

의료용 전문가 시스템에서 추론에 관한 연구¹

김진상² · 신양규³

요약

본 논문에서는 의료용 전문가 시스템에 사용 가능한 의료지식을 수리논리적으로 표현하고 이에 대한 연역 및 진단추론을 행하는 방법을 제시하였다. 문제해결을 위해서는 연역추론을 행하며 원인의 규명을 위해서는 진단추론을 행하지만 일차논리 언어로 표현된 의료지식에서 두 종류의 추론을 병행할 수 있다. 그리고 의료지식에 자주 발생하는 시간에 따라 가변적인 결과의 추론방법도 함께 고찰하였다.

주제어: 지식표현, 진단추론, 시간추론, 의료용 전문가시스템.

1. 서론

인공지능과 정보통신 분야의 이론 및 기술 발전으로 최근에는 개인이 로봇을 가지는 시대로 접어들고 있다. 현재 이러한 개인용 로봇은 대부분 컴퓨터에서 프로그램을 실행시켜 주어진 임무를 완수하는 응용 프로그램 형태이기 때문에 소프트봇(softbot) 혹은 로봇 에이전트(agent)라고 부른다.

한편 인공지능은 로봇 에이전트나 컴퓨터로 하여금 사람이 지능이 있기 때문에 가능한 일들을 대신 수행하도록 하는 소프트웨어에 관해 연구하는 분야라고 할 수 있는데, 지능이 있기 때문에 가능한 일들은 전문적인 조언을 해 주는 것, 자연 언어를 이해하는 것, 상식적인 추론을 행하는 것, 혹은 새로운 지식을 습득하는 것 등을 예로 들 수 있다. 다른 한편, 의학분야에서 인공지능의 활용은 인공지능의 기법을 기반으로 하여 의료기술을 보조해 주는 방향으로 발전 및 전개를 시키고 있는데, 구체적인 응용 영역은 전문가시스템, 지식표현, 지식의 획득과 설명, 의료 영상이나 신호의 분석 및 처리, 시간 및 공간적 추론, 진단이론, 불확실한 지식의 처리 등을 들 수 있다.

의학분야에서 발생하는 지식의 주된 근원은 인간의 해부학이다. 예를 들어, 의료용 전문가시스템은 궁극적으로 의학에서 발생하는 문제들에 대해 전문의 수준의 답을 제공함을

¹ 이 논문은 1998년 한의학연구원 연구비의 일부에 의하여 연구되었음

² 대구시 달서구 신당동 계명대학교 컴퓨터 전자공학부교수

³ 경북 경산시 점촌동 경산대학교 정보과학부교수

목표로 하는 지식기반 시스템이다. 의료용 전문가시스템들이 가지는 지식은 해부학의 범주에서 발생하는 사실과 믿음, 그리고 이론 및 구체적인 휴리스틱으로서, 이러한 지식은 실제 전문의로부터 획득하여 형식화된 언어로 표현하고 있다. 더 구체적인 예로서, 미국 스탠포드 대학에서 1970년대 말부터 개발된 의료용 전문가시스템인 MYCIN[4]은 약 400개 정도의 형식적인 형태의 문장을 지식으로 보유하고 있는데, 각 문장은 “만약...(이)면...이다” 형태의 규칙으로 구성되어있다.

규칙 형태로 지식을 표현할 때 문제해결을 위해서는 연역추론을 사용하는 것이 바람직하지만, 증상의 원인을 규명(진단)하기 위해서는 진단추론을 행하는 것이 적합하다. 추론 방법상의 이러한 차이는 지식의 표현에 직접 영향을 미치게 된다. 즉 연역추론을 이용한 문제해결의 경우에는 임의의 규칙 문장이 “if P then Q ”일 때, P 에 소속된 모든 조건들은 해결 가능한 것들 이어야 한다. 그러나 진단추론을 이용한 원인의 규명을 위해서는 규칙 “if P then Q ”를 자연적인 조건문이 아닌 “ P causes Q ”와 같이 원인 대 결과를 의미하는 인과관계로 해석하는 것이 더욱 타당할 것이다. 따라서 이 때 Q 는 관찰이 가능한 증상이어야 하고 그로부터 증상의 원인이 되는 P 를 진단할 수 있어야 한다.

본 논문에서는 이와 같은 점에 착안하여 규칙을 이용하여 의료지식을 표현하되 문제해결과 증상의 원인 규명을 목적별로 구분하여 나타내고, 이에 따른 연역 및 진단추론을 행하는 방안을 모색한다. 나아가 의료지식의 경우 특히 자주 발생하는 시간에 따른 사실의 변화 과정을 함께 해결할 수 있는 통합된 표현 양식을 함께 제시한다.

2. 규칙을 이용한 지식의 표현과 문제해결 방법

전문가시스템은 규칙 형태로 지식을 표현하는 것이 일반적이데, 그렇지 않은 것과 구별하기 위해서는 규칙기반 전문가시스템이라 부르기도 한다. 여기서 말하는 규칙의 형태는 서론에서 언급한 것처럼 “if P then Q ”의 문법적 구조를 취하는데, 이 때 P 는 통상적으로 “and”로 결합된 더 세부적인 조건을 갖는다. 따라서 일반적인 규칙의 형태는 다음 문장 (1)의 형태를 취한다:

$$\text{if } P_1 \text{ and } P_2 \text{ and } \dots \text{ and } P_n \text{ then } Q \quad (1)$$

이 때 각 $P_i (1 \leq i \leq n)$ 와 Q 는 단순한 사실이어야 한다. 이와 같은 방법으로 의료용 전문가시스템인 MYCIN에서 사용한 규칙의 한가지 예는

$$\begin{aligned} \text{if } [2-1] \text{ the_infection_is_primary_bacteremia and} \\ [2-2] \text{ the_site_of_the_culture_is_one_of_the_sterile_sites and} \\ [2-3] \text{ the_suspected_portal_of_entry_is_gastrointestinal} \\ \text{then } [2-4] \text{ there_is_suggestive_evidence_that_infection_is_bacteriod} \end{aligned} \quad (2)$$

와 같다. 규칙 (2)에서 사용된 세 개의 조건 [2-1], [2-2], [2-3]은 모두 직접 혹은 간접적으로 해결가능한 명제들 이어야 한다. 직접 해결이 가능하다는 것은 이와 동일한 사실이 지식베이스에 포함되어 있든지 아니면 또 다른 규칙의 결론으로 이들 조건이 나타나 있고 그 규칙의 조건들이 해결가능하면 된다. 또한 간접적으로 해결 가능하다는 것은 지식 베이스에는 존재하지 않지만 질문을 통해서 진위 여부를 입력하는 경우를 말한다.

그러나 위의 (2)와 같은 규칙의 표현 방법으로는 일반적인 규칙을 나타낼 수 없다. 그 이유는 규칙 (2)에서 사용된 조건 및 결론이 모두 명제 형태이기 때문이며, 이러한 문제점은 변수를 포함한 상수 및 함수를 사용하는 술어 형식을 이용하여 규칙표현의 제한성을 극복할 수 있다. 다음의 예제 규칙 (3)은 술어 형태로 표현된 간단한 문장이다:

if [3 - 1] *Person* complains_of *Symptom* and
 [3 - 2] *Drug* suppresses *Symptom* and
 [3 - 3] **not** *Drug* is_unsuitable_for *Person* (3)
 then [3 - 4] *Person* should_take *Drug*

규칙(3)에서 사용된 대문자로 시작하는 이탤릭체 기호들은 모두 변수이며 이들은 문장 제일 앞에서 “모든 ...”으로 수식되어 있다⁴. 또한 규칙 (3)의 조건 [3-3]에 나타난 “not”은 부정을 의미한다. 규칙(3)과 더불어 다음 (4)-(10)까지의 규칙 및 사실들이 지식베이스에 있다고 가정한다:

if [4 - 1] *Drug* aggravates *Condition* and
 [4 - 2] *Person* suffers_from *Condition* (4)
 then [4 - 3] *Drug* is_unsuitable_for *Person*

aspirin suppresses pain (5)

lomotil suppresses diarrhea (6)

aspirin aggravates peptic-ulcer (7)

lomotil aggravates impaired_liver_function (8)

john complains_of pain (9)

john suffers_from impaired_liver_function (10)

이때 “John이 어떤 약을 복용해야 하는가”를 알기 위한 문제해결은 연역추론⁵ 시스템에 질문 “john should take *Drug*”을 하여 “*Drug=aspirin*”이라는 결론을 얻을 수 있다. 또

⁴*Person* complains_of *Symptom*은 $\forall Person, Symptom (Person \text{ complains_of } Symptom)$ 을 의미한다

⁵“if P then Q”와 “P”에서 “Q”를 유도하는 추론을 가리킨다.

한 “어떤 약이 설사를 진정시키는지”를 알기 위한 질문 “*Drug suppresses diarrhea?*”을 이용한 연역추론으로 “*Drug=lomotil*”이라는 결과를 얻을 수 있다.

이상과 같이 명제 혹은 술어 형태의 형식언어를 이용하여 지식을 규칙으로 나타낼 때 연역추론은 문제해결에 효과적으로 활용될 수 있다.

3. 규칙에서의 진단추론 방법

앞에서 살펴 본 두 가지 예에서 처럼 문제해결을 위한 연역추론의 규칙의 조건부가 해결 가능한 것들이어야 함은 명확하다. 그러나 만약 규칙의 결론이 관찰의 결과로 알 수 있는 사실들이고 그 규칙의 조건부가 추측해 내어야 하는 경우라면 연역추론의 방법으로는 불가능하다. 예를 들어, 다음과 같은 하나의 규칙과 사실을 가정한다.

if *Person* caught cold then *Person* suffers_from head_ache (11)

john suffers_from head_ache (12)

이때 문장(12)는 관찰에 의해 확인이 가능한 사실이며 이는 진단추론⁶을 이용하여 규칙(11)에 의해 “john이 (아마도) 감기에 걸렸다”라는 가설 “john caught cold”을 얻을 수 있다. 그러나 진단추론은 연역추론과는 달리 항상 참인 가설을 생성하지는 못한다. 또 다른 규칙이 하나 더 추가되었다고 가정한다.

if *Person* had indigestion then *Person* suffers_from head_ache (13)

이 경우 진단추론은 사실 (12)로부터 두 가지 가능성, 즉

(john caught cold) or (john had indigestion)

을 결론으로 제공한다.

진단추론을 이와 같이 관찰된 증상에서 개체가 그러한 증상을 보이게 된 원인을 관련된 인과규칙을 이용하여 추론해 내는 기능을 제공한다. 그러나 유도된 원인이 일반적이거나 포괄적이어서는 효용성이 없다. 예를 들어,

if kid_of(*Person*) caught cold then *Peron* caught cold (14)

을 추가로 가정할 경우, (12)로부터 “john caught cold”가 아닌 “kid of(john) caught cold”를 유도하지 않아야 한다. 다시 말해서 진단추론에 의해 얻어지는 결론은 항상 “극소성(minimality)”을 유지해야 한다.

⁶“if *P* then *Q*”와 “*Q*”에서 “*P*”를 유도하는 추론을 가리킨다.

4. 시간에 따른 사실의 변화 추론 방법

환자의 질병 정도는 시간에 따라 가변적이지만, 앞에서 본 지식표현의 방법은 시간가념을 나타내지 못한다. 예를 들어, John이 감기에 걸린 날짜가 11월 20일이라면 일반적으로 그 전날인 11월 19일에는 감기에 걸리지 않았을 것이라는 가설을 얻을 수 있어야 할 것이다. 또한 통상 감기 환자는 2 - 3일 정도 경과해야 감기에서 낫는다면 John은 11월 21에도 여전히 감기로 고생할 것이라는 사실 역시 추론할 수 있어야 한다. 이러한 상황을 표현하고 추론하기 위해서 다음과 같은 일반적인 규칙을 가정한다:

$$\begin{array}{ll}
 \text{if} & \textit{Event} \text{ happens and} \\
 & \textit{Event} \text{ initiates } \textit{Relation} \\
 \text{then} & \textit{after}(\textit{Event}, \textit{Relation}) \text{ holds}
 \end{array} \tag{15}$$

$$\begin{array}{ll}
 \text{if} & \textit{Event} \text{ happens and} \\
 & \textit{Event} \text{ terminates } \textit{Relation} \\
 \text{then} & \textit{before}(\textit{Event}, \textit{Relation}) \text{ holds}
 \end{array} \tag{16}$$

그리고 John이 감기에 걸린 사건과 그에 따른 부수적인 가설을 다음과 같이 가정한다:

$$\begin{array}{ll}
 [17 - 1] & \text{event1 happens} \\
 [17 - 2] & \text{event1 initiates john caught cold} \\
 [17 - 3] & \text{event1 terminates john has good_voice} \\
 [17 - 4] & \text{time_point_of event1} = 11/20
 \end{array} \tag{17}$$

문장 (17)의 [17-1]과 [17-2]를 규칙 (15)에 적용하여 연역추론을 행하면 “after(event1, john caught cold) holds”를 알 수 있고, [17-1]과 [17-3]을 규칙 (16)에 적용하여 연역추론을 행하면 “after(event1, john has good voice)holds”가 아님을 알 수 있다. 즉, event1이 발생할 11월 20일 바로 다음에 John은 감기에 걸렸고, 그 사건으로 인해 John의 목소리는 상태가 나빠졌다는 결론이다. 나아가 감기가 걸린 상태의 지속성은 다음 규칙의 적용으로 추론할 수 있다:

$$\begin{array}{ll}
 \text{if} & \textit{after}(\textit{Event1}, \textit{Relation}) \text{ holds and} \\
 & \textit{Event2} \text{ happens and} \\
 & \text{time_point_of } \textit{Event1} < \text{time_point_of } \textit{Event2} \text{ and} \\
 & \text{not } (\textit{Event2} \text{ terminates } \textit{Relation}) \\
 \text{then} & \textit{after}(\textit{Event2}, \textit{Relation}) \text{ holds}
 \end{array} \tag{18}$$

따라서 규칙 (18)에 의하면 John이 감기에 걸린 11월 20일 이후에 병원에 가서 치료를 받았다는 등의 또 다른 사건이 발생하지 않았다면 John은 여전히 감기를 앓고 있다는 결론을 유도하게 된다. 물론 이때 규칙 (18)만 존재한다면 John은 영원히 감기로 고생을 할 것이고, 따라서 “일반적인 감기(질병)는 며칠간의 시간이 경과하면 자연 치유가 된다”는 규칙을 추가하면 이 문제를 해결할 수 있다.

5. 결과

이상과 같이 본 논문에서는 의학분야에서 발생하는 다양한 종류의 지식 중에서 특히 논리언어로 표현이 가능한 선언적 지식을 논리언어로 나타내고 이를 이용하여 문제해결과 진단추론이 동시에 가능함을 보였다. 문제해결을 위한 연역추론과 원인 규명을 위한 진단추론은 단순히 결과의 유도와 극소성을 가진 원인의 계산보다는 훨씬 더 복잡한 이론에 근거를 두고있다[5,6,9]. 예를 들어, 4절에서 언급한 것처럼 시간에 따라 변할 수 있는 지식을 처리하기 위해서는 더욱 정교한 추론이 요구된다. 규칙 (18)을 이용하여 “(10일 정도 경과 후) John은 감기가 나았다”라는 결론을 얻기 위해서는 디폴트(default) 추론 기능을 규명해야 하고, 디폴트 추론의 결과로 “john has good.voice”를 다시 유도하기 위해서는 비단조(non-monotonic) 추론 기능을 적용해야 한다. 여기서는 이론적인 내용을 고찰하는 대신 실용성을 입증하기 위해 지식의 표현과 추론을 응용하는 방법을 제시하였다. 또한 시간에 따른 사실의 변화를 추론하는 방법 역시 *Interval Calculus*[3]보다는 프로그램으로의 구현이 용이한 *Event Calculus*[8] 방법을 사용하였다.

6. 고찰

본 논문에서 제외한 내용 중에서 중요한 사안은 불확실한 지식의 표현과 처리방법이다. 불확실한 지식의 표현과 이에 대한 추론은 MYCIN에서처럼 *Certainty-Factor Algebra*를 이용하여 조건이나 결론이 성립할 확률을 지식에다 대응시킴으로서 불확실한 지식을 처리하는 경우가 있다[1,4,7,9]. 또 다른 방법으로는 지식을 구성하는 개념의 모호성을 해결하기 위해 *Fuzzy Logic*을 이용한 퍼지추론과 같은 방법이 있다[2]. 따라서 연역추론과 진단추론이 기본적인 추론 기능인 의료용 전문가시스템에서 불확실한 지식을 통일된 한 가지 형식으로 명확하게 표현하고 처리할 수 있는 기능은 중요한 사항이다.

본문에서 예제로 사용한 문장들을 직접 적용한 프로그램을 그림으로 보이지는 않았으나, 일차논리 언어의 특수한 경우를 프로그래밍 언어화한 PROLOG와 같은 언어를 이용하면 쉽게 구현하고 시험할 수 있다. 즉, 문장 (1)을 PROLOG로 변환한

과 같은 형식을 규칙(rule)이라 하는데, 이것과 (1)의 차이는 가정과 결론을 도치시키고 “if” 대신 “:-”을 사용하며 “and” 대신 “,”로 표시하며, 마지막으로 문장 끝에 마침표를 붙이는 것이다. 따라서, 문장 (11)의 경우

`caught_cold(Person) :- suffers_from(Person, head_ache).`

로 나타낼 수 있다. 또한 사실(fact)이란 조건이 없는 규칙이므로 결론만 표현하는데, 예를 들어 문장 (12)는 “`suffers_from(john, head_ache).`”와 같이 프로그램으로 나타낼 수 있다. 연역추론은 PROLOG 시스템이 제공하기 때문에 별도의 구현이 필요없지만 진단추론은 표현 자체의 변형을 통한 연역추론으로 처리할 수 있다[6].

결론적으로 본 연구의 의의는 인공지능의 대표적인 지식표현 방법인 수리논리시스템을 응용하여 의학분야의 지식을 효과적으로 표현할 수 있음을 간단한 예를 통해 보였으며, 또한 표현된 지식에서 연역, 진단, 그리고 시간추론 등 다양한 추론기능을 동시에 확보할 수 있음을 보였다는 점이다. 여기에다 불확실한 지식을 처리할 수 있는 기능과 처리 결과를 새로운 지식으로 학습할 수 있는 능력을 추가하는 것이 향후 중요한 연구 방향이라고 본다.

참고문헌

1. 신양규 (1995). 확률적 명제 논리 프로그래밍, *Journal of Statistical Theory & Methods*, 6, 13-21.
2. 신양규 (1996). 제약 조건 만족과 불확실한 지식의 처리, *Journal of Statistical Theory & Methods*, 7, 17-27.
3. Allen J. (1983). Maintaining Knowledge about Temporal Intervals, *Communications of ACM*, 26, 832-843.
4. Buchanan B., Shortliffe E. (1984). *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley Publishing Company.
5. Charniak E., Shimony S. (1994). Cost-based Abduction and MAP and Explanation, *Artificial Intelligence*, 66(2), 345-374.
6. Esghe K., Kowalski R. (1988). Abduction through Deduction, Technical report, Department of Computing, Imperial College.
7. Kim J., Jang C. (1994). A Probabilistic Extension of Metalogic for Abduction, *Bull, I.I.S.*, 17, 133-143.

8. Kowalski R. (1992). Database Updates in the Event Calculus, *Journal of Logic Programming*, 12, 121-146.
9. Poole D. (1993). Probabilistic Horn Abduction and Bayesian Networks, *Artificial Intelligence*, 64, 81-129.

A Study on Reasoning for Medical Expert Systems ⁷

Jinsang Kim⁸ · Yangkyu Shin⁹

Abstract

We investigate a logical approach to represent medical knowledge, reason deductively and diagnostically. It is suggested that medical knowledge-bases can be formulated as a set of sentences stated in classical logic where each sentence reflects a doctor's knowledge about the human anatomy or his/her view of patient's symptoms. It is also suggested that a form of temporal reasoning can be captured within the same framework because each sentence can have a different truth value based on time. We apply our logical framework to formalize diagnostic reasoning, where the primary cause of illness is chosen among the set of minimal causation on the basis of abductive hypotheses. Most of our examples are given in the context of medical expert systems.

Key Words and Phrases: Knowledge Representation, Diagnostic Reasoning, Temporal Reasoning, Medical Expert Systems

⁷This work was partly supported by Korea Institute of Oriental Medicine Grant for 1998

⁸School of Computer and Electronics Engineering, Keimyung University, Taegu, Korea

⁹School of Information Science, Kyungsan University, Kyungpook, Korea