접미어 트리 구조를 이용한 효율적인 XML 경로 인덱싱

이덕형[0], 원정임, 노관준, 윤지희
한림대학교 컴퓨터공학과
{hanuhm, jiwon, pillar, jhyoon}@hallym.ac.kr

A Suffix Tree Approach for Efficient XML Path Indexing

DeokHyung Lee[0], JungIm Won, KwanJoon Roh, JeeHee Yoon
Dept. of Computer Engineering, Hallym University

요 약

최근 인터넷 상에서 XML 문서의 사용이 급속도로 보편화, 일반화됨에 따라 정보 검색을 위한 다양한 XML 질의 언어가 제안되고 있다. XML 질의의 공통 특징으로서 '*' 문자 등을 사용한 정규화 경로식(regular path expression)에 의한 손쉬운 구조정보 검색 기능을 들 수 있다. 본 논문에서는 접미어 트리(suffix tree)를 이용한 새로운 경로 인덱싱 기법을 제안한다. 제안하는 기법에서는 XML 문서상의 각 경로를 축약된 유일한 문자열로 인코딩하며, 인코딩 된 각 문자열의 모든 접미어 정보를 인덱스에 저장한다. 본 기법은 일반 정규화 경로식을 포함하는 구조질의를 매우 효율적으로 처리하며, 또한 경로 정보가 부정확하게 기술된 경우에도 근사 질의 처리를 효과적으로 처리할 수 있다.

1. 서론

XML은 인터넷 상의 데이터 표현 및 교환의 새로운 표준으로 인식되어 그 사용이 급속도로 증가하고 있다. 최근 XML 문서를 효과적으로 저장, 검색하기 위한 기법에 대한 연구가 활발히 진행되고 있으며, 다양한 XML 질의언어[1, 2, 3]가 제안되고 있다. 일반적인 XML 질의는 문서의 구조나 태그 정보 등을 기반으로 표현되므로, 질의 기술을 위해서는 문서의 정확한 스키마 정보에 관한 지식이 요구된다. 따라서 대부분의 질의 언어에서는 '*' 문자 등을 사용한 정규화 경로식(regular path expression)[4]을 허용하여 정확한 경로식을 직접 기술하여야 하는 질의 기술의 부담을 덜 수 있도록 지원한다.

본 논문에서는 정규화 경로식 등을 포함한 사용자의 ad-hoc 질의[5]를 보다 효율적으로 처리할 수 있는 XML 문서의 인덱싱 기법과 질의 처리 기법을 제안한다. 제안하는 인덱스는 접미어 트리[6] 구조를 가지며, XML 문서상의 각 경로를 축약된 유일한 문자열로 인코딩 하여, 인코딩 된 각 문자열의 모든 접미어 정보를 인덱스에 저장한다. 정규화 경로식 등의 질의 처리 시, 본 인덱싱 구조는 질의 경로의 길이가 길거나, 부분적으로 생략, 오류가 있는 경우에도 중간 조인 등을 수행하지 않고 직접 필요한 정보를 검색하도록 지원한다. 본 연구의 기여점은 다음과 같다.

(1) XML 문서상의 모든 경로 정보와 각 경로의 모든 접미어를 인덱스에 추가시켜, 정규화 경로 질의를 효율적으로 처리할 수 있다.

(2) 제안된 인덱스 구조는 심볼 인덱스, 경로 인덱스, 값 테이블로 구성되어, 구조 기반 질의와 내용 기반 질의, 혼용 질의를 모두 효율적으로 처리한다.

(3) 사용자 질의의 경로 기술에 오류가 있는 경우, 본 인덱스 구조를 사용하여 문자열로 표현된 각 경로 사이의 최소 에디팅 거리를 이용한 근사 질의처리가 가능하다.

본 연구는 정보통신부의 정보통신 기초 기술연구 지원사업(정보통신연구진흥원)(과제번호 : C1-2002-146-0-3)으로 수행한 연구 결과임.

논문 구성은 다음과 같다. 2장에서는 기존의 인덱싱 기법과 본 연구의 차별성에 대하여 기술한다. 3장에서는 본 논문에서 제안하는 인덱싱 기법을 소개하고, 이를 이용한 질의 처리 기법의 효율성에 대하여 설명한다. 4장에서는 결론 및 향후 연구 과제에 대하여 기술한다.

2. 관련 연구

기존의 XML 문서의 단순 트리 운행 방식에 의하여 정규화 경로 질의 등을 처리하기 위해서는 문서의 상당 부분을 검색하여야 하므로 실행 비용이 매우 높아지게 된다[4, 5].

정규화 경로 질의를 효율적으로 처리하기 위한 많은 인덱싱 기법들이 제안되어 있으며, 대표적인 기법으로서 Lorel[7], T-index[8] 등과 최근에 발표된 XISS[4], Index Fabric[9] 등을 들 수 있다.

XISS[4]에서는 확장적인 노드 순서화 기법(numbering scheme)을 기반으로 하는 새로운 인덱스 구조(엘리먼트 인덱스, 애트리뷰트 인덱스, 구조 인덱스)를 제안하고 정규화 경로 질의를 효율적으로 수행하기 위한 경로 조인 알고리즘(EE-Join, EA-Join, KC-Join)을 제안하고 있다.

Index Fabric[9]에서는 질의에 출현할 수 있는 경로식(raw path, refined path)을 우선 키워드로 추출하고 이 들 경로식을 문자열로 인코딩 한 후, 패트리시아 트리(patricia tree)[10]를 사용하여 이 들을 디스크 기반의 압축 형태로 구현하고 있다.

본 논문에서 제안하는 인덱싱 기법에서 경로 정보를 문자로 인코딩 하여 인덱스를 구성하는 점에 있어 Index Fabric과 공통점이 있으나, 다음과 같은 차이점을 보인다.

Index Fabric은 문서의 예상 질의 형태(refined path)를 키워드로 추출하여 인덱싱 대상으로 하는 경우, 좋은 성능을 보장할 수 있으나, 모든 일반 정규화 경로질의에 대한 성능 보장에 문제가 있다. 또한 부모 노드나 자식 노드를 구하는 일반적인 문서의 구조적인 정보 질의를 위하여는 별도의 인덱스가 필요하며, 경로식을 문자로 인코딩 하여 트리에 삽입하는 경우, 중첩, 생략되는 문자가 거의 없기 때문에 문제의 성격상 패트리시아 트리의 정보 압축 효과가 적다.

3. 인덱싱 기법

XML 질의는 크게 내용 기반 질의, 구조 기반 질의, 이 들의 혼용 질의로 구별된다. 이 들 질의를 효율적으로 처리하기 위한 시스템 구성 및 인덱스 구조를 그림 1에 보인다. XML 문서로부터 구조/내용 정보가 추출되어 인덱스 구조(심볼 인덱스, 값 테이블, 경로 인덱스)로 표현되며, 인덱스는 원시 문서로의 오프셋 정보를 포함할 수 있다.

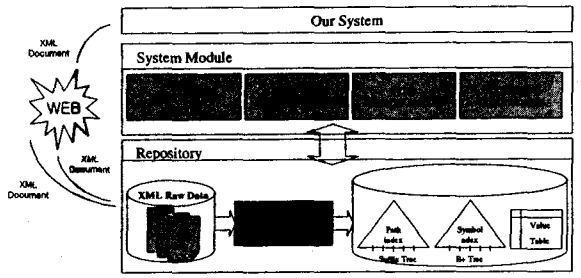

그림 1. System 구성

3.1 인덱스 구조

심볼 인덱스는 모든 식별 가능한 엘리먼트명과 애트리뷰트명을 유일한 심볼로 변환하여 주는 인덱스 구조로서 심볼로서 2바이트 문자를 사용한다. 2바이트 문자의 각 바이트에 ASCII 코드의 33번부터 126번까지의 94개의 문자를 사용하면, 총 8836개의 심볼을 생성할 수 있으며, 이는 일반적인 규모의 XML 문서의 모든 엘리먼트명과 애트리뷰트명을 구별 표현하기에 충분하다[11]. 그림 2에 XML 문서(DBLP 문서)의 예제를 보인다. 다음 그림 3은 그림 2의 문서에서 출현한 모든 엘리먼트명과 애트리뷰트명을 심볼로 변환한 예를 보인다. 값 테이블은 내용 정보를 효율적으로 검색하기 위한 인덱스 구조이다.

DOC1:
```
<dblp>
 <inproceedings key="conf/vldb/Novaretti01">
   <author>Serge Novaretti</author>
   <title>French government activity in the conservation
        of data and electronic documents.</title>
   <pages>623-624</pages>
   <ee>http://www.vldb.org/conf/2001/P621.pdf</ee>
   <year>2001</year>
   <crossref>conf/vldb/2001</crossref>
   <booktitle>VLDB</booktitle>
   <url>db/conf/vldb/vldb2001.html#Novaretti01</url>
 </inproceedings>
</dblp>
```

DOC2:
```
<dblp>
  <book key="books/acm/Kim95">
    <editor>Won Kim</editor>
    <title>Modern Database Systems: The
    Object Model, Interoperability, and Beyond.
    </title>
    <publisher>ACM Press and
        Addison-Wesley</publisher>
    <year>1995</year>
    <isbn>0-201-59098-0</isbn>
    <url>db/books/collections/kim95.html</url>
  </book>
</dblp>
```

그림 2. 예제 XML 문서

<inproceedings>	IN	<cdrom>	CD
<dblp>	DB	<publisher>	PU
<editor>	ED	<url>	UR
<crossref>	CR	<ee>	EE
<book>	BK	<isbn>	IS
<pages>	PA	<title>	TI
<booktitle>	BT	<year>	YE
<author>	AU	key (attribute)	KY

그림 3. 심볼 변화 예

경로 인덱스는 문서에 출현한 모든 경로 정보를 트리 구조로 저장한 인덱스 구조로서 생성 방법은 다음과 같다.

(1) 문서상의 모든 경로 정보를 추출한 후, 이들 각각의 경로에 출현하는 엘리먼트명, 애트리뷰트명을 심볼로 인코딩 하여 (심볼 인덱스를 이용) 경로를 하나의 유일한 문자열로 변환한다.

(2) 문자열로 변환된 각 경로의 모든 접미어를 추출한다.

(3) (2)에 의하여 얻어진 모든 접미어를 트리에 삽입하여 접미어 트리를 생성한다.

(4) 접미어 트리의 단말 노드에는 문서의 ID 정보 및 오프셋 정보 등이 포함된다.

(5) 접미어 트리의 루트 노드에 직접 연결된 단일 엘리먼트, 단일 애트리뷰트 노드에는 부모 노드의 정보를 삽입한다.

그림 4에 이와 같이 생성되는 심볼 인덱스와 경로 인덱스의 구조를 보인다.

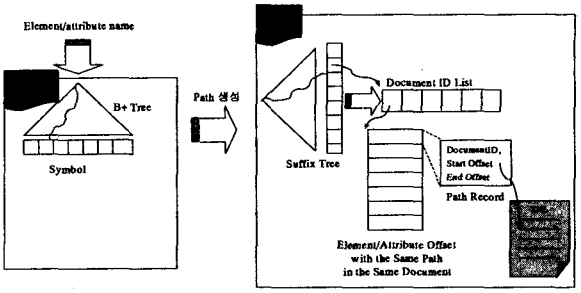

그림 4. 심볼 인덱스와 경로 인덱스의 구조

문서 1		문서 2	
<dblp><inproceedings key>	DB IN KY	<dblp><book key>	DB BK KY
<dblp><inproceedings><author>	DB IN AU	<dblp><book><editor>	DB BK ED
<dblp><inproceedings><title>	DB IN TI	<dblp><book><title>	DB BK TI
<dblp><inproceedings><pages>	DB IN PA	<dblp><book><publisher>	DB BK PU
<dblp><inproceedings><ee>	DB IN EE	<dblp><book><year>	DB BK YE
<dblp><inproceedings><year>	DB IN YE	<dblp><book><isbn>	DB BK IS
<dblp><inproceedings><crossref>	DB IN CR	<dblp><book><url>	DB BK UR
<dblp><inproceedings><booktitle>	DB IN BT		
<dblp><inproceedings><url>	DB IN UR		

그림 5. 심볼로 변환된 경로의 예

그림 5는 그림 2의 예제 문서로부터 추출된 경로를 문자열로 변환한 예를 나타내며, 그림 6은 이 들로부터 추출된 모든 접미어 정보를 나타낸다. 이와 같은 과정에 의하여 생성된 예제 문서에 대한 접미어 트리 구조의 경로 인덱스를 그림 7에 보인다.

문서 1		문서 2	
DB IN KY	DB IN YE	DB BK KY	DB BK IS
IN KY	IN YE	BK KY	BK IS
KY	YE	KY	IS
DB IN AU	DB IN CR	DB BK ED	DB BK UR
IN AU	IN CR	BK ED	BK UR
AU	CR	ED	UR
DB IN TI	DB IN BT	DB BK TI	
IN TI	IN BT	BK TI	
TI	BT	TI	
DB IN PA	DB IN UR	DB BK PU	
IN PA	IN UR	BK PU	
PA	UR	PU	
DB IN EE		DB BK YE	
IN EE		BK YE	
EE		YE	

그림 6. 경로의 접미어 정보 추출 예

3.2 질의처리

제안된 인덱스 기법을 사용한 질의 처리 방식을 설명하면 다음과 같다. 질의로는 그림 2의 XML 문서에 대한 예를 사용하며, 질의 평가를 위하여 그림 7의 경로 인덱스를 사용한다고 가정한다.

3.2.1 정규화 경로 질의

(1) 단일 엘리먼트 혹은 단일 애트리뷰트 검색 (Q1 : title 혹은 @key="books/acm/Kim95" 검색) : 단일 엘리먼트와 애트리뷰트는 접미어 트리의 루트노드에 직접 연결되어 있다. Q1의 경우, "TI"와 "KY"의 노드 정보를 이용하여 질의를 평가한다.

(2) 엘리먼트명과 애트리뷰트명이 연속적으로 출현하는 경로식의 검색 (Q2 : /dblp/inproceedings/author 혹은 /dblp/inproceedings[@key ="conf/vldb/Novaretti01"의 검색) : 접미어 트리 상에는 문서에 출현하는 모든 경로(완전경로, 부분경로)가 인코딩 되어 있으므로 이와 같은 질의의 경우, 직접 검색

이 가능하며 추가적인 조인 연산 등이 필요 없다.

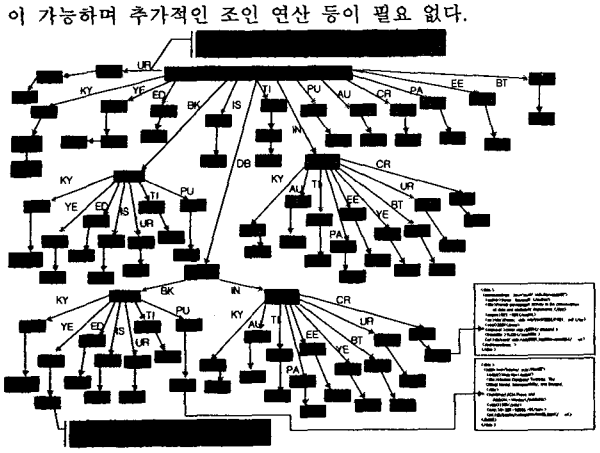

그림 7. 경로 인덱스 구성 예

(3) 와일드 카드(*)가 정규화 경로식에 포함된 경우 (Q3 : /dblp/*/title) : Q3처럼 와일드 카드(*)가 정규화 경로식에 포함된 경우, 접미어 트리에서 에지 정보가 "DB"인 노드를 찾아 그 노드의 모든 자손 노드 중 에지가 "TI"인 노드까지의 모든 경로를 검색한다.

(4) 임의의 경로 다음에 오는 특정 엘리먼트를 찾는 정규화 경로식의 경우 (Q4 : //book/isbn) : Q4처럼 임의의 경로 다음에 오는 특정 엘리먼트를 찾는 정규화 경로식의 경우, 접미어 트리에서 "BK"로 시작되는 접미어 에지를 찾아 다음 경로를 검색하면 된다.

(5) 합연산(union), Kleene closure 연산 등을 포함하는 정규화 경로식의 경우 (Q5 : (e1/e2)*/e3/((e4[@a=v])|(e5/*/e6)))[4] : 두 개 이상의 경로의 합연산은 각각의 경로식을 접미어 트리에서 검색한 후 이를 합연산하여 결과를 얻는다. 또한 (e1/e2)* 형태의 검색은 접미어 트리에 삽입되어 있는 경로 중에서 직접 검색이 가능하다.

3.2.2 조상 / 자손 검색

(1) 조상(부모)노드 검색 (Q6 : <isbn> 의 조상 노드 검색) : 접미어 트리에는 모든 단일 엘리먼트와 애트리뷰트에 대하여 조상 노드정보가 삽입되어 있으므로 이를 이용한다. Q6의 경우, 루트노드에서 "IS"로 시작하는 단일 엘리먼트를 찾아 저장된 부모 노드의 정보를 참조하여 부모 노드를 찾을 수 있으며, 재귀적으로 부모 노드의 부모 노드 정보를 참조함으로써 조상 노드를 쉽게 찾을 수 있다.

(2) 자손(아들)노드 검색 (Q7 : <isbn> 의 자손 노드 검색) : 접미어 트리 상에서 해당 엘리먼트를 나타내는 노드를 검색한 후 그 하위 노드를 검색하여 아들과 자손 노드를 검색할 수 있다. Q7의 경우, 접미어 트리의 루트 노드에서 "IS"로 시작되는 노드의 모든 하위 노드들을 순회하면서 간단히 자손 노드를 검색 할 수 있다.

3.2.3 내용검색

내용 질의의 경우, 값 테이블을 주로 참조하게 된다. 예를 들어 (Q8 : "Serge Novaretti"를 포함하는 문서 혹은 엘리먼트를 검색하라)와 같은 질의가 주어지면, 값 테이블에서 내용을 검색하여 해당 문서의 엘리먼트를 찾아 응답할 수 있다.

3.2.4 근사 질의

질의의 경로 기술에 오류가 있는 경우, 근사질의 평가를 수행하여, 사용자에게 유사한 결과의 집합을 제공하거나, 유사 경로의 집합을 제시하여 사용자의 질의 기술 환경을 지원한다. 예를 들어 (Q9 : /dblp/inproceedings/publisher = "ACM Press and Addison-Wesley")의 질의가 주어 졌을 경우, 질의 처리기는 경로식을 "DB IN PU"로 인코딩 하여 경로식을 평가한다. 단, 그림 7에서와 같이, "DB IN PU"라는 경로는 존재하지 않는다. 이 경우 질의 처리기는 질의 경로와 유사허용치가 한계값(관리자 혹은 사용자 지정) 이하인 유사한 경로들의 후보들을 제시하고, 가능한 결과를 출력한다. Q9의 경우, 질의 처리기는 "DB BK PU"의 가능한 후보를 보여주고, 이 후보의 결과인 "DB BK PU = ACM Press and Addison-Wesley"를 질의 대상으로 변경할 수 있다.

4. 결론

본 논문에서는 XML 문서의 구조와 내용에 대한 효율적인 검색을 지원하는 인덱스 기법을 제안하였다. 문서의 모든 경로 정보를 문자열로 표현하여, 각 문자열의 접미어를 트리에 삽입한 접미어 트리 구조를 인덱스로 사용하였다. XISS[4]에서는 정규화 경로식이 길어지는 경우, 분할된 단순 경로식에 대하여 각각 조인 연산을 수행하여야 한다. 또한 Index Fabric[9]에서는 문서에 대한 일반적인 구조질의(조상/자손 검색)에 대해 별도의 인덱스가 요구되며, 일반화 된 정규화 경로 질의 평가에 제약점이 있다. 본 논문에서 제안한 경로 인덱싱 방법은 일반적인 정규화 경로식, 조상/자손 검색, 내용 기반 질의 등 다양한 질의를 동시에 효과적으로 처리할 수 있고, 부정확한 경로 질의에 대하여도 최소 에디팅 거리를 이용한 근사 질의 처리가 가능하다.

향후 연구 과제로는 다양한 실제의 XML 문서에 대하여 본 논문에서 제안한 인덱스 기법과 기존의 인덱스 기법들과의 실험을 통한 성능 평가를 수행하는 것이다.

참고문헌

[1] Jonathan Robie, Joe Lapp, David Schach, "XML Query Language (XQL)", http://www.w3.org/TandS/QL/QL98/pp/xql.html

[2] A. Deutsch, M. Fernandez, D. Florescu, A. Levy, D. Suciu. "XML-QL: A Query Language for XML", http://www.w3.org/TR/1998/NOTE-xml-ql-19980819/

[3] Don Chamberlin, Daniela Florescu, Jonathan Robie, Jrme Simon, Mugur Stefanescu, "XQuery : A Query Language for XML W3C working draft." Technical Report WD-xquery-20010215, World Wide Web Consortium, February 2001.

[4] Quanzhong Li, Bongki Moon, "Indexing and Querying XML Data for Regulra Path Expressions", Preoceedings of the 27th VLDB Conference, Roma, Italy, 2001

[5] Albrecht Schmidt, Martin Kersten, Menzo Windhouwer, "Querying XML Documents made Easy : Nearest Concept Queries", 17th International Conference on Data Engineering, 2001.

[6] G. A. Stephen, String Searching Algorithms, World Scientific Publishing, 1994.

[7] S. Abiteboul, D. Quas, J. McHugh, J.Widom, J.Widom, and J.L. Wiener., "The Lore query language for semistructured data" International Journal on Digital Libraries, 1(1):68-88, April 1997.

[8] T. Milo and D. Suciu, "Index Structures for Path Expressions" , Intl. Conf. on Database Theory, 1997.

[9] Brian F. Cooper, Neal Sample, Michael J. Franklin, Gísli R. Hjaltason, Moshe Shadmon, "A Fast index for Semistructured Data", Preoceedings of the 27th VLDB Conference, Roma, Italy, 2001

[10] Donald Knuth. The Art of Computer Programming, Vol. III, Sorting and Searching, Third Edition. Addison Wesley, Reading, MA, 1998.

[11] 강형일, 박종관, 유재수 , "XML 문서를 위한 효율적인 인덱싱 모델", 컴퓨터 정보통신 연구회 학술지, Vol.8, No. 2, 2000.