순차패턴에 기반한 XML 문서 클러스터링

황 정 희[†] · 류 근 호[††]

요 약

인터넷의 사용 증가로 정보의 양은 기하급수적으로 증가하고 있으며 웹 데이터의 표준인 XML의 데이터 표현의 유연성으로 인해 EDMS(Electronic Document Management System), ebXML(e-business eXtensible Markup Language) 등 웹 기반의 전자문서를 이용하는 시스템들은 XML를 문서 교환 방식 및 표준 문서 형식으로 도입하고 있는 실정이다. 그러므로 점차 확산되어 가고 있는 XML 문서에 대한 효율적인 문서의 관리와 검색을 위한 연구가 필요하다. 이 논문에서는 다중 문서간의 구조적 유사성을 분류하기 위하여 엘리먼트의 순서적 의미를 갖는 XML 문서를 대상으로 순차패턴을 이용하여 문서의 특성을 반영하는 대표구조를 추출하고 추출된 구조를 기반으로 유사 구조 문서를 클러스터링하는 방법을 제시한다. 이 논문의 제안 알고리즘은 클러스터의 응집도와 클러스터간의 유사도를 함께 고려하는 비용계산 방식을 이용하므로써 클러스터링의 정확도를 높일 수 있는 효과를 얻을 수 있다.

XML Document Clustering Based on Sequential Pattern

Jeong Hee Hwang[†] · Keun Ho Ryu[††]

ABSTRACT

As the use of internet is growing, the amount of information is increasing rapidly and XML that is a standard of the web data has the property of flexibility of data representation. Therefore electronic document systems based on web, such as EDMS (Electronic Document Management System), ebXML (e-business eXtensible Markup Language), have been adopting XML as the method for exchange and standard of documents. So research on the method which can manage and search structural XML documents in an effective way is required. In this paper we propose the clustering method based on structural similarity among the many XML documents, using typical structures extracted from each document by sequential pattern mining in pre-clustering process. The proposed algorithm improves the accuracy of clustering by computing cost considering cluster cohesion and inter-cluster similarity.

키워드 : 문서 클러스터링(Document Clustering), XML 문서(XML Document), 순차패턴(Sequential Pattern), 구조 유사성(Structural Similarity)

1. 서 론

웹의 정보를 표현하는 XML(External Markup Language)은 웹 상에서의 데이터 교환과 저장을 위해 제안된 표준 언어이다. HTML과 XML은 SGML의 부분집합이지만 HTML 태그는 데이터 표현에 중점을 두는 반면 XML 태그는 데이터 그 자체를 기술한다는 특징이 있다[1, 2]. 즉, 사용자가 임의로 엘리먼트를 정의할 수 있고 엘리먼트는 하위 엘리먼트를 가짐으로써 계층적 구조를 형성한다. 이러한 XML의 구조적 특징은 정보 검색, 문서 관리시스템, 그리고 데이터 마이닝 등에 커다란 영향을 미치고 있다[3, 4, 7, 8].

최근 EDMS(Electronic Document Management System), ebXML(e-business eXtensible Markup Language) 등 웹 기반의 전자문서를 이용하는 시스템들은 현재 XML을 문서 교환 및 표준 문서 형식으로 도입하고 있는 실정이다. 이것은 웹의 사용자와 더불어 정보량이 증가함에 따라 웹 상에 존재하는 문서들이 문서의 "내용(content)"보다는 문서의 글자체, 문단의 모양, 문서 구조와 같은 문서의 "외관(format)"을 표현하는 마크업을 사용하므로써 문서의 특징을 표현하고 같은 구조의 문서를 공유하기 위함이다[4-6]. 따라서 점차로 증가되고 있는 XML 문서를 웹이나 다른 사용자 또는 타기업으로부터 획득하여 기존문서와 통합 및 분류하고자 할 경우 유사구조를 갖는 XML 문서에 대해 자동으로 분류할 수 있는 기술이 필요하다[7, 8].

데이터 마이닝 기법은 최근에 증가되고 있는 XML 문서에서 유용한 지식을 탐사하기 위해 필요한 요소이다. XML 문서에 대한 데이터 마이닝 기법중의 하나가 XML 구조 마이닝이며 이 기법은 주로 XML의 스키마 추출을 위해 사용

※ 이 논문은 2002년도 한국학술진흥재단의 지원에 의하여 연구되었음(KRF-2002-002-D00152).
† 준 회 원 : 충북대학교 대학원 전자계산학과
†† 종신회원 : 충북대학교 전기전자컴퓨터공학부 교수
논문접수 : 2003년 7월 24일, 심사완료 : 2003년 9월 8일

되었다[9, 10]. 즉 XML 문서의 데이터를 관계형 데이터베이스로 저장할 때 구조 발견을 목적으로 하는 연구로써 유사 문서를 대표할 수 있는 하나의 구조 정보를 추출하는 것을 목적으로 문서의 엘리먼트 정보와 그 발생 빈도수를 이용하여 스키마를 추출하고 이를 저장 또는 질의에 대한 검색에 사용한다.

마이닝 기법을 이용하는 기존연구의 DTD 추출과 이 연구의 차이점은 DTD 추출은 효율적인 저장을 위해 많은 유사 문서에서 일정한 구조 표현(regular expression)을 만족하는 공통의 구조를 찾는 것이고 이 연구는 공통의 구조 발견이 아닌 서로 다른 구조를 가지는 다중 문서에서 구조적으로 유사성을 갖는 문서를 클러스터링하여 분류한다. 그러므로 문서내의 개별적 단어 검색보다는 전체적인 형식과 주요 제목을 보고 유사성을 판단하는 검색 또는 XML 기반의 문서를 다루는 EDMS, ebXML에서의 문서 관리 및 기업간의 공유 문서 교환, 저장 그리고 문서의 병합을 위한 구조 검색 등에 사용될 수 있다.

먼저 이 논문에서는 각 문서의 대표구조를 찾기 위해 순차 패턴 마이닝 기법[11]을 이용하여 빈발 구조 패턴의 구조적 특징을 추출한다. XML 문서를 구성하는 엘리먼트는 문서 내용에 대한 기본 정보 및 문서의 특성을 구별할 수 있는 의미있는 단어로 구성되므로 이 정보를 문서 분류를 위한 기초 자료로 사용하는 것이다. 그리고 추출된 문서의 계층 구조는 문서를 분류하기 위한 입력자료가 되며 이 논문에서는 클러스터의 정확도를 높이기 위해 [12]에서 제안한 클러스터링 기법을 변형한 새로운 문서 분류 기법의 제시 및 이를 적용하고, 실험을 통해 기존 알고리즘과의 성능을 비교한다.

이 논문에서 제안하는 클러스터링 기법은 클러스터의 주요항목(large item)을 기반으로 클러스터의 응집도와 클러스터간의 유사도를 함께 고려하기 위해 유도된 비용 계산 방식을 이용하므로써 문서 분류의 정확성을 높일 수 있다는 특징이 있다.

이를 위한 논문의 구성은 다음과 같다. 먼저 2장에서는 이 논문의 연구 기반이 되는 XML과 마이닝 그리고 문서의 구조 추출에 대한 기존 연구 내용을 알아보고 3장에서는 순차패턴을 기반으로 하는 XML 문서의 구조 추출방법을 기술한다. 그리고 4장에서는 3장에서 추출된 구조를 가지고 유사 구조를 포함하는 문서들을 자동으로 분류하는 클러스터링 기법을 기술한다. 그리고 5장에서는 제안한 XML의 유사 구조 문서 클러스터링에 대한 실험을 통해 결과를 분석하고 마지막으로 6장에서는 결론을 맺는다.

2. 관련 연구

대부분의 XML에 대한 기존 연구에서는 XML 문서를 저장하기 위한 스키마 추출[13], 인덱스[14], 그리고 질의 처리[15]등에 초점을 두었다. 그러나 다양한 구조를 갖는 XML 문서의 사용이 점차 증가하고 있으므로 XML 문서에 대해 유용한 정보를 추출하고 분류할 수 있는 기법의 연구가 필요하다[7, 8].

[4]에서는 웹 상의 반구조적 문서(Semistructured Document)에서 공통의 일반적인 구조를 발견하는 방법을 제시하였다. 사용자가 지정한 최소의 지지도를 만족하는 문서들에 대한 공통된 구조를 찾기 위해 labeling, nesting, ordering 등과 같은 문서의 구조적 패턴 특성을 고려하여 사용자가 문서들에 대한 구체적 정보보다는 일반적인 공통의 정보를 원할 때 적용할 수 있도록 한다. 그리고 [19]에서는 반구조 데이터를 관계형 데이터베이스에 저장하고 관리하기 위해 반구조 데이터와 질의 언어로 표현되는 관계형 데이터와의 매핑에 대해 빈발 패턴 마이닝을 이용하였다. 데이터 지지도와 데이터 질의의 결합 지지도를 계산하여 그들의 결합 지지도가 커질수록 저장 패턴(Storage Pattern)의 크기가 커지도록 하는 높은 지지도를 가지는 서브패턴에 대한 반환 포인터(backpointer)를 유지하고 다시 발견하고자 하는 패턴위치를 식별할 수 있도록 한다.

[14]의 DTD-Miner는 XML 문서의 구조를 나타내는 트리에 대해 순서와 방향을 가지는 Spanning 그래프로 표현하고 서브모듈에 대한 반복적 그래프의 병합을 통해 모든 문서 트리의 병합이 끝나면 경험적 규칙을 적용하여 Spanning 그래프로부터 생성되는 DTD 생성 기법을 핵심으로 한다. 이 방법에서는 구체적으로 마이닝 기법을 이용하지 않고 사용자가 입력한 유사구조의 XML 문서 집합에 대해 자동으로 공통의 구조를 추출하여 사용자에게 DTD를 제공한다. 이와는 다르게 [16]에서는 XML의 목표문서 인식을 위하여 XML 문서간 유사성을 도출하고 이를 기준으로 일정 임계치 이상의 유사성을 갖는 문서를 목표문서로 인식할 수 있는 유사성 척도의 계산방법을 제시하였다. 그리고 두 문서간의 유사성을 추출하기 위하여 [17]의 순차패턴 마이닝 알고리즘을 이용하여 유사한 엘리먼트로 인식된 것을 중심으로 비교 문서에서 나타난 경로들이 기준 문서에도 나타나는지에 따라 경로를 확장하여 최대 유사 경로를 발견하는 방법을 제시하였다. 그러나 이 연구의 목적은 기준이 되는 문서와 가장 유사한 문서를 찾기 위한 기준문서와의 상대적 유사성에 따른 문서의 분류이며, 다양한 구조를 갖는 다중 XML 문서를 분류하는 방법과는 차이가 있다.

[7, 8, 18]에서는 XML 문서에 대해 적용할 수 있는 마이닝 기법의 분류를 기술하였다. 즉, XML의 구조추출을 위해 XML 구조 마이닝은 필수적 연구이고 하나의 문서에 대한 구조적 마이닝(Intra-structured mining)과 여러 문서에 대한 구조적 마이닝(Inter-structured mining)으로 분류하였다. 그리고 임의의 XML 문서를 미리 정의된 문서 클래스

의 어느 클래스에 속하는지를 예측하기 위해서는 분류 규칙(Classification)을 이용할 수 있다는 것과 다양한 문서간의 유사성을 식별하기 위해서는 군집화(Clustering)를 통해 분류할 수 있음을 언급하였다. 그러나 문서의 분류와 예측을 위해 적용할 수 있는 구체적인 알고리즘은 제시하지 않았다.

[20]에서는 점차 증가하는 XML 문서들에 대한 관리의 필요성을 언급하고 문서의 구조를 표현하는 태그(tag)와 일반 텍스트의 내용에 대한 클러스터링 기법을 제시하였다. 그러나 이 연구에서는 문서의 클러스터링을 위해 거리 값을 기반으로 하는 K-means 알고리즘을 사용하여 문서의 양과 분포에 따라 달라질 수 있는 클러스터의 수를 요구하므로 클러스터링에 대한 유연성이 부족하다.

이와 같이 기존 연구에서는 XML 문서의 구조 추출을 위해 마이닝 기법을 일부 적용하고 있으나 다양한 구조를 갖는 다중 XML 문서를 분류하기 위한 마이닝 기법에 대한 연구는 아직 미진한 상태이다. 특히 마이닝을 이용한 XML 문서의 구조 추출에만 초점을 두고 있을 뿐 추출된 구조를 클러스터링하기 위한 구체적인 연구 방법은 거의 제시되지 않고 있다.

따라서 이 논문에서는 기존 연구에서 일반적으로 사용되는 클러스터간의 중심과 거리를 비교하여 클러스터를 할당하는 거리 기반의 군집 분석 방식을 사용하지 않고 클러스터에 대한 주요항목을 유지하면서 클러스터의 유사도를 높일 수 있는 응집도와 낮은 클러스터간의 유사성을 동시에 고려하여 최상의 클러스터가 되도록 하는 비용 정책을 기반으로 한다. 그러므로 동적인 클러스터링이 가능하고 클러스터의 질을 향상시킬 수 있다.

3. 순차패턴을 이용한 XML 문서의 구조 추출

기존의 텍스트 문서의 분류에서는 문서의 특징을 추출하기 위해 단어의 발생빈도를 가지고 측정하였다[16]. 그러나 XML문서는 기존 문서의 비순차적인 구조와는 다르게 의미가 부여된 엘리먼트의 순차적이고 계층적인 구조로 이루어져 있다. 따라서 XML 문서의 엘리먼트 순서와 부여된 엘리먼트 그 자체는 XML 문서의 형태와 종류를 구분할 수 있게 해 주는 특징을 가지고 있다[9, 10]. 아래 XML 문서의 일부는 XML 문서의 특성을 설명하기 위한 것이다. (a)와 (b) 문서는 같은 엘리먼트들로 구성되어 있지만 (a)는 학생

```
<학생>
 <이름>홍길동</이름>
 <년도>1980</년도>
 <학교>○○대학교</학교>
</학생>
```

(a) 학생 정보 문서

```
<학교>
 <이름>○○대학교</이름>
 <년도>1950</년도>
 <학생>1000명</학생>
</학교>
```

(b) 학교 정보 문서

정보를 나타내는 문서이고 (b)는 학교 정보를 나타내는 문서이다. 이와 같이 엘리먼트의 구조와 순서에 따라 문서의 내용이 다르게 구분되므로 엘리먼트를 통한 문서 구조의 추출은 엘리먼트의 발생횟수뿐만 아니라 발생순서를 고려하는 순차패턴 마이닝을 이용한다.

3.1 전처리 과정

XML 문서는 계층적 구조로 이루어져 있으므로 하나의 문서에 포함되어 있는 경로는 다양하다. 다양한 경로의 XML 문서에서 의미있는 구조를 추출하는 것은 문서의 주제어를 추출하는 것과 유사하다고 볼 수 있다. 즉, XML 문서의 엘리먼트는 문서의 내용을 예측할 수 있도록 하는 문서의 특성을 나타내는 단어로 구성되므로 엘리먼트에 대한 구조 경로를 통해 계층적 구조를 이루는 문서의 의미있는 대표 구조를 추출할 수 있다. 다음의 예제 문서는 문서의 빈발구조를 찾는 전처리과정을 설명하기 위한 책 관리 문서의 일부이다.

```
<bookinventory>
  <nation>
    <English>
      <booktitle>Great Expectations</booktitle>
    </English>
  </nation>
  <book>
    <title>Great Expectations</title>
      <author>
        <name>Charles Dickens</name>
        <born>1812</born>
        <died>1879</died>
        <nationality>English</nationality>
      </author>
      <count>10</count>
  </book>
</bookinventory>
```

위 문서는 레벨 1의 <bookinventory>부터 레벨 5에 해당하는 <name>, <born>, <died>, <nationality> 등으로 구성되어 있다. 이렇게 여러 가지 엘리먼트로 구성되어 있는 문서에서 문서의 특성을 나타내는 구조를 추출하기 위해서는 각각의 엘리먼트를 쉽게 구별할 수 있는 재명명의 절차가 필요하다. 그러므로 위 예제 문서에서 추출될 수 있는 엘리먼트의 구조를 쉽게 식별하기 위하여 다음과 같은 엘리먼트의 매핑 테이블을 이용하여 알파벳으로 재명명한다.

(그림 1)과 같이 재명명된 문서의 구조를 토대로 실제적인 내용을 포함하는 엘리먼트들에 대한 문서 경로를 고유의 번호(id)를 갖는 시퀀스로 간주하고 주어진 최소 지지도를 만족하는 순차 패턴 마이닝 알고리즘을 적용하여 가장 빈발한 시퀀스 패턴, 즉 그 문서를 대표할 수 있는 엘리먼트 구조 정보를 찾는다. 위 예제 문서에서 의미있는 엘리먼

(그림 1) 엘리먼트 매핑 테이블

레벨 1		레벨 2		레벨 3		레벨 4		레벨 5	
엘리먼트	재명명	엘리먼트	재명명	엘리먼트	재명명	엘리먼트	재명명	엘리먼트	재명명
bookinventory	a	nation	b1	English	c1	booktitle	d1	name	e1
		book	b2	title	c2	author	d2	born	e2
						count	d3	died	e3
								nationality	e4

트 경로의 시퀀스는 <표 1>과 같이 표현된다.

〈표 1〉 구조 경로 시퀀스

s_id	s_pass
1	a/b1/c1/d1
2	a/b2/c2
3	a/b2/c2/d2/e1
4	a/b2/c2/d2/e2
5	a/b2/c2/d2/e3
6	a/b2/c2/d2/e4
7	a/b2/c2/d3

문서의 구조 특성을 추출하기 위한 의미있는 엘리먼트의 고유한 경로 구조는 빈발패턴을 찾기 위한 입력 정보가 되며 하나의 경로에 포함되어 있는 엘리먼트들은 발생 시퀀스를 구성하는 하나의 항목으로 고려된다.

3.2 문서 구조 추출

순차 패턴 마이닝 알고리즘은 연관 규칙과는 달리 트랜잭션의 발생 횟수와 발생 순서를 고려하므로 XML 문서의 구조 추출에 적합하다[9-11].

시퀀스들의 집합에 대한 빈발 구조를 추출하기 위하여 이 논문에서는 후보패턴을 생성하지 않는 [11]의 PrefixSpan 알고리즘을 이용한다. 이것은 루트노드에서 시작하여 깊이 우선 검색(Depth First Search)순으로 노드를 확장하면서 PrefixSpan 트리를 만들어 가는 방식이다. 노드를 확장할 때에 그 노드가 나타내는 빈번한 시퀀스를 포함하고 있는 시퀀스만을 모은 시퀀스(prefix) 이후의 부분만을 지정한 Project DB를 이용하는 방법으로 성능의 우수성이 [11]에서 증명되었다.

<표 1>에 대한 빈발구조를 추출하기 위해 PrefixSpan 알고리즘을 적용하는 방법을 간략히 설명하면 다음과 같다.

각 엘리먼트의 경로로 구성된 시퀀스 집합에 대해 최소 지지도를 2로 가정하면 최소 지지도를 만족하는 크기가 1인 엘리먼트의 빈발도는 a : 7, b2 : 6, c2 : 6, d2 : 4이다. 이러한 빈발 요소를 중심으로 시퀀스의 스캔비용과 Project DB 형성 비용을 줄일 수 있는 length-2 빈발 패턴에 대한 S-matrix는 (그림 2)와 같이 형성된다.

(그림 2)에서 크기가 2인 M[a, b2]에 대한 시퀀스, 즉 (a, b2)를 prefix로 하는 projection_DB의 구성요소는 < c2 >, < c2, d2, e1 >, < c2, d2, e2 >, < c2, d2, e3 >, < c2, d2, e4 >, < c2, d3>이고 이 시퀀스를 기초로 다시 빈발 항목에 대한 Project DB의 매트릭스를 (그림 2)와 같은 방식으로 형성하여 점차로 빈발구조의 크기를 늘려가고 더 이상 Projected DB를 만들 필요가 없을 때까지 이 과정을 반복한다.

a				
b2	6			
c2	6	6		
d2	4	4	4	
	a	b2	c2	d2

(그림 2) Length-2 빈발구조의 S-matrix

이 때 XML 문서에서 발견된 빈발 구조의 비율은

F_S = 빈발 구조를 포함하는 시퀀스의 수/ 문서 전체 경로의 수

이고, 위의 예에서 발견된 최대 빈발 구조 < a/b2/c2/d2 >는 전체 문서 경로에 대해 약 57%(4/7)의 비율로 발생한다.

일반적으로 순차패턴 마이닝에 의해 발견된 최대 빈발패턴은 문서에서 가장 공통적으로 사용되는 구조로서 중요한 의미를 갖지만 이 논문에서는 최대빈발패턴의 구조가 아니더라도 최대 빈발 구조에 대한 일정 비율이상을 만족하는 구조(예, 최대빈발구조 길이 5 × 80% = 빈발 구조 길이 4)도 중요한 의미로 간주하고 중복되지 않는 구조에 대해서는 클러스터링을 위한 기초 입력 자료에 포함한다. 이것은 하나의 문서에 여러 가지 주제가 함께 나타날 수 있는 경우, 즉 최대 빈발 패턴의 구조만이 그 문서를 대표하는 유일한 구조가 아닐 경우를 고려하기 위한 것이다.

이렇게 순차 패턴 알고리즘을 통해 찾아진 빈발패턴은 문서에서 하위노드 엘리먼트를 많이 포함한 엘리먼트일수록 발생 빈도수가 많이 나타나게 되고 이것은 문서에서 그 엘리먼트가 차지하는 비중이 크다는 것을 의미한다. 따라서 순차패턴에 의한 빈발 구조 추출은 그 문서의 구조적 특징을 가장 잘 반영하는 문서의 분류 기준이 된다.

4. 주요항목에 기반한 유사 구조 문서 클러스터링

XML 문서의 유사 구조 문서 클러스터링은 문서간의 구조적 유사도를 바탕으로 연관된 문서들을 그룹화 함으로써

많은 XML 문서들을 분류하고 검색하는 데 있어 효율적이다. 이 장에서는 3장의 순차패턴에 의해 추출된 XML 문서의 구조적 특성을 기반으로 [12]에서 제안한 주요항목 기반의 클러스터링 알고리즘을 변형하여 적용하는 새로운 클러스터링 방법을 제시한다.

4.1 클러스터링을 위한 기본 정의

순차패턴 마이닝에 의해 추출된 각 문서의 빈발구조는 전체 문서에서 미리 주어진 지지도에 대해 만족하는, XML 문서의 엘리먼트 순서에 기반하여 추출된 구조이다. 주요항목을 이용한 클러스터링을 하기 위해서는 하나의 문서를 트랜잭션으로 가정하고 각 문서에서 추출된 빈발 구조를 트랜잭션의 항목으로 취급하여 유사한 항목기준의 그룹으로 문서를 클러스터링 한다.

모든 트랜잭션에 포함되어 있는 항목들의 집합 $I = \{i_1, i_2, \cdots, i_n\}$하고, 클러스터 집합 $C = \{C_1, C_2, \cdots, C_m\}$, 트랜잭션 집합 $T = \{t_1, t_2, t_3, \cdots, t_k\}$이라 표기한다. 먼저 클러스터링을 수행하기 위한 주요항목의 개념을 정의하면 다음과 같다.

[정의 1] 주요항목(Large Item)

클러스터 Ci에 대한 항목의 지지도는 Ci에서 항목 i_j($j <= n$)를 포함하고 있는 트랜잭션의 수이고, 사용자가 지정한 최소 지지도, $\theta(0 < \theta \leq 1)$에 대해 Ci내에서 그 항목을 포함하고 있는 트랜잭션의 수가 항목의 지지도 $Sup = \theta \times |Ci|$ 이상이면 항목 i_j는 Ci의 주요항목이다.

$$Ci(Li_j) = |Ci(Ti_{j \in I})| >= Sup$$

이 때 $|Ci|$ = 클러스터 Ci에 포함된 전체 트랜잭션의 수이며, $|Ci(Ti_{j \in I})|$는 클러스터 Ci에 포함된, 항목 i_j를 포함하고 있는 트랜잭션의 수를 의미한다.

하나의 클러스터내의 트랜잭션에 포함되어 있는 모든 항목은 주요항목과 비주요항목으로 나뉘어지며 비주요항목은 주요항목과 반대 개념의 후보항목이다. 임의의 클러스터에 새로운 문서가 할당되면 그에 따라 지지도의 수가 변화하고 주요항목과 비주요항목에도 변화가 발생한다. 즉, 비주요항목은 주요항목으로 변환될 수 있도록 하는 공통의 구조 항목이 삽입되면 주요항목이 될 수 있다.

[정의 2] 비주요항목(Small Item)

비주요항목은 클러스터 Ci에서 일정 지지도를 만족하지 못하는 항목으로, 클러스터내의 모든 트랜잭션중에서 항목 i_j를 포함하고 있는 트랜잭션의 수가 일정 지지도를 만족하지 못하면 항목 i_j는 Ci의 비주요항목이다.

$$Ci(Si_j) = |Ci(Ti_{j \in I})| < Sup$$

따라서 클러스터 Ci에 존재하는 모든 항목을 D(Ci), Ci의 주요항목을 Ci(L), 비주요항목을 Ci(S)라 하면, $D(Ci) = Ci(S) \cup Ci(L)$라 할 수 있고 비주요항목의 집합 $Ci(S) = D(Ci) - Ci(L)$로 표현할 수 있다.

기존의 연구[12]에서는 클러스터에 포함되어 있는 트랜잭션에 대해 주요항목과 비주요항목으로 구분하고 클러스터내의 비유사성을 측정하기 위해 전체 클러스터의 비주요 항목 집합의 수를 나타내는 intra-cluster 비용과 클러스터간의 유사성 정도를 측정하기 위한 전체 클러스터에서의 중복 주요항목의 수를 나타내는 inter-cluster 비용을 합한 비용이 최소가 되도록 하는 클러스터를 구성한다. 이것은 intra-cluster 비용을 통하여 각 클러스터에 대한 비주요항목의 수가 작도록 하므로써 유사성이 높은 그룹으로 군집화하고자 하는 것이며 inter-cluster 비용을 통해 서로 다른 클러스터간의 비유사성을 유지하고자 하는 의도이다.

그러나 이것은 하나의 클러스터에 포함되어 있는 주요항목과 비주요항목의 비율, 그리고 각 클러스터에서 발생하는 공통의 비주요항목에 대한 크기를 고려하지 않아 실제적으로 클러스터내의 유사성과 클러스터간의 비유사성을 잘 반영할 수 없다. 따라서 이 논문에서는 클러스터의 응집도를 높게 하고 클러스터간의 유사도를 낮게 하는 양질의 클러스터 생성을 위하여 클러스터의 할당을 위한 비용 계산 방식을 새롭게 정의한다. 그리고 비용 계산을 정의하기 위해 함께 고려되어야 하는 클러스터의 응집도, 클러스터간의 유사도 및 클러스터의 정제과정에서 고려되는 클러스터 참여도를 정의한다.

클러스터의 응집도는 클러스터에 포함된 문서들에 대한 유사밀도를 나타내는 척도로써, 클러스터에 포함되어 있는 대표구조를 구성하는 문서들의 유사 정도를 의미하며 다음과 같이 정의한다.

[정의 3] 클러스터 응집도

클러스터 Ci의 응집도 c(Ci)는 클러스터 Ci에 포함된 전체 항목 D(Ci) 중에서 주요항목이 차지하는 비율이다. 이것은 식 (1)과 같이 계산하고 1의 값에 가까울수록 좋은 응집도를 나타낸다.

$$c(Ci) = \frac{Ci(L)}{D(Ci)} = \frac{D(Ci) - Ci(S)}{D(Ci)} \quad (1)$$

식 (1)은 하나의 클러스터에 포함된 문서들에서 주어진 지지도를 만족하는 항목들에 대한 전체 항목에 대한 비율이고 클러스터에 포함될 수 있는 문서의 결합 가능성에 대한 비교 기준이 된다. 다른 클러스터와 비교하여 응집도가 크다는 것은 유사 구조의 문서가 더 잘 밀집되어 있는 클러스터를 의미하며 전체적인 클러스터의 응집도는 다음과 같이 계산한다.

$$c(C) = \frac{\sum_{i=1}^{n} c(Ci)}{n}$$

[정의 4] 클러스터간의 유사도

클러스터 Ci, Cj에 포함된 모든 항목 집합에 대한 각 클러스터의 주요항목들에 대한 공통 주요항목의 비율과 비주요항목들에 대한 공통의 비주요항목의 비율의 합을 클러스터 Ci, Cj의 유사도라 한다. 이것은 식 (2)와 같이 계산하고 0에 가까울수록 거의 유사성이 없는 좋은 클러스터를 나타낸다.

$$n\_s(Ci, Cj) = \frac{\frac{Ci(L)+Cj(L)-L(Ci\cup Cj)}{Ci(L)+Cj(L)} + \frac{Ci(S)+Cj(S)-S(Ci\cup Cj)}{Ci(S)+Cj(S)}}{2} \quad (2)$$

이 때 Ci(L) + Cj(L) - L(Ci∪Cj)는 클러스터 Ci와 Cj의 주요항목에서 공통으로 포함되어 있는 항목의 수이고 Ci(S) + Cj(S) - S(Ci∪Cj)는 공통의 비주요항목 수이다. 식 (2)에 의한 클러스터 Ci와 Cj의 유사도 n_s(Ci, Cj)가 낮을수록 클러스터간의 유사성이 적다.

또한 클러스터링 결과로써 생성된 전체 클러스터간의 유사도 측정은 위의 식을 이용하여 다음과 같이 계산한다.

$$n\_s(C) = \frac{\frac{\cap(L)}{\sum_{i=1}^{n} Ci(L)} + \frac{\cap(S)}{\sum_{i=1}^{n} Ci(S)}}{n}$$

여기서 ∩(L)와 ∩(S)은 전체 클러스터에 대한 중복의 주요항목 수와 비주요항목의 수를 나타내므로, 이것은 다시

$$\cap(L) = \sum_{i=1}^{n} Ci(L) - \cup_{i=1..n} Ci(L),$$

$$\cap(S) = \sum_{i=1}^{n} Ci(S) - \cup_{i=1..n} Ci(S)$$

로 표현할 수 있다.

클러스터간의 유사도 측정은 클러스터의 응집도와 마찬가지고 임의의 문서를 하나의 클러스터에 할당하기 위해 다른 클러스터들과의 유사도 정도를 고려하는 것으로 클러스터의 할당 기준이 되는 각 클러스터의 구별성을 의미한다.

[정의 3]과 [정의 4]에서 정의된 클러스터의 응집도와 클러스터간의 유사도를 함께 고려하기 위해 유도된 클러스터의 할당 기준을 나타내는 클러스터의 비용 정의는 다음과 같다.

[정의 5] 클러스터 비용

클러스터에 삽입되는 문서에 대한 주요항목의 공통 항목 비율과 문서의 삽입으로 인해 비주요항목으로의 변환 비율의 차이를 클러스터 비용이라 한다. 이것은 식 (3)과 같이 계산하고, 비용이 최대가 되는 클러스터에 문서를 할당한다.

$$\text{cost(Ci)} = \frac{Ci(L)\cap tk}{|tk|} - \frac{Tran(S)}{Ci(S)} \quad (3)$$

여기서 |tk|는 하나의 문서에 포함된 항목 수, Ci(L) ∩ tk는 주요항목과의 공통 항목 수, 그리고 Tran(S)는 문서의 삽입으로 인해 비주요항목으로 변환되는 수를 의미하며 이것은 Tran(S) = Ci(S) - Before(Ci(S))로 표시할 수 있다.

새로운 문서를 특정 클러스터에 할당하기 위한 비용 산출식은 삽입되는 문서의 항목과 주요항목과의 공통항목을 많이 포함할수록 더 높은 응집도를 나타내고, 반면에 비주요항목의 수를 적게 하므로써 다른 클러스터와의 유사성을 낮게 유도하는 개념을 반영한다. 즉, 주요항목과의 공통항목 비율에 대한 비주요항목으로 변환되는 비율의 차이 값은 클러스터의 할당 가능성의 정도를 의미한다.

그리고 cost(Ci)의 최소 비용은 0보다 커야 삽입이 가능하다. 최소 비용의 의미는 클러스터에 포함됨으로 인해 응집도가 더 이상 작아지는 것을 방지하여 전체적으로 좋은 클러스터링 결과를 유도한다.

클러스터의 정제과정에서 초기 할당된 클러스터에 대한 새로운 클러스터의 할당 여부를 결정하는 기준은 다음에 정의되는 클러스터의 참여도에 의한다.

[정의 6] 클러스터 참여도

두 개의 클러스터 Ci, Cj가 있을 때 Ci에 초기 할당된 문서 t_k가 Cj에 속할 가능성의 정도를 클러스터의 참여도라 하고 식 (4)와 같이 표현한다.

$$p(Ci(t_k) \Rightarrow Cj) = \frac{Ci(|tk|)\cap Cj(L)}{Cj(L)} >= \omega \quad (4)$$

(ω : 최소 참여도 가중치)

$Ci(|t_k|)$는 클러스터 Ci에 속하는 문서 t_k에 포함된 모든 항목을 나타내며 새로운 클러스터의 주요항목과 공통항목이 최소 참여도의 가중치 ω를 만족할 때 클러스터에 참여할 수 있는 가능성이 있다고 판단하며 식 (4)에 의한 값이 높을수록 참여 가능성은 높다.

4.2 XML 문서 클러스터링

이 논문에서 제시하는 XML 문서 클러스터링 방법은 앞의 3장에서 제시한 순차패턴을 이용한 XML 문서의 빈발 구조의 추출 과정 통해 클러스터링을 수행한다.

단계 1 : XML 문서에서 내용값을 갖는 엘리먼트에 대한 경로를 추출한다.

단계 2 : 추출된 경로를 대상으로 순차패턴을 이용하여 각 문서를 대표할 수 있는 빈발 구조를 추출하고 추출된 구조는 유사 구조 문서를 클러스터링하기 위 한 기초 자료로 입력한다.

단계 3 : 각 문서의 빈발 구조를 하나의 트랜잭션 항목으로 간주하여 클러스터의 응집도가 높고 클러스터간의 유사도를 낮게 유도하는 클러스터의 비용계산에 의

해 가장 높은 비용을 나타내는 클러스터에 각 문서를 할당한다. 이 때 할당되는 클러스터는 새로운 클러스터의 생성 및 이미 생성되어 있는 클러 스터이고, 이 단계에서 모든 문서는 각 클러스터에 초기 할당된다.

단계 4 : 초기 할당된 각 클러스터의 주요항목을 기준으로 정제과정에서는 클러스터의 참여도를 계산하고 참여도가 높은 클러스터로 문서를 이동시킨다. 이것은 초기 할당된 클러스터보다 새로운 클러스터로 이동시켜 클러스터의 응집도와 클러스터간의 유사도에서 더 좋은 결과를 생성할 수 있도록 하기 위한 재할당 과정이다.

단계 5 : 더 이상 높은 클러스터의 응집도와 낮은 클러스터간의 유사도를 만족하는 것이 없을 때까지 단계 4 과정을 계속 반복한다.

XML 문서의 구조적 유사성에 의한 클러스터링은 각 문서에 포함되어 있는 빈발 구조를 대상으로 임의의 클러스터에 포함되었을 경우의 비용 계산, 즉 클러스터의 응집도와 클러스터간의 유사도를 고려하는 비용의 비교를 통해 가장 좋은 클러스터에 문서를 할당한다. 이 때 클러스터의 할당을 위한 비용은 기존 연구[12]에서 단지 주요항목과 비주요항목만을 가지고 비용을 측정하므로써 클러스터간의 비유사성과 클러스터의 응집도를 고려하지 않은 단점을 개선하여 양질의 클러스터 생성을 유도한다.

또한 정제과정에서는, 기존 연구의 클러스터에 포함되어 있는 각 트랜잭션을 적어도 한번씩 다른 클러스터로 이동시켜 비용을 다시 계산하는 과정을 개선하여 높은 클러스터의 참여도를 나타내는 클러스터에 대한 이동을 고려하는 비용의 예측을 통해 클러스터의 이동을 결정하므로 데이터 저장공간과 수행시간을 줄일 수 있도록 한다.

다음 예는 이 논문에서 제시한 주요항목기반의 문서 클러스터링에 대하여, 클러스터의 할당 기반이 되는 클러스터의 응집도와 클러스터간의 유사도 측정 방법을 나타낸 예이다.

(예 1) 각 문서의 빈발구조 항목(빈발 구조 경로를 단순화하기 위해 숫자로 표기한다)으로 구성되어 있는 문서 t1, t2, t3, t4, t5, t6 {ti ∈ T}의

t1 = {1, 2, 3}, t2 = {1, 2, 3, 6}, t3 = {1, 3, 4, 5},
t4 = {3, 4, 5}, t5 = {7, 8, 5}, t6 = {7, 8, 6}를 고려하면

사용자 정의 최소 지지도를 60%라 할 때 클러스터의 생성과 그에 대한 응집도 및 클러스터간의 유사도 산출은 다음과 같다.

① C = {C1 = {t1, t2, t3, t4}, C2 = {t5, t6} }일 경우
C1에 대한 주요항목 집합은 적어도 C1에 3개(60%×4)의 문서에 포함되어 있어야 하므로 C1(L) = {1, 3}, C1(S) = {2, 6, 4, 5}이고 C2(L) = {7, 8}, C2(S) = {5, 6}이다. 그리고 이 에 대한 응집도는 c(C1, C2) = (2/6 + 2/4) / 2 = 0.41이고 클러스터간의 유사도는 n_s(C1, C2) = (0/4 + 2/6) / 2 = 0.16이다.

② C = {C1 = {t1, t2}, C2 = {t3, t4}, C3 = {t5, t6}}일 경우
C1에 대해 주요항목의 집합은 적어도 C1에 2개(60% × 2)의 문서에 포함되어 있어야 하므로 C1(L) = {1, 2, 3}, C1(S) = {6}이고 C2(L) = {3, 4, 5}, C2(S) = {1}, C3(L) = {7, 8}, C3(S) = {5, 6}이다. 그리고 클러스터의 응집도는 c(C1, C2, C3) = (3/4 + 3/4 + 2/4) / 3 = 0.66이고 클러스터간의 유사도는 n_s (C1, C2, C3) = (1/8 + 1/4) / 3 = 0.12이다.

산출된 클러스터의 응집도와 클러스터간의 유사도를 비교하면 ②가 ①보다 더 좋은 결과를 나타내므로 ②와 같은 방법으로 클러스터링되는 것이 바람직하며 이것은 클러스터의 할당을 위해 클러스터의 응집도와 클러스터간의 유사도를 근거로 하는 비용에 의한 클러스터링 결과와도 같게 나타난다.

5. 실험 및 분석

이 장에서는 이 논문에서 제안하는 알고리즘에 대한 클러스터의 정확성을 측정하기 위해 기존 알고리즘 [12]와의 비교에 대한 실험을 기술한다.

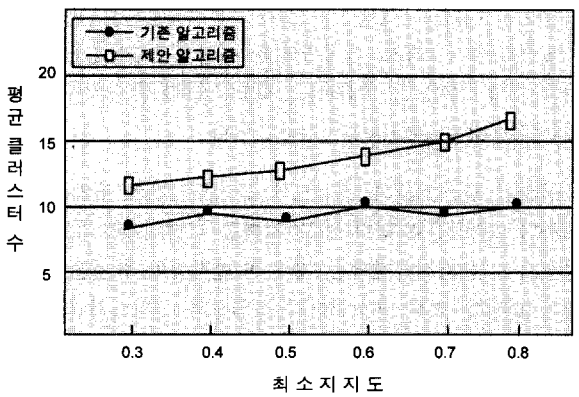
실험에 사용된 문서는 실제 웹 상에 XML 문서가 많이 존재하지 않으므로 분야별 검색을 지원하는 야후(yahoo)에서 5개 분야 즉, 학교(학교소개, 특별활동, 입시안내, 학과안내, 도서자료 문서 등), 음악(장르소개, 가요제, 악기 사용법, 연주 및 대회안내 문서 등), 스포츠(스포츠 종류와 특성, 경기방식소개, 경기장 안내, 선수 소개, 스포츠뉴스 등), 문화(전통문화 소개, 지역별 문화, 사회 단체, 예술인 자료 등), 정치(정치인, 뉴스, 단체 및 학회, 정당 소개문서 등)분야의 관련문서에서 총 100개의 HTML 형식 문서를 수집하였다. 그리고 각 문서에서 자주 사용되는 텍스트를 중요한 의미 단어로 간주하여 엘리먼트화하고 XML문서로 변형하였다. 유사한 의미를 갖는 텍스트는 같은 엘리먼트로 구성하여 엘리먼트 매핑이 이루어지도록 하였다.

유사 구조에 기반하는 클러스터링을 수행하기 위한 전제과정인 구조 추출은 앞의 3장에서 설명된 순차 패턴 알고리즘을 사용하였고, 최소 지지도는 2로 하여 빈발 패턴 구조를 추출하였다. 추출된 구조에서 중복되지 않는 최대 빈발구조 길이의 80% 이상의 구조 길이를 클러스터링을 위한 기초 입력 자료로 하였다.

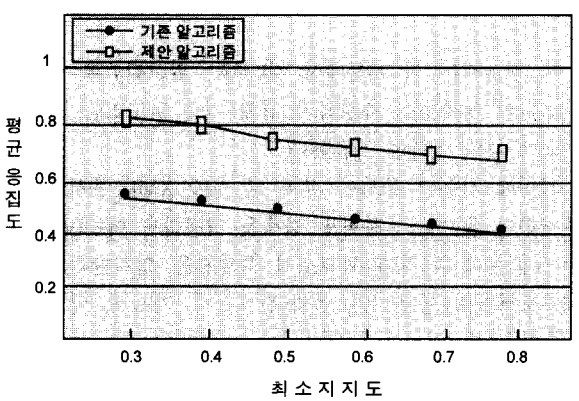
(그림 3)은 클러스터링을 위한 최소 지지도의 변화에 따라 생성되는 클러스터의 수에 대한 변화를 나타낸다.

제안 알고리즘은 최소 지지도가 높을수록 점차로 생성되

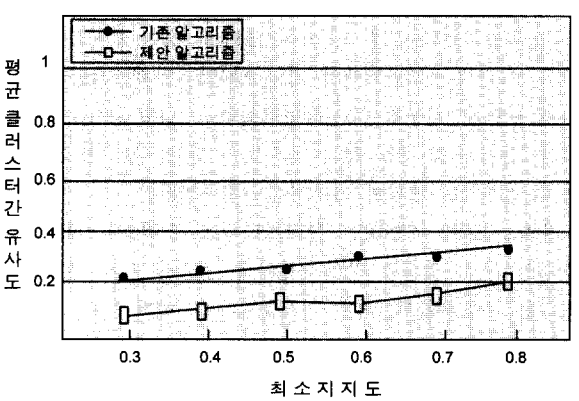
는 평균 클러스터의 수가 증가하는 것을 알 수 있다. 이것은 지지도가 커질수록 클러스터내에 주요항목을 포함하는 문서의 수가 많아야 좋은 클러스터를 유지할 수 있는 비용의 조건에 의해서 생성되는 결과이다. 그러나 기존 알고리즘은 지지도에 대해 생성되는 클러스터의 수에는 많은 변화를 보이지 않았고 생성되는 클러스터의 수도 제안 알고리즘보다 평균적으로 적게 나타났다.



(그림 3) 평균 클러스터 수



(a) 클러스터 응집도



(b) 클러스터간의 유사도

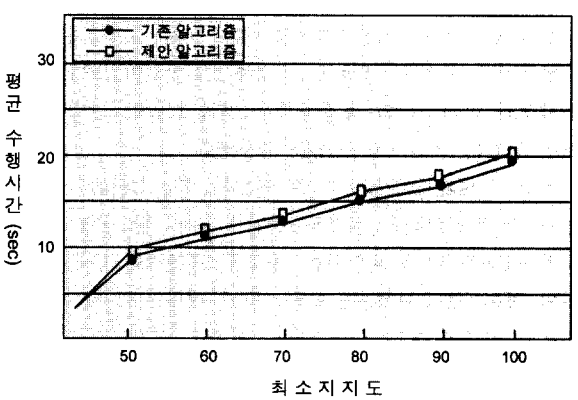
(그림 4) 클러스터 응집도와 클러스터간의 유사도

생성되는 클러스터의 수와 클러스터의 정확도를 나타내는 응집도 및 클러스터간의 유사도와의 관계를 알아보기 위하여, 지지도에 따라 변화되는 클러스터의 응집도(a)와 클러스터간의 유사도(b)의 차이를 비교한 것이 (그림 4)이다.

클러스터의 응집도를 나타내는 (그림 4)(a)를 보면 제안 알고리즘이 기존 알고리즘보다 평균적으로 높은 응집도의 분포를 나타낸다. 그리고 (그림 4)(b)의 클러스터간의 유사도에서도 기존 알고리즘보다 더 낮은 결과를 보이므로 질적으로 더 좋은 클러스터가 생성되는 것임을 확인할 수 있다.

또한 (그림 3)의 지지도 변화에 따라 생성되는 평균 클러스터의 수와 (그림 4)의 응집도 및 클러스터간의 유사도 관계를 살펴보면 지지도가 높을수록 응집도는 조금씩 낮아지고 클러스터간의 유사도는 더 높아지는 것을 알 수 있다. 이것은 지지도가 높을수록 클러스터를 유지하기 위하여 주요항목을 포함하는 문서를 더 많이 요구하므로 적정 크기 이상의 클러스터를 생성하게 되는 것으로 판단된다. 따라서 적절한 지지도가 주어졌을 때 전체적으로 양질의 클러스터를 생성할 수 있음을 알 수 있다.

(그림 5)는 기존 알고리즘과 비교하여 문서의 수 변화에 따른 클러스터링의 평균 수행시간을 비교한 것이다.



(그림 5) 수행시간

문서의 수가 증가할수록 점차로 기존 알고리즘에 비해 작은 차이지만 시간이 더 소요되는 것을 볼 수 있다. 이 시간의 차이는 기존 알고리즘을 이용하여 생성되는 클러스터의 수가 제안 알고리즘을 적용했을 때보다 평균적으로 적게 생성되므로 클러스터의 정제단계에서 클러스터의 이동여부를 비교해야 하는 클러스터의 수도 적기 때문에 수행시간에 차이가 나는 것임을 확인할 수 있었다. 그러나 클러스터의 수행시간에 대한 작은 차이보다는 (그림 4)에서 확인했던 것처럼 제안 알고리즘의 수행에 의한 클러스터의 응집도와 클러스터간의 유사도에서 더 많은 차이를 나타내므로 클러스터링의 정확도에서 보다 좋은 결과를 보인다.

위의 실험 결과와 더불어 생성된 클러스터를 살펴볼 때 각 분야별 문서에서 같은 엘리먼트를 사용하게 되는 공통의 엘리먼트 즉, 이름, 특징, 종류, 방법, 연도 등과 같이 분야

에 관계없이 자주 사용되는 엘리먼트들을 많이 사용한 문서들은 다른 분야의 문서일지라도 같은 분야의 문서로 분류되는 것을 볼 수 있었다. 그러나 문서의 구조 추출과정에서 다른 분야와 구별되는 분야별 특정 엘리먼트를 사용하게 되면 문서 분류가 다르게 구분되었다. 이것은 엘리먼트를 어떤 단어로 선택하여 XML 문서를 구성했느냐가 유사 구조 문서의 분류에 많은 영향을 미친다는 것을 알 수 있다.

6. 결 론

이 논문에서는 다양한 구조를 가지는 XML 문서들에 대해 구조적 유사성을 갖는 문서의 분류 및 저장, 그리고 사용자에게 효과적인 문서검색의 결과를 제공할 수 있는 XML 문서의 유사 구조를 기반으로 하는 클러스터링을 제안하였다. 이 제안 기법은 XML 문서를 구성하는 엘리먼트의 순서와 발생 빈도를 동시에 고려할 수 있는 순차패턴을 이용하여 일정한 지지도를 만족하는 빈발 구조 패턴을 추출하고 추출된 구조를 기준으로 주요항목 기반의 클러스터링 기법을 적용하여 유사 구조의 문서를 그룹화하였다.

이 논문에서 제안하는 클러스터링 알고리즘은 기존 연구에서 단지 주요항목과 비주요항목만을 가지고 비용을 측정하는 방식을 개선하여 각 문서에 포함되어 있는 빈발 구조를 대상으로 클러스터의 응집도와 클러스터간의 유사도를 동시에 고려하기 위해 유도된 비용의 비교를 통해 가장 높은 비용의 클러스터에 문서를 할당한다. 이것은 기존 연구와의 비교를 위한 실험에서 알 수 있듯이 기존 알고리즘에 비해 더 높은 클러스터의 응집도와 더 낮은 클러스터간의 유사도의 결과를 얻을 수 있음을 확인할 수 있었다.

이 연구는 문서 내용에 대한 기본 정보와 문서의 특성을 구조로 표현하는 XML 문서의 엘리먼트 구조를 기반으로 하므로 구조 중심의 문서 분류에 사용될 수 있으며, 문서의 전체적인 형식에 의한 분류, 문서의 주요 핵심 단어에 의한 분류 및 문서의 병합 여부를 위한 구조 검색과 유사 구조의 저장 관리에 효율적으로 적용할 수 있다.

향후 연구로는 XML 문서에서 엘리먼트의 의미적 유사 매칭을 통한 구조 추출 기법의 연구와 대량의 XML 문서를 대상으로 하는 실험의 보완이 필요하다.

참 고 문 헌

[1] W3C, Extensible Markup Language(XML) 1.1., http://www.w3.org/TR/xml11, W3C Working Draft. April, 2002.

[2] Natanya Pitts, editor, "XML Black Book 2nd Edition," Young-Jin, 2001.

[3] P. Kotasek, J. Zendulka, "An XML Framework Proposal for Knowledge Discovery in Database," The Fourth European Conference on Principles and Practice Knowledge Discovery in Databases, 2000.

[4] K. Wang, H. Liu, "Discovery Typical Structures of Documents : A Road Map Approach," In ACM SIGIR Conference on Information Retrieval, 1998.

[5] A. P. Asirvatham, K. K. Ravi, "Web Page Classification based on Document Structure," In IEEE Conference, 2001.

[6] J. T. Wang, D. Shasha, G. J. S. Chang, "Structural Matching and Discovery in Document Databases," In ACM SIGMOD Conference, 1997.

[7] J. Wison, "Data Management for XML : Research Directions," IEEE Computer Society Technical Commitee on Data Engineering, 1999.

[8] R. Nayak, R. Witt, A. Tonev, "Data Mining and XML Documents," International Conference on Internet Computing, 2002.

[9] T. Asai, K. Abe, S. Kawasoe, H. Arimura, H. Sakamoto, "Efficient Substructure Discovery from Large Semi-structured Data," SIAM on Data Mining, 2002.

[10] J. W. Lee, K. Lee, W. Kim, "Preparation for Semantics-Based XML Mining," IEEE International Conference on Data Mining (ICDM), 2001.

[11] J. Pei, J. Han, B. M. Asi, H. Pinto, "PrefixSpan : Mining Sequenctial Pattern Efficiently by Prefix-Projected Pattern Growth," Int, Conf. Data Engineering (ICDE), 2001.

[12] K. Wang, C. Xu, "Clustering Transactions Using Large Items," In Proc. of ACM CIKM-99, 1999.

[13] S. Nestorov, S. Abiteboul, R. Motwani, "Extracting Schema from Semistructured Data," In Proc. of SIGMOD conference, 1998.

[14] C. H. Moh, E. P. Lim, W. K. Ng, "DTD-Miner : ATool for Mining DTD from XML Document," Int, Workshop on Advance Issues of E-Commerce and Web-Based Information Systems (WECWIS), 2000.

[15] S. J. DeRose, "XQUERY : a unified syntax for linking and querying general XML," In Proceeding. Query Languages workshop (QL '98), Boston, Mass., December, 1998.

[16] J. W. Lee, K. H. Lee, "Methodology for Identifying XML-based Target Document for EDMS," Korean Database Conference (KDBC), 2002.

[17] R. Srikant, R. Agrawal, "Mining Sequential Patterns : Generalizations and Performance Improvements," The 5th International Conference on Extending Database Technology (EDBT), Avognon, France, March, 1996.

[18] A. G. Buchner, M. Baumgarten, M. D. Mulvenna, R. Bohm, S. S. Anand, "Data Mining and XML : Current and Future Issues," WISE, 2000.

[19] A. Deutsch, M. F. Fernandez, D. Suciu, "Storing Semi-structured Data with STORED," In proceedings of ACM SIGMOD International Conference on Management of Data, Philadelphia, USA, pp.431-442, 1999.

[20] A. Doucet, H. A. Myka, "Naive Clustering of a Large XML

Document Coolection," The Proceedings of the 1st INEX, Germany, 2002.

[21] M. Steinbach, G. Karypis, V. Kumar, "A Comparison of Document Clustering Techniques," Technical Report of Department of Computer Science and Engineering, University of Minnesota, 2000.

황 정 희

e-mail : jhhwang@dblab.chungbuk.ac.kr

1991년 충북대학교 전산통계학과(이학사)

2001년 충북대학교 대학원 전자계산학과(이학석사)

2001년~현재 충북대학교 대학원 전자계산학과 박사과정

관심분야 : XML, 데이터 마이닝, 능동 데이터베이스, 시공간 데이터베이스

류 근 호

e-mail : khryu@dblab.chungbuk.ac.kr

1976년 숭실대학교 전산학과(이학사)

1980년 연세대학교 공업대학원 전산전공(공학석사)

1988년 연세대학교 대학원 전산전공(공학박사)

1976년~1986년 육군군수 지원사 전산실(ROTC 장교), 한국전자통신연구원(연구원), 한국방송통신대 전산학과(조교수) 근무

1989년~1991년 Univ. of Arizona Research Staff(TempIS 연구원, Temporal DB)

1986년~현재 충북대학교 전기전자컴퓨터공학부 교수

관심분야 : 시간 데이터베이스, 시공간 데이터베이스, Temporal GIS 및 지식기반 정보검색 시스템, 데이터 마이닝 및 데이터베이스 보안, 바이오 인포메틱스