

문제 유형을 고려한 학습자 중심의 문제은행 시스템 설계

오원욱, 김용수
경원대학교 전자계산학과
e-mail : wonwook@ku.kyungwon.ac.kr

A Design of Learner-oriented Item Bank System Considering Type of Content

Won-Wook Oh, Yong-Soo Kim
Dept of Computer Science, Kyung-Won University

요 약

본 논문에서는 웹기반 문제은행의 발전된 모형으로 문제 유형에 따른 학습자의 맞춤형 문제를 제공하는 시스템을 설계하였다. 본 시스템의 최종 목표는 능동적, 자기 주도적 학습을 실현할 수 있는 맞춤형 이러닝을 통해 학습자의 학업 성취도 향상에 있다. 교수자 및 개발자 중심의 이러닝 서비스가 아닌 학습자의 학습 결과에 능동적으로 반영하고 유연한 서비스를 제공하기 위해 문제 유형에 따른 적합한 다음 단계의 문제를 제공함으로써 이러닝의 효율을 극대화 한다. 기존의 고정 출제 또는 무작위 출제 방식에서 탈피하여 개별 학습자의 오답과 문제 유형을 고려하여 문제의 키워드, 선택된 보기의 키워드, 유동적 난이도에 따른 문제 추출 방식을 설계하였다. 이 모델은 보조 학습의 수단으로 문제은행 시스템을 학습자 중심에 근접하여, 융통적이고 효과적으로 개인화된 서비스를 제공한다.

키워드 : 문제 유형, 웹 기반 문제은행, 학습자 중심, 이러닝

1. 서론

정보화 시대 이전의 학습은 학습자들의 특성을 고려하지 않은 획일적인 전통적 교육 방식이라면, 1990년대 후반부터 인터넷을 이용한 이러닝(e-learning)이 도입되면서 웹을 이용한 교육의 접근성이 발달되었고, 이제 웹 어플리케이션을 이용한 다양한 학습자 중심의 모델들이 개발되어지고 있다. 학습 과정이 학습자에게 수동적, 획일적으로 받아들여지기 보다는 능동적, 자기 주도적 학습에 맞도록 교육 환경이 진화하고 있다는 것을 말한다.

자기 주도적 학습이란 자신의 관심과 일치하는 학습내용, 자신의 능력에 맞는 학습 계획, 자신이 요구하는 학습 목표를 스스로 설정하여, 스스로 진도를 조절하며, 스스로 평가하는 것을 말한다. 자기 주도적 학습을 실현하기 위해서 우선 학습목표에 대한 방향성을 잃지 않는[1] 학습자의 노력과 그에 맞는 학습자의 특성을 고려한 시스템이 필요하고, 학습자 스스로가 능동적으로 구성할 수 있도록 평가는 과제 수행 과정에서 연속적으로 이루어져야 한다.[2]

유비쿼터스 컴퓨팅 환경에서는 시스템이 자율적으로 변화하는 상황에 대응하여 사용자에게 개인화된 서비스를

제공해야 할 것이다[3].

본 논문에서는 이러닝이 가장 활성화된 중, 고등학교에 초점을 두어 시간과 공간적 제약을 받지 않는 이러닝의 장점과 학습자의 능동적 참여를 유도하는 맞춤형 문제은행 시스템의 모델을 도출하고, 문제 유형을 고려한 웹기반 문제은행 시스템의 설계에 대해 기술한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 자동 문제 출제 시스템의 관련연구에 대해 살펴본다. 3장에서는 본 시스템의 전체 구성과 기본 설계에 대해서 소개하고, 4장에서는 구현된 가상 시나리오를 기술하였다. 끝으로 5장에서 결론 및 향후 보완할 연구 사항에 대해 논의 하였다.

2. 관련연구

중, 고등학교에서 정규 수업 이외에 보조학습의 수단으로 이러닝을 이용하고 있다. 이러닝의 활용은 단순한 VOD 방식의 전달형 수업이 주를 이루고 있지만, 자기 주도적 학습의 필요성이 확대되면서 학습자의 능동적 참여를 유도하는 교육 지원 시스템의 모델이 있다. 그 예로 학습자가 스스로 학습 계획을 수립하고 수행 및 평가할 수 있도록

록 안내하는 모듈과[1], 학습 정보 분석을 통하여 학습자 스스로가 취약한 학습 단원이나 학습 내용을 파악하고 학습 계획과 목표를 설정하는 시스템이 있다[4].

본 논문의 관련연구에서는 문제은행과 관련된 주제로 개인화된 맞춤형 자동 출제 문제은행 시스템과 사용자 선호도를 이용한 문제은행 시스템에 대해 살펴본다.

2.1 자동 문제 출제 시스템

일반적으로 문제의 출제 방식은 고정 출제 또는 무작위 출제 방식이었다. 교수자의 난이도 조절의도가 반영되지 못하며 학습자의 피드백 학습이 제대로 이루어지지 않았다. 이에 반해 자동 문제 출제 시스템은 출제되는 문제에 난이도를 부여하므로 출제자의 출제 의도를 반영한다. 또한 학습자의 평가 결과를 문제의 난이도에 재부여하므로 신뢰성을 높이고, 나아가 학습자의 피드백 학습을 제공하므로 재평가를 가능하게 한다. 최초 교수자의 주관적인 가중치가 부여 된 것을 학습자의 평가가 거듭되면서 객관적인 난이도로 정확도가 상승하게 된다.

출제방식도 다음과 같이 두 가지로 지원된다. 첫째는 교수자의 개별 선택방식으로 문제 리스트에서 문제를 보고 선택할 수 있고, 둘째는 자동 난이도 출제 방식으로 예상평균 점수와 출제 문제수를 입력하여 자동으로 출제할 수 있다. 또한 자동 채점 기능을 지원하므로 학습자는 평가 후 틀린 문제와 그 단원에 대한 피드백 학습이 가능하고, 학습자의 난이도 조절을 통하여 재평가를 치르고 학업성취도에 따라 문제의 유형을 달리 할 수 있다[5].

그러나 이와 같은 자동 출제 시스템의 경우 출제자 중심의 모델로 동일한 문제를 학습자에게 제공하는 한계가 있다.

2.2 사용자 선호도를 반영한 문제 출제 관리 시스템

문항의 난이도를 이용하여 적절한 안배를 통해 문제를 출제하는 시스템이다. 난이도는 문항의 쉽고 어려운 정도를 나타내는 지수로 문제가 쉽고 어려운 정도를 나타내는 지수로 사전, 결과, 그리고 사후 난이도로 나뉜다. 사전 난이도(D_{pre})는 문항의 어려운 정도를 초기에 정하는 지수로 문항을 신규로 입력할 때 교수자에 의해 정해진다. 결과 난이도(D_{rst})는 문항을 수험자 집단에 적용한 결과를 나타내는 지수이다. 사후 난이도(D_{post})는 검사 대상자에 의해 문항의 난이도가 급격히 변경되는 것을 방지하기 위해 사전 난이도와결과 난이도를 모두 반영하여 구한다. 난이도 결정 및 조정 과정에서 기존의 연구에서 제시하는 5단계로는 문항의 고유한 난이도를 가지지 못하기 때문에 10단계로 확장하고, 신뢰성을 확대하기 위해 지수평균 공식을 적용하였다.

결과 난이도 공식은 다음과 같다.

$$D_{rst} = (R/N + (P/N*A))*10$$

R : 문항에 대한 정답을 한 수험자 수

P : 문항에 대한 부분정답을 한 수험자 수

N : 총 수험자 수

A : 부분점수 반영률

결과 난이도는 정답률(R/N)과 부분 정답률($P/N*A$)을 더하여 얻어지는 값에 10을 곱한 값으로 난이도 지수를 정수로 표기하였다.

사후 난이도 공식은 다음과 같다.

$$D_{post} = \alpha D_{rst} + (1-\alpha)D_{pre} \quad (0 \leq \alpha \leq 1)$$

D_{pre} : 사전 난이도

D_{rst} : 결과 난이도

D_{post} : 사후 난이도

α : 매개변수

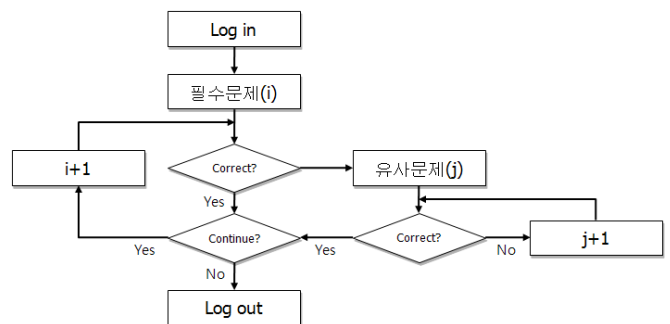
사후 난이도는 결과 난이도와 사전 난이도를 모두 고려하기 위해서 SJF(Shortest-Job-First) 스케줄링에서 사용되는 지수평균 공식(exponential average)을 사용하였다. 매개변수는 결과 난이도와 사전 난이도의 상대적인 가중 값을 제어한다.

만약 $\alpha = 0$ 이면 $D_{pre} = D_{post}$ 으로 사전 난이도와 사후 난이도가 같고, $\alpha = 1$ 이면 $D_{rst} = D_{post}$ 로 사후 난이도는 결과 난이도만 반영한 것이 된다. 위 매개변수 값은 출제자가 정하도록 한다[6][7][8].

사용자 선호도를 반영한 문제 출제 관리 시스템의 경우 난이도를 10단계의 난이도로 신뢰성을 높였다.

3. 유형별 문제은행 시스템

본 논문에서 제안하는 문제은행 시스템은 여러 가지 문제의 유형에 적합한 자동 출제 시스템으로 학습자에게 적합한 문제를 출제할 수 있다. 본 논문을 진행함에 앞서 다음과 같은 전제 조건을 부여한다.



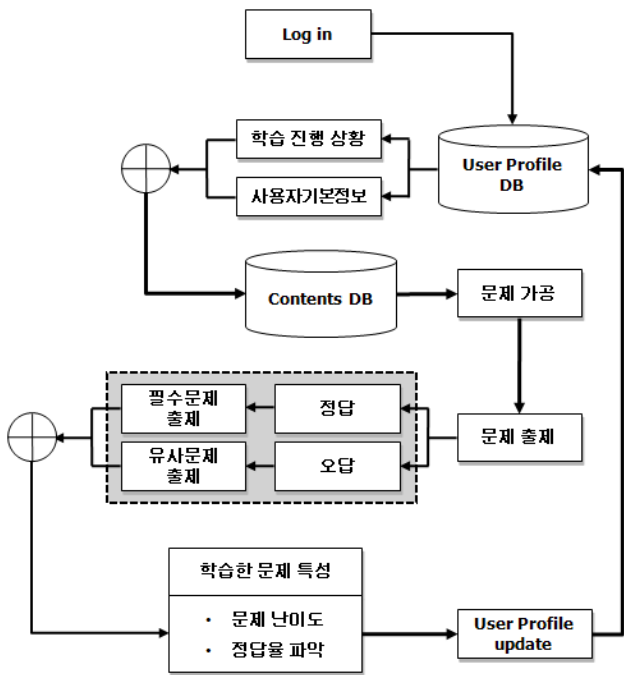
(그림 1) 필수-유사문제 Flowchart

1. 중, 고등학생을 위한 문제은행 시스템을 다룬다.

2. (그림 1)과 같이 필수문제(i) 풀이 후 정답일 경우 다음 단계의 필수문제($i+1$)로 진행되고, 오답을 선택할 경우 유사문제(j)가 출제 된다. 유사문제 풀이 단계에서도 오답이면 다음 단계의 유사문제($j+1$)로 진행된다.

위 전제 조건에 따라 두 가지 문제 유형별 문제은행 시스템에 대해 설명한다. 학습자는 필수 문제의 정답을 선택하면 다음 단계의 문제가 출제되고, 오답을 선택하면 문제의 유형에 따라 다음의 두 가지 방법을 통하여 유사 문제가 출제 된다. 첫 번째는 키워드를 이용한 추출로 문제의 키워드와 선택된 보기의 키워드를 검색하는 방법이고, 두 번째는 유동적 난이도에 의한 추출 방법이다.

(그림 2)는 문제은행 시스템 구성도를 나타낸 것으로 두 가지 문제 유형별 시스템에 모두 적용이 되는 모형이며, 정답과 오답의 구분을 통해서 필수문제와 유사문제로 이동하는 과정을 도식화하였다.



(그림 2) 문제은행 시스템 구성도

3.1 키워드에 따른 문제 추출

키워드에 따른 문제 추출 방법은 두 가지로 분류 할 수 있다. 문제 키워드(Subject Keyword)를 이용하거나 선택된 보기 키워드(Instance Keyword)를 이용하게 된다.

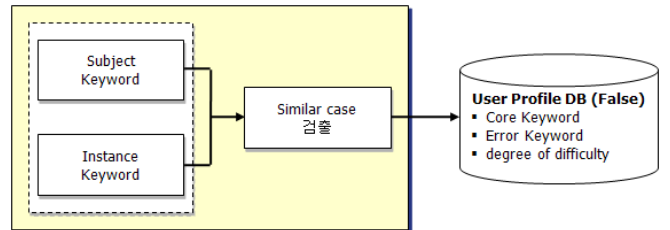
(그림 3)은 문제 키워드와 보기 키워드를 이용한 처리 과정을 나타낸 것이다.

문제 키워드를 이용한 방법은 오답으로 인한 유사문제 풀이의 경우 필수문제에 등록된 키워드를 통해 유사문제를 출제 하는 방식이다. 키워드를 검색하는 방법은 출제자가 사전에 문제의 정확한 정보를 입력해야 하지만, 본

시스템은 자동으로 키워드를 인식하여 유사 문제를 찾아내어 제공한다. 이 방법은 문제에 키워드가 존재할 때 적용된다.

선택된 보기 키워드를 이용한 방법은 문제의 키워드와 동일한 방법이나 문제가 아닌 선택된 보기에 키워드가 존재할 때 적용되어진다.

위 두 가지 문제 키워드와 선택된 보기 키워드 중에서 핵심 키워드(Core Keyword)를 검색하여 가장 근접한 유사 문제(Similar Case)를 출제하고 User Profile DB에 저장하게 된다.

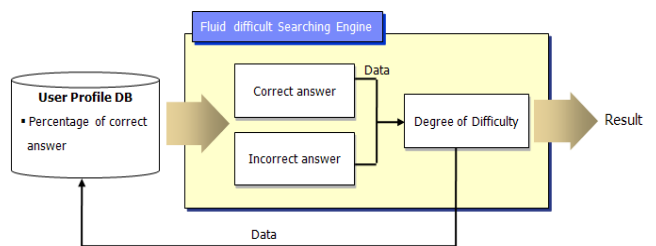


(그림 3) 문제 및 보기 키워드 처리 과정

3.2 유동적 난이도를 이용한 문제 추출

유동적 난이도를 이용한 문제 추출 방법은 정답률에 따라 유사문제를 출제 하는 방식이다. 유동적 난이도는 모든 학습자의 정답률이 반영되어 다음의 문제를 추출할 때 정답을 선택한 경우 한 단계 상향되고, 오답을 선택한 경우 한 단계 하향되어 학습자의 수준을 고려하여 문제가 추출된다. 이 방법은 문제와 보기의 키워드를 이용할 수 없는 수학 문제에 응용한다.

(그림4)는 난이도 처리 과정을 나타낸 것으로 정답(Correct answer)과 오답(Incorrect answer)의 빈도를 이용하여 문제의 난이도(Degree of Difficulty)를 User Profile DB에 저장하게 된다.



(그림 4) 난이도 처리 과정

위의 두 가지 문제 추출 방식을 복합적으로 이용하여 정답의 경우 단계별 난이도가 조정이 되면서 필수 문제 위주의 학습이 가능하고, 오답의 경우 학습자가 필요로 하는 유사문제를 출제하여 기존의 출제자 중심의 문제은행 시스템을 학습자에 맞춤형 모델로 서비스가 가능하다.

4. 과목별 가상 시나리오

본 논문에서 설계한 자동 문제 추출 방법을 과목별로 적용하겠다. 기본적으로 단원별 지정이 되어 유사문제의 분류가 선행되어야 한다. 수학과목의 경우 대다수 문제의 보기는 숫자 및 수학기호로 구성되기 때문에 문제 키워드와 보기 키워드를 이용하여 자동 출제 되는 방법을 이용하기 어렵고, 난이도를 통하여 그 단원의 유사 문제를 출제 하는 방식의 적용이 유용하다. 영어와 국어 과목처럼 문제 유형이 다양한 경우 역시 난이도를 통한 방식이 필요하다. 난이도를 통한 방법의 경우 소단원의 전체 문제들을 단계별로 제공하게 되고, 학습 과정은 순차적으로 진행이 가능하다. 사회와 과학 문제의 경우 문제와 보기에 많은 키워드가 존재하므로 키워드를 이용하여 유사문제를 출제하는데 유용하다. 또한 키워드와 난이도를 종합적으로 고려해 최적화된 문제를 출제 하는데 효과적이다. 중, 고등학교 사회 및 과학 과목과 문제 유형이 유사한 자격증, 국가고시의 문제들에도 자동문제 출제 시스템 적용이 가능하다.

5. 결론

본 논문은 중, 고등학생의 문제은행을 이용한 이러닝 활용에 초점을 두었다. 기존의 문제 은행의 출제 방식은 고정 출제, 무작위 출제, 그리고 지정된 유사문제 출제 중심이었다. 이는 출제자 중심의 방식으로 자기 주도적 학습 및 능동적 참여를 기대하는 개인화된 맞춤형 문제은행 시스템을 구현하기 위해 문제의 키워드, 선택된 보기의 키워드, 유동적 난이도 출제 방식으로 한 단계 더 고려하여 학습자에 출제 하는 모델이다. 키워드 추출 방식은 문제와 선택된 보기의 핵심 키워드를 이용한 문제의 유형에 적용하는 모델이고, 유동적 난이도 방식은 학습자의 정답률을 유동적으로 인식하여 정답 선택의 경우 한 단계 상향된 문제를 출제하고 오답 선택의 경우 한 단계 하향된 문제를 출제하는 모델이다. 과목별 특성에 맞는 문제의 유형들에 적용하였을 때 이 두 가지 모델을 동시에 적용함으로써 학습자에 적합한 문제를 제공하게 된다.

새롭게 설계한 이 두 가지 모델을 시행하기 위해서 학습자의 능동적 참여가 필요하다. 출제자의 경우 문제에 대한 정확한 의도를 규정하는 별도의 작업 없이 단원별 분류를 통해 자동으로 문제를 출제 되므로 편리하다.

학습의 효율을 증명하기 위해서 학습 모집단에 대한 테스트를 거쳐야 한다. 또한 중, 고등학교의 학습 과정 뿐 아니라 모든 이러닝 분야에 적용할 수 있도록 추가적인 연구가 필요하다.

참고문헌

- [1] 장덕성, 조현욱, “자기 학습 계획을 갖는 웹기반 학습 시스템의 설계 및 구현”, 정보처리학회논문지A 제11-A권 제4호, pp.297, 8, 2004
- [2] 이종득, 정택원, “인지적 교수 학습을 위한 멀티미디어 보조학습시스템의 설계 및 구현”, 한국컴퓨터산업교육학회 논문지, Vol. 3, No. 6, pp.719, 6, 2002
- [3] 이창열, 조규찬, 김현숙, 조위덕, “자가 성장하는 상황기반 사용자 모델을 이용한 개인화 커뮤니티 서비스 자동 제공 방법”, 정보과학회논문지 컴퓨팅의 실제 및 레터 제 14권 제7호, pp.739, 10, 2008
- [4] 정화영, “웹 서비스 기반 자기조절학습을 위한 이러닝 시스템의 구현”, 한국컴퓨터교육학회 제11권 제2호, pp.83, 3, 2008
- [5] 김경아, 최은만, “웹기반교육에서의 자동 문제 출제 시스템”, 정보처리학회논문지A 제9-A권 제3호, pp.304, 9, 2002
- [6] 김에스더, “사용자 선호도를 반영한 문제 출제 관리 시스템”, 석사학위 논문 성신여대 교육대학원 2007
- [7] 성태제, "문항제작 및 분석의 이론과 실제", 학지사, 1998
- [8] 채선희, "문제은행의 본질적 가치와 효율적 이용 방안", 교육평가학회 1996
- [9] Peter K. Wiesner. "E learning in 2001, : Proceedings of the 25th Annual International Computer Software and Application Conference(COMPSAC), 2001
- [10] J. Y. Nie and M. Brisebois, “An Inferential Approach to Information Retrieval and its Implementation using a Manual Thesaurus”, Artificial Intelligence Review, Vol.10, No.5, 1996
- [11] B. Y. Ricardo and R. N. Berthier, *Modern Information Retrieval*, Addison-Wesley, 2000
- [12] C. Buckley, G.Salton and J.Allan, “The Effect of Adding Relevance Information in a relevance Feedback Environment”, In Proceedings of the 17th Annual International ACM/SIGIR Conference, Dublin, Ireland, 1994
- [13] M. Mitra, A. Singhal and C. Buckley, “Improving Automatic Query Expansion”, In Proceedings of the 21th Annual International ACM/SIGIR Conference, Melbourne, Australia, 1998