사용자 중심의 멀티미디어 데이터 검색 방안

정성주, 박희숙, 김성록, 조우현
부경대학교 컴퓨터공학과

Multimedia data search method for User

Sung-Ju Jung, Hee-Sook Park, Seong-Rok Kim, Woo-Hyun Cho
Dept of Computer Engineering, Pukyong National University

요 약

인터넷의 보급으로 사용자는 일반문서에 대한 검색뿐만 아니라 멀티미디어 데이터에 대한 검색도 할 수 있게 되었다. 기존 포탈사이트의 검색은 주로 html 문서위주로 제공되고 있으며, 검색방법은 html 문서의 단어, 구를 이용하는 검색방식을 주로 사용하고 있다. 멀티미디어 데이터에 대한 검색 또한 데이터 제공자(Data provider)가 제시한 검색어구를 바탕으로 이루어진다. 본 논문에서는 사용자(User)에게 관심이 있는 멀티미디어 데이터 부가정보를 인덱스로 유지하고 구성하여 제공하는 XML 트리 형식의 검색 시스템을 제안한다

1. 서론

인터넷의 보급과 사용자 컴퓨터의 증가로 웹상의 문서나 멀티미디어 데이터를 편리하게 이용할 수 있게 되었다. 과거에는 전송 선로의 제한으로 주로 웹문서의 검색이 이루어 졌지만, 현재는 웹문서 뿐만 아니라 이미지, 동영상 등의 멀티미디어에 대한 검색도 증가하였다. 인터넷 상에 문서나 멀티미디어 데이터의 증가로 데이터 검색에 대한 중요성이 부각되었다. 기존 포탈사이트에서는 웹문서에 존재하는 단어나 구를 입력하면 원하는 문서를 검색할 수 있다. 또한 웹 문서를 검색하는 경우 문서내의 단어, 구등을 이용해 검색할 수 있다. 웹문서의 경우 기존에는 html태그를 이용한 문서가 주를 이루었지만, 문서간의 호환과 표현 능력의 한계로 현재 xml 표준이 이용되고 있다. 또한 xml 문서의 경우 xml 태그의 성격과 구성을 조작할 수 가 있어 다양한 표현을 할 수 있고, 다른 형식의 xml 문서로 변환할 수도 있다. 멀티 미디어 데이터는 자체에 단어나 구를 가지지 않기 때문에, 데이터 제공자가 제시한 검색어를 바탕으로 사용자가 검색을 할 수 있다. 그러나 멀티 미디어 데이터의 경우, 데이터 제공자가 제시한 검색 정보만으로는 사용자가 원하는 멀티미디어 데이터를 효과적으로 검색할 수 없다. 데이터 제공자가 제시한 검색정보로는 멀티미디어 데이터의 부가 정보를 모두 표현할 수 없기 때문이다. 또한 멀티미디어 데이터베이스가 원거리에 떨어져 있고 데이터베이스의 종류가 다르다면 통합된 검색을 하기가 어렵게 된다. 따라서 본 논문에서는 사용자가 제시한 멀티미디어 데이터에 대한 부가정보를 바탕으로, xml문서의 트리 구성방식을 적용한 인덱스를 구성하여 사용자의 검색에 활용하는 방안을 제안한다. 본 논문은 다음과 같이 구성된다. 2장에서 기존시스템의 문제점과 새로운 시스템의 검색방법에 대한 제안을 기술하고, 3장에서는 새로운 시스템의 구현방법을 기술하며, 마지막으로 4장에서 결론을 내린다.

2. 기존 시스템의 문제점과 새로운 시스템 제안

2.1 기존 시스템의 문제점

기존 시스템[1]의 경우 멀티미디어 데이터(영화)에 대한 검색을 데이터에 대한 타이틀과 사용자가 데이터베이스를 검색한 히스토리(history)를 이용하여 검색하는 방법을 사용하고 있다. 검색 모델을 title model과 user model로 나누어, user model에는 사용자의 히스토리, title model에는 데이터의 특성에 대한 인덱스가 저장된다. 기존의 방법은 데이터 제공자가

제공한 정보를 기반으로 한 검색만이 가능하기 때문에 데이터 제공자가 제시한 정보만으로는 다양한 사용자의 요구를 만족시키기 위한 충분한 검색을 제공하지 못하게 된다. 예를 들어, 제공자가 제공한 데이터는 영화 동영상의 경우 제목, 장르, 스토리 등으로 제한된다. 사용자가 데이터 내용에 대한 정보를 충분히 가지지 않는다면, 효과적인 검색을 할 수 없고 데이터베이스에 대한 과도한 액세스를 유발할 수도 있다.

2.2 새로운 시스템 제안

데이터 내용에 대한 정보와 부가 정보를 중간 단계에서 관리하여 인덱싱 서비스를 제공한다면 사용자가 원하는 멀티미디어 데이터를 검색하기 위해 직접 데이터 베이스에 접근하지 않아도 되기 때문에 이에 따른 전송 부하를 감소시킬 수 있을 뿐만 아니라 사용자에게 다양한 멀티미디어 데이터 검색 기능을 제공할 수 있으므로 효과적인 검색 서비스를 제공할 수 있다.

xml 문서의 트리 형식과 데이터베이스의 데이터 저장 방법을 이용하여, 멀티미디어 데이터의 부가정보에 대한 인덱스를 구성하여 검색 서비스를 제공한다. 부가 정보는 사용자가 멀티미디어 데이터를 검색하기 위해 직접 입력한 단어나 구의 형태로 제시하는 것을 말하며, 검색을 수행할 때는 검색에 필요한 부가정보를 사용자가 직접 입력하여 이용하거나 데이터 제공자가 제공한 이전의 부가정보를 바탕으로 제공한 정보를 이용하여 검색을 한다. 즉, 다른 사용자들이 그 검색어를 이용할 수 있게 한다. 현 시점의 마지막 사용자가 입력한 부가정보는 다른 사용자의 검색을 위한 정보로 제공되어 진다.

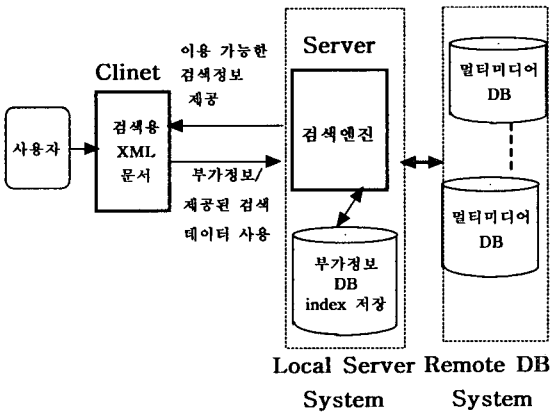

그림 1. 전체 시스템 구성도

전체 시스템의 구성도는 그림1과 같다.

3. 시스템 구현방법

3.1 시스템의 모델 구조

검색 모델의 구조는 xml문서의 트리 표현기법을 이용하여 멀티미디어 데이터의 부가 정보를 저장 및 관리를 한다. 부가 정보를 위한 트리는 Element노드의 경로와 attribute노드, text등으로 이루어진다. 멀티미디어 데이터의 부가정보를 트리로 구성하여 각각의 데이터에 대한 부가정보를 유지한다. 검색 트리의 기본 Element들의 초기 구성은 시스템 구현시 기본적으로 제공되어지며 또한 사용자가 임의의 Element를 만들어서 추가할 수도 있다. 검색트리 모델의 구조는 데이터베이스 데이터에 대한 검색 정보를 트리로 구성하여 그림2와 같은 형태로 제공한다.

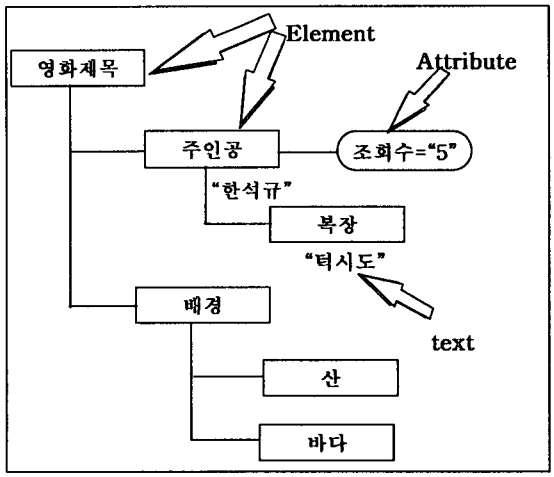

그림 2. 부가정보 트리 구성

위의 그림2에서 Element는 검색분류를 나타내며, text는 각 검색분류에 따른 검색단어 즉, 실제 데이터를 검색하기 위해 사용한다. Attribute는 검색한 수 즉, Element의 우선순위를 표현하기 위해 사용되며 그 Element의 유효성을 표현한다.

위와 같은 부가정보 트리를 구성하여 사용자에게 정보를 제공하게 되면 사용자는 다른 사용자들이 검색에 자주 이용하는 검색어와 검색경로를 제공받게 되므로 보다 효율적인 데이터 검색이 이루어 질 수 있을 것이다. 또한 동일한 구성 형태의 부가정보 트리를 정보제공자에게도 제공한다면 정보 제공자는 이를 바탕으로 자신의 정보가 효율적으로 검색되어 질 수 있도록 검색 시스템에 검색어, 구를 제시할 수 있다.

검색 모델 트리를 이용하여 사용자가 원하는 적절한 데이터를 검색하지 못한 경우에는 멀티미디어 데이터베이스에 직접 접근하여 검색을 하게 된다. 이런 방식으로 데이터를 검색하게 되는 경우에는 이전에 입력한 검색어구들이 부가정보 트리에 추가되어 진다.

검색어구는 경로를 포함하는 이름과 값으로 구성된다. 예를 들어, 주인공의 복장 중 턱시도를 입은 영화를 검색하는 경우에는 '주인공-복장' 이 Element 경로(Path)가 되며 '턱시도' 가 text가 된다.

3.2 부가정보 트리의 구성내용

부가정보 트리 구성에서 한 개의 노드를 구성하는 요소는 다음과 같다

<data_id, element_id, parent_element_id, element, text>

data_id 는 멀티미디어 데이터베이스에 저장되어 있는 데이터의 식별자(identifier)를 의미하며, element_id는 현재 노드의 식별자, parent_element_id는 상위 노드의 식별자를, element는 Element의 이름을, 그리고 text는 Element의 텍스트 값을 의미한다.

각 노드의 조회수, 유효성 등과 같은 노드의 특성은 Attribute 노드로 구성하며, 그 구성은 다음과 같은 요소들을 포함한다.

<element_id, attribute_name, attribute_value>

element_id는 Attribute가 속한 Element의 식별자를 의미하고, attribute_name은 Attribute의 이름을 의미하며, attribute_value는 Attribute의 값(value)을 의미한다.

먼저 자주 사용되는 경로를 포함하는 노드들을 이용하여 검색 트리를 구성한다. 사용자가 경로명과 대응되는 text를 입력하여 검색요청을 하면 검색 엔진은 검색 트리에서 먼저 검색을 한다. 만약 검색을 요청한 경로명과 text가 트리내에 있다면 관련된 정보를 내림차순 정렬하여 사용자에게 보여준다. 검색을 요청한 값이 트리내에 없다면 사용자가 입력한 검색어구를 임시로 저장하고, 데이터베이스에서 데이터 내용에 대한 검색을 직접 수행하게 된다. 데이터베이스에서 관련된 데이터를 찾게 된다면, 사용자가 이전에 입력한 검색어구를 검색 트리에 후보 노드로 등록한다.

3.2 부가정보 트리의 구현

사용자가 검색 트리에 후보 노드를 추가하는 과정은 아래와 같은 알고리즘으로 수행된다.

```
public void addUserTree(int parent_id){
  Vector movie = new Vector();
  Vector attri = new Vector();
  //사용자가 생성한 Element의 속성을 설정
  attri.addElement(new Attribute(0, "validate",
"false"));
  attri.addElement(new Attribute(0, "count", "0"));
  //Element를 생성한다
  Element element = new Element((int)0, parent_id,
(int)0, "movie", "movie_name", attri);
  movie.addElement(element);
}
```

사용자가 경로를 포함하는 노드와 값을 이용한 데이터 검색과정을 수행하는 알고리즘은 다음과 같다.

```
void findMovie(String path, String value){
  Element element = new Element(0,0,0,path, value,
null);
  Vector element_list = new Vector();
  element_list.addElement(element);
  //Element경로와 text로 데이터 검색을 한다
  Vector movie_id =

xml_movie_tree.findMovie(element_list);
}
```

다른 Element보다 조회수가 큰 Element는 다른 Element의 검색 기회를 부여하기 위해서 조회수를 일정비율로 감소시키게 되며 이를 수행하는 알고리즘은 다음과 같다.

```
void decreaseCount(int rate){
  count_value = count_value - count_value*rate/100;
}
```

검색에 이용되지 않는 Element들은 먼저 Attribute를 '쓰레기(trash)'로 설정한다. 속성이 '쓰레기'로 된 노드들이 계속해서 조회수가 적다면 접속 통계를 계산할 때 삭제를 하게 되며 이를 수행하는 알고리즘은 다음과 같다.

```
void setTrash(int element_id, int limit_count){
  if(count_Value < limit_count){
    Attribute att = new Attribute(element_id, "Trash",
"true");
  }
}
```

3.3 검색선호도 제공을 위한 통계

사용자가 많이 검색한 Element의 경로와 단어의 리스트(list)를 구성하여 제공한다. 경로검색 리스트와 단어리스트는 새로운 사용자와 데이터 제공자에게 검색 정보로 제공된다. 사용되지 않는 노드들 즉, Attribute가 쓰레기(trash)인 노드들의 삭제 리스트로 구성한다. 그리고 삭제 리스트의 노드들은 트리에서 제거하기 위해 사용된다. 각 리스트 통계값으로 Attribute노드의 조회수를 사용한다.

4. 결론 및 향후 연구

본 논문에서는 부가정보를 이용한 사용자중심의 멀티미디어 검색 서비스를 제안하였다. 기존의 시스템에서는 제한된 범위의 검색서비스를 제공하게 되지만, 연구된 제안 시스템에서는 사용자의 선호에 따라 동적으로 검색 트리가 재구성 될 수 있으므로, 사용자에게는 보다 효과적인 검색 방법을 제공하고 데이터 제공자에게는 사용자의 선호도에 대한 정보를 제공함으로써 제공자의 데이터가 효과적으로 검색되기 위한 검색 모델을 제시할 수 있다. 향후 과제로는 본 논문에서 제안한 검색 시스템의 효율성을 객관적으로 비교분석할 수 있는 측정 기준이 마련되어야 하며 사용자의 선호도를 사전에 판단 할 수 있는 예측 모델에 대한 연구가 필요하다.

[참고 문헌(References)]

[1] Takashi MITSUISHI, Jun SASAKI and Yutaka FUNYU, A Design of A Kansei Retreival System for Distributed Multi-media Database

[2]Hiroshi Ishikawa, Manabu Ohta, Querying Web Distributed Databases for XML-based E-Businesses: Requriement Analysis, Design, and Implementation

[3] Ian Newman, Applying Experiences of providing effective information identification and retrieval for predominantly text based distributed databaes to multimedia databases

[4] http://www.w3.org/TR/xpath

[5]DEITEL, DEITEL, NIETO, NIN & SADHU, xml how to program

[6]Thomas Kyte, expert one-on-one Oracle