

사례 기반 추론 시스템에서 적응 지식 자동 획득 모델에 관한 연구

A Study on Adaptive Knowledge Automatic Acquisition Model from Case-Based Reasoning System

이상범 · 김영천 · 이재훈 · 이성주

Sang-Bum Lee, Young-cheon kim, Jae-Hoon Lee, Sung-joo Lee
Department of Computer Science, Chosun University
E-mail : sblee@stmail.chosun.ac.kr, yckim@stmail.chosun.ac.kr,
nuridepo@cafe.chosun.ac.kr, sjlee@chosun.ac.kr

ABSTRACT

In current CBR(Case-Based Reasoning) systems, the case adaptation is usually performed by rule-based method that use rules hand-coded by the system developer. So, CBR system designer faces knowledge acquisition bottleneck similar to those found in traditional expert system design. In this thesis, I present a model for learning method of case adaptation knowledge using case base. The feature difference of each pair of cases are noted and become the antecedent part of an adaptation rule, the differences between the solutions in the compared cases become the consequent part of the rule. However, the number of rules that can possibly be discovered using a learning algorithm is enormous. The first method for finding cases to compare uses a syntactic measure of the distance between cases. The threshold for identification of candidates for comparison is fixed th the maximum number of differences between the target and retrived case from all retrievals. The second method is to use similarity metric since the threshold method may not be an accurate measure. I suggest the elimination method of duplicate rules. In the elimination process, a confidence value is assigned to each rule based on its frequency. The learned adaptation rules is applied in given target problem. The basic process involves search for all rules that handle at least one difference followed by a combination process in which complete solutions are built.

Keyword : Case-Based Reasoning, Adaptive Knowledge, Reasoning.

I. 서론

사례 기반 추론(Case-Based Reasoning : CBR) 시스템은 과거의 문제를 해결하기 위해 사용되었던 해를 재사용하거나 적응시킴으로써 현재의 문제를 해결하는 방법이다. 최근 지식 기반 의사 결정 지원 시스템, 예측, 진단, 계획 등 다양한 영역에서 이 방법이 사용되고 있다. 입력 문제를 해결하기 위한 사례 기반 시스템의 기본적인 추론 과정은 과거의 사례들을 사례 베이스에 저장 한 후, 사례 베이스로부터 현재 문제와 유사한 사례를 찾아내는 추출(retrieval) 단계, 추출된 사례를 사용하여 문제를 해결하기 위해 시도하는 재사용(reuse) 단계, 추출된 사례가 현재의 문제와 유사하지 않을 경우 추출된 사례의 해를 수정하는 사례 적응(adaptation 또는 revise) 단계 그리고 새로운 해를 새로운 사례로서 사례 베이스에 저장하는 사례 저장(retain)단계로 이루어진다. 사례 베이스로부터 추출된 사례는 현재의 문제와 유사한 문제일 뿐 완전히 동일한 것은 아니므로 이전 사례의 해를 적용하는 단계가 필수적이다.

본 논문에서는 사례기반 추론 시스템의 사례 적응 문제를 해결하기 위해 사례 베이스에 존재하는 사례들

로부터 적응 지식을 자동으로 획득하는 모델을 제안하였다. 본 논문은 다음과 같이 구성되어 있다. 2장에서는 관련 연구 및 사례 적응 방법에 대해 고찰한 후 3장에서는 본 논문에서 제안한 적응 지식 자동 획득 모델 내용을 기술하였으며, 4장에서는 제안한 방법을 평가하기 위한 적용 영역과 적용한 실험 결과 및 평가 내용을 기술하고 마지막으로 5장에서는 결론과 향후 과제에 대해 기술하였다.

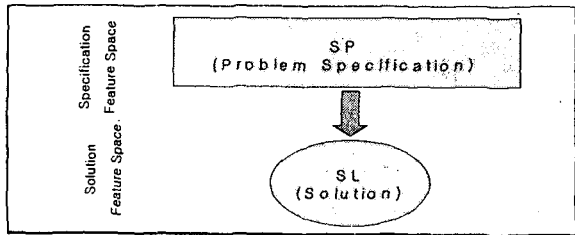
II. 관련 연구 및 사례 적응 방법

2.1 경험에 기반한 문제 해결

문제 해결 과정에서 가장 중요한 목적은 문제에 대한 명세를 이용하지 않고, 정확한 해를 찾아내는 것이다. 해와 명세는 속성의 집합으로 기술될 수 있다. 이러한 문제들을 해결하기 위한 전통적인 인공지능 방법에서 문제 해결 과정은 문제의 명세를 기술한 속성 공간으로부터 해를 찾아가는 하나의 생성과정으로 볼 수 있다[11].

2.2 사례 기반 추론

새로운 문제가 주어졌을 때 과거의 유사한 사례를 추출한 후, 이 사례의 해를 적용시킴으로써 탐색에 대한 비용을 줄이기 위해 노력한다. 과거의 문제 해결 과정을 저장하고 있는 양질의 사례와, 해를 유도하기 위해 휴리스틱(heuristic) 적용 규칙에 의존함으로써 정교한 영역 지식을 요구하지 않는 기반 추론에 대한 개념은 [그림 2-1]와 같다.



[그림 2-1] 사례기반 추론에서의 문제 해결 과정

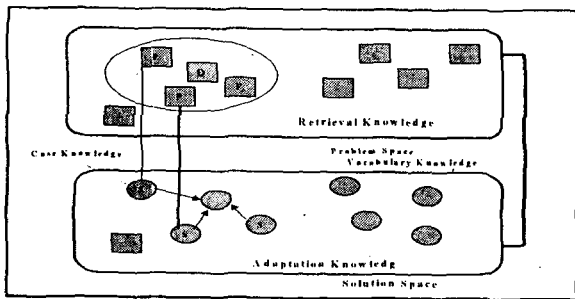
2.3 기계 학습과 데이터 마이닝

인공지능 분야에서 실제계에 존재하는 지식의 획득은 지식 획득의 병목 현상으로 인하여 그리 쉬운 일이 아니다. 이러한 지식 시스템의 지식 획득 문제를 완화하기 위해서는 시스템에 학습 기능을 부여함으로써 시스템 자체가 문제 해결의 경험을 통하여 지식을 획득하고 지식을 정제해 가는 기반 기술을 통털어서 기계 학습(machine learning) 방법이라 한다[1].

데이터 마이닝은 인공지능 분야의 기계 학습 이론에 그 뿌리를 두고 있으며 “대규모 데이터베이스에 존재하는 데이터로부터, 함축적이고 기존에 알려지지 않은 유용한 정보를 찾아내는 과정”으로 정의할 수 있다[4, 8].

2.4 지식 컨테이너

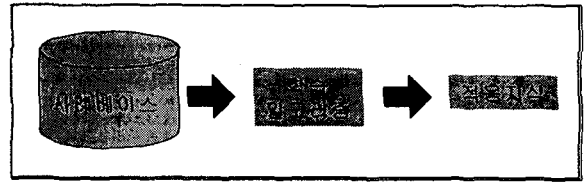
사례기반 추론은 지식을 사용하는 지식기반 문제 해결 방법 중의 하나로, 여기에서 지식기반은 시스템 안에 이미 획득되어 존재하는 지식으로, 지식을 포함할 수 있는 네 개의 지식 컨테이너(container)를 어휘 지식, 추출 지식, 적용 지식, 사례 지식으로 정의하여 [그림 2-3]과 같이 표현할 수 있다[9,12].



[그림 2-3] 지식 컨테이너

III. 적응 지식 자동 획득 모델

본 논문은 지식 컨테이너의 개념 및 기계 학습 이론과 데이터 마이닝의 개념을 기반으로 하여 사례 적용 지식의 자동 학습 모델을 [그림 3-1]과 같이 제안하는 데 있다.

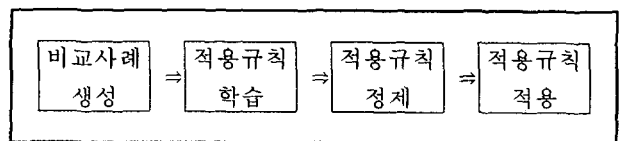


[그림 3-1] 학습 모델의 개념

제안 모델은 데이터 마이닝의 연관 규칙 추출의 개념을 이용하여 데이터 마이닝의 대상이 되는 대량의 데이터베이스를 사례베이스로 가정하고 이 사례베이스의 사례들 간의 상호 관련 있는 연관 규칙을 추출하여 그 규칙을 사례 적용 시에 이용하는데 그 초점이 있다. 데이터 마이닝에서 연관 규칙의 획득을 위한 여러 알고리즘이 제안되었으나 그 알고리즘을 본 연구에서 바로 적용하는 것은 문제가 따른다. 왜냐하면 사례는 규칙과 비슷하게 문제 및 해에 대한 묘사로 이루어져 있으므로 연관 규칙을 획득할 때 문제의 차이가 해에 미치는 영향을 찾아내야 하기 때문이다.

따라서 본 논문에서는 사례와 사례의 속성을 직접 비교하는 방법을 사용하였으며 이 알고리즘을 사용하기 위해서는 영역으로부터 추출된 사례의 지식이 영역 전문가에 의해서 검증된 사례라고 가정하였다. 본 연구에서 제안한 자동 획득 모델을 통하여 생성된 적용 규칙은 영역 지식으로부터 획득된 지식과 통합되어야 하며 또한 문제 해결 단계에서 사용자와 상호 작용을 통하여 지식을 획득 할 수도 있어야 한다. 그러나 이러한 각각의 지식을 통합하는 과정은 그리 간단한 일이 아니다. 따라서 본 논문에서는 사례베이스로부터 적용 지식을 획득하여 이 지식을 사례 적용 시에 이용하는 것에 주된 관심을 두었다.

사례 베이스를 이용하여 적용 지식을 획득하기 위한 본 모델의 기본 개념은 [그림 3-2]와 같다. 먼저 사례로부터 적용 지식을 학습하기 위해 비교 사례의 기준을 정한 후 이 기준을 사용하여 비교 사례를 선택하며 선택된 사례 사이의 속성 비교를 통하여 적용 지식을 학습한다. 적용 지식을 학습하기 위해서는 사례 베이스안의 각 사례의 속성의 차이를 계산하여 적용 규칙의 전제부를 구성하고 해의 차이점을 계산하여 적용 규칙의 결론부를 구성한다. 이렇게 획득된 적용 규칙들은 중복된 규칙이 많으므로 규칙의 정제 과정을 거쳐야 한다. 마지막으로 추출된 사례의 해를 적용시키기 위해 적용 지식을 탐색하고, 적용할 적용 규칙이 많은 경우 적용의 우선 순위 등을 결정해야 한다.



[그림 3-2] 제안 모델의 수행 절차

3.1 비교 사례의 선택

비교 사례를 선택하기 위한 방법 중 임계값을 이용하는 방법은 사례 사이의 거리를 측정하기 위해 단순히 구문론적(syntactic)방법을 이용하는 것이다. 이 방법은 사례간의 속성값의 차이를 조사하여 차이의 수를 사례

간의 거리로 설정하는 방법이다. 이 임계값을 구하기 위해서는 사례베이스 내에 존재하는 각 사례를 대상 문제로 설정한 후 이 문제와 가장 유사한 사례를 추출한다. 추출된 사례와 이 사례간의 속성의 차이를 기록하고 이 과정을 사례 베이스 안에 존재하는 모든 사례에 대해서 적용한 후 그 중 속성 차이를 개수가 가장 큰 값을 임계값으로 설정하는 방법이다. 유사도 함수를 정하기 위해서는 영역 지식을 이용하여, 각 속성의 중요도를 결정한 후, 가중치 값을 주어야 하는데 실세계에서 이 유사도 함수를 정확히 결정하는 것은 그리 간단하지 않다. 따라서 이 방법은 추출되는 적용 지식은 많을 수 있지만 이것 때문에 유사도 함수를 정확히 유지해야 하는 수고를 덜 수 있다. 즉 추출된 사례가 문제의 사례와 거리가 멀다 하더라도 적용 지식이 이러한 문제를 보완할 수 있음을 의미이다. 이것은 사례 추출과 사례 적용 사이의 관계가 상호 보완적이라는 것으로 볼 수 있다. 사례베이스로부터 임계값을 찾기 위한 알고리즘은 다음과 같다.

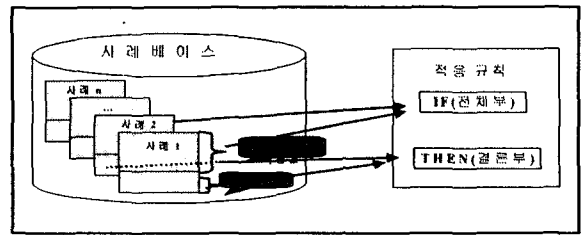
```
[알고리즘 3-1] 임계값 결정 알고리즘
1. old_diff=0; new_diff=0; threshold=0;
2. for(i=1; i<case_number; I++){
3.   problem=case_base(i);
4.   r_case=problem사례와 가장 유사한 사례를 추출;
5.   new_diff=problem사례와 r_case의 속성차이를 계산;
6.   threshold=max(new_diff,old_diff);
7.   old_diff=new_diff;
8. }
9. return(threshold);
```

임계값 이용법은 단지 속성값 차이의 개수만을 계산하는 구문론적 방법을 사용하므로 각 속성의 중요도에 따라 사례간의 차이를 계산하는 의미론적방법을 사용하는 것에 비해 정확도가 떨어질 수 있다. 예를 들어 속성 중의 어떠한 것은 전혀 관계가 없는 것도 있을 것이며 속성간의 중요도의 우선 순위가 존재할 수 있기 때문이다. 이러한 관계들을 고려하여 사례들을 비교하면 좀 더 정확한 유사도 값을 계산할 수 있을 것이다. 따라서 이 방법을 사용하기 위해서는 사례 속성들 간의 중요도를 고려한 유사도 함수가 존재해야 한다. 적용 지식 획득을 위한 비교 사례 선택에 있어서 가장 중요한 것은 유사한 사례들간의 거리를 어떻게 적용을 통하여 연결해 줄 것인가 하는 것이다. 만약 사례의 추출시 유사도 매트릭스를 사용한다면, 사례 적용 지식 추출을 위해서도 동일한 유사도 매트릭스를 사용해야 한다. 왜냐하면 유사도 매트릭스를 통하여 계산되거나 유사도 값이 특정 유사도 값보다 높을 경우만 적용 지식을 획득해야 되기 때문이다. 이 방법은 유사도 함수를 사용하여 비교할 사례 대상을 찾아내므로 추출되는 적용 지식은 임계값 이용법보다는 상당히 적을 것이다. 또한 이 방법에서 사용하는 유사도 함수가 완전히 정확할 필요는 없다. 왜냐하면 추출된 적용 지식은 유사도 함수의 정확도에 따라 적용 범위가 결정되기 때문이다.

3.2 학습 알고리즘의 선택

메이터 마이닝에서 숨겨진 패턴을 탐사하는 연구 중에서 연관 규칙 탐사 알고리즘의 적용으로 새로운 패턴을 찾아내는 연구가 많이 수행되고 있다. 연관 규칙은 항목들의 그룹과 다른 항목들의 그룹 사이에 강한

연관성이 있음을 밝혀준다. 적용 지식을 학습하는 것은 크게 적용 지식의 생성 단계와 적용 단계로 구분할 수 있다. 적용 지식의 생성 단계는 사례 베이스의 사례들을 쌍으로 묶어서 서로 비교하는 과정이다. 이 때 모든 사례를 비교할 필요가 없는 이유는 문제 사례에 대해 사례 베이스로부터 추출된 사례는 매우 유사하므로 이 사례간의 차이점을 해결 할 수 있는 적용 규칙만을 생성하면 되기 때문이다. 사례 비교를 통하여 사례간의 속성의 차이는 적용 규칙의 전제부로 구성되고 비교된 사례간의 해의 차이는 결론부를 구성하는 적용 지식의 생성은 [그림 3-3]과 같다.



[그림 3-3] 적용 지식의 생성

이와 같은 적용 지식을 사례 베이스로부터 학습하기 위한 알고리즘은 다음과 같다.

```
[알고리즘 3-2] 적용 지식 학습 알고리즘
1. adapt_rule_base = {}; //생성된 적용 규칙 저장
2. single_rule = {}; //한 개의 규칙 저장
3. for(i=1; i<case_number; I++){
4.   problem = case_base(i);
5.   for(j=i+1; j<=case_number; j++){
6.     target=case_base(j);
7.     if(mode="임계값") {
8.       diff=problem과 target의 속성 차이의 수 계산;
9.       if(diff<=ATT_THRESHOLD) {
10.        single_rule=problem과 target으로부터 규칙 생성;
11.        adapt_rule_base = single_rule U adapt_rule_base;
12.       }
13.     }
14.     else if(mode="유사도") {
15.       sim_value=problem과 target의 유사도 계산;
16.       if(sim_value>=SIM_THRESHOLD) {
17.        single_rule=problem과 target으로부터 규칙 생성;
18.        adapt_rule_base = single_rule U adapt_rule_base;
19.       }
20.     }
21.   }
22. return(adapt_rule_base)
```

3.3 적용 지식의 정제

사례 비교를 통하여 획득된 규칙은 각 속성의 차이를 나타내는 규칙의 전제부가 하나의 해 변화를 나타내는 결론부로 연결되도록 적용되기 이전에 정제 과정을 거쳐야 한다. 사례 비교를 통해 획득된 규칙 집합은 중복된 규칙을 포함할 수 있다. 중복에 대한 정의는 적용 영역에 따라 다양할 수 있지만 본 논문에서는 동일한 속성의 차이(전제부가 같고)가 거의 유사한 해 변화와 관련이 있는(결론부) 두 규칙을 중복된 규칙으로 간주하였다. 중복된 규칙 집합에 의해 표현된 지식들은 해

당 규칙들의 해 변화 부분의 평균값을 취하여 하나의 규칙으로 만들 수 있다. 이것은 유사한 결론부를 가지고 있는 여러 규칙들에 대한 일반화의 과정으로 볼 수 있다. 이 단계에서는 각 규칙에 대해 다음과 같은 식을 이용하여 확신도 값을 할당한다.

$$\text{규칙의 확신도} = \frac{\text{해당 규칙의 중복수}}{\text{규칙 집합에서 규칙의 최대 중복수}}$$

확신도 값이 높을수록 해당 규칙이 사례비교를 통해 많이 획득되었다는 의미를 나타내며, 또한 이것은 해당 규칙이 나타내는 지식이 좀 더 진실에 가깝다는 것을 의미한다. 이러한 규칙의 확신도 값은 적용할 적용 규칙이 많은 경우 적용 규칙의 우선 순위를 결정하기 위한 기준의 하나로 사용된다.

```
[알고리즘 3-3] 규칙의 정제 알고리즘
1. 각 규칙의 중복수 및 최대 중복수 계산;
2. duplicate_rules={}; refined_rule_base={};
   refined_rule={};
3. for(i=1; i<=rule_number; I++) {
4. rule=rule_base(i);
5. if(rule!={}) {
6. duplicate_rules=rule과 전제부가 동일한 규칙 탐색;
7. refined_rule=duplicate_rules의 결론부의 평균을 계산
   하여 새로운 규칙 생성;
8. refined_rule=refined_rule U 규칙의 확신도;
9. refined_rule_base=refined_rule_base U refined_rule;
10. duplicate_rules에 속한 규칙을 rule_base에서 제거; }
11. return(refined_rule_base);
```

3.4 적응 지식의 적용

위에서 언급한 적응 규칙의 생성 단계에서는 사례의 속성의 차이가 해의 차이에 어떤 관련을 갖고 있는 지를 나타내는 많은 규칙들을 생성해낸다. 그러나 이러한 규칙들을 적용하기 위한 방법을 제공하지 않는다면, 이런 규칙은 아무 의미가 없다. 적응 규칙을 학습하는 것과 별개로 본 논문에서는 이러한 적응 규칙을 적용하기 위한 방법을 제안하였다.

```
[알고리즘 3-4] 적응 규칙의 적용 알고리즘
1. apply_rules = {};
2. diff=문제 사례와 추출 사례의 속성의 차이를 계산;
3. apply_rules=적용 규칙 중 적용 가능한 모든 규칙 선택;
4. apply_path=apply_rules을 이용하여 적용할 패스 선택;
5. new_solution=apply_path를 사용하여 새로운 해를 구함;
6. return(new solution);
```

IV. 실험 및 평가

4.1 적용 영역

본 실험의 적용 영역은 자동차 가격의 산출과 여행 가격의 결정을 대상으로 하였다. 실험 자료로 사용한 사례는 AI-CBR Group에서 제공하는 Case-Base Archive로부터 획득하였다[5]. 각 영역에서 획득한 사례의 해는 모두 가격을 나타내는 숫자로 이루어져 있으며, [표 4-1]은 각 적용 영역에서 획득한 사례 수 및 각 사례 베이스의 속성 수를 나타낸 것이다.

[표 4-1] 적용 영역의 사례 특성

적용 영역	사례 수	사례의 속성 수
자동차 가격 결정	205	24
여행 가격 결정	1,475	8

각 영역에서 획득한 사례는 [표 4-2]와 [표 4-3]과 같은 속성 및 속성 값을 갖는다.

[표 4-2] 자동차 가격 결정 영역에서의 사례 구성

속 성	속 성 범 위
make:	alfa-romero, audi, bmw, dodge, chevrolet, honda, isuzu, jaguar, mazda, mercedes-benz, mercury, mitsubishi, nissan, peugot, ssab, plymouth, porsche, renault, subaru, toyota, volkswagen, voly.
fuel-type:	diesel, gas.
aspiration:	sdt, turbo.
no-of-doors:	four, two.
body-style:	hardtop, wagon, sedan, hatchback, convertible.
drive-wheels:	4wd, fwd, rwd.
engine-location:	front, rear.
wheel-base:	continuous from 86.6 to 120.9
length:	continuous from 141.1 to 208.1
width:	continuous from 60.3 to 72.3
height:	continuous from 47.7 to 59.8
curb-weight:	continuous from 1488 to 4066
no-of-cylinders:	twelve, eight, five, six, four, three, two.
engine-type:	dohc, dohcv, l, ohc, ohcf, ohcv, rotor.
engine-size:	continuous form 61 to 326
fuel-system:	lbbl, 2bbl, 4bbl, idi, mfi, mpfi, spdi, spfi.
bore:	continuous from 2.54 to 3.94.
stroke:	continuous from 2.07 to 4.17.
compression-ratio:	continuous from 7 to 23.
horsepower:	continuous from 48 to 288.
peak-rpm:	continuous from 4150 to 6600.
city-mpg:	continuous from 13 to 49.
highway-mpg:	continuous from 16 to 54.
price:	continuous from 5118 to 45400.

[표 4-3] 여행 가격 결정 영역에서의 사례 구성

속 성	속 성 범 위
HolidayType:	Bathing, Active, Active, Education, City, Recreation, Wandering, Language, Skiing
Numero fPersons:	continuous from 1 to 12
Region:	유럽 및 아프리카의 56개 지역
Transportation:	plane, car, train
duration:	1주, 2주, 3주
season:	Jan. Feb. Mar, Oct, Nov, Dec.
Accommodation:	Holidayflat,two/three/four/five stars
price:	continuous from 319 to 6609

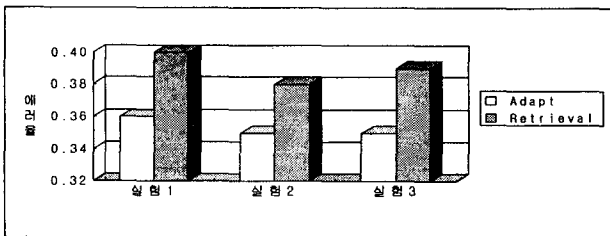
4.2 실험 결과

본 논문에서 제안한 적응 지식 자동 획득 모델을 평가하기 위하여 윈도우즈98에서 Visual Basic 6.0을 이용하여 구현하였으며, 사례 인덱스는 배열을 사용하였다. 시스템을 평가하기 위한 평가 기준은 시스템이 입력 문제를 얼마나 정확하게 해결하는가를 나타내는 정확도를 기준으로 정하였다. 정확도를 평가 기준으로 결정한 이유는 사례로부터 획득한 지식이 시스템의 정확도를 향상시키는 데 얼마나 도움을 주는가를 측정해 보기 위함이었다. 정확도를 측정하기 위하여 시스템 제공한 해와 전문가가 제공한 해를 바탕으로 다음과 같은 식을 사용하여 시스템의 에러율(E)을 계산하여 평가하였다.

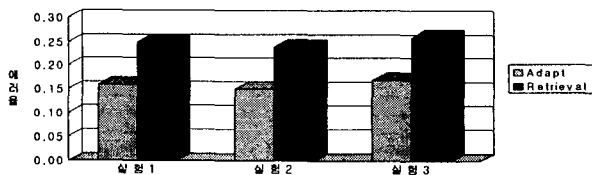
$$E = \frac{\sum_{i=0}^N |Ssolution[i]-Esolution[i]|}{Esolution[i]}$$

4.2.1 적응 및 추출 방법의 정확도 비교

[그림 4-1]과 [그림 4-2]는 학습된 적응 지식을 사용할 경우 문제 사례의 가격 결정에 대한 예측의 정확도가 향상됨을 보여주고 있다. 자동차 가격 결정의 영역에서는 약 3%~4%정도 향상됨을 보여주고 있으며, 여행 가격의 결정 영역에서는 약 9%정도의 향상을 보여주고 있다. 이와 같이 영역에 따라 정확도의 향상이 다르게 나타나는 이유는 영역으로부터 획득한 사례 수 및 학습한 적응 규칙의 수가 정확도의 향상에 영향을 줄 수 있음을 의미한다.



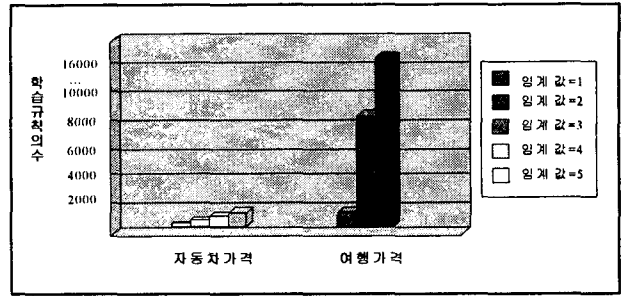
[그림 4-1] 적응/검색 시스템의 에러율 비교(자동차)



[그림 4-2] 적응/검색 시스템의 에러율 비교(여행)

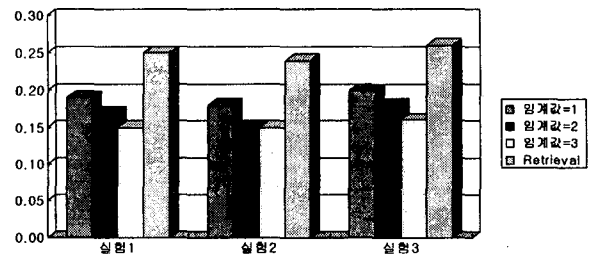
4.2.2 임계값에 따른 정확도 비교

적용 지식을 추출할 때 규칙의 양은 임계값에 의해 좌우될 수 있다. 이 실험에서는 임계값이 해의 정확도에 미치는 영향을 알아보기 위해 임계값을 다양하게 변화시켜 가며 해의 정확도를 평가하였다. [그림 4-3]은 각 영역의 사례로부터 임계값의 변화에 따라 추출된 적응 규칙의 개수를 나타내고 있다.



[그림 4-3] 임계값에 따른 획득 규칙의 수

[그림 4-4]에서 임계값 계산 알고리즘에 의해 구해진 임계값보다 임계값이 작아짐에 따라 시스템의 정확도는 감소함을 알 수 있다. 여행 가격의 실험 2의 평가에서는 임계값이 2와 3인 경우에 에러율이 변화하지 않음을 볼 수 있다. 이러한 이유는 추출된 사례와 문제 사례의 속성의 차이가 주로 2개인 경우로 해석할 수 있다. 왜냐하면 임계값이 3인 경우엔 속성의 차이가 2개인 경우의 규칙을 모두 포함하고 있기 때문이다. 또한 임계값이 1인 경우에도 사례 추출만을 지원하는 시스템에 비해 자동차 영역에서는 약 0.5%~2%정도 정확도가 높으며, 여행의 영역에서는 약 6%정도 향상되었음을 확인할 수 있다.



[그림 4-4] 임계값에 따른 에러율 비교(여행)

V. 결론 및 향후 연구

현재까지 사례기반 추론 시스템에서 사례 베이스의 유일한 목적은 현재 문제에 대한 유사한 해를 제공하는 데이터 베이스내의 레코드들과 마찬가지로, 사례 적응 단계에서 이용 가능한 내재적인 적응 지식을 포함하고 있다. 이러한 사례들 사이에 포함된 내재적인 지식을 추출하기 위해 사례 비교를 통한 적응 지식을 획득하는 알고리즘을 제안하였다. 사례 베이스에 존재하는 사례들로부터 적응 지식을 획득하기 위해서는 적응 지식을 획득할 비교 사례의 선택을 위해서 본 논문에서는 단순히 사례의 속성 차이의 수를 계산하여 유사 사례를 선택하는 유사도 방법을 제안하였다. 임계값은 사례 베이스 안에 존재하는 각 사례를 문제 사례로 가정할 후 가장 큰 값을 임계값으로 결정하였다. 또한 제안한 방법을 이용하여 획득한 적응 규칙은 많은 중복 규칙을 포함할 수 있다. 획득한 적응 규칙의 적용을 위해 중복 규칙을 하나의 규칙으로 만드는 방법과 적응 규칙의 적용시 가장 적절한 적응 규칙을 찾기 위한 방법을 제안하였다.

사례 비교를 통하여 획득된 적응 규칙이 문제 해결의 정확도를 향상시키는데 얼마나 유용한가를 자동차 및

여행 가격 결정의 영역에서 실험하였다. 실험 결과, 사례 추출만을 지원하는 시스템보다 추출된 적응 지식을 이용하는 시스템에서 제공하는 해의 정확도가 높았으며, 사례 속성의 손실값이나 사례의 해가 부정확한 경우에도 획득된 적응 규칙은 정확도의 향상에 도움을 줄 수 있음을 확인하였다. 따라서 본 논문에서 제안한 모델을 이용하면 기존의 사례 기반 시스템 개발 시 개발자가 영역으로부터 적응 규칙을 획득한 후, 직접 적응 규칙을 코딩하는 어려움을 줄일 수 있으며, 시스템 개발에 소요되는 비용을 줄일 수 있다고 사료된다.

참고문헌

- [1] 김은수, 이승현, 이재수, 유충상, 공역, "뉴로, 퍼지, AI 핸드북", 도서출판 技多利, 1998.
- [2] 박영태, 이강로, "ID3 계열의 귀납적 기계 학습", 정보과학회지, 제13권 제5호, pp.6-19, 1995.
- [3] Agnar Aamodt and Eentic Plaza, "Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches", AI Communications, Vol.7, No. 1., pp.39-52, 1994.
- [4] Carbonell, J. G., "Learning by analogy : Formulating and Generalizing Plans from Past Experience", In Machine Learning: An Artificial Intelligence Approach, eds. R.Michalski, J. Carbonell and T. Michell, San Francisco, Calif: Morgan Kaufmann, pp. 137-162. 1983.
- [5] <http://www.ai-cbr.org/theindex.htm>
- [6] Ian Watson, "Applying Case-Based Reasoning : Techniques for Enterprise Systems", Morgan Kaufmann Publishesrs, 1997.
- [7] Kolodner, J., "Case-Based Reasoning". Morgan Kaufmann, 1993.
- [8] Quinlan, J. R. "Introduction of decision trees", Machine Learning, 1(1), pp.81-106, 1983.
- [9] Richter, M.M., "The knowledge contained in similarity measures". Invited Talk the ICCBR'95, Sesimbra, Portugal, October 25, 1995.
- [10] Riesbeck, C. K., and Schank, R. S. "Inside Case-Based Reasoning", Northvale, NJ:Erlbaum, 1989.
- [11] Smyth, B, "Case-Based Design", Ph.D. Thesis, Departent of Computer Science, Trinity College Dublin, Ireland, 1996.
- [12] Wilke W., vollrath I., Althoff K.D., and Bergmann R., "A Framework for Learning Adaptation Knowledge Based on Knowledge Light Approaches", 5th German workshop on Case-Based Reasoning, GWCBR'97, 1997.