

사례기반 추론을 위한 적응 지식의 자동 학습

이 재 필[†] · 조 경 달[†] · 김 기 태^{††}

요 약

사례기반 추론은 이전에 해결된 문제에 대한 해를 재사용 함으로써 새로운 문제를 해결한다. 그러나 과거의 사례와 새로운 문제 사이에는 차이가 있다. 이러한 문제점을 해결하기 위해서 사례기반 추론 시스템은 이전 사례의 해를 새로운 상황에 맞게 적응 시켜야 한다. 최근의 사례기반 추론 시스템에서, 사례 적응은 시스템 개발자에 의해 손으로 코딩된 규칙을 사용하는 규칙기반 방법을 이용한다. 따라서 시스템 설계자는 과거의 전통적인 전문가 시스템 설계에서 발견되는 지식획득의 병목 현상에 직면하게 된다. 이러한 문제점을 해결하기 위해서 본 논문에서는 사례 베이스를 이용한 사례적응 지식의 자동학습 방법을 제안하였다. 적응 지식을 학습하기 위해 사례베이스의 사례들을 비교하는 방법을 사용하였다. 본 논문에서 제안한 방법은 여행 가격 결정을 위한 영역에서 실험하였다. 실험 결과 사례 추출만을 지원하는 시스템보다 해에 대한 정확도가 향상되었음을 확인할 수 있었다.

An Automatic Learning of Adaptation Knowledge for Case-Based Reasoning

Jae-Pil Lee[†] · Kyoung-Dal Cho[†] · Ki-Tae Kim^{††}

ABSTRACT

Case-Base Reasoning(CBR) solves the new problems by reusing the solutions to previously solved problems. But, there are differences between previously known case and a new problem. To solve this problem, Case-Based System have to adapt the solution of the case to suit a new situation. In current CBR systems, 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 akin to those found in traditional expert system design. To solve this problem, in this thesis, we present an automatic learning method of case adaptation knowledge using case base. we use a method of comparing cases in the case base to learn adaptation knowledge. The system is tested in the domain for the decision of travel-price. The result shows accuracy improvement in comparison with case retrieval-only system.

1. 서 론

인공지능의 궁극적인 목표 중의 하나는 인간의 문제 해결 과정과 유사한 기계를 만들어 내는 것이라고

할 수 있다. 이러한 문제 해결에 관한 인공지능의 초기 접근법은 전문가가 특정 영역의 지식을 규칙으로 만들어 문제를 해결하도록 하는 규칙기반 추론(Rule-Based Reasoning : RBR) 시스템이 주된 것이었다. 그러나 규칙기반 추론 시스템의 여러 문제점을 해결하기 위하여 최근에는 문제 해결에 관한 과거의 경험을 재 사용하는 방법에 관한 연구들이 많이 진행되어오고 있

† 준 회원 : 중앙대학교 대학원 컴퓨터공학과
†† 정 회원 : 중앙대학교 컴퓨터공학과 교수
논문접수 : 1998년 8월 10일, 심사완료 : 1998년 11월 4일

다[1][3][12]. 이러한 연구 중에서 사례기반 추론(Case-Based Reasoning : CBR)은 이전에 해결된 사례로부터 유사한 사례를 찾아 이 사례의 해를 적용시킴으로써 새로운 문제를 해결하는 방법이다[3][6][12]. 사례기반 추론은 지식기반의 의사 결정 지원 시스템, 예측, 진단, 계획 등 다양한 영역에서 사용되어지고 있다. 그러나 지금까지 개발된 사례기반 추론 시스템을 살펴보면 대부분의 시스템이 과거의 사례를 사례 베이스에 저장한 후, 현재 문제와 유사한 사례만을 찾아 직접 해로써 제안하거나 단순한 적응 지식을 이용하여 사례를 적용시켜 해로써 제안하는 시스템이 주를 이루고 있는 실정이다. 즉 사례기반 추론 과정에서 검색된 사례를 현 상황에 맞게 수정하는 사례 적용 단계에 대한 연구가 많이 진행되어지지 않았다. 이러한 이유는 사례 적용을 위한 지식을 영역으로부터 획득하는 과정은 과거의 지식 기반 시스템의 개발 단계에서 발생하는 지식 획득의 병목 현상(knowledge-acquisition bottleneck)과 동일한 문제를 초래하거나, 사례 적용 단계를 구현하기 어렵기 때문이었다[9][7][12]. 또한 어떤 학자들은 사례 베이스로부터 검색된 사례가 일반적으로 입력된 문제를 어느 정도 해결해주기 때문에 사례 적용이 필요 없다고 주장하기도 한다[15]. 이러한 주장에도 불구하고 대부분의 사례기반 추론을 연구하고 있는 학자들은 사례기반 추론이 좀 더 완벽하게 이루어지기 위해서는 사례 적용 과정을 포함해야 한다는 데는 거의 공감하고 있는 실정이다.

현재까지의 사례기반 추론시스템에서의 사례 베이스의 유일한 목적은 현재 문제와 유사한 해를 제공하는 것으로 사용되었다. 그러나 사례 베이스는 사례 적용 단계에서 이용 가능한 내재적인(implicitly) 적응 지식을 포함하고 있다고 볼 수 있다. 즉 대량의 데이터 베이스 내에 존재하는 데이터들 간의 상호 관련성을 조사하여 이제까지 몰랐던 새로운 정보를 추출하는 데이터 마이닝(data mining)처럼 사례 베이스 안에 있는 사례들 간의 상호 관계를 조사함으로써 추출된 사례를 수정하기 위한 적응지식을 사례 베이스 자체가 포함하고 있을 수 있다는 것이다. 이러한 가정을 토대로, 본 논문에서는 추출된 사례를 수정하기 위해 사례 베이스로부터 사례 적응지식을 자동적으로 찾아서 학습하는 방법을 제안하였고 추출된 적응 지식의 유용성을 평가하였다.

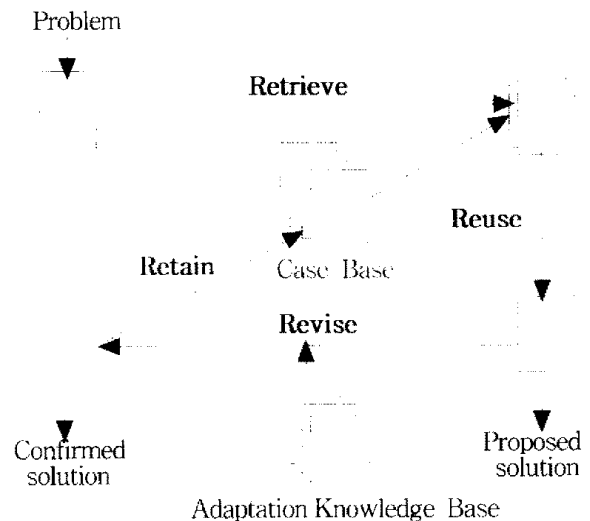
본 논문의 구성은 다음과 같다. 2장에서는 사례기반

추론의 일반적인 추론과정 및 현재까지 제안된 사례 적용 방법을 기술하였으며, 3장에서는 관련 연구를, 4장에서는 본 논문에서 제안하는 학습 방법을 기술하였다. 그리고 5장에서는 제안한 방법을 이용한 구현 및 성능 평가의 결과를 기술하였으며, 마지막으로 6장에서는 결론 및 향후 연구과제를 기술하였다.

2. 사례기반 추론(Case-Based Reasoning)

2.1 추론 과정

사례기반 추론이란 과거의 어떤 문제를 해결하기 위해 사용했던 경험을 바탕으로 새로운 문제를 해결하는 방법이라고 할 수 있다[3][6][12]. 이 추론 방법은 사람들이 문제를 해결하는 방법과 흡사하다. 예를 들어 환자를 진찰하는 의사는 이전에 이와 유사한 환자의 증상을 기억해낼 것이며, 프로그래머는 새로운 문제를 풀기 위해 이전에 코딩된 루틴을 재사용할 것이다. 사례기반 추론의 가장 큰 전제는 전혀 새로운 문제를 푸는 것보다는 현재 문제와 유사한 문제를 해결하기 위해 사용되었던 과거의 해를 이용하는 것이 문제를 해결하기 쉽다는 것이다. 사례기반 추론은 경험으로부터 추론을 수행하기 때문에 학습은 이 추론 과정에서 자연스럽게 생성되어지는 부산물로 볼 수 있으며, 지식 획득의 수단으로 사용되어 질 수 있다. 추론 과정은 정형화되어 있지 않지만 일반적인 추론 과정은 다음과 같다.



(그림 1) 사례기반추론 과정
(Fig. 1) Case-Based Reasoning Cycle

위의 그림에서 보듯 사례기반 추론 과정은 크게 4 단계(4RES)로 구분되어 있다.

(1) **REtrieve** : 이전에 경험했던 문제 중에서 현재 해결하고자하는 문제와 가장 유사한 사례를 추출해내는 단계

(2) **REuse** : (1)에서 검색된 사례를 재사용하여 새로운 문제를 해결하기 위해 시도하는 단계

(3) **REvise(adaptation)** : 검색된 사례의 해를 사용하여 새로운 문제를 해결하지 못한 경우, 새로운 문제 상황에 맞게 검색된 사례의 해 부분을 적응(수정)하는 단계

(4) **REtain** : 새로운 문제를 해결한 후 해결된 문제를 새로운 사례로서 사례 베이스에 저장하는 단계

사례기반 추론시스템의 성능은 사례의 범위, 즉 시스템에 저장된 사례가 새로운 문제를 해결하는데 얼마나 유용하게 사용되느냐에 의존적이라고 할 수 있다. 일반적으로 새로운 문제와 이전의 사례 사이에는 차이가 존재하기 때문에 사례의 적용 범위는 추출된 사례의 해를 새로운 문제 상황에 맞도록 적응(수정)할 수 있으나 시스템의 정확도가 좌우된다고 할 수 있다. 현재까지 적응은 적응 규칙을 획득하고 적용하기 위한 방법이 많지 않아서 사례기반 추론 분야에서 그다지 성공적인 연구가 진행되어 있지 않다. 적응을 수행하는 대부분의 사례기반 추론 시스템은 적응 규칙을 손으로 작성하므로 전통적인 지식기반 시스템에서 직면하게 되는 것과 유사한 지식획득의 병목현상에 처하게 된다. 사례기반 추론시스템이 적응을 수행하려 한다면 적응 규칙을 적은 비용으로 획득하는 방법이 필수적인 요인이 된다.

이러한 이유 때문에 사례기반 추론에서 사례 표현, 유사도 측정, 사례 추출과 같은 이론은 상당한 연구 및 발전이 이루어져 있으나, 사례 적응 분야는 연구가 별로 이루어져 있지 않으며 사례기반 추론에서 가장 어려운 분야로 인식되어오고 있다. 2.2절에서는 현재까지 제안된 사례 적응 방법에 대해서 기술한다.

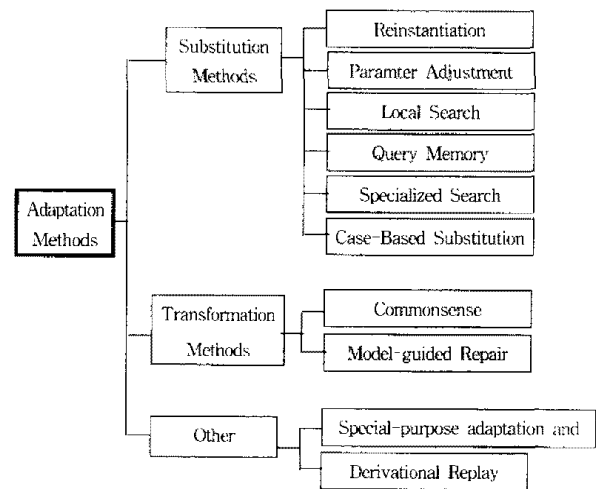
2.2 사례 적응 방법의 분류

현재까지 제안된 사례기반 추론 시스템의 적응 방법은 Riesbeck and Schank, Hinrichs, Kolodner과 Wilke등에 의해서 분류되었다[2][3][6][7]. 내용은 거의 비슷하지만 여기에서는 Kolodner에 의한 분류법을 소

개하고자 한다. Kolodner는 (그림 2)와 같이 적응 방법을 치환(substitution), 변환(transformation) 및 기타의 3종류로 구분하였다[6].

- **치환(substitution) 방법** : 이 방법은 추출된 사례의 해의 일부분을 주어진 문제 상황에서 요구되는 새로운 값으로 치환하는 방법이다. 이 방법을 사용하고 있는 대표적인 시스템에는 CHEF, JUDGE 등이 있다.

- **변환(transformation) 방법** : 치환 방법은 추출된 사례의 해 중 부적절한 값에 치환될 항목이나 개념이 이미 존재하는 경우에 적합하다. 그러나 요구되어지는 항목이 존재하지 않는다면 사용되어질 수 없다. 이 경우 부적절한 해의 일부분을 현재의 문제 상황에 맞게 변환 시켜나아가야 한다. 이 방법을 사용하고 있는 대표적인 시스템에는 CASEY, JULIA 등이 있다.



(그림 2) 적응 방법의 분류
(Fig. 2) Taxonomy of adaptation methods

- **기타** : 이 분류에 포함되는 적응 방법 중에는 특별한 목적을 위한 적응 방법 및 유도 유추에 의한 방법이 포함된다. 이 방법을 사용하고 있는 대표적인 시스템에는 PRODIGY, BOGART 등이 있다.

3. 관련 연구

3.1 CARMA(Case-based-Range Management Adviser)

CARMA는 곤충 통제를 위한 생물학적인 예측을 위한 사례기반 예측 시스템이다[14]. CARMA에서 사례

베이스는 영역으로부터 추출된 적용 규칙의 정체를 위해 사례 베이스를 이용하였다. 이전의 사례는 사례 베이스에 저장되고, 곤충의 특성이나 발육 과정 등에 대한 일반적인 영역 지식은 영역 전문가에 의해 적응 지식으로 표현된다. CARMA는 영역으로부터 추출한 사례 적용지식의 가중치 값을 언덕오르기 알고리즘을 이용하여 사례 베이스로부터 학습한다. 이렇게 학습된 가중치 값은 사례 적용 시에 문제 사례와 가장 유사한 사례의 해를 적용시키는데 이용된다. CARMA에서는 적응 규칙은 손으로 코딩되며, 이 적응 규칙을 적용할 때 어떤 항목에 대해서 가중치를 부여할 것인가를 사례 베이스로부터 학습하는 것이다.

3.2 DIAL(Disaster response with Introspective Adaptation Learning)

DIAL은 지진, 화공 약품의 누출 등과 같은 자연과 인간에 의해 야기되는 재난에 대한 대처하기 위한 계획 지원 시스템이다[4][5]. 이 시스템에서의 사례는 과거의 재난에 대처한 계획들이다. DIAL은 두 종류의 사례, 즉 일반적인 사례와 추출된 사례의 적응 과정을 기록해 놓은 적응 사례를 저장하고 있다. 또한 영역 지식으로부터 추출한 적응 규칙을 가지고 있다. 새로운 문제가 입력되면 현재의 문제와 가장 유사한 사례를 일반적인 사례 베이스로부터 추출하고, 추출된 사례를 적용시키기 위해 적응 사례들을 조사한다. 만약 현재의 문제와 유사한 적응 사례를 적응 사례로부터 추출할 수 있으면 이 적응 사례를 이용하여 현재의 문제를 적용시키고, 그렇지 않을 경우는 영역으로부터 추출한 적응 규칙을 사용하여 적응을 수행한다. 즉 DIAL은 적응 규칙과 적응 사례를 이용하여 추출된 사례의 적응을 수행하는 시스템이다.

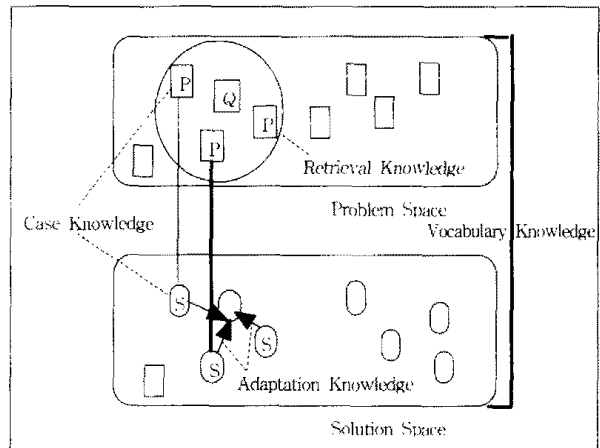
위에서 기술한 관련 연구는 모두 손으로 작성된 사례 적용 지식을 이용하며, 사례 베이스를 적응 지식을 획득하는 데 직접적으로 사용하지 않는다. 다만 좀 더 정확한 사례 적용을 하기 위해 사례 베이스를 부수적인 목적으로 이용하기만 한다.

4. 적응 규칙의 학습

4.1 지식 컨테이너

Richter는 사례기반 시스템에서 지식을 포함하고 있는 네 개의 컨테이너(container)를 (그림 3)과 같이 정

의하였다[10]. 어휘 지식(vocabulary knowledge)은 전체 영역을 묘사하기 위해 사용되는 속성(attributes), 시술(predicates)등과 같은 어휘에 포함된 지식이며, 사례 지식(cases knowledge)은 사례 베이스 안에 있는 사례들이 가지고 있는 지식이다. 그리고 추출 지식(retrieval knowledge)은 유사 사례 추출 시에 유사도 측정 등을 위해 사용되는 지식이며, 적응 지식(adaptation knowledge)은 추출된 사례의 해를 변환시키기 위해 사용되는 지식이다. (그림 3)은 위와 같은 지식이 사례 기반 추론 단계에서 어떻게 사용되어지는 가를 설명한 것이다.



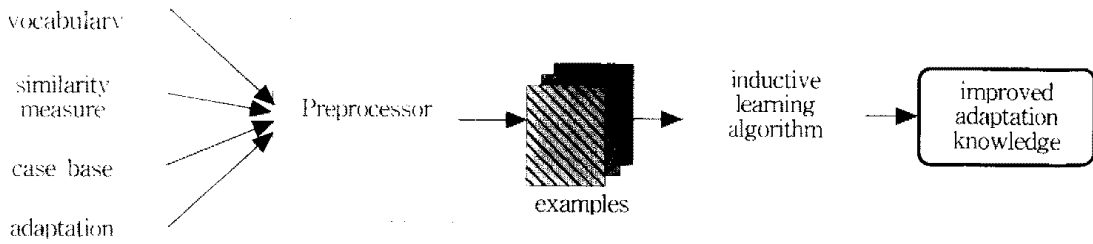
(그림 3) 사례기반 추론 과정과 사용된 지식
(Fig. 3) The CBR process and used knowledge

사용자에 의해 질문(Q)이 입력되면 추출 지식을 사용하여 사례 베이스로부터 가장 유사한 사례(S)를 추출한다. 사례 지식은 이미 알려진 문제와 그에 대응하는 해로써 사례 베이스에 표현되어 있다. 시스템은 새로운 문제를 해결하기 위하여 이렇게 추출된 과거 사례의 해를 적응 지식을 이용하여 적용시킨다. 이러한 적응 지식을 이용한 사례의 적응은 과거 사례의 적용 범위를 넓혀주는 역할을 수행한다.

4.2 학습 골격

Wilke는 Richter가 정의한 네 개의 지식 컨테이너로부터 적응 지식을 학습하기 위한 골격을 다음과 같이 제안하였다[8].

이 골격에서는 Richter가 정의한 네 개의 지식 컨테이너의 지식을 전처리 과정을 거쳐 적절한 표현으로 변환다음, 이 데이터를 귀납적 알고리즘을 사용하여



(그림 4) 적응 지식 학습을 위한 골격
(Fig. 4) A framework for learning adaptation knowledge

적용 지식을 추출한다. Wilke는 "어려 지식 컨테이너로부터 유도된 지식을 어떻게 통합할 것인가?", "지식 컨테이너의 지식을 어떻게 전처리 할 것인가?" 등에 관한 여러 가지 문제점을 향후 연구 과제로 정의하였다.

이러한 연구 결과를 토대로 하여 본 논문에서는 4.1 절에서 기술한 지식 컨테이너에서 적응 지식이 없다고 가정하고(적용 지식을 영역으로부터 직접 추출하여 손으로 코딩하지 않음), 사례 베이스를 이용하여 전처리 과정을 거치지 않고 자동적으로 적용 지식을 학습하는 방법을 제안하였다. 본 논문에서는 적용 지식을 학습하기 위해 사례 베이스의 사례를 서로 비교한다. 이러한 사례 비교 방법은 적용 문제에 있어 비교적 비용이 적게드는 한가지 해법이라고 할 수 있다. 그러나 이렇게 추출된 지식을 유용하게 만들기 위해서는 여러 가

지 어려운 문제를 해결해야한다. 먼저 사례 비교를 통해 획득되는 관계 규칙의 수가 많을 수 있다는 것이다. 즉 연관 규칙의 수를 줄일 수 있는 방법을 찾아야 한다는 것이다. 두 번째로 사례 비교를 통해 추출된 규칙 집합을 정제화해야 한다는 것이다. 예를 들어 동일 규칙이 여러 개 존재하면, 이 규칙을 하나의 규칙으로 만들어야 하며, 규칙의 전체부는 동일하고 결론부가 다른 경우 이것을 하나의 규칙으로 만들어야 한다. 본 장에서는 이러한 문제점에 대한 해결 방법에 관한 사항들을 기술한다.

4.3 사례 베이스를 이용한 적응 지식의 학습

사례 베이스는 영역 지식에 대한 많은 내재적인 지식을 가지고 있다. 본 논문에서는 사례 베이스를 이용

Feature	Values
JourneyCode	1
HolidayType	Active
NumberOfPersons	2
Region	Egypt
Transportation	Plane
Duration	14
Season	June
Accommodation	TwoStars
Hotel	"Hotel White House, Egypt".
Price	\$1978

Feature	Values
JourneyCode	2
HolidayType	Bathing
NumberOfPersons	2
Region	Egypt
Transportation	Plane
Duration	14
Season	April
Accommodation	TwoStars
Hotel	"Hotel White House, Egypt".
Price	\$2498

Feature	Values
JourneyCode	3
HolidayType	City
NumberOfPersons	2
Region	Cairo
Transportation	Plane
Duration	7
Season	April
Accommodation	ThreeStars
Hotel	"Hotel Victoria, Cairo".
Price	\$1978

Feature	Values
JourneyCode	4
HolidayType	City
NumberOfPersons	2
Region	Cairo
Transportation	Plane
Duration	7
Season	April
Accommodation	ThreeStars
Hotel	"Hotel New Marwa Palace, Cairo
Price	\$2308

(그림 5) 여행 가격 결정 사례의 예
(Fig. 5) The case examples of travel price decision

하여 적응 지식을 학습하는데 중점을 두고 있다. 두 사례 베이스의 사례를 서로 비교함으로써 서로의 연관 관계를 찾아 적응 지식 베이스를 구성하여 이것을 사례 적용 시에 이용하는 것이다.

예를 들어 그림 5와 같은 사례가 사례 베이스에 있다고 가정하면, 사례 1과 사례 2를 비교하여 그림6에 있는 규칙 1을 얻을 수 있고, 사례 3과 사례 4를 비교하면 규칙 2를 얻을 수 있다.

규칙 1 : 만약 Holidaytype이 Bathing에서 active로 변하고, Season이 April에서 June로 변경되면 여행 경비는 \$520감소한다.

규칙 2 : 만약 Hotel이 "Hotel New Marwa Palace, Cairo"에서 "Hotel Victoria, Cairo"로 변경되면 여행 경비는 \$330감소한다.

(그림 6) 사례베이스로부터 생성된 규칙의 예 (Fig. 6) The examples of rules produced from case base

위에서 획득된 적응 규칙 1과 2는 적응 규칙 베이스(adaptation rule base)에 포함된다. 만약 적응 규칙 베이스가 규칙 1과 2만을 포함하고 있을 경우 다음과 같은 문제 사례가 입력되어 사례16이 가장 유사한 사례로 추출되어진 경우를 가정해보자.

문제와 추출된 사례16은 HolidayType, Season, Hotel에 대한 특성들이 차이가 있으므로 이것에 대해 사례 적용 단계에서는 사례 16의 여행가격을 적절히 조정하여야 한다. 이것을 하기 위하여 적응 지식 베이스 중 이러한 차이점에 대하여 적용할 수 있는 모든

Feature	Values
JourneyCode	Problem1
HolidayType	Active
NumberOfPersons	2
Region	Cairo
Transportation	Plane
Duration	7
Season	June
Accommodation	ThreeStars
Hotel	"Hotel Victoria, Cairo".
Price	?

(그림 7) 문제와 추출된 사례의 예 (Fig. 7) An example of problem and retrieved case

규칙은 찾아낸다. 여기에서는 규칙1과 규칙2가 적용 가능한 규칙으로 선정되어지고 먼저 규칙1을 적용하여 사례16의 여행 가격을 \$2798(3318에서 520을 뺀 값)로 조정한다 다음, 적용 가능한 규칙은 규칙2이므로 2798에서 330을 뺀 값인 \$2468로 사례 16의 여행 가격을 조정하여 문제의 해로 정한다. 이와 같은 적응 지식을 사례 베이스로부터 학습하기 위한 알고리즘은 다음과 같다.

```

Procedure Generate_adaptation_rules(CASE_BASE)
'적용 규칙을 저장할 변수
Adapt_Rule_Set = {}
'한개의 규칙을 저장할 변수
Single_Rule_Set = {}
'사례베이스의 사례를 서로 비교한다.
FOR I = 1 TO CASE_NUMBER - 1
    Problem = CASE_BASE(I)
    FOR J = I + 1 TO CASE_NUMBER
        Target = CASE_BASE(J)
        'Problem과 Target사이의 특성차이의 개수인 계산한다.
        Diff = Calculate_DiffofAttributes(Problem,Target)
        '계산된 특성차이의 개수가 설정 임계값보다 작으면 해당 적응
        규칙 생성
        IF Diff <= THRESHOLD THEN
            Single_Rule = Make_adapt_rule(Problem,Target)
            Adapt_Rule_Set = Single_Rule ∪ Adapt_Rule_Set
        END IF
    NEXT J
NEXT I
'획득한 적응 규칙을 정제한다.
Refine_adapt_rules(Adapt_Rule_Set)
    
```

(그림 8) 적응 규칙 생성 알고리즘 (Fig. 8) Adaptation rule generation algorithm

위 알고리즘에서 보듯 적응 규칙을 학습하기 위해서 사례 베이스에 있는 모든 사례를 비교하여 적응 규

Feature	Values
JourneyCode	16
HolidayType	Bathing
NumberOfPersons	2
Region	Cairo
Transportation	Plane
Duration	7
Season	April
Accommodation	ThreeStars
Hotel	"Hotel New Marwa Palace, Cairo
Price	\$3318

적용 추출하는 것이 아니다. 즉 적용 규칙의 생성을 위한 비용을 줄이기 위하여 비교 사례 중 특성의 차이의 수가 임계값 보다 낮은 사례만을 비교하여 적용 규칙을 만들어 낸다. 사례 추출기에 의해 매우 유사하다고 판단되는 사례들 사이의 차이를 연결시켜주기 위해 본 논문에서는 임계값을 사용하였다. 임계값에 대한 결정은 5.3절의 평가 부분에서 설명하였다. Make_adapt_rule 함수는 4.3절에서 기술한 것처럼 인자로 전달받은 사례간의 속성 차이를 계산하여 규칙의 전체부를 구성하고, 사례가 지장하고 있는 해의 차이를 계산하여 해당 규칙의 결론부로 만든 후, 그 규칙을 반환하는 함수이다. 이렇게 추출된 사례 적용 규칙들은 서로 중복적인 규칙들을 포함할 수 있기 때문에 적용되기 위해서는 적용 규칙의 중복 규칙을 없애야 한다.

4.3.1 중복 규칙의 제거

사례 비교를 통하여 획득된 규칙은 중복된 규칙을 포함할 수 있다. 중복에 대한 정의는 적용 영역에 따라 다양할 수 있지만 이 논문에서는 숫자(여행 경비)를 해로써 갖는 영역을 고려하였기 때문에 영역에서 추출될 수 있는 중복 규칙을 크게 두 가지로 분류할 수 있다.

첫째, 전체부 및 결론부가 동일한 동일 규칙의 중복이 중복될 경우이다. 이 경우에는 이 규칙을 하나의 규칙으로 만들며 이 규칙에 대한 확신도 값을 다음과 같은 식으로 계산하여 할당한다. 아래 식에서 해당 규칙의 빈도수는 추출된 규칙 중 전체부와 결론부가 동일한 규칙이 발생한 빈도수를 나타내며, 최대 빈도수는 각 규칙의 빈도수 중 가장 큰 값을 취하였다. 규칙의 확신도는 적용 규칙의 충돌이 발생한 경우 확신도가 높은 값을 먼저 적용하기 위해 계산된다.

$$\text{확신도} : \text{해당 규칙의 빈도수} / \text{최대 빈도수}$$

두 번째로 전체부가 같고 결론부가 다른 규칙이다. 이러한 규칙들은 해당 규칙들의 결론부의 평균값을 취하여 결론부를 만들어 하나의 규칙으로 만든다.

4.3.2 적용 규칙의 적용

현재의 문제와 가장 유사한 사례가 추출되면 이 사례는 위에서 획득된 적용 규칙을 적용하여 사례의 결론부를 적용시켜야 한다. 이러한 적용 규칙을 적용하기 위한 알고리즘은 다음과 같다.

```

Procedure Apply_adapt_rules(P_Case, R_Case, Adapt_Rule_Set)
  'P_Case : 문제 사례, R_Case : 사례 베이스로부터 추출된 사례
  Apply_rule={}
  '문제 사례와 추출된 사례의 특성 차이를 찾는다.
  Diff = Cal_DiffofAttributes(P_Case, R_Case)
  '적용 규칙에서 적용 가능한 규칙을 모두 찾는다.
  Apply_rule = Select_allApplyrules(Diff, Adapt_Rule_Set)
  '적용 가능한 규칙을 추출된 사례의 해에 적용하여 새로운 해를 구한다.
  New_Solution = Apply_adapt_rules(R_Case, Apply_rule)
  '새로운 해를 문제 사례의 해로 제안한다.
  Propose_Newsolution(New_Solution, P_Case)
    
```

(그림 9) 적용 규칙의 적용 알고리즘
(Fig. 9) Adaptation rule application algorithm

새로운 문제가 시스템에 입력되면 시스템은 사례 베이스로부터 가장 유사한 사례를 추출한다. 그런 다음 추출된 사례와 입력된 문제간의 특성 차이를 측정하여 추출된 사례의 해를 적용시킨다. 먼저 추출된 사례와 문제간의 차이를 계산한다. 그런 다음 적용 규칙 베이스를 탐색하여 추출된 사례와 입력된 문제 사이의 차이점을 하나라도 해결할 수 있는 모든 규칙을 찾아낸 후, 이 규칙들을 적용하여 새로운 해를 찾는다. 규칙의 적용 순서는 확신도의 값이 높은 규칙을 먼저 적용한다.

5. 구현 및 평가

본 논문이 제안하는 방법의 프로토타입을 Windows95 환경 하에서 Visual Basic 5.0을 사용하여 구현하여 성능을 평가해 보았다.

5.1 구현

본 논문에서 선택한 영역은 여행 경비를 산출하는 여행 정보를 대상으로 하였다. 각 사례는 10개의 특성 값을 가지고있다. 시험 자료로 사용한 사례는 AI-CBR Group에서 제공하는 Web상의 Case-Base Archive의 Travel Agents Case-Base로부터 획득하였다[16]. 제공된 사례의 총 수는 모두 1,469개며 1,269개를 사례 베이스로 구축하였으며, 200개는 본 논문에서 제안한 방법을 평가하기 위해 임의로 추출한 평가 사례이다.

본 논문을 평가하는 데 있어 기본 가정은 사례 베이스에 있는 사례의 결론이 정확하다고 가정하였다. 사례 베이스는 계층 구조가 아닌 평면 메모리(flat memory)로 구성하였다. 또한 사례 추출을 위한 알고

리즘은 최대 근접 추론(Nearest-Neighbour Retrieval)을 사용하였으며, 이를 위한 유사도 함수는 다음의 같다.

$$Similarity(T, S) = \sum_{i=1}^n f(T_i, S_i) \times W_i$$

T : 문제 사례
S : 사례 베이스의 사례
n : 각 사례의 특성 갯수
i : 1에서 *n*까지의 특성 갯수
f : 유사도 함수
w : 특성에 대한 가중치

(그림 10) 유사도 함수
(Fig. 10) Similarity function

5.2 평가

구현된 프로토타입 시스템에 테스트 문제를 입력한 후 3가지 방법으로 출력 결과의 정확도를 비교하였다. 첫 번째 방법(평가 1)은 사례 베이스로부터 추출된 적응 지식이 얼마나 신뢰성 있는 지식인가를 평가해보기 위해 사례 적응을 하지 않는 시스템과 비교하였고, 두 번째는(평가 2) 적응 지식을 추출할 때 사례들 간의 특성차이의 개수를 나타내는 임계값을 특성 수에 따라 다양하게 변화시켜가며(2, 4, 6, 8) 테스트하였고, 마지막으로(평가 3) 사례 특성의 손실 값을(5%, 10%)로 정하여 손실 값이 추출된 적응 규칙의 정확도에 어느 정도의 정확도를 미치는지를 평가하였다.

위와 같이 평가 방법을 결정한 이유는 사례 베이스로부터 구축한 적응 지식이 문제를 해결의 정확도를 높이는 데 얼마나 기여하는가를 측정하기 위해서다. 시스템이 제공한 해와 전문가가 제공한 해의 에러율은 아래와 같은 식을 이용하여 계산하였다.

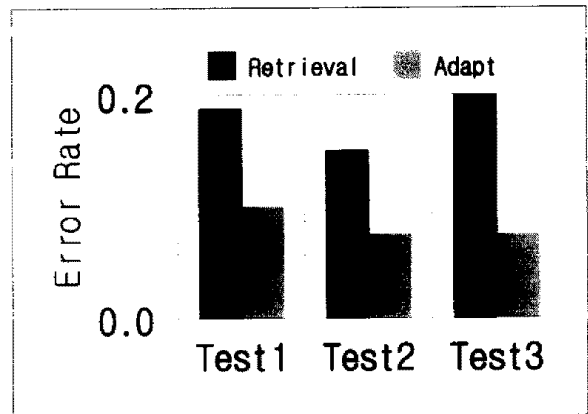
$$\frac{\sum_{i=1}^N |Solution[i] - Esolution[i]|}{N}$$

위 식에서 N은 테스트 사례의 총 개수를 의미하며, Solution은 구현된 프로토타입 시스템의 해를 나타낸다. 그리고 Esolution은 전문가의 해, 즉 테스트 사례에 있는 해를 나타낸다. 만약 프로토타입 시스템이 출력한 결과가 전문가가 예측한 결과와 거의 동일한 경우에는 에러율이 0에 가까운 값을 나타낼 것이다. 평가는 평가 사례가 다른 3개의 테스트 집합을

각각 3번에 걸쳐 수행되었으며, 다음절에서는 위에서 기술한 3가지 평가 방법에 대한 결과를 그래프로 나타낸 것이다.

5.2.1 평가 1

평가 1은 사례 베이스로부터 추출된 적응 지식이 얼마나 신뢰성 있는 지식인가를 평가해보기 위해 사례 적응을 하지 않는 시스템과 비교한 것이다. 1,269개의 사례를 이용하여 사례 특성의 임계값을 3으로 설정하여 적응 지식을 추출한 결과 13,710의 규칙이 생성되었다. 이렇게 생성된 규칙중 위에서 설명한 방법을 이용하여 중복 규칙을 제거한 후 평가하였다.



(그림 11) 평가1의 결과
(Fig. 11) Result of evaluation 1

위 그림은 적응 규칙을 이용하여 해를 제안 한 경우, 정확도의 향상을 보여주고 있다. 이 실험 결과 적응 규칙을 사용한 경우 정확도의 향상은 약 6%에서 8%까지 향상되었음을 나타낸다. 이것은 사례 베이스로부터 획득한 적응 지식이 문제 해결의 정확도를 향상시키는 데 도움을 줄 수 있음을 의미한다.

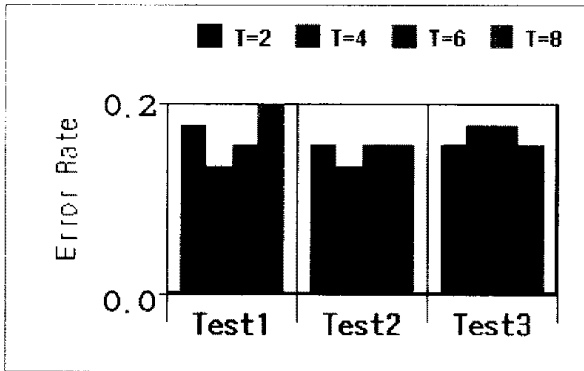
5.2.2 평가 2

평가 2는 적응 지식을 추출할 때 사례들간의 특성차이의 개수를 나타내는 임계값을 특성 수에 따라 다양하게 변화시켜가며(2,4,6,8) 테스트하였다. 이렇게 테스트 한 이유는 사례 비교를 통하여 적응 지식을 추출할 때 임계값을 어떻게 설정하여 규칙을 추출하는 것이 시스템의 정확도 향상에 더 높은 기여를 하는지를 알아보고자 한 것이다. 다음 표는 임계값의 변화에 따라 추출된 적응 규칙의 개수를 나타내고 있다.

〈표 1〉 임계값과 생성된 규칙의 수
 〈Table 1〉 Threshold and number of produced rules

임계값	추출된 규칙의 수
2	1,778
4	64,020
6	387,290
8	739,033

〈표 1〉에서 각 임계값이 클수록 생성되는 규칙의 수가 많아지며, 그 결과 규칙을 추출하기 위한 비용은 상당히 증가함을 확인할 수 있었으며, 다음 그래프는 평가 2에 대한 결과를 나타낸 그래프이다.

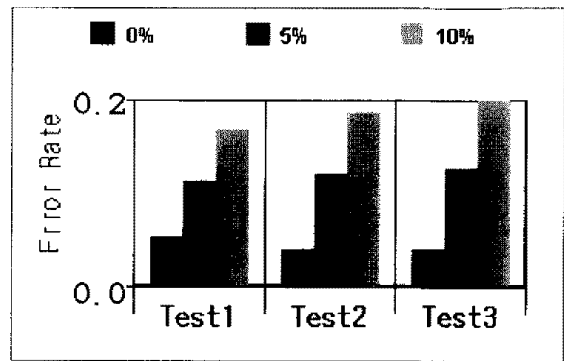


(그림 12) 평가 2의 결과
 (Fig. 12) Result of evaluation 2

위 그림의 결과를 살펴보면 임계값을 얼마로 정하느냐에 따라 시스템의 정확도의 변화는 심하지 않았다. 이러한 이유는 사례 베이스로부터 추출된 사례는 입력 사례와 유사한 사례를 해로써 제안하고, 사례 적용은 추출된 사례에 대해서 수행하기 때문에 사례 베이스로부터 추출된 적용 지식 중 실제 사례 적용에 이용되는 규칙은 특성 차이가 적은 부분에만 적용되기 때문이라고 생각된다. 따라서 평가2의 실험 결과로부터 적용 규칙의 생성을 위한 임계값의 결정은 사례 베이스로부터 추출된 사례와 입력된 문제사이의 특성 차이의 개수를 파악하여 결정하는 것이 타당하다는 것을 알 수 있다. 논문에서 테스트한 영역에서는 입력 사례에 추출된 사례 사이의 특성 차이는 최대 3이었으므로 임계값을 3으로 결정하였다. 이렇게 함으로써 얻을 수 있는 이익은 임계값을 높이면 사례 베이스로부터 규칙을 추출하는데 드는 비용이 증가하기 때문에, 임계값을 입력 사례와 추출된 사례간의 특성 차이의 수로 정함으로써 적용 규칙 생성을 위한 비용을 줄일 수 있다.

5.2.3 평가 3

영역으로부터 획득된 사례에서 사례의 특성을 나타내는 속성값이 없는 경우가 종종 존재한다. 사례 특성의 손실률을 각각 5%와 10%로 정하여 손실률이 추출된 적용 규칙의 정확도에 어느 정도의 정확도를 미치는지를 평가하였다. 평가 1은 사례 특성의 손실률이 0%이고, 이 결과와 특성 값의 손실 값을 5%와 10%를 주고 정확도에 어떠한 영향을 끼치는 지를 보았다. 이 평가에서의 임계값은 3으로 하였고, 다음 그래프는 이에 대한 결과를 나타낸 것이다.



(그림 13) 평가 3의 결과
 (Fig. 13) Result of evaluation 3

이 그래프의 결과를 보면 사례 특성값의 손실률이 높을수록 에러율이 높게 나타나는 것을 볼 수 있다. 사례 베이스의 구성 시 사례를 구성하는 특성 값이 정확히 파악이 안되거나 모르는 경우가 있다. 이러한 사례는 사례 자체의 신뢰성을 떨어뜨리는 요인으로 작용한다. 이러한 사례의 특성 손실값을 어떤 값으로 정할 것인가 하는 것은 또 다른 문제이다. 다만 여기에서는 사례 자체의 신뢰성이 없을 경우 이것을 이용하여 추출된 적용 지식에도 영향을 미쳐, 해의 정확도를 떨어뜨릴 수 있는 요인으로 작용할 수 있다는 것이다. 그래프에 나타난 것처럼 손실률이 5%인 경우는 사례 추출만을 지원하는 방법과 에러율이 비슷하나 손실률이 10%인 경우는 정확도가 약 8%에서 11%정도 떨어짐을 알 수 있다.

6. 결 론

본 논문에서는 사례 베이스로부터 사례 적용 지식을 획득하는 방법을 제안하였다. 현재까지의 사례기반 추론시스템에서의 사례 베이스의 유일한 목적은 현재

문제와 유사한 해를 제공하는 것으로 사용되었다. 그러나 사례 베이스는 사례 적용 단계에서 이용 가능한 내재적인 적용 지식을 포함하고 있다. 즉 대용량의 데이터 베이스에서 연관 규칙을 찾아내 의사결정 시 데이터 마이닝을 통하여 획득한 지식을 이용하듯이 사례 적용의 문제도 이와 유사하게 사례 베이스에서 사례들 간의 특성의 차이가 사례의 해에 어떠한 영향을 미치는지를 찾아낼 수가 있다. 본 논문에서는 이러한 사례들 사이에 포함된 내재적인 지식을 추출하기 위하여 사례 비교를 통하여 적용 규칙을 획득하였다.

사례 비교를 통하여 획득된 규칙이 문제 해결의 정확도를 향상시키는데 얼마나 유용한가를 평가해보았다. 평가 1에서는 기존의 대부분의 사례 기반 추론 시스템에서 사용하는 사례 추출만을 지원하는 방법과 본 논문에서 제안한 방법을 사용하여 획득된 사례 적용 지식을 사례 적용 시 이용하는 방법을 비교한 결과 약 8%의 정확도 향상을 가져올 수 있었다. 평가 2에서는 적용 지식 획득 시 특성 비교의 임계값을 어떻게 결정할 것인가에 대한 실험으로, 이 실험을 통하여 임계값은 추출된 사례들과 문제 사례들 간의 특성 차이의 최대값으로 결정하는 것이 적용 지식을 획득 시 비용을 줄일 수 있다는 것을 확인할 수 있었다. 평가 3에서는 사례 특성 값의 손실률이 적용 지식의 정확도에 어떠한 영향을 미치는지를 알아보았다. 이 실험 결과 특성 값의 손실률이 높아질 수록 사례 적용 지식의 정확도를 하락시키는 요인으로 작용한다는 것을 확인할 수 있었다.

본 논문에서 제안한 방법은 기존 사례 기반 추론 시스템처럼 사례 적용 지식을 시스템 개발자가 직접 손으로 코딩하거나, 사례 적용 지식을 구축하기 위해 별도로 영역 전문가와 상의하지 않고, 사례 베이스를 통하여 사례 적용 지식을 구축할 수 있는 방법을 제공하고 있다.

추후 연구 과제로는 사례 베이스로부터 추출한 사례 적용 지식은 완전한 사례 적용 지식이 아니다. 왜냐하면 실세계에 존재하는 많은 영역 지식이 있기 때문이다. 따라서, 이러한 영역 지식과 자연스럽게 결합할 수 있는 방법과, 본 논문에서 테스트한 사례는 결론이 숫자로 이루어져 있어 사례들 간의 특성 차이를 통하여 쉽게 규칙으로 표현할 수 있었다. 그러나 많은 영역에서 사례의 결론은 숫자가 아닌 문자로 이루어져 있다. 따라서 사례 적용 지식 획득 시 이러한 문자들

간의 의미는 어떻게 파악할 것인가 하는 점이다. 또한 사례 적용을 통하여 제안된 결론이 거부될 경우 어떻게 전문 사용자의 지식을 사례 적용 지식으로 활용할 것인가 하는 것이다.

참 고 문 헌

- [1] Carbonell, J.G., "Derivational Analogy and Its Role in Problem Solving," In Proceedings of the 3rd Annual National Conference on Artificial Intelligence, AAAI-83, Washington, D.C., USA, August 1983, AAAI, Morgan Kaufmann Publishers.
- [2] Hinrichs T., "Problem solving in Open worlds:A Case Study in Design," Northvale, NJ:Erlbaum, 1992.
- [3] Riesbeck C. and Schank R., "Inside Case-Based Reasoning," 1993.
- [4] David B. Leake, Andrew Kinley, and David Wilson, "Learning to improve case adaptation by introspective reasoning and CBR," Proceedings of the First International Conference on Case-Based Reasoning, Sesimbra, Portugal, 1995.
- [5] David B. Leake, Andrew Kinley, and David Wilson, "Acquiring Case adaptation knowledge : A hybrid approach," Proceedings of the thirteenth national conference on artificial intelligence, AAAI press, Menlo Park, CA, 1996.
- [6] Kolodner, J., "Case-Based Reasoning," Morgan Kaufmann, 1993.
- [7] Wilke W. and Bergmann R., "Techniques and Knowledge used for adaptation during Case-Based Problem Solving," On the 1th International conference on Industrial and Engineer Application of Artificial Intelligence and Expert Systems, IEA'98.
- [8] 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.
- [9] Feigenbaum, E. and McCorduck, P., "The fifth

Generation." Addison Wesley., 1983.

[10] Richter, M. M. "The knowledge contained in similarity measures." Invited Talk the ICCBR'95 in Sesimbra, Portugal, October 25, 1995.

[11] Lenz, M. "Cabata a hybrid cbr system." In Althoff, K. D., Richter, K., & Wess, S.(Eds), Proceedings of the First European Workshop on Case Based Reasoning, pp.204-209 Kaiserslautern, 1993.

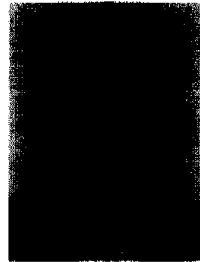
[12] Ian Watson, "Applying Case-Based Reasoning : Techniques for Enterprise Systems," Morgan Kaufmann Publishers, 1997.

[13] Agnar Aamodt, Enric Plaza, "Case Based Reasoning : Foundational, Methodological Variations," and System, AICom-Artificial Intelligence Communications, Vol.7, No.1, 1994.

[14] John D. Hastings, L. Karl Branting, Jeffrey A. Lookwood, "Case adaptation using an incomplete causal model," Proceedings of the First International Conference on Case-Based Reasoning, Sesimbra, Portugal, 1995.

[15] Mark, W., Simoudis, E., and Hinkle, D., "Case -Based Reasoning : Expectations and Results. In Case-Based Reasoning : Experiences, Lessons, & Future Directions, 267-294, edited by D.B.Leake. Cambridge, MA:AAAI Press/MIT Press, 1996.

[16] <http://www.surveying.salford.ac.uk/ai-cbr-mirror/cases.html>



이재필

e-mail : jplee@ailab.cse.cau.ac.kr

1993년 2월 중앙대학교 전자계산학과 졸업(이학석사)

1995년 2월 중앙대학교 컴퓨터공학과 졸업(공학석사)

1995년 3월~현재 중앙대학교 컴퓨터공학과 박사과정

관심분야 : 사례기반 추론, 기계학습, 데이터 마이닝, 전자 상거래 등



조경달

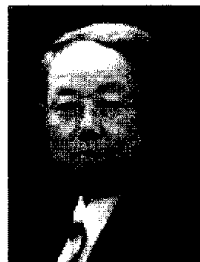
e-mail : kdcho@ailab.cse.cau.ac.kr

1990년 2월 경기대학교 전자계산학과 졸업(공학사)

1992년 2월 중앙대학교 전자계산학과 졸업(공학석사)

1992년~현재 (주)미래산업 책임연구원

1995년 9월~현재 중앙대학교 컴퓨터공학과 박사과정
관심분야 : 퍼지이론, 데이터 마이닝, 신경망 등



김기태

e-mail : ktkim@ailab.cse.cau.ac.kr

1970년 중앙대학교 수학과 응용수학(이학석사)

1988년 숭실대학교 전자계산학과(공학박사)

1972년 중앙대학교 전자계산학과 강사

1986년~1987년 일본동경공업대학 연구교수
1991년~1992년 미국 Connecticut 주립대학 객원교수
1979년~현재 중앙대학교 컴퓨터공학과 교수
관심분야 : 컴파일러, 알고리즘, 전문가 시스템, 신경망 시스템 등