

유비쿼터스 학습(u-Learning)을 위한 미디어이터 기반의 분산정보 활용방법

주길홍

연세대학교 컴퓨터과학과

요약

컴퓨터와 통신 기술이 발전함에 따라 네트워크를 통한 일반 사용자들의 컴퓨터 활용 빈도와 요구하는 데이터의 양이 급격히 증가되었다. 이에 따라 최근의 교육 시스템들은 정보의 활용성을 향상시키기 위하여 이질적인 시스템들을 의미상으로 연결하고 있다. 따라서 최근의 웹 기반 교수-학습은 학습자 스스로 학습 내용, 학습 시간 및 학습 순서를 선택하고 조직하는 유비쿼터스 학습방향으로 나아가고 있다. 즉, 학습자 개개인의 특성(선수 지식, 학습 양식, 흥미, 관심)에 맞는 적응적인 교수-학습 환경을 제공하는 방향으로 변화되고 있다. 본 논문은 유비쿼터스 학습 환경에서 다양한 분산정보의 통합을 위하여 사용자들이 요구하는 학습내용을 각 지역서버의 자치성을 유지하면서 효과적으로 학습하기 위한 미디어이터내의 처리방법에 대해 제안한다. 또한 과거와 최근의 학습내용의 활용형태가 다양하게 변할 수 있으므로 시간에 따른 감쇄율을 활용빈도에 적용하여 최근의 활용빈도의 변화에 민감하게 반응하고 활용형태의 변화에 따라 적응적으로 학습내용을 사용할 수 있는 방법을 제안한다.

A Practical Method of a Distributed Information Resources Based on a Mediator for the u-Learning Environment

KilHong Joo

Yonsei University, Dept. of Computer science

ABSTRACT

With the rapid advance of computer and communication technology, the amount of data transferred is also increasing more than ever. The recent trend of education systems is connecting related information semantically in different systems in order to improve the utilization of computerized information. Therefore, Web-based teaching-learning is developing in the ubiquitous learning direction that learners select and organize the contents, time and order of learning by themselves. That is, it is evolving to provide teaching-learning environment adaptive to individual learners' characteristics (their level of knowledge, pattern of study, areas of interest). This paper proposes the efficient evaluation method of learning contents in a mediator for the integration of heterogeneous information resources. This means that the autonomy of a remote server can be preserved to the highest degree. In addition, this paper proposes the adaptive optimization of learning contents such that available storage in a mediator can be highly utilized at any time. In order to differentiate the recent usage of a learning content from the past, the accumulated usage frequency of a learning content decays as time goes by.

keyword : u-Learning, mediator, distributed query processing, data materialization

1. 서론

유비쿼터스 컴퓨팅(Ubiquitous Computing)은 수많은 지능형 컴퓨터들이 우리의 일상생활 속으로 스며들어 밖으로 드러나 보이지 않는 상태에서 서로 유기적으로 연결되어 협조함으로써 언제 어디서나 필요한 정보나 서비스를 맞춤형으로 즉시 제공하여 우리의 삶의 질을 향상시키는 새로운 컴퓨터 환경을 제공한다 [16,17]. 즉, 유비쿼터스 컴퓨팅은 현재의 컴퓨터에 어떠한 기능을 추가하는 것이 아니라 세상에 존재하는 모든 사물에 컴퓨터 칩이 심어져 있는 미래(Computer Everywhere)를 지칭한다. 유비쿼터스 환경에서 무엇보다 중요한 것은 컴퓨터와 사람간의 친화도를 높이는데 있다. 다양한 미디어 기술들은 사람에게 컴퓨터를 자연스럽게 사용할 수 있는 기능들을 제공한다.

산업시대의 학생들은 교실을 찾아다니며 공부했고 정보화시대에는 학생들이 인터넷을 통해 학습정보를 얻었다. 그러나 유비쿼터스 교육시대는 학습정보가 학생들을 스스로 찾아다닐 것이다. 학생들이 언제 어디서나 어떤 내용에 상관없이, 어떤 단말기로도 학습할 수 있는 교육환경을 조성해줌으로써 보다 창의적이고 학습자가 중심이 된 교육과정을 실현하는 것이 유비쿼터스 학습의 목표이다. 그러나 유비쿼터스 컴퓨팅 환경에서는 영상 등의 고품질 커뮤니케이션 및 콘텐츠 전송, 인덱스화, 축적이 가능해진다. 또한 감성이나 요령에 가까운 분야의 지식, 즉 지혜에 해당하는 영역의 교화, 공유의 장이 창조되는 것이다. 따라서 누구나 어디서나 네트워크에 참여할 수 있고, 보다 간단하게 커뮤니케이션을 할 수 있는 환경을 제공하는 것이다. 이러한 유비쿼터스의 기술을 이용한 학습을 바로 '언제 어디서나 원하는 학습을 할 수 있는 '유비쿼터스 학습(u-Learning)'이라고 한다 [12].

유비쿼터스 학습환경은 학생들이 각자의 개별화된 욕구에 따라 학습할 수 있으며, 학생과 교사들간의 상호작용도 자연스럽게 편안하게 이뤄지게 될 것이다. 유비쿼터스 교육에 대한 연구와 준비는 이미 유비쿼터스 컴퓨팅의 등장과 함께 논의되었다. 유비쿼터스 교육체제로 가는 지름길은 학생들이 간편하게 들고 다닐 수 있는 유비쿼터스 교육용 컴퓨터의 개발과 교육용 시스템온칩(SoC), 칠판·책상·학습판·교구의 지능화등의 하드웨어적 기술과 교육용 무선 네트워크의 표준화, 네트워크화를 위한 실시간 운영체제 및 사용자가 필요한 정보를 신속하게 제공하는 소프트웨어적인 기술이 필요하다. 이를

위해 사용자의 정보 활용패턴을 분석하여 정보의 우선순위를 결정하는 방법이 절실하게 요구된다. 또한 유비쿼터스 환경을 지원하기 위하여 이기종 컴퓨터들 사이의 정보 통합 및 상호 운용성(Interoperability)을 지원하기 위한 CORBA(Common Object Request Broker Architecture)의 ORB와 같은 소프트웨어 구조나 미디어이터(Mediator) 시스템의 연구가 활발하게 진행되고 있다 [5].

미디어이터란 서로 이질적인 하드웨어 환경 및 이질적인 데이터베이스와 소프트웨어 등에 대해서 사용자가 이질적인 환경을 느끼지 못하게 은폐하면서 하나의 시스템에 접근하는 것과 같은 효과를 제공하는 미들웨어(middleware) [14]를 의미한다. 미디어이터 시스템은 의미적으로 통합된 전역스키마를 사용하지 않고 사용자가 필요로 하는 기능에 대한 정보단위로 연관된 지역스키마들에 대한 뷰를 정의한다. 본 논문에서는 미디어이터 시스템이 분산되어 있는 정보를 통합한 후에 사용자의 정보 활용패턴을 분석한 후 정보의 우선순위를 결정하여 신속하게 사용자에게 양질의 정보를 제공하는 방법에 대하여 기술한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구에 대해서 기술하고, 3장에서는 유비쿼터스 환경에서 미디어이터의 질의처리를 위한 사용자 활용패턴에 적응적으로 모델링하는 방법을 기술한다. 4장에서는 정보의 활용형태에 따라 정보의 우선순위를 결정하는 방법을 기술한다. 5장에서는 3장과 4장에서 제안된 내용에 대한 다양한 실험을 수행하고 이에 대한 결과를 분석한다. 마지막으로 6장에서는 최종적인 결론을 맺는다.

2. 관련연구

최근에 주목을 받기 시작한 디지털형 네트워크 학습인 e-Learning은 정보화시대에 맞는 학습의 방법이라고 할 수 있다. e-Learning은 전자적 도움을 통해서 이루어지는 웹기반의 디지털형 학습을 지칭하는 것이다 [1]. e-Learning은 다음과 같은 특징이 있다. 첫째로 '교육'보다는 '학습'을 강조한다. e-Learning은 교육에 대한 새로운 관점을 전제로 하고 있는데 그중 가장 두드러지는 특성은 전통적인 "교수"의 의미보다는 "학습"의 의미가 강조되고 있으며, 이에 따라 e-Learning을 통해서 전개되는 교육은 전통적인 교실 수업과는 다르게 학습자를 중심으로 이루어진다는 점이다. 따라서 학습자는 자

신의 학습 목표에 따라서 학습 과정을 이수해 나가게 된다 [5]. 예를 들어 진도를 스스로 조절하여 알맞게 진행해 나갈 수도 있고, 원할 때 원하는 만큼 학습이 가능하다. 둘째로, 시간적, 공간적 제약에서 자유로워졌다. 학습자가 원할 때 기본적인 제반여건(네트워크에 연결된 컴퓨터 등)만 갖추어지면 자유로운 시간, 공간에서 학습이 가능하다.

e-Learning이 장점만 있는 것이 아니라 다음과 같은 문제점도 있다. 첫째는 오감을 통한 학습의 부족이다. e-Learning은 공간과 시간이 일치함으로 해서 느낄 수 있는 오감의 작용을 통한 학습(오프라인 수업시간에 느낄 수 있는 학습의욕의 고취나, 현장에서 수업을 통한 서로의 의사소통)을 제공할 수 없다. 둘째는 학습자의 소속감 부족문제이다. 학습자가 커뮤니티에 대해 느끼는 소속감과 존재감이 당연히 현재로서는 오프라인에 비해 월등히 떨어질 수 밖에 없다. e-Learning을 통해서 학습기회를 접하기 쉬워졌다는 것은 그만큼 이탈될 확률도 높다는 것이다.

이와같은 문제점은 유비쿼터스 학습(u-Learning)을 통하여 해결할 수 있다 [12]. 그러나 유비쿼터스 학습을 위해서는 이질적인 환경을 가지는 여러 개의 정보 시스템들을 통합하는 기술이 절실하게 필요하다. 이때 중점적으로 고려해야 할 사항은 사용자에게 이질적인 환경을 은폐하면서 원격지에 존재하는 시스템들의 자치성을 최대한 유지하는 것이다.

TSIMMIS [6]는 미디어이터 시스템을 ORB환경에서 구축하여 미디어이터를 통하여 이질적인 분산 환경의 데이터베이스에도 접근이 가능하다. 이를 위해 미디어이터에서는 전체 스키마와 뷰를 MSL (Mediator Specification Language)로 정의를 하고, 래퍼에서는 원격지서버의 관련정보를 WSL(Wrapper Specification Language)로 정의하기 때문에 MSL과 WSL을 매핑시키는 작업이 필요하고 자체적으로 정의한 특정한 툴킷과 언어를 통하여 데이터베이스에 접근이 가능하다. 또한 HERMES [15]는 이미지와 텍스트, 그리고 데이터베이스등의 다양한 기능들을 통합하는 미디어이터 시스템이다. 이 시스템은 단일 기종의 시스템에서만 사용이 가능하고, 데이터베이스, 데이터 구조, 그리고 소프트웨어 패키지 등을 통합하는 방법을 제시하였다. HERMES는 자체적인 논리적 언어(logic-based declarative language)를 통해 통합하고자 하는 서버와 데이터 교환을 수행하여 각 서버와의 통합을 가능하게 한다. 따라서 미디어이터 관리자가 미디어이터의 언어에 기반을 두어 연결된 서버와 데이터베이스에 필요한 입력형식과 출력형식을 일일이 열거해야한다.

HERMES는 미디어이터의 스키마와 관련된 원격지서버의 데이터와의 연동을 지원하기 위해 "yellow page" 기능을 제공한다. TSIMMIS 시스템과 HERMES 시스템은 미디어이터의 전역 질의를 수정방법으로 구현하고 있다.

유비쿼터스 학습을 위한 미디어이터에서는 자주 활용되는 학습내용을 구체화방법으로 구현하면 높은 효율성을 가질 수 있다. 그러나 수많은 학습내용들 중에서 구체화방법으로 구현할 학습내용을 결정하는 문제는 해당 시점에서의 활용형태에 따라 구체화관리비용과 수정관리비용을 모두 고려하여 신속하게 관리방법을 결정해야 하는 어려움이 있다. 이러한 어려운 문제 때문에 기존의 미디어이터들 [2, 3, 6, 15]은 수정방법만 사용하여 네트워크 트래픽과 처리속도가 증가하는 단점이 있다. 그러나 미디어이터 시스템에서도 학습자의 학습내용에 대한 접근형태나 갱신형태에 대해 과거의 시점과 최근의 시점의 변화를 모델링하여 관리방법을 결정한다면 미디어이터에서의 성능을 향상시킬 수 있다 [10].

이때 전체 학습내용의 수가 증가할수록 가능한 모든 구현방법의 조합을 고려하여 구체화집합을 찾는 문제가 NP-complete임으로 [4, 7, 9] 학습내용의 수가 증가될수록 최적화된 구현방법을 생성하는데 많은 시간이 필요하다. 따라서 본 논문에서는 미디어이터에서 유동적으로 변화되는 학습내용의 활용형태 및 갱신형태에 대해 신속하고 자체 적응적인 최적알고리즘을 제안한다.

3. 사용자 활용패턴 분석

3.1 유비쿼터스 학습을 위한 사용자 활용패턴

유비쿼터스 컴퓨팅 환경은 '커뮤니티 파워의 증대'와 '형태지의 교환, 공유'라는 사회 변화를 촉진하고 있다. 형태지 교환, 공유는 유통 콘텐츠의 증대와 사용자와 네트워크 간의 관계성 다양화의 조합이 심화됨으로써 종래의 형식화된 지식기반 공유의 세계에서 형식화하기 어려운 지혜나 요령, 노하우 공유의 세계로 비약하는 계기가 될 것이다. 종래의 인터넷 세계는 킬러 콘텐츠로 대표되는 전자 메일과 같은 텍스트 기반의 정보교환이 주류였다. 따라서 자신의 생각을 문자로 전환시켜서 전달하거나 축적해 두어야 했다. 또한 텍스트로 표현되는 전자 메일에서는 감성적인 부분이나 말하고 싶어도 직접 말할 수 없는 부분을 잘 표현하고 전달할 수 없었다.

유비쿼터스의 기술을 이용한 u-Learning의 핵심

은 일상생활을 하면서 '빈 시간'을 활용, 장소에 구애 받지 않고 학습할 수 있게 함으로써 학습자들의 학습 효율성 및 효과성을 극대화할 수 있다는 데 있다. 또한 학생들이 언제 어디서나 어떤 내용에 상관없이, 어떤 단말기로도 학습할 수 있는 교육환경을 조성해 줄 수 있다. 예를 들어 우리가 도로를 지나가다 어떤 문화재를 보게 된다면 서로 연결된 유비쿼터스의 환경은 그와 관련된 내용을 학습하게 도와준다. 학습자가 가지고 있는 단말기의 칩과 문화재 속의 칩이 서로 연결되어 학습자의 단말기에 관련 내용이 나타나게 되는 것이다. 즉, 유비쿼터스 교육환경은 획일적인 교육환경이 아니라 세상의 모든 곳이 교실이 될 수 있는 열려 있는 환경인 것이다.

본 논문에서는 학습자에게 제공되는 다양한 학습 정보를 질의(Query)라고 정의한다. 학습자가 학습하고자하는 내용이 질의형태로 표현되어 시스템에서 인식하게 되며, 질의에 대한 학습자의 활용패턴이 정의되어진다. 생성된 질의의 활용패턴은 접근회수(evaluation frequency)와 갱신회수(update frequency)로 모델링 한다. 접근회수는 해당 학습내용을 사용하는 회수이고, 갱신회수는 학습자가 사용한 학습내용이 갱신된 회수이다. 즉, 학습한 내용이 변경된 경우 갱신회수의 갱신이 발생한다. 유비쿼터스 학습을 위해 이러한 과정을 미디어이터에서 관리하게 되며, 학습내용에 대한 사용자의 활용패턴을 정확하게 파악함으로써 정확하고 의미있는 정보를 학습자에게 신속하게 제공할 수 있다.

미디어이터에서는 질의를 처리하기 위해 수정방법과 구체화방법을 사용한다[4]. 수정방법은 각 원격서버에서 모든 해당 질의를 처리하며, 구체화방법은 자주 사용되어지는 질의의 결과를 미디어이터에 저장하여 반복된 질의가 요구될 때 신속하게 정보를 제공한다. 따라서 서로 다른 질의에 같은 결과가 구체화로 구현되어 있다면 질의의 결과를 공유할 수 있으므로 매우 효율적이다. 또한 자주 사용되는 질의를 구체화로 구현할 경우 수정방법으로 구현할 경우보다 네트워크의 트래픽을 감소시키고 학습자에게 신속하게 결과를 제공할 수 있다[11]. 본 논문에서는 수정방법으로 구현되는 질의를 수정질의(modified query)라고 정의하고, 구체화방법으로 구현되는 질의를 구체화질의(materialized sub-query)라고 정의한다.

3.2 감쇄율 기반의 사용자 활용 패턴

미디어이터에서 학습자에게 정확한 정보를 제공하기

위하여 각 질의의 최근 활용패턴을 감시하는 것은 매우 중요한 일이다. 따라서 본 논문에서는 과거에 일어난 사용자의 접근이나 갱신보다 최근에 일어난 사용자의 접근이나 갱신에 더 높은 비중을 두기 위해 감쇄율을 적용한다. 이것은 질의의 활용이나 갱신에 대해 반감기(half_life)[8]를 두어 이전회수와 최근회수에 차이를 두는 방법이다. 반감기란 질의의 활용이 영향을 50%만큼 주는 시점을 말한다. 따라서 반감기를 h 라고 할 때 감쇄율 d 는 식(1)과 같이 정의한다.

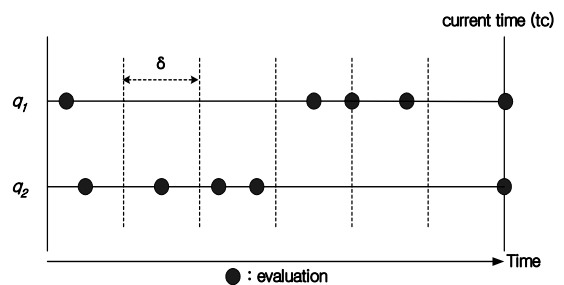
$$\text{감쇄율 } d = 2^{-\frac{1}{h}} \quad (1)$$

사용자가 정의한 시간간격을 \mathbb{B} 라고 할 때 감쇄율이 적용된 이전의 접근회수와 갱신회수의 값을 다음과 같이 표현한다.

$$e_{i.new} = e_{i.old} \cdot d^t + 1 \quad \left(t = \left\lfloor \frac{R}{\delta} \right\rfloor \right) \quad (2)$$

$$u_{i.new} = u_{i.old} \cdot d^t + 1 \quad \left(t = \left\lfloor \frac{R}{\delta} \right\rfloor \right) \quad (3)$$

이때 $e_{i.old}$ 는 질의 q_i 에 대한 가장 최근의 활용이 R 시간 이전에 발생했을 때에 계산된 접근회수이고, $e_{i.new}$ 는 $e_{i.old}$ 를 기반으로 가장 최근에 새롭게 계산된 질의 q_i 의 접근회수이다. 이와 마찬가지로 $u_{i.old}$ 는 질의 q_i 에 대한 가장 최근의 접근이 R 시간 이전에 발생했을 때에 계산된 갱신회수이고, $u_{i.new}$ 는 $u_{i.old}$ 를 기반으로 가장 최근에 새롭게 계산된 질의 q_i 의 갱신회수이다. R 이 \mathbb{B} 보다 크다면 커진 범위만큼의 감쇄율이 적용되고, 이와는 반대로 \mathbb{B} 보다 작다면 동일한 범위 내에서 수행이 된 것으로 간주한다. 예를 들어 그림 1에서 질의 q_1 과 q_2 의 접근회수는 감쇄율이 적용되지 않는다면 동일하게 5로 결정되어진다. 그러나 질의 q_1 은 질의 q_2 보다 최근에 사용되어졌기 때문에 똑같은 회수를 반영하는 것은 정확한 결과를 사용자에게 제공하지 못한다. 따라서 감쇄율을 적용하여 질의의 접근회수를 새롭게 설정한다.



(그림 1) 감쇄율이 적용된 사용자 패턴

그림 1에서 반감기가 1이라고 하면 감쇄율을 적용한 질의 q_1 의 접근회수는 $e_{q_1.tc} = 1 + 2 \cdot 2^{-1} + (2^{-2}) + (2^{-5}) = 2.28$ 이고, 동일하게 질의 q_2 의 접근회수는 $e_{q_2.tc} = 1 + 2 \cdot (2^{-3}) + (2^{-4}) + (2^{-5}) = 1.34$ 이다. 즉, 시간간격 δ 를 어떠한 값으로 정의하는가에 따라서 질의의 활용패턴의 모델링 단위를 조절할 수 있다. 따라서 시간간격 δ 를 작게 설정할수록 질의의 활용패턴은 더욱 정확하게 모델링 된다.

미디어이터에서 사용자의 질의가 발생하면 식(2)에 의해서 해당 질의별로 감쇄율을 적용하여 접근회수를 증가시킨다. 이때 수정질의는 미디어이터에 결과가 저장되어 있지 않기 때문에 사용자의 질의가 발생할 때마다 원격지서버로부터 질의의 결과를 전송 받아 사용자에게 제공한다. 이와는 반대로 구체화질의는 미디어이터에 질의 결과가 저장되어 있기 때문에 원격지서버에 질의의 실행을 요청하지 않고 신속하게 질의결과를 제공할 수 있다. 그러나 질의의 대상이 되는 기본정보가 갱신 된 경우에는 새롭게 갱신된 내용이 반영된 결과를 사용자에게 제공하여야 하기 때문에 미디어이터에 구체화되어 저장된 갱신 이전의 질의 결과를 무효화시키고, 식(3)에 의해서 갱신 후에 해당 질의의 기본정보에 대해 첫 번 사용자의 접근이 발생할 때 해당 질의에 대한 감쇄율을 적용하여 갱신회수를 증가시키고, 원격지서버로부터 질의의 처리결과를 제공받아 사용자에게 전달하며, 동시에 새롭게 구체화시킨다.

4. 효율적인 학습을 위한 정보 순위 결정 방법

미디어이터 시스템의 저장공간은 제한적이며, 질의의 활용형태의 변화가 유동적으로 변경될 수 있으므로 한정된 저장공간에서 현재 질의의 활용패턴을 고려한 효율성을 보장하는 것은 매우 중요한 문제이다.

본 논문에서 각 질의의 처리비용은 감쇄율에 근거한 활용빈도를 기반으로 현재시점에서 계산되어진다. 수정질의는 각 질의 별로 활용빈도 만큼 원격지서버에서 주어진 질의를 처리하여 결과를 미디어이터에게 전달하기 때문에 임의의 수정질의 q_i 의 비용은 식(4)와 같이 표현된다.

$$c(q_i, mod) = e_i \cdot size(q_i) \quad (4)$$

이때 $size(q_i)$ 는 네트워크를 통해 전달되는 가장 최근의 질의 q_i 의 결과크기를 나타낸다. 따라서 식(4)에 의해서 수정질의의 처리비용은 감쇄율이 적용된 접근회수의 가중치에 가장 최근의 질의 크기의 결과 값으

로 표현되어진다. 이때 원격지서버의 질의 처리비용이 고려되어야 하지만 네트워크를 통한 전송비용이 상대적으로 매우 크기 때문에 유비쿼터스 환경에서는 고려하지 않는다. 동일하게 구체화질의의 비용은 갱신회수만큼 결과를 전송 받기 때문에 식(5)와 같이 계산되어진다.

$$c(q_j, mat) = u_j \cdot size(q_j) \quad (5)$$

모든 질의에 대하여 최적화의 목적은 미디어이터의 주어진 저장공간을 최대한 활용하면서 최소의 질의 처리비용을 갖는 질의 집합을 찾는 것이다. 미디어이터에서 새로운 질의집합을 찾는 시점이 되면 미디어이터 시스템의 연결형태와 질의의 관리방법에 따라 식(4)와 식(5)에 의해서 각 질의의 두 가지 구현방법 및 그에 따른 처리비용을 계산한다. 이를 기반으로 각 질의의 구현방법을 결정하는 중요한 요소인 이득을 계산한다. 각 질의는 수정질의의 처리비용과 구체화질의의 처리비용의 차가 클수록 구체화질의로 관리하는 것이 처리비용이 적고 효율적이기 때문에 질의 q_i 에 대한 이득은 식(6)과 같이 수정질의 처리비용과 구체화질의의 처리비용의 차로 정의한다.

$$g(q_i) = c(q_i, mod) - c(q_i, mat) \quad (6)$$

이득이 질의의 구현방법을 결정하는 중요한 요소이나 각 질의의 결과를 구체화하는데 필요한 저장공간의 활용도를 고려하지 않는 단점이 있다. 따라서 이득으로만 질의의 구현방법을 결정하게 되면 용량이 큰 질의가 구체화질의가 될 확률이 높기 때문에 미디어이터의 한정된 저장공간의 활용도를 고려하여 질의의 크기를 고려하여야 한다. 다음 그림 2는 이러한 내용들을 모두 고려한 구체화 질의선택 알고리즘을 자세하게 나타낸다.

```

SELECTION_MATERIALIZED_QUERY(Q)
For each  $q_i \in Q$  ( $1 \leq i \leq n$ )
    Determine the  $G\_rank(q_i)$  and  $S\_rank(q_i)$ 
     $O\_rank(q_i) = (G\_rank(q_i) + S\_rank(q_i)) / 2$ 

Sort  $Q$  by ascending order based on the value of  $O\_rank(q_i)$ 
 $MA = \emptyset$ 
For each  $q_i \in Q$  ( $1 \leq i \leq n$ )
    If  $size(q_i) \leq T - Size(MA)$  Then
         $MA \leftarrow MA \cup \{q_i\}$ 
 $MO \leftarrow Q - MA$ 
    
```

(그림 2) 구체화질의의 선택 알고리즘

전체 질의집합을 Q 라고 할 때 질의집합 Q 는 $MA \cup MO = Q$, $MA \cap MO = \emptyset$ 를 만족하는 구체화 질의집합 MA 와 수정질의집합 MO 로 나타낼 수 있다. 또한 미디어이터의 저장공간을 T 라고 할 때 $Size(MA) \leq T$ 이다. $G_rank(q_i)$ 와 $S_rank(q_i)$ 를 각각 질의 q_i 의 이득의 순위와 질의결과 크기의 순위라고 할 때 $O_rank(q_i)$ 는 질의 q_i 에 대한 두 순위의 평균순위라고 정의한다. 이렇게 구해진 $O_rank(q_i)$ 에 의해서 질의구현방법이 결정되어진다.

예를 들어 총 7개의 질의로 구성된 집합 $Q = \{q_1, q_2, q_3, q_4, q_5, q_6, q_7\}$ 가 저장공간이 90MB인 미디어이터에서 처리된다고 하자. 이때 각 질의에 대한 학습자 활용패턴에 따른 처리비용과 그에 따른 순위가 표 1과 같다고 하자.

<표 1> 구체화 질의 선택 예제

	q_1	q_2	q_3	q_4	q_5	q_6	q_7
용량	22	27	36	14	32	20	18
구체화 비용	6910	13908	11367	3127	13248	10292	3800
수정비용	22520	24840	44172	8255	29008	12312	15908
이득	15610	10932	32805	5128	15760	2020	12108
이득순위	3	5	1	6	2	7	4
용량순위	4	3	1	7	2	5	6
전체순위	3.5	4	1	6.5	2	6	5

각 질의의 구체화비용 및 수정비용은 식(4)와 식(5)에 의하여 계산되어지고, 이득은 식(6)에 의해서 계산되어진다. 따라서 전체 미디어이터의 질의처리공간이 90MB이므로 전체순위에 의해 q_3 , q_5 및 q_1 이 구체화질의가 된다. 이는 학습자가 가장 많이 학습하는 내용이 q_3 , q_5 및 q_1 이라는 것을 의미한다.

5. 실험 결과 및 분석

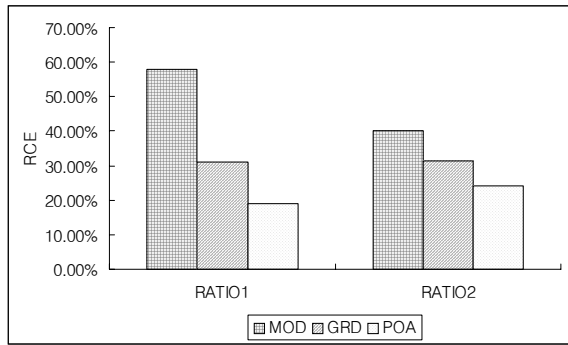
본 논문에서는 제안된 알고리즘을 다양한 관점에서 평가하기 위하여 여러 가지 실험을 수행하였다. 본 장에서는 특별한 설명을 하지 않을 경우 실험에서 300개의 학습 내용을 사용하였으며, 학습 내용을 처리하는 미디어이터의 전체 저장공간은 3GB로 설정하였다. 또한 각 학습내용의 초기 결과 크기는 10MB에서 200MB사이에서 자유롭게 생성하였다. 학습내용에 대한 학습자의 활용빈도와 학습 내용의 갱신빈도에 대한 정확성과 다양한 환경의 결과를 제공하기 위하여 서로 다른 두 종류의 로그로 실험하였다. 첫 번째 로그는 학습내용의 갱신이 발생하지 않고 학습자의 사용만 발생하는 것으로써 $RATIO1$ 이라고 한다. 일반적인 유비쿼터스 학습에서는 $RATIO1$ 의 로그가 사용되어 진다.

또 다른 로그는 학습내용에 대한 학습자의 사용빈도와 학습내용의 갱신빈도가 50:50의 비율을 갖는 것으로써 이를 $RATIO2$ 라고 하였다. 각 학습내용들에 대해 3000번의 사용 및 갱신빈도를 위의 비율에 따라 자유롭게 적용하였다. 마지막으로 각 학습내용들에 갱신이 발생했을 경우 이전 학습내용 결과의 0.1배에서 10배 사이에서 자유롭게 생성되도록 하였으며, 실험의 정확성을 위해서 각 실험들은 10번씩 수행하여 평균값으로 표현하였다.

본 논문에서 제안하는 구체화질의 선택방법의 효율성을 표현하기 위하여 본 논문에서 제안된 알고리즘(POA : Practically Optimum Algorithm)은 일반적인 최적화 방법인 탐욕적인 방법(GRD : GReedy algorithm)[13]과 기존의 미디어이터 시스템에서 사용된 수정방법만 사용하였을 경우 (MOD : MODification only)와 결과를 비교하였다. 또한 최적의 상태를 찾기 위하여 항상 최적의 계획을 찾는 방법인 분기한정방법[13]을 사용하였으며, 최적의 비용과 비교하여 모든 결과를 표현하였다. 따라서 최적비용(OC : Optimum Cost)과 최적화 방법을 통한 질의처리비용(TC : Total evaluation Cost)의 차이를 연관비용 효율성(RCE : Relative Cost Efficiency)이라고 정의하고, 모든 실험 결과를 RCE 를 사용하여 표현하였다. 따라서 RCE 를 다음과 같이 정의한다.

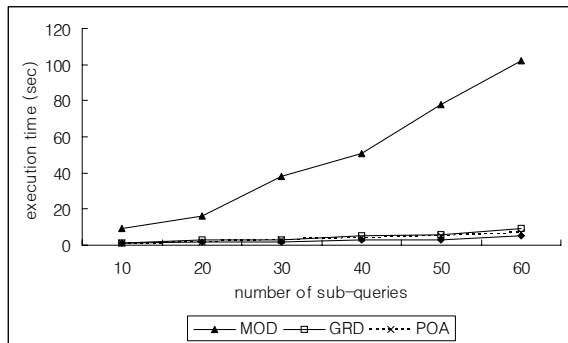
$$RCE = \left(\frac{TC - OC}{OC} \right) \times 100 (\%)$$

그림 3은 본 논문에서 제안하는 방법의 효율성을 다른 최적화 방법들과 비교하여 나타낸다. 유사 데이터에 대한 연속적인 최적화 결과가 아닌 단일 최적화 과정에 대한 비용만을 비교하기 위하여 $RATIO1$ 로그와 $RATIO2$ 로그 각각에 대해 첫 번 최적화 과정의 처리 비용만으로 비교하였다. 그림 3에서 보듯이 본 논문에서 제안한 방법이 최적화에 가장 가깝게 접근함을 알 수 있다. 평균적으로 제안된 본 논문에서 제안된 방법은 수정방법만 사용한 방법보다 40%의 비용감소를 보였다. 또한 실제 최적화 비용과는 19%의 차이를 보였다. $RATIO2$ 로그에 대해서는 수정방법만 사용하였을 경우보다 16%의 비용감소를 가져왔으며, 실제 최적화 비용과는 24%의 차이를 보였다. 즉, $RATIO1$ 로그일 경우 $RATIO2$ 로그보다 더 좋은 효과를 얻었다. 이는 학습내용의 갱신회수가 적을수록 더욱 좋은 효과를 얻는다는 것을 알 수 있다. 따라서 유비쿼터스 학습에서 더욱 좋은 효과를 기대할 수 있다.



(그림 3) 학습자 활용패턴에 따른 비용비교

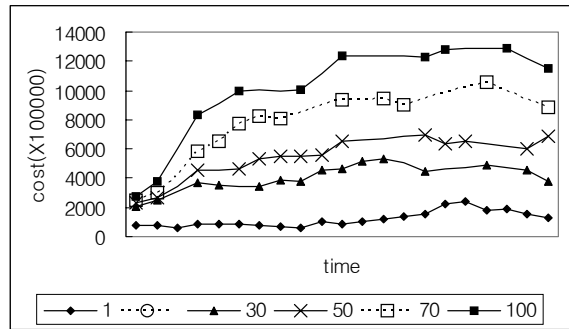
그림 4는 다양한 최적화 방법들의 학습자의 질의 처리시간과 최적화 계획의 처리시간과의 비교를 나타낸다. 각 미디어이터의 학습내용의 관리 수를 3에서 60 까지 변화를 시키면서 실험을 수행하였다. 그림 4에서 보듯이 관리하는 학습내용의 수가 증가할수록 수행시간의 변화가 증가하였다. 그림 3의 RATIO1 로그에서 비록 최적화 비용과 본 논문에서 제안하는 방법이 19%의 차이를 보였으나, 최적화를 수행하는 시간은 최적화 비용을 구하는 것 보다 훨씬 빠르다. 탐욕적인 방법은 최적화 수행시간은 실제구현계획 방법과 거의 비슷하지만 그림 3에서 최적화 비용의 차이가 본 논문에서 제안하는 방법이 훨씬 우수하기 때문에 더 효과적인 방법이라고 할 수 있다. 평균적으로 RATIO1 로그와 RATIO2 로그의 수행시간은 거의 동일하다.



(그림 4) 전체 최적화 수행 시간 비교

감쇄율의 효과를 보기 위하여 제안된 그림 5에서는 알고리즘에 대해 감쇄율의 효과를 나타내었다. RATIO1 로그에서 학습내용에 대한 연속적인 활용패턴의 변화에 대해 각각 반감기를 다르게 적용했을 때의 연속적인 최적화 과정의 수행변화를 나타내며, RATIO2 로그에서도 거의 같은 결과를 나타낸다. 그림 5에서 보듯이 반감기가 작을수록 최적화 과정이 자주 수행되어진다. 따라서 반감기가 작을수록 최근의 학습내용의 활용패턴의 변화에 더 높은 가중치를 부여

하게 되어 최근의 학습자가 학습한 학습내용의 활용패턴에 민감하게 적용할 수 있다. 유비쿼터스 학습에서 사용자들이 최근에 자주 사용하는 학습내용들은 오래 전에 학습했던 내용들보다 더 자주 사용할 가능성이 높기 때문에 반감기를 조정함으로써 학습자에게 정확한 정보를 신속하게 제공할 수 있다.



(그림 5) 반감기의 효과

6. 결론

본 논문에서 제안된 알고리즘은 각 학습내용들의 활용패턴을 주기적으로 관찰함으로써 유비쿼터스 환경에서 다양한 학습을 가능하게 하였다. 이를 위해 학습내용의 활용패턴에 대해 감쇄율을 적용함으로써 항상 최근의 활용패턴에 민감하게 반응하여 정확한 학습내용을 신속하게 전달 할 수 있다. 또한 본 논문에서 제안된 알고리즘은 학습내용의 수가 크게 증가하여도 미디어이터 저장공간의 효율성을 최대로 활용할 수 있기 때문에 실제적으로 매우 효율적이나 항상 최적의 상황이 아닐 수도 있다. 그러나 본 논문에서 제안하는 알고리즘은 처리비용과 시간의 많은 절감을 가져오기 때문에 매우 효율적이며, 이를 다양한 실험을 통하여 증명하였다.

7. 참고문헌

- [1] 유영만 (2000), e-Learning의 정체. 그리고 가능성과 한계
- [2] Anthony Tomasic, Remy Amouroux, Philippe Bonnet, Olga Kapitskaia, Hubert Naacke and Louiqa Raschid (1997), "The Distributed Information Search Component(DISCO) and the World Wide Web", Proceedings of the ACM SIGMOD International Conference of Management of Data, pp.546-548, AZ, USA.

[3] Anthony Tomasic, Louiqa Raschid and Patric Valduriez (1998), "Scaling Access to Heterogeneous Data Source with DISCO", IEEE Transactions on Knowledge and Data Engineering, Vol 10, No.5, pp.808-823.

[4] A. Leinwand and K. F. Conroy (1996). "Network Management", Addison-Wesley Publishing Company, Inc..

[5] Briton, D. and Taylor, J. (2001), "Online workers education: How do we tame the technology." International Journal of Instructional Media, Vol. 28, No. 2, pp. 117 - 135.

[6] Chen Li, Ramana Yerneni, Vasilis Vassalos, Hector Garcia-Molina, Yannis Papakonstantinou, Jeffrey Ullman and Murty Valiveti (1998). "Capability Based Mediation in TSIMMIS", Proceedings of the ACM SIGMOD International Conference of Management of Data, pp.564-566, Seattle, WA, USA.

[7] Chuan Zhang, Xin Yao, Jian Yang (2001), 'An Evolution Approach the materialized view selection in a data warehouse environment' IEEE Trans. On systems, man and cybernetics, part C, Vol. 31, No.3.

[8] Harold S.Javitz and Alfonso Valdes (1994), "The NIDES Statistical Component: Description and Justification", Technical report, SRI International, Menlo Park, California.

[9] H. Gupta and L. S. Mumick (1999), "Selection of views to materialize under a maintenance cost constraint", Proceedings of the International Conference on Database Theory, pp. 453-470, Jerusalem, Israel.

[10] Kil Hong Joo and Won Suk Lee (2004), "Adaptive Selection of Materialized Queries in a Mediator for the Integration of Distributed Information Resources", The book of High Performance Scientific and Engineering Computing, Kluwer book.

[11] Nita Goyal, Charles Hoch, Ravi Krishnamurthy, Brian Meckler and Michael Suckow (1996), "Preliminary Report on (Active) View Materialization in GUI Programming" proceeding of the Workshop on Materialization Views : Techniques and Applications, pp. 56-64.

[12] Peter H. Sawchuk, Zenon Gawron and

Jeffery Taylor (2003), "E-Learning and Union Mobilization", Journal of distance education Vol. 17, No. 3, pp.80-96.

[13] Richard E. Neapolitan, Kumarss Naimipour (1997), Foundations of algorithms using C++ pseudocode, Jones and Bartlett publishers.

[14] Sophie Cluet, Claude Delobel, Jerome Simeon and Katarzyna Smaga (1998), "Your Mediators Need Data Conversion!", Proceedings of the ACM SIGMOD International Conference of Management of Data, pp.177-188, Seattle, WA, USA.

[15] V.S. Subrahmanian, Sibel Adali, Anne Brink, Ross Emery, James J.Lu, Adil Rajput, Timothy J.Rogers, Robert Ross and Charles Ward, HERMES : A Heterogeneous Reasoning and Mediator System, <http://www.cs.umd.edu/projects/hermes/overview/thesis>.

[16] Weiser, M.,(1991) "The computer for the 21st Century", Scientific American, Vol. 265, No. 3, pp. 94-104.

[17] Weiser, M., (1993) "Some computer science issues in ubiquitous computing", Communication of ACM, Vol. 36, No. 7, pp. 75-84.

저자 소개

주길홍



1998년 인천대학교 전자계산학과 (공학사)

2000년 연세대학교 컴퓨터과학과 (공학석사)

2004년 연세대학교 컴퓨터과학과 (공학박사)

1998 - 2004 연세소프트웨어 응용연구소 연구원

2000-2004 연세 프로테움 연구센터 연구원

2005년 - 현재 경인교육대학교 컴퓨터교육과 전임 강사

관심분야 : 컴퓨터교육, 데이터베이스, 분산처리, 미디어이터 시스템, 유비쿼터스

E-mail : faholo@amadeus.yonsei.ac.kr