HTML태그패턴을이용한웹정보추출시스템

박 병 권 (동아대학교 경영정보과학부)

I. 서 론

오늘날에는 많은 조직들이 웹을 통하여 정보를 제공하므로 방대한 양의 정보가 HTML 페이지 형태로 기술되어 있다. 미국 특허정보사이트[14]는 수백만건이 넘는 특허정보를 자신의 웹사이트를 통해 제공하고 있다. 웹페이지에 들어 있는 정보를 알아내기 위해서는 웹브라우저를 통해 읽어 보아야 한다. 그러나, 수많은 웹페이지들을 모두 다 읽기보다는 질의하는 것이 보다 효과적이다.

웹페이지를 질의하기 위해서는 웹 페이지에 들어 있는 정보를 추출하여 구조화된 데이터 (예, SQL 질의를 위한 관계 데이터) 또는 반구조화된 데이터 (예. XQuery 질의를 위한 XML 데이터)로 변환하는 것이 필요하다. 현재 많은 웹정보추출 도구들이 존재한다[11]. 그들은 크게 자동추출 도구와 수동추출 도구로 분류할 수 있다. 웹페이지가 정해진 스키마를 가진 구조화 데이터로 이루어져 있다면 자동 추출이 가능할 것이다 [2, 3, 7, 8]. 그러나, 많은 웹 페이지들은 정해진 스키마가 없는 비구조화된 데이터를 포함하고 하다. 예를 들면, 미국특허 웹 페이지는 구조화된 데이터 (특허번호, 제목, 발명자, 등록일 등)와 비구조화된 데이터 (클레임, 도면, 세부내용 등)를 함께 가지고 있다. 이러한 웹 페이지는 정해진 스키마가 없으므로 사용자가 추출하고자 하는 데이터의 스키마를 명시해 주어야 한다.

사용자가 명시한 스키마를 가진 데이터를 추출하기 위해서는 추출 규칙이 필요하다. Embley의 연구[8]에서는 사용자가 웹페이지의 도메인에 대한 온톨로지를 제공한다. 또다른 시스템에서는 사용자가 웹페이지의 전반적인 구조를 그래픽 사용자 인터페이스를 통하여 지정해 준다. 예를 들면, NoDoSE (Northwestern Document Structure Extractor) [1]에서는 사용자가 웹페이지를 계층적으로 나눈 후 흥미있는 영역을 지정하고 내용을 기술한다. DEByE (Data Extraction By Example) [12]에서는 사용자가 웹페이지에서 몇 개의 객체를 선택한 후 그들을 이용하여 어떻게 테이블을 형성하는 지를 보여준다. 이러한 시스템들은 모두 사용자 입력을 하나의 학습 예제로 삼아 추출 규칙을 기계학습하도록 한다.

본 논문에서는 새로운 웹정보추출시스템을 제안한다. 제안한 시스템은 사용자가 명시한 목표 스키마와 HTML 태그 패턴을 기반으로 한다. 목표 스키마를 명시하기 위한 모델링 언어로서 UML (Unified Modeling Language) [5] 언어를 사용한다. 그리고 HTML 태그 패턴을 기술하기 위하여 TagRex (Tag Regular Expression) 언어를 개발

한다. TagRex를 이용하면 HTML 페이지 내의 특정한 양식 부분에 대한 HTML 태그 패턴을 기술할 수 있다.

제안한 시스템은 또한 사용자에게 TagRex를 이용하여 추출 규칙을 작성할 수 있는 언어 xRule을 제공한다. xRule을 통하여 사용자는 목표 스키마에 있는 각 객체의 애트리뷰트 값이 웹페이지 내에서 어떤 HTML 태그 패턴으로 나타나는 지를 명시할 수 있다. 각 객체의 애트리뷰트 값은 웹페이지 내에서 구별될 수 있는 어떤 양식을 띠고 있다. 따라서 웹페이지 내에서 특정 애트리뷰트의 값을 찾을 때에는 그 양식을 찾으면 된다. 그런데 어떤 양식은 웹페이지 내에서 HTML 태그 시퀀스로 표현되므로 이를 HTML 태그에 대한 정규식(regular expression)으로 나타낼 수 있다. 그러면 특정 애트리뷰트의 값을 추출해 내기 위하여 정규식으로 표현된 패턴을 웹페이지 내의 HTML 태그 시퀀스와 매치시키면 된다.

제안한 시스템의 정확도를 평가하기 위하여 미국 특허 웹사이트에 적용하였다. 그 결과, 수천건의 특허 정보가 HTML 페이지로부터 XML 데이터로 성공적으로 추출되었다. 이것은 미리 정해진 스키마가 없는 웹페이지로부터 구조화된 데이터와 비구조화된 데이터를 모두 추출하는데 있어서 우리의 접근법이 효과적이고 견고함을 입증한다.

본 연구 결과의 공헌을 요약하면 다음과 같다. (1) 웹페이지로부터 사용자가 지정한 목표 스키마를 가진 데이터를 추출하기 위한 규칙을 기술할 수 있는 새로운 언어 XRule을 개발하였다. xRule은 미리 정해진 스키마를 가지고 있지 않은 웹페이지로부터 구조화된 데이터와 비구조화된 데이터를 모두 추출할 수 있다. (2) HTML 태그 시퀀스의 패턴을 정규식으로 기술할 수 있는 새로운 정규 언어 TagRex를 개발하였다. 지금까지의 정규식은 문자 스트링에 대한 것인데 비해 TagRex 정규식은 HTML 태그 스트링에 대한 것이다. (3) 웹정보추출시스템을 개발하고 미국 특허사이트에 적용하여 수천건의 특허정보를 성공적으로 추출하였다.

본 논문의 구조는 다음과 같다. 제 2장에서는 UML을 이용한 목표 스키마의 개념적 모델링을 논한다. 제 3장에서는 추출 규칙을 기술하는 언어와 HTML 태그 시퀀스에 대한 정규식 언어에 대하여 논한다. 제 4장에서는 제안한 웹정보추출시스템에 대하여 논한다. 제 5장에서는 제안한 시스템을 미국 특허정보사이트에 적용하여 그 성능을 평가한다. 제 6장에서는 결론을 맺는다.

II. 웹 정보에 대한 개념적 모델링

본 장에서는 웹 페이지로부터 추출하고자 하는 데이터의 스키마를 모델링하는 법을 논한다. 본 논문을 통하여 미국 특허사이트를 예로 사용한다. 그림 1은 미국특허 웹페이지의 첫부분을 보여주고 있다. 여기에는 특허번호(patent number), 등록일(registration date), 제목(title), 요약(abstract), 발명자(inventors), 권리기관(assignee), 출원번호(application number), 접수일(filed date), 미국클래스(current US class), 국제클래스(international class), 그리고 검색분야(field of search) 등의 정보가 포함되어 있다.

United States Patent 6,820,082
Cook, et al. November 16, 2004

Rule based database security system and method

Abstract

A rule-based database security system and method are disclosed. A method for processing requests from a user to perform an action with respect to data stored in an electronic database includes defining a plurality of user defined rules containing security constraints for accessing the data and receiving a request at a user interface. The request is transferred from the user interface to a rule engine and the plurality of rules are applied to the request to determine if the request passes the security constraints.

Inventors: Cook; William R. (Redwood City, CA); Gannholm; Martin R. (San Francisco, CA)
Assignee: Allegis Corporation (San Francisco, CA)
Appl. No.: 541227
Filed: April 3, 2000

Current U.S. Class:707/9; 707/3
Intern'l Class: G06F 017/30
Field of Search: 707/9,4,101,103 Z,513 713/201,200

그림 1. 미국 특허정보 웹페이지 (Part 1).

그림 2는 또다른 부분을 보여주고 있다. 여기에는 참조된 미국특허(the list of US patents referenced), 심사자(examiners) 그리고 대리인(attorney) 등의 정보가 포함되어 있다.

References Cited [Referenced By]

U.S. Patent Documents

5265221	Nov., 1993	Miller	711/163.
5355474	Oct., 1994	Thuraisngham et al.	707/9.
5408657	Apr., 1995	Bigelow et al.	395/600.
5630127	May., 1997	Moore et al.	707/103.
5680614	Oct., 1997	Bakuya et al.	395/614.
5696898	Dec., 1997	Baker et al.	713/201.
5720033	Feb., 1998	Deo	713/200.
5751949	May., 1998	Thomson et al.	713/201.
5765160	Jun., 1998	Yamaguchi	707/103.
5826268	Oct., 1998	Schaefer et al.	707/9.
5905984	May., 1999	Thorsen	707/9.
5999978	Dec., 1999	Angal et al.	709/229.

Primary Examiner: Vu; Kim
Assistant Examiner: Hamilton; Monplaisir
Attorney, Agent or Firm: Van Pelt & Yi LLP

그림 2. 미국 특허정보 웹페이지 (Part 2).

그림 3은 마지막 부분을 보여주고 있다. 여기에는 클레임들이 나열되어 있고 각 클레임은 번호(claim number)와 내용(claim content)으로 구성되어 있다.

Claims

What is claimed is:

1. A method for processing requests from a user to perform an action with respect to data stored in an electronic database, the method comprising:

2. The method of claim 1 wherein said user defined rules are based on a relation between the user and said data.

3. The method of claim 1 wherein said user defined rules are based on rights associated with said user.

4. The method of claim 1 wherein said user defined rules are based on a relation between said user and requested data and rights associated with said user.

5. The method of claim 1 wherein said action is requesting data from said database and transferring said request comprises transferring a query.

그림 3. 미국 특허정보 웹페이지 (Part 3).

웹 페이지로부터 정보를 추출하기 위해서는 먼저 웹 페이지에 담긴 정보에 대한 개념적 모델 즉, 목표 스키마를 설계해야 한다. 본 논문에서는 목표 스키마로 UML 클래스 다이어그램을 사용한다. 클래스 다이어그램은 클래스 이름, 애트리뷰트 이름과 데이터 타입, 클래스간의 관계 정보를 가지고 있다. 그림 4는 미국특허 정보에 대한 클래스 다이어그램을 보여주고 있다.

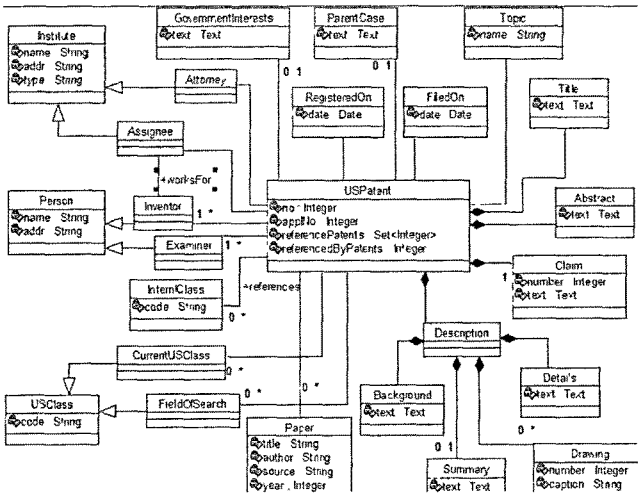

그림 4. 미국 특허정보에 대한 UML 클래스 다이어그램.

클래스 다이어그램은 텍스트 포맷으로 변환될 수 있다. 본 논문에서는 Rational Rose 의 Petal 포맷을 사용한다. Petal 포맷은 Rational Rose 에 의하여 자동적으로 생성된다. 그림 5는 그림 4의 클래스 다이어그램을 Petal 포맷으로 변환한 것의 일부를 보여주고 있다.

```
(object Class "USPatent"
  quid        "40F2F8020125"
  class_attributes        (list class_attribute_list
    (object ClassAttribute "no"
      quid                "40F2F8020126"
          type            "Integer")
    (object ClassAttribute "applNo"
      quid                "40F2F8020128"
          type            "Integer")
    (object ClassAttribute "referencePatents"
          quid            "410BCBBE0190"
          type            "Set<Integer>")
    (object ClassAttribute "referencedByPatents"
      quid                "410BCC2E02B4"
          type            "Integer")))
```

그림 5. 클래스 다이어그램의 텍스트 표현 (Petal).

III. 추출 규칙

본 장에서는 추출 규칙을 기술하기 위하여 두 개의 언어를 기술한다. 하나는 추출 규칙 명세 언어 (xRule) 이고, 다른 하나는 태그 시퀀스에 대한 정규식 언어 (TagRex) 이다. 본 논문에서는 웹 페이지를 하나의 HTML 태그 시퀀스로 간주한다.

3.1 추출규칙 명세 언어

본 절에서는 목표 스키마에 명시된 클래스의 애트리뷰트 값을 추출하기 위한 추출 규칙을 기술할 수 있는 언어 xRule에 관하여 논한다. 하나의 xRule 문장은 하나의 애트리뷰트 값에 대한 추출 규칙이다.

기본 형식: 하나의 xRule 문장은 그림 6과 같은 기본 형식을 가진다. 먼저 'CONTEXT'란 키워드 다음에 클래스 이름이 온다. 그리고, 할당 연산자 '=' 의 왼편에 애트리뷰트 이름이, 오른편에 TagRex 표현식이 온다. TagRex 표현식의 결과값이 애트리뷰트의 값으로 할당된다.

```
CONTEXT: ClassName
attributeName = TagRex expression for HTML tag sequence
```

그림 6. xRule 문장의 기본 구조.

변수: xRule 문장은 변수를 가질 수 있다. 그림 7과 같이 변수는 TagRex 표현식의 결과값을 저장한다. 모든 변수는 전역변수이므로 다른 xRule 문장에서 사용할 수 있다.

```
_variableName = TagRex expression for HTML tag sequence
```

그림 7. 변수의 정의.

반복문: xRule에는 두 종류의 반복문이 있다. 하나는 Array 타입을 가진 애트리뷰트 값을 위한 것이고, 다른 하나는 여러 개의 인스턴스 객체를 위한 것이다. 그림 8은 하나의 애트리뷰트에 대해 반복문을 수행하면서 Array 값을 할당하는 예이고, 그림 9는 반복문을 수행하면서 여러 개의 인스턴스 객체가 각각의 애트리뷰트에 할당되는 예이다. 또한 그림 10은 반복문 속에 반복문을 가진 중포 루프의 예를 보여주고 있다.

```
CONTEXT: ClassName
for _variableName repeat
    attributeName = TagRex Expression
end
```

그림 8. Array 타입.

```
for _variableName repeat
    CONTEXT: ClassName  attributeName = TagRex expression
end
```

그림 9. 객체 인스턴스.

```
for _variableName repeat
    for _variableName repeat
        CONTEXT: ClassName  attributeName = TagRex Expression
    end
end
```

그림 10. 중포 루프.

시작위치 지정: xRule 문으로 기술된 추출 규칙을 실행할 때 항상 웹 페이지의 처음부터 매칭이 시작된다. 그러나 시작위치 지정문 'Set Location' 을 통해 매칭의 시작 위치를 변경할 수 있다. 그림 11은 시작위치 지정문의 형식을 보여주고 있다. 키워드 'Set Location' 오른편의 TagRex 표현식과 매치되는 곳이 시작 위치를 나타내며, 시작위치 지정문이 실행되고 나면 항상 지정된 위치부터 매칭이 시작된다.

```
set location = regular expression for HTML tag sequence
```

그림 11. 시작위치 지정.

본 절에서는 xRule 언어를 논하였다. xRule은 배우기 쉽고 간단하지만 Array, 다중 이느턴스, 반복문 등 필요한 기능은 모두 가지고 있다. xRule은 객체지향 모델과 잘 부합되며 xRule로 기술된 추출 규칙들은 서로 독립적이므로 병행하여 실행할 수 있는 장점이 있다.

3.2 HTML 태그에 대한 정규식 언어

본 절에서는 HTML 태그 시퀀스 패턴을 기술하기 위해 특별히 고안된 정규식 언어인 TagRex에 대하여 논한다. TagRex 정규식은 xRule 문에 포함되어 사용된다. 표 1은 TagRex 정규식의 종류를 보여주고 있다. TagRex 정규식은 HTML 태그 스트링을 대상으로 하는 것이므로 문자열을 대상으로 하는 일반적인 정규식과는 다르다. TagRex 정규식에서는 매칭을 위한 기본 단위가 문자가 아니고 HTML 태그이다. HTML 문서에서 텍스트 세그먼트도 하나의 태그로 간주한다. 그러면 하나의 HTML 문서는 하나의 HTML 태그 시퀀스로 볼 수 있다.

표 1. TagRex 정규식.

정규식	설명
#...#	TagRex 정규식의 구분 표시자
#<a>~<b>#	<a>부터 <b>사이의 임의의 HTML 태그 시퀀스와 매치
#<a>~!<b>#	<a>부터 <b>이전까지의 임의의 HTML 태그 시퀀스와 매치
#<a>~!(<b>|<c>)#	<a>부터 <b> 또는 <c>이전까지의 HTML 태그 시퀀스와 매치
#<a>%<b>#	<a>와 <b>사이의 임의의 HTML 태그 시퀀스와 매치하고 그 사이의 HTML 태그 시퀀스를 반환
#<a>%!<b>#	<a>와 <b>이전까지의 임의의 HTML 태그 시퀀스와 매치하고 그 사이의 HTML 태그 시퀀스를 반환
$...$	문자열에 대한 일반적인 정규식
#<a>@href=$...$#	<a>의 애트리뷰트 href 값에 대한 일반 문자열 정규식

Return: 식별자 '#'으로 구분되는 TagRex 정규식은 HTML 태그 시퀀스 상에서 원하는 원하는 부분의 패턴을 나타낸다. TagRex 정규식 중에는 매치된 결과를 반환할 수 있으며 반환된 결과는 xRule 문에서 한 객체의 애트리뷰트 값으로 할당된다. TagRex 정규식에서 기호 '%'이 반환할 부분을 명시한다. 그림 12는 두 개의 예를 보여주는데 하나는 숫자를 반환하고 다른 하나는 텍스트를 반환하는 예이다.

```
CONTEXT: Patent
no = #<td><b>%</b></td>#

CONTEXT: Abstract
text = #<center><b>Abstract</b></center><p>%</p>#
```

그림 12. Return 예.

Skip: TagRex 정규식에서 기호 '~' 는 Skip을 명시한다. 매칭 시 '~' 는 임의의 HTML 태그와 모두 매치된다. 그림 13은 Skip의 한 예이다. 즉, 처음에 두 개의 연속적인 태그 *<tr><td>*를 만나면 그 다음 세 개의 연속적인 태그 *</td><td><b>*를 만날 때까지의 모든 태그를 Skip 한다. 그런 후, 세 개의 연속적인 태그 *<b></td></tr>*

을 만날 때까지의 모든 태그를 반환한다

```
CONTEXT: RegisteredOn
date = #<tr><td>~</td><td><b>%</b></td></tr>#
```

그림 13. Skip 예.

Choice: TagRex 여러 개의 웹 페이지에서 같은 것에 대하여 서로 다른 단어를 사용할 경우 TagRex 정규식에서는 이들을 모두 선택적으로 표현할 수 있어야 한다. 이를 위하여 기호 '| '를 사용한다. 그림 14는 태그
 다음에 'DESCRIPTION' 또는 'EMBODIMENT'가 포함된 단어가 나타나야 함을 보여주고 있다.

```
CONTEXT: Details
text = #<br>($DESCRIPTION$|$EMBODIMENT$)<br>%|<br><center><b>#
```

그림 14. Choice 예.

일반 문자열에 대한 정규식: 웹 페이지 내에는 많은 텍스트 세그먼트들이 존재한다. 이들을 위하여 TagRex는 기존의 일반 문자열에 대한 정규식을 허용하며 TagRex 정규식과 구분하기 위하여 식별자로 기호 '$'를 사용한다. 그림 15의 예에서는 클레임 텍스트 내에서 클레임 번호를 추출하기 위하여 일반 문자열에 대한 정규식을 사용하고 있다.

```
_claims = #<CENTER><B><I>$Claims$</B></I></CENTER><HR>%<HR>#

for _claims repeat
    CONTEXT: Claim number = $([0-9]+)\.$ text = $(?i)(.*?)(?=(?:<BR><BR>[0-9]+\.|\z))$
end
```

그림 15. 일반 문자열에 대한 정규식 예.

태그 애트리뷰트: HTML 태그 애트리뷰트는 문자열 값을 가진다. TagRex 정규식에서 기호 '@' 는 HTML 태그 애트리뷰트를 표시한다. 그림 16의 예에서는 태그 <a>의 애트리뷰트 'href'의 값에 포함되어 있는 특허번호를 추출한다.

```
CONTEXT: Patent
referencedByPatents = #<center><b>References Cited<a>@href=$.*(?=ref/)ref/(.*)$#
```

그림 16. 태그 애트리뷰트 예.

본 절에서는 HTML 태그 시퀀스를 위해 특별히 고안된 정규식 언어 TagRex에 관하여 기술하였다. TagRex 정규식을 이용하면 문자 수준이 아닌 태그 수준에 기반한 패턴을 기술할 수 있다. 하나의 HTML 웹 페이지에 포함된 문자의 수와 태그의 수를

비교해 보면 태그 기반 정규식이 훨씬 작성하기가 쉬움을 알 수 있다. 실제로 미국특허 웹 페이지의 경우 문자 수보다 태그 수가 10배 이상 적다.

IV. 웹 정보 추출 시스템

본 장에서는 본 논문에서 제안한 웹 정보 추출 시스템의 기능에 대하여 기술한다. 그림 17은 제안한 웹 정보 추출 시스템의 아키텍춰를 보여주고 있다. 제안한 시스템은 RME(Rule Matching Engine)와 XDG(XML Document Generator)라는 두 개의 주요 모듈로 구성되어 있다.

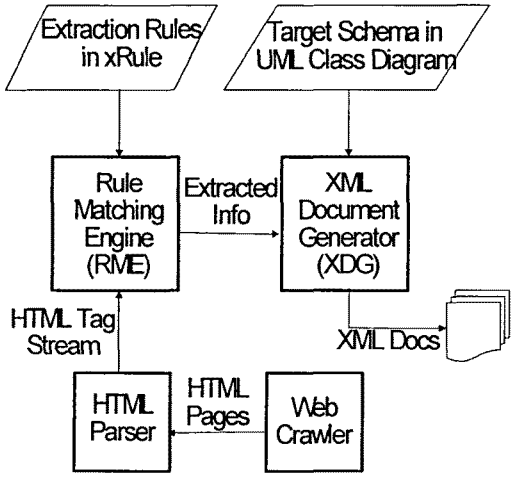

그림 17. 웹 정보 추출 시스템 아키텍춰.

RME는 사용자로부터 xRule 언어로 기술된 추출 규칙을 입력받아 웹페이지와의 매칭을 효율적으로 하기 위해 추출 규칙을 컴파일한다. 그리고 HTML 파서는 웹 크롤러가 수집한 HTML 페이지를 HTML 태그 시퀀스로 바꾸어 RME로 넘겨준다. RME는 추출 규칙과 HTML 태그 스트림을 매치시킨다. XDG는 추출된 웹 정보를 XML 문서로 변환한다. 이를 위해 사용자로부터 Petal 포맷으로 변환된 목표 스키마를 입력받는다. RME는 추출된 웹 정보를 객체 인스턴스 형태로 XDG에게 전달한다. XDG는 목표 스키마에 기술된 클래스 구조에 근거하여 이들이 목표 스키마에 합당함을 평가하고 XML 문서로 변환한다. 이 때, 각 객체 인스턴스는 하나의 XML Element로 변환된다. 변환된 XML 문서는 XML 데이터베이스에 저장된다.

V. 평가

제안한 웹 정보 추출 시스템의 정확도를 평가하기 위하여 미국 특허 웹사이트에 적용하였다. 그 결과, 수천건의 특허 정보가 HTML 페이지로부터 XML 데이터로 성공적으로 추출되었다. 이것은 미리 정해진 스키마가 없는 웹페이지로부터 구조화된 데이터와 비구조화된 데이터를 모두 추출하는데 있어서 제안한 시스템이 효과적이고 견고함을 입증한다. 그림 18은 미국특허 정보 추출을 위한 메인 화면을 나타내고, 그림 19는 실제로 미국특허 웹사이트로부터 HTML 문서를 다운로드 받아 웹 정보를 추출하고 있는 과정을 보여 주며, 그림 20은 추출된 특허 정보를 XML 문서로 변환한 예를 보여 주고 있다.

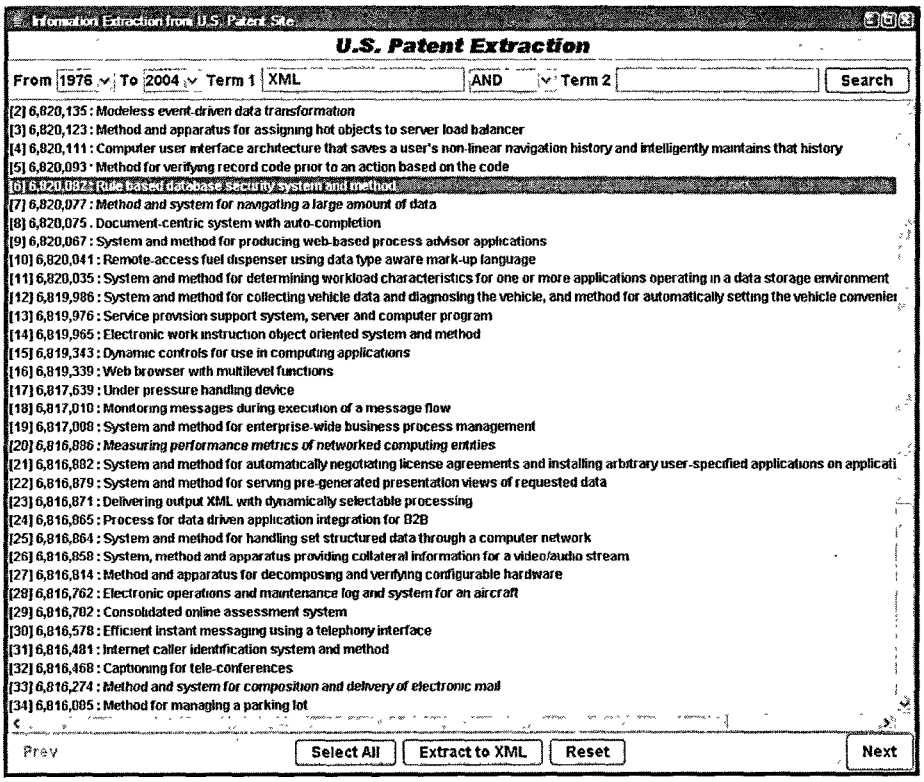

그림 18. 미국특허정보추출시스템 메인화면.

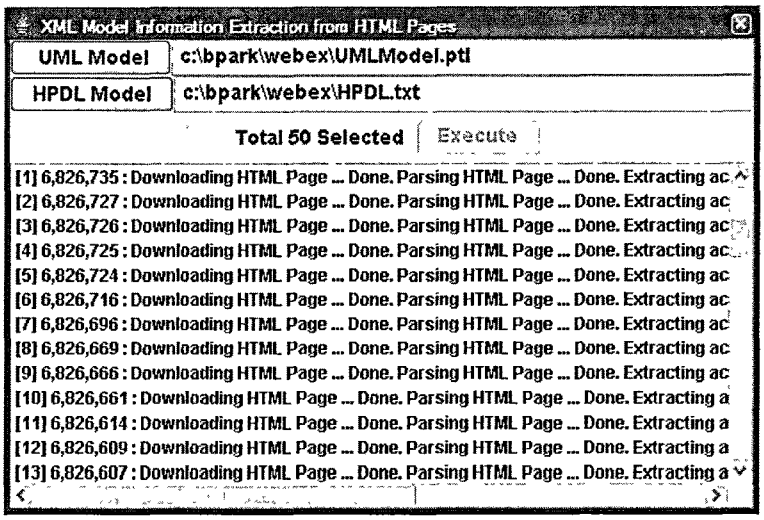

그림 19. 추출진행 예.

```
<U.S. Patent>
  <Title>
    <text> Rule based database security system and method </text>
  </Title>
  <Abstract>
    <text> A rule-based database security system and method are disclosed. A method for ... </text>
  </Abstract>
  <Inventor>
    <name> Cook; William R. </name>
    <addr> Redwood City, CA </addr>
    <name> Gannholm; Martin R. </name>
    <addr> San Francisco, CA </addr>
  </Inventor>
  <Patent>
    <no> 6,820,082 </no>
    <applNo> 541227 </applNo>
    <referencePatents> "5265221", "5355474", "5408657", "5630127", "5680614", "5696898", ...
    </referencePatents>
  </Patent>
  <CurrentUSClass> <code> 707/9; 707/3 </code> </CurrentUSClass>
  <InternlClass> <code> G06F 017/30 </code> </InternlClass>
  <FieldOfSearch> <code> 707/9,4,101,103 Z,513 713/201,200 </code> </FieldOfSearch>
  <RegisteredOn> <date> November 16, 2004 </date> </RegisteredOn>
  <FiledOn> <date> April 3, 2000 </date> </FiledOn>
  <Claim>
    <number> 1 </number>
    <text> A method for processing requests from a user to perform an action with respect ...</text>
    <number> 2 </number>
    <text> The method of claim 1 wherein said user defined rules are based on a relation...</text>
    <number> 3 </number>
    <text> The method of claim 1 wherein said user defined rules are based on rights ... </text>
    <number> 4 </number>
    <text> The method of claim 1 wherein said user defined rules are based on a relation ...</text>
    <number> 5 </number>
    <text> The method of claim 1 wherein said action is requesting data from said database...</text>
  </Claim>
  ...
</U.S. Patent>
```

그림 20. 추출된 XML 문서 예.

VI. 결론

본 논문에서는 새로운 웹 정보 추출 시스템을 제안하였다. 이를 위해, xRule 언어와 TagRex 언어를 제안하였다. 본 논문의 공헌은 다음과 같다. (1) 목표 스키마를 통해 사용자가 명시한 객체의 각 애트리뷰트 값을 웹 페이지로부터 추출하기 위한 추출 규칙을 기술할 수 있는 언어를 개발하였다. (2) HTML 태그 시퀀스의 패턴을 기술하기 위한 새로운 정규식 언어를 개발하였다. 이것은 태그 시퀀스에 관한 최초의 정규식 언어라고 믿는다. (3) 새로운 웹 정보 추출 시스템을 구현하고 실제 미국특허 웹사이트로부터 수천건의 미국특허 정보를 추출하였다.

제안된 시스템은 시멘틱 웹을 위해서도 사용될 수 있다. 시멘틱 웹을 위해서는 의미에 대한 주석을 다는 것이 필요하다. 제안된 시스템을 사용하면 웹 페이지의 내용이 XML 문서로 추출되면서 각 객체의 의미가 XML 태그로 표현된다. 따라서 시멘틱 웹을 구축하는데 제안된 시스템이 바로 사용될 수 있다.

참 고 문 헌

[1] B. Adelberg, "NoDoSE - A tool for Semi-Automatically Extracting Structured and Semistructured Data from Text Documents," In *Proc. Int'l Conf. on Management of Data*, ACM SIGMOD, pp. 283-294, Seattle, 1998.

[2] A. Arasu and H. Garcia-Molina, "Extracting Structured Data from Web Pages," In *Proc. Int'l Conf. on Management of Data*, ACM SIGMOD, pp. 337-348, San Diego, June 2003.

[3] C. Chang and S. Lui, "IEPAD: Information Extraction based on Pattern Discovery," In *Proc. Int'l Conf. on World Wide Web (WWW10)*, pp. 681-688, Hong Kong, May 2001.

[4] C. Y. Chung, M. Gertz, and N. Sundaresan, "Reverse Engineering for Web Data: From Visual to Semantic Structures," In *Proc. Int'l Conf. on Data Engineering (ICDE02)*, pp. 363-374, San Jose, California, 2002.

[5] J. Conallen, *Building Web Applications with UML*, Addison Wesley, 2000.

[6] V. Crescenzi and G. Mecca, "Grammars Have Exceptions," *Information Systems*,

Vol. 23, No. 8, pp. 539-565, 1998.

[7] V. Crescenzi, G. Mecca, and P. Merialdo, "RoadRunner: Towards Automatic Data Extraction from Large Web Sites," In *Proc. Int'l Conf. on Very Large Data Bases*, pp. 109-118, Rome, 2001.

[8] D. W. Embley, D. M. Campbell, Y. S. Jiang, S. W. Liddle, D. W. Lonsdale, Y.-K. Ng, and R. D. Smith, "Conceptual-model-based data extraction from multiple-record Web pages," *Data & Knowledge Engineering*, Vol. 31, No. 3, pp. 227-251, 1999.

[9] J. Hammer, H. Garcia-Molina, J. Cho, A. Crespo, and R. Aranha, "Extracting Semistructured Information from the Web," In *Proc. Workshop on Management of Semistructured Data*, 1997.

[10] J. Hammer, H. Garcia-Molina, S. Nestorov, R. Yerneni, M. Breunig, and V. Vas-salos, "Template-Based Wrappers in the TSIMMIS System," In *Proc. Int'l Conf. on Management of Data*, ACM SIGMOD, pp. 532-535, AZ, USA, 1997.

[11] A. H. F. Laender, B. A. Ribeiro-Neto, A. S. da Silva, and J. S. Teixeira, "A Brief Survey ofWeb Data Extraction Tools," *SIGMOD Record*, Vol. 31, No. 2, pp. 84-93, June 2002.

[12] A. H. F. Laender, B. Ribeiro-Neto, and A. S. da Silva, "DEByE - Data Extraction By Example," *Data & Knowledge Engineering*, Vol. 40, No. 2, pp. 121-154, 2002.

[13] A. Sahuguet and F. Azavant, "Looking at theWeb through XML glasses," In *Proc. IFCIS Intl Conf. on Cooperative Information Systems (CoopIS99)*, pp. 148-159, 1999.

[14] United States Patent and Trademark Office, http://www.uspto.gov/

<Abstract>

Web Information Extraction using HTML Tag Pattern

Byung-Kwon Park
Dong-A University

To query the vast amount of web pages which are available in the Internet, it is necessary to extract the encoded information in the web pages for converting it into structured data (e.g. relational data for SQL) or semistructured data (e.g. XML data for XQuery). In this paper, we propose a new web information extraction system, PIES, to convert web information into XML documents. PIES is based on a user-specified target schema and HTML tag pattern descriptions. The web information is extracted by the pattern descriptions and validated by the target schema. We designed a new language to describe extraction rules, and a new regular expression to describe HTML tag patterns. We implemented PIES and applied it to the US patent web site to evaluate its correctness. It successfully extracted more than thousands of US patent data and converted them into XML documents.