이동에이전트를 이용한 XML 정보의 수집 및 분류

서효정 , 방대욱
계명대학교 컴퓨터공학부

Information Gathering Agent System using XML

Hyo-Jeong Seo , Dae-Uook Bang
Dept. of Computer Engineering, Keimyung University

요 약

요즘처럼 웹을 이용하여 정보 검색시 너무나 많은 양의 정보를 수집, 정리, 관리 해야 하는 문제에 직면하게 되었다. 또한 인터넷상에는 기존의 텍스트 자료 이외에도 이미지, 사운드, 데이터 베이스 등 우리가 원하는 여러 유형의 자료가 존재 한다. 하지만 웹상에서는 텍스트만을 위주로 자료를 검색, 수집, 분류를 한다. 이러한 문제점을 해결하기 위해 XML 를 이용하여 정보의 종류에 관계없이 수집할 수 있다. 이 논문에서는 이동 에이전트를 이용한 정보 검색 모형을 제시하고 이때 이동에이전트가 정보의 표현방법으로 XML 를 사용한다. 또한 XML 의 계층적인 특성을 활용하여 XML 문서의 분류, 병합을 할 수 있다. 따라서 수집된 정보의 정리된 형태로 쉽게 얻을 수 있다.

1. 서론

인터넷의 발달과 더불어 정보의 양이 하루가 다르게 늘어나고 있다. 그래서 폭발적으로 증가하는 데이터 속에서 사용자들은 원하는 정보를 찾아내는 데는 많은 시간과 노력이 필요하게 된다. 특히 사용자들이 요즘 인터넷 상에서 수집하는 자료는 텍스트는 물론이고 이미지, 사운드, 지도, 데이터 베이스 등 여러 가지 형태를 이룬다. 이러한 여러 형태의 자료를 동시에 검색하고 수집하고 심지어는 분류까지 하는 작업은 사용자들에게 많은 시간과 노력을 요구하는 힘든 작업이 된다.

또한 정보 검색 시에 정보의 위치는 어느 특정한 한곳에 모여있는 것이 아니라 여러 곳에 흩어져 있으며 이러한 자료들이 항상 변화하고 있다.

문제 해결의 방법으로 XML[1]을 사용한다. XML 은 마크업 언어로써 기존의 HTML 이 가지는 방법을 유지하면서 SGML 의 복잡성을 해결하고 있다.

XML 의 장점은 SGML[8]중 필요한 극히 일부의 기능만을 취하여 인터넷의 콘텐트 전송에 적합하도록 고안된 언어이며 또한 고정된 DTD 을 가지는 HTML 과는 달리 XML 은 DTD 가 고정되어 있지 않으므로 만약 필요하다면 작성자가 정의한 대로 다양한 논리적 구조를 표현할 수 있는 유연성을 제공한다. 또한 XML 언어는 어떤 종류의 자료를 표현할 때 구조적인 구성을 이루고 있다.

XML 을 검색한 자료를 담는 하나의 콘텐트로 생각하며, 에이전트가 자료를 검색하여 그 자료의 종류에 관계없이 XML 문서에 담는다. 이때 이미 정의된 DTD 을 가지고 있으면 좋으나, 만약 그렇지 못한 경우 새로운 태그를 정의하여 자료를 담는다.

이렇게 검색된 XML 은 특성상 구조화된 계층 구조를 이룬다[3]. 이들 검색된 자료들이 그대로 이용자에게 보여 질 수도 있고 이들 자료들 중 원하는 부분만을 XML 문서로 재 편집한다. 즉 계층 구조를 이루는 XML 은 트리로 표현될 수 있고 원하는 자료가 있는 서버 트리 부분만을 병합 시킴으로써 새로운 하나의 XML 를 만들 수 있다.

이 논문에서는 이동에이전트를 이용하여 정보 검색 모형을 제시하고, 이때 이동에이전트가 정보의 표현 방법으로 XML 를 사용한다. 또한 XML 의 계층적인 특성을 활용하여 XML 문서의 분류, 병합이 이루어 진다. 수집한 정보의 정리된 형태로 쉽게 얻는다.

본 논문은 2 장 이동에이전트를 이용한 정보 검색 모형에 대해 살펴보고 3 장에서는 정보의 표현 방법에 대하여 4 장에서는 합병에 의한 정보의 정리에 대해 마지막 5 장에서는 결론 및 향후 연구 방향을 다룬다.

2. 이동 에이전트를 이용한 정보 검색 모형

정보수집을 위해 이동에이전트를 이용한다. 이 이동에이전트는 자료가 있는 곳의 위치를 담은 아이티너리를 보고 이동한다. 이 논문에서 제시하는 에이전트의 이동방법은 사용자의 지시를 받은 부모 에이전트는 에이전트가 보내져야 할 곳 만큼 즉 아이티너리에 등록된 수 만큼의 에이전트를 생성 시킨다. 그리고 생성된 자식 에이전트는 아이티너리에 적혀 있는 곳으로 보내지게 된다. 각 자식 에이전트가 정보를 검색하여 결과 XML 을 부모 에이전트에 넘겨 주게 된다. 부모 에이전트은 보낸 자식 에이전트가 모두 되돌아 올 때 까지 기다리게 된다 이때 부모 에이전트는 자식 에이전트로부터 받은 XML 을 가지고 구조화된 트리를 만들게 된다[5,6].

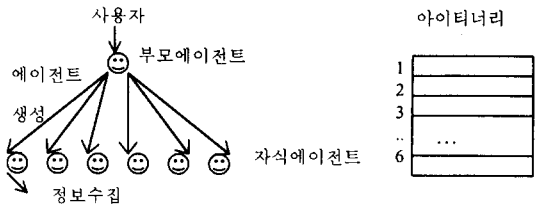

그림 1. 이동에이전트에 의한 정보 검색

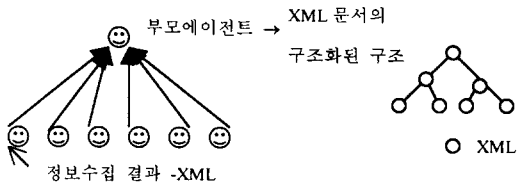

그림 2. 이동에이전트에 의한 정보 수집

3. 정보의 표현

이 논문에서는 모든 자료는 XML 에 담겨져 수집, 정리 된다. XML 장점중의 하나인 사용자가 원하는 태그를 작성할 수 있다는 것이다. XML 에 의해 표현된 2 가지 문서의 예이다.

```
수영

1. 수영의 유래
   수영의 초기 시작은 .....
2. 기본기를 익히자
   수영에서 제일 중요한 것은 호흡으로써 먼저.....
3. 자유형을 배우자
.....
4. 배영을 배우자
5. 평형을 배우자
6. 접영을 배우자
7. 심판
```

그림 3 문서 1

```
<?XML VERSION= "1.0" ENCORDING="UTF-8' RMD="NONE">
< !DOCUMENT  수영 SYSTEM  "수영.DTD">
<수영>
<INFO>
<SUBJECT> 수영 </SUBJECT>
<AUTHOR> Lee </AUTHOR>
</INFO>

<PART NO="1">
<CHAP NO ="1">
<CHAPTEXT>수영의 유래</ CHAPTEXT >
<C> 수영의 초기 시작은 .....</C>
</CHAP>

<CHAP NO ="2">
<CHAPTEXT>기본기를 익히자</ CHAPTEXT >
<C> 수영에서 제일 중요한 것은 .....</C>
</CHAP>
</PART>
...
</수영>
```

그림 4 문서 1 의 XML 인스턴스

```
레져 - 수영

1. 수영의 유래
   동서양을 막론하고 수영 발달의 ...
2. 4 가지 영법
   2.1 자유형
       자유형에서는 ...
   2.2 배영
       배영에서는
   2.3 평영
   2.4 접영
3. 세계 주요 아름다운 야외 수영장
4. 수중에어로빅이란?
```

그림 5 문서 2

```
<?XML VERSION= "1.0" ENCORDING="UTF-8' RMD="NONE">
< !DOCUMENT  레져-수영 SYSTEM  "수영.DTD">
<수영>
<INFO>
<SUBJECT> 수영 </SUBJECT>
<AUTHOR> Kwon </AUTHOR>
</INFO>

<PART NO="1">
<CHAP NO ="1">
<CHAPTEXT>수영의 유래</ CHAPTEXT >
<C> 동서양을 막론하고 수영 발달의  .....</C>
</CHAP>

<CHAP NO ="2">
<CHAPTEXT>4 가지 영법</ CHAPTEXT >
<PARA NO="1">
<PARATEXT> 자유형 < /PARATEXT>
<C> 자유형에서는 .....</C>
</PARA>

< PARA NO="2">
<PARATEXT> 배형 < /PARATEXT>
<C> 배형에서는 .....</C>
</PARA>

...
</CHAP>
</PART>
....
</수영>
```

그림 6 문서 2 의 XML 인스턴스

위의 그림은 사용자가 '수영'에 관한 정보를 수집하고자 할 때 검색되어진 2 개의 문서이다. 문서 1 의 경우에는 각 단원 아래의 하위 단원이 존재 하지 않는다. 그래서 그림 4 에서 <CHAP>라는 태그 하나만으로도 충분하지만 문서 2 에서는 CHAP2 하부에 다시 4 개의 단원이 존재하게 된다. 이때는 <PARA>라는 태그를 사용한다[1,2].

4. 합병에 의한 정보의 정리

검색된 자료들은 XML 형태로 이루어져 있으며 이들은 계층적 구조를 이루는 트리로 만들수 있다. 이때 부모 에이전트는 자식 에이전트가 가지고 오는 XML 마다 하나의 트리를 만들수도 있으나 이렇게 되면 너무나 많은 트리가 생기게 된다. 그래서 이 논문에서는 부모 에이전트가 자식 에이전트가 가지고 오는 XML 을 즉시 합병하는 방법을 택한다. 이때 모든 XML 을 하나의 트리로 표현하는 것이 아니라 자료중 관련이 적은 것은 따로 하나의 트리로 작성하거나 트리의 깊이가 어느 정도를 넘어가면 새로운 트리로 만든다[7].

앞의 예에서 나온 문서를 정리한 형태는 아래와 같다

```
<?XML VERSION= "1.0" ENCORDING="UTF-8' RMD="NONE">
< !DOCUMENT  수영 SYSTEM  "수영.DTD">
<수영>
<INFO>
<SUBJECT> 수영 </SUBJECT>
<AUTHOR>SUB </AUTHOR>
</INFO>

<PART NO="1">
<CHAP NO ="1">
<CHAPTEXT>수영의 유래</ CHAPTEXT >
<C> 수영의 초기 시작은 ……</C>
<C> 동서양을 막론하고 수영 발달의  ……</C>
</CHAP>

<CHAP NO ="2">
<CHAPTEXT>기본기를 익히자</ CHAPTEXT >
<C> 수영에서 제일 중요한 것은 ……</C>
</CHAP>

<CHAP NO ="3">
<CHAPTEXT>4 가지 영법</ CHAPTEXT >
<PARA NO="1">
<PARATEXT> 자유형 < /PARATEXT>
<C> 자유형에서는 ……..</C>
<C>…..                    </C>
</PARA>

< PARA NO="2">
<PARATEXT> 배형 < /PARATEXT>
<C> 배형에서는 ……..</C>
</PARA>
</CHAP>
</PART>

</수영>
```

그림 7. 수영에 대해 정리된 문서 1

```
<?XML VERSION= "1.0" ENCORDING="UTF-8' RMD="NONE">
< !DOCUMENT  레져-수영 SYSTEM  "수영.DTD">
<수영>
<INFO>
<SUBJECT> 수영 </SUBJECT>
<AUTHOR>SUB </AUTHOR>
</INFO>

<PART NO="1">
<CHAP NO ="1">
<CHAPTEXT>수중에어로빅이란?</ CHAPTEXT >
<C> 요즘 가장 인기 있는 레져의 한 종목으로  ……</C>
</CHAP>
</PART>

</수영>
```

그림 8. 수영에 대해 정리된 문서 2

정리된 문서는 크게 2가지로 나눌 수 있다. 그림 7은 문서 1과 문서 2를 정리한 결과 문서로써 두 문서중에서 수영의 역사와 영법 등의 내용이 담겨져 있고 그림 8은 문서 정리시 관련이 적은 수중에어로빅의 내용이 담겨져 있다.
XML은 하나의 보고서 형태로 만들어 지거나 또는 여러 개의 보고서 형태로 만들어 질 수 있다. XML을 트리 구조로 만든다. 이때 관련된 부분의 Tree 상위 노드에서 하위 노드를 하나로 merge 함으로써 이루어 진다.
위의 예에서는 문서 1과 문서 2의 XML 인스턴스중에서 <PART> <CHAP> <PARA> 즉 문서를 분류하는 태그를 중심으로 이들을 계층구조 모형으로 만든다. 즉 <CHAP>는 <PART>의 하위 레벨이고 <PARA>는 <CHAP>아래에 존재하게 된다. 들었다 이때 사용되는 개괄적 알고리즘은 아래와 같다[2,4].

1. 각 XML 문서를 파싱하여 태그의 구성를 보고 문서의 계층 구조로 만듬
   1.1 문서의 정보를 담고 있는 INFO 부분을 제거
   1.2 문서의 XML DTD 를 보고 문서의 구조 파악
   1.3 사용자 정의 태그가 있으면 분석
   1.4 태그에 대한 트리 형성

2. 여러 개의 트리를 태그의 제목이나 주제어를 기준하여 하나의 트리로 병합
   2.1 XML 인스턴스를 보고 트리에서 해당 태그의 주제어 파악
   2.2 여러 다른 트리의 태그 주제어와 비교
   2.3 동일 주제어를 합병
   2.4 다른 주제어는 새 트리 생성 또는 기존의 트리에 합병

3. 하나 또는 여러 개의 트리를 XML 문서로 표현
   3.1 합병 문서만큼의 문서 INFO 부분 작성
   3.2 각 트리의 XML DTD 를 새롭게 작성
   3.3 각 트리를 새로 작성한 INFO 부분과 결합하여 XML 인스턴스 작성
   3.4 XML DTD 와 XML 인스턴스를 가지고 사용자에게 보고서 문서를 만들어 줌

그림 9 XML 문서 정리의 개괄적 알고리즘

5. 결론

이 논문에서 이동에이전트를 이용하여 정보를 수집하며 수집된 정보를 XML 를 콘텐트로 생각하여 XML 에 담는다 XML 의 특성인 계층 구조를 이루므로 XML 문서들을 트리 형태로 표현함으로써 문서를 쉽게 분류 할 수 있고 유사성을 가지는 부분들 끼리 병합할 수 있으므로 사용자들에게 하나의 정리된 정보를 제공할 수 있다. 앞으로는 앞에서 자료 수집에 있어서 부모와 자식 에이전트간의 통신 부분에 대해 좀 더 연구해야 하며 이러한 에이전트 시스템을 이용하여 여러 응용 분야에 적용 시키도록 연구 해야 할 것이다.

참고문헌

[1] Dan Connolly and Jon Bosak, Extensible Markup Language(XML) , 1977, http://www.w3.org/XML/
[2]Tim Bary and C.M.Sperberg-McQeen, "Extensible Markup Language(XML):Part1. Syntax" 1997.6, http: www.w3.org TR WD-xml-lang.html
[3]Jon Bosak, "XML , Java, and Future of the web" 1997.3
http://sunsite.unc.edu/pub/sun-info/standards/xml/why/xmlapps.htm
[4]Paul Prescod, "Introduction to DSSSL"1997.7
http://itrc.uwqterloo.ca/~papersco/dsssl/tutorial.html
[5]Victor Lesser, Bryan Horling, "A Resource-Bounded Information Gethering Agent". 1998.
[6]Chanda Dharap, Martin Freeman,"Information Agents for Automated Browsing,1996.11
[7]Mark Devaney," Dynamically Adjusting Concepts to Accommodate Changing Contexts
[8]정회경, "SGML 가이드", 사이버 출판사, 1997