바벨탑을 쌓으려는 인간의 최후 이상과 같은 문제인 것이다.

언어는 형태소(morphology), 구문(syntax)과

의미(semantics)가 결합된 복합 다양체이다. chomsky가 언어의 심층구조(deep structure) 개념을 제안한 이후 변형생성문법이나 격문법 등의 언어의 보편적 구조를 규명하려는 노력과 더불어 자연언어 이해를 통한 man-machine interface에 대한 연구는 가속화되어 왔다.

자연언어이해 시스템의 일반 구조는 <그림-9>와 같다.

문장이 입력되면 단어단위로 분할한다. 일본어와 같이 띄어쓰기가 명확하지 않은 언어의 경우에는 이 단어단위 분할은 시스템 전반에 걸쳐 커다란 영향을 미치게 된다.

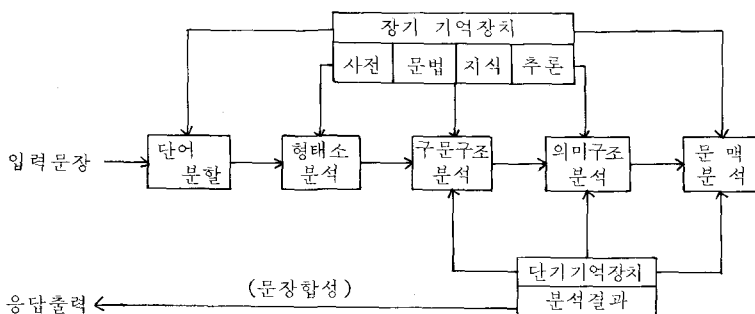
분할된 단어로부터 문법형태소를 추출한다. 추출된 형태소는 구문구조 분석의 기본정보가 되기 때문에 명확히 구조화 되어야 한다. 그러므로 한국어와 같이 조사와 어미구조가 발달한 언어에서는 많은 문제가 내포되어 있다. 예를 들어 "사시면서"와 같은 경우

- ① (오래동안)사+(으)시+면서("살다"의 불규칙)
- ② (컴퓨터를)사+시+면서("사다")와 같이 구분하기 위해서는 문맥정보까지 필요한 경우가 발생한다.

구문구조 분석은 문장의 구조를 정해진 문법의 구조로 분석하여 문장의 문법조직을 형성하는 것으로 이 과정을 parsing이라 한다. 일반적으로 문법을 문맥자유문법(context free grammar)형으로 표시되고 parsing의 결과는 parsing tree로 표시한다.

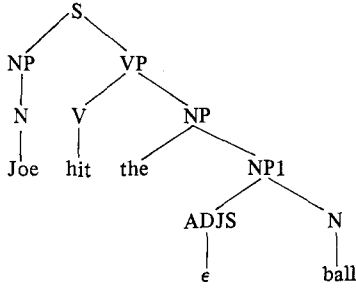
<그림-10>에 간단한 영어문법표현과 이에 의해 생성된 parsing tree를 보였다.

parsing은 문법규칙을 적용함으로써 해결되는 간단한 문제는 아니다. parsing을 하기 위해서는 먼저 문법규칙이 명확히 규정되어야 한다. Chomsky의 변형생성문법, Fillmore의 격문법, Halliday의 systemic



<그림-9> 자연언어이해 시스템의 구조

S → NP VP
 NP → the NP1
 NP → NP1
 NP1 → ADJS N
 ADJS → ε ADJ ADJS
 VP → V
 VP → V NP
 N → Joe boy ball
 ADJ → little | big
 V → hit lan



〈그림 - 10〉 Parsing Tree의 생성

grammar, Montague의 Montague grammar 또는 그 이외의 어떤 문법규칙을 가질 것인가를 결정해야 한다. 언어학적인 문법규칙을 컴퓨터 내부표현으로 변환하여 처리되는데도 해결되지 않은 문제점이 많이

남아 있다. 이 문제는 자연언어이해 시스템의 전반적 성격을 규정하는 것으로 충분한 연구가 있어야 한다. 다음에 parser의 출력 형태를 규정해야 한다. parsing tree, semantic network 등의 여러 방법이 있으며 이 출력형태는 의미해석과 관련하여 면밀한 검토가 있어야 한다.

parsing 방법은 일반적으로

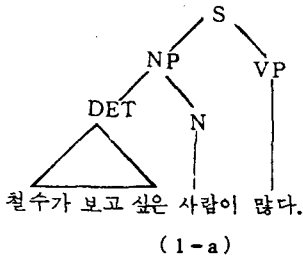
- top-down / bottom-up
- depth-first / breadth-first

등의 방법이 있으나 시스템 구성에는 각각 장·단점이 있다. parsing의 결과는 항상 정확한 결과만을 얻을 수 있는 것은 아니다.

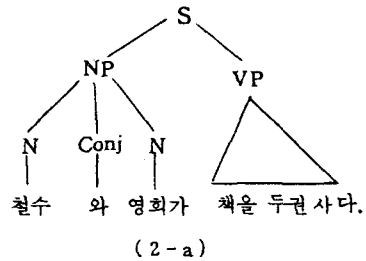
- ① 철수가 보고 싶은 사람이 많다.
- ② 철수와 영희가 책을 두권 사다.

위 문장은 〈그림-11〉과 같이 두가지로 구문분석이 될 수 있다. ①은 수식어구의 연결시 ②는 접속구 연결시 각각 구문적 애매성(ambiguity)이 발생한다. 이와 같이 자연언어는 표면구조(surface structure)의 애매성과 다양성을 갖기 때문에 문법규칙을 임의적으로 적용하여 parsing할 수는 없다.

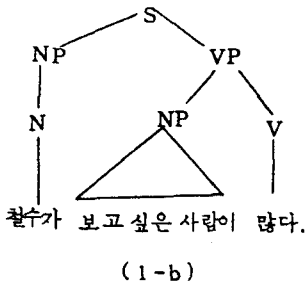
자연언어이해 시스템에서 문법은 문장 성립의 기본요건에 지나지 않으며 궁극적으로는 내포하고 있는 의미를 추출함으로써 이해의 차원에 도달할 수 있다. 단어



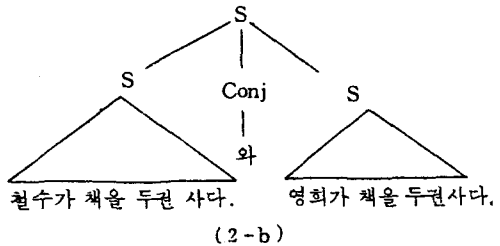
(1-a)



(2-a)

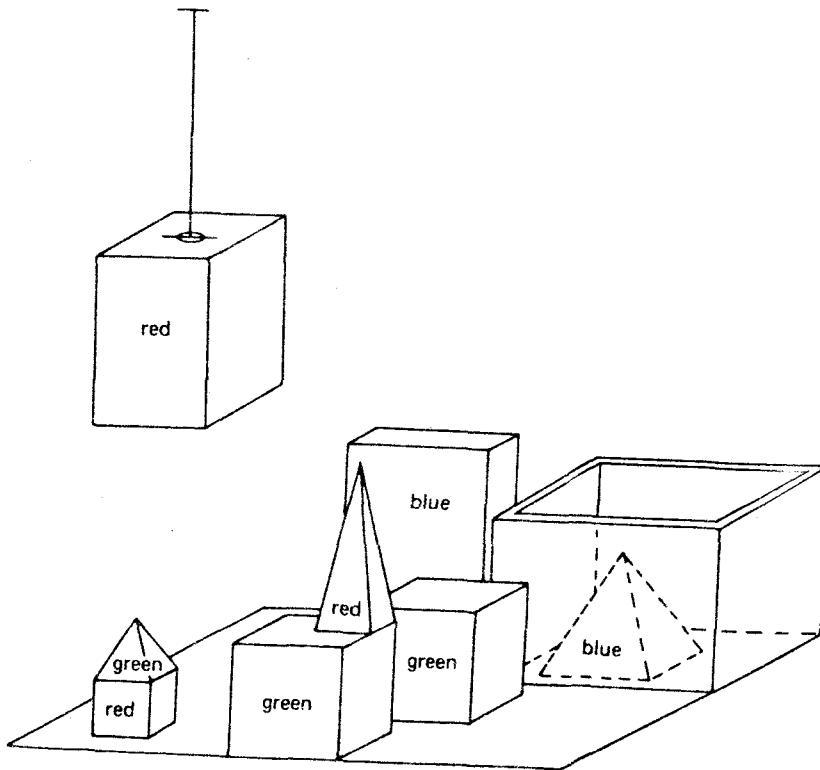


(1-b)



(2-b)

〈그림 - 11〉 애매문장의 Parsing Tree



- Pick up a big red block.
OK.
- Grasp the pyramid.
I DON'T UNDERSTAND WHICH PYRAMID YOU MEAN.
- Find a block which is taller than the one you are holding and put it into the box.
BY "IT", I ASSUME YOU MEAN THE BLOCK WHICH IS TALLER THAN THE ONE I AM HOLDING.
OK.
- What does the box contain?
THE BLUE PYRAMID AND THE BLUE BLOCK.
- What is the pyramid supported by?
THE BOX.
- How many blocks are not in the box?
FOUR OF THEM.
- Is at least one of them narrower than the one which I told you to pick up?
YES, THE RED CUBE.
- Will you please stack up both of the red blocks and either a green cube or a pyramid?
OK.
- Which cube is sitting on the table?
THE LARGE GREEN ONE WHICH SUPPORTS THE RED PYRAMID.

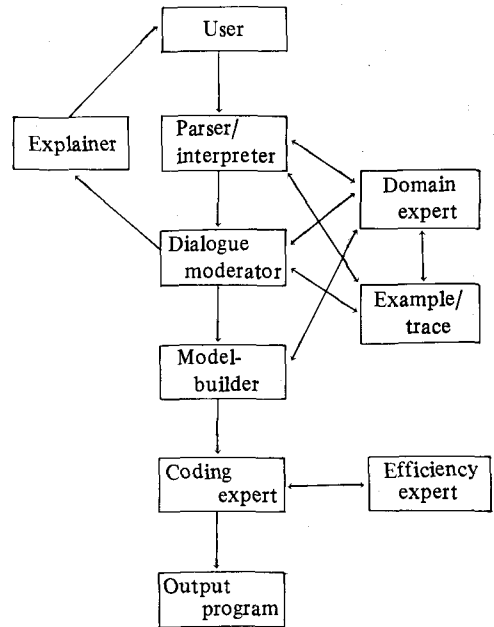
〈그림 - 12〉 SHRDLU의 대화과정

의미의 표현으로는 의미표지(semantic marker)가 사용되어 왔다. 이것은 예를 들어 “아버지”라는 단어의 의미를 아버지 : [+Animate, +Human, +Horofic, +Male, +Married, ...] 등과 같이 기본 의미소(Semantic primitive)의 결합으로 표현하는 방법이다. 의미소는 의미의 형식화 방법을 제공할 수 있지만 의미소 선정에 개관성이 없고 자연언어에서는 은유와 비유의 표현이 많기 때문에 의미 결합이 파괴되는 경우가 많다. 예를 들어 “이 냉장고는 전기를 많이 먹는다”라 하였을 때 “먹다”는 [+Animate]을 주어로 하지 않는다. 또한 의미구조 분석에서 이해의 결과를 어떠한 방법으로 표현해야 할 것인가는 지식표현과 연관된 중요한 문제이다.

자연언어이해 연구는 60년대의 침체상태를 거쳐 70년대에 들어와 Winograd의 SHRDLU system이 발표되면서 새로운 가능성이 제시되었다. 이 시스템은 가상의 로봇과 인간이 블럭(Block)에 대한 대화를 하는 것으로 상당히 고차적인 인간의 언어구조도 이해하고 실행하는 능력을 보여 주었다. <그림-12>에 이 시스템의 대화과정을 보였다. 살펴 본 바와 같이 자연언어 이해의 연구는 오랜 역사를 가지고 있다. 문법정보에 기초한 구문해석 연구, 이해에 중점을 둔 의미처리와 의미표현의 연구라는 두가지 측면이 포출되었으며 컴퓨터 처리의 관점에서 자연언어의 전반적인 현상을 고찰할 수 있었다. 이러한 자연언어처리 연구를 통하여 각종 방법과 문제점이 명확히 될 수 있었다. 자연언어는 인간의 가장 복잡한 정보표현 수단이므로 자연언어 이해의 대상은 인간의 지적 능력 전체를 대상으로 한다고 하여도 좋다. 전체 대상이 확실해지고 방법과 문제점이 명확해졌다고 해서 컴퓨터 처리가 가능해졌다는 것은 아니다. 그러나 이후 자연언어 이해는 더 현실적인 시스템으로 나타날 것만은 확실하다.

4) 자동프로그래밍

컴퓨터의 하드웨어 기술은 비약적인 발전을 가져왔지만 핵심이 되는 소프트웨어 기술은 아직 인간의 손을 거쳐야 하는 수동적인 상태에 있다. 컴퓨터가 발달함에 따라 이에 수반하는 소프트웨어의 비중은 더욱 높아져 컴퓨터 성능의 90% 이상이 소프트웨어에 의해 좌우되고 있지만 소프트웨어의 생산성은 매우 낮아 컴퓨터 발전에 커다란 장애요인이 되고 있다. 이것을 소프트웨



<그림-13> 자동프로그래밍 시스템

어 위기(software crisis)라 하며 현대의 컴퓨터가 직면한 최대의 약점이며 문제이다.

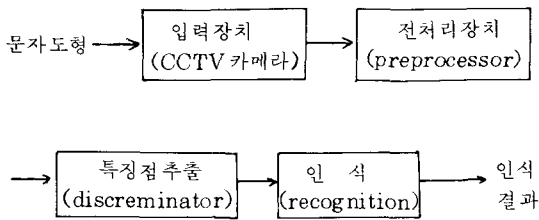
이러한 소프트웨어를 인공지능의 문제해결 기법을 도입하여 자동으로 작성하려는 것이 자동프로그래밍(automatic programming)이다. 종래의 고급언어에 의한 프로그램 작성도 넓은 의미에서 자동프로그래밍이라 할 수 있지만 그 보다 근본적으로 소프트웨어의 bottle neck을 제거하려는 노력이 인공지능 분야에 있었다.

일반 자동프로그래밍 시스템은 <그림-13>과 같이 사용자와 컴퓨터가 작성하려는 프로그래밍에 대한 대화를 통하여 컴퓨터 스스로 문제를 해결하여 프로그램을 작성한다.

이러한 자동프로그래밍 시스템은 일반 인공지능문제와 마찬가지로 유연한 문제해결 능력이 없이는 실현되기 어려운 과제이다. 현재는 주로 실험적인 시스템이 제작·운영되고 있지만 주변의 여건이 성숙되면 커다란 진전이 있을 것이 분명하다.

5) 컴퓨터 인식

인간은 대단히 우수한 감각계(Sensory system)를 갖고 있다. 이러한 감각계를 통하여 인간에게 인지되는



〈그림-14〉 패턴인식 시스템

정보는 일상생활을 유지하는 필요 불가결한 요소가 된다. 맹인의 경우 시각정보를 얻을 수 없기 때문에 불편한 생활을 영유할 수밖에 없음을 이러한 이유에서이다. 이와 같은 경우로 컴퓨터가 인간과 유사한 감각계를 갖고 있다면 다방면으로 그 유용성은 확대될 것이다. 인간의 감각계 중에서도 시각과 청각이 주로 인공지능의 연구대상이 되고 있다. 시각을 통한(image) 정보처리와 청각을 통한 음성(speech) 정보처리가 그 주요 대상이다. 이러한 감각계 정보의 인식(recognition 또는 preception)과 처리는 컴퓨터가 지능화되기 위한 필수요건이라 하여도 과언이 아니다.

영상정보처리는 차원적인 도형, 특히 문자패턴인식(character pattern recognition)을 시점으로 활발한 연구가 있었다. 컴퓨터가 인쇄글자 또는 필기글자를 그대로 인식함으로써 인간은 컴퓨터의 입력기능을 다양화할 수 있게 된다. 지금까지는 키보드가 키보드로부터 입력하여야 하는 수작업 없이 즉시 데이터의 입력이 가능해진다. 문자패턴인식 시스템은 일반적으로 〈그림-14〉와 같이 구성된다.

CCTV카메라 등 영상입력 장치를 통해 문자패턴을 입력한다. 입력패턴은 연속신호로 구성되므로 컴퓨터 처리를 위해 2차 신호 값으로 양자화(quantization)시킨다. 양자화된 패턴으로부터 인식을 위한 특징점(feature)을 추출하고 인식하는 과정을 거친다.

문자패턴은 인식방법에 따라 결정론적(decision theoretic) 방법과 구문론적(syntactic) 방법으로 분류된다.

결정론적인 방법은 패턴공간(pattern space)을 설정하여 패턴을 특징에 따라 이 공간에 투사하여 구별하는 방법이다.

구문론적 방법은 패턴의 구성요소를 추출하여 이들의

결합관계를 형식언어(formal language)를 사용하여 문법규칙을 작성하고 이것을 parsing함으로써 인식하는 방법이다. 이러한 문자패턴인식 뿐만 아니라 현재는 도형패턴과 3차원 물체인식 등 인식대상을 다양화시키고 있다. 시각정보의 인식과 응용은 그 분야가 다양하여 농작물 작황현황, 광물자원탐사, 해양탐사와 기상관측 등 여러 분야에서 응용되고 있다. 그러나 패턴인식은 유사한 패턴이 많고 인식상황에 따라 쉽게 달라지기 때문에 아직도 해결해야 할 문제가 산재하고 있다. 그러나 인공지능적 인식이 되기 위해서는 이들 인식방법이 인간의 인식과 어떠한 관계가 있는가를 명확히 하여야 하며 인간의 인식과정에 대한 철저한 규명이 선행되어야 할 것이다.

청각정보 인식은 시각패턴의 인식만큼 중요한 정보이다. 컴퓨터정보센터에 전화를 하여 “오늘의 문화행사에 대하여 안내하여 주시오” 하면 컴퓨터가 즉시 모든 정보를 음성으로 알려주는 음성응답시스템을 생각할 수 있다. 또한 말하는대로 원고를 타이핑하는 음성인식 타자기 역시 음성인식시스템의 일종이다. 그런데 음성인식시스템은 시·공간적인 음성신호를 대상으로 하고 있기 때문에 잡음, 화자의 성격 등 주변환경에 의하여 예기치 않은 문제가 많이 파생된다. 현재는 숫자나 단어의 제한, 제한된 화자 등 한정된 분야에 대하여 실용화된 시스템이 많이 있으며 계속 연구중에 있다.

3. 맺는 말

인공지능은 고도로 성숙한 정보사회를 구축하는 미래 지향적인 학문으로 주변과학의 발전으로 실용화와 시스템구축에 박차를 가하고 있다. 선진국들은 야심적인 대단위 인공지능계획과 제5세대 컴퓨터개발계획을 추진하고 있으며 도래할 정보사회의 주체가 되기 위하여 노력하고 있다. 우리도 보다 차원높은 정보문화를 구축하고 과학기술의 혁신을 가져오기 위해 인공지능 연구에 보다 많은 관심을 기울여야 할 것이다.

특히 인공지능에서 개발한 제반 기법을 산업과 각 연구분야에 응용하여 합리적이고 체계적인 응용분야의 확대와 지능을 갖춘 시스템의 개발에 중점을 두어야 할 것이다. 인공지능은 연구개발중에 있는 가능성이 많은 학문으로 그 역할은 더욱 증대되어 갈 것이다. ♣