# 비트맵 인덱싱 기반 유사한 XML 문서 검색 기법

이재민, 황병연
가톨릭대학교 컴퓨터공학과
e-mail : {likedawn, byhwang}@catholic.ac.kr

# A Search Method of Similar XML Documents based on Bitmap Indexing

Jae-Min Lee, Byung-Yeon Hwang
Dept. of Computer Engineering, The Catholic University of Korea

**요 약**

XML 검색을 위한 기존의 비트맵 인덱싱은 XML을 문서, 경로, 단어로 구성된 3차원 비트맵 인덱스에 매핑하고 이를 이용해 정보를 추출함으로써 뛰어난 성능을 입증하였다. 그러나 이것은 유사 문서를 수집하고 검색하기 위해 경로 전체를 인덱스 구성의 단위로 사용함으로써 유사 경로를 인식하지 못하는 문제를 초래하였으며 유사 경로를 인식하지 못함으로 인해 유사 문서 검색의 치명적인 성능 저하가 발생하게 되었다. 이에 따라 본 논문에서는 기존의 XML 검색을 위한 비트맵 인덱싱이 유사 경로를 인식하지 못하는 문제점을 해결하기 위해 유사 경로 탐색을 위한 새로운 비트맵 인덱스를 설계하고 이를 통해 효과적으로 유사 문서를 검색할 수 있는 기법을 제안한다. 제안된 기법은 노드들을 단위로 하는 새로운 비트맵 인덱스를 구성하고 구성된 인덱스의 중심을 통해 유사 경로 탐색을 위한 클러스터들을 선별적으로 검색한다. 그리고 유사 경로 탐색을 통해 추출된 경로들을 유사 문서 검색에 활용함으로써 비트맵 인덱싱의 빠른 성능을 그대로 유지하면서 기존의 XML 검색을 위한 비트맵 인덱싱이 유사 경로를 인식하지 못함으로써 발생하는 유사 문서 탐색의 성능 저하를 효과적으로 해결한다.

## 1. 서론

XML[1]은 현대의 많은 연구들과 새로운 기술들에서 핵심적인 요소로 자리매김하고 있다. 이에 따라 XML을 효과적으로 저장 및 검색하고 이것을 활용하기 위한 연구는 XML을 활용하는 많은 분야의 기반 기술로써 중요한 의미를 갖는다. XML에 대한 연구는 일반적으로 특정 자료를 표현하기 위한 구조를 미리 정의함으로써 XML의 구조적 상이성을 미연에 방지하기 위한 것과 구조적으로 상이한 문서에서 효과적으로 정보를 저장 및 검색하기 위한 것으로 나눌 수 있다. 이미 유명한 RDBMS 벤더들은 자신의 시스템에서 XML 검색을 효과적으로 지원하기 위한 방법을 제시하고 있으며 다양한 XML 전문 검색 시스템들도 등장하고 있다.

XML 검색을 위한 기존의 비트맵 인덱싱은 XML을 문서, 경로, 단어로 구성된 3차원 비트맵 인덱스에 매핑하고 Bit-wise 연산을 통해 빠르게 정보를 추출함으로써 뛰어난 성능을 입증하였다. 그러나 이것은 유사 문서를 수집하고 검색하기 위해 경로 전체를 인덱스 구성의 단위로 사용함으로써 유사 경로를 인식하지 못하는 문제를 초래하였다. 결과적으로 비트맵 인덱싱을 통한 유사 문서 검색은 단순히 완전히 일치하는 경로를 얼마나 포함하는지를 측정하는 것이 고작이었다.

이에 따라 본 논문에서는 기존의 비트맵 인덱싱이 유사 경로를 인식하지 못함으로써 발생하는 문제점을 해결하기 위해 경로 탐색을 위한 새로운 비트맵 인덱스를 설계하고 이를 통해 효과적으로 유사 경로를 탐색할 수 있는 기법에 대해 제안한다.

## 2. 관련연구

### 2.1 XML 검색을 위한 비트맵 인덱싱

XML 검색을 위한 기존의 비트맵 인덱싱인 BitCube[2]와 xPlaneb[3]는 XML 문서에서 효과적으로 정보를 추출하기 위해 비트맵 인덱스를 활용한다. 이것은 많은 XML 문서에서 효율적으로 정보를 추출하기 위해 클러스터내의 문서와 경로 그리고 단어 ID로 구성된 Bit-wise 연산이 가능한 1비트의 필드들로 구성된 3차원 배열 인덱스를 구성한다. 그리고 유사한 구조를 갖는 여러 문서들은 하나의 3차원 비트맵 인덱스에 수집된다. 그리고 문서의 경로를 중심으로 유사한 구조의 문서들을 수집하는데 이 때 각각의 문서가 얼마나 같은 경로를 고르게 포함하고 있는지의 여부가 문서간의 유사도를 판별하는 기준이 된다. 3차원 비트맵 인덱싱은 이렇게 구성된 인덱스에서 Bit-wise 연산을 수행함으로써 빠른 속도로 결과를 도출한다. BitCube는 기존의 XQEngine[5], XYZFind[6]와 같은 시스템들과의 성능 평가에서 빠른 검색 속도를 통해 이미 뛰어난 성능을 입증하였으며 xPlaneb는 클러스터에 대량의 문서가 적재되는 경우 BitCube보다 뛰어난 성능을 보임을 입증하였다[2,3].

그러나 이와 같은 3차원 비트맵 인덱싱은 유사한 XML 문서를 수집하는 과정에서 경로 전체를 단위로 사용함으로

써 정확하고 융통성있는 유사 문서의 탐색이 불가능하다. 다시말해 기존의 3차원 비트맵 인덱싱은 완전히 일치하는 경로는 인식할 수 있으나 완전히 일치하지 않는 경로들은 전혀 인식할 수 없으므로 유사 문서 탐색에 치명적인 결함을 갖는다.

### 2.2 경로 구성 유사도를 이용한 XML 문서 인식

경로 구성 유사도(Path Construction Similarity: P.C.Sim.)[4]는 기존의 3차원 비트맵 인덱싱의 문제점을 해결하기위해 경로를 구성하는 노드들의 구성이 얼마나 순차적이고 정확한지를 판별하기 위해 제안되었다. 경로 구성 유사도에서는 기준 경로와 비교 대상 경로의 유사도를 다음과 같이 정의한다.

경로 구성 유사도

$P.C.Sim. = Path.P.C.Cor. \bullet List.P.C.Cor.$

- $Path.P.C.Cor.$ = 기준 경로의 경로 구성 정확도
- $List.P.C.Cor.$ = 비교 대상 경로의 경로 구성 정확도

경로 구성 정확도

해당 경로를 구성하는 모든 태그의 수를 k라 하면,

$$P.C.Cor. = \frac{\sum_{i=1}^{k} Tag.Value_i}{k}$$

- $Tag.Value$ = 경로에 존재하는 태그의 가치

태그의 가치

연결에 거리가 발생하지 않는다면,

$Tag.Value = 1$

연결에 n개의 거리가 발생한다면,

$$Tag.Value = \frac{1}{2^n}$$

경로 구성 유사도는 경로들의 유사성을 판별하고 유사 문서 탐색을 가능하게 하므로써 3차원 비트맵 인덱싱이 유사 문서를 인식하지 못하는 문제점을 해결하였다. 그러나 이를 위해 별도의 유사 경로 테이블을 구성하고 유지하므로써 성능 저하를 초래하였다.

본 논문에서는 기존의 3차원 비트맵 인덱싱의 유사 경로를 인식하지 못하는 문제점을 해결하고 이에 따른 성능 저하를 최소화하기 위해 경로를 구성하는 노드를 단위로 하는 새로운 3차원 비트맵 인덱스를 제안한다. 그리고 이를 이용하여 효과적인 유사 문서 검색이 가능한 새로운 XML 문서 검색 시스템을 설계 및 구현한다.

## 3. 유사 경로 탐색을 이용한 유사 문서 검색

XML 검색을 위한 3차원 비트맵 인덱싱은 완전히 일치하지 않는 경로를 인식하지 못함으로 인해 유사 문서 검색에 큰 문제점을 내포하게 된다. 예를 들어 그림 3.1과 같은 두 개의 문서가 존재하며 이 문서들은 완전히 동일한 구조이지만 하나의 문서는 A와 C, B, D 노드 사이에 G라는 다른 노드가 삽입되어 있다. 기존의 3차원 비트맵 인덱싱은 경로 전체를 하나의 단위로 인덱스를 구성함으로 하나의 문서에서 추출한 경로 A.C.E와 다른 문서에서 추출한 경로 A.G.C.E는 완전히 다른 경로로 인식되며 이는 다른 경로의 경우도 마찬가지이다. 결과적으로 두 문서는 일치되는 경로가 없으므로 완전히 다른 문서로 인식된다.

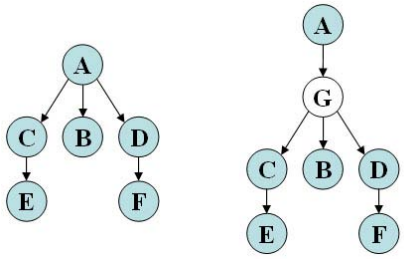


**그림 3.1** 유사 구조를 갖는 문서들

### 3.1 유사 경로 탐색을 위한 비트맵 인덱스

유사 경로를 정확하게 인식하기 위해서는 경로를 구성하는 노드들을 단위로 탐색이 수행되어야 한다. 이를 위해 경로와 그 경로를 구성하는 노드들로 이루어진 비트맵 인덱스를 구성한다. 그림 3.2는 유사 경로 탐색을 위한 비트맵 인덱스이다. 이것은 경로 이름 리스트와 노드 이름 리스트를 갖는다. 기본 비트맵 인덱스(Base Bitmap Index)는 직접 탐색에 활용되기 보다는 클러스터를 구성하는 경로들간의 유사도를 유지하기 위해 클러스터를 분할하거나 재구성할 때 사용된다. 또 각각의 경로의 순서를 기록하고 있는 Next Node Ptr.과 함께 클러스터의 대표 경로나 중심을 추출하기위해 사용된다. 다음 절에서는 이 새로운 비트맵 인덱스를 통해 유사 경로를 탐색하는 기법에 대해 논의한다.

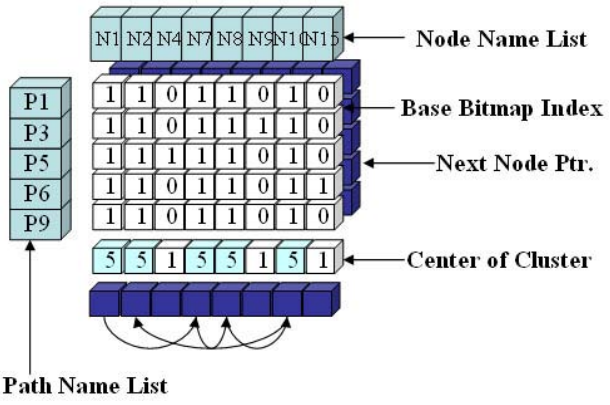


**그림 3.2** 유사 경로 탐색을 위한 비트맵 인덱스

### 3.2 유사 경로 탐색

기존의 유사 문서 탐색을 위한 비트맵 인덱스는 인덱스를 구성하는 각각의 경로들 사이의 순서가 어떤 의미를 갖지 않는다. 하지만 유사 경로 탐색을 위한 비트맵 인덱스의 경우 경로를 구성하는 노드들의 순서가 매우 중요한 의미를 갖게 된다. 그러므로 클러스터의 중심을 추출할 때 노드의 순서를 결정하고 이를 인덱스내에 기록하는 것이 필요하다. 다시말해 인덱스로부터 대표 경로를 추출하는 것이 필요하다. 노드의 순서 결정은 처음 클러스터가 생성될 때부터 수행되며 클러스터에 경로가 추가될 때마다 재수행된다.

어떤 유사 경로 탐색을 위한 클러스터에 그림 3.3과 같이 P1~5의 다섯 가지 경로가 존재하고 A~F의 6개의 노드가 존재한다면 대표 경로는 A.B.E.F가 된다. 만약 이 클러스터에 P6이라는 경로가 추가된다면 클러스터에는 P1~6의 여

섯 가지 경로가 존재하고 A-G까지의 7 개의 노드가 존재하게 되며 대표 경로는 A.B.D.E.F로 갱신된다.

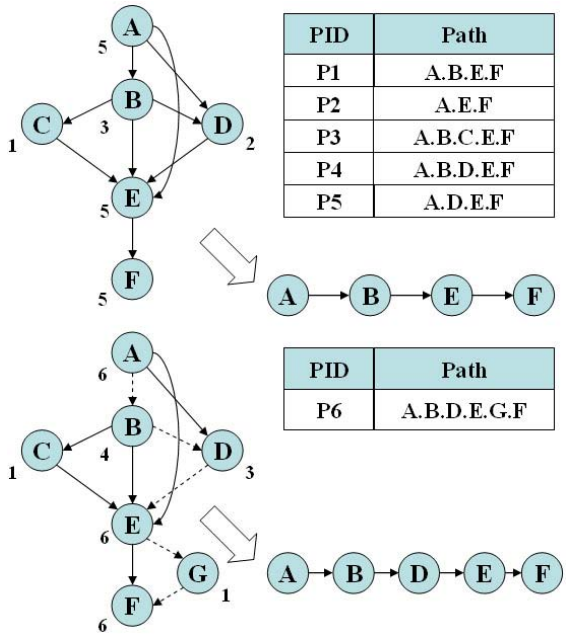


**그림 3.3** 유사 경로 탐색의 예

특정 클러스터의 대표 경로를 추출하는 것은 단순히 BFS 탐색과 카운터만을 사용한다. 만약 그림 3.3과 같이 5개의 경로가 존재하는 클러스터는 각각의 경로에서 하나씩 노드를 읽고 해당 노드에 대한 카운터에 기록한다. 다시말해 첫 번째 탐색에서 각각의 경로들의 첫 번째 노드들을 읽고 해당 노드가 클러스터내의 노드 이름 리스트에 존재한다면 그 노드의 카운터에 기록한다. 두 번째 탐색에서는 각각의 경로들의 두 번째 노드들을 읽고 기록한다. 이와 같은 방식으로 모든 노드를 탐색한다. 표 3.1은 그림 3.3과 같은 클러스터에 A.B.D.E.G.F라는 새로운 경로가 삽입되었을 때 대표 경로가 추출되는 과정을 보여준다.

**표 3.1** 대표 경로 추출

| Depth / Node Name | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|
| A | 6 | – | – | – | – | – |
| B | 0 | 4 | – | – | – | – |
| C | 0 | 0 | 1 | 1 | 1 | 1 |
| D | 0 | 1 | 3 | – | – | – |
| E | 0 | 1 | 3 | – | – | – |
| F | 0 | 0 | 1 | 3 | – | – |
| G | 0 | 0 | 0 | 0 | 1 | 1 |

대표 경로를 추출하기 위해서는 대표 노드들을 순위에 따라 선출해야 한다. 대표 노드의 선출은 대표성에 의해 결정된다. 각각의 노드는 자신의 대표성(Representation)이 어떤 임계치를 넘어서면 카운트를 정지하고 대표 노드로 선출된다. 대표성은 다음과 같이 정의한다.

$$rep. = \frac{C}{P}$$

- $C$ = 현재 노드의 발견 횟수
- $P$ = 클러스터내의 모든 경로의 수

첫 번째 노드 탐색에서 A가 대표 경로를 구성하는 첫 번째 대표 노드로 선출되는 것을 알 수 있다. 그리고 탐색을 통해 순차적으로 B, D, E, F가 대표 노드로 선출된다. 이 순차적인 대표 노드의 리스트가 해당 클러스터의 대표 경로가 된다.

선출된 대표 경로는 새로운 경로가 삽입되거나 이로 인해 클러스터에 수집된 문서들의 유사성이 저하될 때 갱신된다. 대표 경로는 유사 문서 검색을 수행하기 전에 유사 경로 탐색을 수행한다. 이때 질의로 들어온 문서는 복수의 기준 경로를 갖게 되며 각각의 기준 경로는 비교 경로로써 대표 경로와 유사도를 측정하여 유사 구조의 경로들을 추출한다. 기존의 경로 구성 유사도는 유사도 측정시에 어떤 경로에 존재하는 특정 노드의 유무를 판별할 수 없었다. 그로 인해 경로의 유사성을 판별하기 위한 탐색에서 낙관적 추정을 통해 탐색의 종료 시점을 결정할 필요가 있었으며 이와 같은 유사도 측정을 모든 경로에 대해 수행해야 하는 문제점을 내포하고 있었다. 그러나 이미 3.1절에서 본 것처럼 제안된 유사 경로 탐색을 위한 비트맵 인덱스는 어떤 경로에 포함된 노드의 유무를 즉시 판단할 수 있다. 그러므로 기존의 경로 구성 유사도를 이용한 탐색보다 탐색 종료 시점을 단축할 수 있다. 그리고 새로운 비트맵 인덱스는 클러스터 중심을 통해 유사도 측정 범위를 효과적으로 축소시킨다. 예를 들어 새로운 경로가 질의로 들어온다면 시스템은 이 경로와 모든 경로들과의 유사도를 측정하는 대신 유사 경로가 수집된 각각의 클러스터의 중심을 통해 해당 클러스터의 대표 경로와 새로운 경로의 유사도 측정의 필요성 여부를 판단한다. 만약 측정할 필요가 없다고 판단된다면 해당 클러스터에 속한 경로들은 탐색에서 배제될 것이며 필요성이 있다고 판단된다면 비로소 클러스터의 대표 경로와 새로운 경로와의 유사도 측정을 수행한다. 유사도 측정의 필요성은 기대 유사도( *Expected Similarity* )라 하며 다음과 같이 정의한다.

i) $T - M < 2$ 인 경우

$$Exp.Sim. = \frac{M + \frac{\left\lfloor \frac{T}{M-1} \right\rfloor \cdot T \bmod (M-1)}{2}}{T}$$

ii) $T - M \geq 2$ 인 경우

$$Exp.Sim. = \frac{M + \frac{4T - 4M - 6 \cdot T \bmod (M-1) \cdot \left\lfloor \frac{T}{M-1} \right\rfloor}{2^{\left\lfloor \frac{T}{M-1} \right\rfloor}}}{T}$$

- $M$ = 일치하는 노드의 수
- $T$ = 전체 노드의 수

이것은 새로운 경로와 대표 경로가 가장 이상적으로 구성된 경우의 경로 구성 유사도와 같다. 비트맵 인덱싱은 Bit-

Wise 연산을 통해 유사도 측정의 필요성을 판별하기 위해 필요한 일치되는 노드의 수와 일치하지 않는 노드의 수를 굉장히 빠른 속도로 알아내고 유사도 측정이 필요하다고 판단되는 클러스터에 대해서만 실제로 경로 구성 유사도를 측정한다. 만약 사용자가 질의한 문서에 A.D.B.H.E.F라는 새로운 경로가 포함되어 있다면 이 경로는 우선 유사 경로 탐색을 위한 각각의 클러스터의 중심들과 유사도 측정의 필요성을 판별한다. 그림 3.3의 클러스터를 대상으로 유사도 측정의 필요성을 판별한다면 일치하는 노드의 수( $M$ )는 4이고 일치하는 않는 노드의 수( $N$ )는 2이므로 기대치는 0.83이 된다. 결국 이 클러스터는 새로운 경로와 유사도를 측정할 필요가 있다고 판단되므로 실제로 대표 경로와의 경로 구성 유사도를 측정한다. 표 3.2는 그림 3.3에서 추출된 대표 경로와 새로운 경로와의 경로 구성 유사도를 측정하는 과정을 나타낸다. 두 경로의 경로 구성 유사도는 0.74가 되며 유사 경로 탐색에 대한 임계치가 0.7이라면 두 경로는 유사 경로로 인식된다. 유사 경로들은 질의와 함께 유사 문서 검색을 위한 비트맵 인덱스에 전달되고 이를 활용하여 기존의 비트맵 인덱스는 유사 문서 검색을 수행한다.

**표 3.2** 경로 구성 유사도 계산

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 대표경로 | Depth | 0 | 1 | 1 | 2 | 3 | 4 |
| | Node Name | A | B | B | D | E | F |
| | Node Value | 1 | 0.5 | 1 | 0.5 | 1 | 1 |
| | P.C.Cor. | 1 | 0.75 | 1 | 0.83 | 0.87 | 0.9 |
| 비교경로 | Depth | 0 | 1 | 2 | 3 | 4 | 5 |
| | Node Name | A | D | B | H | E | F |
| | Node Value | 1 | 0.5 | 1 | 0.5 | 1 | 1 |
| | P.C.Cor. | 1 | 0.75 | 0.83 | 0.75 | 0.8 | 0.83 |
| P.C.Sim. | | 1 | 0.56 | 0.83 | 0.62 | 0.69 | 0.74 |

결과적으로 제안된 기법은 유사 경로 탐색에서 탐색 범위를 비약적으로 축소하고 빠르게 유사 경로를 탐색함으로써 뛰어난 성능으로 기존의 비트맵 인덱싱의 단점을 보완한다.

## 4. 설계 및 구현

제안된 유사 경로 탐색을 위한 비트맵 인덱스는 유사 문서 검색을 위한 비트맵 인덱스와 연동하여 기존의 시스템이 유사 경로를 인식하지 못함으로써 발생하는 문제점을 해결한다. 사용자의 질의는 질의 분석기(Query Processing Module)를 통해 복수의 경로로 변환되어 유사 경로 탐색을 위한 비트맵 인덱스(Bitmap Index for Similar Paths Retrieval)에 전달된다. 유사 경로 탐색을 위한 비트맵 인덱스는 전달받은 각각의 경로들과 유사한 경로들을 추출하고 이를 기존의 경로들과 함께 유사 문서 검색을 위한 비트맵 인덱스(Bitmap Index for Similar Documents Retrieval)에 전달한다. 유사 문서 검색을 위한 비트맵 인덱스는 전달받은 경로들을 통해 유사 문서를 검색하고 이를 사용자에게 반환한다. 그림 4.1은 새로운 유사 문서 검색 시스템의 구조를 나타낸다.

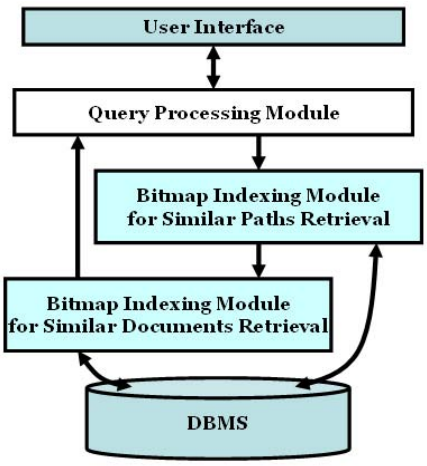


**그림 4.1** 유사 문서 검색 시스템의 구조

## 5. 결론

기존의 XML 검색을 위한 비트맵 인덱싱은 유사 문서를 검색하기 위해 경로 전체가 인덱스를 구성하는 단위가 되었다. 그러므로 문서의 구조가 변경되는 경우 변경전에 문서에서 추출한 경로와 변경 뒤에 문서에서 추출된 경로가 정도에 상관없이 완전히 다른 것으로 인식될 수 밖에 없었으며 결과적으로 XML 검색을 위한 기존의 비트맵 인덱싱은 유사 문서 검색을 위해 단순히 완전히 일치하는 경로를 얼마나 포함하는지를 측정하는 것이 고작이었다. 이에 따라 본 논문에서는 기존의 XML 검색을 위한 비트맵 인덱싱이 유사 경로를 인식하지 못함으로써 발생하는 문제점을 해결하고자 유사 경로 탐색을 위한 새로운 비트맵 인덱스를 설계하고 이를 통해 효과적으로 유사 문서를 검색할 수 있는 기법을 제안하였다. 제안된 기법은 전체 경로를 단위로 하는 인덱스 대신 경로를 구성하는 노드들을 단위로 하는 유사 경로 탐색을 위한 새로운 비트맵 인덱스를 구성하고 구성된 인덱스의 중심을 통해 유사 경로를 탐색하기 위한 클러스터들을 선별한다. 그리고 선별된 클러스터들에 대해 선택적으로 유사도 측정을 수행함으로써 많은 클러스터에서 빠르게 유사 경로를 추출한다. 그리고 유사 문서 검색을 위한 기존의 비트맵 인덱스에 유사 경로 탐색을 통해 추출된 경로들을 전달하고 이를 검색에 활용함으로써 비트맵 인덱싱의 빠른 성능을 그대로 유지하면서 기존의 XML 검색을 위한 비트맵 인덱싱이 유사 경로를 인식하지 못함으로써 발생하는 문제점을 효과적으로 해결하였다.

앞으로 제안한 기법이 유사어나 동음이의어로 인해 인식률이 저하되는 문제점을 해결하고 정확한 클러스터링을 통해 더욱 뛰어난 성능을 도출할 수 있도록 연구를 지속할 것이다.

## 참고문헌

[1] W3C, "Extensible Markup Language(XML) Version 1.0 (Second Edition)," http://www.w3c.org/TR/ REC-xml, October 2000.

[2] J. Yoon, V. Raghavan, and V. Chakilam, “BitCube: Clustering and Statistical Analysis for XML Documents,” 13th International Conference on Scientific and Statistical Data-base Management, Virginia, July 18~20, 2001.

[3] 이재민, 황병연, “xPlaneb: XML 검색을 위한 연결 리스트를 이용한 3차원 비트맵 인덱싱,” 정보과학회논문지, 게재예정.

[4] 이재민, 황병연, "경로 구성 유사도를 이용한 비트맵 인덱싱 기반 XML 문서 인식 기법," 한국정보처리학회 춘계학술발표논문집 , 제10권, 제1호, pp.1515~1518, 2003.5.

[5] D. Egnor and R. Lord, "XYZFind: Structured Searching in Context with XML," ACM SIGIR Workshop, Athens, Greece, July 2000.

[6] XQEngine, http://www.fatdog.com.