

---

# ADL기반의 학습수준별 동적 콘텐츠 구성

## ADL based Construction of Dynamic Contents for Learner's Tailoring Learning

---

정화영\*, 홍봉화\*\*

경희대학교 교양학부\*, 경희사이버대학교 정보통신학과\*\*

Hwa-Young Jeong(hyjeong@khu.ac.kr)\*, Bong-Hwa Hong(bhhong@khcu.ac.kr)\*\*

---

### 요약

많은 학습시스템이 학습자의 학습효율을 높이기 위하여 학습 난이도를 평가 및 적용하고 있다. 그러나 대부분의 적용방법들은 학습 전 또는 학습 후 학습결과를 분석할 때 산출되어 학습자가 학습도중에 난이도를 변경하면서 학습 콘텐츠를 제공받기 어렵다. 본 연구에서는 학습자의 수준을 학습 과정 중에 적용하고, 이 수준에 맞는 학습 콘텐츠를 다음 학습에 바로 적용시키는 방법을 제시하였다. 이를 통해 학습자는 사전 검사를 통해 설정된 난이도에 따라 다음 학습 콘텐츠를 제공 받을 수 있어 학습향상에 도움이 될 수 있도록 하였다.

■ 중심어 : | 이러닝 | 학습수준 | ADL | 동적 콘텐츠 |

### Abstract

A lot of learning systems are applying and verify evaluating the item difficulty to increase learner's learning effect. But most of this methods were calculated the item difficulty when it analyze the learning result before or after learning. So, it is hard to support learning contents with changing item difficulty during learning to learner. In this research, we proposed the method that system can support learning contents to next learning to fit learner's level immediately as apply to calculate item difficulty during the proceed learning. Through this method, learner could supported learning contents by calculated difficulty through pre-test and it caused this method was helped learner to increase learning effect.

■ keyword : | E-Learning | Learning Level | ADL | Dynamic Contents |

---

## I. 서론

e-Learning은 기존의 교실 위주의 수업이 온라인을 통한 개별적 학습 공간에서 이루어지는 교육 방식으로 학습자는 개인의 자율성과 창의력이 존중되는 한편, 학습자 중심의 자기주도형 학습이 이루어진다[1]. 이때 면대면 학습과 같이 교수자의 실시간의 직접적인 지도를

받기 어려운 웹 기반 학습에서 학습자의 학습효율을 향상 시키려는 노력은 다양한 기법을 활용하여 전개되고 있다. 그 중 학습효과를 극대화 하는 방법으로 학습자의 수준별 학습을 들 수 있다. 수준별 학습의 중요성은 오프라인 학습에서도 나타나 있으며, 학습자의 학습능력 향상을 위해서는 반드시 필요하다는 자료를 소명하고 있다[2-4]. 그러나 웹 기반 학습에서는 다양하고 방

대한 학습 콘텐츠의 적용과 학습자의 학습수준이 일정하지 않기 때문에 수준별 학습을 진행하기가 매우 어렵다. 그래서 학습자의 학습 수준을 판별하고 적용하기 위한 방법으로 문항분석이론, 문항반응이론 또는 컴퓨터 기반 검사(CBT: Computer Based Testing)등이 시행되고 있다. 그러나 문항분석이론이나 문항반응이론은 학습문항에 대해 학습자의 정, 오답을 기반으로 분석하는 기법으로 매우 까다로운 과정 및 절차를 수행하여야만 한다. 또한 컴퓨터 기반 검사는 단순히 학습자의 답안만을 기초로 학습내용의 이해도를 판정하면서 너무 단순한 논리에 의해 학습수준이 판정되는 일이 많다. 또한 학습자의 학습수준이 판정되면 학습이 종료되고 다음 학습문제가 제시될 때 까지 결정된 학습 수준이 유지된다. 이러한 문제는 학습자가 학습과정 중 이해하고 있는 내용이 있을 경우에도 획일적인 학습과정에 따라 학습자의 선택 없이 학습이 이루어지게 되는 문제점을 가진다. 따라서 학습자의 학습과정이 진행 중에도 수시로 학습과정이나 학습 콘텐츠가 학습자에 맞춰 변경이 가능하도록 하는 기능의 지원이 필요하다. 또한 많은 학습 시스템들이 정확한 학습 프로세스의 명세 없이 개발되고 있어 추후의 시스템 변경이 어렵고 유지보수에 많은 시간이 필요하게 된다.

본 연구에서는 학습자의 학습수준에 따른 동적 학습 콘텐츠를 구성하는 방안을 제시하고자 한다. 이는 학습이 진행 중에도 수시로 각 단계마다 학습자에게 학습문항을 제시하여 다음 학습의 학습 콘텐츠를 결정하도록 하였다. 또한 학습 프로세스는 프로세스의 정형기법인 ADL(Architecture Description Language)를 통해 기술함으로써 각 프로세스들의 연결 관계를 명확히 하고, 이에 대한 처리를 정형화함으로써 추후 수정이나 유지보수에 효율적으로 대처하게 하였다. 본 연구는 서론에 이어 제2장에서 수준별 학습, ADL과 기존연구사례의 관련연구를 기술하였고, 제3장에서는 ADL을 통해 학습자 수준별 동적 학습 콘텐츠 구성을 기술하였으며, 제4장에서는 본 연구의 적용 및 결과를 제시하여 제5장에서 결론을 맺었다.

## II. 관련 연구

### 1. 수준별 학습

제 7 차 교육과정의 특징 중의 하나는 수준별 교육과정의 도입이라고 할 수 있다. 수준별 교육과정은 교과 성격이나 학년의 특징에 따라 단계형, 심화·보충형, 그리고 과목 선택형 수준별 교육과정으로 편성, 운영된다. 그러나 수준별 교육과정에서 제시하는 심화·보충의 개념은 절대적이지 아니라 창의적이고 융통성 있게 운영 할 수 있다[5]. 이러한 개념은 학습자의 개인차에 기인하는 것으로 학습속도, 학습능력, 학업성취도, 흥미나 적성 혹은 진로에 따른 수준이 구분될 수 있으며 다음과 같이 제시된다[6].

- '학습속도'를 기준으로 수준을 나눈다면, 보충 과정 대상자에게는 학습 능력이 부족한 학생이 아니라 학습속도가 상대적으로 느린 학생이라 볼 수 있다.
- '학습능력'을 기준으로 수준을 나눈다면, 보충 과정 대상자에게는 기본 과정의 내용을 보다 쉽게 구성하여 제시하여야 할 것이다.
- '학업성취도'에 따라 수준을 나눈다면, 학업 성취도에 도달한 학생에게는 심화과정을, 학업 성취도에 미달한 학생에게는 보충과정을 제공할 수 있다. 보충과정 학생에게는 학습 결손 요인에 따라 학습 시간을 더 제공하거나 보다 난이도를 낮추어 제공하여야 할 것이다.
- '흥미·적성·진로'에 따라 수준을 나눌 수 있다. 과목 선택형 수준별 교육과정은 수준의 개념을 학습 속도, 학습 능력, 학업 성취도로 한정하지 않고, 흥미와 적성, 진로까지 확대하여 적용한 것이다.

컴퓨터 기반으로 학습자의 학습 수준을 평가하는 방법은 학습문항을 통해 학습자의 수준을 검사한다. 이는 전적으로 출제자 1인의 전문성에 의존해서 개발된 문항들은 문항 고유번호와 출제자가 제시한 난이도 및 문항 관련 정보 등을 가지고 문제은행에 등록된다. 이들 문항은 필요시 컴퓨터화 된 무선 추출에 의해 자동적으로 검사지가 구성되며, 검사 개발자에 의해 분할점이 결정된다[7].

2. ADL

ADL은 소프트웨어 시스템 모델링 컨셉 구조를 위한 요소들을 제공하는 언어이며[8], 정형방법으로 소프트웨어 구조를 기술하고 특정 소스 모듈의 세부사항을 구현하는 것 보다 더 높은 단계의 소프트웨어 설계를 표현한다[9]. 그림 1은 ADL의 하나인 Wright[10]의 간단한 예를 나타낸다.



Style Client-Server

Component Client

Port p = request → reply → p∩§

Computation = internalCompute →

p.request → p.reply → Computation∩§

Component Server

Port p = request → reply → p∩§

Computation = p.request →

internalCompute → p.reply →

Computation ∩ §

Connector Link

Role c = request → reply → c∩§

Role s = request → reply → s∩§

Glue = c.request → s.request → Glue

∩ s.reply → c.reply → Glue

∩ §

Constraints

∃!s ∈ Component, ∀c ∈ Component

: TypeServer(s) ∧ TypeClient(c) ⇒

connected(c,s)

EndStyle

Configuration Simple

Style Client-Server

Instances C : Client ; L : Link ; S : Server

Attachments C.p as L.c ; S.p as L.s

EndConfiguration

그림 1. Wright의 클라이언트-서버 예

이때 request는 이벤트의 발생을 나타내며, ∩은 프로세스의 내부선택, ∩은 프로세스의 외부선택을 나타내고, 프로세스의 §은 종료를 나타낸다. 이때 Role은 커넥터의 역할을, Glue는 연결에 대한 프로세스를 나타낸다.

3. 기존연구사례

수준별 학습에 대한 기존 연구에서, 이명희의 연구 [11]는 다양한 멀티미디어 학습 콘텐츠와 문항을 준비하였고, 이를 학습자에게 사전검사를 시행하여 학습자의 수준을 판별하였다. 이때 학습결과 분석은 문항반응 이론을 사용하였으며, 학습자의 수준이 판별되면 해당 학습 콘텐츠를 제공하는 방식을 제시하였다. 그러나 이는 사전검사를 통해 학습자의 학습수준이 결정되면 정해진 학습과정에 따라 나머지 학습이 획일적으로 이루어지고 있어 학습자가 이해하고 있는 부분이 있을 경우라도 정해진 과정에 따라 학습을 하여야만 하는 문제점이 있다. 또한 학습 시스템도 각 프로세스의 명확한 규명이 없이 전통적인 방법으로 개발되어, 학습 프로세스의 운영오류나 신뢰성을 검증하기 어렵고 유지보수도 어렵다. 김정렬의 연구[5]에서는 각 차시별 학습단계와 내용 및 학습활동을 명시하였고 주어진 과제수행을 통해 보충집단과 심화집단으로 분류하여 학습이 이루어지도록 하였다. 이때 과제를 상, 중, 하의 난이도에 따라 제시함으로써 학습자의 학습수준을 판정하여 학습을 지원하였다. 그러나 이는 학습자 개개인의 특성을 반영하기 보다는 사전 과제 수행의 형태를 검사하여 보충 또는 심화집단으로 그룹화 함으로서 학습자 개개인이 부족한 학습내용을 보완하기 어려우며, 과제에서도 상, 중, 하의 단순한 난이도를 가짐으로서 학습자의 학습능력이나 이해도를 측정하는데 어려움이 있다. 전영주의 연구[2]에서는 학습자의 학습수준을 평가하기 위하여 학습 전 듣기평가를 실시하였다. 이를 통해 학습 수준이 결정되면 보충, 기본, 심화단계의 난이도를 가진 학습 콘텐츠 중 해당 수준에 맞는 학습 콘텐츠를 제공하는 방식을 제시하였다. 그러나 이는 사전 듣기평가를 통해 한번 학습수준이 결정되면 나머지 학습 콘텐츠는 분류된 학습단계의 내용을 제공받아 학습하게 되며, 학

습과정 중 학습 수준의 변경이 어려웠고, 적용된 학습 수준이 너무 단순하여 학습자에게 정확한 학습수준에 맞는 학습을 제공하기 어려웠다. 따라서 학습수준에 맞는 학습 콘텐츠를 제공하기 위해서는 다양한 단계의 난이도를 가진 학습 콘텐츠와 문항을 제공하고, 학습자의 학습과정에서도 다음 학습의 내용을 학습자가 이해하고 있다면 난이도를 조정하여 학습자의 수준에 맞는 학습을 제공하는 것이 필요하다.

### III. 학습자 수준별 동적 학습 콘텐츠 구성

#### 1. 학습자 수준에 맞는 학습 콘텐츠 구성 과정

제 7 차 교본 연구는 학습자 수준에 맞는 학습 콘텐츠를 구성하여 학습효율을 향상시키는 방법을 제시하고자 한다. 학습자의 수준을 평가하는 방법으로는 문항난이도를 이용하였으며, 학습자가 학습을 진행 중 각 학습단계마다 사전검사를 반복적으로 적용하여 학습자의 학습 내용 이해도에 따라 학습 콘텐츠를 차별적으로 적용하였다. 이를 위한 처리과정은 [그림 2]와 같다. 이때 학습자는 학습하고자 하는 학습단원과 난이도를 선택하게 되면 학습시스템은 미리 구성된 학습과정에 따라 학습 소단원이 설정된다. 다음 단계로 학습할 소단원의 문항을 무작위로 선택하여 제공되고 학습자가 문제를 풀면 그 결과를 문항난이도에 따라 분석하게 된다. 분석된 결과를 통해 학습자가 이해하고 있는 내용이라면, 초기에 학습자가 선택한 난이도보다 한 단계 높은 난이도의 문항이 다시 출제된다. 또 제시된 문항을 이해하고 있다면 다시 한 단계 높은 난이도의 문항이 출제된다. 만일 오답을 제시하였다면 현재 조정된 난이도를 학습자의 학습 수준으로 판정한다. 학습자의 수준이 결정되면 난이도에 맞는 학습콘텐츠가 제공되고 학습이 진행된다. 학습의 단원은 여러 개의 소단원으로 구분되는데 이때 소단원의 학습이 종료되고 다음 학습으로 전환될 때마다 현재의 소단원 학습 문항이 제시되어 학습된 내용의 이해도를 평가하고 다음 소단원 학습을 위해 현재 설정된 난이도의 다음 학습 소단원 문항이 제시된다. 이는 학습 전에 미리 시행하였던 사전검사와 학습

결과 분석의 과정을 반복적으로 다시 적용하고 다음 소단원의 학습 콘텐츠도, 사전검사에 의해 결정된 난이도에 따라 제시된다.

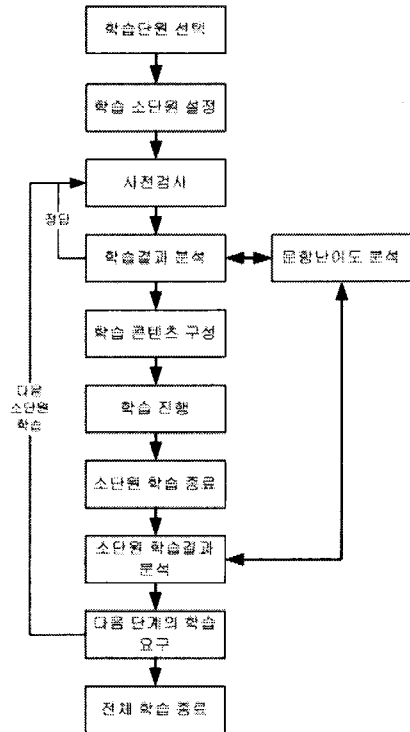


그림 2. 학습수준별 학습 콘텐츠 지원 처리과정

이러한 학습과정이 반복되면서 전체의 학습과정을 종료하고 학습결과를 학습자에게 제시하게 된다. 학습에 구성되는 학습 콘텐츠는 [표 1]과 같이 난이도에 따라 구성된다. 이러한 구성은 난이도가 높을수록 텍스트 기반의 학습 콘텐츠를 주로 이용함으로써 보다 많은 학습내용을 전달하고, 난이도가 낮을수록 학습자의 학습 이해를 돕기 위해 텍스트, 그림, 사운드, VOD 등 다양한 방법으로 쉽게 설명된 학습 콘텐츠를 이용하도록 하였다. 또한 학습문항에서도 쉽고 어려운 문항을 난이도에 따라 개발하여 학습자에게 학습 수준에 맞는 적절한 문항을 제시할 수 있도록 하였다.

표 1. 단원별 난이도에 따른 학습 콘텐츠 구성

단원	난이도	콘텐츠 구성	학습문항	
대단원	소단원 1	상	텍스트	어려운 문항
		중	텍스트, 그림, 사운드	일반문항
		하	텍스트, 그림, 사운드, VOD	쉬운 문항
소단원 2	상	텍스트	어려운 문항	
	중	텍스트, 그림, 사운드	일반문항	
	하	텍스트, 그림, 사운드, VOD	쉬운 문항	

2. ADL기반의 수준별 학습 프로세스

학습자의 학습 수준별 콘텐츠 지원을 위한 처리 프로세스는 컴포넌트로 설계되었다. [그림 3]은 각 프로세스에 대한 컴포넌트 다이어그램을 나타낸다.

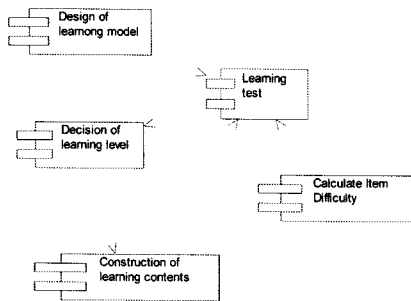


그림 3. 학습수준별 컴포넌트 다이어그램

Design of learning model에서는 학습단원 및 학습 난이도를 결정하게 된다. 결정된 난이도는 Learning test에 의해 사전검사를 수행하고 이는 문항난이도에 의해 분석된다. 학습자의 답안에 따라 사전검사와 난이도 분석을 반복하고 Decision of learning level에 의해 학습자의 학습수준을 결정하게 된다. Construction of learning contents에서는 결정된 학습 수준을 실제 학습 콘텐츠를 구성하여 학습자에게 제공하며, 소단원의 학습이 종료되면 다시 다음 학습을 위해 사전검사와 난이도 분석을 반복하게 된다. 컴포넌트 다이어그램을 통해 각 프로세스의 명세 설계는 ADL 중 Wright를 이용하여 다음과 같이 구성하였다.

Style LearningContents

Component Design\_of\_learning\_model

Port p = request → reply → p∩§

Computation =

Selectlearningcontents → p.selectlearning →

p.makelearningmodel → Computation∩§

□ Selectdifficulty → p.selectdifficulty →

p.makelearningmodel →

Computation∩§

Component Learning\_test

Port p = request → reply → p∩§

Computation =

Getquestionitem → p.getquestion →

p.makequestion → Computation∩§

□ Callitemdifficulty → p.itemdifficulty →

p.getitemdifficulty →

Computation∩§

Component Calculate\_Item\_Difficulty

Port p = request → reply → p∩§

Computation =

Getanswer → p.calculateitemdifficulty →

p.analysisitemdifficulty → Computation∩§

□ Getlearnerprofile → p.makeprofile →

p.modifylearnersinform →

Computation∩§

Component Decision\_of\_learning\_level

Port p = request → reply → p∩§

Computation =

Getitemdifficulty → p.getdifficulty →

p.makedifficultylevel → Computation∩§

Component Construction\_of\_learning\_contents

Port p = request → reply → p∩§

Computation =

Constructioncontents → p.findlearningcontents

→ p.makelearningcontents → Computation∩§

□ Nextquestion → p.makequestion →

p.createNextquestion → Computation∩§

LearningContents Binder

Role s1 = request → reply → s1∩§

```

Role s2 = request → reply → s2;§
Role s3 = request → reply → s3;§
Role s4 = request → reply → s4;§
Role s5 = request → reply → s5;§
Glue = s1.request → s1.Selectlearningcontents
      → s2.request → Glue □
      s1.request → s1.Selectdifficulty →
      s2.request → Glue □
      s2.request → s2.Getquestionitem →
      s2.request → Glue □
      s2.reply → s2.Callitemdifficulty →
      s3.request → Glue □
      s3.request → s3.Getanswer →
      s3.request → Glue □
      s3.reply → s3.Getlearnerprofile →
      s4.request → Glue □
      s4.reply → s4.Getitemdiffulty →
      s5.request → Glue □
      s5.request → s5.Constructioncontents →
      s5.request → Glue □
      s5.reply → s5.Nextquestion →
      s2.request → Glue □ §

```

Constraints

```

∃! s1 ∈ Design_of_learnong_model ;
∀s2 ∈ Learning_test ;
∀s3 ∈ Calculate_Item_Difficulty ;
∀s4 ∈ Decision_of_learning_level ;
∀s5 ∈ Construction_of_learning_contents
TypeComponent(s1) ∧ TypeComponent(s2)
...TypeComponent(s5) ∧ TypeComponent(s2)
⇒ connected(s1,s2) ; ... ;
connected(s5,s2)

```

EndStyle

Configuration LearningContentsConstruction

Style LearningContents

```

Instances S1 : Design_of_learnong_model ;
I : Interface ; S2 : Learning_test ;
S3 : Calculate_Item_Difficulty ;

```

```

S4 : Decision_of_learning_level ;
S5 : Construction_of_learning_contents
Attachments S1 as I ; I as S2 ; ... ;
Sn-1 as I ; I as Sn
EndConfiguration

```

#### IV. 적용 및 결과

본 시스템의 적용을 위하여 학습 모집단 20명을 대상으로 개발된 학습 시스템을 실행하였다. 학습 시스템은 영어 학습 중 명사절 부분을 제작하였으며, 소단원 별 학습 콘텐츠는 [표 2]와 같다. 각 소단원의 내용에서 명사절 1은 “That으로 인도되는 명사절”, 명사절 2는 “간접의문문”, 명사절 3은 ”명사절의 응용학습“을 나타낸다.

표 2. 예제 학습 콘텐츠의 구성

대단원	소단원	난이도	내용
명사절	명사절 1	상	텍스트
		중	텍스트, 그림, 사운드
		하	텍스트, 그림, 사운드, VOD
	명사절 2	상	텍스트
		중	텍스트, 그림, 사운드
		하	텍스트, 그림, 사운드, VOD
	명사절 3	상	텍스트
		중	텍스트, 그림, 사운드
		하	텍스트, 그림, 사운드, VOD

이때 VOD는 동영상 강좌를 나타내며, 학습의 이해를 돕기 위하여 자세한 설명을 포함하였다. [표 3]은 학습 모집단을 대상으로 학습을 실시한 후 학습과정 중에 변경 및 적용된 난이도를 나타낸다. 이를 통하여 학습자의 학습 수준이 초기에 결정되었을지라도 학습과정 중에 소단원 별 사전검사를 통하여 다음 학습의 난이도가 변경 및 적용됨을 알 수 있었고, 학습자는 다음 학습의 내용이 이미 이해하고 있거나 못할 경우 난이도가 변경된 학습 콘텐츠를 제공받고 있음을 알 수 있었다.

표 3. 학습 모집단의 소단원 별 학습 중 난이도 조정

소단원	학습자																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
명사절 1	상	하	중	하	중	중	중	상	중	하	하	중	상	중	하	하	중	하	하	상
명사절 2	상	중	중	하	상	중	중	상	중	중	하	상	상	중	하	하	중	중	하	상
명사절 3	중	중	상	중	상	중	상	상	중	중	하	상	중	상	중	중	중	상	하	중

[그림 4]는 각 학습자의 10회 학습 후의 평균점수를 산출하여 나타낸 것이다. 이를 통하여 각 학습자들이 본 학습을 통하여 평균 학력신장이 있었음을 알 수 있었다.

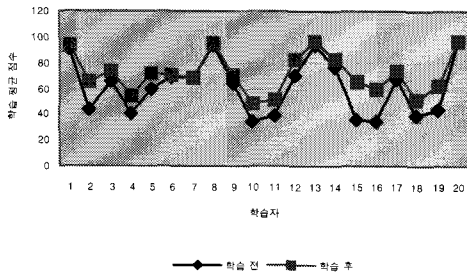


그림 4. 학습 적용 전·후의 비교

V. 결론

본 연구는 현재 많은 e-Learning 시스템에서 사용하는 학습난이도 산출 방법에서 학습 전이나 전체 학습의 종료 시에 학습 결과 분석과 함께 난이도를 조정하는 방식을 변경하여, 각 소단원 학습 과정 중에 다음 학습의 사전검사를 매시 실시하여 난이도를 설정하고 이를 다음 학습 콘텐츠에 즉시 반영하는 방법을 제시하였다. 이를 위하여 학습 콘텐츠는 각 난이도에 따라 별도로 준비되어야 하며, 교수자는 같은 내용의 학습 콘텐츠를라 할지라도 난이도에 따라 각기 어렵고, 쉬운 방식으로 학습 내용을 설명하는 콘텐츠를 제작하여야 했다. 그러나 학습자는 전체 대단원의 학습에서 분류된 소단원의 학습을 진행하는 중에도 다음 학습 내용을 이해도에 따라 학습 콘텐츠를 다르게 제공받을 수 있었다. 본 학습

의 적용을 위해 학습 모집단 20명을 대상으로 영어 학습 중 명사절을 설명하는 학습 콘텐츠를 소단원의 내용에 따라 분류 및 제작하여 학습을 시행하였다. 이때 소단원별 학습에서 학습자의 난이도가 조정됨을 보였으며, 이는 학습자의 학습 수준에 따라 학습이 제공되고 있음을 제시하였다. 또한 학습 전 후의 평균 학습 결과를 비교함으로써 본 방법의 효율성을 제시하였다.

향후 연구과제로서 본 제안방법은 학습자가 다음 학습을 위해 사전검사를 시행하게 되는데 이때 소요되는 시간이 많아 실제 학습 콘텐츠를 제공받기까지 일반 e-Learning 시스템보다 많은 시간을 필요로 한다. 따라서 학습자의 학습 난이도를 빠른 시간 내에 산출할 수 있는 방법이 요구된다.

참고 문헌

- [1] 김명희, 이현태, 오용선, "학습자 특성을 고려한 적응적 학습 관리 시스템의 설계 및 구현", 한국콘텐츠학회논문지, 제4권, 제1호, 2004.
- [2] 전영주, 강용구, 송해성, "WBI 수준별 수업을 통한 영어 의사소통능력 향상 방안", 영어어문교육, 제12권, 제4호, 2006.
- [3] 현분석, "초등학생의 영어능력 수준별 효과적인 학습전략", 초등영어교육, 제13권, 제1호, 2007.
- [4] 방기혁, "초등 교과교육을 위한 수준별 교수-학습 과정의 설계 및 운영 방안", 한국실과교육학회지, 제19권, 제4호, 2006.
- [5] 김정렬, 조혜정, "e-Learning 기반 초등 영어 수준별", 영어교육연구, 제19권, 제3호, 2007.
- [6] 이재근, "초등학교 영어교육에서 수준별 교육과정의 기본적인 개념 이해", 영어교과교육, 제6권, 제1호, 2007.

- [7] 정미경, 김경현, "사이버학습체제에서의 자기조절 학습모형 개발", 영재와 영재교육, 제4권, 제1호, 2005.
- [8] Michael William Rennie, Vojislav B. Mišić, Towards a Service-Based Architecture Description Language, University of Manitoba technical report TR 04/08, 2004.
- [9] Y. Zhenhua and C. Yuanli, Object-Oriented Petri nets Based Architecture Description Language for Multi-agent Systems, International Journal of Computer Science and Network Security, Vol.6, No.1B, 2006.
- [10] A. Robert, D. Rémi, and G. David, "Specifying and Analyzing Dynamic Software Architectures," Proceedings of 1998 Conference on Fundamental Approaches to Software Engineering, 1998.
- [11] 이명희, 손건태, 정희영, "일본어과 수준별 수업을 위한 컴퓨터 기반 문제은행 구축 및 테스트 시스템 개발", 일어일문학, 제24권, 2004.

**저자 소개**

**정 화 영(Hwa-Young Jeong)**

정회원



- 1994년 2월 : 경희대학교 전자계산공학과(공학석사)
- 2004년 8월 : 경희대학교 전자계산공학과(공학박사)
- 2000년 3월 ~ 2005년 2월 : 예원예술대학교 만화게임영상학

부 조교수

- 2005년 3월 ~ 현재 : 경희대학교 교양학부 조교수
- <관심분야> : 소프트웨어공학, CBD, 교육공학

**홍 봉 화(Bong-Hwa Hong)**

정회원



- 1987년 2월 : 경희대학교 전자공학과(공학사)
- 1992년 8월 : 경희대학교 전자공학과(공학석사)
- 2001년 8월 : 경희대학교 전자공학과(공학박사)

- 1997년 9월 ~ 2004년 2월 : 세명대학교 컴퓨터수리정보학과 교수

- 2004년 3월 ~ 현재 : 경희사이버대학교 정보통신학과 교수

<관심분야> : 방송정보, 디지털컨텐츠, 교육공학