2001년 한국정보처리학회 춘계 학술발표논문집 제8권 제1호

XML문서에서의 다중 스키마 추출에 관한 연구

김성림*, 윤용익
숙명여자대학교 컴퓨터과학과
srkim@cs.sookmyung.ac.kr
yiyoon@sookmyung.ac.kr

The Study on Multi-level Schema Extraction for XML documents

Sungrim Kim, Yong-Ik Yoon
*Dept. of Computer Science, Sookmyung Women's University

요약

XML이 인터넷상에서 데이터를 표현하고 교환하는 새로운 표준으로 등장하고 있다. XML은 미리 정의된 스키마가 없고, 문서 자체에 데이터와 데이터 구조를 갖고 있기 때문에 기존의 관계형 데이터베이스나 객체 지향 데이터베이스에서 사용되는 SQL이나 OQL을 바로 적용하기가 어렵다. 따라서 이러한 XML에 대해 새로운 질의어와 질의 처리를 위한 스키마 추출에 대한 많은 연구가 이루어지고 있다.

본 논문에서는 XML 문서에 대한 스키마 추출 방법과 그래프 프로젝션을 통한 질의 처리 방법을 제안하였다. 여러 단계의 스키마 추출을 가능하게 함으로써 사용자의 질의에 대해 보다 효율적인 질의 결과를 제공해 줄 수 있다.

1. 서론

XML(eXtended Markup Language)는 인터넷상에서 데이터를 표현하고 교환하는 새로운 표준으로 등장하고 있다[BPS98, Bos97]. XML 태그는 데이터 자체를 기술한다. XML의 자기 서술적인 특징(self-describing)을 바탕으로 XML 문서를 여러 형태로 보여줄 수 있고, 내용을 기반으로 데이터를 필터링하거나 어플리케이션의 목적에 맞게 재구성이 가능하다[ABS00, FS00].

XML 문서는 기존의 데이터베이스에서처럼 정해진 스키마를 갖고 있지 않지만 문서마다 구조(Document Type Definition : DTD)를 갖고 있다고 볼 수 있다. XML의 데이터 모델은 구조상 기존의 데이터베이스와 많은 차이점이 있고, 또한 SQL 이나 OQL을 바로 적용하기에 부적합하다. 따라서 이러한 XML 문서들에 대해 스키마를 추출하는 방법과 질의어에 대한 연구가 활발히 진행되고 있다[Levy99, STHZDN99].

본 논문에서는 XML 문서에 대해 스키마를 추출하고, 추출된 스키마를 바탕으로 데이터의 빈도 수에 따라 새로운 여러 단계의 스키마를 추출하는 방법을 제시하고자 한다. 질의 수행결과가 너무 적거나 많을 때 XML 문서에 대해 추출된 여러 단계의 스키마에 적용해서 질의를 수행함으로써 사용자의 요구를 효율적으로 반영할 수 있게 한다. 그리고 사용자 질의에 대해 질의 그래프를 생성하고, 이 질의 그래프를 스키마 그래프에 프로젝션하여 원하는 질의 결과를 효율적으로 얻을 수 있는 방법을 제안하고자 한다.

2. 관련 연구

2.1 트리표현식의 발생 빈도수에 따른 스키마 추출

트리 표현식들의 발생 빈도에 따라 최대의 트리 표현식으로 공통적인 스키마를 추출하는 방법이 있다[WL97,WL98].

트리 표현식에서의 스키마 추출은 다음과 같이 이루어진다. 트리 표현식(tree expression) *te*에서 *te*의 지지도(support : MINISUP)는 도큐먼트 *d* 보다 표현식이 약한 *te*를 갖는 도큐먼트의 개수이다. 사용자가 정의한 최소 지지도 MINISUP에 대해, *te*의 지지도가 크다면, *te*가 빈도수가 높다고(frequent) 볼 수 있다. *te*의 빈도수가 높고, 다른 어떤 트리 표현식보다 빈도수가 높다면, 이러한 *te*는 최대 발생 빈도수(maximally frequent)를 갖는다고 한다.

2.2 발생 빈도 패턴 트리

발생 빈도 패턴을 찾는 방법은 트랜잭션 데이터베이스, 시계열데이터베이스등 많은 데이터베이스 분야에서 연구되어 왔다. 여러 방법 중에서 발생 빈도 패턴 트리(Frequent Pattern Tree : FP-tree)를 구축하여 최대 패턴을 구하는 방법이 제시되었다[HPY00].

트랜잭션 데이터베이스가 있고, 임계치 ξ를 3 이라고 했을 때, FP-tree를 구축하는 순서는 다음과 같다.

단계 1) 데이터베이스를 스캐닝하여 각 아이템의 발생 빈도수를 계산하다.

단계 2) 각 트랜잭션의 발생 빈도 아이템의 집합을 저장한다.

단계 3) 여러 트랜잭션이 발생빈도 아이템 집합을 공유했을 때 이를 하나로 표현하고, 발생빈도수는 더해지면서 계산한다.

단계 4) 두 개의 트랜잭션이 앞부분에 공유하는 부분이 있으면, 하나로 표현하고, 발생 빈도수는 더해지면서 계산한다.

2.3 Lore 시스템

Lore는 스탠포드대학에서 개발한 XML을 위한 데이터베이스 관리 시스템이다[AQMWW97, GMW99, GMW00]. 1995년 처음 개발될 때는 반구조적 데이터를 관리하기 위한 시스템이었다. 이후 XML 개념이 도입되면서 XML을 위한 시스템으로 발전되었다.

Lore는 XML 데이터를 각각의 엘리먼트, 어트리뷰트로 분해한다. Lore 시스템에서 사용되는 질의어는 Lorel (Lore language)라고 하고, Lorel은 SQL과 비슷한 SELECT/FROM/WHERE 의 문법 구조와 XML 그래프를 이동의 표현하는데 필요한 경로 표현식으로 구성되어 있다.

XML이나 Lore 모두 미리 정의된 스키마를 가지고 있지 않은데, 태그나 어트리뷰트 패턴이 없는 경우에 사용자가 의미있는 질의를 만들기는 어렵다. 또한 질의 엔진도 질의를 효율적으로 수행하기 위해서는 데이터베이스의 구조를 어느 정도 이해하고 있을 필요성은 있다. 이러한 기능을 위해 Lore에서는 DataGuide를 제공한다.

DataGuide는 XML 데이터베이스에 대해 정확하고, 동적으로 정리된 구조를 표현해줌으로써 데이터베이스 스키마나 DTD 역할을 수행하게 된다. 사용자는 DataGuide를 통해 데이터베이스의 전체적인 구조를 파악하여 질의를 만들 수 있게 된다.

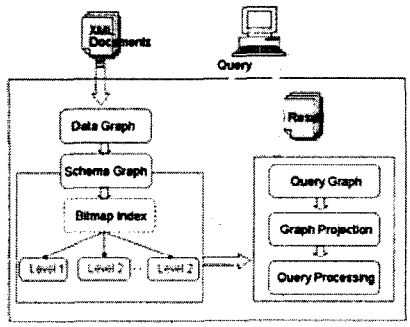
3. 스키마 추출

본 논문에서 제안하는 시스템의 전체적인 구조는 다음과 같다. XML 문서에 대해 모든 데이터가 표현될 수 있는 '데이터 그래프'를 생성한다. 그리고 깊이 우선 탐색 기법을 바탕으로 데이터에 있는 모든 경로가 단 한번만 표현되는 '스키마 그래프'를 생성한다.

스키마 그래프에서 XML 문서에 대한 모든 레이블이 존재한다. 스키마 그래프와 XML 문서를 바탕으로 루트 노드와 리프 노드로의 레이블 경로에 대한 비트맵 인덱스를 구축한다. 비트맵 인덱스는 레이블 경로가 존재하는 경우는 1의 값을, 존재하지 않는 경우는 0의 값을 갖는다. 비트맵 인덱스에서 레이블의 발생 빈도수에 따라 그 임계치를 다르게 함으로써 여러 단계의 스키마를 생성할 수가 있다.

사용자 질의에 대해 질의 그래프를 작성하고, 이를 스키마 그래프와 그래프 프로젝션을 수행함으로써 질의에 맞는 레이블 경로들을 가지고 있는 XML 문서들을 찾을 수가 있고, 이러한 문서들에 대해 Quilt XML 질의 언어로 변환 후 질의 처리를 한다.

전체적인 시스템 구성도는 다음과 같다.



3.1 예제 XML 문서와 DTD

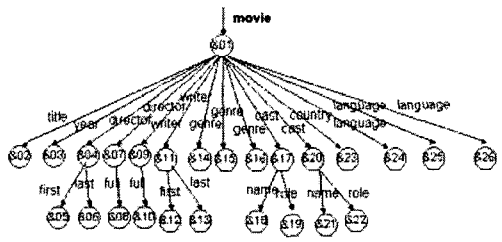
본 논문에서 사용하는 DTD와 일부 XML 문서는 다음과 같다.

```
<!ELEMENT movie  (title, year, director+, writer+,
                  genre+, cast+, language*, country* ) >
<!ELEMENT title (#PCDATA) >
<!ELEMENT year (#PCDATA) >
<!ELEMENT director ((lastname, firstname) | fullname)>
<!ELEMENT writer ((lastname, firstname) | fullname)>
<!ELEMENT lastname (#PCDATA) >
<!ELEMENT firstname (#PCDATA) >
<!ELEMENT fullname (#PCDATA) >
<!ELEMENT genre (#PCDATA) >
<!ELEMENT cast (name,role) >
<!ELEMENT name (#PCDATA) >
<!ELEMENT role (#PCDATA) >
<!ELEMENT language (#PCDATA) >
<!ELEMENT country (#PCDATA) >
```

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE MOVIE SYSTEM "movie.dtd">
<movie><title>SixthSense</title><year>1999</year>
    <director>M.Night Shyamalan</director>
    <writer>M.Night Shyamalan</writer>
<genre>Thriller</genre><genre>Drama</genre>
<genre>Horror</genre>
<cast><name>Bruce                Willis</name><role>Malcolm
Crowe</role><name>Haley  Joel  Osment</name><role>Cole
S e a r < / r o l e > < / c a s t >
<language>English</language><language>Spanish</language>
<language>Latin</language></movie>
```

3.2 데이터 그래프

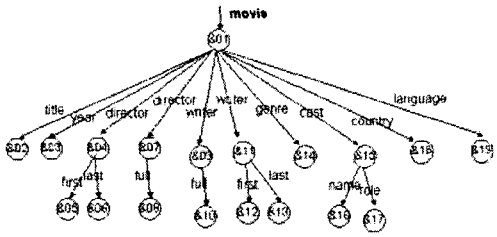
DTD(예:movie.dtd)와 그를 바탕으로 하는 XML문서에 대해 labeled directed graph를 그리면 다음과 같다. 본 논문에서는 XML 문서의 모든 데이터가 표현되는 edge labeled directed graph를 '*데이터 그래프(Data Graph)*'라고 정의한다.



데이터 그래프

3.3 스키마 그래프

본 논문에서는 데이터 그래프에서 깊이 우선 탐색 기법을 바탕으로 모든 경로가 단 한번만 표현될 수 있도록 만들어진 그래프를 '*스키마 그래프(Schema Graph)*'라고 정의한다. 이는 Lore 시스템의 DataGuide처럼 모든 레이블 경로가 유일하고(concise), XML 문서에 있는 모든 데이터는 표현이 되어야 하고(accuracy), 각 노드의 구성이 어떻게 되어 있는지(convenience) 알 수 있도록 한다.



스키마 그래프

3.4 비트맵 인덱싱을 이용한 스키마 추출

3.4.1 비트맵 인덱싱

비트맵 인덱스 기법을 스키마 그래프에서 레이블 경로에 적용하였다[YK00]. 루트 노드에서 리프 노드까지의 레이블 경로를 비트맵 인덱스로 표현한다. 각 XML 문서에 대해 레이블 경로가 존재하면 1의 값을, 존재하지 않으면 0의 값을 갖는다. 예제 스키마 그래프에 대해 레이블 경로와 비트 벡터는 표는 다음과 같다. *pi*는 레이블 경로이고, *di*는 XML 문서이다.

	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13
d1	1	1	0	0	1	0	0	1	1	1	1	0	1
d2	1	1	0	0	1	0	0	1	1	1	1	1	1
d3	1	1	1	1	0	0	0	1	1	1	1	0	0
d4	1	1	1	1	0	1	1	0	1	1	1	1	1

3.4.2 스키마 추출

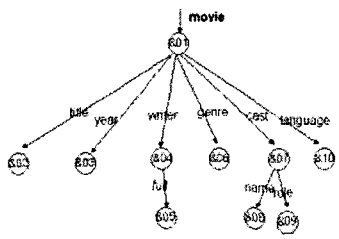
레이블 경로에 대한 비트맵 인덱스를 구한 결과는 다음과 같다. 모든 문서에 대해 p1은 1111, p2는 1111, p3는 0011등의 값을 갖는다. 이러한 모든 XML 문서에 대한 비트맵 인덱스에 대해 Bitwise-OR를 하게 되면 모든 정보를 포함하게 되는 스키마를 얻게 되고, 그 결과는 스키마 그래프와 동일하다.

	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13
d1	1	1	0	0	1	0	0	1	1	1	1	0	1
d2	1	1	0	0	1	0	0	1	1	1	1	1	1
d3	1	1	1	1	0	0	0	1	1	1	1	0	0
d4	1	1	1	1	0	1	1	0	1	1	1	1	1
Bitwise-OR													
	1	1	1	1	1	1	1	1	1	1	1	1	1

비트맵 인덱싱을 이용하여 생성된 스키마 그래프를 바탕으로 각 레이블 경로의 빈도수를 이용하여 여러 단계의 스키마 그래프를 생성할 수 있다.

이렇게 계산되어진 레이블 경로의 빈도수를 임계치로 하여 여러 단계의 스키마를 추출할 수가 있다. 예를 들어 빈도수가 3 이상인 경로에 대해서만 스키마를 추출한다면 빈도수가 3보다 작은 경로 p3, p4, p5, p6, p7, p12 는 새로 생성되는 스키마에서 제외되고 나머지 경로들에 대해서만 다음과 같은 새로운 스키마가 생성된다.

	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13
d1	1	1	0	0	1	0	0	1	1	1	1	0	1
d2	1	1	0	0	1	0	0	1	1	1	1	1	1
d3	1	1	1	1	0	0	0	1	1	1	1	0	0
d4	1	1	1	1	0	1	1	0	1	1	1	1	1
Frequency													
F_i	4	4	2	2	2	1	1	3	4	4	4	2	3

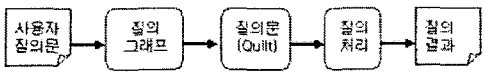


빈도수 > 3 인 스키마 그래프

4. 그래프 프로젝션을 이용한 질의 처리

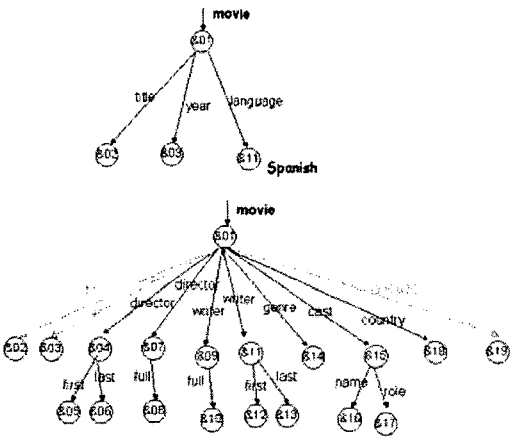
XML 문서에 대한 질의 처리를 다음과 같은 순서에 따

라 실행된다. 사용자로부터 입력받는 질의에 대해 질의 그래프를 작성하고, 질의 그래프에 대한 비트맵 인덱스와 모든 XML 문서들에 대해 bitwise-AND 연산을 실행(그래프 프로젝션)하여 비트맵 인덱스를 갖는 문서들을 찾고, 이를 XML 질의어인 Quilt로 변환하여 질의를 수행하게 된다.



4.1 질의 그래프

여러 단계의 스키마에 대해 사용자는 질의를 실행할 수 있다. 예를 들어, 빈도수가 1인 스키마 그래프를 이용하여, "스페인어로 되어 있는 영화의 제목과 제작년도를 알고자" 하는 질의문을 수행한다고 했을 때 다음과 같은 질의 그래프가 생성된다.



스키마 그래프와 질의 그래프의 프로젝션

4.2 질의 처리

질의 그래프에 대해 XML 질의 언어인 Quilt 로 변환하면 다음과 같다.

```
<spanish-version>
    FOR $m IN document("d1.xml")//movie
    WHERE $m/language="Spanish"
    RETURN $m/title, $m/year
</spanish-version>
```

FOR 절에서 movie가 루트 노드인 d1.xml 문서에 대해 변수 m을 생성하고, WHERE 절에서 $m의 하위노드인 language의 값이 spanish인 조건절을 표현하고, RETURN 절에서 그래프의 노드는 movie가 루트노드이면서 하위노드가 title 과 year인 새로운 그래프를 생성한다.

XML 문서 d1, d2, d4 에 대해 각각 질의문이 수행되고, 다음과 같은 결과를 생성한다.

```
<movie><title>Sixth Sense</title>
<year>1999</year></movie>
```

5. 결론

XML이 인터넷상에서 데이터를 표현하고 교환하는 새로운 표준으로 등장하고 있다. XML은 미리 정의된 스키마가 없고, 문서 자체에 데이터와 데이터 구조를 갖고 있기 때문에 기존의 관계형 데이터베이스나 객체 지향 데이터베이스에서 사용되는 SQL이나 OQL을 바로 적용하기가 어렵다. 따라서 이러한 XML에 대해 새로운 질의어와 질의 처리를 위한 스키마 추출에 대한 많은 연구가 이루어지고 있다.

본 논문에서는 XML 문서에 대한 스키마 추출 방법과 그래프 프로젝션을 통한 질의 처리 방법을 제안하였다. 여러 단계의 스키마 추출을 가능하게 함으로써 사용자의 질의에 대해 보다 효율적인 질의 결과를 제공해 줄 수 있다.

참고문헌

[ABS00] S. Abiteboul, P. Buneman, Dan Suciu, "Data on the Web : From Relations to Semistructured Data and XML", Morgan Kaufmann, 2000

[AQMWW97] S. Abiteboul, D. Quass, J. McHugh, J. Widom and J. Wiener, "The Lorel Query Language for Semistructred Data", International Journal of Digital Libraries, 1(1):66-88, 1997

[Bos97] Jon Bosak, "XML, Java, and the Future of the Web",http://webreview.com/wr/pub/97/12/19/xml/index.html

[BPS98] Tim Bray, Jean Paoli , C. M. Sperberg-McQueen , "Extensible Markup Language (XML)1.0", http://www.w3.org/TR/REC-xml#dt-xml-doc, 1998

[FS00] Daniela Florescu, Jerome Simeon, "XML Data : From Research to Standards", VLDB 2000 Tutorial, Cairo Egypt

[GMW00] Roy Goldman, Jason McHugh, Jennifer Widom, "Lore: A Database Management System for XML", Dr. Dobb's Journal, 25(4):76-80. April 2000, http://www.ddj.com/articles/2000/0004/0004i/0004i.htm?topic=xml

[HPY00] Jiawei Han, Jian Pei, Yiwen Yin, "Mining Frequent Patterns without Candidate Generation", Proceedings of the 2000 ACM SIGMOD on Management of data, 2000, pp. 1-12

[Levy99] Alon Levy, "More on Data Management for XML", University of Washington, May 9th, 1999. http://www.cs.washington.edu/homes/alon/widom-response.html

[STHZDN99] J. Shanmugasundaran, K.Tufte etc., "Relational Databases for Querying XML Documents: Limitations and Opportunities", Proceedings of the 25th VLDB Conference 1999

[YK00] J. Yoon, S. Kim, "Schema Extraction for Multimedia XML Document Retrieval", in Proc. of International Database Symposium on Mobile, XML and Post-Relational Databases, Hong Kong, June 2000