Proceedings of KFIS Autumn Conference 2007 Vol. 17, No. 2.

URL 패턴을 이용한 웹문서의 선택적 자동수집 방안

A Method of Selective Crawling for Web Document Using URL Pattern

정준영[1], 장문수[2]

[1] 서울시 성북구 서경대학교 소프트웨어학과
E-mail: grounder@naver.com

[2] 서울시 성북구 서경대학교 소프트웨어학과
E-mail: cosmos@skuniv.ac.kr

요 약

특정 분야별로 구축되는 온톨로지에 관하여 그 인스턴스를 쉽고 빠르게 구축하기 위해서는 구조화된 문서를 이용하는 것이 효율적이다. 그러나, 일반적인 웹 문서는 모든 분야에 대하여 다양한 형식으로 표현되어 존재하기 때문에, 대상이 되는 구조 문서를 자동으로 수집하기는 쉽지 않다. 본 논문에서는 웹사이트의 URL 패턴을 XML 기반의 스크립트로 정의하여, 필요한 웹 문서만을 지능적으로 수집하는 방안을 제안한다. 제안하는 수집 방안은 구조화된 형태로 정보를 제공하는 사이트에 대해서 매우 빠르고 효율적으로 적용될 수 있다. 본 논문에서는 제안하는 방법을 적용하여 5만개 이상의 웹 문서를 수집하였다.

Key Words : Web Crawling, URL Pattern, URL Filtering, Ontology, Semantic Web

1. 서 론

점차 방대해지는 웹에서 필요한 자료만을 찾기가 어려워지고 있다. 이러한 상황에서 현재 웹은 충분히 구조적이지 못하고 문서의 내용을 알기가 점차 어려워지고 있다. 이에 대한 대안으로 시맨틱 웹(Semantic Web)이 주목받고 있다. 시맨틱 웹이란 "기존의 웹이 가지는 한계들을 극복하고, 컴퓨터가 정보의 의미를 이해하고 의미를 조작할 수 있는 웹"이다[1]. 기존 웹은 데이터베이스에서 자료를 사용하지만 시맨틱 웹은 온톨로지(Ontology)를 이용한다. 온톨로지는 용어 사이의 관계를 정의하고 있는 일종의 사전과 같은 것이다. 온톨로지를 구축하려면 방대한 데이터가 필요한데 기존 웹정보를 이용하는 것이 한 방법이다.

본 논문에서는 온톨로지 구축을 위한 웹문서를 수집하기 위하여 웹 크롤링(web crawling) 기법을 제안한다. 기존 크롤링 기법들은 문서를 모으는 것 자체가 목표였기 때문에 웹의 모든 링크를 추적하여 문서를 수집하였다. 온톨로지는 분야별로 구축되기 때문에 그 분야의 정보만 수집되어야 한다. 본 논문에서는 필요한 분야의 정보만을 수집하는 크롤링 기법을 개발하고자 한다. 제안하는 방법은 대용량의 DB를 가지고 웹서비스를 제공하는 웹사이트들을 대상으로 사이트의 컨텐츠를 가리키는 주소인 URL(Universal Resource Locator)의 패턴을 이용하여 필요한 부분만을 수집한다.

이후 2장에서는 기존의 웹 크롤러와 웹 로봇에 관해 설명한다. 3장에서는 URL패턴을 이용한 문서 수집 방법과 이를 적용한 수집 시스템을 제안한다. 4장에서는 제안된 방법으로 문서를 수집한 결과를 나타내고, 마지막으로 5장에서는 결론과 향후 계획을 기술한다.

2. 연구 배경

웹 크롤러는 초기의 URL로 지정된 인터넷상의 웹서버를 접근하여 HTML문서를 읽어와 참조되지 않은 하이퍼링크(HyperLink)를 자동으로 추적하여 원하는 정보를 수집하고 자신의 데이터베이스에 내용을 저장한다. 이때 중복하여 수집하지 않도록 참조된 모든 URL을 저장하여 비교한다. 이 과정에서 웹 문서 수집기는 웹 페이지의 모든 링크를 접근하면서 필요없는 페이지를 수집하기도 한다. 이런 문제를 해결하기 위해 URL의 Ordering을 통한 웹 크롤링 방법[2]과 같이 링크에 가중치를 주어서 중요한 문서를 먼저 수집하도록 하였다.

초기의 크롤러는 정보 검색엔진의 색인 데

이터로 사용하기 위하여 최대한 문서를 많이 수집하는 웹로봇으로 발전해왔다. 그 후, 인터넷 정보의 활용도가 커짐에 따라 정보 수집을 위한 웹 크롤링 기술이 요구되어 한 사이트의 모든 문서를 수집하는 기술이 개발되기도 하고, 필요한 분야의 문서를 수집하는 기술이 연구되었다.

웹 크롤러는 일반적으로 문서를 분류할 수 없기 때문에 웹사이트의 모든 페이지를 수집해서 필요한 부분을 추출하였다. 그러나 웹 트래픽양이 증가하고 HTML의 기능이 다양화됨에 따라 전체 웹을 수집하는 것은 불가능한 일이 되어 갔다. 본 논문에서는 웹사이트의 모든 링크를 조사하지 않고 필요한 문서의 URL 패턴을 분석하여 필요한 링크만을 크롤링하여 수집하는 방법을 제시한다.

3. URL패턴을 이용한 문서 수집

웹에는 수많은 정보가 존재한다. 또한 따로 디지털화를 거치지 않아도 사용할 수 있다. 웹의 정보들 중에서 표나 리스트 등으로 구조화된 자료를 수집하고 해당 정보들을 온톨로지에 사용한다면 수작업을 하는 것보다 정확하고 방대한 데이터를 구축할 수 있다.

기존의 방식으로 대용량의 웹 문서를 수집하게 되면 필요한 문서외의 문서들을 가지고 온다. 본 논문에서는 대용량의 DB를 가지고 있는 웹사이트의 구조적인 문서를 가지고 오는 방법을 제시하려고 한다.

3.1 기본 URL 패턴

국내 대부분의 대용량 DB를 가진 웹사이트는 카테고리별로 DB내용을 분류한다. DB의 내용은 웹서버측의 서버 사이드 페이지(ASP, ASP.NET, JSP, PHP 등)를 이용하여 보여준다. 서버 사이드 페이지는 문서를 동적으로 생성하기 때문에 정적인 웹페이지와 다른 URL 패턴을 가지고 있다.

```
http://www.web.com/detail.aspx?prod_id=10032
0&view=contents
http://웹사이트주소/서버사이드페이지?변수전달
용 쿼리
```

그림 1. 서버 사이드 페이지 URL 패턴

그림 1은 웹사이트에서 사용하는 일반적인 서버 사이드 페이지 URL의 형식과 한 예를 나타낸 것이다. 이 URL의 뒷부분은 변수 전달 쿼리로 구성되는데 여기서 DB와 연결되는 ID값으로 DB에 저장된 각각의 웹페이지 정보와 연결된다. 따라서 이 ID값을 이용하면 사이트 내의 문서에 접근할 수 있다.

3.2 웹페이지 유형

상품 정보나 뉴스와 같이 많은 양의 DB화된 페이지를 가진 사이트들의 사이트 구성을 분석해 보면 몇 가지 페이지 유형으로 분류할 수 있는데, 일반 페이지, 리스트 페이지, 상세내용 페이지로 구분을 할 수 있다. 일반 페이지는 웹사이트 내에서 다른 메뉴로 가기 위한 중간 단계, 즉 정적인 자료를 서비스하기 위한 페이지들이다. 리스트 페이지는 특정 카테고리의 상세내용 페이지들을 나열하여 보여준다. 그리고 상세내용 페이지는 웹사이트에서 제공하는 주 컨텐츠로, 상품 정보의 경우 표를 이용한 구조문서와 특정한 문자열 패턴으로 표시되는 반 구조 문서로 되어 있다[3].

본 논문은 온톨로지 구축에 있어서 온톨로지의 인스턴스 정보를 추출하기 위한 정보 수집용 웹문서를 수집 대상으로 한다. 따라서 본 논문에서 제안하는 웹문서 수집 방법을 구현하는 시스템은 앞에서 설명한 페이지 유형 중에서 이러한 정보를 많이 가지고 있는 리스트 페이지와 상세정보 페이지를 수집한다.

3.3 웹문서 수집 시스템

웹문서 수집 시스템은 URL 추출 모듈, URL 필터링 모듈, 수집목록 관리 모듈, 문서수집 모듈로 구성된다. 그림 2는 제안하는 시스템의 구성도를 나타내고 있다.

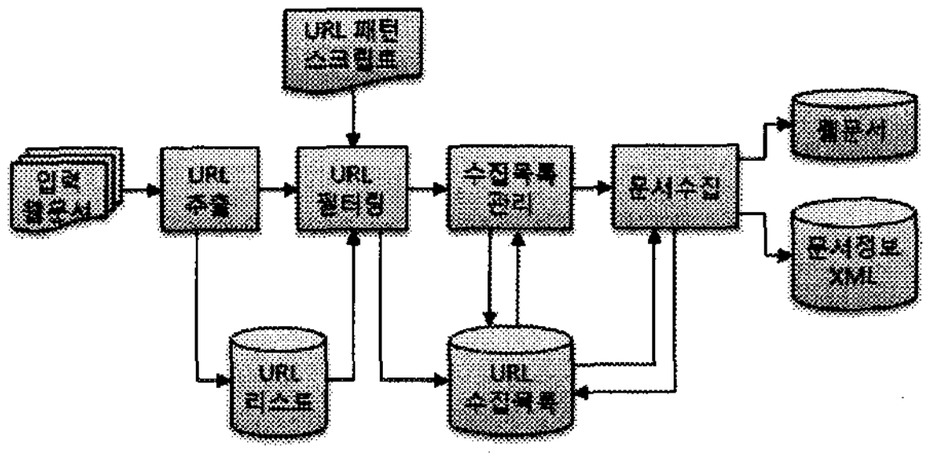

그림 2. 웹문서 수집 시스템의 구성도

URL 추출 모듈은 입력된 문서의 링크 정보로부터 추출이 가능한 URL을 전부 추출한다. URL 필터링 모듈은 추출된 URL 중에서 수집 대상이 되는 URL을 패턴 정보를 이용하여 걸러낸다. 수집목록 관리 모듈은 수집된 URL의 목록을 정해진 우선순위에 맞춰 순서를 조정하고 중복된 항목을 제거하는 등 수집목록을 관리한다. 문서수집 모듈은 수집 대상 URL 목록으로부터 URL을 읽어 HTML 문서를 수집하고 수집한 문서의 정보를 XML 형태의 정보파일에 기록한다.

3.3.1 URL 필터링 모듈

하나의 웹문서에는 여러 종류의 링크 정보가 있다. 수집에 필요한 페이지로 연결되는 링크뿐만 아니라 사이트 내의 일반 페이지로의 링크나 전혀 다른 사이트로 연결되는 광고 페이지 링크도 존재한다. 또한 자바 스크립트나 웹사이트에서 제공하는 리다이렉션(Redirection) 링크가 있다. URL 필터링 모듈에서는 먼저 불필요한 URL들을 필터링하여 제거하고 수집에 필요한 URL은 URL 수집 목록에 등록한다.

최근에 DB화된 많은 웹사이트들은 웹과 DB를 연결하기 위하여 리다이렉션 링크를 많이 사용한다. 이 링크 자체만으로는 수집 판단이 어렵기 때문에 본 논문에서는 이것을 URL 패턴이 나타나는 링크로 변환한다.

본 논문에서는 URL 필터링 과정에 필요한 여러 가지 URL 패턴을 URL 패턴 스크립트로 정의하여 사용한다. 이 스크립트는 XML 형식으로 작성된다. 그림 3은 URL 패턴 스크립트의 예를 나타내고 있다.

```
<crawlscript>
<site title="사이트 이름">
<identity value="www.web.com" />
<url type="list" class="인물" >
<originalAddr
type='url'>http://www.web.com/list.aspx?cate_id={0}
&page={1}</originalAddr>
<targetAddr>http://www.web.com/list.aspx?cate_id={
0}&page={1}</targetAddr>
</url>
<url type="detail" class="인물" >
<originalAddr
type='javascript'>javaScript:show_detail({0})</origin
alAddr>
<targetAddr>http://www.web.com/detail.aspx?per_id
={0}</targetAddr>
</url>
</site>
</crawlscript>
```

그림 3. URL 패턴 스크립트

URL 패턴 스크립트는 복잡한 URL 구조를 분석하여 기술하기 때문에 수작업으로 작성할 경우 오류를 유발할 가능성이 많다. XML로 작성된 스크립트는 향후 개선에 따른 확장성이 좋고, 스크립트 작성을 위한 어플리케이션 개발이 용이한 장점이 있다.

URL 패턴은 수집 대상 사이트의 링크 정보를 분석하여 패턴화가 가능한 부분을 패턴처리를 하여 스크립트에 추가한다. 그림 4는 URL의 패턴화 과정을 보여준다. 이러한 패턴을 이용하게 되면 기존의 웹 크롤러가 무작위로 모든 문서를 수집하여 목적에 적합하지 않는 문서를 모두 수집하는 문제점을 해결하여 필요한 문서의 URL만 남김으로써 문서 수집 속도와 수집된 문서의 질을 향상시킨다.

```
http://people.naver.com/directory/list.nhn?query=연구
관련단체인&where=job&dirid=467
                    ↓
http://people.naver.com/directory/list.nhn?query={0}&
where={1}&dirid={2}
{0} = 카테고리 명(필수)
{1} = 카테고리 분류 기준(필수)
{2} = 카테고리 고유ID(필수)
```

그림 4. URL 패턴 생성

3.3.2 수집목록 관리 모듈

URL 수집과정에서는 여러 문서들의 링크를 통하여 URL을 수집하기 때문에 중복되는 링크들이 많이 존재한다. 수집목록 관리 모듈에서는 우선적으로 목록에서 중복된 URL들을 제거한다.

수집 목록에 등록된 URL들은 현재 페이지가 속한 카테고리와 관련된 리스트 페이지와 상세정보 페이지를 가리키는 링크들이다. 수집목록 관리 모듈에서는 URL 수집목록에 등록된 URL들을 우선순위에 따라 순서를 조정한다. 상세정보 페이지는 목표한 수집 문서이므로 HTML 문서 수집을 위해 최우선 순위로 그 URL을 목록의 상위에 올린다.

리스트 페이지는 수집이 요구된 카테고리 소속인지 판단해야 한다. 리스트 페이지의 URL에는 시스템의 입력으로 들어온 시작 URL에 있는 것과 같은 종류의 링크가 있다. 이 URL의 패턴에는 수집하고자 하는 카테고리의 ID가 있는데, 수집목록에 있는 리스트 페이지 URL 중에 카테고리 ID가 같은 페이지들은 동일 카테고리 페이지의 링크들이다. 이 URL들은 수집 목록에서 상세정보 URL 다음의 우선순위를 부여하여 목록의 순서를 조정한다. 나머지 타 카테고리 리스트 페이지의 URL들은 하위 우선순위로 수집목록에 등록된다.

3.3.3 문서수집 모듈

문서수집 모듈은 HTML 문서 수집과 수집정보의 저장, 수집목록의 업데이트로 구성된다. HTML 문서는 자체적인 파일명 생성을 거쳐서 로컬 하드디스크에 저장된다. 이때 기수집된 문서의 중복 수집을 피하기 위하여 URL 수집목록에 수집완료에 따른 정보 업데이트를 동시에 수행한다.

파일 저장이 완료된 후 수집과정에 사용된 정보를 XML 형식의 수집문서 정보파일로 저장한다. 이것은 수집된 웹문서로부터 온톨로지 인스턴스를 생성할 때 온톨로지 클래스와의 매칭이나 인스턴스 정보 추출 알고리즘의 지식정보로 활용된다. 그림 5는 수집된 하나의

HTML 문서에 대한 저장 정보이다.

```
<doc type="struct">
<source>
<url>http://newprice.empas.com/pd/pd_list.php?cid=01037
0010</url>
<file>./Data/제품_노트북_엠파스_/00000000.html</file>
<site>엠파스</site>
<domain>제품</domain>
<category>노트북</category>
<date>Sun Aug 12 19:55:32 KST 2007</date>
</source>
</doc>
```

그림 5. 수집 정보 XML

4. 실험 및 결과

웹문서 수집 시스템은 J2SDK 1.5.0_03-b07과 공개 소스인 JTidy R7 개발자용 버전을 이용하여 구현하였다. 본 논문에서는 구현된 시스템을 이용하여 IT 분야 온톨로지 구축을 위한 웹문서를 수집하였다. 수집을 위한 사이트 선정은 주요 포털, 가격비교 사이트, 분야별 전문 웹사이트를 대상으로 하였다.

표 1은 9개 웹사이트에 대해서 문서를 수집한 결과를 나타낸 것이다. 전체 문서량은 추정치로서 경우에 따라서는 개수를 알 수 없는 사이트도 있다. 전문 사이트의 경우에는 수집한 문서 수가 전체 문서와 비슷하고, 일반 가격비교 사이트는 IT 외에도 많은 상품 문서가 있기 때문에 전체 문서 수보다 상당히 적은 양이 수집된 것을 알 수 있다.

표 1. IT분야 웹문서 수집 결과

웹사이트	전체 문서(추정)	수집 문서
인물정보1	약 5만개	7332개
인물정보2	약 30만개	9803개
가격비교1	약 55만개	13939개
가격비교2	약 1천개	3358개
가격비교3	알 수 없음	859개
가격비교4	약 8천개	7004개
도서정보1	알 수 없음	968개
기업정보1	1077개	976개
기업정보2	약 1만 3천개	152개

본 논문에서 제안하는 URL 패턴을 이용한 필터링 효과를 확인하기 위하여 하나의 웹문서에서 추출되는 URL의 수를 확인하였다. 표 2는 실험 대상 9개 사이트의 리스트 페이지를 하나씩 선정하여 URL 패턴 필터링 전후의 수집 대상 URL의 개수를 비교한 것이다. 사이트의 페이지의 구성 정책에 따라 리스트 내용과 관련 없는 정보가 많은 사이트는 필터링 전의 많은 URL이 필터링 후에 제거되었다. 전체적으로 많은 링크들이 제거된 것을 알 수 있다.

표 2. URL 패턴 필터링 결과

웹사이트	필터링 전	필터링 후
인물정보1	222개	29개
인물정보2	145개	39개
가격비교1	125개	10개
가격비교2	81개	6개
가격비교3	832개	21개
가격비교4	2835개	417개
도서정보1	359개	21개
기업정보1	30개	15개
기업정보2	67개	13개

5. 결론 및 향후 과제

인터넷이 방대해짐에 따라 필요한 정보를 찾는 일이 점점 어려워지고 있고, 인터넷에 있는 자료를 정보로 사용하기 위해 필요한 문서만 수집하는 일도 어려워졌다. 여기에는 다양해진 링크 방식과 동적인 문서의 비중이 높아진 것도 큰 원인이 된다. 본 논문에서는 이러한 웹 상황에서 필요한 문서를 수집하기 위하여 URL 패턴을 이용하여 문서를 수집하는 방법을 제시하고 시스템을 구성하였다.

본 논문에서 제시한 웹 문서 수집기는 필요없는 링크를 수집목록에서 제외시킴으로써 필요한 문서만을 수집할 뿐만 아니라 웹서버의 트래픽을 줄이고, 수집 속도를 빠르게 하였다.

그러나, URL 패턴으로 구분되지 않는 링크들은 제안하는 시스템으로는 처리가 불가능하므로 이에 대한 보완 연구가 필요하다. 그리고 URL 패턴 스크립트의 작성이 용이하지 않은 문제점도 해결해야할 향후과제로 남아있다.

참 고 문 헌

[1] Tim Berners-Lee, "Enabling Standards &Technologies,"(http://www.w3.org/2002/Talks/04-sweb/slide12-0.html)

[2] J. Cho, "Efficient Crawling through URL ordering," Computer Networks and ISDN Systems, Vol.30, pp. 161-172, 1998.

[3] 장문수, 강선미, “도메인지식의 계층화를 통한 온톨로지 인스턴스의 속성정보 추출”, 퍼지및지능시스템학회 논문지, 17권 3호, pp.291-296, 2007.6.