

XML 템플릿 기반 지식베이스 관리 기법과 그 응용

이 혜 자* · 정 병 수** · 박 승 훈***

요 약

전문가의 지식을 다양하고 효율적으로 활용하기 위해서는 복잡한 지식을 구조적으로 표현하고 변화에 유연하게 적응하도록 관리하는 것이 무엇보다도 중요하다. 본 논문에서는 XML 템플릿(template)을 기반으로 한 지식베이스 관리 방법을 제안한다. 제안하는 방법에서는 복잡하고 다양한 정보를 구조적으로 표현하는 데 효과적인 XML을 이용하면서, XML 문서의 종류와 개수가 많아지면서 발생하는 문서관리상의 비효율성 문제를 해결하기 위하여, 복잡한 지식을 XML로 표현된 다수의 템플릿으로 분류하여 작성하고, XML 템플릿들 사이의 연관관계 등을 정의한 메타템플릿을 이용하여 구조적으로 관리한다. 제안한 방법의 효율성을 입증하기 위해 개인의 건강과 체력에 적합한 운동을 처방하는 데 필요한 지식베이스와 운동처방 프로세스 중 대상자의 체력유형을 분류하는 프로그램을 개발하고 그 결과를 분석하였다. 실험을 통해 본 논문에서 제안하는 XML 템플릿 기반 지식베이스 관리 방법은 복잡한 전문지식을 구조적으로 표현하고, 새로운 유형의 정보가 필요하거나 기존 정보의 변경이 필요할 경우 유연하게 대처할 수 있으며, 메타템플릿을 이용하여 XML 템플릿들을 구조적으로 관리함으로써 효율적으로 추론할 수 있음이 확인되었다.

A Knowledge Base Management Technique Based on XML Templates and its Application

Hiye-Ja Lee* · Byeong-Soo Jeong** · Seung-Hun Park***

ABSTRACT

For using the knowledge of experts diversely and efficiently, it is very important that the system could represent the complex knowledge in well structured manner and also adapt well to the frequent schema changes. In this paper, we propose an efficient method for knowledge base management by using XML-based templates. In our proposed method, to solve the inefficiency of management of many XML documents, we represent the complex knowledge on XML-based templates, and manage the templates structurally by using a meta-template that defines relationship among the templates. In order to show the effectiveness of our method, we developed a knowledge base to formulate an exercise prescription for a subject and an application program to classify the subject's fitness. The experimental study shows that XML-based templates provide a flexible, extensible and structured way of representing expert knowledge. The inference using XML-based templates can be controlled systematically and efficiently by using meta-template.

키워드 : XML, 지식베이스(Knowledge Base), 지식표현(Knowledge Representation), 지식기반 시스템(Knowledge-based System), 전문가 시스템(Expert System), 운동처방(Exercise Prescription)

1. 서 론

정보기술의 발달로 거의 모든 분야에서 컴퓨터를 사용하고 있으며, 매우 다양하게 컴퓨터를 활용하고자 하는 요구는 계속 증가하고 있다. 이와 함께 복잡한 전문지식을 좀더 효율적으로 활용하고자 하는 전문가시스템에 대한 요구도 증가하고 있다. 전문가의 지식을 적절하게 활용하기 위해서는 복잡하고 시간에 따라 변할 수 있는 전문지식을 구조적으로

잘 표현하고 효율적으로 관리하는 것이 무엇보다도 중요하다[1, 2].

전문가시스템에서 이용되는 지식베이스는 문제를 해결하는 데 필요한 사실과 그 사실들을 바탕으로 논리적인 추론을 할 수 있는 규칙으로 구성된다. 지식베이스에서 지식을 표현하고 관리하는 방법에 대한 연구가 많이 진행되어 왔다. 그러나 모든 방법들은 장단점을 갖고 있으며, 실용화에 문제가 있다. 특히 쉽게 효율적으로 이용할 수 있는 도구가 부족하다[3-7].

인터넷의 급속한 발전과 함께 인터넷 문서의 표준 형식으로 제안된 XML(eXtensible Markup Language)은 상이한 포맷으로 존재하는 문서를 통합 관리하는 데 용이하다. 다

* 본 연구는 보건복지부 보건의료기술진흥사업의 지원에 의하여 이루어진 것임(과제고유번호: 02-PJ3-PG6-EV01-001).

† 정 회 원 : 용인송담대학 의료정보시스템과 교수

†† 종신회원 : 경희대학교 전자정보학부 교수

††† 정 회 원 : 경희대학교 동서의료공학과 교수

논문접수 : 2004년 4월 30일, 심사완료 : 2004년 10월 1일

양한 형태의 정보를 구조적으로 표현하고 태그에 의미를 부여할 수 있어 상세하게 정보를 검색할 수 있으며, 구조적인 정보를 데이터의 손실 없이 데이터베이스에 저장, 검색, 관리할 수 있게 할 뿐만 아니라 기존 정보의 변경이나 새로운 유형의 정보 추가도 쉽다는 장점을 가지고 있다. 이러한 장점으로 인해 XML은 데이터의 저장과 전송에 있어 표준 언어가 되고 있다[8-11].

본 논문에서는 XML의 장점을 이용한 지식베이스 관리 방법을 제안한다. 제안 방법에서는 인간 전문가의 복잡한 전문지식을 XML로 표현된 다수의 템플릿(template)으로 저장하여 관리함으로써 복잡한 전문지식을 구조적으로 표현하고 효율적으로 활용하며 새로운 유형의 정보가 필요하거나 기존 정보의 변경이 필요할 경우 유연하게 대처하고자 한다. 또한 XML 템플릿들 사이의 연관관계 등을 정의한 메타템플릿을 이용하여 구조적으로 관리함으로써, XML 문서의 종류와 개수가 많아지면서 발생하는 문서관리상의 비효율성 문제를 해결하고자 한다.

본 논문의 2장에서는 관련 연구에 대해 살펴보고, 3장에서는 제안하는 지식베이스 관리 방법을 소개한다. 4장에서는 본 논문에서 제안하는 방법을 실험하기 위해 개인의 건강과 체력에 적합한 운동을 처방하는 데 필요한 지식베이스와 운동처방 프로세스 중 운동처방 대상자의 체력유형을 분류하는 프로그램을 소개하고 그 결과를 분석하며, 5장에서는 제안하는 방법에서 미해결된 사항들을 소개하고 향후 연구 방향을 제시하고자 한다.

2. 관련 연구

2.1 지식표현과 지식베이스 관리 방법

전문가 시스템에서 이용되는 지식베이스에서 지식을 표현하는 방법으로는 논리(logic)를 이용하는 방법, 의미망(semantic network)을 이용하는 방법, 프레임(frame)을 이용하는 방법, 생성규칙(production rule)을 이용하는 방법, 그리고 여러 가지 방법을 혼합한 복합적 방법 등이 있으며, 각 방법의 장단점을 비교하면 <표 1>과 같다[3-7].

최근 객체지향기법이 일반화되면서 객체지향기법과 규칙에 의한 지식표현 방법이 혼합된 형태가 많이 연구되고 있다. 이 방법에서는 지식들을 클래스, 객체 및 속성 등으로 분류하여, 주어진 문제들을 인간이 생각하는 자연스러운 방식으로 표현하게 되므로, 개발자 및 사용자가 쉽게 표현하고 이해할 수 있으며 시스템의 유지보수성과 확장성을 증대시킬 수 있다[3-7]. 그러나 쉽게 효율적으로 이용할 수 있는 도구가 부족한 편이며, 생성규칙을 이용하여 규칙을 표현함에 따라 문제해결 시 제어의 흐름을 추적하기가 어렵고 모든 규칙들에 대한 비교, 선택, 처리과정을 거쳐야 하는 비효율성은 가지고 있다.

<표 1> 지식표현 방법의 장단점 비교

방법	장점	단점
논리를 이용한 방법	- 나타내고자 하는 의미를 정확히 표현할 수 있다. - 여러 사실로부터 유추되는 결과를 명확하게 알 수 있다.	- 사실이 많아질수록 추론이 어려워진다. - 지식표현과 추론과정이 분리되어 있어 지식베이스에 저장된 사실을 사용하는 방법을 결정하기가 어렵다.
의미망을 이용한 방법	- 개념을 공유할 수 있어 계층적이거나 분류학적인 문제영역에 효율적이다.	- 정형적으로 뚜렷한 표현구조가 없어 자연스럽고 기술적인 지식을 표현하기가 어렵다.
프레임을 이용한 방법	- 다른 지식표현 방법에 비해 효과적인 지식검색 작업을 할 수 있다.	- 구성이 복잡하고 모든 지식이 이런 방법으로 쉽게 구현될 수 없다
생성규칙을 이용한 방법	- 통일된 하나의 형태로 지식을 표현하므로 지식의 추가, 제거, 변경 및 이해가 쉽다. - 서술적인 지식을 표현하기에 도 적절하다.	- 형태가 엄격하여 문제해결 시 제어흐름을 추적하기가 어렵고, 모든 규칙들에 대한 비교, 선택, 수행과정을 거치므로 비효율적이다.
객체지향 기법과 생성규칙을 혼합한 방법	- 주어진 문제들을 자연스러운 방식으로 표현하므로 개발자 및 사용자가 쉽게 표현하고 이해할 수 있다. - 시스템의 유지보수성과 확장성을 증대시킨다.	- 쉽게 이용할 수 있는 도구가 부족한 편이다. - 생성규칙을 이용함에 따라 문제해결 시 제어흐름 추적이 어렵고, 모든 규칙들에 대해 비교처리과정을 거치므로 비효율적이다.

앞으로 지식기반 시스템을 구축하는 데 있어 강조점은 무에서 구축하기보다는 기존 자원을 고쳐 재사용하는 데 두어지고 있으며, 해당분야 전문가들이 지식엔지니어들의 도움을 받지 않고도 시스템을 수정할 수 있고, 시스템의 구조가 좋아져 개발 및 유지보수 비용을 낮출 수 있는 방향으로 발전되고 있다[1].

본 연구에서는 지식을 표현하고 지식베이스를 관리하는데 있어 구조적 표현이 가능하고 유지보수성과 확장성이 좋은 XML을 이용하고자 한다.

2.2 XML과 지식관리

XML이 정보를 기술하는 형식으로 널리 쓰이게 되자, 많은 데이터베이스 관리 시스템(DBMS) 제조사들이 그들의 시스템들에 관계형 또는 객체지향 데이터를 XML 형식으로 내보내는 능력을 추가해 오고 있으며, 주석이 추가된 텍스트나 파워포인트 발표자료 등 다른 데이터 형식들도 손쉽게 XML로 전환되고 있어 XML은 실질적인 정보교환표준이 되었다[12, 13].

최근 여러 분야에서 이러한 XML의 장점을 이용한 연구가 활발히 진행되고 있다. 그 중에는 XML문서를 합성적인 관점에서 재구성하여 지식을 효율적으로 도출하는 방법[8], 확정적이지 않은 확률적인 데이터를 다루기 위해 XML을 이용하는 방법[14], XML을 이용하여 반구조적인 자료를 구

조적으로 추론하는 방법[15] 등 효율적인 지식 검색 및 추론을 위한 여러 가지 연구들이 진행되고 있다.

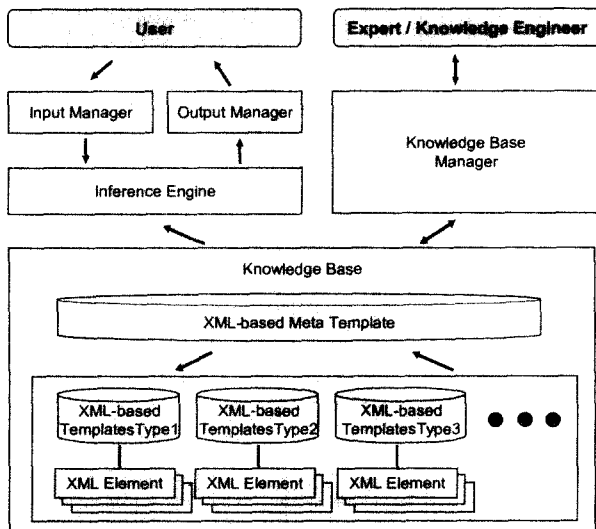
또한 교육용 웹 사이트의 효율적 관리, 인문과학 분야의 지식 관리, 법률문서 관리, 문서 기반 대학관리시스템 구축 등 여러 응용분야에서 복잡한 지식 및 정보를 관리하기 위해 XML을 이용하고 있다[16-19].

본 연구에서 대상으로 하는 지식은 시간의 흐름에 따라 정보의 구조가 변할 수 있는 복잡한 대용량의 지식이다. XML을 이용할 경우 복잡한 지식을 구조적으로 표현하는 것은 가능하나, XML 문서의 종류와 개수가 많아지면 문서의 관리가 어려워진다. 따라서 본 연구에서는 XML을 이용하되, 다양한 대용량의 XML 문서를 효율적으로 관리하기 위하여, 지식을 XML로 표현된 다수의 템플릿으로 분류하여 작성하고, XML 템플릿들 사이의 연관관계 등을 정의한 메타템플릿을 이용하여 구조적으로 관리하고자 한다.

3. 지식베이스 관리 기법

3.1 시스템의 구조

XML 템플릿을 이용한 지식관리 시스템의 구조는 전체적으로는 일반적인 전문가시스템과 같은 구조를 취한다. 단지 지식베이스 내부의 구조와 추론엔진이 XML 템플릿을 기반으로 하고 있다는 점에서 차이가 있다. (그림 1)에서 볼 수 있듯이 지식을 관리하는 전문가 또는 지식공학자는 지식베이스 관리 모듈(Knowledge Base Manager)을 이용하여 지식베이스(Knowledge Base)를 관리한다. 지식베이스에 저장된 지식을 이용하는 일반 사용자는 입력관리 모듈(Input Manager)을 통하여 필요한 자료를 입력하고 출력관리 모듈(Output Manager)을 통하여 원하는 결과를 얻게 된다. 입력 및 출력관리 모듈과 지식베이스의 사이에는 XML 템플릿으로 표현된 지식으로부터 원하는 결과를 얻는 데 이용되는 추론엔진(Inference Engine)이 있다.



(그림 1) XML 기반 지식관리 시스템의 구조

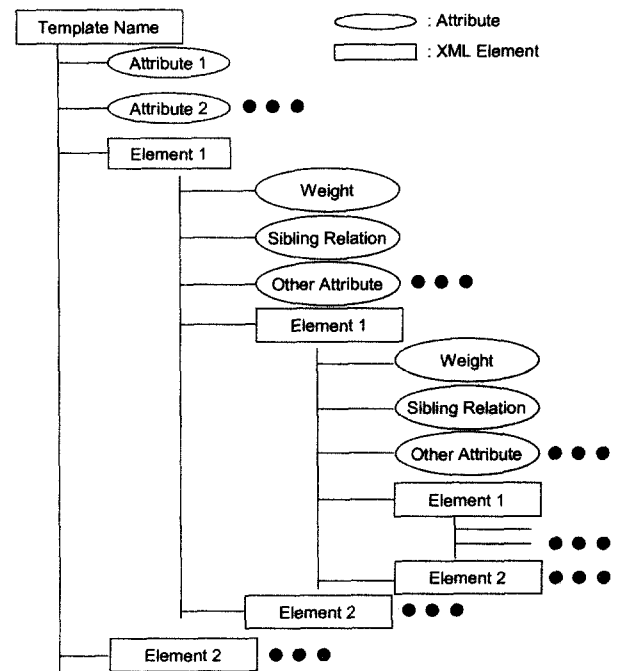
3.2 지식표현과 지식베이스 구조

전문가시스템에서 지식표현이라는 것은 전문가가 해당 분야에 대해 알고 있는 것을 체계적인 방식으로 코드화하는 것을 의미한다[4]. 전문가시스템에서 지식표현 시 고려해야 할 사항으로는 선택된 지식이 일반적이어야 하며, 표현된 지식을 다른 사람들이 쉽게 이해할 수 있어야 하며, 표현된 지식에 추가하여 지식을 삽입, 삭제 또는 수정하고자 할 시 기존의 지식표현을 크게 수정함이 없이 작업할 수 있어야 하고, 추론이 가능하도록 지식을 표현해야 한다는 것을 들 수 있다[6, 7].

본 제안 방법에서는 XML의 장점을 이용하여 지식을 구조적으로 표현하되, XML 문서의 종류와 개수가 많아지면서 발생하는 문서관리상의 비효율성을 해결하기 위하여, 지식을 용도별로 또는 내용별로 구분하여 각각의 특징에 맞는 XML 템플릿으로 나타낸다.

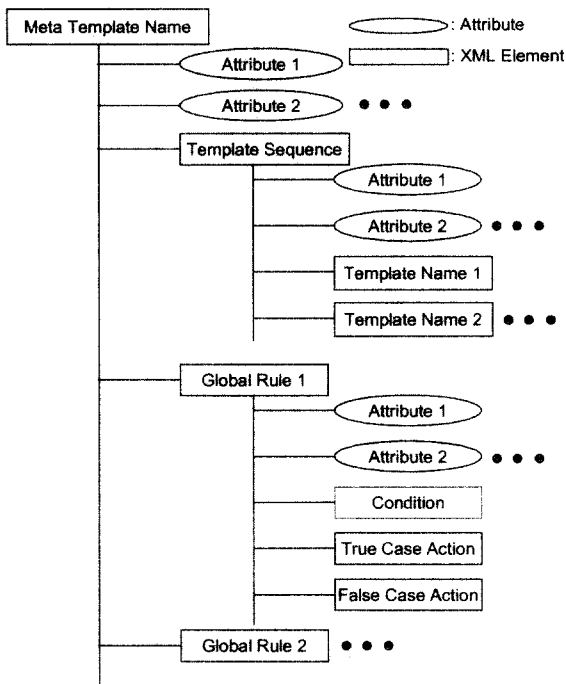
XML 템플릿은 XML로 표현된 템플릿을 의미하며, XML의 장점을 지니고 있으므로 다양한 형태의 복잡한 지식을 구조적으로 쉽게 표현하고 새로운 유형의 지식을 추가하거나 기존의 정보를 수정 또는 삭제할 시 유연하게 대처할 수 있는 특징을 가지고 있다.

XML 템플릿의 종류를 크게 구분하면 2가지이다. 하나는 내용별 또는 용도별 세부지식을 담고 있는 일반 템플릿이고 다른 하나는 템플릿들간의 연관관계 등에 관한 내용으로 구성되어 있으며 일반 템플릿의 상위에 위치하는 메타 템플릿(meta-template)이다.



(그림 2) XML 템플릿의 구조

하나의 XML 템플릿은 결국 하나의 XML 문서(XML document)와 같은 형태로 표현된다. (그림 2)는 XML 템플릿의 구조를 보여주고 있다. 각 템플릿 내에는 다수의 XML 요소들(elements)과 속성들(attributes)이 있고, 각 XML 요소에는 다수의 하위 요소들과 속성들이 있다. 각 하위요소 내에는 더 하위의 요소들과 속성들이 있는 형태가 반복된다. (그림 3)은 용도별 또는 내용별 지식을 담고 있는 일반적인 템플릿의 상위에 위치하는 메타 템플릿의 구조이다. 지식 활용 시 필요한 다수의 일반 템플릿들의 적용 순서와 전역 규칙들(global rules)이 정의되어 있다.



(그림 3) 메타 템플릿의 구조

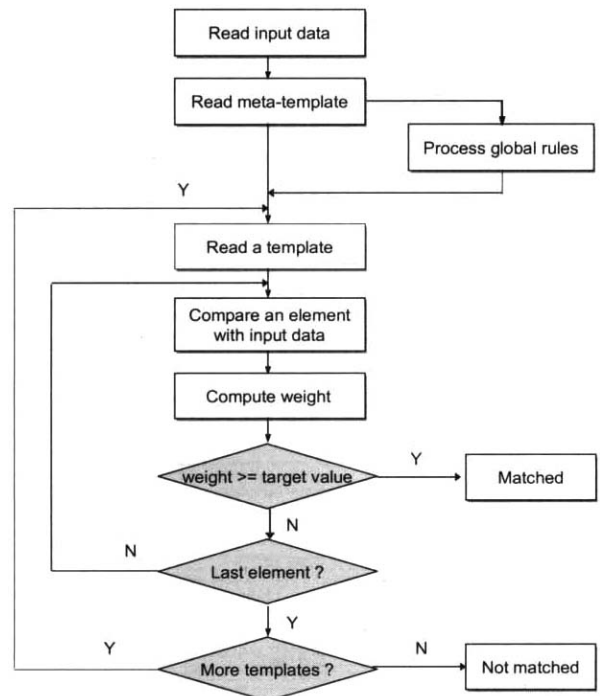
사실이 저장된 지식을 바탕으로 추론할 때 필요한 규칙은 템플릿 자체에 포함되어 있다. 즉 대부분의 세부 규칙들은 내용별 또는 용도별 세부지식을 담고 있는 일반 템플릿 내에, XML 요소들에 대한 속성과 값으로 표현된다. 나머지 모든 템플릿에 적용되는 템플릿 적용 순서나 전역 규칙들은 메타 템플릿에 표현된다. 따라서 메타 템플릿은 일반 템플릿의 상위에 위치하게 된다(그림 1).

3.3 추론 엔진

XML 템플릿으로 표현된 지식을 이용하여 추론하는 방법은 입력 자료와 지식베이스에 저장되어 있는 패턴별 표준 템플릿들을 매칭하는 방법으로 이루어진다. 패턴 매칭 순서는 템플릿들간의 연관관계 등에 관한 내용이 표현되어 있는

메타 템플릿에 정의된 적용순서를 기준으로 한다. 패턴 매칭 시 규칙을 적용하는 순서는 메타 템플릿에 정의된 규칙이 있는 경우에는 그것을 우선적으로 적용하고, 그 다음에 각 템플릿의 XML 요소별로 속성과 값으로 정의된 규칙들을 적용한다.

XML 템플릿으로 표현된 지식을 이용한 추론의 처리과정은 (그림 4)와 같은 순서도로 요약할 수 있으며, 그 알고리즘을 C언어 구문으로 표현하면 (그림 5)와 같다. 먼저 입력 자료와 메타 템플릿을 읽는다. 메타 템플릿에 전역 규칙(Global_Rules)이 정의되어 있는 경우에는 그 규칙을 읽어서 처리한다. 그 다음에는 메타 템플릿에 정의되어 있는 템플릿 매칭 순서(Template_Sequence)를 읽고, 그 순서에 따라 각 템플릿을 읽어서 처리한다. 각 템플릿에 대한 처리는 루트 요소에서부터 차례로 항목별 표준 값을 가지고 있는 말단 요소를 찾아 표준 템플릿의 값과 입력 값을 비교하는 과정이 반복된다. 이와 같이 요소 값을 비교하는 함수가 재귀적으로(recursively) 호출되어 템플릿 내의 표준 값을 가지고 있는 모든 요소들에 대한 비교가 이루어진다. 입력 값과 표준 값을 비교한 결과, 입력 값이 표준 값의 범위에 해당되면(규칙을 만족하면), 각 요소에 대한 가중치를 계속 더하여 나간다. 전체 가중치 합이 정해진 기준 값(Target_Value)을 초과하면 패턴을 찾은 경우로, 더 이상 비교하지 않고 패턴 매칭 작업을 끝낸다.



(그림 4) XML 템플릿 기반 추론의 흐름도

```

read input data
read Meta_Template
if Global_Rules exist in Meta_Template { /* 메타템플릿에 전역
규칙이 정의되어 있는 경우 */
    read Global_Rules
    process_GlobalRules()
}
read Template_Sequence in Meta_Template /* 메타템플릿에서 템
플릿 매칭 순서를 읽는다. */
do {
    /* Template_Sequence에 정의된 템플릿 매칭 순서에 따라 각
템플릿을 읽는다. */
    read a Template_Name
    /* 템플릿의 루트 노드(요소)를 읽는다. */
    read root_node of the Template_Name
    /* 노드마다 비교함수를 호출하고, 리턴 값이 'Matched'인 경우,
더이상 비교하지 않고 작업을 끝낸다. */
    if (compare_Element(root_node) = "Matched")
        return Template_Name
} until last Template_Name

compare_Element(current_node) /* 노드(요소) 비교함수 */
{
    do {
        /* 현재노드가 말단요소이면 노드의 값과 입력 값을 비교
한다. */
        if current_node has no child {
            /* 입력 값이 표준 값의 범위에 해당되면, 그 요소에
대한 가중치를 더한다. */
            if (compare current_node.value with input data)
                = True {
                    totalWeight = totalWeight +
                    current_node.getAttribute(Weight)
                /* 현재노드와의 논리적 관계가 "OR"이면, 나머지
현재노드는 더이상 비교하지 않는다. */
                if (current_node.getAttribute(Sibling_Relation)
                    = "OR") return
            }
        } else { /* 현재노드가 말단요소가 아니면, 첫 번째 자식
노드를 현재노드로 하여 자신의 함수를 재귀적으로 부른
다. */
            compare_Element(current_node.firstChild)
        }
        /* 전체 가중치 값이 정해진 기준 값을 초과하면, 패턴을
찾았음을 알리고 비교작업을 끝낸다. */
        if (totalWeight >= Target_Value)
            return "Matched"
        /* 다음 현재노드를 현재노드로 하여 비교작업을 반복 처
리하도록 한다. */
        current_node = current_node.nextSibling
    } until current_node is NULL
}

```

(그림 5) XML 템플릿 기반 패턴매칭 알고리즘

3.4 지식의 획득과 관리

3.4.1 지식의 획득 과정

지식 획득 과정에서 먼저 해야 할 일은 전문가의 복잡한 지식을 용도별로 또는 내용별로 구분하고, 템플릿 유형(template type)을 만드는 것이다. 도출된 각 템플릿 유형별로 해당 세부 지식을 구조적으로 표현하기 위한 XML 요소들을 도출하고 요소들간의 관계를 분석하여 XML 트리 형

태로 구성한다. 그리고 필요 시 각 요소에 대해 속성들을 정의한다.

세부 지식을 구성하는 XML 템플릿 유형들에 대한 정의가 완료되면, 템플릿 유형별로 XML 스키마(XML Schema)를 작성한다. 각 템플릿 유형에 대한 XML 스키마가 완성되면, 각 XML 스키마를 기반으로 하여 해당 유형의 지식들을 XML로 표현한다. 이 때 표현되는 지식은 전문가가 경험을 통해 알고 있는 지식 그대로일 수도 있고 데이터 마이닝 등의 지식관련 기술을 이용하여 정제된 지식일 수도 있다.

각 세부 지식에 대한 템플릿들이 완성되면, 템플릿들간의 연관관계 즉, 템플릿들의 적용 순서, 모든 템플릿들에 적용되는 규칙 등에 관한 내용으로 구성된 메타 템플릿을 작성한다.

XML 템플릿으로 표현된 세부 지식은 인터넷 기반의 지식베이스 관리자(Knowledge Base Manager)를 통하여 해당 XML 스키마를 기준으로 한 유효성 검증 작업을 거친 뒤, XML 문서 파일로 정해진 폴더나 관계형 데이터베이스에 저장된다.

3.4.2 새로운 유형의 지식 추가

환경의 변화, 인식의 발전 등으로 시스템 개발 완료 후 새로운 유형의 지식을 추가해야 할 필요성은 점점 더 늘어나고 있다. 관계형 데이터베이스를 이용할 경우 새로운 유형의 정보 또는 지식을 추가하려면 데이터베이스의 스키마를 변경하고 필요시 관련 프로그램을 수정해야 할 것이다. 일반 텍스트 파일이나 인덱스 파일을 이용할 경우에도 마찬가지로 파일 구조에 대한 정의를 변경하고 관련 프로그램을 수정해야 할 것이다. 그러나 XML 템플릿 기반 시스템에서는 간단히 XML 템플릿과 XML 스키마에 해당부분을 추가하면 된다.

3.4.3 기존 정보의 수정 및 삭제

지식의 수정과 삭제 역시 새로운 유형의 지식 추가 시와 마찬가지로 관계형 데이터베이스와 일반 파일을 이용할 때 보다 간단히 이루어진다.

예를 들어, 심폐지구력을 측정하기 위한 항목으로 최대산소섭취량(VO2Max)의 측정값 자체를 이용하다가 측정값을 몇 개의 범위로 구분한 등급으로 수정할 경우, 또는 심박수(heart rate)의 측정값의 세부항목 구분(운동전, 운동후, 안정시 등)을 삭제할 경우에는 XML 템플릿과 관련 스키마 파일에서 해당 요소를 수정하거나 삭제하면 된다. 관련 프로그램은 모든 요소에 대한 반복처리 과정에서 수정 또는 삭제된 항목에 영향을 받지 않고 자동적으로 동일하게 처리하므로 수정할 필요가 없다.

4. 실험적용 및 결과분석

4.1 체력유형분류 시스템의 개요

제안 기법의 유용성을 검증하기 위해 개인의 건강과 체력에 적합한 운동을 처방하는 전문가시스템에서 필요한 지식 베이스를 개발하고, 운동처방 프로세스 중 운동처방 대상자의 체력유형을 분류하는 프로그램을 작성하였다.

운동은 너무 가벼우면 충분한 효과를 얻을 수 없고 과도하면 오히려 위험이 따른다. 운동은 자신의 건강상태와 체력을 정확히 평가하고 그 결과에 따라 작성된 운동처방에 따라 운동을 실시해야 효과적으로 운동 목적을 달성할 수 있다[20-22]. 운동처방이란 체력의 향상과 건강의 유지 및 증진을 목적으로 개인의 체력수준, 건강상태, 연령 등을 고려하여 운동의 종류와 운동의 형식을 선택해 주고, 운동의 강도와 빈도를 제시하는 것으로, 설문조사 및 체력측정, 체력평가, 운동처방, 운동실시, 실시결과의 피드백 과정이 순환되는 프로세스로 진행된다[20, 21, 23].

운동처방대상자의 체력유형을 분류하는 시스템은 운동처방 프로세스 중 운동처방의 기준이 되는 체력평가 과정에서 활용될 수 있으며, 복잡한 전문지식이 요구된다.

본 연구에서 설계, 구현한 체력유형분류 시스템은 운동처방 대상자의 설문조사 및 체력측정 자료를 인터넷을 통하여 입력 받아 XML 템플릿으로 표현된 지식베이스를 이용하여 대상자의 체력유형을 분류한다. 지식베이스에는 운동처방대상자의 체력유형을 분류할 때 이용되는 표준 템플릿들이 저장되어 있다. 각 템플릿은 체력유형 분류 시 기준이 되는 요소들과 각 요소에 대한 속성과 값들이 XML로 표현되어 있다. XML로 작성된 템플릿은 인터넷을 통하여 저장, 조회, 수정할 수 있으며, 저장 및 수정 시에는 템플릿에 대한 XML 스키마를 이용하여 유효성 검사를 거친다. 체력유형분류 결과는 데이터베이스에 저장되어 운동처방 시 활용된다.

체력유형분류 시스템의 구조는 분류결과를 데이터베이스에 저장하는 부분을 제외하고는 (그림 1)에서 제시했던 일반적인 XML 템플릿 기반 지식관리 시스템의 구조와 동일하다. 운동처방 대상자의 체력유형을 분류하는 프로그램들은 ASP.NET을 이용하여 구현되었으며, 웹 브라우저를 통해 접근 가능하다.

4.2 체력유형 분류에 필요한 입력과 출력

운동처방 대상자의 체력유형을 분류하기 위해 시스템에 입력되는 자료는 <표 2>과 같이 크게 3가지 분류, 즉 건강상태에 관한 설문조사 자료, 건강관련 체력(health-related fitness) 측정 자료, 운동관련 체력(skill-related fitness) 측정 자료로 구분된다.

<표 2> 체력유형분류 시스템의 입력자료

입력자료 분류	입력자료 세부내역
건강상태에 관한 설문조사 자료	절대적 운동 금기 사항, 상대적 운동 금기 사항, 치료 중인 질병이나 복용중인 약물, 심폐계/대사계 질환의 징후 및 증상, 연령, 임신상태 등
건강관련 체력 측정 자료	신체질량지수, 체지방, 심폐지구력, 혈압, 유연성, 근력, 근지구력 등
운동관련 체력 측정 자료	민첩성, 평형성, 조정력, 반응시간 등

운동처방 대상자의 체력을 분류하여 얻어지는 출력은 대상자의 입력 특징에 따라 19가지의 체력유형 중 하나의 유형으로 분류된다. 이 19종의 최종 목적 패턴은 운동처방 전문가의 자문결과에 따라 설정한 것으로, 운동금기군(Contraindication Group) 2종, 질환군(Disease Group) 4종, 질환잠재군(Borderline Group) 3종, 특수대상군(Special Group) 4종, 건강군(Healthy Group) 6종으로 분류되어 있다.

4.3 체력유형분류를 위한 지식베이스

XML 템플릿으로 표현된 지식베이스에는 체력유형을 분류할 때 이용되는 유형별 표준 템플릿들이 저장되어 있다.

```

<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" targetNamespace="ns:subjectPattern" xmlns:sp="ns:subjectPattern">
  <xsd:simpleType name="class1">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="Contraindication Group" />
      <xsd:enumeration value="Disease Group" />
      <xsd:enumeration value="Borderline Group" />
      <xsd:enumeration value="Special Group" />
      <xsd:enumeration value="Healthy Group" />
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:complexType name="cardioendurance">
    <xsd:sequence>
      <xsd:element name="VO2max" type="sp:VO2range" />
      <xsd:element name="HR" />
    </xsd:sequence>
  </xsd:complexType>
  <xsd:element name="SubjectPattern">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="TemplateName" />
        <xsd:element name="SubjectGroup">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="Level1Class" type="sp:class1" />
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="HealthStatus">
          <xsd:element />
        </xsd:element>
        <xsd:element name="HelathRelatedFitness">
          <xsd:element />
        </xsd:element>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>

```

(그림 6) XML 스키마의 예

각 템플릿은 체력유형 분류 시 기준이 되는 요소들과 각 요소에 대한 값들이 XML로 표현되어 있다. 메타 템플릿에는 패턴 매칭 시의 템플릿들의 적용 순서 등 표준 템플릿들간의 연관관계에 대한 내용이 기록되어 있다.

(그림 6)은 체력유형 분류를 위한 템플릿에 대한 스키마의 일부이다. 이 스키마에는 체력유형 분류의 세부 내역(class1, class2, class3)과 체력유형 분류 시 기준이 되는 요소들 즉, 건강상태(HealthStatus), 건강관련 체력(Health-RelatedFitness), 운동관련 체력(SkillRelatedFitness)에 관한 세부 항목들에 대한 계층적인 구조와 각 요소에 대한 속성 및 값들의 유효한 범위가 정의되어 있다. 이 XML 스키마는 XML 템플릿을 저장할 때 유효성 검증의 기준으로 이용된다.

```
<?xml version="1.0"?>
<SubjectPattern>
<TemplateName>SP510</TemplateName>
<SubjectGroup>
<Level1class>Healthy Group</Level1class>
<Level2class>Fitness Strengthening Group</Level2class>
<Level3class>Cardiopulmonary Strengthening
Group</Level3class>
</SubjectGroup>
<HealthStatus>
<AbsoluteContraindications Weight="0.08"
Sibling_Relation='AND'>No</AbsoluteContraindications>
<RelativeContraindications Weight="0.08"
Sibling_Relation='AND'>No</RelativeContraindications>
<KnownDiseasesMedicationUse Weight="0.08"
Sibling_Relation='AND'>No</KnownDiseasesMedicationUse>
...
</HealthStatus>
<HealthRelatedFitness>
<BodyComposition>
<BMI Weight="0.09" Sibling_Relation='OR'>less than 25</BMI>
<PercentBodyFat Weight="0.09" Sibling_Relation='OR'> less than
25%</PercentBodyFat>
</BodyComposition>
<CardiorespiratoryEndurance>
<VO2max Weight="0.09" Sibling_Relation='AND'>not more than
C grade </VO2 max>
...
</CardiorespiratoryEndurance>
...
</HealthRelatedFitness>
<SkillRelatedFitness>
...
</SkillRelatedFitness>
</SubjectPattern>
```

(그림 7) XML 템플릿의 예

각 체력유형은 하나의 XML 템플릿으로 작성되어 있다. (그림 7)은 XML 템플릿의 한 예로, 체력유형 분류를 위해 만들어진 템플릿들 중 심폐기능 강화가 특히 요구되는 심폐기능 강화군의 체력유형에 대한 템플릿의 일부이다. 이 템플릿에는 체력유형 분류 시 기준이 되는 요소들(절대적 운동금지사항, 질병과 약물복용 등 건강상태에 관한 사항들, 신체질량지수, 체지방 등 건강관련 체력에 관한 사항들, 민첩성, 평형성 등 운동관련 체력에 관한 사항들)과 각 요소에 대한 속성과 값들이 XML로 표현되어 있다.

운동처방 대상자의 체력유형 추론에 필요한 규칙은 체력유형별 템플릿 내 요소의 속성과 값들에 내포되어 있다. 각 체력유형에 해당되기 위해 필수적으로 요구되는 항목기준들

은 템플릿 요소들로 정의되어 있고, 각 항목의 세부 규칙은 요소의 속성과 값으로 표현되어 있다. 예를 들어, 각 항목이 체력유형에 영향을 미치는 정도는 'Weight'라는 속성으로, 관련된 같은 수준의 항목과의 논리적인 관계는 'Sibling_Relation'이라는 속성으로 표시되어 있다. 프로그램에서 처리 시는 템플릿을 순서대로 읽으면서 알고리즘에 따라 처리하면 된다.

(그림 8)은 XML 메타 템플릿의 일부분으로, 패턴 매칭 작업 시의 템플릿 적용 순서(TemplateSequence)가 XML로 표현되어 있다. (그림 8)의 예제에서 템플릿 적용 순서는 템플릿 이름(TemplateName)으로 제시되어 있으며, 템플릿 SP110, 템플릿 SP120, 템플릿 SP210, ... 과 같은 순서로 적용하라는 것을 의미한다.

```
<?xml version="1.0"?>
<MetaTemplateForSubjectClassification>
<TemplateSequence>
<TemplateName>SP110</TemplateName>
<TemplateName>SP120</TemplateName>
<TemplateName>SP210</TemplateName>
...
</TemplateSequence>
</MetaTemplateForSubjectClassification>
```

(그림 8) XML 메타 템플릿의 예

4.4 실험 결과 및 분석

4.4.1 체력유형분류

실험을 위한 입력자료(체력유형분류 대상자의 건강상태 및 체력에 관한 자료로, 총 30가지의 항목으로 구성되어 있음)는 운동처방 전문가의 자문결과를 기초로 하여 입력항목 별로 예상되는 입력 값을 임의로 생성하였다.

(표 3) 체력유형분류 결과

대분류	중/소분류	빈도 (%)
운동금지군	- 절대적 운동금지군	45(0.90)
	- 상대적 운동금지군	44(0.88)
질환군	- 대사계 질환군	53(1.06)
	- 심혈관계 질환군	50(1.00)
	- 호흡기계 질환군	52(1.04)
	- 근골격계 질환군	40(0.80)
질환잠재군	- 심폐계 및 대사계 질환잠재군	153(3.06)
	- 근골격계계 질환잠재군	2(0.04)
	- 비만군	1,499(29.98)
특수대상군	- 임신부	25(0.50)
	- 노인	448(8.96)
	- 어린이	147(2.94)
	- 단순비만군	1,705(34.10)
건강군	(체중조절군)	
	- 체중과다군	452(9.04)
	- 체중과소군	8(0.16)
	(체력강화군)	
	- 심폐강화군	22(0.44)
	- 근강화군	28(0.56)
- 유연성강화군	2(0.04)	
- 체력회복군	0(0.00)	
분류 (소계)		4,775(95.50)
분류 안됨		225(4.50)
계		5,000(100.00)

이렇게 임의로 생성한 5,000건의 입력 자료에 대하여 시스템의 기능을 시험한 결과, 95.5%의 입력 자료에 대해서는 대상자의 체력유형을 분류하였고, 4.5%의 입력 자료에 대해서는 체력유형을 분류하지 못하였다<표 3>.

5,000건의 입력자료 중 체력유형을 분류하지 못한 225건(4.5%)은 입력항목별 예상되는 범위의 입력 값을 임의로 생성함에 따라 발생한 결과였다. 즉, 입력자료의 값이 일반적인 상황인 아닌 특이한 경우로, 본 연구에서 작성한 지식베이스(체력유형별 표준 템플릿에 정의된 사실과 추론규칙들)가 대처하지 못하는 경우였다. 입력 항목 각각에 대해서 예상되는 범위의 값을 임의로 생성하였으나, 30가지의 입력 항목들이 조합되어 하나의 입력자료가 만들어지면서 특이한 상황의 자료가 만들어졌다.

따라서 본 연구에서 사용한 입력자료를 가지고 우리가 제안한 XML 템플릿을 기반으로 한 방법이 아닌 기존의 생성 규칙을 이용한 방법을 이용할 경우에도 같은 결과가 초래될 것이다. 그러나 기존의 패턴분류 방법 중 가장 가까운 유형으로 분류해 주는 방법을 이용한다면 모두 분류될 수 있을 것이다.

향후 이 문제를 해결하기 위해서는 모든 경우에 대처할 수 있도록 지식베이스의 내용(체력유형별 표준 템플릿에 정의된 사실과 추론규칙들)을 보완하거나, 패턴매칭에서 분류되지 않은 경우, 가장 가까운 체력유형으로 분류해 주는 보완적인 방법을 추가해야 할 것이다.

4.4.2 지식표현과 관리측면

운동처방대상자의 체력을 평가하기 위해 필요한 자료의 종류가 많은 편이고 각 항목에 대한 기준 값들은 일반화된 지식뿐만 아니라 경험에 의한 지식도 많다. 따라서 관련 전문가들의 자문을 받을 때마다 수정 또는 추가해야 하는 항목들이 많았으나, XML 템플릿을 기반으로 한 제안 방법에서는 지식의 표현과 변경을 쉽게 처리할 수 있었다.

예를 들어, 체력유형분류에 필요한 기준 항목으로 현재 알고 있는 질병이나 약물복용 여부에 대한 항목(Known DiseasesMedicationUse)을 추가해야 할 경우에, 아래와 같이 해당 XML 템플릿에서 건강상태를 나타내는 정보를 가지고 있는 XML 요소를 찾아 하위 요소로 해당 항목을 추가하기만 하면 된다.

```
<HealthStatus>
...
<KnownDiseasesMedicationUse Weight = '0.08'
Sibling_Relation = 'AND'> No </KnownDiseasesMedicationUse>
...
</HealthStatus>
```

해당 XML 스키마에도 해당 부분을 추가하지만 다른 부분에는 영향을 미치지 않는다. 새로운 유형의 지식 추가와 관련된 처리프로그램은 모든 요소에 대한 반복처리 과정에

서 추가된 항목에 대한 처리도 자동적으로 동일하게 처리하므로 프로그램을 수정할 필요가 없다.

지식을 추론하는 과정에서는 체력유형별로 표준 템플릿을 만들어 두고 새로운 대상자의 입력 자료가 입력되었을 때 표준 템플릿과 비교하는 패턴매칭 방법을 이용하였다. 용도별로 지식을 템플릿화하면서 해당 패턴을 규정하는 규칙의 대부분이 템플릿 내에 포함되어 정의됨에 따라 매칭 알고리즘은 (그림 5)에서 볼 수 있었듯이 단순하게 만들어졌다. 또한 템플릿의 적용순서는 일반 템플릿들의 상위에 존재하는 메타 템플릿에 정의하여 구조적으로 접근할 수 있었다.

지식표현과 관리측면에서 살펴본 정성적인 실험결과를 토대로 하여, 본 논문에서 제안한 XML 템플릿을 기반으로 한 방법을, 기존의 지식표현 방법들<표 1> 중, 복잡한 지식 표현에 있어 대표적인 방법인, 객체지향기법과 생성규칙을 혼합한 방법과 비교하였다. <표 4>는 비교 결과를 요약한 것이다.

<표 4> 기존 지식표현 방법과 비교분석

방법	장점	단점
객체지향기법과 생성규칙을 혼합한 방법 (기존방법)	- 인간이 생각하는 자연스러운 방식으로 문제를 표현하므로, 개발자와 사용자가 쉽게 표현하고 이해할 수 있다. - 시스템의 유지보수성과 확장성을 증대시킨다.	- 도구 부족 - 문제해결 시 제어흐름 추적이 어렵다. - 모든 규칙들에 대해 비교과정을 거치므로 비효율적이다.
XML 템플릿 기반 방법 (제안방법)	- 기존의 객체지향기법과 생성규칙을 혼합한 방법의 장점을 모두 가지고 있다. - 다량의 복잡한 지식을 용도별 또는 내용별로 구분하기가 쉽다. - XML 기술의 발달로 도구가 많은 편이다. - 메타템플릿과 패턴매칭 방법을 이용하여, 제어흐름을 추적하기 쉽고 처리과정이 효율적이다.	- 하나의 템플릿 안에 사실과 규칙이 함께 표현되어 있어 복잡해 질 수 있다.

XML 템플릿은 객체의 그룹으로 볼 수 있다. 각 XML 요소가 하나의 객체이고 그 객체들이 구조적으로 그룹화되어 있으며, 지식을 용도별 또는 내용별로 분류하여 별도의 XML 템플릿으로 표현한다. 따라서 XML 템플릿 기반 지식표현 방법은 객체지향기법이 가지고 있는 장점은 모두 가지고 있으며, 복잡한 지식을 보다 쉽게 용도별 또는 내용별로 구분할 수 있다.

또한 기존의 객체지향기법과 규칙을 혼합한 방법이 쉽게 이용할 수 있는 도구가 부족한 편인데 반해, 제안 방법은 XML 기술을 기반으로 함에 따라, XML 기술의 발달과 함께 지식표현과 관리를 위해 이용할 수 있는 도구가 많은 편이며 계속 발전되고 있다.

기존의 객체지향기법과 규칙을 혼합한 방법이 가지고 있는 또 다른 단점은 문제해결 시 제어흐름 추적이 어렵고, 모든 규칙들에 대해 비교, 선택, 처리과정을 거치므로 비효율적이라는 점이다. 이에 반해, XML 템플릿 기반 방법에서는 메타템플릿을 이용하여 구조적으로 관리하고, 지식 추론과

정에서도 패턴매칭 방법을 이용하여 알고리즘이 단순하고 패턴을 발견할 때까지만 비교하면 되므로, 상대적으로 제어 흐름을 추적하기가 쉽고 처리과정이 효율적이다.

하나의 XML 템플릿 안에는 사실과 규칙이 함께 표현되어 있다. 이는 지식표현과 관리 측면에서 많은 장점을 주고 있지만, 상대적으로 경우에 따라 복잡해지게 만들 수도 있다.

5. 결론 및 향후 과제

본 논문에서 제안한 지식베이스 관리 기법은 복잡하며 시간에 따라 변할 수 있는 전문지식을 좀더 효율적으로 표현하고 사용하기 위한 것이다. 제안 방법에서는 복잡하고 다양한 정보를 구조적으로 표현하는 데 효과적인 XML을 이용하면서, XML 문서의 종류와 개수가 많아지면서 발생하는 문서관리상의 비효율성 문제를 해결하기 위하여, 복잡한 지식을 XML로 표현된 다수의 템플릿으로 분류하여 작성하고, XML 템플릿들 사이의 연관관계 등을 정의한 메타템플릿을 이용하여 구조적으로 관리한다. 제안 방법의 유용성을 입증하기 위하여 개인의 건강과 체력에 적합한 운동을 처방하는 전문가시스템에 필요한 지식베이스를 만들고 운동처방 대상자의 체력유형을 분류하는 활용 프로그램을 작성한 후 그 결과를 분석하였다.

실험을 통해 본 논문에서 제안하는 지식베이스 관리 기법은 XML을 이용하여 지식을 표현함으로써 복잡한 지식을 구조적으로 표현하고 시간에 따라 자주 변할 수 있는 동적인 부분에 적용할 수 있으며, 필요에 따라 지식을 여러 가지의 템플릿으로 구분하여 표현함으로써 전문가 시스템에서 지식표현의 어려운 부분을 극복할 수 있음이 확인되었다. XML, 템플릿으로 표현된 지식은 이해하기가 쉬워, 전문가들이 엔지니어의 도움을 받지 않고도 시스템을 수정할 수 있을 것으로 보이며, 이는 지식기반시스템의 발전 방향에도 부합한다. 또한 템플릿들간의 연관관계 등에 관한 내용을 상위 개념의 메타 템플릿으로 나타냄으로써 추론 시 이용되는 규칙을 구조적으로 표현하고 다양한 구조적인 추론이 가능하도록 하였다.

제안 방법에서 저장되거나 출력되는 모든 정보는 XML 형태로 만들어지므로 시스템의 기능을 확장하거나 다른 시스템과 연계 시 XML 문서로 정보교환이 가능하며, 따라서 손쉽게 시스템의 호환성을 확보할 수 있을 것이다.

제안 방법의 유용성을 입증하기 위해 설계, 구현한 운동처방 대상자의 체력유형을 분류하는 시스템에서는 추론을 위한 규칙의 대부분이 템플릿 자체에 포함되도록 만들어져서 다양한 규칙 표현이 필요하지 않았다. 제안 방법이 보편적으로 적용될 수 있기 위해서는 좀 더 다양한 방식의 추론이 필요한 전문가시스템에 적용하여 실험해 보는 작업이 추가로 필요하다. 이 부분은 본 논문에서 미해결 사항으로 남아 있으며, 향후 다양한 대상 시스템에 적용하여 실험해 보고 그 결과를 분석하는 작업이 연구과제로 남아 있다.

또한 운동처방 대상자의 체력유형을 분류하는 시스템의 기능 테스트에서는 대량의 임상자료 확보의 어려움으로 각 입력 항목별 예상되는 값을 임의로 생성하여 시험하였다. 향후 이 시스템을 운동처방업무에 사용하기 위해서는 실제

임상자료를 가지고 추가로 테스트하는 작업이 필요하다.

참 고 문 헌

- [1] William R. Swartout, Future directions in knowledge-based systems, ACM Computing Surveys Vol.28, Issue 4es, Dec., 1996.
- [2] 윤경배, 최준혁, 왕창중, "하이브리드 SOM을 이용한 효율적인 지식베이스 관리", 정보처리학회논문지B, 제9-B권 제5권, pp.635-642, 2002.
- [3] J. Giarratano, G. Riley, Expert Systems Principles and Programming, 3rd ed., pp.1-157, PWS Publishing Company, 1998.
- [4] P. Jackson, Introduction to Expert Systems, 3rd ed., pp.6-258, Addison Wesley, 1999.
- [5] G. F. Luger and W. A. Stubblefield, Artificial Intelligence, 3rd ed., pp.203-246, Addison Wesley, 1998.
- [6] 김화수, 조용범, 최종욱, 전문가시스템, pp.36-70, 집문당, 1998.
- [7] 이재규, 최형림, 김현수, 서민수, 주식진, 지원철, 전문가시스템 원리와 개발, pp.18-165, 법영사, 1998.
- [8] M. Cannataro, A. Guzzo and A. Pugliese, "Knowledge Management and XML : Derivation of Synthetic Views over Semi-structured Data," ACM SIGAPP Applied Computing Review, Vol.10, Issue 1, pp.32-36, 2002.
- [9] P. J. Wagner and T. K. Moore, "Integrating XML into a Database Systems Course," Proc. of SIGCSE 2003, pp. 26-30, 2003.
- [10] D. Hunter, et al. Beginning XML, Wrox Press, 2000.
- [11] E. Damiani, V. Elia and M. Madravio, "Guest Editorial," ACM SIGAPP Applied Computing Review, Vol.10, Issue 1, pp.32-32, Spring, 2002.
- [12] I. Manolescu, D. Florescu, D. Kossmann, F. Xhumari and D. Olteanu, "Agora : Living with XML and Relational," Proc. of the 26th VLDB Conference, pp.623-626, 2000.
- [13] P. Selinger, "Information Integration and XML in IBM's DB2," Proc. of the 28th VLDB Conference, 2002.
- [14] A. Nierman and H. V. Jagadish, "ProTDB : Probabilistic Data in XML," Proc. of the 28th VLDB Conference, pp. 646-657, 2002.
- [15] J. S. Key and R. K. Wong, "Structural Inference for Semi-structured Data," Proc. of the 10th International Conference on Information and Knowledge Management, pp.159-166, 2001.
- [16] H. L. Van, "A system based on XML for Supporting the Management of educational Web Sites," ACM SIGAPP Applied Computing Review, Vol.10, Issue 1, pp.37-42, 2002.
- [17] F. Niccolucci, "XML and Humanities Computing," ACM SIGAPP, Applied Computing Review, Vol.10, Issue 1, pp. 43-47, 2002
- [18] Andrea Marchetti, Fabrizio Megale, Enrico Seta, Fabio Vitali, Using XML as a means to access legislative documents : Italian and foreign experiences, ACM SIGAPP, Applied Computing Review, Vol.10, Issue 1, pp.54-62, 2002.

- [19] Andrea Trentini, "A Java-based framework to support computer-assisted creation of structured XML documents," ACM SIGAIPP, Applied Computing Review, Vol.10, Issue 1, pp.48-53, 2002.
- [20] American College of Sports Medicine, ACSM's Resource Manual for Guidelines for Exercise Testing and Prescription, 4th ed. Lippincott Williams & Wilkins, 2001.
- [21] D. C. Nieman, Exercise Testing and Prescription : A Health-Related Approach, 4th ed., pp.1-46. Mayfield Publishing Company, 1999.
- [22] 유승희, 박수연, 현대인의 건강관리를 위한 운동처방, 도서출판 태근, 2000.
- [23] 체육과학연구원, 전문가를 위한최신 운동처방론, 도서출판 21세기 교육사, 2000.



이혜자

e-mail : hjlee@ysc.ac.kr
 1982년 서울대학교 간호학과(학사)
 1989년 연세대학교 산업대학원 전산전공
 (공학석사)
 1982년~1996년 (주)데이콤, (주)삼보
 컴퓨터등 근무

1996년~1998년 한국보건의료관리연구원 수석연구원
 1998년~현재 용인송담대학 의료정보시스템과 조교수
 1999년~현재 경희대학교 대학원 컴퓨터 공학과(박사과정)
 관심분야 : 의료정보시스템, 데이터베이스, 정보 모델링, 전문가
 시스템, 지식관리 등

정병수



e-mail : jeong@nms.kyunghee.ac.kr
 1983년 서울대학교 전자계산기공학과
 (공학사)
 1985년 한국과학기술원 전산학과(석사)
 1995년 Georgia Institute of Technology,
 College of Computing(박사)

1985년~1989년 한국데이터통신(주) 정보통신연구소 선임연구원
 1996년~현재 경희대학교 전자정보학부 부교수
 관심분야 : 병렬 데이터베이스, 실시간 데이터베이스

박승훈



e-mail : parksh@khu.ac.kr
 1981년 서울대학교 전기공학과(학사)
 1984년 서울대학교 대학원 제어계측
 공학과(공학석사)
 1990년 미국 플로리다대학교 전기공학과
 (공학박사)

1985년~1990년 한국전자통신연구원 연구원
 1991년~2000년 건국대학교 의공학과 부교수
 1996년~1998년 한국전자통신연구원 초빙연구원
 1997년~1999년 건국대학교 청주캠퍼스 전산소장
 2000년~현재 경희대학교 동서의료공학과 교수
 관심분야 : 인터넷 기반 의료정보시스템, 의료용 가상현실, 생체
 신호 모니터링 및 해석 등