도서 메타데이터와 데이터베이스를 이용한 성능 측정

이은경, 박제호
단국대학교 전자계산학과
e-mail : aboothreede,dk_jhpark@dankook.ac.kr

Performance Evaluation for Library System utilizing Metadata and Database

Eun-Kyung Lee, Je-Ho Park
Dept. of Computer Science, Dan-Kook University

요 약

인터넷 기반으로 다양한 웹서비스와 멀티미디어형 자료를 제공하는 디지털 도서관의 설립과 활용도가 높아지면서, 기존의 도서 서비스 시스템에서 사용하던 기술은 개선을 요구하고 있다. 기존 도서시스템에서 활용하고 있는 국제 표준 메타데이터인 MARC 와 MODS 는 급속히 발전하는 인터넷 기반 디지털 도서관의 서비스를 만족스럽게 충족시키기에는 개선이 필요하다. 본 논문에서는 도서시스템의 메타데이터 스키마를 설계하고, 검색 시간 향상을 위한 방법을 제안하고, 실용성 검증을 위한 실험결과를 보였다.

1. 서론

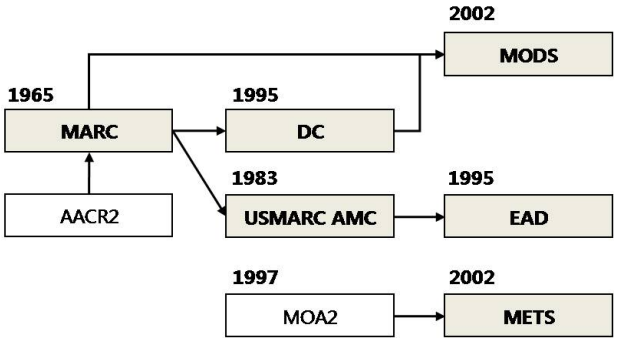
지식정보사회에서 정보 이용자는 다양한 유형의 디지털 매체 자료정보를 검색한다. 기존의 도서시스템에 적용하고 있는 도서 목록 국제 표준 메타데이터 형태는 급속히 발전하는 인터넷 기반 디지털 도서관에서 등장한 웹 자원 및 멀티미디어 자료를 지원하기에는 비효과적이다. 메타데이터는 데이터에 대한 데이터로 정의되며, 일반적으로 다양한 업무를 지원하기 위하여 사용되는 정보자원에 대한 구조화된 데이터를 의미한다. 메타데이터란 용어는 전자정보와 관련하여 처음 사용되었으며, 점차 그 응용이 확장되어 정보자원에 대한 모든 표준화된 기술정보를 포괄하는 의미로 확장되었다[1].

디지털 도서시스템은 지리적 시스템환경 측면에서 분산되어 있고, 서로 성질이 다른 원천 정보들에 대한 효율적인 검색을 제공하는 시스템이다. 현재 디지털 도서관 시스템들은 다양한 형태와 구조를 적용하여 서비스를 지원하고 있다. 디지털 도서관 시스템은 개념적 측면에서 사용자와 정보 원천을 어떻게 이어줄 것인가 라는 물음에 대한 해답의 역할을 하게 된다. 일반적으로 디지털 도서시스템은 서비스 사용자(user)와 서비스의 목적 혹은 결과물인 디지털 객체(digital object), 그리고 이들을 연결시켜주는 바인딩(binding)으로 구성된다. 바인딩은 사용자와 디지털 객체간에 의미적 불일치와 형식적 불일치에 발생하는 문제를 해결하는 접근방법이다[2]. 의미적 불일치는 사용자의 질의와 의미 사이에 발생하는 불일치를 의미하고, 형식적 불일치는 사용자가 요구하는 디지털 객체의 형식과 실제 디지털 객체의 형식간의 차이를 의미한다. 서로 다른 디지털 객체에 사용자가 하나의 질의로 같은 결과물을 얻기 위해서는 의미적 불일치와 형식적 불일치를 해결해야 한다.

기존의 도서시스템은 디지털 객체의 모든 필드를 검색하여 결과물을 산출하였으나, 검색할 수 있는 모든 필드를 검색하는 것 보다 관련성의 확률이 높은 일부 필드만을 검색하는 것이 검색 시간의 단축을 통해 전체 시스템의 성능 향상 시킬 것이다. 본 논문은 도서시스템의 메타데이터 포맷인 MARC 와 MODS 를 비교 분석하여 적합한 메타데이터 스키마를 제안하고, 서로 다른 두 개의 디지털 객체에 맞게 사용자의 질의를 변환 시키는 방법을 제안한다. 또한 효율성 검증을 위해 실시한 실험 결과를 예시한다.

2. 연구배경

문헌정보 메타데이터 발전과정은 그림 1 과 같다. 1965 년 MARC 를 시작으로 여러 메타데이터에 대한 표준이 발표되었다.



(그림 1) 문헌정보 메타데이터 발전 과정

2.1 메타데이터의 상호운용성

분산된 메타데이터를 수집하여 통합된 정보체계를 구축 할 때, 메타데이터 사이의 상호변환의 필요성에 대한 판단을 위해 특정 메타데이터가 가지는 정보 포용력에 대한 이해는 필수적이다. 상호운용성은 특정 환경에서 생긴 정보를 고도의 자동화된 방법으로 체계화하고 통합하여 이질적 환경에서의 사용이 가능한 시스템의 재구축을 의미한다. 다양하고 상이한 목적과 용도를 위해 제안된 메타데이터가 지속적으로 증가함에 따라 이질적인 메타데이터간 상호운용성에 대한 중요성이 점차 확대되어 가고 있다. 현재 메타데이터들은 각각 독립적으로 개발되고, 개발환경에 따라 전문적인 용어와 방법을 사용하고 있어 상이한 메타데이터들 간의 상호연동을 위해 매개 시스템은 필수적이다. 이러한 이유로 NISO 에서는 상호연동을 위한 원칙들을 9 가지로 정의하고 있다. 9 가지 원칙은 조화(harmonization), 공동용어(common terminology), 메타데이터 요소(element)들의 특성(properties), 구성(organization), 과정(process), 의미론적 매핑(semantic mappings), 요소간의 매핑(element to element mapping), 계층구조(hierarchy), 컨텐츠 변환(content conversation)이다[3].

2.2 메타데이터 표준

MARC(Machine-Readable Cateloguing) 는 서지관련 정보의 표현과 교환을 위한 표준 형식으로 개발되었다[4]. 초기 목적은 자기테이프 상에서 서지데이터 전송형식을 지원하고 인쇄목록의 합리적인 관리이었다. MARC 형식을 이용한 문헌 정보의 컴퓨터처리 유용성이 입증됨에 따라 서지용 포맷을 비롯하여 전거용 포맷, 소장용 및 분류용 포맷 등으로 도서관 업무에 적합한 여러 형식이 개발되었다. 인터넷 확산과 더불어 네트워크 정보자원에 대한 서지정보 기술의 필요성이 증대됨에 따라 USMARC 을 개정한 MARC21 표준안이 제정되었다. MARC21 은 단행본을 비롯한 연속간행물, 전자자료, 지도, 악보, 영화, 음성/비디오 레코딩 등의 다양한 형태의 정보자원에 대하 서지정보를 종합적으로 표현할 수 있도록 설계되었다[5].

MARC21 은 5 종류의 계열 표준안으로 구성되어 있으며, 각 형식의 MARC 레코드는 다음과 같이 3 가지 주요 요소로 구분된다. 리더(leader), 디렉토리(directory), 가변 필드(variable field) 주요 요소와 01X-8XX 형식의 태그를 갖는 가변 데이터 필드는 지시자(indicator), 서브필드(subfield)의 2 종류로 구성된다. MARC 형식은 기본적으로 정보의 고정된 모델링을 고려하지 않고 필요에 따라 정보 요소를 추가, 보완하고 있기 때문에 정보 요소간의 형식적인 체계성을 기술하지 못한다[6]. 또한, MARC 의 구조는 매우 복잡하여 이에 대한 전문지식 없이는 작성이 어렵기 때문에 정보의 생산과 갱신이 끊임 없이 일어나고 있는 웹과 같은 환경에서 정보자원의 생산자가 그 내용을 직접 기술할 수 있다는 문제점을 지닌다.

MODS(Metadata Object Description Schema)는 도서관 영역의 디지털 자원의 서지정보 표준 메타데이터로서, 디지털 도서관의 메타데이터 표준 환경을 제공한다. Dublin Core 의 단순함과 MARC 의 복잡함을 절충하기 위해 개발되었으며, MARC21 의 서지정보요소를 기반으로 한 메타데이터 형식의 표준안이다. XML 기반으로, 서지정보 요소간의 계층구조와 의미를 표현 할 수 있고 구조에 확장성과 유연성이 있으며, XML 응용환경을 그대로 활용할 수 있다[7]. MODS 에서는 <titleInfo>, <classification> 등 19 개의 상위 메타데이터 요소를 정의하고 있으며[8], 이 요소들은 하위요소 및 속성과 함께 사용된다. 모든 요소와 속성은 선택사항이고, 속성이 반드시 순서대로 와야 한다거나 반복되는 것은 아니다.

2.3 메타데이터 필드 별 비교

도서용 메타데이터를 설계하기 위하여 MARC 와 MODS 을 요소 별로 비교하여 표 1 에 정리하였다.

<표 1> MARC 와 MODS 비교

한글명	MARC	MODS
서명사항	245 $a, $b, $n, $p 등	titleInfo
저자사항	100, 700 $a, $u, 110, 710 $a	name
자료유형	006/00 전자자료 / 008 leader 부의 22, 23 column	typeOfResource
장르	655 $a	genre
발행사항	250 $a, 260 $a, $b, $c, 265$a 773 $t, $a, $b, $d, $g, $k, $p, $r, $x 등	originInfo
언어	008 leader 부의 35-37 column	language
형태사항	300 $a, $b, $c 등	physicalDescription
초록	520 $a	abstract
목차	505 $a	tableOfContents
이용대상	521 $a	targetAudience
주기	500 $a, 502 $a, 506 $a, 510 $a, $c, 513 $a, $b 등	note
주제	650 $a, 653 $a	subject
분류	050 $a, 080 $a	classification
관련정보	440 $a, 490 $a 등	relatedItem
식별기호	538 $a, 856 $a	identifier
소장위치	049 $a, $i 등	location
이용제한	506 $a	accessCondition
로컬정보	900 $a	extension
레코드정보	040 $a, $b, $c 등	recordInfo

MODS 는 다른 메타데이터를 수용할 수 있는 장점을 지녔지만, MODS 와 MARC 의 데이터 손실 및 매핑 불일치로 인하여 불가능하다는 한계점이 있다. 메타데이터 스키마의 작성 조건은 다면적으로 복잡한 설계를 적용할 경우에는 자료의 표현이 유리한 반면 일발적인 접근 용이성을 허용하기 위해서는 메타데이터 스키마는 단순화를 필요로 한다.

3. 상호운용성 실험을 위한 환경

3.1 메타데이터 스키마

이질적 메타데이터의 상호운용성 연구를 위해 실제로 사용되고 있는 도서시스템의 주요 요소를 중심으로 메타데이터 스키마를 설계하고, 시뮬레이션 기법을 이용하여 비교 측정할 두 개의 데이터베이스에 메타데이터 스키마를 구성하였다. 하나의 데이터베이스에는 기본 정보만으로 구성된 스키마를 적용하고, 다른 데이터베이스에는 필드를 추가한 스키마를 적용하여 기본적으로 불일치성을 유도하였다.

<표 2> 데이터베이스 1 스키마

Field	Data Type	unique	not null	설명
B_Name	text	●	●	책 제목
Author1	varchar(50)		●	저자 1
Author2	varchar(50)			저자 2
Publisher	varchar(50)		●	출판사
ContentsOfABook	text			책소개
Author_1	varchar(50)		●	저자 1
A1_Contents	text			저자 1 소개
Author_2	varchar(50)			저자 2
A2_Contensts	text			저자 2 소개
Contents	text			목차

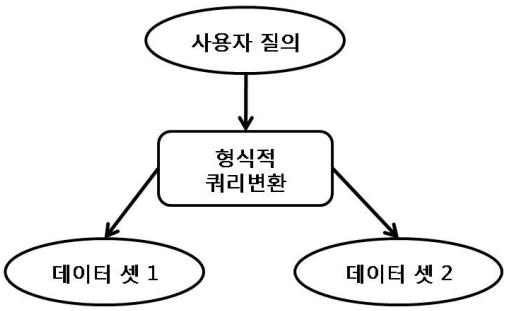
<표 3> 데이터베이스 2 스키마

Field	Data Type	unique	not null	설명
B_Name	text	●	●	책 제목
Author1	varchar(50)		●	저자 1
Author2	varchar(50)			저자 2
Publisher	varchar(50)		●	출판사
ContentsOfABook	text			책소개
Author_1	varchar(50)		●	저자 1
A1_Contents	text			저자 1 소개
Author_2	varchar(50)			저자 2
A2_Contensts	text			저자 2 소개
Contents	text			목차
Publisher_Contents	text			출판사 제공 책소개

두 스키마의 공통된 필드인 B_Name 은 책 제목을 담고 있는 필드로 데이터 값이 유일해야 하며 null 값을 허용하지 않는다. Author1 과 Author2 는 저자에 관한 정보를 담고 있는 필드로 저자가 2 명 이하인 책으로만 데이터베이스를 구성한다. Author1 필드에는 null 값이 허용되지 않는다. Publisher 는 출판사에 관한 정보를 담고 있는 필드이며 null 값을 허용하지 않는다. ContentsOfABook 은 책 소개에 관한 정보를 담고 있는 필드이다. Author_1 과 Author_2 는 저자소개를 위한 저자이름만을 위한 필드이고 Author_1 은 null 값이 허용되지 않는다. A1_Contents 는 Author_1 필드의 저자소개를 담고 있고 A2_Contents 는 Author_2 필드의 저자소개를 담고 있는 필드이다. Contents 필드는 책의 목차를 담고 있다. 데이터베이스 2 에만 적용된 Publisher_Contents 필드는 출판사가 제공하는 책소개를 포함한다. 책의 주요 정보를 필드로 구성한 데이터베이스 1 스키마와 데이터베이스 2 스키마를 각각 <표 2>와 <표 3>에 보였다.

3.2 상호운용성을 위한 제안

기존의 도서 정보 및 판매를 지원하는 서비스는 사용자가 입력한 질의에 포함된 키워드를 기본으로 검색 가능한 모든 필드를 순차적으로 검색한다. 본 논문에서는 제안하고자 하는 방법은 사용자의 입력 질의에 대한 응답을 마련하기 위해 관련성이 있는 필드에서만 검색하도록 하여 검색 효율을 개선하는 방법을 제안한다.



(그림 2) 형식적 쿼리변환

두 데이터베이스의 스키마를 비교하면, 데이터베이스 1 에는 출판사가 제공하는 책소개를 위한 Publisher_Contents 요소가 존재하지 않는다. 하지만 데이터베이스 1 과 2 스키마에 각각 포함되어 있는 책소개 필드(ContentsOfABook)와 데이터베이스 2 의 출판사 제공 책소개(Publisher_Contents)는 모두 도서소개용 정보를 포함하고 있다. 하지만, 동일한 정보의 목적성을 가지고 설정된 각 필드는 유사 분류에 속하는 자료를 포함있지만, 데이터베이스 1 의 ContentsOfABook 이 포함하고 있는 정보 수준은 데이터베이스 2 의 Publisher_ContentsOfABook 의 정보 수준과 일치하지 않는다.

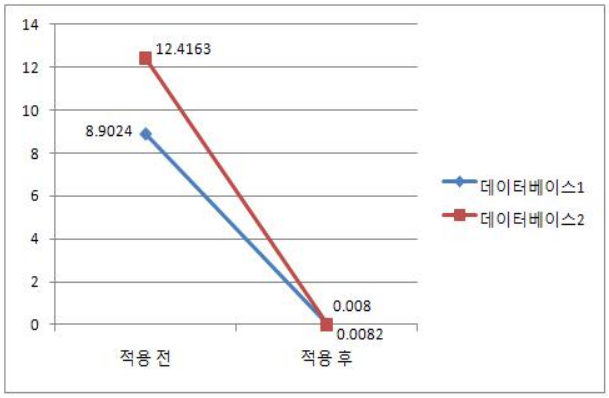
이러한 정보수준의 불일치를 해결하기 위해 그림 2 에서 보는 것과 같이 사용자 쿼리를 메타데이터 스키마의 목적에 부합할 수 있도록 쿼리변환을 수행하여야 한다. 책소개 정보 검색을 위한 사용자 질의가 입

력되면 데이터베이스 1 은 ContentsOfABook 필드를 검색하고 데이터베이스 2 는 ContentsOfABook 필드와 Publisher_ContentsOfABook 필드를 검색한다.

4. 실험 환경과 결과

본 논문에서 구현한 시스템을 실험하기 위해 데이터베이스를 각각 10000 개의 레코드로 구성하였다. 각 레코드는 MySQL 을 사용하여 데이터베이스에 저장하였다. 정확한 실험 결과를 얻기 위하여 두 개의 데이터베이스는 동일한 책에 대한 정보로 구성하였다. 실험용 데이터베이스 구축을 HTML Parser 를 구현하여 도서 URL 을 텍스트 파일에 저장하면, 해당 텍스트 파일을 읽어와 도서 정보를 분석하고 데이터 요소만을 저장하도록 하였다.

HTML 문서를 분석하여 데이터베이스에 저장하고, 제안된 룰에 의한 검색을 실시하였다. 그림 3 은 각 데이터 셋의 응답 시간을 그래프로 나타내었다. 표 4 는 제안된 룰을 적용하지 않았을 때의 평균 검색 응답시간과 적용했을 때의 평균 검색 응답 시간을 보여준다. 검색 성능을 분석하기 위해 성능결과는 초를 사용하였다.



(그림 3) 검색 응답시간

<표 4> 평균 검색 응답 시간

	룰 적용 전 평균 검색 시간 (s)	룰 적용 후 평균 검색 시간 (s)
데이터베이스 1	8.9024	0.0082
데이터베이스 2	12.4163	0.0080

실험결과에서 알 수 있는 바와 같이, 주어진 질의의 결과를 위해 전체 정보량 측면에서 검색하여야 하는 필드가 많은 데이터베이스 2 는 검색시간이 오래 걸리는 것을 볼 수 있다. 하지만, 질의의 의미적 측면을 고려한 검색에서는 거의 동일한 성능을 얻을 수 있었다. 이 실험을 통해 알 수 있는 것은 의미적 분석을 고려한 도서 정보 검색은 기존의 무차별적인 검색의 단점을 개선시킬 수 있다는 것이다.

5. 결론

디지털 콘텐츠가 급격히 증가함에 따라 효과적으로 적합한 문헌을 검색할 수 있게 해주는 메타데이터의 중요성이 점차 확대되어 가고 있다. 특정 정보 자원을 기술하는데 있어서 어느 한 가지의 형식의 메타데이터로 기술하기보다 다양한 유형의 메타데이터들을 이용함으로써 정보 자원의 내용 및 특성을 보다 정확하게 기술할 수 있다. 인터넷을 통한 정보의 검색, 교환, 운영에 메타데이터는 그 핵심에 있으며, 메타데이터 정보의 교환 체계를 통한 상호운용성이 강조되고 있다.

본 논문에서는 도서 검색 시스템 문제점을 발견하고 검색 시간 향상을 위한 방법을 제안하고 이를 적용하였을 때의 검색 시간을 측정하였다. 이를 위해 문헌정보 메타데이터들을 분석하고, 분석한 메타데이터들을 비교하였다. 이를 바탕으로 새로운 메타데이터 스키마를 설계하였다. 도서 정보가 저장되어있는 구조화된 HTML 문서를 HTML Parser 로 파싱하여 두 개의 데이터베이스를 구성하였다. 관련된 필드만을 검색하는 룰을 제안하여 검색 시간을 측정한 결과 검색 시간이 개선된 것을 알 수 있었다. 시스템의 서비스 시간을 감소하여 생기는 효과는 적은 자원으로 유사한 트래픽의 사용자를 지원할 수 있다. 따라서, 제안하는 방법론은 검색의 정확성을 유지하면서, 성능을 개선시킬 수 있는 방법으로 그 효용성이 있다고 할 수 있다.

참고문헌

[1] 심 경, “메타데이터 통합 방안”, 한국도서관 · 정보학회지, Vol. 34, No. 3, 2003. 9.

[2] 정재헌, 이상구, 이상호, “디지털 도서관 환경에서 지식기반 질의 변환기법을 이용한 검색 에이전트”, 한국정보과학회, Vol. 26, No. 2, 1999. 2.

[3] 윤세진, 오경묵 “메타데이터간 상호운용성을 위한 비교연구”, 한국도서관 · 정보학회지, Vol. 33, No. 2, 2002. 6.

[4] Library of Congress, MARC Standards, <http://www.log.gov/marc/marc.html>

[5] 이현실, 전양승, 한성국, “MARC 의 개념 모델링 연구”, 한국도서관 · 정보학회지, Vol. 36, No. 3, 2005. 9.

[6] Roy Tennant, “A Bibliographic Metadata Infrastructure for the 21st Century, “Library Hi Tech, Vol. 22, No. 2, p.175-181, 2004

[7] Sally H. McCallum, “Library of Congress Metadata Landscape”, Vol. 50, No. 3, p.184, 2003.

[8] Rebecca S. Guenther, “MODS: the metadata object description schema”, Potal: Libraries and the Academy, 2003.