# COVA: 내용 기반 강의 검색을 지원하는 원격 학습 시스템

## (COVA: A Distance Learning System supporting Content-based Lecture Retrieval)

차 광 호 †

(Guang-Ho Cha)

**요 약** 인터넷, 데이타베이스, 멀티미디어 기술의 복합적인 영향으로 교육과 학습의 형태가 크게 변하고 있다. 그러나 강의 내용을 효과적으로 관리하고 검색할 수 있는 시스템과 도구의 부족으로 원격 학습은 크게 효과적이지 못하다. 이 논문은 대용량 강의 데이타베이스에서 사용자가 내용에 기반하여 관심있는 강의 부분만 발췌하여 접근할 수 있도록 하는 프로토타입 시스템 COVA를 소개한다. COVA는 원격 학습에서 내용 기반 강의 검색을 위한 다음과 같은 새로운 기법을 포함한다: (1) 강의 내용을 표현하기 위한 XML 기반의 준구조적(semistructured) 데이타 모델; (2) XML 강의 데이타베이스의 구조적 요약, 즉, 스키마 추출 기법; (3) 원하는 강의 부분의 빠른 탐색을 위한 색인 기법.

**키워드** : 내용 기반 검색, 원격 학습, XML, 준구조적 데이타 모델

***Abstract*** Education and training are expected to change dramatically due to the combined impact of the Internet, database, and multimedia technologies. However, the distance learning is often impeded by the lack of effective tools and system to manage and retrieve the lecture contents effectively. This paper introduces a prototype system called COVA that enables remote users to access specific parts of interest by contents from a large lecture database. COVA includes several novel techniques to achieve the content-based lecture retrieval in distance learning: (1) The XML-based semistructured model to represent lecture contents; (2) The technique to build structural summaries, i.e., schemas, of XML lecture databases; (3) Index structures to speed up the search to find appropriate lecture contents.

**Key words** : content-based retrieval, distance learning, XML, semistructured data model

## 1. 서 론

인터넷, 데이타베이스, 멀티미디어 기술의 복합적인 영향으로 교육과 학습의 형태가 크게 변하고 있다. 온라인 교육은 강의실을 직접 가지 않아도 되는 경제적인 효과 외에도 교육 과정 자체를 크게 변화시키고 있다. 비디오는 이미지와 음성을 포함하는 표현력으로 인해 원격지 또는 미래의 사용자에게 강의를 제공하기 위한 가장 효과적인 매체라고 할 수 있으나 현재의 비디오에 기반한 강의는 원하는 강의 부분에 대한 검색의 어려움이 있다. 또한 각 강좌는 내용이 다양하고, 동일한 과목이라 하더라도 다른 강사에 의해 다른 내용으로 여러번 개설될 수 있으므로 특정 강좌의 내용을 엄격히 정의된 스키마를 따르게 하기가 어렵다. 본 논문에서는 내용 기반 원격 학습 시스템을 위한 다음 세가지 주요 주제를 다룬다: (1) 다양한 강의 내용의 표현 기법; (2) 효과적인 검색과 브라우징을 위한 강의 데이타베이스의 구조적 요약 기법; (3) 강의 데이타베이스 검색을 위한 색인 기법.

원격 학습을 위한 강의 데이타베이스에서 브라우징과 질의는 쉽게 사용할 수 있어야 하며, 원하는 특정 부분에 집중할 수 있도록 하는 것이 필요하다. 그러나 강의 데이타베이스에서 이것을 실현하는 것은 간단하지 않으며, 각 강의의 장, 절, 세부 항목 등을 구분해야 하고, 강의의 개괄을 살펴보고 원하는 것을 도출할 수 있도록 목차와 색인 페이지를 만들어야 한다.

하나의 강의를 가치있는 교육적인 도구로 만들려면

· 본 연구는 한국과학재단 목적기초연구(R05-2000-000-00403-0)지원으로 수행되었음

† 종신회원 : 숙명여자대학교 멀티미디어학과 교수
ghcha@sookmyung.ac.kr

논문접수 : 2003년 6월 9일
심사완료 : 2003년 12월 1일

다음의 다섯 단계가 필요하다: (1) 첫째, 하나의 강의를 내용이나 책의 계층적인 구조를 이용하여 개별적인 강의 세그먼트들로 분해한다. 하나의 강의 세그먼트는 강의 노트의 집합과 비디오 강의를 형성하는 연속적인 비디오 클립으로 구성된다. (2) 둘째, 각 강의 세그먼트의 내용을 키워드, 의미있는 속성, 이미지들로 요약하고 검색과 브라우징에 용이한 효과적인 구조로 구성한다. (3) 세째, 학습자가 강의 데이타베이스에 대해 질의를 형성하고 브라우징을 할 수 있도록 하는 도구가 필요하다. (4) 네째, 원하는 특정 강의 세그먼트를 빨리 찾기 위해 유용한 객체들을 색인할 필요가 있다. (5) 마지막으로, 탐색 공간을 줄이고 탐색 속도를 높이기 위한 질의 최적화 기법이 필요하다.

본 논문에서는, 내용 기반 강의 검색을 위해 XML에 기반한 준구조적 데이타 모델을 도입한다. 본 모델은 XML 데이타를 지원하고 엄격히 고정된 스키마 없이 강의 내용을 표현한다. 또한, 사용자에게 강의 데이타베이스에 대한 질의를 형성하고 브라우징을 돕기 위해 데이타베이스 구조 요약, 즉, 스키마 도출, 기법을 개발한다. 우리는 특정 강의 세그먼트뿐만 아니라 관련된 강의 세그먼트 집합에 대한 검색을 효율적으로 수행하기 위한 두가지 색인 기법을 개발한다.

## 2. 강의 세그먼트 구성

강의를 위한 비디오 클립은 일련의 프레임들로 구성되지만 강의 비디오 검색을 위해 개별적인 프레임을 검색 단위로 사용하기는 어렵고, 비디오의 의미있는 세그먼트(샷: shot)를 규정하고 그것을 검색 단위로 사용하는 것이 효과적이다. 그러나 이러한 샷 기반 강의 검색을 원격 학습에 적용하기에는 다음과 같은 문제점이 있다: (1) 강의 비디오에는 샷의 변화를 찾아 낼만한 시각적인 단서를 찾기가 어려우며, (2) 샷을 통해 강의의 의미 구조를 나타내기 어렵다.

이와 같은 이유로 본 논문에서는 영상 처리에 기반하여 비디오를 파싱하지 않고, 강사의 강의 노트로부터 텍스트 및 이미지 정보를 자동적으로 추출하고, 필요한 의미적인 비디오 세그먼트와 구조적인 정보를 수동으로 기술한다. 그 다음, 비디오 강의는 자동으로 색인하고, 웹에서 서비스할 수 있는 형태로 바꾼다.

하나의 강의는 발표용 슬라이드(즉, 강의 노트)와 비디오 세그먼트들로 구성된다. 각 슬라이드는 한 페이지의 강의 노트에 해당되고 XML로 표현된다. 학습자가 어떤 과목의 특정 강좌에 들어가서 만나는 강의 모습은 그림 1과 같은 형태이다. 사용자는 주 윈도우 내에 나타나는 강사의 비디오, 강의 노트 슬라이드, 각 강좌의 전체 구조를 나타내는 창을 통해 학습을 수행한다. 또한,

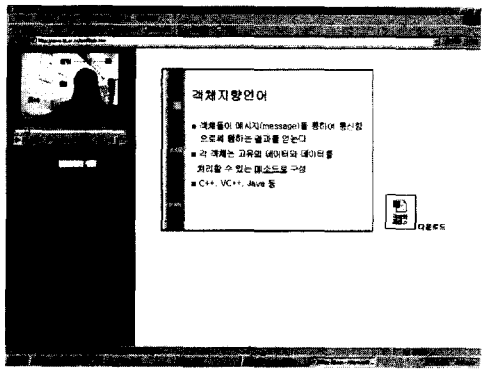


그림 1 온라인 강의 창

검색 창을 통해 해당 키워드를 포함하는 강의 객체(즉, 강의 슬라이드, 강의 절 또는 장, 또는 강좌 전체)를 검색할 수 있다. 또한 이미지 검색을 통해 해당 이미지와 유사한 이미지를 포함하는 강의 객체를 검색할 수 있다. 본 연구의 초점은 학습자가 원하는 특정 부분만을 효과적으로 발췌, 학습할 수 있는 시스템을 개발하는 것이다.

## 3. 데이타 모델

데이타 모델은 사용자의 접근이 용이하도록 데이타를 어떻게 표현하는가 하는 문제를 다룬다. 우리가 알기로는 지금까지 강의 데이타베이스를 모델링하고자 하는 시도는 없었다. 비디오 데이타를 위한 데이타 모델링 기법에 대한 연구는 많았지만, 강의 비디오의 경우에는 장면 전환에 따른 시각적인 차이가 거의 없어서 기존의 데이타 모델의 사용은 바람직하지 않다.

우리는 강의 노트에서 추출한 설명 정보를 표현하기 위해 준구조적 데이타 모델[1-3], 특히, XML에 기반을 둔 준구조적 모델을 사용한다. 준구조적 모델을 채택한 이유는 강의 내용 설명에 유연성과 다양성을 제공하기 위함이다. 강의 내용은 다양하고 풍부하기 때문에 강의 데이타베이스를 엄격히 미리 정의된 스키마에 따라 고정시키는 것은 바람직하지 않다. 또한 XML을 지원하는 이유는 인터넷을 통한 강의 데이타의 원활한 교환을 이루고자 하는데 있다. 준구조적 데이타는 미리 고정된 스키마를 갖고 있지 않는 데이타를 의미하며, 그 구조는 정형화되어 있지 않다. 준구조적 데이타를 위한 표준 모델[1-3]처럼 강의 데이타베이스는 레이블이 부착된 방향성 그래프로 생각할 수 있다. 그림 2는 3 개의 강의(두 개의 데이타베이스 강좌와 하나의 멀티미디어 강좌)를 포함하는 강의 데이타베이스의 한 부분이다. 각 노드는 XML element에 해당하고 작은 원으로 표현된 것은 XML attribute에 해당한다. 비록 준구조적 데이타 모델이 임의의 그래프 형태의 데이타베이스를 허용하지만,

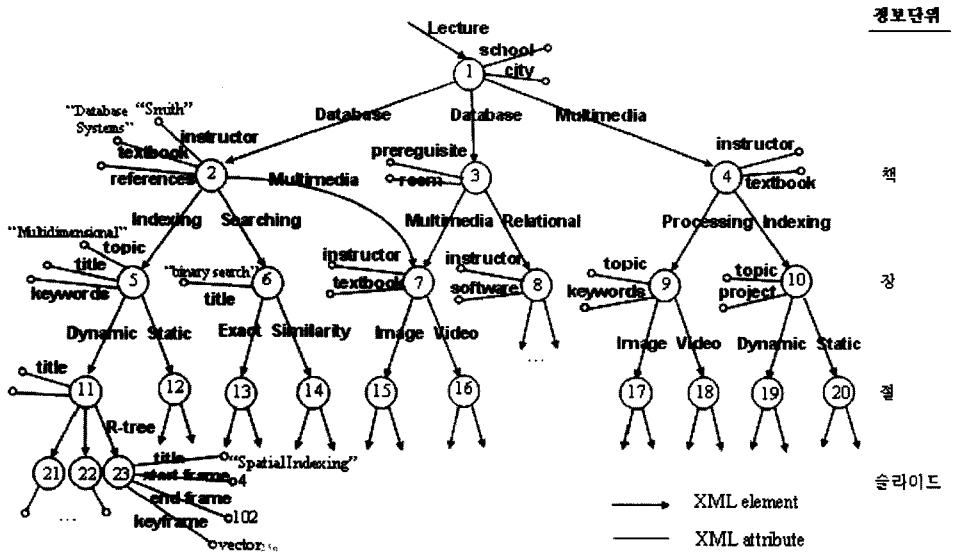


그림 2 강의 데이타베이스 예(일부 노드와 속성에 대해서만 표시)

그림 2의 데이타베이스는 강의 교재(책)가 대부분 계층 구조(책-장-절 등)의 특징을 갖고 있으므로 거의 트리 형태를 취하고 있다. 예제 데이타베이스의 각 레벨은 정보(강의 내용)의 세밀도를 나타낸다. 예를 들어, 노드 2에서 4까지는 책, 노드 5에서 10까지는 장, 노드 11에서 20까지는 절, 나머지 노드들은 슬라이드 자체를 나타낸다.

표준 준구조적 모델과 달리, 본 논문의 데이타 모델은 XML 데이타를 완전히 지원한다. 즉, 속성을 그래프 노드(XML elements)에 첨가할 수 있다. 본 데이타 모델에서는 그래프 상의 각 노드를 강의 객체(lecture object: LO)라 부르고, 각 강의 객체에는 해당 비디오 세그먼트와 강의 슬라이드, 그리고 검색을 위한 속성들이 부착된다. 하나의 강의 객체는 다음 6-요소로 구성된다: (PID, OID, 비디오 세그먼트, 강의 슬라이드 집합, 부분 요소들의 집합, 설명 속성의 집합). 여기서 중요한 한가지는 강의 객체에 부착되는 요소와 속성들은 미리 고정되지 않는다. 각 강의 객체(LO)는 유일한 객체 식별자를 갖고 있으며(그림 2의 노드 번호 1에서 23), 자신의 부분 요소로 가는 간선을 갖고 있다. 모든 강의 객체는 하나의 특정 형(type)에 속하고, 각 형은 경로 식별자(path identifier: PID)로 구분된다. 본 모델에서, 형은 데이타베이스로부터 추출하는 스키마 그래프 상의 경로에 의해 정의된다. 각 간선에 레이블이 부착되고 이 레이블은 강의 객체나 속성의 이름 역할을 한다. 그림 2의 예제 데이타베이스는 Lecture 데이타베이스를 나타내는 하나의 루트 LO와 세 개의 부분 LO(두 개의 Database, 하나의 Multimedia LO)로 구성된다. Database 강의 객체 LO 2는 그 것의 instructor, textbook, references를 나타내는 3 개의 속성-값 쌍을 갖고 있고, Database 강의 객체 LO 3는 2 개의 속성 prerequisite과 room을 갖고 있다. 표준 준구조적 모델과 달리 본 모델의 강의 객체 아래의 부분 강의 객체는 그것이 포함하는 비디오 세그먼트의 시간적 순서를 반영하기 위해 순서를 갖는다.

## 4. 강의 데이타베이스 요약

사용자는 질의와 브라우징을 통해 원하는 부분의 강의를 도출할 수 있다. 질의 처리기는 사용자에게 질의를 하기 위한 질의 형성 도구와 브라우징을 위한 최적의 시작점을 제공하여 이 두 가지 형태의 검색 요구를 처리한다. 강의 검색 요구를 일련의 질의와 브라우징의 반복적인 과정으로 모델링할 때, 질의 처리기는 각 단계에서 검색 공간을 줄이는 필터 역할을 수행하여 다음 단계에 더 정제된 검색공간을 제공한다. 우리는 전체 데이타베이스를 요약하는 정보(즉, 스키마)를 제공하여 사용자가 질의 및 브라우징을 할 수 있도록 하고, 스키마는 또한 색인과 질의 최적화를 통해 시스템 성능을 향상시키는데 큰 역할을 한다.

그림 3은 그림 2의 강의 데이타베이스를 요약한 것이다. 사각형은 데이타베이스의 XML element이고, 작은 검은 원은 XML element에 있는 XML attribute이다. 원 데이타베이스의 모든 XML element는 거기에서 몇

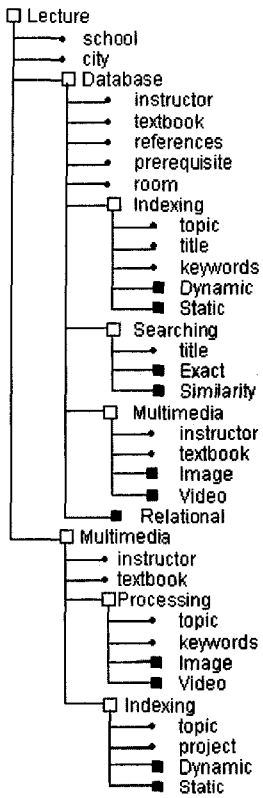


그림 3 그림 2의 예제 데이타베이스에 대한 구조적 요약

번 나왔는지에 관계없이 구조적 요약에서 정확히 한 번만 기술되고, 원 데이타베이스에 나오지 않는 XML element가 구조적 요약에서 나오는 경우는 없다. 구조적 요약을 통해 사용자는 그래프 기반의 데이타베이스를 질의하고 브라우즈할 수 있다. 구조적 요약의 사각형을 클릭함으로써 강의 객체를 펴거나 접는다. 흰색의 사각형은 강의 객체가 펼쳐진 것을 나타내고, 검은 사각형은 접혀진 상태를 표시한다. 예를 들어, Database와 Multimedia 강의 객체는 펼쳐져 있고, Dynamic과 Static 강의 객체는 접혀있다.

*DataGuide*[4]는 준구조적 데이타베이스의 간결하고 정확한 요약이다. 그러나 불행히도 DataGuide는 원 데이타베이스에 대한 멱집합(powerset)이 되어 계산 비용이 굉장히 비싸다. 일반 그래프에서 DataGuide를 형성하는 알고리즘은 원 데이타베이스에 대해 시간과 공간 비용이 지수에 비례할 수 있다. Buneman 등[5]은 이를 해결하기 위해, 요약한 데이타베이스의 크기가 원 데이타베이스 크기의 선형 비례하도록 *simulations*나 *bisimulations*[6]에 기초하여 데이타베이스 요약을 형성하였다. 그러나 요약한 스키마 그래프의 간선에서 중복이 나타나 여전히 그 크기가 만족스럽지 못하다. Nestorov 등[7]은 데이타로그(Datalog) 프로그램과 클러스터링의 최대 fixpoint 개념에 기초하여 스키마를 추출하는 기법을 개발하였다. 비록 이 기법은 스키마의 크기를 원하는 크기로 줄일 수 있지만 알고리즘의 복잡도로 인해 실행 비용이 아주 비싸다. 또한 이 방법은 원하는 크기의 스키마를 얻기 위해 클러스터링을 수행해야 하므로 동적인 환경에서는 사용하기 어려운 문제가 있다.

본 연구에서는 짧은 시간에 가능한한 작은 크기의 데이타베이스 요약을 형성하는 기법을 개발한다. 이 기법으로 만들어진 요약은 스키마 그래프에서 노드와 간선에 어떠한 중복도 포함하지 않는다. 먼저 예를 통해서 우리의 기법과 Buneman 등의 기법, DataGuide, 그리고 Nestorov 등의 기법을 비교해 보자. 그림 4는 (가) 데이타베이스 그래프 *DB*와 (나) *DB*에 대한 본 논문의 요약 결과, (다) Buneman 등의simulation에 기초한 요약, (라) DataGuide, (마) Nestorov 등의 최소 완전 형(minimal perfect typing)에 의한 스키마를 나타낸다. 크기에 기반하여 스키마를 비교해보자. DataGuide는 원 데이타베이스에 대한 멱집합을 필요로 하므로 최악의 경우에 원 데이타베이스 크기의 지수에 비례하는 비용이 될 수 있다. 그림 4(라)에서 보면 요소 7과 13이 DataGuide의 노드에서 중복되어 있다. simulation에 기초한 스키마의 크기는 원 데이타베이스의 크기에 선형으로 비례한다. 그러나 그림 4(다)에서 보듯이 같은 노드에서 나오는 간선 *a*가 중복되어 있다. 그림 4(마)에서는 노드와 간선 모두에 중복이 있음을 볼 수 있다. 반면에, 그림 4(나)의 우리의 데이타베이스 스키마는 노드와 간선에서 어떠한 중복도 보이지 않는다. 본 스키마의 이러한 간결성은 데이타베이스 요약뿐만 아니라 질의 계산에서도 효율성을 나타낸다.

### 4.1 알고리즘

먼저 몇 가지 용어를 정의한다.

**정의 1.** 데이타 객체(data object)는 데이타베이스 그래프에 있는 노드, 즉, 강의 객체(LO)이다.

**정의 2.** 경로 *l*에 대한 목표 집합(target set)은 데이타베이스 그래프에서 경로 *l*을 통해 도달할 수 있는 데이타 객체들의 집합이다.

**정의 3.** 스키마 객체(schema object)는 데이타베이스 그래프에서 경로 *l*의 목표 집합에 대응하는 스키마 그래프에 있는 노드이다.

본 논문의 스키마 추출 알고리즘은 간단하다. 데이타베이스 그래프의 루트 데이타 객체가 루트 스키마 객체가 된다. 깊이 우선 순위 탐색으로 한 스키마 객체에서 나오는 모든 유일한 경로를 통해 도달할 수 있는 모든 자식 스키마 객체를 도출한다. 유일한 경로 *l*에 대한 새

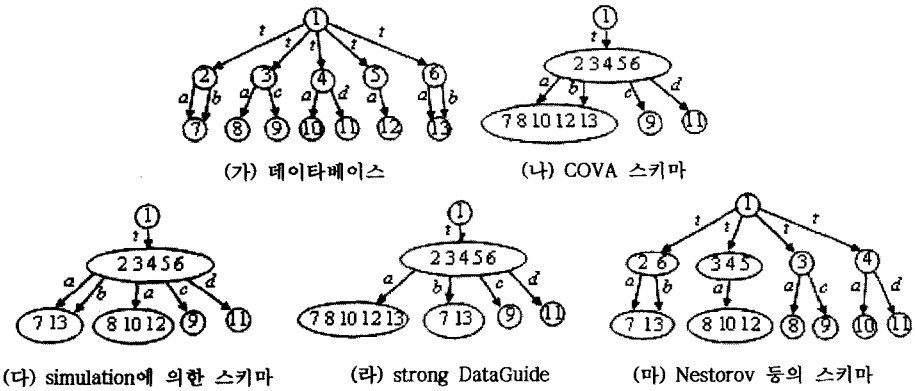


그림 4 데이타베이스 스키마 비교

로운 목표 집합을 만날 때마다 새로운 스키마 객체 $s$를 만든다. 경로 $l$를 통해 스키마 객체 $s$에 도달하고, 만약 데이타 객체 $o$가 이미 다른 경로 $m$을 통해 스키마 객체 $s$에 포함되어 있다면 새로운 스키마 객체를 만들지 않고 단순히 간선 $l$을 스키마 객체 $s$에 첨가한다. 스키마 형성 알고리즘은 다음과 같다.

```
Algorithm ExtractSchema(o)
// Input: root oid o of a database
// Output: database schema s
{
   s := CreateSchemaObject();
   Insert {o} to s;
   RecursiveMake(s);
}
Algorithm RecursiveMake(s)
{
   Let S be a set of current target sets under s;
   Let Sj denote a certain target set included in S;
   For each unique label li outgoing from s {
      o := target set reachable by li;
      If (o and Sj have data objects in common) {
         Add an edge li from s to the schema
            object corresponding to Sj;
         Sj := o ∪ Sj;
      }
      Else {
         s2 := CreateSchemaObject();
         Insert s2 to s;
         Add an edge li from s to s2;
         RecursiveMake(s2);
      }
   }
}
```

## 5. 색인 기법

전통적인 데이타베이스에서는 특정 속성 값을 갖는 객체를 빨리 찾기 위해 하나의 속성에 대해 색인을 형성한다. 준구조적 데이타 모델을 갖는 강의 데이타베이스에서는 데이타베이스 그래프를 효율적으로 순회해야 하므로 전통적인 데이타베이스에서 사용하는 값에 기반한 색인(value index) 만으로는 충분하지 않다. 본 논문에서는 데이타베이스 내에서 관련된 객체와 특정 간선 및 경로를 찾는데 유용한 다수의 색인이 필요하다. 또한, 강의 노트에 있는 그림이나 이미지에 기반하여 관련 강의 부분을 찾기 위한 이미지 색인 기법도 필요하다. 강의 데이타베이스에 대한 접근은 대체로 읽기 위한 것이므로 질의 처리 속도를 높이기 위해 여러 형태의 색인 구조를 사용하는 것은 타당하다.

본 논문에서는 두 개의 새로운 색인 구조 P-index (path index)와 GB-index(grid bitmap index)를 소개한다. P-index는 데이타베이스 그래프 상의 경로를 색인하고, GB-index는 강의 노트 상의 이미지를 색인 하기 위한 것이다. GB-index의 더 자세한 사항은 [8]을 참고할 수 있다.

### 5.1 P-index

여기서는 몇 가지 예와 함께 데이타베이스 그래프 상의 경로를 색인하기 위한 P-index를 소개한다.

5.1.1 예제

**보기 1**: title이 "Spatial Indexing"인 강의 객체를 도출하라.

**Select** x
**Where** *.x.title = "Spatial Indexing"

와일드 문자 "*"는 길이 0 이상의 임의의 경로를 나타낸다. 이러한 형태의 값에 의한 질의의 처리를 위해 속성 title에 대한 B+-tree 색인 구조를 고려할 수 있으며, 결과로 그림 2의 LO 23을 얻을 수 있다. 그러나, 만약 목표 강의 객체와 함께 강의 객체들 간의 구조적인 상호 관계가 주어지지 않는다면, 단순히 같은 값을 갖는 객체

를 탐색하는 것만으로는 유용한 결과를 얻지 못할 수 있다. 예를 들어, 강사는 학생이 이전의 강의 객체를 학습하지 않고는 다음 강의 객체를 읽지 않는다고 가정할 수 있다. 따라서, 학생은 앞의 강의 객체를 읽지 않고는 문맥을 이해하지 못할 수 있다. 따라서 만약 학생이 원한다면 다음과 같이 그래프 계층 구조를 순회할 수 있도록 질의 결과를 출력하는 것이 바람직하다:

- Lecture (LO 1)
  - Database (LO 2)
    - Indexing (LO 5)
      - Dynamic (LO 11)
        - R-tree (LO 23)

**보기 2**: title이 "Spatial Indexing"인 Database 강의 객체를 도출하라.

**Select** x
**From** Database x
**Where** x.*.title = "Spatial Indexing"

이러한 형태의 질의는 title이 "Spatial Indexing"인 강의 객체를 찾은 후에 모든 Database 객체까지 역으로 순회해야 하기 때문에 적절한 색인이 없다면 처리 비용이 매우 비쌀 수 있다. 위의 질의를 처리하기 위해서 위에서 아래로 검사를 진행할 수도 있다, 즉, 모든 Database 강의 객체를 찾고, 거기서 시작하여 모든 경로를 평가하여 아래에 있는 객체의 title이 "Spatial Indexing"인지를 검사할 수도 있다. 이러한 형태의 탐색은 Database 객체에서 시작하여 모든 경로를 순회해야 하므로 질의 처리 비용이 비싸다. 이러한 형태의 질의를 지원하기 위해 본 논문에서는 해당 객체로부터 루트까지의 역경로를 따라 모든 객체를 저장하는 색인 기법을 제시한다.

5.1.2 P-Index Structure

본 연구의 준구조적 데이타 모델에서처럼 형(type)이 완화된 데이타 모델에서는 색인되는 속성이나 경로에 대해 사용자가 기술하거나 동적으로 제어하는 형 제약자를 부과할 수 있어야 한다. 우리는 데이타베이스의 스키마 그래프 상의 레이블이 붙여진 경로에 대해 형을 부여한다. 예를 들어, 경로 Database.Indexing.Dynamic.R-tree에 대해 4가지 형을 부과한다: Database, Database.Indexing, Database.Indexing.Dynamic, Database.Indexing.Dynamic.R-tree. 각 형(또는 경로)는 경로 식별자(PID)에 의해 유일하게 결정된다.

P-index의 구조는 $B^+$-tree에 기초한다. P-index는 다른 동적 색인 구조처럼 내부 노드와 리프 노드로 구성된다. 내부 노드는 $B^+$-tree와 같은 형태의 구조를 갖고, 리프 노드는 내부 노드와는 다른 구조를 갖는다. 리프 노드는 $f$ 개의 색인 엔트리로 구성되고 ($f$는 리프 노드의 fanout) 각 색인 엔트리는 그림 5와 같은 구조를 갖는다. 경로 색인을 위해 P-index는 리프 노드에 특정 값을 갖는 객체로부터 루트까지의 레이블 경로 상의 강의 객체를 포함한다. 만약 리프 노드 엔트리의 크기가 한 페이지 크기를 초과하면 추가 범람 페이지(overflow page)를 할당한다. P-index는 객체지향 데이타베이스에서 사용된 클래스-계층구조 색인기법 [9]과 개념적으로 다소 비슷하다. 클래스-계층구조 색인기법은 클래스 계층 구조의 한 공통 속성에 대해 하나의 색인을 유지한다. 반면에 비정형적인 준구조적 데이타베이스에서는 공통 속성의 개념이 없고, P-index는 특정 값을 갖는 객체로부터 루트까지의 모든 레이블 경로에 대해 하나의 색인을 유지한다.

### 5.2 GB-index

사용자는 강의 노트 속의 특정 이미지나 그림을 포함하는 강의 객체를 찾기 위해 데이타베이스를 질의할 수 있다.

**예제 3**: 주어진 이미지 $P$와 비슷한 이미지를 포함하는 강의 객체를 구하라.

**Select** x
**Where** x.*.image $\approx P$

이미지 색인의 두 가지 주된 이슈는 차원의 저주 (curse of dimensionality)와 복합 유사 질의 (complex similarity query) 처리이다. 차원의 저주는 데이타를 표현하는 차원의 수가 늘어남에 따라 검색 성능이 급격히 나빠짐을 나타낸다. 이러한 고차원 데이타 공간에서는 전통적인 정교한 색인 기법들의 성능이 질의 객체와 데이타베이스 객체를 하나씩 비교하는 순차 탐색의 성능보다도 오히려 나빠진다. 복합 유사 질의는 다음과 같이 여러 특성들이 불리언(Boolean) 조합으로 연결된 유사 질의를 의미한다:

($topic$ = *'mobile database'*) $\wedge$ ($color$ = *'red'*) $\wedge$ ($texture$ = *'smooth'*) $\vee$ ($shape$ = *'round'*).

고차원 데이타 공간에서 이러한 복합 질의를 효율적으로 처리하기 위해 본 연구에서는 새로운 비트맵 색인 (bitmap index)인 GB-index를 도입한다. 비트맵 색인은 AND, OR로 결합된 복합 질의 처리에 아주 효과적

| key value | overflow page pointer | number of PIDs | $PID_1$ | number of OIDs | {$OID_{11}$,..., $OID_{1i}$} | ... | $PID_N$ | number of OIDs | {$OID_{N1}$,..., $OID_{Nj}$} |
|---|---|---|---|---|---|---|---|---|---|

그림 5 P-index의 리프 노드의 엔트리 구조

이며, GB-index에서 새롭게 고안된 유사성 척도(similarity measure)와 색인 기법은 차원의 저주 문제를 효과적으로 처리한다.

비트맵은 단순히 비트들의 배열이다. 속성 $A$에 대한 비트맵 색인은 $A$가 취할 수 있는 각 값에 대해 하나씩의 비트맵으로 형성된다. 각 비트맵은 데이타베이스에 있는 객체의 수만큼의 비트로 구성된다. 값 $v_j$에 대한 비트맵의 $i$번째 비트는 만약 i번째 객체가 속성 $A$에 대해 값 $v_j$를 가지면 1로, 그렇지 않으면 0으로 지정된다.

복합 질의를 효과적으로 처리하기 위해선 각 속성(또는 각 데이타 차원)을 독립적으로 처리하는 것이 필요하다. 따라서 유사성 척도에서도 각 차원을 독립적으로 처리할 수 있어야 한다. GB-index는 각 차원에 대해 데이타베이스 객체들을 $K$-means 알고리즘에 따라 $K$ 개의 클러스터로 구분한다. 이 때, 각 클러스터가 포함하는 객체의 수는 최소한 $n$ 개 이상이 되도록 한다. 각 차원 $i$의 $j$ 번째 클러스터 $I_{ij}$에 대해 하한치 $l_{ij}$와 상한치 $u_{ij}$를 설정한다. 여기서, 차원 $i$에서의 두 객체 $x$와 $y$ 간의 거리 함수 $Dist_i$는 다음과 같이 정의된다:

$$Dist_i(x_{ij}, y_{ij}) = \frac{|x_{ij} - y_{ij}|}{u_{ij} - l_{ij}}$$

분모에 있는 $u_{ij} - l_{ij}$는 서로 다른 클러스터에서의 거리에 대해 정규화(normalization)시키는 역할을 한다. 전체 유사 함수 $Sim(x, y)$는 다음과 같이 정의된다.

$$Sim(x, y) = \left[ \sum_{i \in SR[x, y]} w_i (1 - Dist_i)^p \right]^{1/p}$$

여기서 $w_i$는 차원 $i$에 할당된 가중치이고, 본 논문에서 $p = 1$로 할당한다. $SR[x, y]$는 각 차원 $i$에서 같은 클러스터에 속하는 객체들의 집합을 나타낸다.

GB-index 는 다음과 같이 형성한다:

(1) 각 차원 $i$에 대해서, $1 \le i \le d$, $d$는 전체 차원의 수, 전체 데이타베이스 객체를 $K$-means 알고리즘을 통해 $K_i$ 개의 클러스터의 집합으로 분류한다. 차원 $i$에서 $j$번째 클러스터를 $I_{ij}$로 나타내자.

(2) 유사 질의 (즉, $k$-최근접 질의)의 $k$에 대해, 각 클러스터 $I_{ij}$ 마다 검색 범위 $R_{ij}$를 계산하고 $R_{ij}$ 들의 리스트를 유지한다.

(3) 각 클러스터 $I_{ij}$ 마다, 비트맵 $b_{ij}$를 만들고 하한치와 상한치 $[l_{ij}, u_{ij}]$를 유지한다.

(4) GB-index는 생성한 비트맵들의 배열과 각 클러스터에 대한 검색 영역 $R_{ij}$들의 리스트, 그리고 각 검색 영역의 하한치와 상한치 $[l_{ij}, u_{ij}]$ 들의 리스트로 형성된다.

특성 벡터 $q$, 질의 가중치 $w = (w_1, w_2, \cdots, w_d)$, 원하는 객체의 개수 $k$가 주어졌을 때, GB-index 상에서의 유사 질의($k$-최근접 질의)는 다음과 같이 처리된다:

(1) 질의 $q$에 대해, 각 차원 $i$에 대해, $1 \le i \le d$, $q$가 놓이는 클러스터 $I_{ij}$에 대한 범위 $R_{ij}$를 구한다.

(2) $R_{ij}$, $1 \le i \le d$,에 해당하는 클러스터에 대한 비트맵을 형성한다.

(3) 같은 차원에 대해 선택된 비트맵들에 대해 비트 단위의 OR을 수행하여, 각 차원에 대해 하나의 비트맵을 형성한다.

(4) 모든 차원에 대해 선택된 비트맵들에 대해 비트 단위의 AND를 수행하여 최종 비트맵 $b$를 형성한다.

(5) $b$에서 1의 값을 갖는 비트에 해당하는 객체들의 집합 $P$를 구한다.

(6) $P$에 있는 객체에 대해, $q$에 대한 유사도를 계산하고, 가장 유사도가 큰 순으로 $k$ 개의 객체를 반환한다.

## 6. 시스템 구조

현재 우리는 사이버대학 프로젝트의 일환으로 원격 학습 시스템 COVA를 개발 중이며, 그림 6은 COVA의 전체 구조도를 나타낸 것이다. 사용자는 상용 웹 브라우저를 사용하여 COVA가 제공하는 원격 학습 서비스를 받을 수 있다. COVA는 (1) 데이타 모델러, (2) 텍스트 처리와 주석기, (3) 비디오 처리및 주석기, (4) 데이타베이스 구조 요약기, (5) 색인기, (6) 저장 관리기, (7) 질의 처리기, (8) 질의 처리와 브라우징을 통합하는 사용자 인터페이스, (9) 스트리밍 미디어 처리기의 9 개의 주요 요소로 이루어져 있다.

**• 데이타 모델러**

강의 노트와 비디오를 강의 노트로부터 추출한 키워드와 색인어, 그리고 강의에 할당한 속성으로 구성되는 메타데이타와 함께 강의 객체로 저장하고, 전체 강의 데이타베이스를 준구조적 데이타베이스로 구성해서 저장한다.

**• 텍스트 처리와 주석기**

텍스트 처리기는 강의 노트에서 텍스트만 추출한 다음, 검색과 색인에 필요한 색인어를 도출해 낸다. 텍스트 주석기는 사용자가 각 강의 객체에 필요한 속성을 할당할 수 있도록 하는 편집기이다.

**• 비디오 처리 및 주석기**

비디오 처리기는 한 편의 비디오를 비디오 샷 단위로 분할한다. 비디오 주석기는 각 비디오 샷에 속성을 할당할 수 있도록 하는 편집기이다.

**• 데이타베이스 구조 요약기**

데이타베이스 구조 요약기는 준구조적 데이타베이스에서 구조를 추출하여 데이타베이스 스키마를 형성한다. 사용자는 데이타베이스를 브라우징하고 질의를 형성하는데 이 스키마를 이용하고, COVA는 색인과 질의 처리에 이 스키마를 이용한다.

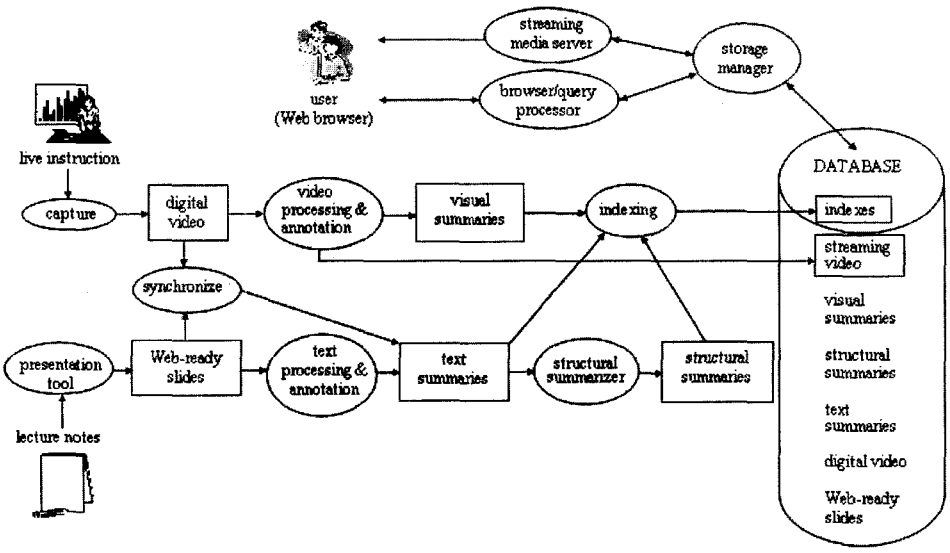


그림 6 원격 학습 시스템 COVA의 구조

**• 색인기**

색인기는 대용량 데이타베이스에서 빠른 검색을 지원하는 색인을 생성, 관리하는 모듈이다. 원격 교육과 학습은 쓰기 보다는 읽기 중심의 응용이며, 멀티미디어 데이타 검색 및 관리와 그래프 형태의 데이타 검색 및 관리 등으로 다양한 형태의 정보를 처리하기 때문에 응용과 데이타 형태에 따른 많은 색인 구조를 필요로 한다. COVA에는 현재 역화일(inverted-file), P-index, GB-index가 개발되어 있다.

**• 질의 처리기**

질의 처리기는 크게 이미지 검색을 위한 질의 처리기와 그래프 탐색을 위한 질의 처리기로 구분된다. 이미지 검색을 위한 질의 처리기는 유사성 질의 처리에 초점을 맞추고, 그래프 탐색 질의 처리기는 아래 방향 그래프 순회와 윗 방향 순회 전략, 그리고 이들을 통합한 복합 탐색 전략에 초점을 맞춘다.

**• 질의 · 브라우징 통합 사용자 인터페이스**

COVA 사용자 인터페이스의 특징 중의 하나는 데이타베이스 브라우징과 선언적인 질의를 통합하는 사용자 인터페이스이다. 웹 학습자는 탐색을 위해 간단한 질의를 통해 검색 공간을 축소한 후에, 얻은 결과에 대해서만 브라우징을 수행할 수 있다. 즉, 학습자는 질의와 브라우징을 섞어가며 원하는 강의 객체를 찾아갈 수 있다.

**• 저장 관리기**

저장 관리기는 강의 데이타베이스 및 색인을 디스크에 저장, 관리하는 책임을 진다. 여기서의 주요 이슈는 준 구조적 모델의 의미를 어떻게 디스크에서 구현하는가 하는 것이다. 대부분의 그래프 기반 데이타 모델에서 객체들은 들어오는 간선의 레이블에 의해 구별된다. COVA의 저장 관리기는 이러한 가정 하에서 동일한 입력 레이블을 갖는 객체의 클러스터링을 구현한다.

**• 스트리밍 미디어 처리기**

스트리밍 미디어 처리기는 비디오를 압축한 비율과 동일하게 인터넷에서 전달하고 사용자로부터의 피드백을 관리하는 책임을 진다.

## 7. 요약 및 결론

인터넷 스트리밍 비디오와 멀티미디어 및 데이타베이스 기술의 발전은 교육과 학습의 형태에 새로운 기회를 제공하고 있으며, 본 논문에서는 XML에 기반한 준구조적 데이타 모델을 사용하여 원격 학습에 대한 새로운 접근법을 제시하였다. 준구조적 모델을 통해 우리는 강의 내용 표현의 유연성과 인터넷을 통한 전달에 용이하도록 하였다. 이 모델에 기초하여 그래프 기반의 강의 데이타베이스에서 스키마를 추출하는 기법을 개발하였고, 효율적인 탐색을 위한 두 가지 색인 기법, P-index와 GB-index를 개발하였다. P-index는 경로에 기반한 색인을 위해 개발되었고, GB-index는 강의 노트 속의 이미지에 의한 강의 객체 검색에 사용된다. 마지막으로 내용 기반 강의 검색을 지원하는 원격 학습 시스템 COVA의 전체 구조를 기술하였다. COVA의 개발에는 많은 요소 기술이 요구되며, 본 논문에서는 그 중 중요한 일부 기술에 대해 제시하였다.

## 참 고 문 헌

[1] Abiteboul, S., "Querying Semistructured Data," *Proc. of ICDT*, pp. 1~18, 1997.
[2] Buneman, P., "Semistructured Data," *Proc. of the ACM PODS*, pp. 117~121, 1997.
[3] Papakonstantinou, Y., Garcia-Molina, H., and Widom, J., "Object Exchange Across Heterogeneous Information Sources," *Proc. of ICDE*, pp. 251~260. 1995.
[4] Goldman, R. and Widom, J., "DataGuides: Enabling Query Formulation and Optimization in Semistructured Databases," *Proc. of VLDB Conf.*, pp. 436~445, 1997.
[5] Buneman, P., Davidson, S., Fernandez, M., and Suciu, D., "Adding Structure to Unstructured Data," *Proc. of ICDT*, 1997.
[6] Henzinger, M., Henzinger, T., and Kopke, P., "Computing simulations on finite and infinite graphs," *Proc. of Symp. on Foundations of Computer Sciences*, pp. 453~462, 1995.
[7] Nestorov, S., Abiteboul, S., and Motwani, R., "Extracting Schema from Semistructured Data," *Proc. of ACM SIGMOD*, pp. 295~306, 1998.
[8] Cha, G.-H., "Bitmap Indexing Method for Complex Similarity Queries with Relevance Feedback," *Proc. of ACM MMDB Workshop*, pp. 55~62, Nov. 2003.
[9] Kim, W., Kim, K.-C., and Dale, A., "Indexing Techniques for Object-Oriented Databases," *Object-Oriented Concepts, Databases, and Applications*, pp. 372~394, ACM Press, 1989.

차 광 호
정보과학회논문지 : 데이타베이스
제 31 권 제 1 호 참조