시멘틱 웹/온톨로지 기술을 이용한 개인용 전자문서 검색 시스템

Personal Electronic Document Retrieval System Using Semantic Web/Ontology Technologies

김학래(Kim Hak Lae)*, 김홍기(kim Hong Gee)**

초 록

개인 사용자가 전자문서를 쉽게 사용하려면 전자문서를 효과적으로 분류하고, 정확하게 검색할 수 있는 기능이 필요하다. 그러나 개인 사용자의 컴퓨터에 저장된 문서를 효율적으로 관리하기 위한 방법이나 도구에 대한 연구는 상대적으로 미흡한 상태이다. 본 연구는 개인 사용자가 전자 문서를 효과적으로 관리하고 검색하기 위한 방법을 제안한다. 연구 결과인 ONTALK은 모든 전자문서의 메타데이터를 온톨로지 기반으로 생성하고, 추론엔진(inference engine)을 이용하여 의미적(semantics) 정보 검색을 제공한다.

ABSTRACT

There are many kinds of applications or software components to manage files in a local computer, but it is very difficult to organize personal documents in a consistent way and to search expected ones in a precise way. In this paper, we present our development of a document management and retrieval tool,which is named *Ontalk*. Our system provides a semi-automatic metadata generator and an ontology-based search engine for electronic documents. *Ontalk* can create and import various ontologies in RDFS or OWL for describing the metadata. Our system that is built upon .NET technology is easily communicated with or flexibly plugged into many different programs.

키워드 : 문서관리, 시맨틱 웹, 온톨로지, 정보검색, 추론
Document Management, Semantic Web, Ontology, Information Retrieval, Inference

* Digital Enterprise Research Institute, National University of Ireland, Galway, IDA Business Park, Upper Newcastle, Galway, Ireland

** 서울대학교 치과대학

1. 서 론

전자 문서(Electronic Document)는 정보 처리 시스템에 의하여 전자적 형태로 작성, 송수신 또는 저장된 정보를 말한다. 컴퓨팅 환경의 발전에 따라 전자 문서는 기하급수적으로 증가하고 있고, 개인이나 조직의 의사 결정과 정보 관리를 위한 핵심적인 역할을 하고 있다.

그러나 지금까지 문서의 관리는 데이터를 조직화하고, 쉽게 접근하는데 집중되어 왔다. 다시말해, 문서에 담겨 있는 개념과 내용보다 텍스트를 보다 효과적으로 생성하고, 인쇄하고, 전송할 수 있는 방법에 초점을 맞추어 왔다. 그러나 문서 관리는 단순히 텍스트 처리 수준을 넘어 지식 관리의 측면도 요구하고 있다.

개인의 컴퓨터에 저장되는 문서의 양도 점차 증대되어 이들 정보를 효과적으로 관리할 수 있는 방법이 요구되고 있다. 개인 컴퓨터에 저장된 문서는 사용자의 특정한 목적에 맞는 것들을 저장해 놓기 때문에 웹에 존재하는 문서와는 성격을 구분할 필요가 있다. 즉 개인 사용자의 컴퓨터에 저장된 문서들을 단순히 문서의 집합이기 보다 특정한(specific) 주제와 목적을 지닌 개인화된 지식(personalized knowledge)이라고 할 수 있다.

개인 사용자가 문서를 쉽게 관리하기 위해 효과적으로 문서를 분류할 수 있는 방법과

정확하게 원하는 문서를 검색할 수 있는 기능이 요구된다. 그러나 개인 사용자의 컴퓨터에 저장된 문서를 효율적으로 관리하기 위한 방법이나 애플리케이션에 대한 연구는 상대적으로 미흡한 상태이다. 본 논문에서는 시맨틱 웹과 온톨로지 기술을 이용하여 이러한 문제를 해결하고 있다.

본 논문의 구성은 다음과 같다. 2장에서는 문서 관리와 검색의 문제점을 살펴 본다.

3장에서는 시맨틱 웹과 온톨로지 연구의 이론적 배경과 추론 언어에 대해 살펴본다.

4장에서는 ONTALK의 아키텍처, 핵심 기술과 구현 방법을 설명하고 있다. 마지막 5장에서는 본 연구의 기여점에 대해 살펴본다.

2. 문서 관리와 검색의 문제점

개인 사용자의 컴퓨터에 저장된 전자 문서는 인터넷이나 전자도서관에 존재하는 것과 양적인 측면을 비교할 수 없지만, 개인의 특정 목적에 따라 맞는 "지식"의 특성을 갖고

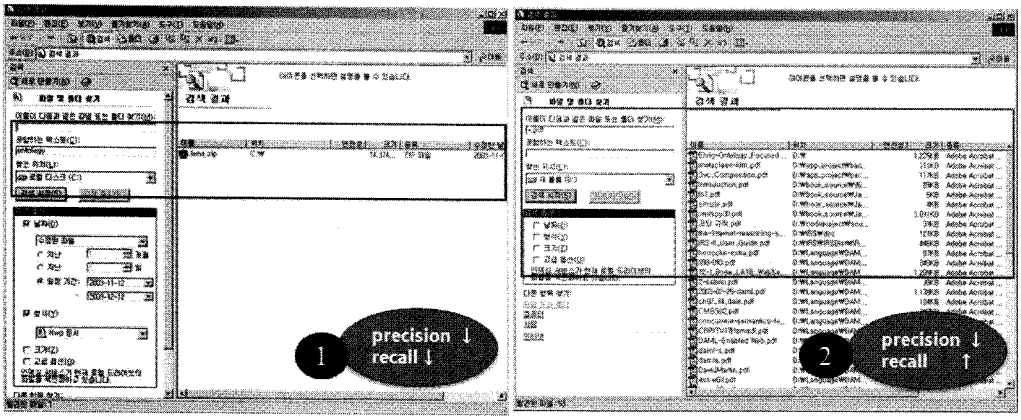
있다. 그러나 개인사용자가 직관에 의존하여 문서를 관리하는 것은 한계가 있다. 일반적으로 개인 사용자는 문서를 개인이 구분한 폴더에 저장하며 그 저장 방식을 기억한다. 그러나 폴더의 구조가 복잡해 지고 문서의 양이 증가했을 때, 이러한 방식은 문서의 관리와 검색의 효율성을 저하시킨다. 개인 사용자가 문서를 사용하기 위해 요구되는 최소의 정보는 물리적(physical) 정보와 문서의 내용(content) 정보로 구분된다. 물리적 정보는 전자문서가 갖고 있는 파일의 형태, 크기와 같은 파일 시스템과 관련된 내용이다. 반면 문서의 내용 정보는 실제 문서가 포함하고 있는 문서의 제목, 저자, 키워드와 같은 정보이다. 문서 관리 및 검색의 문제점은 〈표 1〉과 같이

〈표 1〉 전자문서의 관리와 검색의 문제점

관리 방법	문 제 점
물리적 정보 기반의 문서 검색	· 파일 혹은 폴더명을 이용하여 검색 · 검색 옵션이 물리적 정보에 국한 · 문서 컨텐츠에 대한 검색 불가 · 파일명의 검색이 패턴 매칭에 의존적
디렉토리에 한정적인 문서 관리	· 사용자에 따라 디렉토리 관리 방법에 편차가 큼 · 문서 분류의 효율성 저하 · 디렉토리 구조가 복잡할 경우 · 새로운 디렉토리의 중복/생성이 빈번함
내용 정보	· 문서의 내용을 구분할 수 있는 도메인 정보가 존재하지 않음 · 도메인 정보는 디렉토리에 국한됨 · 문서의 종류, 주제 분류에 따른 복잡성이 증가할 때 구분하기 어려움

요약할 수 있다. 대부분의 개인 사용자는 전자문서를 문서 내용보다 물리적 정보 기반으로 관리하고 검색하고 있다.

특히 이러한 문제들은 사용자가 문서를 검색할 때 더욱 심각해 진다. 〈그림 1-①〉은 '포함하는 텍스트'에 "ontology"를 이용할 때 결과를 보여주고 있다. 이 경우 반환된 결과는 단지 1건(실제 문서는 35개)에 불과하다. 〈그림 1-②〉는 "*.pdf" 옵션을 이용한 경우 이전과 다르게 많은 결과를 보여주고 있다. 두 가지 검색의 결과는 사용자에게 만족스러운 결과를 주지 못한다. 이러한 문제 때문에 사용자들은 자신의 컴퓨터에 저장된 문서를 재활용하지 못하고, 심지어 대상 문서를 다시 인터넷이나 전자도서관에서 다시 검색하게 된다. 따라서 이를 해결하기 위한 적절한 방법



〈그림 1〉 윈도우에서의 검색

이 요구된다. 본 연구에서는 이러한 문제를 해결하기 위해 메타데이터와 온톨로지 기술을 적용하였다. 또한 문서의 의미적 검색을 위해 온톨로지 기반의 추론을 지원한다.

3. 관련 연구

데시맨틱 웹은 간단히 말하면 '컴퓨터가 정보의 의미를 이해하고(machine-understandable)처리할 수 있는(machine-processable) 웹'이라 할 수 있다[1]. 시맨틱 웹은 기계가 처리할 수 있는 언어로 표현된 정보 공간이며, 형식적이고(formal) 의미적으로(semantically) 연결된 정보의 웹이다. 형식적이라는 것은 기계가 읽고 처리함으로써 문서의 내용을 자동으로 표현할 수 있음을 의미한다.

시맨틱 웹에서 형식적이고 의미적인 정보를 표현하기 위해 W3C[1]는 수년에 걸쳐 온톨로지 언어의 개발에 노력해 왔다. 웹에서 지식을 온톨로지로 표현하는 언어는 RDF(Resource Description Framework)를 시작으로 DAML+OIL로 발전해 왔다. W3C는 웹 온톨로지 언어의 후보 권고안[2] 을 발표함으로써 시맨틱 웹 환경에서 온톨로지를 사용할 수 있는 토대를 마련하였다.

RDF는 W3C에서 발표한 메타데이터의 기술과 교환을 위한 표준이다. RDF는 다양한 메타데이터간의 상호운용을 위해 의미(semantics), 구조(structure) 및 구문(syntax)에 대한 공통적인 규칙을 제공한다. 그러나 속성의 제약과 클래스의 상속 관계를 표현하는데 제한이 있기 때문에 복잡한 온톨로지에 적용하는데 한계가 있다.

웹 온톨로지 언어(OWL[3] : Web Ontology Language)는 웹에서 정보를 표현하고 애플리케이션이 직접 내용을 처리할 수 있도록 설계된 언어이다. OWL은 기계 또는 에이전트가 처리할 수 있는 풍부한 어휘(vocabulary)와 형식적 의미(formal semantics)를 제공한다. 따라서 어떤 용어의 의미와 용어 사이의 관계를 명시적으로 표현할 수 있다.

OWL은 XML, RDF, RDF 스키마보다 풍부한 의미 표현이 가능하기 때문에 웹에서 에이전트가 내용 정보를 쉽게 처리할 수 있다.

RDQL[A. Seaborne et al. 2001]은 RDF 문서를 질의하기 위한 언어로 Jena2[4]에 내장되어 있다. RDQL은 SquishQL[L.Miller,2001]과 rdfDB[R.V. Guha, 2002]에서 파생된 질의 언어로 SQL과 유사한 구문 형태를 제공한다. RDQL은 데이터 지향(data-oriented)의 질의 모델을 제공한다.

*"데이터 지향"*은 단지 모델이 갖고 있는 정보에 대한 질의를 사용하는데 목적이 있을 뿐, 추론을 위한 메커니즘을 제공하지 않는다. 다시 말해, RDQL은 RDF의 그래프 구조를 기반으로 기술된 데이터를 질의할 수 있지만,

---

1) http://www.w3c.org
2) W3C에서 표준을 발표하는 단계를 말한다. 초안→제안→추천 권고안→권고안
3) 웹 온톨로지 언어의 정식 명칭은 WOL이 될 수 있었으나, 개발에 참여한 사람들에 의해 OWL(아울)로 표기하기로 결정하였음.
4) http://jena.sourceforge.net

추론을 위한 방법은 제공하지 않는다.

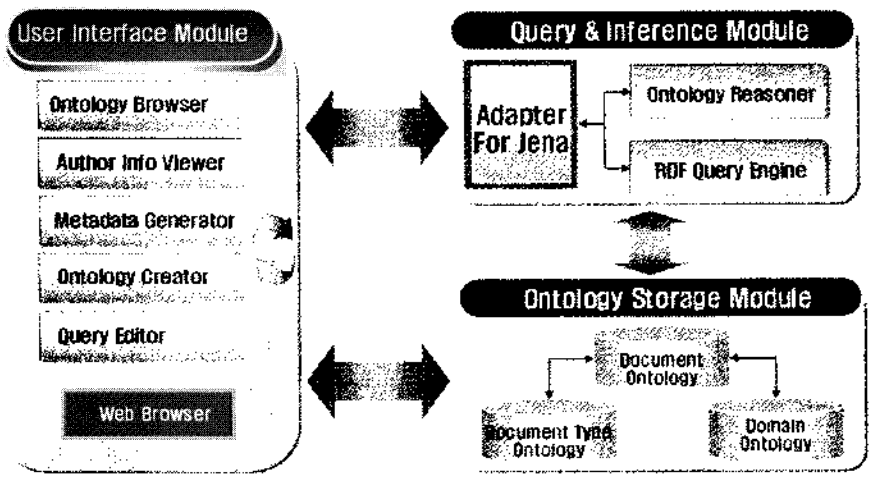
RDQL 기반의 검색은 사용자가 제공한 키워드를 이용하여 명시적(explicit)으로 정의된 형태의 데이터만을 검색한다. 반면 Jena2의 Reasoner API를 이용하여 지식베이스의 클래스(subClassOf) 혹은 특성(subPropertyOf)의 관계성을 검색할 수 있다. 따라서 단순히 키워드의 매칭을 통한 검색보다 관련성 높은 검색 결과를 얻을 수 있다.

4. ONTALK 시스템의 구현

ONTALK은 온톨로지 기반의 문서 관리와 검색 기능을 제공하는 애플리케이션이다. 현재 사용되는 대부분의 전자 문서는 .NET에서 제공하는 *웹 브라우저 컨트롤*을 사용하여 열람(display)이 가능하다. ONTALK은 MS 오피스 형식, PDF 등과 같은 다양한 형식을 열람할 수 있는 기능을 지원한다. 문서의 검색은 Jena2를 이용한다. ONTALK은 RDQL의 패턴 매칭(pattern matching)을 이용한 검색이 가능하며, 온톨로지 기반의 추론도 제공한다.

ONTALK을 구성하는 핵심 구성 요소는 〈그림 2〉와 같이 온톨로지(Ontology) 저장(storage)모듈, 질의 및 추론(query and inference) 모듈, 사용자 인터페이스(interface) 모듈로 구성된다.

시스템 개발 측면에서 ONTALK은 .NET 프레임워크와 자바 기반의 프레임워크를 동시에 사용하고 있다. 즉, ONTALK은 전자 문서의 효율적 열람과 사용자 인터페이스의 편리성을 높이기 위해 .NET 을 기반으로, 온톨로지 기반의 검색을 위해 Jena2를 이용하고 있다. 이러한 프레임워크의 상이함에서 발생하는 문제를 해결하기 위해 ONTALK은 어댑터를 제공하고 있다. 어댑터(Adapter)는



〈그림 2〉 ONTALK 시스템 아키텍처

.NET 기반에서 생성되는 사용자의 요구 사항을 처리하여 자바 기반의 Jena2에 전송하고 처리된 결과를 반환하여 사용자에게 전송하는 역할을 한다.

4.1 온톨로지의 생성

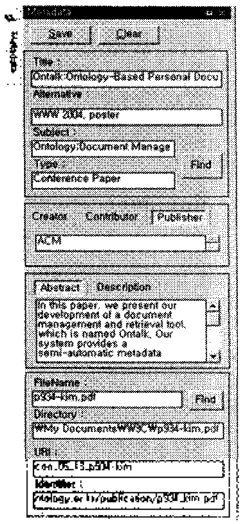
문서형태(Document Type) 온톨로지는 전자 문서를 발행한 목적, 내용, 발간 형태등과 같은 정보와 이들 사이의 관계를 포함하고 있다. 일반적으로 애플리케이션에 종속적인 특성을 갖는 데이터 표현 형식은 문서의 물리적 특성에 관련된 정보만을 정의한다. 즉 사용하는 애플리케이션에 따라 HTML, DOC, PDF 등과 같은 형식으로 표현된다. 그러나 전자문서는 출간 목적에 따라 학술지, 기사, 논문 등과 같이 다양하게 구분될 수 있다. 이것은 사용자가 문서의 종류를 구분하는데 매우 유익한 정보를 제공한다. 또한 문서의 종류를 온톨로지로 구성하여 서로 다른 문서간의 관계성(relationship)을 표현할 수 있다. ONTALK은 문서 종류를 구분할 온톨로지를 내장 형태(built-in)로 제공한다.

ONTALK은 문서의 물리적 정보를 제공하기 위해 더블린 코어 메타데이터의 Format 항목을 이용하고, 문서 형태와 관련된 정보는 documentType 항목을 이용하여 기술한다. 문서의 형식은 클래스의 제약 조건(constraint)에 위배되지 않는다면 복수(multiple)의 값을 허용한다. 그러나 Master Thesis와 Doctoral Thesis와 같이 서로소(disjoint) 관계를 갖는 경우는 복수로 값을 할당할 수 없다. 문서 형태 온톨로지는 문서의 검색을 통해 반환되는 결과의 정확성을 높이는데 중요한 요소이다. 사용자는 문서 형태 온톨로지를 검색 옵션으로 사용함으로써 반환되는 결과의 정확성을 높일 수 있다.

도메인 온톨로지(Domain Ontology)는 특정한 주제를 포함하는 내용을 기술하기 위해 사용된다. ONTALK 은 도메인 온톨로지를 플러그-인(plug-in) 형태로 시스템에 적용할 수 있고 자유롭게 선택할 수 있다. 사용자는 자신이 원하는 온톨로지를 시스템에 적용할 수 있다. 사용자는 온톨로지 생성 템플릿을 이용하여 원하는 형태의 온톨로지를 구성할 수 있으며, 이들 온톨로지는 도메인에 따라 별개의 파일로 존재한다. ONTALK은 기본적으로 온톨로지, 시맨틱 웹(Semantic Web), 도서관(Library), 여행(travel) 도메인의 온톨로지를 제공한다.

4.2 인스턴스의 생성

임의의 문서 파일은 하나의 인스턴스로 만들어져 RDF 파일에 저장된다. 인스턴스 생성은 사용자가 선택한 문서에 있는 메타데이터를 추출하고 저장하는 과정을 말한다. 만약 어떤 문서에 메타데이터 정보가 이미 포함되어 있으면 자동적으로 추출할 수 있다. 메타데이터의 저장은 더블린코어(Dublin Core) 메타데이터를 기반으로 각각의 항목을 저장한다. <그림 3>은 인스턴스를 생성하는 화면을 보여주고 있다. 그림에서 [Input Items]에 포함된 항목은 더블린코어 메타데이터의 항목들이다. 인스턴스를 생성할 때, 사용자는 온톨로지 뷰어, 저자명 뷰어 등과 같은 템플릿




```
<document ID="con_0512_p934">
 <dc:title>Ontalk:Ontology-BasedPersonal Document Management System</dc:title>
 <dcq:alternative>The 14th International World Wide Web Conference Poster</dcq:alternative>
 <dc:subject>ontology</dc:subject>
 <dc:subject>document management</dc:subject>
 <dc:subject>Inference</dc:subject>
 <dc:type>Conference Paper</dc:type>
 <dc:creator>Hak Lae Kim</dc:creator>
 <dc:creator>Hong Gee Kim</dc:creator>
 <dc:creator>Kyung Mo Park</dc:creator>
 <dc:contributor>The Korea Health 21 R&D Project</dc:contributor>
 <dc:publisher>ACM</dc:publisher>
 <dcq:abstract>
  In this paper, we present our development of a document management and retrieval tool ....
 </dcq:abstract>
 <description>Managing electronic data becomes a more challenging task for end users as the
personal data storage capacity increases.....
 </dcq:description>
 <dc:source>C:\Documents and Settings\Hak Lae Kim\My Documents\W3C\C:\Documents and
Settings\Hak Lae Kim\My Documents\W3C\p934-kim.pdf</dc:source>
 <dc:identifier>http://www.ontology.or.kr/publication/p934.pdf</dc:identifier>
</document>
```

[part of the document.owl]

〈그림 3〉 인스턴스 생성

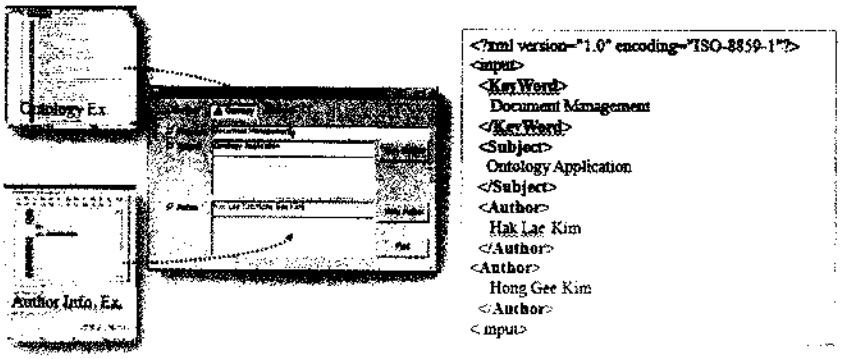
을 이용할 수 있다. 입력된 항목들은 RDF(kb_instance.rdf)나 OWL(kb_instance.owl) 파일로 저장된다.

4.3 RDQL을 이용한 검색

ONTALK은 RDF에 정의된 메타데이터를 기반으로 한 다양한 문서 검색 옵션을 제공한다. 기존의 문서 검색은 개인 컴퓨터에 저장되어 있는 파일의 물리적 형태에 대한 검색에 초점이 맞추어져 있다. 예컨대, 윈도우의 검색 프로그램은 파일명, 날짜, 형태와 같은 물리적 정보를 검색하는데 적당하다. 그러나 사용자가 문서를 검색할 때 필요한 정보는 문서의 물리적 정보보다는 문서의 내용과 관련된 항목들이다. ONTALK은 인스턴스 파일에 기술된 메타데이터를 기반으로 검색 항목을 구성할 수 있다.

- 문서의 제목(title), 저자(author), 초록(abstract) 정보 등
- 문서의 주제(subject), 문서 형태(Document Type) 정보 등

문서의 제목, 저자와 같은 정보는 필터링(filtering)을 통해 정확성을 높일 수 있고, 주제와 문서 형태는 온톨로지에 표현된 클래스의 전이적(transitive) 특성을 이용한 추론에 사용된다. 검색어는 키워드를 기반으로 구성되며 사용자가 키워드를 조합할 때, 논리곱(disjunction)과 논리합(conjunction) 방식을 선택할 수 있다. 풀-텍스트(full-text) 검색은 매체에 따른 검색 속도나 정확성의 편차가 큰 단점이 있다. 일반적으로 위에 열거한 요소들은 실제 문서가 갖고 있는 핵심 정보로 인식된다. 특히 설명 요소는 사용자가 직접 문서의 내용과 관련된 정보를 요약 또는 기술한



〈그림 4〉 검색 기능

```
AND ?title ~=/perspective|knowledge|engineering/i,
      ?abstract ~=/perspective|knowledge|engineering/i

or

AND ?title ~=/^(?=.*perspective)(?=.*knowledge)(?=.*engineering)/i,
?abstract ~=/^(?=.*perspective)(?=.*knowledge)(?=.*engineering)/i
```

〈그림 5〉 정규 표현식을 이용한 AND 절

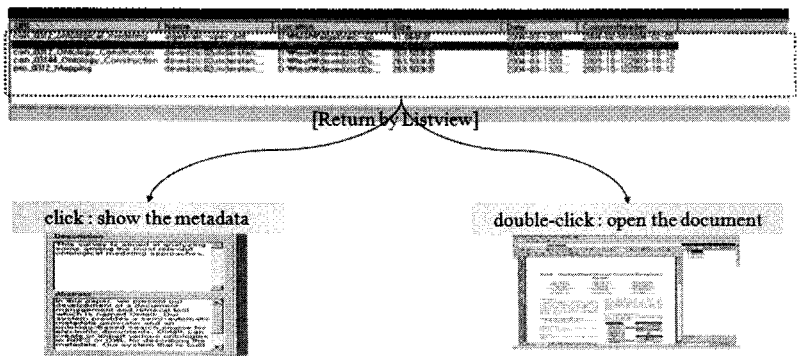
정보이기 때문에 사용자가 입력한 키워드와 연관성이 매우 높다고 할 수 있다.

검색 결과의 정확도를 높이기 위해 ONTALK은 문서의 내용을 정의한 도메인과 문서형태에 대한 정보를 제약할 수 있게 한다 (그림 4 참고). 문서의 주제는 도메인 온톨로지에서 추출할 수 있다. 검색은 키워드, 도메인 온톨로지, 저자명을 조합하여 실행할 수 있다. 예를 들어, 사용자가 지식 공학에 대한 관점을 기술한 문서를 찾기 위해 "perspective", "knowledge engineering"이란 단어를 키워드를 입력했다면, RDQL의 구문은 〈그림 5〉와 같이 구성할 수 있다. ①은 키워드를 논리적 OR 연산으로 결합한 구문이고, ②는 AND 연산으로 처리한 것이다. ②에서 [i]는 대소문자를 구별하지 않기 위해 사용되는 패턴 매칭 변경자(modifier)이다. 예를 들어 knowledge/i는 "knowledge", "KNOWLEDGE","Knowledge" 등과 매치된다. 키워드를 논리적 연산자로 구성하기 위한 옵션은 환경 설정메뉴에서 선택할 수 있다.

그러나 정규 표현을 이용한 패턴 매칭은 기

```
SELECT ?title, ?alternative, ?subject, ?type, ?creator, ?contr
ibutor, ?right, ?source, ?language, ?identifier, ?format
FROM <instance.rdf>
WHERE
      ( ?x, dc:title, ?title),
      ( ?x, dcterms:altenative, ?alternative),
      ( ?x, dc:subject, ?subject),
      ( ?x, dc:type, ?type),
      ( ?x, dc:creator, ?creator),
      ( ?x, dc:contributor, ?contributor),
      ( ?x, dc:right, ?right),
      ( ?x, dc:source, ?source),
      ( ?x, dc:language, ?language),
      ( ?x, dc:identifier, ?identifier),
      ( ?x, dc:format, ?format).
AND ?title =~/Knowledge/i,
USING
dcq for <http://dublincore.org/2000/03/13/dcq#>,
rdf for <http://www.w3.org/1999/02/22-rdf-syntax-ns#>,
vcard for <http://www.w3.org/2001/vcard-rdf/3.0#>,
dc for <http://purl.org/dc/elements/1.1/>
```

〈그림 6〉 검색을 위한 RDQL 구문



〈그림 7〉 검색 실행 결과

본적으로 처리할 수 있는 작업에 한계가 있다. 특히 컨텐츠의 의미를 검색하기 위한 어떤 메커니즘도 제공하고 있지 않기 때문에 온톨로지를 추론하는데 많은 제약이 있다. 이러한 기능상의 한계점을 극복하기 위해 ONTALK은 RDF의 타입 시스템을 기반으로 한 질의 형태를 함께 제공한다. 〈그림 6〉은 ONTALK에서 적용한 RDQL 구분이다. 질의 구문은 인스턴스의 모든 요소를 반환한다. ? title은 사용자가 입력한 키워드의 값을 갖

고 있으며, subject와 documentType은 해당되는 온톨로지의 타입을 검색한다.

〈그림 6〉의 질의를 통해 검색조건에 맞는 인스턴스 즉, 문서들을 찾을 수 있다. 검색결과는 물리적 정보와 컨텐트 정보를 함께 보여줄 수 있다. 〈그림 7〉은 검색 결과의 예를 보여 주고 있는데, 특정한 파일을 선택하면 메타데이터 정보를 보여주거나 문서를 직접보여줄 수 있다.

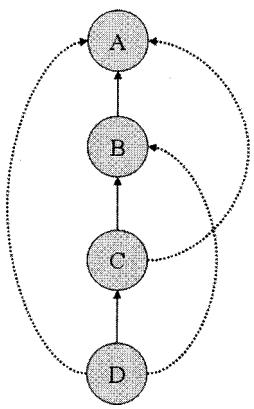
4.4 Jena2를 이용한 추론

RDF는 유향 그래프(directed graph) 구조를 갖고 있으며 그래프에서 각각의 노드(node)는 클래스(class)와 인스턴스(instance), 속성(property)의 값들을 포함한다. 만일 어떤 클래스 x가 클래스 y의 상위 클래스라면 이들 사이에는 서브클래스(subclass) 혹은 수퍼클래스(superclass) 관계가 성립된다. 또한 어떤 객체 z가 클래스 y의 인스턴스라면, 객체 z는 모든 상위클래스의 인스턴스가 된다. 다시 말해 서브 클래스라는 관계성은 전이적(transitive)인 특성이 있다. 위에 언급한 관계를 술어 논리식으로 표현하면 〈그림 8〉과 같다.

이러한 전이적 속성은 인스턴스의 특성을 추론할 때 사용될 수 있다. RDF스키마는 subClassOf와 subPropertyOf를 이용하여 추론 메커니즘을 제공한다. 그러나 위에서 언급한 RDQL은 명시적으로 선언된 데이터 모델에서 질의만을 수행할 뿐 전이적 특성을 추론하는 기능은 제공하지 않는다. 〈그림 9〉는 임

$$\forall x, y, z, \mathrm{subclass}(y, x) \wedge \mathrm{instance}(z, y) \supset \mathrm{instance}(z, x)$$
$$\forall x, y, z, \mathrm{subclass}(y, x) \wedge \mathrm{subclass}(x, z) \supset \mathrm{subclass}(y, z)$$
$$\forall x, y, p, v, \mathrm{instance}(x, y) \wedge \mathrm{property}(y, p, v) \supset \mathrm{property}(x, p, v)$$

〈그림 8〉 클래스-인스턴스 속성의 논리적 표현



〈그림 9〉 클래스의 전이적 관계

의의 클래스 A, B, C, D의 관계를 노드(nodes)와 아크(arcs) 그래프를 사용하여 표현한 예이다.

각각의 클래스는 subClassOf(B, A), subClassOf(C, A), subClassOf(D, C), subClassOf(E, C)의 관계를 갖고 있다. 비록 명시적으로 표현되지는 않았지만 C, D는 A의 서브 클래스임을 알 수 있다. 즉 점선으로 표시된 것은 암묵적으로(implicit) 서브 클래스 관계를 표현하고 있다. 만약 질의 언어를 통해 클래스의 모든 관계를 얻기 위해서는 다음과 같은 질의 구문을 만들어야 한다.

```
SELECT *
WHERE (?x, <rdfs:subClassOf>, ?a),
  (?c, <rdfs:subClassOf>, ?a),
  (?d, <rdfs:subClassOf>, ?c),
  (?e, <rdfs:subClassOf>, ?c)
```

그러나 위의 구문은 모든 서브 클래스 관계

```
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:ok="http://www.ontologykorea.org/">
    <rdfs:Class rdf:about="http://www.ontologykorea.org/B">
        <rdfs:label>B</rdfs:label>
        <rdfs:subClassOf rdf:resource="http://www.ontologykorea.org/B"/>
        <rdfs:subClassOf>
            <rdfs:Class rdf:about="http://www.ontologykorea.org/A"/>
        </rdfs:subClassOf>
        <rdfs:subClassOf>
            <rdf:Description
            rdf:about="http://www.w3.org/2000/01/rdf-schema#Resource">
                <rdfs:subClassOf
                rdf:resource="http://www.w3.org/2000/01/rdf-schema#Resource"/>
            </rdf:Description>
        </rdfs:subClassOf>
    </rdfs:Class>

    <rdfs:Class rdf:about="http://www.ontologykorea.org/A"
    rdfs:label="A">
        <rdfs:subClassOf rdf:resource="http://www.ontologykorea.org/A"/>
        <rdfs:subClassOf
        rdf:resource="http://www.w3.org/2000/01/rdf-schema#Resource"/>
    </rdfs:Class>
    <rdfs:Class rdf:about="http://www.ontologykorea.org/C"
    rdfs:label="C">
        <rdfs:subClassOf rdf:resource="http://www.ontologykorea.org/C"/>
        <rdfs:subClassOf rdf:resource="http://www.ontologykorea.org/B"/>
        <rdfs:subClassOf rdf:resource="http://www.ontologykorea.org/A"/>
        <rdfs:subClassOf
        rdf:resource="http://www.w3.org/2000/01/rdf-schema#Resource"/>
    </rdfs:Class>
</rdf:RDF>
```

〈그림 10〉 Reasoner를 통해 실행된 추론 결과

를 두 번씩 반복해서 질의를 수행한다. 따라서 클래스의 수가 증가하고 구조가 복잡해질수록 질의를 구성하는 것이 어렵다. 뿐만 아니라 이러한 질의 구문의 핵심적 문제는 암묵적(implicit)인 관계를 추론할 수 없다는 단점이 있다.

즉, C, D가 A와 B의 서브 클래스 관계가 있음을 알 수 없다. 지금까지 보듯이 RDQL은 명시적인 데이터 모델을 질의하는데 유용한 반면 클래스의 전이적 특성을 이용한 추론을 하는데 한계가 있다.

ONTALK에서는 검색 수준에 따라 RDQL과 온톨로지 추론 형태의 두 가지 검색 방법을 제공한다. 온톨로지를 이용한 추론 방법은 Jena2에서 제공하는 Reasoner API를 이용하여 구현되었다. 예컨대 RDF 파일을 추론할 때, ModelFactory 인터페이스를 이용하여 새로운 모델을 만들고 추론하게 된다. 〈그림 10〉에서 보듯이 클래스 A는 rdfs:Resource의 서브클래스이고 클래스 B와 C도 서브클래스 관계를 갖고 있다. 〈그림 10〉에 표현된 클래스는 전이적 관계(transitive relationship : A←B←C)의 관계를 갖는다.

5. 결 론

본 논문에서는 전자 문서를 효율적으로 관리하고 검색하기 위해 시맨틱 웹과 온톨로지 기반의 문서 관리 시스템을 제안하였다. 본 연구를 통해 개발된 ONTALK은 전자 문서를 메타데이터로 관리할 수 있는 방법을 제공하고 있다. 문서를 메타데이터로 관리하면 실제문서의 전체 내용을 파악하거나 검색하지 않고도 내용을 이해할 수 있기 때문에 문서관리나 검색의 효율성을 높여 줄 수 있다. ONTALK은 온톨로지 기반으로 메타데이터를 기술함으로써 문서 검색 시 추론을 통한 정확성을 높일 수 있다.

본 연구를 통해 개발된 시스템이 갖는 핵심적인 특징은 다음과 같다. 첫째, 온톨로지 기반으로 메타데이터를 표현함으로써 기술된 데이터 간의 관계를 추론할 수 있는 방법을 제공한다. 둘째, 다양한 검색 형태를 제공함으로써 사용자가 원하는 수준의 검색 결과를 제공한다. 사용자가 패턴 매칭을 통해 검색 결과를 얻을 수 있을 뿐만 아니라 추론 엔진을 통해 암묵적으로 존재하는 자원 간의 관계를 검색할 수 있다.

본 연구는 문서 관리와 검색 분야에 있어 다음과 같은 기여점을 제공한다. 첫째, 다양한 형태로 존재하는 문서들의 비정형성에 관계없이 문서 자원을 메타데이터로 생성할 수 있다. 사용자는 로컬 컴퓨터에 존재하는 모든 형태의 문서, 심지어 이진 파일까지도 메타데이터로 관리할 수 있다. 둘째, 개인화된 문서를 효율적으로 검색할 수 있다. 현재 사용되는 검색시스템은 대부분 키워드를 기반으로 물리적 정보에 국한된 정보 검색 방법을 제공하는 반면 ONTALK은 문서의 내용에 기반한 검색을 지원한다. 셋째, 시스템의 구성이 비교적 간결하여 ERP나 웹 기반의 경영 관리 솔루션과 같은 패키지에 이식이 가능하다. 이 경우 ONTALK은 검색과 문서 관리를 위한 핵심 모듈이 될 수 있다.

참 고 문 헌

[1] 김홍기, 김학래, 이강찬, 전종홍, "시맨틱 웹 기반의 서비스 명세(Service Description)확장", 2003.09

[2] D. McGuinness and F van Harmelen (eds)., "OWL Web Ontology Language Overview", http://www.w3.org/TR/2003/WD-owl-features-20030331/

[3] HP Labs Semantic Web Research . "Jena - A Semantic Web Framework for Java". http://www.hpl.hp.com/semweb/, 2003

[4] A. Seaborne. "RDQL: A Data Oriented Query Language for RDF Models". http://www-uk.hpl.hp.com/people/afs/RDQL/, 2001

[5] ICS-FORTH. The ICS-FORTH RDFSuite web site. http://139.91.183.30:9090/RDF. March 2002.

[6] L. Miller. "RDF Query using SquishQL. http://swordfish.rdfweb.org/rdfquery/, 2001.

[7] D. Brickley, R.V. Guha. "Resource Description Framework Schema (RDF/S) Specification 1.0". W3C Recommendation. March 27, 2000. http://www.w3.org/TR/rdf-schema

[8] O. Lassila, R. Swick. "Resource Description Framework (RDF) Model and Syntax pecification" . W3C Candidate Recommendation. February 1999. http://www.w3.org/TR/REC-rdf-syntax

[9] Tim Berners-Lee, James Hendler, Ora Lassila. "The Semantic Web". Scientific American. May 2001. http://www.sciam.com/2001/0501issue/0501berners-lee.html

[10] M. Dean, G. Schreiber (eds), F. van Harmelen, J. Hendler, I. Horrocks, D. McGuinness, P. Patel-Schneider, L. Stein. "OWL Web Ontology Language Reference". http://www.w3.org/TR/2003/WD-owl-ref-20030331/

[11] P. Patel-Schneider, P. Hayes, I. Horrocks. "OWL Web Ontology Language Semantics and Abstract Syntax" . http://www.w3.org/TR/2003/WD-owl-semantics-20030331/

[12] D. Fensel et al. "OIL: An Ontology Infrastructure for the Semantic Web", IEEE Intelligent Systems 16, Feb 2001.

[13] P. Patel-Schneider, I. Horrocks, F. van Harmelen, "Reviewing the Design of DAML+OIL: An Ontology Language for the Semantic Web", Proceedings of AAAI'02.

[14] S. Abiteboul, D. Quass, J. McHugh, J. Widom, and J. Wiener. "The Lorel Query Language for Semistructured Data". International Journal on Digital Libraries, 1(1): 68-88. April 1997.

[15] S. Alexaki, V. Christophides, G. Karvounarakis, D. Plexousakis. "On Storing Voluminous RDFDescriptions: The case of Web Portal Catalogs". In Proceedings of the 4th International

Workshop on the Web and Databases (WebDB'01) - In conjunction with ACM SIGMOD/PODS. Santa Barbara, CA. May 24-25, 2001.

[16] D. Brickley, R.V. Guha. "Resource Description Framework Schema (RDF/S) Specification 1.0". W3C Recommendation. March 27, 2000. http://www.w3.org/TR/rdf-schema

[17] S. Ceri, S. Comai, E. Damiani, P. Fraternali, S. Paraboschi, and L. Tanca. "XML-GL: a Graphical Language for Querying and Restructuring XML Documents". In Proceedings of International World Wide Web Conference, Toronto, Canada. 1999

[18] D. Chamberlin, D. Florescu, J. Robie, J. Simeon, and M. Stefanescu. "XQuery: A Query Language for XML". Working draft, World Wide Web Consortium. June 2001. http://www.w3.org/TR/xquery/

[19] Don Chamberlin, Peter Fankhauser, Massimo Marchiori, Jonathan Robie. "XML Query Use Cases". W3C Working Draft, 20 December 2001. http://www.w3.org/TR/xmlquery-use-cases

[20] M.F. Fernandez, D. Florescu, J. Kang, A.Y. Levy, and D. Suciu. "System Demonstration - Strudel: A Web-site Management System". In Proceedings of ACM SIGMOD Conf. on Management of Data, Tucson, AZ. May 1997.

[21] Ian Horrocks and Sergio Tessaris. "Querying the Semantic Web: a Formal Approach". To appear in the 1st International Semantic Web Conference (ISWC2002), June 9-12, 2002. Sardinia, Italy.

[22] D. Florescu, D. Chamberlin, J. Robie. "Quilt: An XML query language for heterogeneous data sources". In WebDB'2000, pages 53-62, Dallas, US. May 2000.

[23] Jeff Heflin, Raphael Volz, Jonathan Dale. "Requirements for a Web Ontology Language". W3C Working Draft. 7 March 2002.

저 자 소 개

김학래 (E-mail : hlkim@dku.edu)

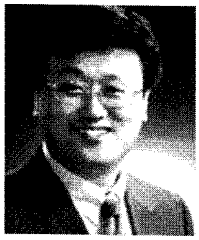
2002. 단국대학교 경영회계학부 (학사)

2004. 단국대학교 일반대학원 경영정보학과 (석사)

2006. 단국대학교 일반대학원 경영정보학과 (박사 수료)

2006 ~ 현재 연구원, Digital Enterprise Research Institute, Galway, Ireland

관심분야 시맨틱 웹, 온톨로지, 시맨틱 데스크탑, 블로그, 웹 2.0



김홍기 (E-mail : hgkim@kud.edu)

1985. 고려대학교 심리학 (학사)

1993. University of Gerogia 인공지능 (석사)

1996. University of Gerogia 인공지능 (박사)

1997 ~ 1998. 국 University of Gerogia 인공지능센터 Fellow

1998 ~ 2004. 단국대학교 경상학부 교수

2001 ~ 현재 충남 전자상거래지원센터 운영위원

2001 ~ 2001. 충남 테크노파크 기술개발부 겸임교수

2002 ~ 현재 한국 전산원 웹 정보화 자문교수

2005 ~ 현재 서울대학교 치과대학 교수

관심분야 시맨틱 웹, 온톨로지, 웹 서비스, 지식관리시스템, 인공지능