

규칙 기반 추론 시스템에서 모순 정보의 검출 기법에 관한 연구

우영운

동의대학교 컴퓨터공학과 조교수
(ywwoo@dongeui.ac.kr)

한수환

동의대학교 멀티미디어공학과 조교수
(swhan@dongeui.ac.kr)

박충식

영동대학교 컴퓨터공학과 부교수
(leciel1007@freechal.com)

이 논문에서는 규칙 기반 추론 시스템에서 추론이 수행되는 과정에서 처리 대상인 입력 정보들에 모순이 존재하는 경우에 모순된 정보를 검출하기 위한 기법을 제안한다. 이 논문에서 제안한 기법은 ATMS의 라벨 표현 방법과 관리 기법을 개선하여 불확실성 값을 갖는 입력 정보를 처리할 수 있도록 하고, 상반된 입력 정보들로 인하여 배타적인 결론들을 동시에 지지하는 경우에 각 결론들과 지지하는 입력 정보들간의 모순을 검출할 수 있는 기법이다.

1. 서 론

일반적으로 규칙 기반 추론 시스템을 실세계의 문제에 적용하고자 하는 경우, 어떠한 이유로 인하여 입력 정보에 모순이 발생한 상태에서 추론을 수행하게 되면 결과를 도출하지 못하거나 결론들의 확신 정도의 차이가 너무 적기 때문에 판단 근거로 사용하기에는 불충분하다는 것을 알 수 있다. 즉, 기존의 전문가 시스템에서 널리 사용되고 있는 규칙 기반 추론 기법은 나름대로의 장점을 갖고 있지만 실세계의 문제에서 발생할 수 있는 모순(contradiction)된 입력 정보들은 처리할 수 있는 방법이 아직까지 없다는 것을 알 수 있었다.

따라서 이 논문에서는 규칙 기반 추론 기법에 의해 추론이 수행되는 과정에서 처리 대상인 입력 정보들에 모순이 존재하는 경우에 해당 업무 분야의 전문 지식을 사용하지 않고도 모순을 검출할 수 있는 새로운 구조 및 처리 기법을 제안하였다.

규칙 기반의 추론 시스템에서 지식 베이스내의 규칙들간의 모순 처리를 위한 기존의 기법으로는 JTMS(Justification-based TMS)[1], ATMS(Assumption-based TMS)[2][3]가 널리 이용되어 왔다. 그러나 이 기법은 지식 베이스내의 논리적인 모순을 해결하기 위해 제안된 것으로서 입력 정보들 간의 모순을 검출할 수는 없는 구조이다. 또한 불확실한 정보들을 수용할 수

있는 표현 방법 및 처리 기법을 제공하고 있지 못하기 때문에 대부분의 입력 정보들이 불확실성 값을 갖는 실세계의 문제에서는 그대로 적용할 수가 없다. 일반적으로 불확실성 값을 갖는 정보들을 처리하기 위한 기법으로는 베이스 규칙(Bayes' Rule)[4], 덤스터 샤퍼 이론(Dempster Shafer Theory)[5], 퍼지 집합 이론(Fuzzy Set Theory)[4] 등의 고전적인 기법들과 랜덤 집합 이론[6]과 인터벌 이론[7]을 함께 적용한 복합 불확실성 값 표현 기법[8] 등이 있다. 그러나 이러한 기법들은 수치적인 불확실성 값을 처리할 수는 있으나 정보들간의 모순을 검출하거나 제거하는 방법을 제공해주지 못하고 있다.

이 논문에서 제안한 입력 정보들간의 모순 검출 기법은 ATMS의 라벨 표현 방법[9]과 관리 기법을 개선하여 입력 정보들이 배타적인(exclusive) 다중 결론들을 모순되게 지지하는 경우, 각 결론들과 지지하는 입력 정보들간의 모순을 검출할 수 있다.

2장에서는 이 논문에서 검출하고자 하는 모순 정보의 정의에 대하여 설명하고 3장에서는 기존 ATMS의 문제점을, 4장에서는 제안한 모순 검출 기법을 설명한 후 5장에서 결론을 맺는다.

2. 모순 정보의 정의

이 논문에서 해결하려는 모순 정보라는 것은, 정보 자체의 불확실성에 관계없이 기존에 습득된 지식 및 정보들을 참조할 경우에 함께 발생할 수 없는 정보들을 일컫는다. 예를 들어 A와 B라는 두 비행체가 있다고 하자. 비행체 A는 꼬리 날개가 삼각형이고 프로펠러에 의해 추진되는 비행체이다. 비행체 B는 꼬리 날개가 사다리꼴이

고 제트 엔진에 의해 추진되는 비행체이다. 즉,

비행체 A :

꼬리 날개(특징 1) = 삼각형

추진 형태(특징 2) = 프로펠러

비행체 B :

꼬리 날개(특징 1) = 사다리꼴

추진 형태(특징 2) = 제트엔진

으로 두 가지 특징을 정리할 수 있다. 우리는 두 비행체가 서로 다른 비행체이고 특징도 위와 같은 것으로 알고 있으며 이 외에 다른 비행체는 존재하지 않는 것으로 알고 있는 상황이다. 만약 이러한 상황에서 하나의 비행체가 출현한 경우 레이더로부터 위의 두 가지 특징이 탐지되었는데 특징 1은 삼각형으로 탐지되고 특징 2는 제트 엔진으로 탐지되었다면 이 경우는 무엇인가 문제가 있는 경우이다. 즉, 레이더가 고장이 났거나, 적의 전자 장비에 의해 교란이 되고 있거나 또는 적이 실제로는 비행체 B인데 비행체 A인 것처럼 위장하기 위하여 꼬리 날개만을 임시로 변형한 것일 수도 있다. 원인이야 무엇이든 기존의 지식과 정보로서는 두 정보가 모순된 정보라는 것을 알 수 있다. 왜냐하면 그러한 두 가지 특징은 동시에 발생할 수 없는 것이기 때문이다.

이러한 문제와는 달리 실세계에서 발생하는 많은 사건(event)들은 자연적으로 실제 발생하는 사건들이며 그 발생 양태를 아직 완전히 파악하지 못하고 있기 때문에 과거에 발생한 통계적 자료를 이용하여 새로운 사건을 수학적인 공식을 이용하여 판단하고 그 결과를 다시 통계적 자료로 누적하게 된다. 또한 실세계에서 발생할 수 있는 사건들은 실제 발생할 것이며 인위적으로

그 사건에 대한 정보를 왜곡시키지 않는다면 모순의 개념은 존재하기 어렵고 따라서 그러한 문제에 대해서는 통계적, 수학적 접근이 최선책이다.

그러나 위의 예와 같이 이미 해당 문제 분야의 내용을 완전히 파악하고 있고 또 해당 문제 분야에서 어떤 특징 정보들을 의도적으로 왜곡하여 제공할 수 있는 경우라면 접근 방법은 달라질 수가 있다. 즉 이러한 경우의 문제는 특정 목적(예를 들어 적을 속이기 위한 목적 : 기만)을 위해 인위적으로 정보가 왜곡되는 특징을 갖기 때문에 그러한 정보의 왜곡은 기존의 지식과 정보로서 파악이 가능할 것으로 생각된다.

특히 이러한 정보의 왜곡은 게임이나 전쟁 등에서 실제 많이 발생한다. 예를 들어 침투시 적의 의도적인 기만이나 고장 및 적의 교란에 의한 센서, 레이더 등 정보 수집 장치들의 오동작으로 인하여 정보의 왜곡이 발생할 수 있다. 따라서 이 논문에서는 실세계의 문제에서 이러한 경우들을 처리할 수 있도록 하기 위하여 정보의 왜곡에 의한 모순 정보를 검출하기 위한 기법을 제안하게 되었다.

3. 기존 ATMS 기법의 문제점

일반적으로 규칙 기반의 전문가 시스템을 구현할 때에는 정상적인 경우에 대한 처리 지식들을 규칙으로 만들게 된다. 즉 전투기가 결론으로 나올 수 있는 규칙들과 헬리콥터가 결론으로 나올 수 있는 규칙들은 개별적인 경우로 생각하여 각각에 대하여 규칙을 구성하게 된다. 이와 같이 구성된 모든 규칙들은 한 지식 베이스 내에 존재할 수 있고 두 경우에 대한 규칙들 간에 관련이

있을 수도 있지만 실제 초기 구성 단계에서는 두 경우를 함께 고려하여 발생할 수 있는 예외적인 상황을 미리 고려하여 규칙을 만들지는 않는다. 이런 식으로 규칙을 구성하게 되면 고려하는 경우들이 많아짐에 따라 발생할 수 있는 예외적인 상황들은 무수히 많아지고 아주 복잡한 상황이 되어 버려, 본래 규칙 기반의 전문가 시스템이 주장하는 개발의 편리함은 사라지며 예외적인 상황을 모두 규칙으로 구성한다는 것은 실질적으로 불가능하기 때문이다. 따라서 모순이 없는 정상적인 상황만을 고려하여 구현된 규칙 기반의 전문가 시스템에서는 입력 정보에 의도적인 기만으로 인한 모순이 발생하게 되면 모순이 없는 정상적인 경우에는 함께 추론될 수 없는 결론들이 함께 추론될 수가 있다.

예를 들어 g 라는 추론 결과는 "X는 전투기이다"라는 결론이고 n 이라는 추론 결과는 "X는 헬리콥터이다"라는 결론이라 하자. 그러면 결론 g 와 n 은 함께 나올 수 없는 결론들이다. g 와 n 이 함께 결론으로 추론된다는 것은 g 를 지지하는 $\{(A), (B)\}$ 라는 입력 정보와 n 을 지지하는 $\{(D)\}$ 라는 입력 정보간에 분명히 문제가 있는 것이다. 즉 A 와 B 가 틀림없는 정보라면 D 가 잘못된 정보일 것이고 D 가 틀림없는 정보라면 A 와 B 가 잘못된 정보일 것이다. 그러므로 이상에서 설명한 기존 ATMS의 처리 과정 외에 g 와 n 의 배타적인(exclusive) 관계를 미리 알아 그 사실을 기록해 놓고 추론된 결론을 감시하는 기능을 부가함으로써 위와 같은 모순 입력 정보로 인한 문제점을 해결할 수 있다. 즉,

$$g : \{(A), (B)\}$$

$$n : \{(D)\}$$

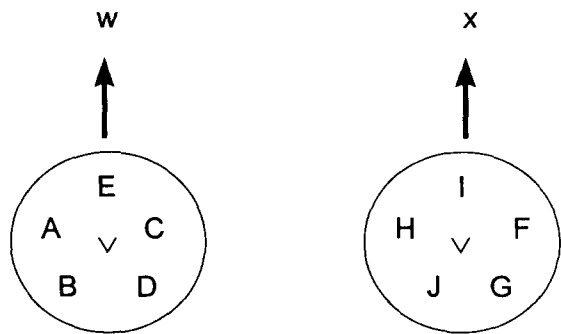
의 형태로 배타적인 두 결론 g와 n을 모두 지지하는 입력 정보가 라벨로서 나타나면 입력 정보에 모순이 발생하였음을 알 수 있고, 또한 {A, B}와 {D}가 서로 모순임을 알 수 있다. 따라서 이상과 같이 ATMS를 응용하면 모든 경우에 모순을 검출할 수가 있을 것으로 생각할 수 있다. 그러나 실제 세계에서 발생할 수 있는 다양한 경우들을 고려하는 과정에서 이 방법만으로는 입력 정보의 모순이 해결될 수 없는 경우가 있음을 알게 되었다. 물론 처리 대상인 정보들의 수치적인 불확실성 값을 수용할 수 없다는 점도 한 이유이지만 그것은 간단한 개선을 통하여 해결될 수 있는 2차적인 이유이고 1차적인 이유는 다음과 같다.

다음의 그림 1과 같이 추론된 결론과 그 결론을 지지하는 입력 정보들과의 관계를 나타내는 두 가지 경우를 생각해 보자. 이 때 추론된 결론 w와 x는 서로 배타적인 결론이라고 가정한다. 그림 1의 (a)는 w와 x가 추론되기 위한 입력 정보에 서로 공통되는 정보가 없음을 알 수 있다. 즉 w를 지지하는 정보들 {A, B, C, D, E}와 x를 지지하는 정보들 {F, G, H, I, J}는 서로 상대적인 정보들이기 때문에 모순이 없는 정상적인 상황에서는 {A, B, C, D, E}나 {F, G, H, I, J} 둘 중의 한 쪽 정보들만이 발생하게 되어 w나 x 중에 하나만이 결론으로 나오게 된다.

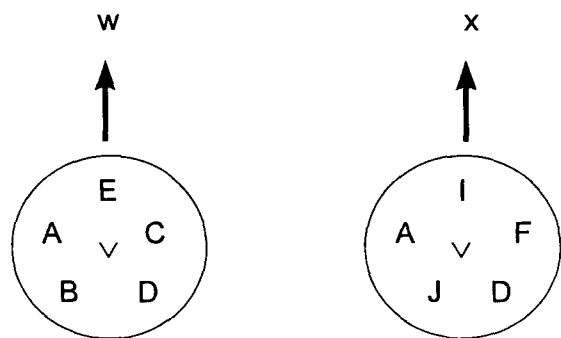
하지만 의도적인 기만으로 인하여 양쪽의 정보들이 함께 입력될 수 있는 모순이 발생한 상황에서는 양립할 수 없는 w와 x의 두 결론에 지지 정보가 모두 나타나게 되며 이로 인하여 모순이 발생하였음을 파악할 수 있다. 그러나 문제는 이렇게 실제 입력 정보에 모순이 발생하여 배타적인 두 결론에 지지하는 정보가 나타나는 것 이외에도 그림 1(b)의 경우처럼 실제 상황에서는 배타적인 두 결론을 지지하는 정보들 중에는 동일

한 정보(A, D)가 있을 수 있다는 것이다.

예를 들어 “비행체의 진행 속도가 900knot 이상이고 1000knot 이하면 그 비행체의 기종은 MIG-25다.”라는 지식과 “비행체의 진행 속도가 950knot 이상이고 1050knot 이하면 그 비행체의 기종은 F-15이다.”라는 지식이 있을 경우에 속도가 990knot로 입력되었다고 하면 판단 대상의 결론인 MIG-25와 F-15라는 두 비행체는 동시에 만족될 수 없는 결론임에도 불구하고 입력된 속도 정보는 지지 정보로서 양쪽에 모두 나타나게 된다.



(a) 동일 입력 정보가 없는 경우



(b) 동일 입력 정보가 있는 경우

<그림 1> 추론 결과와 입력 정보들과의 관계

즉, 입력 정보의 특성상 동일한 입력 정보가 배타적인 두 결론의 지지 정보로서 최종적으로 양쪽에 모두 나타나는 경우가 발생하여 모순이 없는데도 불구하고 모순이 발생한 상황의 (a) 경우와 같은 형태가 되어, 이상에서 언급한 형태만으로 추론된 결론과 지지 정보들의 관계를 관리하게 되면 두 경우를 구별할 수 없는 문제점이 있다. 그림 1의 경우를 예로 들면,

$w : \{(A), (B), (C)\}$

$x : \{(H), (I)\}$

→ 모순이 발생한 경우

$w : \{(A), (B), (C), (D)\}$

$x : \{(A), (D)\}$

→ 모순이 발생하지 않은 경우

의 두 경우를 구별할 수가 없다.

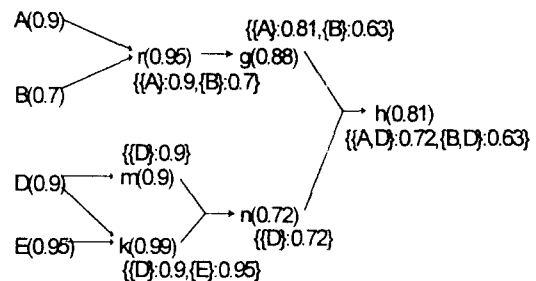
이러한 문제점 때문에 기존의 ATMS 기법에서 사용하는 라벨 관리 기법[10]만으로는 모순 정보의 처리가 불가능하다. 따라서 이 논문에서는 기존 ATMS에서의 라벨 관리 기법보다 한 단계 더 개선된 라벨 관리 기법을 이용하여 추론된 결론과 입력 정보들과의 관계를 더욱 세부적으로 파악할 수 있는 기법을 제안하였다. 제안된 기법에서는 실제 모순이 발생하여 서로 배타적인 결론을 모두 지지하는 정보가 나타나는 경우와 모순은 없지만 동일한 정보가 사용되기 때문에 서로 배타적인 결론을 지지하게 되는 경우를 구별할 수 있어 입력 정보에 존재하는 모순의 검출이 가능하다.

4. 제안한 모순 검출 기법

이상의 문제점을 해결할 수 있는 모순 검출 기법의 설명에 앞서 먼저 수치적인 불확실성 값을 처리할 수 있는 개선된 라벨 표현 방법을 설명한다. 앞서서도 언급하였듯이 기존의 ATMS는 수치적인 불확실성 값을 고려하지 않은 기법이기 때문에 실세계의 문제에서 필수적으로 발생하는 수치적인 불확실성 값을 처리하기 위해서는 입력 정보와 environment들에 불확실성 값을 표현할 수 있어야 한다. 먼저 이를 위한 라벨 표현 방법과 한 environment의 불확실성 값을 계산하는 방법을 4.1에서 설명하고 기존 ATMS에서의 라벨 관리 기법을 개선하여 제안한 새로운 모순 검출 기법의 처리 과정을 4.2에서 설명한다.

4.1 제안한 라벨 표현 방법

이 논문에서는 입력 정보의 불확실성 값과 environment들의 불확실성 값, 그리고 추론된 결론들의 불확실성 값을 그림 2처럼 표현하기로 제안하였다. 이 그림에서는 규칙에도 불확실성 값이 존재하는 경우이다.



<그림 2> 불확실성 값을 갖는 ATMS의 처리 과정

우선 입력 정보들의 불확실성 값은 입력될 때 발생하는 값을 그대로 사용하게 되며 각 environment들의 불확실성 값은 EMYCIN에서 사용되는 확신도(Certainty Factor : CF) 기법[5]처럼 각 입력 정보들의 불확실성 값들 중 최소값에 규칙의 확신도를 곱하는 방법이 사용된다. 그리고 여러 개의 environment들이 모여 한 노드의 라벨을 구성하는 경우는, 여러 증거가 하나의 결론을 주장하는 경우이므로, 주장이 많을수록 그 결론이 더욱 정확한 결론이라 할 수 있으며 주장의 개수가 많을수록 각 주장의 단일 확신도들의 평균보다 더욱 상향 조정된 값이 최종 결론의 확신도가 되어야 한다.

이를 반영할 수 있는 방법들 중의 하나로서 이 논문에서는 확신 비율(certainty ratio)을 이용하여 확신도들을 하나로 합치는 방법[11]을 사용하였다. 즉, 수식 (5-1), (5-2)에서 나타난 것처럼 확신 비율은 r 로 표현되고 확신도는 C 로 표현되며, 각 environment들의 확신도(C_1, C_2, \dots)들을 아래의 수식 (5-1)을 이용하여 확신 비율들로(r_1, r_2, \dots)로 변환한 다음, 변환된 확신 비율을 모두 곱하고 아래의 수식 (5-2)를 이용하여 최종 확신도를 산출하는 방법을 사용하였다. 만약 다음에 설명할 제안 기법에 의해 모순이 밝혀지면 모순된 정보가 제거된 후에 이 방법이 적용되어 최종적으로 추론된 결론의 확신도가 결정된다. 즉, 모순이 해결되기 전까지는 추론된 결론 노드의 확신도는 결정되지 않는다.

$$r = C / (1 - C) \quad (5-1)$$

$$C = r / (1 + r) \quad (5-2)$$

그림 2에서 추론된 결론들의 확신도들은 이 방법을 이용하여 산출된 것이다. 추론이 발생함에

따라 라벨이 형성되는 ATMS의 처리 과정은 라벨 표현에 확신도들이 첨부된 것 이외에는 기존의 ATMS 처리 기법과 동일하다.

4.2 제안한 기법의 처리 과정

앞에서 언급한 문제점을 해결하기 위하여 제안한 모순 검출 기법의 기본 개념은, 항상 추론된 결론 자신에 대한 라벨만을 관리하는 기존 ATMS 방법과는 달리 동일한 정보가 서로 배타적인 결론들을 주장하는 경우 그 결론들을 하나의 집합으로 만들어 이 집합에 대해서도 라벨을 관리하자는 것이다. 이 처리 개념은 뎀스터-사피 이론(Dempster-Shafer Theory)[4]에 나오는 Ignorance 처리 개념을 응용한 것으로서 판단 대상인 배타적인 결론들로 구성되는 임의의 집합들 까지도 라벨을 관리하면 기존 ATMS 기법의 문제점을 해소할 수 있으리라는 것이 기본적인 접근 방법이다. 또 입력 정보들에 불확실성 값이 존재하는 경우에는 동일한 정보가 서로 배타적인 결론을 주장하더라도 의미적으로 긍정적인 주장인지 부정적인 주장인지는 불확실성 값에 따라 달라질 수 있기 때문에 불확실성 값에 따라서 라벨 관리 방법도 달라지게 된다.

제안한 모순 검출 기법 과정의 설명을 위해 우선 불확실성 값이 없는 경우의 처리 기법의 개념을 위의 예를 이용하여 설명한 후, 일반적으로 불확실성 값이 있는 경우의 모순 검출 기법을 설명하고, 모순 검출 기법의 처리 알고리즘을 전개한다.

4.2.1 불확실성 값이 없는 경우의 모순 검출 과정

기존 ATMS 기법으로서는 다음과 같이 표현

되는 그림 1의 두 가지 경우를 구별할 수가 없었다.

$w : \{\{A\}, \{B\}, \{C\}\}$

$x : \{\{H\}, \{I\}\}$

→ 모순이 발생한 경우

$w : \{\{A\}, \{B\}, \{C\}, \{D\}\}$

$x : \{\{A\}, \{D\}\}$

→ 모순이 발생하지 않은 경우

따라서 이 경우를 구별하기 위해서는 서로 배타적인 결론들에 대해 라벨이 완전히 구성된 후, 서로 배타적인 결론에 동일한 입력 정보들(environment)이 있는지를 찾아 그 입력 정보들을 분리하여 그 정보들이 함께 지지하고 있는 결론들을 하나의 집합으로 구성하여 그 집합에 대한 라벨에 그 입력 정보들을 추가한다. 즉, 위의 예는 다음과 같이 바뀐다.

$\{w\} : \{\{A\}, \{B\}, \{C\}\}$

$\{x\} : \{\{H\}, \{I\}\}$

$\{w, x\} : \{\}$

→ $\{w\} \cap \{x\} = \emptyset$: 모순 발생

$\{w\} : \{\{B\}, \{C\}\}$

$\{x\} : \{\}$

$\{w, x\} : \{\{A\}, \{D\}\}$

→ $\{w\} \cap \{w, x\} = \{w\}$

: 모순이 없는 판단 결과

그런 다음 라벨 내에 environment가 존재하는 결론들 및 결론들로 구성된 집합들에 대하여 교집합을 취한다. 그 결과가 공집합이 되면 모순이

발생한 경우이고 공집합이 아닌 어떤 집합이 나오게 되면 그 집합이 최종 판단의 결론이다. 이 처리 개념이 이 논문에서 제안하는 새로운 모순 검출 기법의 기본이 된다.

4.2.2 불확실성 값이 있는 경우의 모순 검출 과정

불확실성 값이 존재하는 경우의 처리 과정도 기본적으로는 위에서 설명한 기법과 동일하지만 environment들을 결론들로 구성되는 임의의 집합으로 보낼 수 있는지 없는지가 각 environment에 첨부된 확신도에 따라 결정된다는 점이 다르다. 이 논문에서는 확신도를 0에서 1사이의 값을 사용하기 때문에, 확신도가 0.5이면 이 정보가 맞는지 아닌지를 잘 모르겠다는 개념을 나타내고 0.5 미만이면 부정적인 지지를 나타내며 0.5 초과이면 긍정적인 지지를 나타낸다.

일반적인 추론 문제에서는 긍정적인 정보들만을 입력받아 추론을 수행하여 결론을 도출하는 경우도 있지만, 입력 정보들을 받아들일 때 확률 개념을 채택하여 한 입력 정보의 확신도가 x (일반적으로 x 는 0.5 이상)라면 그 반대 개념의 정보에는 실제 입력된 것은 아니지만 확신도를 $1-x$ 로 주어 모든 정보들이 존재하는 것처럼 변환하여 추론을 행할 수도 있다. 따라서 동일한 정보가 서로 배타적인 결론의 environment로 존재한다 하더라도 둘 다 긍정적인 지지를 갖는 경우와 둘 다 부정적인 지지를 갖는 경우에만 그 결론들의 집합에 대한 라벨로 옮길 수 있게 된다. 또한 모순을 검출하는 단계에서도 environment가 존재하는 라벨을 갖는 결론들과 결론들의 집합들에 대해 교집합을 취하는 것이 아니라 존재하는 environment들 중에서도 긍정적인 확신도를 갖는 environment들만을 고려하여 처리하여야 한

다. 간단한 예들을 통하여 처리 개념을 살펴보자.

예를 들어 추론 결과, 두 배타적인 결론 k, h에 다음과 같은 형태로 라벨이 결정되었다고 하자.

$$k: \{(I):0.9, \{M\}:0.8, \{O\}:0.95, \{P\}:0.98\}$$

$$h: \{(P):0.97\}$$

이 경우는 모든 environment들이 긍정적인 확신도를 가지고 있는 경우이므로 불확실성 값이 없는 경우와 마찬가지로 처리된다. 단 {h, k}에 첨부되게 되는 확신도의 결정이 문제인데 이 논문에서는 앞에서 설명한 EMYCIN의 CF에서 사용되는 최소값 채택 방법을 이용한다. 그러면 다음과 같이 변화됨을 알 수 있다.

$$\{k\}: \{(I):0.9, \{M\}:0.8, \{O\}:0.95\}$$

$$\{h\}: \{\}$$

$$\{h, k\}: \{(P):0.97\}$$

$$\rightarrow \{k\} \cap \{h, k\} = \{k\}$$

따라서 모순이 없는 정상적인 경우임을 알 수 있다. 또 다른 예로 추론 결과 다음과 같은 형태로 라벨이 결정되었다고 하자.

$$h: \{(B):0.9, \{P\}:0.95\}$$

$$k: \{(M):0.95, \{O\}:0.93, \{P\}:0.91\}$$

이 경우 역시 모든 environment들이 긍정적인 확신도를 가지고 있는 경우이므로 다음과 같이 변화된다.

$$\{h\}: \{(B):0.9\}$$

$$\{k\}: \{(M):0.95, \{O\}:0.93\}$$

$$\{h, k\}: \{(P):0.91\}$$

$$\rightarrow \{k\} \cap \{h\} = \emptyset$$

따라서 모순이 발생한 상황임을 파악할 수 있으며 {B}와 {M, O}가 서로 모순 관계라는 것도 파악할 수 있다. 또 다른 예로 추론 결과 다음과 같은 형태로 라벨이 결정되었다고 하자(실제 입력된 정보는 B, L, N, P이고 I, M, O는 실제 입력되지 않은 반대 개념의 정보들로서 1에서 입력된 정보들의 확신도를 뺀 값이 사용된다).

$$h: \{(B):0.9, \{L\}:0.8, \{N\}:0.95, \{P\}:0.98\}$$

$$k: \{(I):0.3, \{M\}:0.2, \{O\}:0.15, \{P\}:0.98\}$$

우선 동일한 environment인 {P}의 확신도가 모두 긍정적인 경우이므로 {h, k} 집합의 라벨로 옮겨진다.

$$\{h\}: \{(B):0.9, \{L\}:0.8, \{N\}:0.95\}$$

$$\{k\}: \{(I):0.3, \{M\}:0.2, \{O\}:0.15\}$$

$$\{h, k\}: \{(P):0.98\}$$

라벨이 정리된 결과를 보면 긍정적인 environment들과 부정적인 environment들이 함께 존재하는 경우임을 알 수 있다. 즉 {k}의 라벨에 있는 environment들({I}, {M}, {O})은 모두 {k}를 부정적(0.5 미만)으로 지지하는 environment들이고 {h}의 라벨에 있는 environment들({B}, {L}, {N})은 모두 {h}를 긍정적(0.5 이상)으로 지지하는 environment들이다. 따라서 0.5 이하의 environment(부정하거나 잘 모르겠다는 의미를 가짐)들은 고려하지 않고 0.5 이상의 environment들을 하나 이상 갖는 결론들과 그 결론들로 구성되는 모든 집합에 대하여 교집합을 취한 결과가 공집합이 되면 모순이 발생한 경우이고 그렇지 않으면 교집합의 결과가 최종 결론이 된다. 즉 이 예에서는 $\{h\} \cap \{h, k\} =$

{h}가 되므로 모순이 발생하지 않은 경우이고 최종 결론이 {h}임을 알 수 있다.

또 다른 예로 추론 결과 다음과 같은 형태로 라벨이 결정되었다고 하자.

$h:\{B\}:0.9, \{L\}:0.2, \{N\}:0.95, \{P\}:0.18$
 $k:\{I\}:0.3, \{M\}:0.8, \{O\}:0.15, \{P\}:0.20$

이 예의 경우도 바로 위의 예처럼 라벨이 정리된 결과가 긍정적인 environment들과 부정적인 environment들이 함께 존재하는 경우임을 알 수 있다. 우선 동일한 environment인 {P}의 확신도가 모두 부정적인 경우이므로 {h, k} 집합의 라벨로 옮겨진다.

$\{h\}:\{B\}:0.9, \{L\}:0.2, \{N\}:0.95$
 $\{k\}:\{I\}:0.3, \{M\}:0.8, \{O\}:0.15$
 $\{h, k\}:\{P\}:0.18$

마찬가지로 0.5 이상의 environment들을 하나 이상 갖는 결론들과 그 결론들로 구성되는 집합에 대하여 교집합을 취한 결과가 공집합이 됨을 알 수 있고($\{k\} \cap \{h\} = \emptyset$), 0.5 이상의 확신도만을 갖는 {B, N}과 {M}이 서로 모순 관계라는 것도 알 수 있다.

또 다른 예로 추론 결과 다음과 같은 형태로 라벨이 결정되었다고 하자.

$h:\{B\}:0.9, \{L\}:0.2, \{N\}:0.95, \{P\}:0.18$
 $k:\{I\}:0.3, \{M\}:0.8, \{O\}:0.15, \{P\}:0.95$

이 예의 경우에는 {P}가 동일한 environment로서 서로 배타적인 두 결론에 모두 들어있긴 하지만 그 확신도가 하나는 긍정적이고

다른 하나는 부정적이라는 것이 실제 반대 개념의 두 정보가 정상적으로 하나를 긍정적으로 지지하고 다른 하나를 부정적으로 지지하는 경우와 같기 때문에 처리할 필요가 없다. 따라서 최종적으로 관리된 라벨은 다음과 같다.

$\{h\}:\{B\}:0.9, \{L\}:0.2, \{N\}:0.95, \{P\}:0.18$
 $\{k\}:\{I\}:0.3, \{M\}:0.8, \{O\}:0.15, \{P\}:0.95$
 $\{h, k\}:\{\}$

마찬가지로 0.5 이상의 environment들을 하나 이상 갖는 결론들과 그 결론들로 구성되는 집합에 대하여 교집합을 취한 결과가 공집합이 됨을 알 수 있고($\{k\} \cap \{h\} = \emptyset$), 0.5 이상의 확신도만을 갖는 {B, N}과 {M, P}가 서로 모순 관계라는 것도 알 수 있다.

이상의 예들만 보아서는 서로 배타적인 결론들로 구성되는 임의의 집합을 반드시 이용하지 않고 environment들을 단순히 제거하는 것만으로도 모순을 검출할 수 있을 것으로 생각할 수 있으나 이것은 간단한 예를 들었기 때문이며 실제로는 제안한 기법으로 처리해야만 가능하다. 다음의 예를 살펴보자(x, y, z는 서로 배타적인 결론).

$x:\{A\}:0.9, \{B\}:0.8$
 $y:\{C\}:0.7$
 $z:\{C\}:0.7$

이상과 같은 형태로 라벨이 결정된 경우 '{C} : 0.7'을 제거하기만 하면 모순 없이 x가 결론으로 나오게 된다. 그러나 그것은 잘못된 판단이며 실제로는 다음과 같이 모순이 발생한 경우이며, {A, B}와 {C}가 서로 모순이라는 것을 알 수 있다 ($\{x\} \cap \{y, z\} = \emptyset$).

$$\{x\}:\{(A):0.9, (B):0.8\}$$

$$\{y, z\}:\{(C):0.7\}$$

이상의 예들에서 알 수 있듯이 제안한 모순 검출 기법은 서로 배타적인 결론들로 구성되는 집합을 이용하였기 때문에 가능하였다. 이상에서 언급한 모든 경우의 예들을 처리할 수 있는 제안한 모순 검출 기법의 일반적인 처리 알고리즘을 다음 4.2.3에서 정리하였다.

4.2.3 모순 검출 알고리즘

제안한 모순 검출 알고리즘은 실제 입력된 정보들만을 사용하는 경우와 확률 개념을 이용하여 모든 특징 정보들을 사용하는 경우의 두 가지 모두에 대하여 적용이 가능하다. 즉 이상의 예들을 모두 포함하는 일반화된 모순 검출 알고리즘은 아래와 같다.

판단 대상인 배타적 N개의 결론(hypothesis로 명명)들을 $h_1, h_2, h_3, \dots, h_N$ 라고 표현하자. 즉,

$$\{ h_1, h_2, h_3, \dots, h_N \}$$

: *exhaustive and exclusive hypotheses set*

(5-3)

추론 결과, 각 hypothesis에는 입력 정보들의 집합인 environment들로 구성된 라벨들이 형성된다. 각 hypothesis에 대한 라벨들의 일반적인 표현은 다음과 같다.

$$h_1 = \{ \{ A^1_{11}, A^1_{12}, \dots, A^1_{1K_1} \}: CF_{E^1_1},$$

$$\{ A^1_{21}, A^1_{22}, \dots, A^1_{2K_2} \}: CF_{E^1_2},$$

$$\dots \{ A^1_{M_1 1}, \dots, A^1_{M_1 K_{M_1}} \}: CF_{E^1_{M_1}} \}$$

$$h_2 = \{ \{ A^2_{11}, A^2_{12}, \dots, A^2_{1K_1} \}: CF_{E^2_1},$$

$$\{ A^2_{21}, A^2_{22}, \dots, A^2_{2K_2} \}: CF_{E^2_2},$$

$$\dots \{ A^2_{M_2 1}, \dots, A^2_{M_2 K_{M_2}} \}: CF_{E^2_{M_2}} \}$$

$$h_3 = \{ \{ A^3_{11}, A^3_{12}, \dots, A^3_{1K_1} \}: CF_{E^3_1},$$

$$\{ A^3_{21}, A^3_{22}, \dots, A^3_{2K_2} \}: CF_{E^3_2},$$

$$\dots \{ A^3_{M_3 1}, \dots, A^3_{M_3 K_{M_3}} \}: CF_{E^3_{M_3}} \}$$

$$\vdots$$

$$h_N = \{ \{ A^N_{11}, A^N_{12}, \dots, A^N_{1K_1} \}: CF_{E^N_1},$$

$$\{ A^N_{21}, A^N_{22}, \dots, A^N_{2K_2} \}: CF_{E^N_2},$$

$$\dots \{ A^N_{M_N 1}, \dots, A^N_{M_N K_{M_N}} \}: CF_{E^N_{M_N}} \}$$

(5-4)

여기서 각 입력 정보의 의미는 다음과 같다.

$$A^N_{M, K} \quad (5-5)$$

(N : Number of total hypotheses

M_i : Number of environments
in an i -th hypothesis's label ($i = 1 \sim N$)

K_j : Number of assumptions
in a j -th environment ($j = 1 \sim M_i$)

각 입력 정보가 모여 구성되는 environment는 다음과 같이 표기하고,

$$E^N_{M_i} \quad (5-6)$$

(N : Number of total hypotheses

M_i : Number of environments
in an i -th hypothesis's label ($i = 1 \sim N$)

각 hypothesis의 라벨은 다음과 같이 표기한다.

$$L^N \quad (N : \text{Number of total hypotheses}) \quad (5-7)$$

실제 상황에서 결론이 하나도 나오지 않는 경우는 규칙들의 조건부가 너무 엄격하여 하나의 모순만이라도 존재하는 경우에 그 모순으로 인해 원하는 결론에 대한 최종 규칙이 하나도 수행되지 않아 발생하는 경우이다. 모든 결론에 대해 라벨이 존재하는 경우는 개별적인 입력들에 대해 각각의 결론을 내 줄 수 있도록 규칙들이 잘게 나뉘어져서 구성되어 있고 많은 모순이 입력된 경우이다.

먼저 단일 hypothesis들의 라벨에 포함된 environment의 입력 정보들과 그 environment의 불확실성 값을 보고 hypothesis들의 임의의 집합에 대한 라벨을 관리하는 기법은 다음과 같다.

$$n=1 \sim N, m=1 \sim N, \\ i=1 \sim M_n, j=1 \sim M_m, n \neq m$$

인 모든 E^n_i, E^m_j 에 대하여,

$$[(E^n_i \subset E^m_j) \vee (E^n_i \supset E^m_j)] \\ \text{and } [(CF_{E^n_i} \leq 0.5 \text{ and } CF_{E^m_j} \leq 0.5) \\ \text{or } (CF_{E^n_i} > 0.5 \text{ and } CF_{E^m_j} > 0.5)] \quad (5-8)$$

의 조건을 만족하는 경우에 다음과 같이 해당되는 집합의 라벨에 만족된 environment를 추가한다.

$$H_k = \{h_n, h_m\} : L^k \cup \{E^n_i : \text{Min}(CF_{E^n_i}, CF_{E^m_j})\} \\ \text{if } E^n_i \supset E^m_j; \\ H_k = \{h_n, h_m\} : L^k \cup \{E^m_j : \text{Min}(CF_{E^n_i}, CF_{E^m_j})\} \\ \text{if } E^n_i \subset E^m_j; \\ (H_k \text{는 } \{h_1, h_2, \dots, h_N\} \text{의 부분 집합을 나타냄})$$

그런 후, H_k 에 추가된 environment를 다음 식에 의해 단일 hypothesis의 라벨에서 제거한다.

$$h_n = h_n - E^n_i : CF_{E^n_i}, \\ h_m = h_m - E^m_j : CF_{E^m_j}, \quad (5-9)$$

이상의 과정을 거치게 되면 단일 결론들만에 대한 라벨이 그 결론들로 구성되는 임의의 집합까지 포함한 것에 대한 라벨로 정리된다. 다음에 수행해야 할 단계는 새로이 정리된 라벨을 보고 모순을 검출하는 단계이다. 그 과정은 다음과 같다.

$$n=1 \sim 2^N, m=1 \sim 2^N \\ (n \neq m, i=1 \sim M_n, j=1 \sim M_m) \text{에 대하여}$$

$$(\exists E^n_i)(CF_{E^n_i} > 0.5) \text{ 인 } H_n, \\ (\exists E^m_j)(CF_{E^m_j} > 0.5) \text{ 인 } H_m$$

이 존재할 경우 아래 식들에 의해 모순의 존재 여부를 판단할 수 있다. 먼저 모순이 없는 정상적인 경우는 다음과 같은 식에 의해 판단된다.

$$(\forall n, \forall m)(H_n \cap H_m) = H_r \quad (5-10) \\ : \text{모순이 없는 경우의 최종 결과}$$

이 경우 H_r 은 단일 결론일 수도 있고 단일 결론들로 구성되는 임의의 집합일 수도 있다. 단일 결론으로 나오는 경우는 입력 정보가 모순이 없고 충분하여 정상적으로 하나의 결론이 도출된 경우이고, 집합으로 나오는 경우는 모순은 없지만 정보의 불충분으로 인하여 더 이상의 세부적

인 판단을 내릴 수 없는 경우이다.

예를 들어 판단 대상이 $\{x, y, z\}$ 라는 세 개의 배타적인 결론들이라고 가정하면, H_r 이 $\{x\}$ 인 경우는 x 가 단일 결론으로 최종 판단된 경우이고 H_r 이 $\{y, z\}$ 라면 현재 정보들 만으로서는 최종 결론이 y 인지 z 인지는 판단할 수 없지만 분명히 x 는 아니라는 경우를 의미한다. 입력 정보들에 모순이 발생한 경우는 다음과 같은 식에 의해 판단된다.

$$(\forall n, \forall m)(H_n \cap H_m) = \emptyset \quad (5-11)$$

: 모순(contradiction) 발생

단순히 모순을 검출하는 것 이외에도 어떤 정보들이 서로 모순된 관계인지를 파악할 수가 있는데 그것은 다음과 같은 과정에 의해 이루어진다.

이상에서 파악된 $(H_n \cap H_m) = \emptyset$ 인 모든 $n, m(n=1 \sim 2^N, m=1 \sim 2^N)$ 에 대하여

$(CF_{E^n} > 0.5)(i=1 \sim M_n)$ 인 E^n_i 들의 집합을

$S_{E^n} = \{E^n_1, E^n_2, \dots\}$ 이라 하고

$(CF_{E^m} > 0.5)(j=1 \sim M_m)$ 인 E^m_j 들의 집합을

$S_{E^m} = \{E^m_1, E^m_2, \dots\}$ 이라 하자. 그러면

S_{E^n} 과 S_{E^m} 의 두 집합은 서로 모순 관계임을 알 수 있다. 즉,

$$S_{E^n} \rightarrow \text{모순(contradiction)} \leftarrow S_{E^m} \quad (5-12)$$

5. 결 론

이 논문에서는 입력 정보들을 이용하여 추론 결과를 얻기 전에, 판단 대상인 결론들과 그것들을 지지하는 정보들과의 관계를 명확하게 밝혀내어 입력 정보들 간에 모순이 있는지의 여부를 판단할 수 있도록 하는 기존 ATMS 기능을 개선한 모순 검출 기법을 제안하였다. 이 논문에서 제안한 모순 검출 기법은 ATMS의 라벨 표현 방법과 관리 기법을 개선하여 입력 정보들이 배타적인 여러 결론들을 함께 지지하는 경우, 각 결론들과 지지하는 입력 정보들간의 모순을 검출할 수 있으며 불확실성 값을 갖는 정보들도 ATMS에서 처리가 가능한 기법이다.

이 논문의 목적이 실용적인 규칙 기반 추론 시스템에서 사용하기 위한 처리 구조와 알고리즘을 제안한 것이기 때문에 실제로 실용적인지를 증명하기 위해서는 실세계의 문제에 대한 추론 시스템을 운용하면서, 제안한 모순 검출 기법을 이용하여 도출되는 결과가 모순된 상황을 바르게 해결한 것인지를 분석할 필요가 있다. 또한 모순을 검출하는 것에서 끝나지 말고 모순을 제거하기 위한 방법에 대한 구조와 처리 기법에 대한 연구가 필요할 것으로 생각한다.

참고문헌

- [1] Mladen Stanojevic, Sanja Vranes, et. al., "Using Truth Maintenance Systems," in IEEE EXPERT, pp. 46-56, 1994.
- [2] Cao Wen, *An Assumption-based Truth Maintenance System*, 1993.

- [3] J. de Kleer, "An Assumption-Based Truth Maintenance System," in *Artificial Intelligence*, Vol. 28, No. 2, pp. 127-162, Mar. 1986.
- [4] Keung-Chi Ng and B. Abramson, "Uncertainty Management in Expert Systems," in *IEEE EXPERT*, Apr., pp. 29-48, 1990.
- [5] Grzymala-Busse, Jerzy W., *Managing Uncertainty in Expert Systems*, Kluwer Academic Publishers, 1991.
- [6] Joslyn, Cliff, "Aggregation and Completion of Random Sets with Distributional Fuzzy Measures," *International Journal of Uncertainty, Fuzziness, and Knowledge-Based Systems*, Vol. 4, No.4, pp. 307-329, 1996.
- [7] Joslyn, Cliff, "Measurement of Possibilistic Histograms from Interval Data," *International Journal of General Systems*, Vol. 26, pp. 2-3, 1997.
- [8] Luis Rocha and Joslyn, Cliff, "Towards a Formal Taxonomy of Hybrid Uncertainty Representations," *Information Sciences*, 1997.
- [9] K.D.Forbus and J. de Kleer, "Focusing the ATMS," *Proc. of the Seventh National Conference on AI, AAAI*, Menlo Park, Calif., pp. 193-198, 1988.
- [10] Kenneth D. Forbus and Johan De Kleer, *Building Problem Solvers*, The MIT Press Massachusetts Institute of Technology Cambridge, Massachusetts 02142, pp. 151-170, 1993.
- [11] 김재희, *인공지능의 기법과 활용*, 교학사, 서울, 1988.

Abstract

A Detection Method of Contradictory Informations in a Rule-based Inference System

YoungWoon, Woo*
SooWhan, Han*
Choong Shik, Park**

In this paper, a detection method of contradiction between input informations is proposed when the inference is processed in rule-based systems. The proposed method is accomplished by improving the label representation and the label management scheme in a conventional ATMS(Assumption-based Truth Maintenance System). The Proposed method also can represent and process input informations having uncertainty values.

Key words : Rule-based, Contradiction, ATMS, Uncertainty

* Division of Computer Application Eng., Dongeui University

** The faculty of Information and Electronic Eng., Youngdong University