

Contents

- 인공지능에 쓰이는 논리와 추론
- 인공지능에서의 탐색 기법
- 인공지능과 알고리즘
- 규칙기반 전문가 시스템



∰ 인공지능에 쓰이는 논리와 추론

● 규칙기반 인공지능에서의 논리와 추론

- 규칙기반 인공지능은 논리 바탕의 규칙을 통해 추론
- 논리는 다양한 논리 연산으로 폭넓게 활용됨
- 탐색 방법, 문제 해결 알고리즘, 지식처리 등에 필요
- 전문가 시스템 등의 응용에 필수적인 기반 지식
- 논리와 다양한 논리 연산 등의 기초지식이 필요함

• 논리란 무엇인가?

- 인간의 사고가 논리적인지를 판단하는 기준
- 토론이나 논쟁에서 중요한 것은 주장의 논리성
- 상대방을 논리적으로 설득하는 것이 매우 중요
- 객관적이고 명확한 사고 법칙의 준수로 결정됨
- 논리를 통한 입증에 필요한 법칙 제공



● 논리의 활용

- 컴퓨터 관련 학문이나 공학 등의 분야에 폭넓게 응용됨
- 규칙기반(rule-based) 인공지능에 이론적 기반 제공
- AI 학자들이 논리를 규칙기반 인공지능의 실현에 적용
- AI에서 지식 표현이나 추론(inference)의 도구로 쓰임

● 논리 연산

- 명제(propositional) 논리와 술어(predicate) 논리
- 명제 논리는 전체의 참/거짓을 판별하는 법칙 다룸
- 술어 논리는 주어와 술어로 구분하여 참/거짓 법칙 다름
- 명제(proposition)는 논리에서 가장 기초적인 개념
- 명확하게 참(T)/거짓(F)을 구분할 수 있는 문장이나 식
- "2 × 7의 값은 홀수다."는 거짓인 문장
- "사과는 맛있다."는 참/거짓 구분 안 되니 명제 아님
- 합성 명제는 둘 이상의 명제가 결합된 명제



● 논리 부정 (negation)

- 명제 p에 대한 부정은 명제 p의 반대되는 진리값
- 기호로는 ~p, 'not p' 또는 'p가 아니다'라 읽음
- 만약 p의 진리값이 T이면 ~p의 진리값은 F
- p의 진리값이 F이면 ~p의 진리값은 T

p	~ <i>p</i>
Т	F
F	Т

논리합 (disjunction)

- 명제 p, q가 '또는(OR)'일 때 p, q의 논리합
- ⁻ p∨q로 표시, 'p or q' 나 'p 또는 q'라 읽음
- p∨q는 두 명제가 모두 F인 경우에만 F의 진리값 그렇지 않으면 모두 T의 진리값
- "사과는 채소이거나, 시금치는 채소이다."는 T

p	q	pVq
Т	Т	Т
Т	F	Т
F	Т	Т
F	F	F

● 논리곱 (conjunction)

- 명제 p, q가 '그리고(AND)'일 때 p, q의 논리곱
- ⁻ p∧q로 표시, 'p and q' 또는 'p 그리고 q'라고 읽음
- ⁻ p∧q는 두 명제가 모두 T인 경우에만 T, 그 외엔 F
- "사과는 과일이고, 시금치는 채소이다."는 T
- "서울은 대한민국의 수도이고, 3 × 2 = 5이다."는 F

p	q	$p \land q$
Т	Т	Т
Т	F	F
F	Т	F
F	F	F

● 조건 (implication) 또는 함의

- p, q의 조건 연산자는 p → q, 'p이면 q이다'라 읽음
- $^{-}$ p \rightarrow q는 p가 T이고 q가 F일 때만 F, 그 외 모두 T
- →는 인공지능에서 (if p, then q)로 많이 쓰임
- "유채꽃이 노랗다면, 바다가 육지이다."는 F
- "3 × 2 = 5라면, 런던은 미국의 수도이다."는 T
- 즉 조건 명제에서는 가정이 F이면 결론은 항상 T

p	q	$\rho \rightarrow q$
Т	Т	Т
Т	F	F
F	Т	Т
F	F	Т

술어 논리 (predicate logic)

- 변수의 값에 따라 T 또는 F가 결정됨
- 'x2 + 5x + 6 = 0'는 x값이 2나 3일 경우에만 T
- 술어 논리는 지식을 논리적 식으로 표현
- 술어 논리에서는 대상들 간의 관계도 나타낼 수 있음
- "영철이는 남자다."는 남자(영철)와 같이 나타낼 수 있음
- "영철의 아버지는 김춘추 씨다."라는 문장 '아버지(김춘추, 영철)'과 같이 표현 가능

- 추론의 의미와 인공지능
 - 추론이란 '알려진 사실이나 명제를 근거로 삼아 미지의 사실에 대한 판단이나 결론을 이끌어내는 사고 과정'
 - 글의 앞뒤 사실로 미루어 추론하는 것과 같은 원리
 - Prolog(프롤로그)는 추론에 주로 쓰이는 프로그래밍 언어



● 귀납 추론 (induction)

- 개별적인 사실들로부터 일반적인 결론을 이끌어내는 방법
- 연역 추론과 같은 논리적 필연성이 적음
- 새로운 지식이나 이론의 발견과 확장에 쓰이는 추론 방법

귀납 추론의 예

까마귀 1은 까맣다.

까마귀 2도 까맣다.

까마귀 3도 까맣다.

까마귀 4도 까맣다.

• • • • • • •

까마귀 n도 까맣다.

따라서 "모든 까마귀는 까맣다."라고 추론

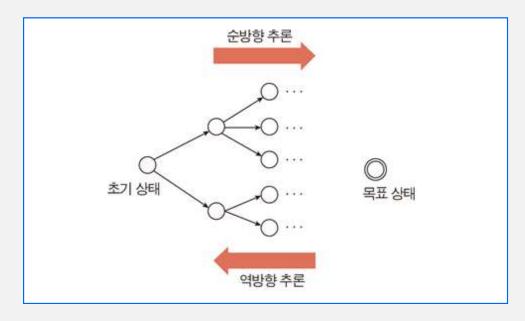
● 연역 추론 (deduction)

- 연역 추론은 가장 널리 쓰이는 추론 방법
- 전제로부터 다른 결론을 도출해내는 추론 방식
- 최초의 대전제가 결론을 이끌어내는 가장 중요한 근거
- 확장성은 부족하나 논리의 일관성이 장점
- 연역 추론 중 삼단논법이 잘 알려짐
 - ❖ 대전제 소크라테스는 사람이다.
 - ❖ 소전제 사람은 모두 죽는다.
 - ❖ 결론 따라서 소크라테스는 죽는다.

遊 인공지능에서의 탐색 기법

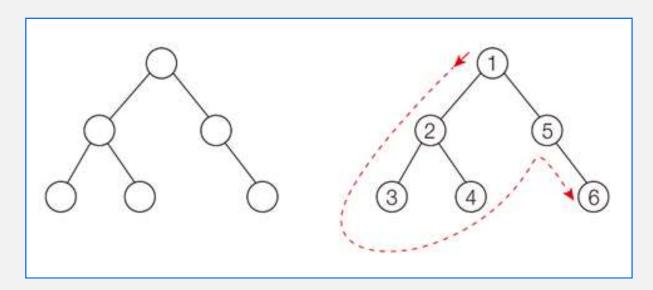
● 추론을 통한 탐색

- 순방향 추론과 역방향 추론이 있음
- 순방향 추론은 출발 상태에서 목표 상태로 진행
- 역방향 추론은 목표 상태에서 출발 상태로 진행
- 순방향 추론과 역방향 추론을 접합한 양방향 추론도 있음



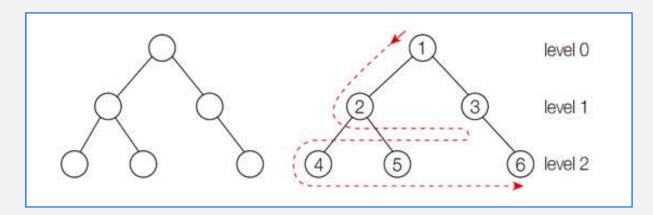
心 인공지능에서의 탐색 기법 (Cont'd)

- 깊이 우선 탐색 (Depth First Search, DFS)
 - 첫 정점(node) 방문, 왼쪽으로 이동하여 계속 탐색
 - 탐색할 정점이 없으면, 되돌아와서 순환적으로 탐색
 - 그림에서 1, 2, 3, 4, 5, 6의 순서로 방문



心 인공지능에서의 탐색 기법 (Cont'd)

- 너비 우선 탐색 (Breadth-First Search, BFS)
 - 시작 정점 방문 후 시작 정점과 연결된 모든 정점들을 왼쪽부터 차례대로 방문
 - 그 후 level의 순서에 따라 차례로 탐색
 - ⁻ 즉 level 0, level 1, level 2, ... 의 순서로 탐색
 - 그림에서 1, 2, 3, 4, 5, 6의 순서로 방문



心 인공지능에서의 탐색 기법 (Cont'd)

● 휴리스틱 탐색 (heuristic search)

- 무정보 탐색(uninformed search)은 사전 정보를 사용하지 않고 일정한 순서대로만 탐색
- 깊이 우선 탐색과 너비 우선 탐색은 무정보 탐색임
- 휴리스틱 탐색은 탐색 과정에서 경험적 지식을 활용

帶 인공지능에서의 탐색 기법 (Cont'd)

- 최소최대 탐색 (minimax search)
 - 주로 상대가 있는 2인용 게임에서의 탐색
 - 미국 어린이들이 즐기는 틱텍토(Tic-Tac-Toe) 게임
 - 가로, 세로, 대각선으로 3개가 연속될 경우 이기는 게임
 - 나의 가능성 최대, 상대방의 가능성 최소 전략 탐색
 - 탐색이 필요 없는 경로를 잘라내는 작업을 전지 작업(pruning)



∰ 인공지능과 알고리즘

● 알고리즘(algorithm)이란 무엇인가?

- 인공지능에서 알고리즘이란 용어가 자주 나옴
- 신경망에서의 역전파 알고리즘 등
- 문제 해결을 위한 단계들을 체계적으로 명시한 것
- 문제를 해결하는 방법의 상세한 특징을 기술하는 것
- 표현 방법으로는 순서도, 유사 코드, 언어적 표현 등
- 알고리즘은 단 하나만 있는 것이 아님



心 인공지능과 알고리즘 (Cont'd)

- 생활 속의 알고리즘
 - 기초적인 수학 연산
 - ✓ 덧셈과 곱셈 방법, 최대공약수를 구하는 방법 등
 - 라면 조리법
 - ✓ 라면을 맛있게 끓이기 위한 조리 방법
 - ✓ 물의 양, 불의 세기, 끓이는 시간 등이 주요 요소



(Cont'd) ### 인공지능과 알고리즘

- 생활 속의 알고리즘 (cont'd)
 - 하루의 일정 계획
 - ✓ 아침에 학교에 가려고 할 때의 일정 계획
 - ✓ 기상 시간, 식사 시간, 버스나 지하철 이용 등
 - 가전제품의 사용 매뉴얼
 - ✓ 세탁기나 전자레인지 등의 단계별 사용 설명서

心 인공지능과 알고리즘 (Cont'd)

- 생활 속의 알고리즘 (cont'd)
 - 지하철 환승 방법
 - ✓ 어느 지하철역에서의 환승이 효율적인지 판단
 - 알파고와 같은 인공지능 방법론
 - ✓ 바둑이나 게임 등에서 복잡한 것 해결 방법
 - ✓ 그 외 생활 속의 알고리즘이 매우 많음

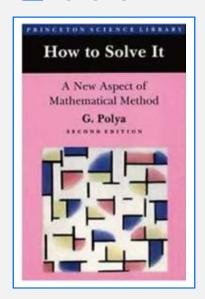
心 인공지능과 알고리즘 (Cont'd)

● 휴리스틱(heuristic)을 이용한 알고리즘

- 휴리스틱은 비슷한 문제에 대한 과거의 경험들을 바탕으로 직관적으로 판단하여 선택하는 의사결정 방식
- 인공지능에서 휴리스틱 알고리즘이 상당히 많이 쓰임
- 논리적이거나 최적의 방법을 보장하는 것은 아님
- 만족할만한 해결책의 비교적 빠른 실용적인 방법
- 현실적으로 만족할 수 있는 답을 찾는 접근법
- 인공지능, 심리학, 경제학 분야에서 많이 사용

鄭 인공지능과 알고리즘 (Cont'd)

● 폴리아의 『How to solve it』에서의 휴리스틱 예

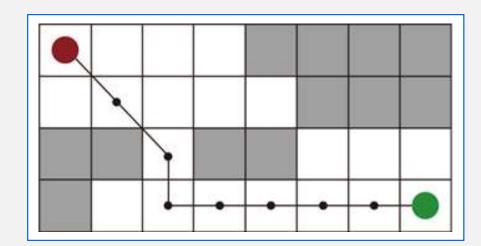


- 어떤 문제를 이해하기 어려우면 그림으로 그려보기
- 해답을 얻었다고 가정하고 반대 방향으로 유도 해보기
- 그 문제가 추상적이면 구체적인예를 시도 해보기

心 인공지능과 알고리즘 (Cont'd)

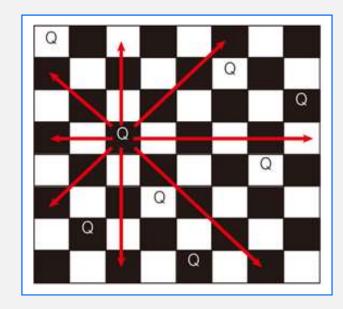
● 인공지능 A* 알고리즘

- A* 알고리즘은 최상우선 탐색 중 가장 잘 알려짐
- 문제 해결에 매우 효과적인 탐색 알고리즘
- 출발점에서 목표지점까지의 최적 경로 탐색의 한 방법
- 가장 비용이 적거나 짧은 경로 찾기
- 평가 함수 f를 사용하여 다음에 이동할 경로를 결정함



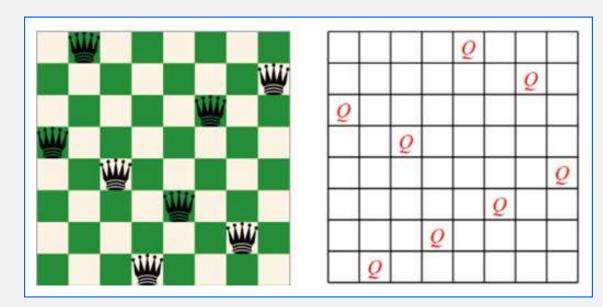
心 인공지능과 알고리즘 (Cont'd)

- 인공지능 알고리즘의 예: 8-Queens 문제
 - 8-Queens 문제는 체스에서 유래
 - 서로를 공격하지 않는 위치에 8개의 퀸(Q) 배치
 - 퀸은 수평, 수직, 대각선 방향으로 몇 칸이든 이동 가능
 - 어떤 퀸도 서로 공격 당하지 않는 위치에 놓여야 함



鄭 인공지능과 알고리즘 (Cont'd)

- 인공지능 알고리즘의 예: 8-Queens 문제
 - 64개의 칸을 가진 8-Queens 문제의 경우의 수?
 - 약 44억 가지의 엄청난 경우의 수
 - 통상 인공지능 방법으로 프로그램 하여 해답을 구함
 - 몇 가지의 해답이 나옴



鄭 인공지능과 알고리즘 (Cont'd)

• 4-Queens 문제

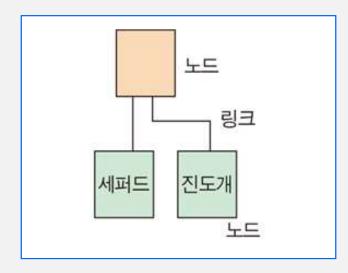
- 체스 판을 4 x 4로 축소한 소위 4-Queens 문제
- 2가지 경우의 해답을 얻을 수 있음
- 수평, 수직, 대각선 방향으로 점검
- 어떤 퀸도 서로 공격 당하지 않는 위치임을 확인

	Q		Si ey			Q	
			Q	Q			
Q							Q
		Q			Q		

∰ 규칙기반 전문가 시스템

● 지식처리 기술과 지식 획득

- 규칙(rule)은 "If ooo then xxx"란 형식으로 표현
- 지식처리 기술은 규칙이나 프레임에 의한 표현
- 의미 네트워크는 개념 간의 관계를 링크(link)로 연결
- 개와 세퍼드, 개와 진돗개를 링크로 연결
- 의미 네트워크는 지식을 체계화 시켜 표현하기에 적합



● 규칙기반 시스템의 작동 예

- 규칙기반 시스템은 사실(fact)을 나타내는 지식베이스와 If ooo then xxx와 같은 추론규칙으로 이루어짐
- 질의에 대해 지식베이스와 추론규칙을 이용하여 추론한 후 그 결과를 알려줌
- '소크라테스의 죽음'의 간단한 예

● 규칙기반 시스템의 간단한 예

지식베이스: Socrates is a human [소크라테스는 사람이다]
SKKU is a human [SKKU는 사람이다]

Happy is an animal [해피(강아지 이름)는 동물이다]

추론규칙: if X is a human, then X dies [모든 사람은 죽는다] if X is an animal, then X dies [모든 동물은 죽는다]

질의: Is Socrates dies? [소크라테스는 죽는가?]

답변: Yes

질의: Tell me what dies? [죽는 것들은 무엇인가?]

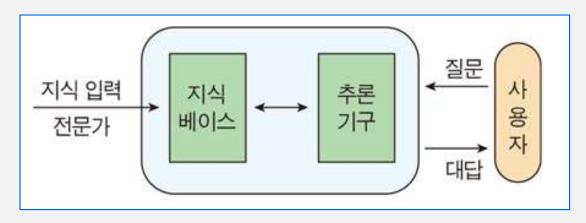
답변: Socrates, SKKU, Happy [소크라테스, SKKU, 해피]

● 규칙기반 시스템의 결과 분석

- 'Is Socrates die?'란 질의에 대한 답변
- 'Socrates is a human'이란 지식베이스의 사실과
 'If X is a human, then X dies'란 추론규칙이 결합하여
 '소크라테스는 죽는다'는 답변을 이끌어냄
- 또 'Tell me what dies?'에 대해서도 'Socrates, SKKU, Happy'가 죽는다는 결론 도출
- 규칙기반 시스템은 Prolog로 편리하게 프로그래밍 가능

● 인간 전문가를 대신하는 전문가 시스템

- 전문가 시스템(Expert System)은 컴퓨터 자문 시스템의 일종
- 특정 분야에서의 인간 전문가의 전문 지식을 활용
- 인공지능의 추론 능력을 이용한 문제 해결 시스템
- 전문가의 지식을 정리하여 지식베이스부터 구축함
- 사용자가 질문하면 추론기구가 지식베이스를 이용하여 추론
- 사용자에게 그 결과를 전해주게 됨



- 전문가 시스템으로 향한 이유
 - 규칙기반 인공지능이 전문가 시스템으로 응용
 - A* 알고리즘 이후 새로운 알고리즘 발견의 어려움



- 전문가 시스템의 적용 분야
 - 전문가 시스템은 규칙기반 인공지능 기법 이용
 - 전문가의 지식을 손쉽게 이용할 수 있는 장점
 - 자동차 전문가 시스템 등 다양한 분야에 적용



● 전문가 시스템의 응용

- 전문가 시스템은 다방면에 걸쳐 응용됨

이 름	기 능	개발 기관
MYCIN	백혈병 진단	스탠퍼드 대학
DENDRAL	질량 분석의 해설	스탠퍼드 대학
PROSPECTOR	광맥 탐사	SRI International
AIRPLAN	항공기 이착륙 관리	U.S. Army
LOGOS	자동 번역	Logos Computer System
ASK	자연어 DB 관리 시스템	Caltech

● 전문가 시스템의 예

- DENDRAL은 스탠퍼드 대학에서 개발한 화학 전문가 시스템
- 분자의 구조를 예측할 수 있도록 개발된 전문가 시스템
- DENDRAL은 휴리스틱을 사용한 후 전문가와 대등한 수준

