

이 접적 퍼지 정보를 지원하는 퍼지 객체 추론 모델의 정형화

# (A Formal Specification of Fuzzy Object Inference Model for Supporting Disjunctive Fuzzy Information)

요약

본 논문에서는 이접적 퍼지 정보를 지원하는 퍼지 객체 추론 모델을 정형화하고, 이접적 퍼지 정보를 지원하는 지식기반 프로그래밍을 위한 구현으로서 ICOT(Integrated C-Object Tool)을 제안한다. 제안된 객체 추론 모델은 객체 추론과 퍼지 추론이 객체-관계형 데이터베이스와 호환성있는 일관된 틀로 병합되어 있으며, 객체 지향 패러다임의 대부분이 관계형 구조로 표현되기 때문에, 의미적으로 이해하기 쉽고 개념적으로 사용하기 단순한 퍼지 추론을 지원한다. 또한 이접적 퍼지 정보를 지원함으로써 데이터의 의미적 표현력을 강화시킨다.

### **Abstract**

In this paper, we provide the formal specification of a fuzzy object inference language and propose ICOT(Integrated C-Object Tool) as its implementation for knowledge-based programming with the disjunctive fuzzy information. The novelty of our model is that it seamlessly combines object inference and fuzzy reasoning into a unified framework without compromising a compatibility with extant databases, especially object-relational ones. In this model most of the object-oriented paradigm is successfully expressed in terms of relational constructs, tailoring fuzzy reasoning style to be well suited to the framework of the databases. It turns out to be useful in preserving its conceptual simplicity as well, since simple-to-use is one of important criteria in designing the databases. Additionally this model considerably enhanced the semantic expressiveness of data allowing

disjunctive fuzzy information.

## 1. 서론

퍼지 규칙 기반 전문가 시스템 언어들의 퍼지 언어 용어(fuzzy linguistic term)를 처리하는 능력은 의료 진단, 정보 검색, 자연어 처리 등의 분야에서 다양한 불확실성을 모델링하는데 자연스런 해석을 제공해 왔다. 그러나, 최근 들어 전문가 시스템 언어들이 복잡한 공학적 응용 영역에까지 확장 적용되면서, 이들이 복합 객체들간의 추론 기능은 지원하지 못한다는 점이 점차 큰 단점으로 부각되었다[4][5]. 한편 전문가 시스템에서 다루는 데이터의 양이 점차 방대해지고, 데이터베이스 시스템에 연역 기능을 추가하는 작업의 중요성이 증대됨에 따라, 전문가 시스템 언어와 데이터베이스를 통합하기 위한 많은 연구가 이루어졌다 [6][7][8][9]. 그러나 이러한 통합은 인공 지능과 데이터베이스 영역에서 사용하는 데이터 모델이 원천적으로 서로 다른데서 오는 불일치(impedance mismatch) 문제를 야기 시킨다. 이 문제를 해결하기 위한 한 방법으로, 데이터베이스 언어에 객체 추론 기능을 추가하는 많은 연구들이 진행되어 왔다. K[6]와 Coral++[8], Adam[9] 등은 데이터베이스 위에 불일치 없이 객체 추론 기능을 구현한 언어들이다. 그러나 이 객체 추론은 지식베이스가 불확실성을 포함하고 있을 때는 적용할 수 없는 한계를 가지고 있다.

불확실성(uncertainty)은 크게, 한 개념이 포함되는 범주의 경계가 불확실한 경우인 모호성(vagueness)과 포함될 수 있는 범주가 두개 이상 있을 경우인 애매성(ambiguity)으로 나눌 수 있다. 예를 들어, 홍길동의 전공이 컴퓨터이거나 전자공학이라는 이접적 정보가 주어졌을 때 홍길동의 전공이 VLSI이라고 할 수 있는지를 판단하고자 한다면 여기서 두 종류의 불확실성이 발생한다. 즉, VLSI가 컴퓨터 전공인지 아닌지를 결정해야하는 모호성과, VLSI가 컴퓨터 전공과 전자공학 전공 어느쪽에 속하는지를 결정해야하는 애매성이 발생한다. 따라서 이러한 이접적 퍼지 정보의 처리는 모호성과 애매성을 동시에 처리함으로써 객체 추론 모델의 표현력을 크게 증가 시킬 수 있다.

본 논문에서는 이접적 퍼지 정보를 지원하는 퍼지 객체 추론 모델을 정형화하고, 이접적 퍼지 정보를 지원하는 지식기반 프로그래밍을 위한 구현으로서 ICOT(Integrated C-Object Tool)을 제안한다. 제안된 객체 추론 모델은 객체 추론과 퍼지 추론이 객체-관계형 데이터베이스와 호환성 있는 일관된 틀로 병합되어 있으며, 객체 지향 패러다임의 대부분이 관계형 구조로 표현되기 때문에, 의미적으로 이해하기 쉽고 개념적으로 사용하기 단순한 퍼지 추론을 지원한다. 또한 이접적 퍼지 정보를 지원함으로써 데이터의 의미적 표현력을 강화 시킨다.

## 2. 이접적 퍼지 정보를 지원하는 퍼지 객체 추론 모델

### 2.1 이접적 퍼지 정보를 지원하는 퍼지 객체 클래스의 표현

ICOT에서의 객체 구조는 관계형 구조와 불일치를 최소화하는 방향으로 설계되었다. 즉, 퍼지 정보를 포함하는 객체는 추상 데이터 형과 함께 관계형 테이블로 표현되고, 객체 지향 패러다임 또한 관계형 구조로 표현된다. 먼저, 기본이 되는 퍼지 언어 용어를 정의하자.

**정의 1** 퍼지 언어 용어 (또는 단순히, 용어)는 퍼지 집합 F 또는 일반 대집합(classical universe of dis-

course) D내의 일반 용어 중 하나이다. 용어가 퍼지 집합임을 명확히 하고자 할 때에는 이를 퍼지 용어로 부르기로 한다.

용어는 다음에 나오는 퍼지 언어 용어 술어와 함께 이접적 퍼지 정보를 표현하는 가장 기본적이 단위이다.

**정의 2** 용어 T의 소속성 함수가 다음과 같이 주어졌다고 하자.

$$T : U \rightarrow [0, 1], \text{ 여기서 } U = D \cup \{F\}.$$

용어  $T \in U$ 에 대응되는 퍼지 언어술어  $T(A)$  (또는 단순히 퍼지 술어)는 다음과 같이 정의된다.

$$\begin{aligned} T(c/A) &= 1, & c &= T \text{ 일 때}, \\ T(c/A) &= \mu_T(c/A), & c &\neq T \in U \text{ 일 때}. \end{aligned}$$

이 정의는,  $T(c/A)$ 의 의미를, 어떤  $c \in U$ 에 대해  $c$ 가  $T$ 에 속하는 정도라기 보다는  $c$ 가  $T(A)$ 를 참으로 하는 정도로 해석하고 있다.

**정의 3** 이름이  $N$ 인 퍼지 클래스(또는 단순히 클래스)는 다음과 같이 정의된다.

$$C_N = \langle E_N, sch(C_N), T_N \rangle$$

- 여기서
- 1)  $E_N$ 은 퍼지 객체들의 집합,
  - 2)  $sch(C_N)$ 은 애트리뷰트들의 집합,
  - 3)  $T_N$ 은 형(type)들의 집합.

$A_i \in sch(C_N), i=1, \dots, n$  일 때  $\tau(A_i)$ 를 애트리뷰트  $A_i$ 의 형이라하면  $\tau(A_i) \in T_N$ 은 퍼지 집합이나 퍼지 집합의 집합으로 정의된다. 퍼지 클래스가 관계-객체형 테이블과 대응됨을 강

조하기 위해, 퍼지 객체는 퍼지 투플(또는 단순히 투플)과 구별없이 부르기로 한다.

다음 정의들은 한 퍼지 용어에 직접 또는 간접적으로 구성원이 되는 용어들의 소속성 정도를 계산하기 위한 것이다.

**정의 4**  $F$  를 퍼지 용어라고 하자. 그러면

$$c \in U \text{ 에 대해 } \mu_F(c) = \max(\min(\mu_{a_i}(c), \alpha))$$

여기서  $a_i \in F$  이고,  $\mu_F(a_i) = 1$  이다

**예 1**  $fuzzy \cap software$ 에 속하는 소속성 정도는 다음과 같이 얻을 수 있다.

$$\begin{aligned} \mu_{software}(fuzzy) &= \max(\min(\mu_{ai}(fuzzy), \mu_{software(ai)}), \min(\mu_{db}(fuzzy), \mu_{software(db)})) \\ &= \max(0.8, 0.7) = 0.8. \end{aligned}$$

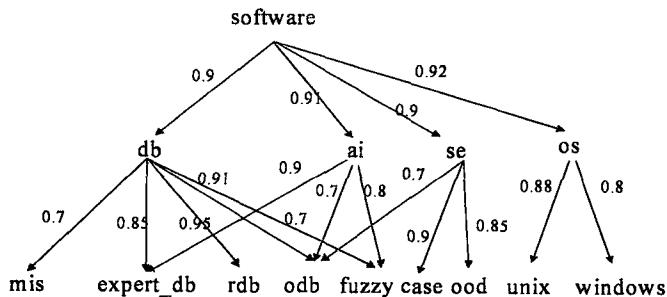


그림 1. MAJOR의 퍼지 소속성 값을 명시하는 그래프

그림 1은 MAJOR에 속하는 값들의 퍼지 소속성을 명시하는 그래프이다. 하위 단계의 퍼지 용어들은 대응되는 에지에 명시된 소속성 정도로 이 퍼지 용어들 각각에 소속되게 된다. 명시되지 않은 용어들간의 소속성 정도는 0으로 가정한다.

**정의 5**  $E_N$ 은  $\tau(A_1) \times \tau(A_2) \times \dots \times \tau(A_n)$ 의 퍼지 부분 집합이고,  $\tau(A_1) \times \tau(A_2) \times \dots \times \tau(A_n)$ 은  $U_1 \times U_2 \times \dots \times U_n$ 의 퍼지 부분집합이다. 여기서  $\times$ 은 카르티전 곱을 나타낸다.

**정의 6** 투플  $t = \langle a_1, a_2, \dots, a_n \rangle$ 라 하면,  $t \in E_N$ 의 정도는  $\mu_{E_N}(t)$  또는 간단히  $\mu(t)$ 이며, 다음과 같은 제약을 만족한다.

$$\mu_{E_N}(t) \leq \min(a_1/\tau(A_1), a_2/\tau(A_2), \dots, a_n/\tau(A_n)).$$

$\mu_{E_N}(t)$ 이 정의 6의 제약을 만족해야 하는 이유는  $E_N \models \tau(A_1) \times \tau(A_2) \times \dots \times \tau(A_n)$ 의 퍼지 부분 집합이기 때문이다. 즉  $t = \langle a_1, a_2, \dots, a_n \rangle$  이  $E_N$ 에 속하기 위해서는 먼저  $t$  가  $\tau(A_1) \times \tau(A_2) \times \dots \times \tau(A_n)$ 에 속해야 하고, 이 속하는 정도는 퍼지 표준 연산자를 사용할 경우  $\min(a_1/\tau(A_1), a_2/\tau(A_2), \dots, a_n/\tau(A_n))$ 으로 정량화될 수 있다.

퍼지 투플과 퍼지 클래스의 한 예를 들어 보면 다음과 같다.

예 2 Engineer 와 laboratory 클래스는 다음과 같이 명시된다.

engineer:

T_ID	NAME	EXPERIENCE	SAL	MAJOR	LAB	$\mu$
e <sub>1</sub>	yang	15	45,000	fuzzy ∨ expert_db	l <sub>2</sub>	0.82
e <sub>2</sub>	kim	10	low	mis	l <sub>1</sub>	0.92
e <sub>3</sub>	park	moderate	50,000	se	l <sub>1</sub>	0.60
e <sub>4</sub>	han	7	high	expert_db	l <sub>3</sub>	0.65

laboratory:

T_ID	NAME	AREA	$\mu$
l <sub>1</sub>	sys	ood	0.89
l <sub>2</sub>	sw	expert_db ∨ odb	0.9
l <sub>3</sub>	os	motif	0.87
l <sub>4</sub>	nw	os	0.91

$$\tau(\text{MAJOR}) = \{0.75/\text{software}, 0.83/\text{db}, 0.92/\text{ai}, 0.91/\text{os}\}$$

$$\text{software} = \{0.9/\text{db}, 0.91/\text{ai}, 0.9/\text{se}, 0.92/\text{os}\},$$

$$\text{db} = \{0.91/\text{odb}, 0.95/\text{fdb}, 0.85/\text{expert_db}, 0.7/\text{mis}, 0.7/\text{fuzzy}\},$$

$$\text{ai} = \{0.8/\text{fuzzy}, 0.9/\text{expert_db}, 0.7/\text{fdb}\},$$

$$\text{os} = \{0.88/\text{unix}, 0.8/\text{windows}\},$$

$$\tau(\text{AREA}) = \tau(\text{MAJOR})$$

도메인에 따른 퍼지 집합의 소속성 함수는 다음과 같다.

$$x \in U_{\text{EXPERIENCE}}, y \in U_{\text{SALARY}},$$

$$\begin{aligned} \mu_{\text{high}}(x) &= (1 + |20-x| / 13)^{-1} && \text{for } x < 20 \\ &= 1 && \text{for } x \geq 20 \\ \mu_{\text{moderate}}(x) &= (1 + |x - 8| / 10)^{-1} && \text{for } x > 1 \\ &= 0 && \text{otherwise,} \end{aligned}$$

$$\begin{aligned}\mu_{\text{high}}(y) &= (1 + 1/120000 * |80000-y|) - 1 && \text{for } y \leq 80000, \\ &= 1 && \text{for } y > 80000 \\ \mu_{\text{moderate}}(y) &= (1 + 1/100000 * |y-50000|) - 1\end{aligned}$$

위의 예에서 engineer 클래스와 laboratory 클래스 내의 major와 area는 이접적 퍼지 정보를 포함하고 있다. 이접적 정보는 둘 중의 한 정보만 참인 배타적 경우와 동시에 다 참이 될 수 있는 포함적인 경우가 있다. 본 모델은 배타적 이접 정보만을 다룬다. 예를 들어, fuzzy  $\vee$  expert\_db 인 경우 fuzzy 또는 expert\_db가 동시에 참인 경우는 배제한다.

다음은 이접적 퍼지 정보를 처리하기 위한 정의들이다.

**정의 7** 이접식 집합, WFDF는 다음과 같이 정의된다.

1. 퍼지 언어 술어는 WFDF에 속한다..
2. 퍼지 언어 술어의 이접은 WFDF에 속한다.
3. 그 외에는 어떤 것도 WFDF에 속할 수 없다.

**정의 8** wfdf가 WFDF에 속한다고 하면,  $wfdf(A) = \bigvee_{i=1}^s a_i$ 에 대해

$$\begin{aligned}wfdf(c/A) &= \max(c(a_i/A), i=1, \dots, s), a_i \text{가 정량값이고 } c \text{가 퍼지용어일 때} \\ wfdf(c/A) &= \max(a_i(c/A), i=1, \dots, s), \text{ 그외.}\end{aligned}$$

**정의 9** wfdf가 WFDF에 속한다면,  $|wfdf| = \{c \mid c \in U \text{에 대해 } wfdf(c/A) = 1\}$ .

**정의 10**  $wfdf_1, wfdf_2$  WFDF자 하자. 그러면  $wfdf_1$ 은  $\alpha$  정도로  $wfdf_2$  보다 일반적이다.

$$\Leftrightarrow \min(c \in |wfdf_2| \text{에 대해 } wfdf_1(c/A)) = \alpha > 0. \text{ 이 관계는 } wfdf_2 \sqsubseteq_\alpha wfdf_1 \text{로 나타낼 수 있다.}$$

**예 3**  $wfdf_1(A) = s/w(\text{MAJOR})$  이고  $wfdf_2(A) = db(\text{MAJOR}) \vee ai(\text{MAJOR})$ 이라고 하자. 그러면,  $wfdf_2 \sqsubseteq_{0.9} wfdf_1$  인데 이유는  $\min(c \in |wfdf_2| = \{db, ai\})$ 에 대한  $wfdf_1(c/\text{MAJOR}) = \min(s/w(db), s/w(ai)) = \min(0.9, 0.91) = 0.9 > 0$  이기 때문이다.

퍼지 데이터베이스 내 클래스들은 클래스 계층 구조로 구조화되는데, 다음 정의는 이 계층 구조를 명시하고 있다.

**정의 11** 퍼지 데이터베이스, FDB는 다음과 같이 POSET(Partial Ordering SET)으로 정의 된다.

$$FDB = \langle C, \leq \rangle$$

여기서  $C$ 는 퍼지 클래스들의 집합이고,  $C_A, C_B \in C$ 에 대하여  $C_A$ 가 다음의 조건을 만족 할 때  $C_A \leq C_B$ 로 표기하며,  $C_A$ 를  $C_B$ 의 부클래스라고 한다.

$$\mu_{E_A}(t) \leq \mu_{E_B}(t) \text{ 이고, 여기서 } t \text{는 } E_A \text{내의 투플이다.}$$

개념적으로,  $C_A$  가  $\alpha$  정도로  $C_B$  의 부클래스 임을 스키마로 명시할 경우,  $\leq$ 는  $\leq_\alpha$  로 나타낸다.

클래스와 객체내에 표현되는 퍼지 정보를 표현하기 위한 구문은 다음과 같다.

```
defclass <class_name> isa[/<real>] <class_name> {{ <attr_name> ':' <f_attr_type> }+
<possibility>
[<var> : ] insert <class_name> {{<attr_name> ':' <fuzzy_attr_val> | (insert <class_name>
<attr_name> ':' <fuzzy_attr_val> )+ <possibility> )+ [ / <real>] }+ <possibility>)+}
<possibility> = poss : <real>
```

**예 4** 다음은 위의 구문을 이용한 스키마 정의문 및 객체 생성문의 예들을 보이고 있다.

```
defclass engineer isa/0.9 employee (name:char experience:f_experience salary:f_salary
lab:laboratory major: f_major)
defclass laboratory isa object (name:char area:f_area)
insert engineer (name.yang experience:15 salary:45,000 lab:<I1>/0.85 major: fuzzy|expert_db
poss:0.82)
insert engineer (name:park experience:moderate salary:50,000 lab:<I1>/0.8 major: se poss:0.6)
<I1> : insert laboratory (name:sys area:ood poss:0.89)
<I2> : insert laboratory (name:sw area:expert_db|odb poss:0.9)
```

'engineer' 클래스에서 'experience'와 'salary'는 퍼지형 애트리뷰트이다. 인스턴스의 선언시 이러한 애트리뷰트는 moderate나 high 등과 같은 퍼지 용어를 값으로 갖는다. major와 area 애트리뷰트는 퍼지 집합의 집합이다. 'engineer' 인스턴스의 예제에서 0.82는 이 인스턴스가 'engineer' 클래스에 속할 가능성을 나타낸다. 퍼지 isa 관계는 'engineer' 클래스에서 나타나 있다. 이 클래스는 0.9의 정도로 'employee' 클래스의 부 클래스임을 나타내고 있다. 복합 애트리뷰트 lab의 복합계층링크도 퍼지화되어있다. <I1> 과 <I2>는 insert 명령에서 다른 객체를 애트리뷰트의 값으로 연결하기 위한 투플 변수이다.

## 2.2 규칙 평가 방법의 정형화

본 절에서는 퍼지 객체 추론을 위한 규칙 평가 프로시저를 설명한다. 다음 정의는 규칙의

기본이 되는 요소를 명시하고 있다.

**정의 12** 단일 변수 A를 갖는 패턴 필드(또는 간략하게 패턴 필드)는 규칙의 LHS (Left Hand Side)를 구성하는 기본 요소이며 다음과 같이 표현된다.

$$f = A:v$$

여기서 v는 wfdf  $\in$  WFDF 이거나 형 변수  $\langle X \rangle$ 이다. 형 변수는 관련된 형으로부터 값을 취하는 변수이다

**정의 13**  $t[A]$ 가 튜플 t의 애트리뷰트 A에 대한 애트리뷰트 값이라 하자, 패턴 필드  $f = A:v$ 에 대해  $f[A] = v$  라하자. 그러면, 패턴 필드 f에 대한 임계 값  $\alpha \in [0, 1]$  정 도의 확정적 담 집합을  $\|f[A]\|_{\alpha^*}$  라고 할 때, 이 집합은 다음과 같이 정의된다.

$$t \in \|f[A]\|_{\alpha^*} \text{ iff } t[A] \leq_{\alpha^*} f[A], \quad \alpha \leq \alpha'.$$

정의 13은  $t[A]$ 보다 패턴 필드의  $f[A]$ 가 더 일반적일 때 t 가 f를 확정적으로 만족시킬 수 있다고 정의하고 있다.

**보조 정리 1** 임계값을  $\alpha$ ,  $0 < 1$ 라 하자. 그러면,

$$t \in \|f[A]\|_{\alpha^*} \Leftrightarrow \min(c \in t[A] \text{ 에 대하여 } f[c/A]) = \alpha' \geq \alpha$$

여기서  $f[A] \in \text{WFDF}$ ,  $t[A] \in \text{WFDF}$  이다.

**증명** i) 보조 정리는 정의 10과 13으로부터 증명 될 수 있다.

**예 5**  $f_1=\text{experience: high}$ 이라 하자. 그러면,  $e_1 = \text{engineer(name:yang experience:15 salary:45000...poss:0.82)}$ 에 대해  $e_1 \in \|f_1[\text{EXPERIENCE}]\|_{\alpha^*}$  이다. 여기서  $\alpha = \mu_{\text{high}}(15) = 0.75$ .

**예 6**  $f_2=\text{major:ai}$ 이라 하자. 그러면  $e_1 = \text{engineer(name:yang experience:15 major:fuzzy |expert_db salary:45000 poss:0.82)}$ 에 대해  $e_1[\text{MAJOR}] \leq f_2[\text{MAJOR}]$  이므로,  $e_1 \in \|f_2[\text{MAJOR}]\|_{\alpha^*}$  이다. 여기서,  $\alpha = (\min(ai[\text{fuzzy}], ai[\text{expert_db}])) = \min(0.8, 0.9) = 0.8$  이다.

**예 7**  $f_3=\text{area : case}\vee\text{ood}$ 이라 하자. 그러면  $l_5 = \text{laboratory(name:cm area:se poss:0.81)}$ 에 대해,  $l_5 \in \|f_3[\text{area}]\|_{\alpha^*}$  이다. 그 이유는  $t[A] \leq f[A]$  이 성립하지 않기 때문이다. 즉 관적으로 t는 나중에 ood가 될 수도 있기 때문이다. (그림 1 참조).

**정리 1** 단순 필드  $f = A : \bigvee_{i=1}^s d_i$ 에 대해  $t \in \|f[A]\|_{\alpha^*}$  이라고 하자. 그러면,

$$\alpha = \min(\alpha_1, \alpha_2, \dots, \alpha_n),$$

$$\alpha_k = \max(\mu_{d1}(c_k/A), \mu_{d2}(c_k/A), \dots, \mu_{ds}(c_k/A), c_k \in t[A]), k=1, 2, \dots, n,$$

**증명** 이 정리는 정의 8과 13, 그리고 보조정리 1에 의해 증명된다.

**예 8**  $f_4 = \text{area:db} \vee \text{ai}$  이라 하자. 그러면  $l_2 = \text{lab}(\text{area} : \text{expert\_db} \mid \text{odb poss } 0.9)$ 에 대해,  $l_2[\text{AREA}] \subseteq f_4[\text{AREA}]$  때문에,  $l_2 \in ||f_4[\text{AREA}]||_{\alpha^* 0}$ 이다. 여기서  $\min(\max(\min(\min(db(expert_db), area(db)), \min(ai(expert_db), area(ai))), \max(\min(db(odb), area(db)), \min(ai(odb), area(ai)))) = \min(0.9, 0.83) = 0.83$ . (그림 2 참조)

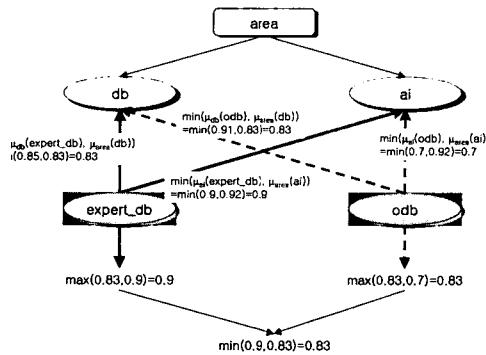


그림 2 이점적 퍼지 정보의 부합

**정의 14** 클래스  $C_N$ 의 패턴 필드들의 논리곱은  $C_N$ 의 단일 패턴이라고 하며 다음과 같이 표현된다.

$$P_m^{C_N} = \langle E \rangle N(f_1, f_2, \dots, f_p)$$

여기서  $p$ 는  $P_m^{C_N}$  내의 패턴 필드들의 수이며,  $\langle E \rangle$ 는 투플 변수이다.

**예 9**  $f_1 = \text{experience : high}$  이고  $f_2 = \text{major : ai}$  이라 하자. 그러면,

$$P_{m_1}^{C_{\text{engineer}}} = \langle E_1 \rangle : \text{engineer}(f_1, f_2) = \langle E_1 \rangle : \text{engineer}(\text{experience:high major:ai}).$$

**정의 15** 단일 패턴들의 논리곱을 패턴  $P$ 라 하고 다음과 같이 정의한다.

$$P = \bigwedge_{j=1}^n P_{m_j}^{C_{N_j}}$$

여기서  $n$ 은  $P$ 를 구성하는 단일 패턴들의 수이고,  $C_{N1}, C_{N2}, \dots, C_{Nn}$ 는 서로 다른 필요는 없는 클래스들이다.

정의 16 패턴들의 논리합은 규칙의 LHS라 하며, LHS인  $L$ 은 다음과 같이 정의된다.

$$L = \bigvee_{i=1}^m P_i$$

여기서  $P_i, i=1, 2, \dots, m$  는  $L$ 을 구성하는 패턴들이다.

$L$ 을 평가하기 위해서는 먼저, 이들 각 요소들과 FDB 내 투플들과의 정합 정도를 체계적으로 산정하여야 한다. 다음은 이를 위한 정의들이다.

정의 17  $\text{Poss}(f(t)) = \alpha, \alpha \in [0,1]$ 는 투플  $t$ 가  $f$ 를 만족하는 가능성 정도를 나타낸다.

정의 18  $P_m^{C_n} = N(f_1, f_2, \dots, f_p)$  가  $C_n$  의 단순 패턴이라면,  $t \in E_N$ 가  $P_m^{C_n}$  을 만족하는 가능성을 다음과 같이 정의된다.

$$\text{Poss}(P_m^{C_n}(t)) = \min(\text{Poss}(f_1(t)), \text{Poss}(f_2(t)), \dots, \text{Poss}(f_p(t)), \mu_{E_N}(t)),$$

예 10  $P_m^{C_{\text{engineer}}} = \langle E_1 \rangle : \text{engineer}(f_1, f_2) \circ$  라 하자. 그러면,

$$\begin{aligned} \text{Poss}((P_m^{C_{\text{engineer}}}(e_1))) &= \min(\text{Poss}(f_1(e_1)), \text{Poss}(f_2(e_1)), 0.82) \\ &= \min(0.75, 0.8, 0.82) = 0.75. (\text{예5, 예6}) \end{aligned}$$

정의 19  $P = (P_m^{C_{n1}})P_m^{C_{n2}})P_m^{C_{n3}})$ 가 클래스  $C_{N1}, C_{N2}, \dots, C_{Nn}$  의 패턴들이라 하자. 그러면,  $t_1 \in E_{N1}, t_2 \in E_{N2}, \dots, t_n \in E_{Nn}$  가  $P$ 를 만족할 가능성은 다음과 같다. ((

$$\text{Poss}(P(t_1, t_2, \dots, t_n)) = \min(\text{Poss}(P_m^{C_{n1}}(t_1)), \text{Poss}(P_m^{C_{n2}}(t_2)), \text{Poss}(P_m^{C_{n3}}(t_3))),$$

여기서  $t_1, t_2, \dots, t_n$  은 각각 서로 다른 필요는 없다.

예 11  $f_5 = \text{salary} : \text{moderate} \circ$  ]고,  $P_m^{C_{\text{engineer}}} = \langle E_2 \rangle : \text{engineer}(f_5, \text{lab}: \langle L_1 \rangle) \circ$  ]고,  $P_m^{C_{\text{laboratory}}} = \langle \langle L_1 \rangle : \text{laboratory}(f_4) \rangle$  일 때,  $P_1 = (P_m^{C_{\text{engineer}}}) \wedge (P_m^{C_{\text{laboratory}}}) \circ$  ]라 하자. 그러면,  $\text{poss}(P_m^{C_{\text{engineer}}}(e_1)) = \min(\text{poss}(f_5(e_1)), 0.85, 0.82) = \min(\mu_{\text{moderate}}(45,000), 0.85, 0.82) = \min(0.95, 0.85, 0.82) = 0.82 \circ$  ]고,  $\text{poss}(P_m^{C_{\text{laboratory}}}(l_2)) = \min(\text{poss}(f_4(l_2)), 0.9) = 0.83$  (예 8).

따라서,  $\text{Poss}(P_1(e_1, l_2)) = \min(\text{Poss}(P_m^{C_{\text{engineer}}}(e_1)), \text{Poss}(P_m^{C_{\text{laboratory}}}(l_2))) = \min(0.82, 0.83) = 0.82 \circ$  ]다.

이제, 다음의 두가지 부수적인 정의들을 이용해 LHS를 평가하도록 한다. 두 정의들은 LHS 내 투플 변수들을 체계적으로 표기하기 위하여 필요하다.

정의 20 투플 리스트  $E$ 는 다음과 같이 정의된다.

1. 투플 변수  $\langle E \rangle$ 는 투플 리스트  $E$ 이다.
2.  $\langle t_1, t_2, \dots, t_n \rangle \in E_{N1} \times E_{N2} \times \dots \times E_{Nn}$  은 투플 리스트  $E$ 이다.
3. 만약  $E_1, E_2, \dots, E_n$  이 투플 리스트이면,  $\langle E_1, E_2, \dots, E_n \rangle$ 도 투플 리스트이다.
4. 그 외에는 투플 리스트가 될 수 없다.

정의 21  $E$  와  $E'$ 를 투플 리스트라 하자. 만약  $E'$  내의 모든 투플 변수가  $E$ 에도 나타난다면  $E'$ 는  $E$ 의 부리스트라고 하며,  $E' \subseteq E$ 로 나타낸다.

규칙의 평가는 LHS 부분을 평가하는 것으로 볼 수 있다. 다음은 이 LHS 평가를 위한 정의이다.

정의 22  $L = \bigvee_{i=1}^m P_i$  를 LHS라 하자. 만약  $E = \langle E_1, E_2, \dots, E_m \rangle$  이  $L$ 의 투플 리스트라면

$$\text{Poss}(L(E)) = \max(\text{Poss}(P_1(E_1)), \text{Poss}(P_2(E_2)), \dots, \text{Poss}(P_m(E_m))).$$

여기서  $E_k$ 는  $P_k$ ,  $k=1, \dots, m$ 의 투플 리스트이다.

예 12  $P_2$  를  $P_2^{C_{\text{engineer}}} = \langle E_1 \rangle : \text{engineer}(f_1, f_2) \circ$  이고,  $L = P_1 \vee P_2$  이라 하자. 그러면

$$\text{Poss}(P_1(e_1, l_2)) = 0.82 \text{ (예 11) } \circ \text{이고,}$$

$$\text{Poss}(P_2(e_1)) = \min(\text{Poss}(f_1(e_1)), \text{Poss}(f_2(e_1)), 0.82) = \min(0.75, 0.8, 0.82) = 0.75 \text{ (예 10).}$$

$$\text{따라서, } \text{Poss}(L(e_1, l_2), \langle e_1 \rangle) = \max(\text{Poss}(P_1(e_1, l_2)), \text{Poss}(P_2(e_1))) = 0.82.$$

정의 23  $E$ 가 LHS의 투플 리스트라면

$$\text{FDB} \vdash_\alpha L(E) \Leftrightarrow \text{Poss}(L(E)) = \alpha$$

예 13 예 13 의  $L$ 에 대해,

$$\text{FDB} \vdash L(e_1, l_2), \langle e_1 \rangle \text{ 이고, } \alpha = \max(0.75, 0.82) = 0.82 \text{ 이다.}$$

정의 16에서 LHS가 논리합의 형태를 취하고 있는 이유를 살펴보자. 다음은 동일한 규칙명을 갖는 두 개의 규칙이다.

$$P_1(E_1) \Rightarrow \text{ACT}(E').$$

$$P_2(E_2) \Rightarrow \text{ACT}(E').$$

여기서 ACT는 수행부이고,  $E' \subseteq E_1, E' \subseteq E_2$  이다. 이때,  $(\text{not } P_1(E_1) \vee \text{ACT}(E')) \wedge (\text{not } P_2(E_2) \vee \text{ACT}(E')) \Leftrightarrow \text{not}(P_1(E_1) \vee P_2(E_2)) \vee \text{ACT}(E')$  이므로, 두 규칙은  $P_1(E_1) \vee P_2(E_2) \Rightarrow \text{ACT}(E')$  이 된다.

규칙의 RHS는 수행들의 논리곱으로 구성되며, 수행부는 LHS가 FDB에 의해 만족될 때

수행된다. 수행의 결과는 FDB를 변경시킬 수도 있기 때문에, 수행부는 또 다른 규칙의 실행을 야기할 수 있다. 규칙이 실행될 때, 모든 투플 리스트는 FDB의 투플들에 의해 실례화 된다. 다음 정의 17은 RHS를 포함하는 규칙을 정의하고 있다.

**정의 24**  $L$ 과  $R$ 을 각각 LHS와 RHS라 하면, 규칙은 다음과 같다.

$$L(E) \Rightarrow R(E'), \text{ 그리고 } R(E') = \bigwedge_{i=1}^l ACT_i(E'_i; P_{m_i})$$

여기서  $ACT_i$  는 한 수행부이고,  $E'_i$ 는 투플 변수이며,  $P_{m_i}$ 는 클래스 이름을 갖지 않는  $\text{type}(E'_i)$ ,  $i = 1, \dots, l$ 의 패턴이다. 패턴들은 필요할 때만 명시된다. 또,  $E$ 와  $E'$ 는  $E' \sqsubseteq E$ ,  $E' = \langle E'_1, E'_2, \dots, E'_l \rangle$ 을 만족하는 투플 리스트이다.

다음은 규칙의 명세 구문과 예를 보이고 있다.

```
defrule <rule_name>
'(' <var>|not '::' [<class_name>] '(' ['!']<attr_name> '::<attr_val>+ ')'+'
[(& 'with:' <a>)]
=>
<command> [<var> '::'] <class_name> '(' <attr_name> '::<attr_val>+ ')' '+'
<attr_val> = <fuzzy_attr_val>|<var>
<command> = modify | insert
```

**예 14** 다음 예는 지식 기반 프로젝트 수행을 위해 engineer를 분류하는 규칙이다.

```
(defrule classification_kb_engineer
<E1>:engineer (name:<x> experience: high major:ai)
(with:0.6)
=>
insert kb_engineer (name:<x>))
(defrule classification_kb_engineer
<E1>:engineer (name:<x> salary:moderate,lab:<L1>)
<L1>:laboratory(area:db|ai)
(with:0.6)
=>
insert kb_engineer (name:<x>))
```

정의 25  $\alpha \in [0,1]$  이고 규칙  $\gamma$  을  $L(E) \Rightarrow ACT(E')$  라고 하면,

$$\begin{aligned}\mu_\gamma(E, E') &= \alpha, \quad \mu_L(E) \leq \mu_{\text{type}}(E')(E') \text{ 일 때} \\ &= 0, \quad \text{그외}\end{aligned}$$

이제, 임계값이 0.6일 때, 객체 e3에 대해 예 9의 규칙이 평가되는 과정을 살펴보자.

<l2> : insert laboratory (name:sw, area:expert\_db|odb poss:0.9)

<e1> insert engineer (name.yang experience:15 salary:45,000 lab:<l2>/0.85 major:fuzzy|expert\_db poss:0.82)

첫 번째 규칙의 LHS에 위의 객체가 정합될 가능성은 0.67이고, 두 번째 규칙의 LHS에 정합될 가능성은 0.7이다. 몇 개의 규칙이 논리합(or) 관계일 때, 가장 높은 가능성이 선택되므로 이 예의 최종 가능성 값은  $\max(0.67, 0.7) = 0.7$ 이다. 이 값은 규칙의 임계값보다 크므로 규칙이 실행되고 객체가 생성된다. 0.7은 수정된 객체가 RHS 부분에서 명시된 클래스에 속할 소속성 정도로 해석된다. 그림 3는 이 규칙의 평가 과정을 보여주고 있다.

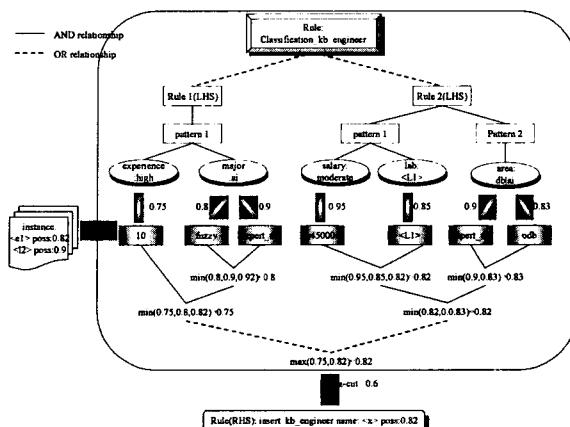


그림 3 폐지 규칙 평가 과정 예

### 3. 결 론

본 논문에서는 기존의 폐지 규칙 기반 전문가 시스템 언어들이 복합 객체 추론 기능이 없으며, 의미적으로 이해하기 쉽고 개념적으로 사용하기 용이한 폐지 추론을 지원하지 못하고, 지식 표현과 추론 방식이 기존의 데이터베이스 모델과 구문이나 의미에서 현격한 차이를 보이고 있다는 단점을 해결하기 위한 폐지 객체 추론 모델의 정형화를 보이고, 이를 지원하기 위한 ICOT을 제시하였다. 본 논문에서 제시한 모델은 객체 지향 방식의 병합으로

기존의 규칙기반 전문가 시스템의 모델링 능력을 향상 시켰으며, 애매성과 모호성을 포함하는 불확실한 정보의 표현과 처리를 지원함으로써 데이터의 의미적 표현력을 강화시킨다.

### 참고문헌

- [1] ADLASSNIG, K. P. Fuzzy set theory in medical diagnosis. *IEEE Transactions on Systems Man. and Cybernetics*, SMC-16, 1986, 260-265.
- [2] ZEMANKOVA, M. FILIP: A Fuzzy Intelligent Information System with Learning Capabilities. *Information System*, 14(6), 1989, 473-486.
- [3] Binaghi, E. A fuzzy logic inference model for a rule-based system in medical diagnosis. *Expert Systems*, 7(3), 1990, 134-141.
- [4] Orchard, R. A. *FuzzyCLIPS*, a users guide. Version 6.02, Knowledge Systems Laboratory: Institute for Information Technology, National Research Council, Canada, September, 1994.
- [5] SHYY, Y. M. and SU, S. Y. W. K: A high-level knowledge base programming language for advanced database applications. In: James Clifford, Roger King, eds. *ACM SIGMOD*, 1991, Denver, Colorado, USA, May, 1991, 29-31.,
- [6] SHENG, O. R. L. and WEI, C. P. Object-oriented modeling and design of coupled knowledge-base/database systems. In: N. J. Cercone, eds. *Proceedings of the International Conference on Data Engineering*, IEEE, Tempe, Arizona, USA, 1992, 98-105.
- [7] DECHAMBOUX, P. and RONCANCIO, C. Peplomd : an object-oriented database programming language extended with deductive capabilities, In: D. Karagiannis, eds. *Proceeding of Database and Expert Systems Applications*, Athens, Greece, 1994, 2-14.
- [8] SRIVASTAVA, D., RAMAKRISHNAN, R., SESHADRI, P. and SUDARSHAN, S. Coral ++ : adding object-orientation to a logic database language. In: R. Agrawals, eds. *Proceedings of the 19th VLDB Conference*, Dublin, Ireland, 1993, 158-170.
- [9] DIAZ, O., PATON, N., AND GRAY, P. Rule Management in Object-Oriented Databases: A Uniform Approach. In G. M. Lohman, eds. *Proceedings of the 17th International Conference on Very Large Data Bases*, Barcelona. Sept. 1991, 317-326.
- [10] Yang, J. D. F\_MP: A fuzzy match framework for rule-based programming. *Data and Knowledge Engineering*, 23(3), 1997.
- [11] Yang, J. D. and Lee, D. , Incorporating Concept-based Match into Fuzzy Production Rules. *Information Sciences*, 104(3/4), 1997.
- [12] J. D. Yang, "A Uniform Framework for the Deductive Manipulation of Fuzzy Information," *Fuzzy sets and Systems*, Vol. 64, pp. 377-385, 1994.