학 습

柳 錫 仁
(正 會 員)
서울大學校 自然科學大學 計算統計學科 助教授

Ⅰ. 학습의 개념

인공지능(artificial intelligence)분야 중에서 최근에 매우 활발하게 진행되고 있는 분야는 기계학습(machine learning)분야이다. 이 분야의 연구 목적은 컴퓨터의 작업성능(performance)을 향상시키고, 또한 스스로 지식을 습득할 수 있도록 하는 것이다.

학자들에 따라 여러 견해가 있지만 일반적으로 학습시스템(learning system)이란 축적되어지는 경험을 바탕으로 새로운 상황-하지만 과거의 그 어느 것과 동일하거나 유사한 상황-에 대해 보다 효과적이고 효율적으로 반응할 수 있도록 스스로 변화해 갈 수 있는 능력을 갖춘 시스템을 말한다.[9]

이러한 학습시스템들은 주로 학습방법(learning strategy), 습득하고자 하는 지식(knowledge)내지 기술(skill)의 유형(type), 응용분야(application domain)등에 따라 분류할 수 있다.

먼저 학습방법에 따른 분류를 살펴보면 다음과 같다.

1) 단순 암기적 학습(rote learning)

이것은 같은 유형의 문제들에 대해 그 문제들의 각 상태(state)에 대한 최선의 수단들을 모두 암기해 두었다가 새로운 문제를 풀어볼 때 과거의 그것과 동일한 상태에 처하면 과거에 외워두었던 수단은 별도의 추론(inference)없이 그대로 여기에 적용시켜 풀어나가는 학습방법이다.

2) 지도에 의한 학습(learning from instruction)

일종의 교사(teacher)가 있어 시스템에 지식을 주입시켜 주고 시스템은 이러한 지식을 사용가능한 시스템 내부적 표현으로 바꾸는 학습방법으로서, 이 방법 역시 학습시스템은 어느 정도 제한적인 추론(inference)만 할뿐 학습의 많은 부분을 교사에 의존한다.

3) 유사성에 의한 학습(learning by analogy)

새로 주어진 문제를 해결하기 위해 이와 유사한, 과거에 이미 풀어본 문제들의 지식이나 해법(solution)을 이용하여 이를 현재 문제에 맞게 변형시킴으로써 현재의 문제에 대한 새로운 지식이나 해법을 얻는 학습방법이다.

4) 예를 통한 학습(learning by example)

이는 귀납적 학습(inductive learning)의 한 특수유형으로서 한 가지 개념에 대한 일련의 예(example)들과 반례(counter example)들이 주어졌을때, 시스템이 이들로부터 모든 예들은 다 포함하고 모든 반례들은 다 배제하는 하나의 일반적 개념표현(general concept description)을 도출(induce)해내는 방식이다. 예와 반례들을 일종의 교사(teacher)가 주느냐, 시스템 자신이 스스로 생성하느냐 아니면 외부 자연환경으로부터 얻느냐에 따라 이 학습방법은 더욱 세분화될 수 있다.

5) 관찰과 발견에 의한 학습(learning from observation and discovery)

교사라는 존재없이 시스템 스스로 환경 (environment)으로 부터 관찰(observation)을 통하여 이론형성(theory formation)이나 분류기준(classification criteria) 생성등의 작업을 수행하는 매우 일반적(general)인 귀납적 학습방법이다.

한편 습득하고자 하는 지식의 유형(knowledge type)별로 나누어 보면, 그것이 대수적 표현(algebraic-expression)으로 된 함수의 파라미터나 계수인 경우, 혹은 각 개체들이 속하는 집단(class)들을 서로 구별지어주는 결정트리(decision tree)인 경우, 특정언어인식(language recognition)을 학습하기 위한 형식문법(formal gramma)인 경우, 생성규칙(production rule)

인 경우, 그 밖에 정형적 논리수식과 형식들, 그래프(graph)나 네트웍(network), 프레임(frame)과 스키마(schema)인 경우 등이 있다.

또한 응용분야에 따라서는 농업, 화학, 인식모델링(cognitive modeling), 컴퓨터 프로그래밍, 교육, 전문가 시스템(expert system), 게임수행(game playing), 영상인식, 수학, 의학진단, 자연어 처리, 계획수립(planning)과 문제해결(problem solving), 로보트 공학, 음성인식 등으로 분류할 수 있다.

학습시스템에 대한 하나의 일반적 모델은 그림 1과 같다.

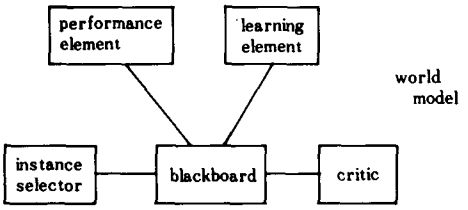

그림 1. 학습시스템의 모델

• 수행요소(performance element)

일명 문제풀이부(problem solving element)라 불리는 부분으로 학습정보를 사용하여 주어진 문제를 풀어내는 부분이다.

• 예선택요소(instance-selector)

환경(environment)으로부터 시스템이 사용하게 될 훈련 예(traing instance)들을 골라내거나 생성해낸다.

• 비평요소(critic)

주어진 문제에 대한 수행요소(performance element)의 출력 및 작용들을 해석, 평가한다.

• 학습요소(learning element)

비평요소로 부터의 비평을 바탕으로 수행요소가 사용하게 될 규칙이나 매개변수 등을 변화시키는 역할을 한다.

• 흑판(blackboard)

시스템 내의 각 요소들 간의 정보교환을 위한 일종의 공유 데이타 베이스이다.

• 환경모델(world model)

시스템의 활동영역(activity domain)을 정의하는 일반적인 가정과 방법이 포함된, 제한된 환경을 나타낸다.

Ⅱ. 문제 풀이에 학습의 활용

1. Checker Player

최초의 학습 시스템인 Samuel의 Checker Player는 단순암기적 학습방법(rote learning)과 상태함수의 파라미터 조정법(parameter adjustment)을 함께 사용하는 프로그램이다.[10]

이 시스템에서 사용되는 상태평가 함수(static evaluation function)는 $w1f1+w2f2+w3f3+\cdots$ (fi : 보드 특성(board features), wi : 가중치) 형태로서 게임이 진행되어 감에 따라 둘을 적절히 조정하여 보다 정확한 평가함수값을 얻어 유리한 게임으로 이끌어 간다.

한편 학습 개념을 탐색 영역에 도입한 대표적인 하나의 예로, Rendell은 과거 얼마동안에 풀어본 문제들의 휴어리스틱 값들의 분포를 바탕으로 회귀분석(regression)이라는 통계적 기법을 사용함으로써, 주어진 문제를 A*알고리즘으로 풀기 위한 수치적 휴어리스틱 값을 구해내는 학습 시스템을 제안하였다.

이 시스템은 주로 8-puzzle과 같은 문제 영역에 주로 사용되는데, 과거에 풀어본 문제들을 모두 저장해 두기 위해서는 기억용량의 소모가 크고, 휴어리스틱 값 분포상의 편향(bias)을 없애는 것이 어렵고, 적용범위가 비교적 제한적이라는 문제점을 가지고 있다.[7]

2. LEX

Mitchell에 의해 개발된 LEX 시스템은 예를 통한 학습방법(learning by example)을 이용하여 심볼 적분문제(symbolic integration problem)를 푸는 것을 배워나가는 학습 시스템이다.[6]

LEX에는 표 1과 같은 적분 연산자(integration operator)들로 된 초기 지식베이스가 주어지고 이들

표 1.

OP1	$\int r \cdot f(x)\,dx \Rightarrow r \int f(x)\,dx$
OP2	Integration by parts : $\int u\,dv \Rightarrow uv - \int v\,du$ (the precondition is internally represented as $\int f1(x)\ f2(x)\,dx$, where $f1(x)$ corresponds to u and $f2(x)\,dx$ corresponds to dv)
OP3	$1 \cdot f(x) \cdot \Rightarrow f(x)$
OP4	$\int f1(x)+f2(h)\,dx \Rightarrow \int f1(x)\,dx + \int f2(x)\,dx$
OP5	$\int \sin(x)\,dx \Rightarrow -\cos(x)+C$
OP6	$\int \cos(x)\,dx \Rightarrow \sin(x)+C$
OP7	$\int \hat{x} r\,dx \Rightarrow [\hat{x}(r+1)]/(r+1)+C$

연산자(operator)가 언제 적용되어야 하는지 그 상황을 나타내는 〈situation〉⇨Apply opprator OPi 형태의 휴어리스틱을 찾아 내는 것이 그 목표가 된다.

바꾸어 말하면 LEX는 각 연산자의 적용이 긍정적인 경우들과 부정적인 경우들을 감안하여 그림 2와 같은 묘사언어 트리(description language tree)로부터 각 연산자가 적용되어야 할 상황에 대한 가장 적합한 일반적 표현(general description)을 찾아가는 시스템이다.

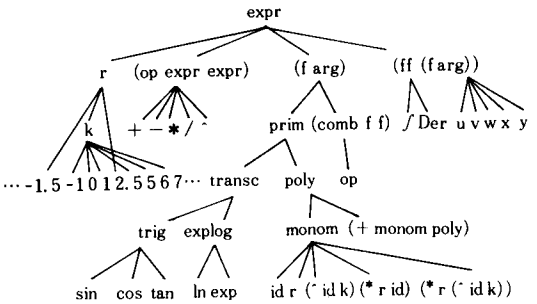

그림 2.

LEX 시스템의 전체 구성은 그림 3에 나타난 바와 같이 4개의 구성요소로 이루어졌다. 문제 해결부(problem solver)에서 푼 문제의 탐색트리(search tree)를 가지고 비평부(critic)에서는 최적해(optimal solution)를 이루는 경로 상에 있는 연산자들의 적용에 대해서는 긍정적(positive)으로, 그 이외 경로상의 연산자들의 적용에 대해서는 부정적(negative)으로 평가하며 이를 바탕으로 일반화부(generalizer)에서는 각 연산자들의 적용가능 상황에 대해 그 적용이 긍정적인 경우들은 다 포함하면서도 부정적인 경우들을 다 배제할 수 있는 한도내에서 가장 구체적인(specific) 표현들의 집합과 가장 일반적인(general) 표현들의 집합들로 그 경계(boundary)를 삼는 version space를 형성하여 이를 수렴해 가고 문제 생성부에서는 version space의 수렴을 촉진시켜 빨리 원하는 학습 수준에 도달할 수 있도록 유용한 문제들을 만들어 문제 해결부에 넘겨준다.

3. PET

Kibler와 Porter에 의해 개발된 PET 시스템은 심볼적분(symbolic integration)과 선형방정식(linear

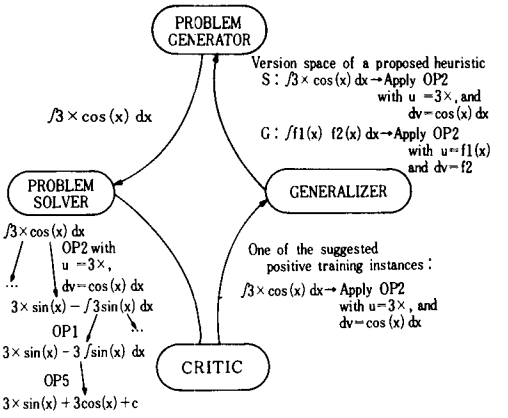

그림 3.

equation)들을 푸는데 사용되는 또 하나의 예를 통한 학습 시스템이다.

이 시스템은 개개 연산자들의 적용조건을 일반화하는 방식의 LEX 시스템과는 달리 이미 풀어본 문제들에 있어 해결로(solution path)의 일부를 이루는 연산자들의 적용 전후조건(precondition and postcondition)들을 관계모델(rdlational model)로 표현하여 일반화함으로써 유용한 연산자들의 순서(operator sequence) – 이를 에피소드(episode)라 부른다 – 들을 학습해 가는 시스템이다.[3]

PET 시스템의 또 하나 큰 특징은 학습효과를 가속화시킬 수 있는 유효적절한 훈련 예(training instance)들을 perturbation이라는 기법을 사용해서 시스템 스스로 생성해낼 수 있다는 것이다.[2]

4. Carbonell의 Analogy

또 하나의 대표적인 학습 시스템에 관한 연구로는 Carbonell의 '유사성을 이용한 학습방법' (learning by analogy)으로, 로보트 계획 수립문제(robot planning problem) 등 문제해결(problem solving) 영역에 폭넓게 적용될 수 있다. 이 학습 시스템은 과거에 푼 문제들 중에서 현재 풀어야 할 문제와 가장 유사한 것의 해(solution)를 변형(transform)시켜 현재 문제에 맞는 해를 찾아가는 과정을 하나의 탐색과정으로, 이러한 탐색공간을 T-공간(T-space), 문제의 해를 변형시키기 위해 적용될 연산자들을 T-연산자(T-operator)들로 설정하고 문제들 간의 차이를 측정해 주는 차이함수(difference function)와 각 차이와 이들을 극복할 수 있는 연산자들을 배열해 놓은 차이

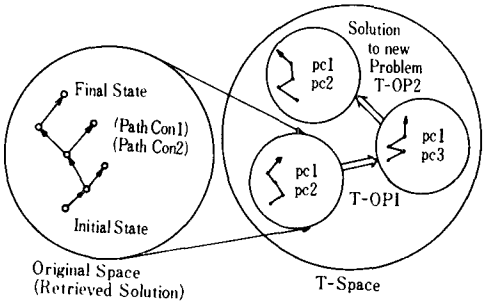

그림 4.

테이블(difference table)을 바탕으로 means-ends analysis 방식으로 적합한 해를 찾아가는 그림4와 같은 학습과정을 취하고 있다.[1]

5. LP

LP[11]는 연구된 예를 조사하여, 방정식을 풀기 위한 새로운 방법을 학습하는 프로그램이다. LP 에서는 일정한 계획하에서, 새로운 개념을 형성하는데 한 개의 예를 이용하고 있다.

연구된 예,

$$\sin(x)-2\sin(4x)+\sin(7x)=0$$
$$2\sin(4x)\cos(3x)-2\sin(4x)=0$$
$$2\sin(4x)\{\cos(3x)-1\}=0$$
$$\sin(4x)=0 \quad \text{or} \quad \cos(3x)-1=0$$
$$x=45n,\quad x=120n$$

LP의 기본적인 연산자들은 방법이라 불리운다. 각 방법은 관련 rewrite rule과 그 방법이 적용될 때를 나타내는 제어정보를 갖고 있다. 제어정보에는 전조건과 후조건이 있다. 전조건은 방법이 적용되기 전에 만족되어야 할 사실들을 나타내고, 후조건은 방법이 적용되고 난 후 만족되어지는 사실을 나타낸다. 후조건은 일정한 계획을 허용해 줄 뿐만 아니라, 원하는 결과를 달성하는 것을 보장해 준다.

LP에서는 두 단계로 학습을 가능하게 한다. 첫번째 단계는 새로운 algebraic identity을 학습하는 것으로, 이는 rewrite rule로서 사용되어질 수 있다. 다음 단계는 새로운 방법을 학습하는 것인데, 그 방법과 관련된 제어정보와 rewrite rule을 학습하는 것이다. LP는 방법들이 사용되는 순서를 조절하기 위해 상위제어정보를 학습하고, 이를 스키마라고 불리우는 계획에 저장해 둔다. LP가 해야할 첫번째 일은 연구된 예의 각 줄이 어떻게 해서 다음 줄로 변형되었는가 찾아내어 연산자 적용을 정당화하는 것이다.

이를 위해 각 줄에서 CT(characteristic tuple)을 찾는다. CT는 식에 대한 특징들로 이루어진 meta-level tuple이다. 탐색의 효율성을 높이기 위해 CT를 사용하는 것은 meta-level inference의 한 기법을 보이는 것이다. LP가 방법을 찾아내, 이를 적용한 것이 성공하면 기록하고, 성공하지 못하면 다른 가능한 방법을 찾는다. 더이상 가능한 방법이 없다면, LP는 그 단계를 설명할 수 있는 rewrite rule을 추측한다. 추측이 정확한지를 확인하기 위해 LP는 사용자에게 일반적인 규칙을 요구한다. 이러한 기법은 일반화 문제를 회피하는 수단이다.

새로운 규칙의 적용을 학습하기 위해서는 새로운 방법을 창조할 필요가 있다. i 번째 줄에 새로운 규칙을 적용하여 j 번째 줄을 만들었다고 하자. j 번째 줄에 방법 M을 적용하여 k번째 줄을 만들었을때, M의 전조건을 먼저 찾는다. 이러한 전조건중에서 i번째 줄에서 만족되고 있는 것을 S라 하고, j번째 줄에서 만족되고 있는 것을 U라고 한다. U가 공집합이 아니면, 새로운 규칙을 적용한 목적은 M이 적용될 수 있도록 U가 만족되어지도록 하는 것이다. U는 새로운 규칙 적용의 주요한 결과가 된다. 새로운 방법의 전조건은 S와 새로운 규칙의 전조건을 합한 것이다. U가 공집합이면 다른 설명을 찾고, 새로운 전조건을 만족시키지 않더라도 M을 적용하여 식을 변화시킬 수 있는 가능성도 있다. 이러한 가능성은 모호한 연산자의 경우 흔히 일어날 수 있는 상황이다. 이렇게 해서 스키마가 만들어지고, 연구된 예에 적용되어진 모든 방법들을 스키마에 나타내게 된다. 이를 새로운 방정식 풀이에 적용하기 위해 관련 스키마가 있는가를 찾아내는 프로시듀어는 다음과 같다.

(1) 방법 M을 적용하여 성공하면, 스키마의 다음 방법으로 넘어가 계속한다.

(2) 실패하면 M과 같은 결과를 가져오는 다음 방법을 찾고, 이 방법이 성공하면 스키마의 다음 방법으로 넘어가 계속한다.

(3) 1, 2가 실패하면 벌써 만족되어진 조건들에 영향을 미치지 않는 다른 방법을 찾고, 이 방법이 성공하면, (1)로 간다.

(4) 위 세가지가 성공하지 못하고 M이 어떤 중요한 결과를 가지지 않으면 2단계는 뛰어 넘는다.[12]

Ⅲ. 자연어 처리에 학습의 활용

1. 문제(형태소 분석 및 합성)

형태소(morpheme)는 독립된 의미를 지닌 최소의 언어표현 단위로서 더 큰 언어표현(단어, 구, 절, 문장 등)을 구성하는 기본 요소이다. 어절이나 단어로부터 그 구성성분이 형태소 단위로 분리하는 작업을 형태소 분석(morphological analysis)이라 하며, 역으로 표현하고자 하는 의미들로부터 그들을 결합하여 올바른 어절이나 단어를 생성하는 작업을 형태소 합성(morphological synthesis)이라 한다. 예를 들어서, 한국어에서 하나의 어절 "도왔다"를 형태소 분석하면 〔돕+았+다〕가 되고, 〔돕+PASTX+DECL〕로부터 "도왔다"를 생성하는 작업이 형태소 합성이다(PASTX는 과거시제를 의미하는 기호이며, DECL은 서술격을 의미하는 기호라고 정의할 때).

형태소 분석과 합성은 자연언어를 인터페이스로 사용하는 시스템(기계번역 시스템, 질의·응답 시스템, 전문가 시스템, 교육시스템, 데이타베이스 시스템 등)에서는 기본적인 작업이며, 이 작업이 올바로 행하여 지지 않으면 다른작업이 제대로 수행될 수 없거나 소기의 목적을 달성할 수 없기 때문에 중요하다. 특히 한국어에서는 형태소 간의 결합 관계가 복잡하고, 굴절현상이 심하여 철자 변형의 복잡하게 일어나므로 이러한 규칙을 일일이 손으로 코딩하여 시스템을 유지 보수하는데는 많은 어려움과 노력이 따른다. 문법 규칙을 기계가 스스로 학습할 수 있다면 이러한 사람의 노력을 대신할 수 있을 것이다.

2. XMAS 개관

시스템 XMAS(xpert in morphological analysis and synthesis)는 다중언어 범용의 형태소 분석 및 합성 시스템이다.[13,14] XMAS는 학습기, 지식베이스, 문제해결기로 구성된 학습 시스템이다. 학습기는 형태소 결합의 예로부터 문법 규칙을 자동 습득한다. 학습된 규칙은 지식베이스에 저장되며, 문제해결기는 이 규칙을 이용하여 형태소 분석과 합성을 한다.

XMAS에서는 형태소 분석과 합성을 상태와 연산자로 구성된 상태공간에서의 탐색문제로 본다. 상태는 임의의 형태소나 그들의 결합체인 스트링으로 표현되고, 연산자는 이 스트링을 다시 쓰기 또는 변형한다. 형태소 분석에 있어서 초기상태는 하나의 스트링으로 주어지며, 목표상태는 공백스트링이다. 형태소 합성 문제에서는 초기상태가 공백 스트링이며, 목표상태는 하나의 스트링이다. 형태소 분석(또는 합성)에 있어서 주어진 초기상태에 대한 문제공간을 그래프로 표현한 것을 분석(또는 합성) 그래프라 한다.

3. 형태소 결합 규칙의 기계학습

XMAS는 이 그래프 탐색 문제를 연산자와 그 적용조건을 학습함으로써 해결한다. 연산자의 적용조건의 학습은 탐색 휴리스틱(heuristic)을 학습하는 것이며, 문제해결기는 이 휴리스틱을 이용하여 탐색을 하고 해당 연산자를 적용함으로써 상태전이를 한다. 따라서 연산자와 그 적용조건은 형태소 합성 및 분석을 위한 문법 규칙이 된다.

문법규칙 = 연산자 + 적용조건(탐색 휴리스틱)

문제는 ① 연산자를 어떻게 습득하는가와 ② 연산자의 적용조건 휴리스틱을 어떻게 학습하느냐이다. 그래프의 탐색은 곧 상태 1 (before, b)에서 상태 2 (after, a)로의 전이의 반복이며 이 전이는 연산자(op)의 적용에 의하여 일어난다.

b ----op----> a

이러한 상태전이의 예는 자연언어의 특성상 무수히 많다. 교사(teacher)는 상태전이의 예를 학습 시스템에 제시함으로써 학습 시스템이 이 예로부터 연산자를 발견하고(관찰에 의한 학습), 같은 연산자에 대한 많은 예들로부터 이 연산자에 대한 적용조건을 일반화 함으로써(예로부터의 학습) XMAS로 하여금 문법 규칙을 학습하도록 한다. 그림 5는 ("아름답"→"아름다운")과 ("곱" → "고운")의 두 개의 예로부터 하나의 연산자를 발견하고 그 적용조건을 학습하는 과정을 간략히 기술한 것이다(상세한 알고리즘과 이 규칙을 이용한 형태소 분석과 합성에 대해서는〔13〕, 〔14〕를 참고).

1-1. "아름답" → "아름다운"

1-2. (%ㅇㅏㄹㅡㅁㄷㅏㅂ%) →
Bl Br

(%ㅇㅏㄹㅡㅁㄷㅏㅇㅜㄴ%)
Al Ar

1-3. (ㅏㅂ%) → (ㅏ운%) ==> opl : ㅣ=ㅂ/운%ㅣ

1-4. Rl : (ㅣ=ㅂ/운%ㅣ((ㅏㅂ%) → (=운%)))

2-1. "곱" → "고운"

2-2. (%ㄱㅗㅂ%) → (%ㄱㅗㅇㅜㄴ%)
Bl Br Al Ar

2-3. (ㅗㅂ%) → (ㅗ운%) ==> opl : ㅣ=ㅂ/운%ㅣ

2-4. R2 : (ㅣ=ㅂ/운%ㅣ((ㅗㅂ%) → (=운%)))

2-5. R1과 R2로부터 일반화한다.

R3 : (ㅣ=ㅂ/운%ㅣ((PVㅂ%) →
(=운%)))

그림 5. 문법 규칙의 학습

첫번째의 예로부터 "ㅏㅂ"이 "ㅏ운"으로 바뀌었다는 것을 찾아내어(그림에서 포인터 Bl과 Al은 왼쪽에서 오른쪽으로 진행하며 문자를 비교하고, Br과 Ar은 오른쪽에서 왼쪽으로 진행하면서 문자를 비교함으로써), 이것으로부터 새록용 연산자 opl을 하나 생성하며 그 적용조건과 함께 하나의 규칙Rl으로표현한다. 두번째 예로부터는 "ㅗㅂ"이 "ㅗ운"으로 변형된 것을 인식하며 이로부터 하나의 연산자를 만든다. 이것을 마찬가지 방법으로 R2로 표현하며 R1과 R2는 같은 연산자에 대한 예이므로 이 둘로부터 연산자 opl에 대한 일반화된 휴리스틱 규칙 R3를 얻는다. 규칙 R3의 의미는 〈스트링의 끝이 양성모음+"ㅂ"이면 "ㅂ"을 지우고 "운"을 첨가한다〉이다 (PV는 양성모음을 나타내는 기호임).

4. 실험 및 응용 결과

앞에서 설명한 형태소 결합 규칙의 기계학습 방법은 한국어 뿐만 아니라 다른 자연언어에도 마찬가지로 적용된다. 한국어와 영어에 대한 다양한 실험 결과, XMAS는 이미 언어학자들이 발견한 문법규칙을 재발견할 수 있었으며, 이를 이용하여 형태소 분석과 합성을 할 수 있었다.

XMAS의 한국어 버젼은 현재 영어-한국어 기계번역 시스템 KSHALT[15]의 한국어 생성에 성공적으로 사용되고 있다. XMAS는 문법규칙을 예로부터 자동적으로 발견하고 습득한다는 점에서 흥미로울 뿐만 아니라, 기계학습 방법을 실제적인 자연언어 처리 시스템에 응용함으로써 시스템을 개발하고 유지보수하는데 드는 노력과 시간을 현저히 줄여 주었다는데 큰 의의가 있다.

Ⅳ. 개념에 의한 무리짓기 (Concept Clustering)에서 학습의 활용

1. 개념에 의한 무리짓기의 개념

자연계에서 관찰되는 객체들은 하나의 개념을 형성하기 위해서 유사성을 가진 객체들끼리 하나의 부류를 이루고 있다. 예를 들면, 토끼, 여우, 호랑이들로 구성된 부류 A와 뱀, 매, 개구리들로 구성된 부류 B가 주어졌을때, 부류 A와 부류 B사이에는 어떤 차이점이 있고, 부류 A가 어떤 공통성을 가진 객체들로 구성 되었는지를 추론함으로써 포유동물이라는 개념을 학습할 수 있다.

개념에 의한 무리짓기는 사회과학자나 자연과학자에 의하여 종래에 사용되던 수치적 분류 방법의 확장으로 1980년 Michalski에 의하여 처음으로 정의되었다. 개념에 의한 무리짓기는 기계학습의 관점에서 볼 때 관찰을 통한 학습방법에 속한다. 이것은 객체들이 주어졌을때 유사한 객체들과 객체들간의 유사성을 측정할 수 있는 척도가 마련되어야 하고 이 기준에 따라 객체들을 여러개의 무리로 나누어서 이 무리가 내포하고 있는 개념을 표현하여야 한다. 무리짓기 프로그램을 사용하기 전에 자료분석자에 의하여 명세화 되어야 할 요인들을 살펴보자.

1) 무리지을 객체들과 그들이 가지는 속성

객체들은 사건, 상태, 관찰등을 통하여 만들어지고 이 객체들은 속성들에 의하여 묘사되어진다. 각 속성들은 여러 유형의 값들로 정의된다. 예를들면 이름, 순서, 구간, 비율, 절대값등이 있다.

2) 객체들을 무리로 분류하는 원칙

객체와 객체사이에 유사성이 큰 것끼리 하나의 무리를 이루고 무리와 무리사이에는 차이가 뚜렷이 나도록 객체들이 분류되어야 한다. 객체사이의 유사성은 객체상호간의 거리로 구할 수 있다.

3) 무리에 대한 표현방법

무리에 대한 표현 목적은 한 무리에 속한 객체들의 특성을 간결하고 일반적으로 표현하는 것이다.

- 목록 묘사
- 구조적 묘사
- 선형 묘사

4) 무리의 질에 대한 판단기준

- 무리에 대한 묘사와 관찰된 객체가 서로 잘 맞는가에 대한 정도
- 무리에 대한 묘사의 단순성 정도
- 무리 사이의 차이정도
- 구별되는 색인 수

2. 객체를 무리짓는 방법

1) 최적화 기법

이 방법은 하나의 속성을 하나의 자원으로 하는 벡터공간에 모든 관찰되는 객체를 표시하고 이 객체들이 상호배타적으로 하나의 무리에 속하게 나누는 방법이다. CLSTER/2[16] 분할 모듈이 최적화 기법으로 무리를 지은 예는 그림 6 과 같다.

2) 계층화 기법

이 기법은 객체의 집합위에서 분류트리를 구하는 것이다. 트리의 각 잎은 개개의 객체를 나타내고 내부노드는 객체의 무리를 나타낸다. 이 계층화 기법은 하향식과 상향식 방법으로 분류트리를 만들수 있

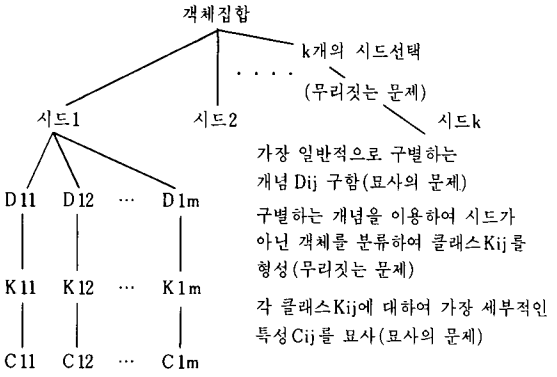

그림 6. CLUSTER/2의 분할 모듈

다. 여기서는 객체의 속성이 무리를 짓는데 직접 이용된다. 임의의 속성에 대하여 같은 속성값을 가지는 객체는 하나의 무리에 속할 수 있는 가능성을 가진다. 따라서 적절한 속성이 선택 되기만 하면 부분적 결정의 연속으로 훌륭한 무리짓기를 할 수 있다. 최적화 기법보다 적은 시간에 무리를 지을 수 있으므로 객체들이 많은 수의 속성들로 표현되는 문제 영역에 적합하다. RUMMAGE, DISCON등이 계층화 기법을 이용한다. 이 프로그램은 객체들을 나타내는 여러개의 속성이 주어졌을 때 하나의 속성에 대하여 같은 속성값을 가지는 객체들을 하나의 무리로 분류하는 방법을 택하고 있다. 이것을 그림 7 과 같이 나타낼 수 있다.

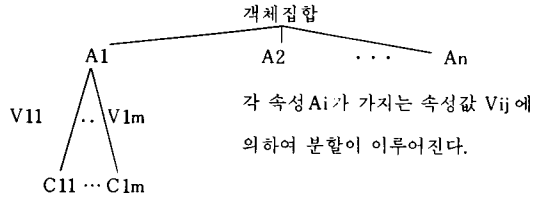

그림 7. RUMMAGE, DISCON의 무리짓기

3) 중첩화 기법

이 방법은 주어진 객체들의 집합위에서 무리들이 서로 중첩되는 것을 허용한다. UNITEM[17]이 중첩화 기법을 사용하는 대표적인 프로그램이다. 이 프로그램은 복잡하고 많은 수의 속성들에 의해서 객체들이 묘사되는 문제 영역에 적합하다. 여기서는 몇 개의 예제를 이용하여 그 속에 내포되어 있는 인과관계를 찾아내고 이것을 일반화 시켜서 하나의 개념을 얻어내는 방법을 사용한다. 구해지는 개념도 일반화의 정도에 따라 개념이 계층적으로 만들어진다. UNITEM의 특징은 개념을 학습하고 학습된 개념이 효과적으로 기억되도록 일반화를 바탕으로한 기억 방법을 이용한다.[18]

3. 무리에 대한 개념 묘사방법

1) 무리의 특성을 일반화시키는 전략

개념에 의한 무리짓기에서 각 무리의 특성에 대한 묘사는 예를 통한 학습방법과 동일하다. 따라서 기존의 여러가지 예를 통한 개념묘사 방법을 그대로 사용하면 된다. 개념을 묘사할 수 있는 공간은 일반성에 따라 순서가 주어진다.

2) 가정공간의 탐색방법

적절한 가정을 선택하기 위하여 가정공간의 탐색은 완전한 탐색과 불완전으로 구별할 수 있다. 완전한 탐색은 깊이 우선 탐색과 너비 우선 탐색으로 나누어진다. 불완전 탐색은 구해진 가정에 대한 평가함수가 정의된다. 따라서 같은 탐색방법을 사용해도 평가함수가 같지 않으면 다른 결과를 나타낸다. 불완전 탐색방법은 비임 탐색과 최적 우선 탐색으로 나누어진다.

3) 자료의 제공방법

관찰되는 객체들이 한꺼번에 전부 제공되느냐, 순차적으로 하나씩 제공되느냐의 두가지 방법으로 나누어진다.

V. 결 론

학습의 개념을 살펴보고, 학습이 활용되는 문제의 영역에서 몇개의 대표적인 시스템을 통하여 그 구현 방법 또는 원리를 생각해 보았다.

현재로써는 모든 문제 영역에 적합한 학습기법이 제시되지 않고 있다. 기계를 더욱 영리하게 만드는 것이 인공지능의 가장 큰 목표일진데, 일반성 있는 학습기법이 연구되어진다면 인공지능의 실용화에 많은 기여를 할 수 있을 것이다. 아직까지는 현실화되고 있지는 않지만, 좁은 문제 영역에서의 성공 추세를 감안하면 범용적으로 활용될 수 있는 학습 시스템이 곧 출현하리라고 생각한다.

국내에서는 학습에 관한 연구가 부족한데 인공지능의 기본적 근간을 이룬다는 점에서 이론적, 실용적 연구가 활발히 추진되어야 할 것이다.

參 考 文 獻

[1] Carbonell, J.,"Learning by Analogy: Formulation and Generating Plans from Past Experience," in Machine Learning, Michalski, Carbonell, Mitchell (Eds.), pp. 137-162, Tioga Press, 1983.

[2] Kibler, D. and Porter, B., "Pertubation: A means for guiding generalization," Proc. IJCAI, pp. 415-418, 1983.

[3] Kibler, D. and Porter, B., "Episodic learning," Proc. of National Conf. on A.I, pp. 191-196, 1983.

[4] Korf, R.E., "Macro-operators: A weak method for learning," *Artificial Intelligence,* vol. 26, pp. 35-77, 1985.

[5] Mitchell, T., "Generalization as search", *Artificial Intelligence,* vol. 18, pp. 203-226, 1982.

[6] Mitchell, T., Utgoff P. and Banerji R., "Learning by Experimentation: Acquiring and Refining Problem Solving Heuristics", in Machine Learning, Michalski, Carbonell, Mitchell (Eds.), pp. 163-190. Tioga Press, 1983.

[7] Rendell, L.A., "A new basis for state-space learning systems and a successful implementation," *Artificial Intelligence,* vol. 20, July 1983.

[8] Simon, H.A.,"Why Should Machine Learn?", in Machine Learning, Michalski, Carbonell, Mitchell (Eds.), pp. 25-37, Tioga Press, 1983.

[9] Smith, R.G., Mitchell, T., Chestek, R.A. and Puchanan, B.G., "A model for learning systems," Knowledge Acquisition, vol. 3.

[10] Samuel, A.L., "Some studies in machine learning using the game of checkers," IBM Journal of Research and Development, no. 3, pp. 211-229, 1959.

[11] Silver, B., "Learning equatior solving methods from examples," Proceedings of the Eight IJCAI, Karlsruhe, W. Ger., pp. 429-431, 1983.

[12] Silver, B., *Meta-Level Inference,* North-Holland, Amsterdam, 1986.

[13] 장병탁, "문법 규칙의 기계학습에 의한 형태소 분석 및 합성," 석사학위 논문, 서울대학교 전자계산기공학과, pp. 92, 1988.

[14] 장병탁, 유석인, 김영택, "XMAS : 형태소 분석 및 합성 전문가," 88춘계 인공지능 학술발표 논문집, 한국정보과학회 인공지능연구회, pp. 179-188, 1988.

[15] S.S. Kang, K.S. Shim, B.T. Zhang, H.C. Kwon, C.S. Woo, and Y.T. Kim, "An English-Korean System for Human Assisted Language Translation," Proc. of TENCON 87, IEEE, pp. 550-555, 1987.

[16] Michalski, R.S., Stepp, R.E., "Learning from Observation: Conceptual Clustering," in Machine Learning: An Artificial Intelligence Approach, Tioga, Palo ASlto, Calif., 1983.

[17] Lebowitz, M., "Integrated Learning: Controlling Explanation," Cognitive Science 10, pp. 219-240, 1986.

[18] Winston, P.H., "Learning by augmenting rules and accumulating censors," in Machine Learing: An Artificial Intelligence Approach, vol. 2., Morgan Kaufmann Pub., 1986.

♣ 用 語 解 說 ♣

Data model (데이타 모델)

현실 세계를 데이터 베이스의 입장에서 취한 추상적 묘상(描像)을 데이터 모델이라 한다. 이러한 데이터 모델은 추상화의 수법에 따라 여러가지 존재하고 있으나 현실의 데이터 베이스 시스템을 실현하기 위하여 사용되고 있는 대표적인 것으로 계층모델, 네트워크 모델 및 관계모델이 있다. 실체 및 관련 모델등 현실 세계를 보다 충실하게 묘사하는 것을 목적으로 한 데이타 모델은 데이타 베이스 시스템의 모델로서 뿐만 아니라 현실적으로 되도록 알맞은 방법으로 데이터를 축척하기 위한 데이터 베이스 설계에 이용되는 것이 보통이다.