클러스터링 기법에 의한 다중 사례기반 추론 시스템

이재식* · 강자영**

Multiple Case-based Reasoning Systems using Clustering Technique

Jae Sik Lee* · Ja-Young Kang**

요 약

The basic idea of case-based reasoning is to solve a new problem using the previous problem-solving experiences. In this research, we develop a case-based reasoning system for equipment malfunction diagnosis. We first divide the case base into clusters using the case-based clustering technique. Then, we develop an appropriate case-based diagnostic system for each cluster. In other words, for individual cluster, a different case-based diagnostic system which uses different weights for attributes is developed. As a result, multiple case-based reasoning systems are operating to solve a diagnostic problem. In comparison to the performance of the single case-based reasoning system, our system reduces the computation time by 50% and increases the accuracy by 5% point.

Key words : Multiple Case-based Reasoning Systems, Clustering, Malfunction Diagnosis

* 아주대학교 경영대학 교수
** 현대정보기술

1. 서 론

사례기반 추론(Case-based Reasoning)이란 과거의 유사한 문제를 해결한 경험을 기초로 새로운 문제에 대한 해를 구하는 기법이다. 즉, 과거의 사례들을 속성들의 집합으로 표현하고 각 사례에 클래스의 이름을 붙인 후에, 이들을 가지고 새로운 사례가 어느 클래스에 속하는지에 대한 개념적 설명을 이끌어내는 감독학습(Supervised Learning) 기법 중의 하나이다[Aha and Wettschereck, 1997]. 사례기반 추론 기법은 다른 인공지능 기법들의 단점인 지식획득의 병목현상, 자동학습의 불가능 등을 해소할 수 있는 대안으로서 여러 분야에 적용되고 있다. 하지만, 사례기반 추론 기법은 인공신경망이나 의사결정트리와 같이 새로운 사례가 입력되기 전에 모든 학습과정을 끝내놓는 사전학습기법(Eager Learning Technique)들과는 달리 새로운 사례에 대한 클래스를 판정하기 위한 학습과정이 새로운 사례가 입력된 후에 시작하는 사후학습기법(Lazy Learning Technique)이다. 그러므로 사례베이스의 크기가 증가하면 수행시간의 증가, 인덱싱의 문제, 사례들간의 일관성 유지[Racine and Yang, 1996], 오류(Nosy) 데이터의 처리 등의 문제가 발생한다.

특히 문제가 되는 것은 수행시간의 증가인데, 이는 새로운 사례를 풀기 위해 사례베이스 안의 모든 사례를 검색해야 하기 때문이다[Aha, 1991]. 이러한 문제를 해결하기 위하여 인덱싱 방법이 연구되었다[Auriol et al., 1995; Fox and Leake, 1995; Marir and Watson, 1995]. 그러나 인덱싱을 통하여 검색할 속성의 수를 줄이는 것만으로는 수행시간을 만족할만하게 감소시켜주지는 못하였다.

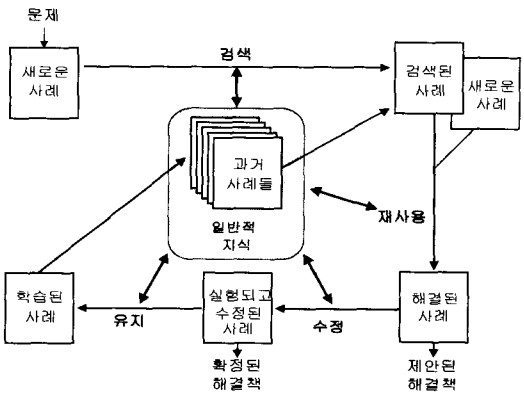
본 연구에서는 클러스터링(Clustering)이라는 자율학습기법을 이용하여 사례베이스의 사례들을 군집화 시킴으로써 대용량 사례베이스 시스템의 속도 및 정확성을 높이고자 한다. 사례베이스가 클러스터링 되어 있으면, 새로운 사례가 입력되었을 때 먼저 그 사례가 속할 수 있는 클러스터를 찾고, 그 클러스터에서만 유사한 사례들을 검색함으로써 사례기반 추론 시스템의 수행시간을 감소시킬 수 있다.

본 논문은 다음과 같이 구성되어 있다. 제 2절에서는 본 논문에서 사용한 추론 기법인 사례기반 추론과, 본 연구의 적용분야인 설비 고장진단 시스템과 사용할 사례들에 대해서 살펴본다. 제 3절에서는 본 논문에서 적용하는 클러스터링 기법인 사례기반 클러스터링에 대해서 기술한다. 제 4절과 5절은 실제 시스템의 구현과 그 평가에 관한 내용들로서, 제 4절에서는 본 논문에서 구현된 시스템의 구조와 구현 과정을 상세하게 기술하고, 제 5절에서는 구현된 시스템의 성능 평가가 제시된다. 마지막으로 제 6절은 결론으로서 본 연구가 가지는 한계점들과 향후 연구방향들에 대해서 논의한다.

2. 사례기반 고장진단 시스템

2.1 사례기반 추론 시스템의 개요

인공지능 기법의 하나인 사례기반 추론은 기억 장치에서 현재의 문제와 유사한 과거의 문제를 찾고, 과거의 문제와 현재의 문제들 간의 차이점을 분석하여 과거 문제의 해법을 현재의 문제에 알맞게 수정하여 문제를 풀어 가는 기법이다[Riesbeck and Schank, 1989]. 사례기반 추론의 기본 아이디어는 과거의 문제를 해결하기 위해 사용한 해법을 수정하여 새로운 문제의 해결에 사용한다는 것인데, 이와 같이 문제 해결 과정



〔그림 2-1〕 사례기반 추론의 순환과정

의 재사용을 통하여 자동적인 학습이 가능해진다.

사례기반 추론은 [그림 2-1]과 같이 네 단계의 순환과정을 거쳐서 수행된다.

각 단계를 간단히 기술하면 아래와 같다[Aamodt and Plaza, 1996].

검색 단계 : 새로운 사례와 가장 유사한 사례 혹은 사례들의 집합을 검색한다.

재사용 단계: 검색된 사례의 정보와 지식을 사용한다.

수정 단계 : 필요하다면 제시된 해결책을 수정한다.

유지 단계 : 미래의 새로운 문제에 대처하기 위해서 유용한 문제 풀이 경험을 유지한다.

2.2 설비 고장진단 문제

본 연구에서 사례기반 추론 시스템이 적용될 도메인은 사무 기기를 제조·판매하고, 판매된 제품에서 발생하는 고장(Equipment Malfunction)에 대해서 수리 및 부품 공급 등의 써어비스 해주는 X회사 고장진단 문제이다. 설비 고장진단의 문제는 제품의 판매뿐만 아니라 고객에 대한 사후 써어비스의 관점에서 고객만족을 향상시키는 매우 중요한 문제이다. X회사에서 설비의 고장 신고에 대해서 업무를 처리하는 과정은 다음과 같다.

고객이 구입한 설비에 고장이 발생하면, 고객은 써어비스 센터에 전화를 걸어서 고장을 신고한다. 이 때에 고객은 자신이 관찰한 설비의 이상증상을 고장접수원에게 말하며 고장접수원은 이 내용들을 고장접수 파일에 입력한다. 접수내용은 신고자의 신원과, 고장 설비의 기종, 이상 증상 등의 정보로 이루어져 있다. 이후 접수원은 지역 관할 영업소에 고객의 위치와 접수된 정보를 알려준다. 그러면, 수리 기술자가 이 정보에 기초하여 필요 도구 및 부품을 가지고 현장으로 간다. 현장에 도착한 수리기술자는 자세히 증상들을 관찰하고 기본적인 검사를 하여 추가적인 정보를 알아내고, 고장부위와 고장원인 파악하여 처치함으로써 써어비스를 마치게 된다. 이러한 전체 과정은 사후에 관리 부서에 보고되고, 사후 관리를 위해 관리 부서에 의해 정리되어, 고객명, 날짜, 기종, 이상증상, 고장원인, 고장부위, 소요시간 등 약 20여개의 속성으로 구성된 데이터베이스에 저장 유지된다.

설비 고장진단 문제에 사례기반 추론을 이용한 기존 연구들을 살펴보면, 사례기반 추론이 단독으로 사무기기의 고장진단[이재식과 전용준, 1995]과 항공기의 고장진단[Magaldi, 1994; Manago and Auriol, 1996]에 사용된 연구들이 있으며, 다른 기법들과 결합되어 하이브리드 시스템[Karamouzis and Feyock, 1993; Tsutsui et al, 1994; Watson and Abdullah, 1994; 이재식과 김영길, 1998]으로 개발된 연구들도 있다.

〔표 2-1〕 사용 속성과 설명

속 성 명	자료형	속 성 설 명
기 종	문자형	복사기의 성능·기능에 따라 분류된 종류
기계번호	문자형	제품 생산 시 부여되는 복사기의 고유 번호
미 터	수치형	복사한 용지의 누적 수량
이상증상	문자형	고장으로 인하여 나타난 이상 현상
고장원인	문자형	이상 증상이 나타나게 된 원인

연구 자료로는 X회사의 제품에 대한 써어비스 내역 중 복사기에 관한 데이터를 사용하였다. 하나의 레코드는 20여개의 속성으로 이루어져 있는데, 이 중에서 본 연구에 필요한 몇 개의 속성만을 추출하여 하나의 테이블로 만들었다. 수집된 사례의 개수는 총 6,000개인데, 이 중 5,000개의 사례는 사례베이스에 저장하였고, 나머지 1,000개는 시스템 구축 후 그 성능을 평가하기 위한 테스트 집합으로 사용하였다. 사용한 속성은 '기종(Machine Type)', '기계번호(Machine No)', '미터(Meter)', '이상증상(Symptom)', '고장원인(Cause)'이다. 각 속성에 대한 자세한 설명은 [표 2-1]과 같다.

이 속성들 중에서 '고장원인'은 종속 속성, 즉 우리가 예측하고자 하는 속성이며, 나머지 네 개의 속성은 독립 속성으로서 사례기반 추론에서 새로운 입력 사례의 속성들로 사용된다.

3. 사례기반 클러스터링

3.1 K-Means 클러스터링

클러스터 분석 또는 클러스터링이란 개체들을 특징이 정의되지 않은 집합으로 군집화 하는 것을 말한다[Afifi and Clark, 1990]. 특징이 정의되지 않았다는 뜻은 군집의 개수나 구조를 미리 정하지 않고 분석을 수행한다는 것이다[Johnson and Wichern, 1982]. 대용량 데이터베이스에는 다차원의 데이터가 많아 의미 있는 패턴을 찾아내기가 쉽지 않은데, 클러스터 분석은 이를 용이하게 해준다. 클러스터 분석은 생물학의 분류학 분야, 의학의 진단 분야에 사용되어 왔으며, 지층이나 화석의 연대를 구분하는데 사용되기도 하고, 경영학의 마케팅부분에서는 구매 고객을 유형별로 나누는데 사용되고 있다.

클러스터링은 개체들, 즉 데이터간의 거리나 유사성에 기초하여 진행된다. 이를 측정하기 위해서 두 점 사이의 거리, 두 벡터 사이의 각, 두 데이터 사이에 일치하는 속성의 수 등을 사용한다. 두 점 사이의 거리나 두 벡터 사이의 각을 이용하는 측정 방법은 정량적 데이터에 적합하고, 두 데이터 사이에 일치하는 속성들의 수를 측정하는 방법은 정성적 또는 범주형 데이터에 적합하다.

아무리 빠른 컴퓨터라도 클러스터링 할 수 있는 모든 가능성을 가늠해 보지는 못한다. 이러한 문제 때문에 모든 가능한 조건들을 고려하지 않고도 합리적인 클러스터를 찾아내기 위해서 다양한 클러스터링 알고리즘이 개발되었다. 클러스터링 알고리즘은 크게 계층형과 비계층형으로 나누어진다. 계층형 클러스터링 방법으로는 병합방법(Agglomerative Method)과 분할방법(Divisive Method)이 있는데, 클러스터간의 거리를 구하는 방법을 이용하여 계속적인 병합이나 분할에 의하여 클러스터링을 하는 방법이다. 비계층형 클러스터링은 임의로 클러스터를 나눈 후에 데이터와 클러스터간의 거리를 측정하여 가장 가까운 클러스터에 데이터를 재할당함으로써 클러스터링 작업을 수행하는 방법으로서 K-Means 클러스터링 방법이 있다.

K개의 클러스터로 클러스터링을 하는 K-Means 클러스터링 알고리즘은 아래와 같은데, 숫자 K

는 미리 정하거나 클러스터링 도중에 정할 수도 있고, K를 변화시켜 가면서 클러스터링한 다음 가장 결과가 좋은 K를 사용하기도 한다.

1단계 : 데이터를 K개의 초기 클러스터로 나누거나 또는 임의로 K개의 데이터를 초기치(seed)로 선택하여 K개의 클러스터를 만든다. 어느 데이터를 어느 클러스터에 할당하느냐는 사용자가 임의로 정하거나 특정 알고리즘에 의하여 정할 수 있다. 여기서 초기치는 클러스터링을 하기 위한 기준 데이터로서 처음의 K개의 데이터를 초기치로 쓸 수도 있고, 임의로 선택할 수도 있다.

2단계 : 각 클러스터의 중심을 구한다.

3단계 : 임의의 한 데이터를 선택하여 각 클러스터 중심까지의 거리를 계산한다. 만일, 이 데이터와 임의의 클러스터 중심까지의 거리 중 가장 가까운 것이 자신이 속한 클러스터라면 그대로 둔다. 만일 그렇지 않다면, 거리가 가장 가까운 클러스터에 재할당한다.

4단계 : 제 3단계를 모든 데이터에 대해 수행한다.

5단계 : 제 2, 3, 4단계를 모든 클러스터의 중심이 변하지 않을 때까지 계속한다.

클러스터링은 자율학습(Unsupervised Learning)의 일종으로서 결과 값이 미리 정해져 있지 않은 상태에서 수행할 수 있다. 그러므로 데이터의 내부 구조에 관한 지식이 없이도 클러스터링 기법을 사용하여 데이터의 감추어진 구조를 발견할 수 있는 것이다. 자율학습이 타율학습에 비해 가지는 장점은 사용자가 가지고 있는 주관적인 요소들을 배제할 수 있으므로 학습이 오직 정의된 학습프로세스에만 의존한다는 것이다[Yoo et al., 1996]. 즉 클러스터링은 감추어진 구조를 발견해줌으로써 타율학습의 수행능력을 개선시킬 수가 있다[De Cavalho et al., 1995] 클러스터링의 단점으로는, 거리측정 방법과 데이터 속성들의 가중치들을 정하는 표준이 없으며, 특히 K-Means 클러스터링에서는 클러스터링을 완료하여 그 결과들을 비교해 보기 전까지는 가장 적합한 K값을 알 수가 없다는 점을 들 수 있다. 하지만, 클러스터링은 용량이 크거나, 다차원이거나, 내부 구조가 복잡한 데이터 집합을 다룰 때 아주 유용한 도구이다. 그러므로 데이터 마이닝(Data Mining)을 수행할 때에 가장 기본적으로 사용되는 기법이다[Berry and Linoff, 1997].

3.2 사례기반 클러스터링 (Case-based Clustering)

본 연구에서 사용한 클러스터링 알고리즘은 사례기반 클러스터링(Case-based Clustering)으로서 K-Means 클러스터링의 변형 알고리즘이다[이재식과 안태훈, 1998]. 사례기반 클러스터링이란 클러스터링을 하기 위해 필요한 거리를 정의할 때, 사례기반 추론의 기본 개념인 부분적 매칭(Partial Matching)을 이용하는 기법이다. 이미 클러스터가 정해진 사례들을 사례베이스로 이용하여, 클러스터링을 하고자 하는 입력 사례와의 유사도가 가장 큰 사례의 클러스터 번호를 입력 사례의 클러스터 번호로 정하는 것이다. 사례기반 클러스터링의 진행 단계는 아래와 같다.

1단계 : 사례들을 임의로 k개의 클러스터로 나눈다.

2단계 : 각 클러스터 내에서 빈도순으로 최상

위 m개의 사례를 선택하여 각 클러스터의 중심으로 정한다.

3단계 : 임의의 사례를 선택하여 각 클러스터의 중심들과 유사도를 구하고 그것들을 모두 합하여 총유사도를 구한다. 모든 클러스터들과의 총유사도를 구한 다음, 총유사도가 가장 큰 클러스터가 사례가 현재 속한 클러스터와 동일하다면 그대로 두지만, 그렇지 않다면 총유사도가 가장 큰 클러스터에 현 사례를 재할당한다.

4단계 : 제 3단계를 모든 사례에 대해 수행한다.

5단계 : 제 2, 3, 4단계를 모든 클러스터의 중심이 변하지 않을 때까지 계속한다.

클러스터링을 하기 위해서는 각 클러스터의 중심을 구해야 한다. 클러스터의 중심이란 것은 본질적으로 그 클러스터 안에 있는 사례들의 특징과 성격을 반영하여 사례들을 대표할 수 있어야 한다. 일반적으로 중심은 클러스터 내의 사례들과의 분산을 최소화하는 값을 사용한다. 정량적 사례는 대체적으로 산술적인 기법을 이용하여 평균을 구하고 이를 중심으로 사용한다. 그러나, 범주형 사례는 산술적인 방법으로 중심을 구할 수가 없다. 범주형 사례의 중심을 구하는 손쉬운 방법 중 하나는 빈도수(Frequency)를 사용하는 것이다. 빈도수를 중심으로 사용할 때는 보통 최빈수를 사용하는데[Huang, 1997], 최빈수는 해당 클러스터의 특징 및 성격을 잘 반영해 주지 못할 때가 있다. 그러므로 본 연구에서 사용하는 사례기반 클러스터링 기법에서는 위의 제 2단계에서 보듯이 다수의 중심을 정하여 클러스터의 특징을 잘 반영할 수 있도록 하였다.

또한 범주형 데이터로 이루어진 사례베이스를 클러스터링할 때 간과하지 말아야 할 것 중 하나는 클러스터를 구성하는 요소들 중 의미 있는 최소단위에 대한 정의이다. 즉, 의미 있는 최소단위가 모든 속성들을 한꺼번에 고려한 하나의 사례인지, 아니면 사례를 구성하는 각 속성들인지를 명확하게 파악해야 한다. 각 속성의 값들이 서로 무수한 조합을 이루어서 하나의 사례를 만들어 낸다면 개별 속성들을 의미 있는 최소단위로 보아야 한다. 즉, 사례기반 클러스터링에서 각 클러스터 내의 중심을 정하기 위해 빈도수를 측정할 때에는 개별 사례의 빈도수를 측정하는 것이 아니라 개별 속성의 빈도수를 측정하는 것이다.

[표 3-1]은 클러스터별 '기종'의 빈도를 예시적으로 보여주고 있는데, 만일 클러스터의 중심을 최빈수 하나로 표현한다면, 'Cluster 1'의 중심은 610번의 빈도수를 갖는 '6285'이라는 기종이 될 것이다. 그렇다면 'Cluster 1'에는 600번의 빈도를 갖는 '6310'이라는 기종도 있으나 이 기종은 마치 'Cluster 1'에 없는 것처럼 인식될 수 있다. 중심이 단 하나로 표현되기 때문에 비슷한 빈도를 갖는 차선의 기종은 고려되지 않기 때문이다. 이러한 문제를 해결하기 위해서는 다수로 중심을 표현할 필요가 있다. 하지만 중심을 필요이상으로 다수로 정한다면, 클러스터링 시 정확한 유사성 측정은 어려울 것이다. 왜냐하면, 클러스터의 극히 작은 특징이 마치 그 클러스터 전체의 특징인 것처럼 나타날 수 있기 때문이다. [표 3-1]에서 중심을 7개로 표현한다면, 'Cluster 2'의 '기종' 속성의 중심은 {5252, 5235, 6035, 5230, 5100, 6315, 6280}으로 표현될 것이다. 이 중에 '6280'의 빈도수는 3번에 불과한데 이것이 중심에 포함되어 클러스터의 특징을 대표하고 있다. 이러한 중심은 클러스터의 특징을 제대로 추상화하지 못하고 있는 것이다. 이러한 문제점을 개선하기 위해 빈도수의 순위에 따라 가중치를 부여할 수도 있고, 또는 중심

〔표 3-1〕 클러스터의 중심을 구하기 위한 기종의 빈도수 비교

기종	Cluster 1	Cluster 2	Cluster 3
5100	2	13	40
5230	7	51	121
5235	2	110	8
5238			7
5250			379
5252		121	
5254	11	2	82
6035		90	12
6260	10		1
6280	199	3	12
6285	610		
6310	600	2	
6315	73	8	9

의 개수를 적당한 범위에서 한정적으로 제한할 수도 있다[이재식과 안태훈, 1998]. 본 논문에서는 중심의 개수를 실험에 의하여 2개로 정하였다. 이에 대한 실험과 평가의 내용은 제 5절에서 자세히 기술한다.

클러스터링을 하기 위해서는 사례와 클러스터 간의 거리를 측정해야 한다. 일반적으로 가장 많이 사용하는 거리 측정 방법은 유클리디안 거리(Euclidean Distance) 측정이다. 그러나 이것은 숫자형 사례의 경우에만 유용하며, 범주형 사례 간의 거리를 측정할 때는 사용할 수가 없다. 범주형 사례는 부분적 매칭 혹은 확률적 접근으로 거리를 구해야 한다. 본 논문에서 사용한 사례기반 클러스터링에서는 부분적 매칭에 의하여 거리를 측정하였다. 이에 대한 자세한 방법은 다음 절에서 기술한다.

4. 다중 사례기반 고장진단 시스템의 개발

4.1 사례베이스와 유사도 측정 기준

본 논문에서 개발한 설비 고장진단 시스템인 M-CBEMD(Multiple Case-Based Equipment Malfunction Diagnosis) 시스템은 하나의 사례베이스를 가지고 있는데 사례들은 관계형 데이터베이스의 구조로 저장되어 있다. 사례베이스에 포함되어 있는 사례의 구조는 제 2.2절에 있는 [표 2-1]의 내용과 같으며, 사례베이스에는 5,000개의 사례가 저장되어 있다. 우리가 개발한 M-CBEMD 시스템의 성능을 평가하기 위한 사례 1,000개를 별도로 준비하여 놓았는데, 이것들은 100개씩 구분해 놓은 10개의 테스트용 집합인 'TestSet1', 'TestSet2', ... 'TestSet10'에 저장되어 있다.

M-CBEMD 시스템은 크게 세 개의 모듈로 구성되어 있다. 모듈 1은 클러스터링 모듈로서 사례베이스의 사례들을 클러스터링하는 모듈이고, 모듈 2는 가중치 훈련 모듈로서 도출된 클러스터별로 그 클러스터에 포함된 사례들만을 사례베이스로 인식하는 사례기반 추론 시스템을 구축하기 위하여 속성별 가중치를 탐색하는 모듈이고, 모듈 3은 고장진단 모듈로서 새로운 고장사례가 입력되었을 때 해당 클러스터를 찾고 이로부터 기존 사례들과의 비교를 통해 가장 유사한 사례를 도출하여 고장원인을 제시하는 모듈이다. 이 세 개의 모듈이 작동하기 위해서는 공통적으로 유사도를 측정할 수 있는 기준이 마련되어야 한다. 즉, 클러스터링 모듈에서는 각 클러스터의 중심과 어떤 사례간의 유사도를 측정하는 유사도 측정 방법이, 가중치 훈련 모듈과 고장진단 모듈에서는 새로운 사례와 기존 사례들 간의 유사도를 측정하는 유사도 측정 방법이 모듈 안에 포함되어야 한다. 이를 위한 유사도 측정 기준은 모두 동일하다. 다만, 클러스터링 모듈의 경우에 중심의 개수가 복수이기 때문에 유사도 측정을 위한 계산이 중심의 개수만큼 이루어진다는 것이 차이점이라고 할 수 있다. 유

사도 측정 기준은 [표 4-1], [표 4-2], [표 4-3], [표 4-4]와 같다.

〔표 4-1〕 기종의 유사도 점수

일치 자릿수	전체가 일치	그렇지 않은 경우
유사도 점수	1	0

〔표 4-2〕 기계번호의 유사도 점수

일치 자릿수	전체가 일치	그렇지 않은 경우
유사도 점수	1	0

[표 4-3] 미터(Meter)의 유사도 점수

유사도점수

$$= 1 - \frac{|\text{새로운 사례의 Meter값} - \text{사례베이스에서 추출한 사례의 Meter값}|}{\text{사례베이스 내의 Meter의 최대값}}$$

〔표 4-4〕 이상증상의 유사도 점수

일치 자릿수	전체가 일치	2자리 일치	1자리 일치	일치 없음
유사도 점수	1	0.7	0.5	0

일치 자릿수의 비교는 왼쪽에서부터 이루어진다. 이상증상은 [표 4-5]와 같이 계층적인 구조를 보이는 코드체계(첫 자리는 대표내용, 나머지 두 자리는 상세내용)를 가지고 있으므로 코드의 왼쪽부터 일치하는 문자의 수에 따라 유사도 점수에 차등을 두었다.

〔표 4-5〕 X회사의 복사기 이상증상에 대한 코드체계

증상코드	증상 대표내용	증상 상세내용
100	PAPER HANDING	FEED부 (MISS FEED)JAM
101	PAPER HANDING	DRUM부 前/기록부前 JAM
102	PAPER HANDING	DRUM부 / 기록부JAM
150	PAPER HANDING	SORTING 불량
200	COPY/PRINT/인자 QUALITY	HIGH BACKGROUND
220	COPY/PRINT/인자 QUALITY	용지 FEEDING방향 흑선/흑대
221	COPY/PRINT/인자 QUALITY	축방향 흑선/흑대

유사도는 각 속성별로 측정이 되는데, 고장진단 모듈의 경우에는 [식 4-1]과 같이 유사도 점수에 해당 속성의 가중치를 곱하여 합산한 것이 두 사례간의 유사도인 $D_Similarity(N, R)$가 된다.

$$D\_Similarity(N, R) = \sum_{i=1}^{l} f(n_i, r_i) \times w_i \qquad \text{[식 4-1]}$$

N : 새로운 사례
R : 사례베이스에서 검색된 사례
$f()$: 두 속성 사이의 유사도를 측정해 주는 함수
n_i : 새로운 사례의 i 번째 속성 값
r_i : 검색된 사례의 i 번째 속성 값
l : 사례의 속성 개수
w_i : i 번째 속성의 가중치

클러스터링 모듈은 최적의 가중치가 도출되기 전에 작동하는 것이므로 모든 속성의 가중치를 1로 하고 유사도를 측정한다. 단, 클러스터링 모듈의 경우에는 [식 4-2]와 같이 속성 별로 중심의 개수만큼 유사도 점수가 합산되어 유사도 $C_Similarity(N, C)$가 계산된다.

$$C\_Similarity(N, C) = \sum_{j=1}^{m} \sum_{i=1}^{l} f(n_i, c_{ji}) \qquad \text{[식 4-2]}$$

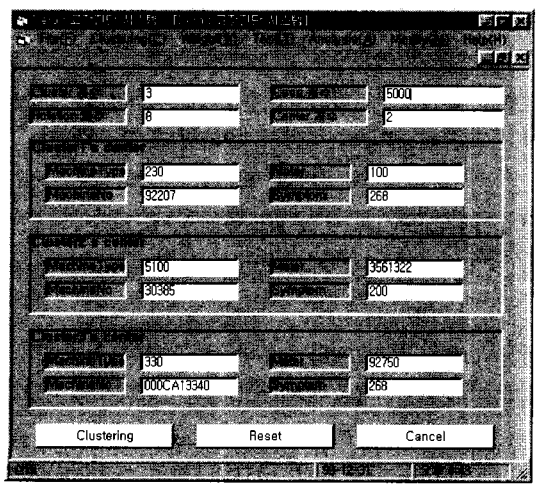
N : 새로운 사례
C : 임의의 클러스터
$f()$: 두 속성 사이의 유사도를 측정해 주는 함수
c_{ji} : 클러스터의 j 번째 중심의 i 번째 속성 값
n_i : 새로운 사례의 i 번째 속성값
m 클러스터의 중심의 개수
l 사례의 속성 개수

4.2 M-CBEMD 시스템의 구조

M-CBEMD 시스템은 Visual Basic 6.0과 Access 7.0을 사용하여 구현하였는데, 제 4.1절에서 언급한 바와 같이 세 개의 모듈, 즉 클러스터링 모듈, 가중치 훈련 모듈 그리고 고장진단 모듈로 구성되어 있다.

4.2.1 클러스터링 모듈 (Clustering Module)

클러스터링 모듈은 사례들을 사례기반 클러스터링 기법으로 클러스터링한다. 클러스터링 모듈이 실행되기 위해 필요한 속성은 '기종', '기계번호', '미터', '이상증상' 등이다. 모듈의 실행이 끝나면 그 결과로 k개의 클러스터와 클러스터별로 m개의 중심이 도출된다. 이 결과는 인덱싱되어 사례베이스에 저장되며 이 정보는 고장진단 모듈에서 사례를 검색할 때 사용된다. 본 연구에서 k값과 m값은 실험을 통하여 구했는데 상세한 내용은 제 5절에서 기술한다. 클러스터링 모듈에서의 구체적인 수행과정은 다음과 같다.



〔그림 4-1〕 클러스터링 모듈

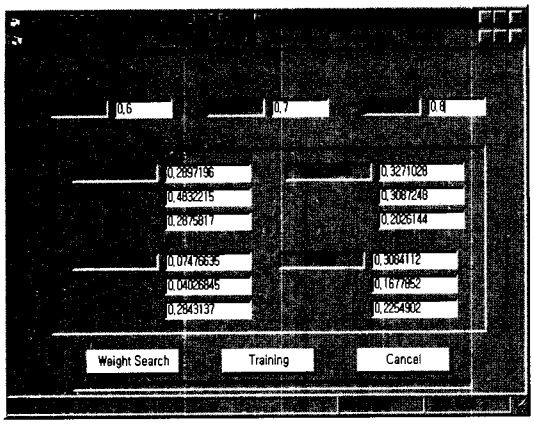
[그림 4-1]의 클러스터링 모듈 화면에서 사용자는 클러스터의 개수인 'Cluster 개수'와 중심의 개수인 'Center 개수'를 입력한다. [그림 4-1]과 같이 클러스터의 개수를 3으로, 중심의 개수를 2로 입력했다면, 클러스터링의 결과 3개의 클러스터가 생성되며 각 클러스터에는 2개씩의 중심이 정해진다.

이렇게 클러스터링이 된 결과가 [그림 4-1]의 화면에 도출된 수치이다. [그림 4-1]에는 각 클러스터별로 중심이 하나만 나타나 있으나, 이는 화면상의 제약으로 인해 가장 빈도수가 높은 중심을 하나만 보여주도록 설계되어 있기 때문이며, 실제로 모듈 내에는 2개의 중심이 모두 저장되어 유사도 측정시 사용된다. 화면 왼쪽 위에 보이는 'Rotation 회수'는 클러스터링이 완료되기까지 전체 사례에 대한 재할당 과정이 몇 번이나 일어났는지를 보여주는 수치이고, 화면 오른쪽 위의 'Case 개수'는 사례베이스 안에 저장되어 있는 전체 사례의 개수를 나타낸다.

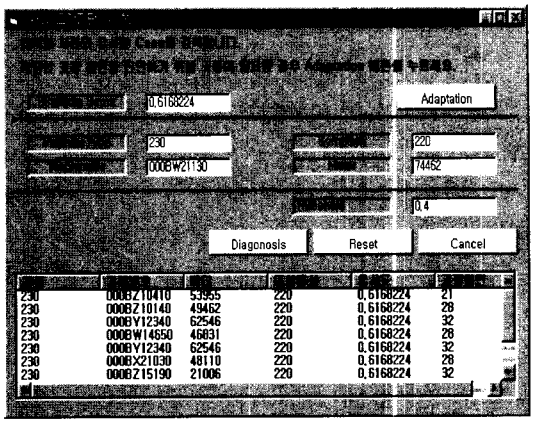
4.2.2 가중치 훈련 모듈(Weight Training Module)

가중치 훈련 모듈은 클러스터링이 끝난 후 진행된다. 각 클러스터를 별개의 사례베이스로 인식하고 가장 적합한 속성별 가중치를 찾음으로써 각 클러스터별로 가장 적합한 사례기반 고장진단 시스템이 구축된다. 속성별 가중치를 구하는 방법은 다음과 같다.

먼저 각 클러스터에 대해서 속성별 가중치 집합을 무작위로 100개 도출해 놓는다. 어떤 클러스터(C_1이라 하자)에 속한 사례들 중에서 무작위로 1개의 사례(N_1이라 하자)를 추출한다. N_1을 새로운 사례로 간주하고 C_1에 남아있는 사



〔그림 4-2〕 가중치 훈련 모듈



〔그림 4-3〕 고장진단 모듈

례들과의 유사도를 구한다. 각 속성의 유사도 점수는 제 4.1절에 제시된 기준을 사용하고, 각 속성의 가중치는 미리 무작위로 도출해 놓은 수치들을 사용한다. 유사도가 가장 높은 사례의 해(Solution)인 '고장원인'과 N_1의 '고장원인'이 동일한 경우, 답을 맞춘 것이다. 하나의 사례에 대해서 100개의 가중치 집합을 실험하는데, 이러한 사례를 중복이 안되도록 100번 추출한다. 그 중 적중률이 가장 뛰어난 가중치 집합을 클러스터 C_1의 가중치 집합으로 채택한다. 이러한 실험을 각 클러스터에 대하여 수행하여 각 클러스터별로 상이한 가중치 집합을 도출할 수 있다.

[그림 4-2]는 가중치 훈련 모듈의 화면을 보여주고 있다. 화면의 중간 부분에 보이는 각 속성명 옆의 세 개의 수치들이 위에서부터 차례대로 클러스터 1, 2, 3에서의 가중치이다. 화면 위 부분에 있는 클러스터 번호 옆의 수치들은 선정된 가중치들이 100개의 사례들을 통한 실험에서 보여주었던 적중률들이다.

4.2.3 고장진단 모듈 (Diagnostic Module)

고장진단 모듈은 새로운 고장 사례를 입력하면 사례기반 추론에 근거하여 유사한 사례들을 검색하여 '고장원인'을 제시하여 주는 모듈이다. 새로운 사례 N이 입력되면 일단 그 사례가 어느 클러스터에 속할 수 있는지를 사례기반 클러스터링 기법에 의하여 판정한다. 판정된 클러스터가 C_i라고 하면, 고장진단 모듈은 C_i에 해당하는 가중치 집합과 C_i에 속한 사례들만을 사용하여 고장진단을 수행하게 된다.

[그림 4-3]은 고장진단 모듈의 화면을 보여주고 있다. 먼저 사용자는 각 속성(기종 : 'Machine Type', 기계번호 : 'Machine No', 미터 : 'Meter', 이상증상 : 'Symptom')에 대해 새로운 사례의 값들을 입력한다. 입력 후 'Diagnosis' 버튼을 누르면 고장진단이 수행된다. 이 때에 유사도가 가장 큰 하나의 사례만을 해로 제시할 수도 있으나, M-CBEMD 시스템에서는 추출된 사례들을 유사도가 큰 순서에 따라 나열하도록 하여, 사용자가 비교·검토할 수 있도록 하였다. 화면 왼쪽 위의 'Similarity Score'는 입력된 사례와 가장 유사한 사례의 유사도를 의미하며, 제시되는 사례의 최대 수는 15개까지로 하였다. 단, 사례의 검색 시 사용자의 재량에 의한 'Threshold'의 조정을 통해 기준에 미달되는 사례는 15개의 사

례 안에 포함될 수 없도록 할 수 있다. 이는 유사한 사례로 판단되어 검색되더라도 기본적으로 만족할 만한 수준에 미달되는 사례가 검색되지 못하도록 하고자 함이다.

M-CBEMD 시스템은 위와 같은 세 가지 모듈로 운영되며, 적중률이 저하되거나 새로운 기종의 고장사례가 수집되는 경우에는 다시 클러스터링을 하거나 유사도 측정 기준을 보완하여 사용할 수 있다.

5. M-CBEMD 시스템의 성능 평가

5.1 클러스터 수에 따른 적중률의 변화

사례베이스 안의 5,000개의 사례에 대해서는 임의의 클러스터 개수 k값에 따라 사례기반 클러스터링을 수행하였는데, K-Mean 클러스터링과 마찬가지로 사례기반 클러스터링도 클러스터의 개수의 변화에 많은 영향을 받는다. 본 연구에서는 클러스터의 개수를 실험에 의하여 구했는데, 그 실험 결과는 [표 5-1]과 같다.

이 실험에서 클러스터의 중심의 개수는 2개로 고정시켰으며, 별도로 분리해 놓았던 테스트 집합 10개(각 집합은 사례 100개로 구성되어 있음)에 대해 실험하여 얻은 10개의 적중률을 평균하여 [표 5-1]에 기록하였다. 즉, 각 클러스터 개수에 대해 100개의 사례에 대한 10번씩의 테스트가 이루어진 것이다. [표 5-1]을 보면, 클러스터의 개수가 3개일 때 가장 적중률이 높은 것을 알 수 있다. 클러스터 개수를 3개 이상으로 늘렸을 때에는 수행속도는 빨라졌지만 적중률은 상대적으로 낮아지는 결과를 보였다. [표 5-1]에서 보듯이 클러스터링을 안한 경우인 클러스터 개수가 1개일 때보다 클러스터의 개수가 2개 혹은 3개일 때에 적중률이 높았다. 즉, 클러스터링 기법을 이용해서 유사한 그룹으로 군집화 한 것이 사례기반 추론 시스템의 적중률을 높이는데 기여했다는 것이다. 이상의 실험을 통하여 클러스터의 개수는 3으로 고정시켰다.

〔표 5-1〕 클러스터 개수에 따른 적중률 변화

클러스터 개수	평균 적중률
1	58%
2	60%
3	61%
4	54%
5	55%

5.2 클러스터의 중심의 개수에 따른 적중률의 변화

클러스터의 중심이 복수이어야 하는 이유에 대해서는 제 3.2절에서 언급하였다. 본 연구에서는 실험을 통해 각 클러스터의 중심의 개수에 따른 적중률의 변화를 살펴보고 가장 적합한 중심의 개수를 구하였다. 단, 이 때의 클러스터의 개수는 앞 절에서 언급한 바와 같이 3으로 고정시켰다. 실험 결과는 [표 5-2]와 같다. [표 5-1]과 마찬가지로 평균 적중률은 테스트 집합 10개에서 얻은 적중률의 평균을 의미한다.

실험 결과에 의하면 중심의 개수가 2개일 경우와 3개일 경우의 적중률이 가장 좋음을 알 수

〔표 5-2〕 클러스터의 중심의 개수에 따른 적중률 변화

중심 개수	평균 적중률
1	54%
2	60%
3	60%
4	57%
5	58%

〔표 5-3〕 각 클러스터의 속성 별 중심

클러스터 1

속성	중심 값
기종	230
	330
기계번호	92207
	000BY21030
이상증상	268
	220
미터	95579
	44004

클러스터 2

속성	중심 값
기종	5100
	230
기계번호	30385
	30033
이상증상	200
	268
미터	3561322
	6653559

클러스터 3

속성	중심 값
기종	330
	230
기계번호	000CA13340
	000CQ23210
이상증상	268
	200
미터	92750
	290307

〔표 5-4〕 각 클러스터별 적용 가중치

클러스터 1

속성	가중치
기종	0.3661972
기계번호	0.2957746
미터	0.07042254
이상증상	0.2676056

예측률 : 0.6

클러스터 2

속성	가중치
기종	0.4026549
기계번호	0.06637168
미터	0.1061947
이상증상	0.4247788

예측률 : 0.7

클러스터 3

속성	가중치
기종	0.2875817
기계번호	0.2843137
미터	0.2026144
이상증상	0.2254902

예측률 : 0.8

있다. 하지만 중심의 개수가 3개일 경우는 중심의 개수가 2개일 경우보다 클러스터링 수행시간이 오래 걸렸다. 따라서 본 논문에서는 클러스터링 수행시간이 좀 더 짧은 경우인 중심 개수 2개를 선택해서 클러스터링하였고, 고장진단 모듈에서 해당 클러스터를 판정할 때에도 동일한 조건을 적용하였다.

위의 두 가지 실험에 의해 최종적으로 도출된 클러스터별 중심은 [표 5-3]과 같다. [표 5-3]에는 세 개의 클러스터가 나타나 있으며, 각 클러스터에는 중심이 2개이기 때문에 속성 당 2개씩의 값이 나타나 있다.

5,000개의 사례를 클러스터링한 결과 클러스터 1에는 1,810개, 클러스터 2에는 1,635개, 클러스터 3에는 1,555개의 사례가 포함되었다.

5.3 클러스터별 가중치 훈련

위의 실험으로 얻어진 각 클러스터에 대해서 가중치 훈련이 수행되었다. 각 클러스터에 가장 적합한 가중치는 무작위로 추출된 300여개의 가중치를 가지고 실험을 통해 구했으며, 이들 중 가장 적중률이 좋았던 가중치가 고장진단 모듈에서 사용된다. 실험에 의해 선정된 가중치와 그 때의 적중률은 [표 5-4]와 같다.

5.4 다른 사례기반 고장진단 시스템과의 성능 비교

본 논문에서는 M-CBEMD 시스템의 성능을 다르게 설계된 사례기반 추론 시스템의 성능과 비교하기 위해서 두 개의 비교 대상 시스템을

〔표 5-5〕 성능비교를 위한 세 개의 시스템

구현된시스템	설 계 내 용
No Cluster	클러스터링을 하지 않은 기본적인 사례기반추론 시스템
3Cluster+1CBR	M-CBEMD와 동일하게 클러스터링을 하되 각 클러스터별로 속성 가중치를 설정하지 않고 모든 클러스터에 동일한 속성 가중치를 설정한 시스템
3Cluster+3CBR	본 논문에서 제시하는 시스템인 M-CBEMD로서 클러스터링을 하고 각 클러스터별로 속성 가중치를 다르게 설정한 시스템

추가로 구현하였다. 첫 째는 5,000개의 사례를 가지고 있는 사례베이스를 클러스터링하지 않고 그대로 하나의 사례베이스로 사용하는 시스템이다. 이 때에는 물론 각 속성에 대해서 하나의 가중치만이 설정된다. 두 번째는 사례베이스를 클러스터링하되 각 속성에 대해서는 하나의 가중치를 사용하는 시스템이다. 비교를 용이하게 하기 위해서 첫 번째 대안 시스템은 'No Cluster', 두 번째 대안 시스템은 '3 Cluster + 1 CBR', 그리고 M-CBEMD는 '3 Cluster + 3 CBR'로 명명하기로 하자. 이들을 일목요연하게 정의한 내용은 [표 5-5]와 같다.

[표 5-6]은 이 세 개의 시스템들이 10개의 테스트 집합에 대해서 보여주는 평균 적중률과 평균 수행시간(괄호 안에 분, 초로 기술되어 있음)이다.

〔표 5-6〕 성능비교를 위한 세 개의 시스템들의 평균 적중률과 평균 수행시간

Test Set	No Cluster	3Cluster+ 1CBR	3Cluster + 3CBR
Set 1	58%(12′21″)	57%(6′10″)	60%(6′21″)
Set 2	59%(10′52″)	63%(5′52″)	66%(5′14″)
Set 3	50%(11′12″)	58%(9′28″)	58%(10′36″)
Set 4	56%(12′56″)	58%(6′47″)	56%(6′02″)
Set 5	63%(11′25″)	70%(5′47″)	70%(5′03″)
Set 6	53%(10′42″)	56%(6′09″)	65%(6′50″)
Set 7	56%(11′36″)	59%(6′54″)	58%(7′11″)
Set 8	57%(11′58″)	62%(5′31″)	59%(5′12″)
Set 9	51%(10′47″)	54%(5′10″)	57%(5′28″)
Set 10	60%(11′29″)	60%(5′51″)	62%(6′31″)
평균	56%(11′53″)	59%(6′36″)	61%(6′34″)

Test set : 각 100개의 Case

<표 5-6>을 보면 클러스터링을 한 두 시스템이 그렇지 않은 경우인 'No Cluster' 시스템보다 평균 적중률이 향상되었음을 알 수 있다. '3 Cluster + 3 CBR' 시스템은 'No Cluster' 시스템에 비해서 평균 적중률이 약 5% 포인트 증가하였다. 이는 입력사례와 유사한 사례들이 모여있는 클러스터에 대해서만 사례기반 추론을 수행함으로써 유사하지 않은 사례들이 미칠 수 있는 잡음효과를 줄일 수 있었기 때문이다. 평균 수행시간을 보면, 클러스터링을 했을 경우에는 'No Cluster' 시스템보다 평균 수행시간이 50% 정도 감소하였다. 시간으로 보면 평균적으로 약 5분 정도 감소하였다. 이것도 위의 경우와 마찬가지로 입력사례와 유사한 사례들이 모여있는 클러스터에 대해서만 사례기반 추론을 수행하므로 검색할 사례의 수가 현저히 줄었기 때문이다.

'3 Cluster + 1 CBR' 시스템과 '3 Cluster + 3 CBR' 시스템을 비교해 보면 평균 수행시간은 거의 비슷하다. 이는 두 시스템이 동일하게 클러스터링 되어 있으며 가중치의 설정에 있어서만 차이를 보이므로 검색 소요시간에는 별 차이가 없기 때문이다. 반면, 평균 적중률의 경우에는 '3 Cluster + 3 CBR' 시스템이 평균적으로는 2% 포인트 높았으나 모든 테스트 집합에 대해서 평균 적중률이 높았던 것은 아니다. 세 시스템간의 평균 적중률 차이를 통계적으로 검증해 보았다. 가설검정 1은 'No Cluster' 시스템과 '3 Cluster + 3 CBR' 시스템간의 비교이고, 가설검정 2는 '3 Cluster + 1 CBR' 시스템과 '3 Cluster + 3 CBR' 시스템간의 비교이다.

가설검정 1 : 'No Cluster' 시스템보다 '3 Cluster + 3 CBR' 시스템의 평균 적중률이 증가했는지를 테스트

〔표 5-7〕 t-Test 결과

	가설검정 1	가설검정 2
가설	$H_0 : \mu_1 = \mu_3$ $H_1 : \mu_1 \neq \mu_3$	$H_0 : \mu_2 = \mu_3$ $H_1 : \mu_2 \neq \mu_3$
Degree of Freedom	18	18
t 기각치(양측검정)	1.734063	1.734063
t 통계량	-1.84271	-1.26789
p value	0.082881	0.220995

가설검정 2 : '3 Cluster + 1 CBR' 시스템보다 '3 Cluster + 3 CBR' 시스템의 평균 적중률이 증가했는지를 테스트

[표 5-7]은 세 시스템의 비교에 대한 통계적 검증 결과이다. [표 5-7]에 나타난 가설검정 1의 결과를 보면, 'No Cluster' 시스템보다 '3 Cluster + 3 CBR' 시스템의 평균 적중률이 확실히 증가하였으며, 이를 신뢰수준 90%로 확신할 수 있다. 하지만 가설검정 2의 결과를 보면, '3 Cluster + 1 CBR' 시스템보다 '3 Cluster + 3 CBR' 시스템의 평균 적중률은 높다고 할 수 있지만 t 통계량이 기각역 밖에 있는 것으로 보아 이 수치의 경우는 그다지 유의한 결론이 아님을 알 수 있다. 하지만 클러스터별로 개별 사례기반 추론 시스템을 구축한 결과 적중률이나 수행시간 면에서 개선 가능성을 볼 수 있었으며, 개별 클러스터의 특성을 면밀히 분석하여 좀더 적합한 개별 사례기반 시스템을 구축하면 향상된 결과를 얻을 수 있을 것이다.

6. 결 론

사례기반 추론 기법은 새로운 사례에 대한 클래스를 판정하기 위한 학습과정이 새로운 사례가 입력된 후에 시작하는 사후학습기법이므로 사례베이스의 크기가 증가하면 수행시간이 증가되는 문제가 발생한다. 본 논문에서는 이러한 문제의 해결 방안으로 사례베이스를 클러스터링하여 시스템을 구축하는 방안을 제시하였다. 클러스터링 기법으로는 범주형 속성을 다루기에 적합한 새로운 클러스터링 기법을 개발하여 사용하였다. 사례베이스를 클러스터링한 후에 각 클러스터를 사례베이스로 이용하는 개별적인 사례기반 시스템을 클러스터의 개수만큼 만든다. 새로운 사례가 입력되면 먼저 그 입력사례가 할당될 수 있는 클러스터를 찾고, 그 클러스터에 해당하는 사례기반 추론 시스템을 작동시킴으로써 시스템의 수행시간을 감소시킬 수 있다. 본 연구에서 구축된 시스템을 복사기 고장진단 문제 적용한 결과, 클러스터링을 하지 않은 전통적인 사례기반 시스템과 비교하여 평균 적중률은 56%에서 61%로 5% 포인트 증가하였고, 평균 수행시간은 100개의 테스트 사례에 대해서 약 12분에서 6분으로 50% 감소하였다.

본 연구에서 도출된 한계점 및 향후 연구 방향은 다음과 같다. 첫째, 개별 클러스터간의 독창적인 구별 속성 발견이 필요하다. 본 연구에서는 클러스터별로 개별 시스템을 개발하고자 했으나, 모든 클러스터에 속성들을 동일하게 사용하고 다만 속성들의 가중치를 다르게 하는 것에 그쳤다. 보다 의미있는 개별 시스템을 구축하기 위해서는 클러스터간에 사용되는 속성도 달라져야 할 것이다. 두 번째는 어느 사례기반 추론 시스템에서나 언급되는 문제로서 가중치 설정에 관한 것이다. 본 논문에서는 무작위로 도출된 수많은 가중치 후보들로부터 무수한 실험을 통하여 가장 적중률이 뛰어난 가중치들을 선정하였다. 하지만 이 방법은 매우 긴 시간이 소요되었으며, 최적의 가중치인지에 대해서 확

신을 할 수가 없다. 그러므로 보다 과학적인 가중치 도출에 대한 연구가 지속적으로 수행되어야 할 것이다.

참 고 문 헌

[1] 이재식, 김영길, "규칙 및 사례기반의 하이브리드 고장진단 시스템," 한국전문가시스템학회지, 4권, 1호 (1998), 115-131.

[2] 이재식, 안태훈, "클러스터링 기법을 이용한 사례기반 시스템의 성능 향상", 한국전문가시스템학회 춘계학술대회 논문집, (1998), 112-115.

[3] 이재식, 전용준, "사례기반 추론에 근거한 설비이상 진단 시스템," 한국전문가시스템학회지, 1권, 2호 (1995), 85-102.

[4] Aamodt A. and E. Plaza, "Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches," AI Communications, Vol.7, No.1 (1996), 39-59.

[5] Afifi, A. A. and V. Clark, Computer-Aided Multivariate Analysis, Chapman & Hall, 1990.

[6] Aha, D. W., "Case-Based Learning Algorithms," Proceedings of the 1991 DARPA Case-Based Reasoning Workshop, 1991.

[7] Aha, D. W. and D. Wettschereck, "Case-Based Learning: Beyond Classification of Feature Vectors," Proceedings of the European Conference on Machine Learning, 1997.

[8] Auriol, E., S. Wess, M. Manago, K. D. Althoff and R. Traphner, "INRECA: A Seamlessly Integrated System Based on Inductive Inference and Case-Based Reasoning," Proceedings of First International Conference on Case-Based Reasoning (1995), 371-380.

[9] Berry, M. and G. Linoff, Data Mining Techniques: For Marketing, Sales, and Customer Support, John Wiley & Sons, Inc., 1997.

[10] De Carvalho, R. R., S. G. Djougovski, N. Weir, U. Fayyad, K. Cherkauer, J. Roden and A. Gray, "Clustering Analysis Algorithms and Their Applications to Digital POSS-II Catalogs," Astronomical Data Analysis Software and Systems IV, ASP Conference Series, Vol.77, 1995.

[11] Fox, S. and D. B. Leake, "Learning to Refine Indexing by Introspective Reasoning," Proceedings of First International Conference on Case-Based Reasoning (1995), 431-440.

[12] Huang, Z., "Clustering Large Data Sets with Mixed Numeric and Categorical Values," SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery, Tucson, Arizona, USA, 1997.

[13] Johnson, R. A. and D. W. Wichern, Applied Multivariate Statistical Analysis, Prentice-Hall, Inc., pp. 532-573, 1982.

[14] Karamouzis, S. and T. Feyock, "Case-based Approach to Handling Aircraft Malfunctions," Proceedings of the SPIE-The International Society for Optical Engineering, Vol.1963 (1993), 274-284.

[15] Magaldi, R. V., "CBR for Troubleshooting Aircraft on the Flight Line," Proceedings. of IEE Colloquium on CBR : Prospects for Applications, Digest No. 1994/057, London, UK (1994), 6/1~6/9.

[16] Manago, M. and E. Auriol, "Using Data Mining to Improve Feedback from Experience for Equipment in the Manufacturing and Transport Industries," Proceedings of IEE Colloquium on

Knowledge Discovery and Data Mining, Digest No. 1996/198, London, UK (1996), 1/1～1/9.

[17] Marir, F. and I. Watson, "Representing and Indexing Building Refurbishment Cases for Multiple Retrieval of Adaptable Pieces of Cases," Proceedings of First International Conference on Case-Based Reasoning (1995), 55-66.

[18] Racine, K. and Q. Yang, "On the Consistency Management of Large Case Bases : the Case for Validation," AAAI Technical Report—Verification and Validation Workshop, 1996.

[19] Riesbeck, C. K. and R. S. Schank, Inside Case-Based Reasoning, Lawrence Erlbaum Associates, Inc., 1989.

[20] Tsutsui, H., A. Kurosaki, T. Sato and Y. Hiraide, "Fault Detection using Topological Case based Modeling and its Application to Chiller Performance Deterioration," Proceedings of IEEE Instrumentation and Measurement Technology Conference, Vol.1, Hamamatsu, Japan (1994), 390-393.

[21] Watson, L. and S. Abdullah, "Developing Case-based Reasoning Systems: A Case Study in Diagnosing Building Defects," Proceedings of the IEE Colloquium on Case-Based Reasoning: Prospects for Applications, Digest No. 1994/057, London, UK (1994), 1-3.

[22] Yoo, J., A. Gray, J. Roden, U. M. Fayyad, R. R. de Carvalho and S. G. Djorgovski, "Analysis of Digital POSS-II Catalogs Using Hierarchical Unsupervised Learning Algorithms," Astronomical Data Analysis Software and Systems V, ASP Conference Series, Vol.101, 1996.