데이터베이스를 이용한 웹로봇 기반의 정보필터링 에이전트 시스템*

강민철*, 신석철**, 정태선*
*아주대학교 정보통신전문대학원
**국방과학연구소
e-mail : ekfo33@nate.com, scshin@add.re.kr, tschung@ajou.ac.kr

Database System for Web Robot based Information Filtering Agent System

Min-Chul Kang*, Seok-Cheol Shin**, Tae-Sun Chung*
*Graduate School of Information and Communication, Ajou University
**Agency for Defense Development

요 약

인터넷은 방대한 정보의 집합체이다. 사용자들은 웹에서 자신이 원하는 정보를 검색하여 사용하고 있다. 하지만 웹은 워낙 방대한 정보를 보유하고 있고 사용자가 원하는 정보가 다양해질수록 이러한 정보를 찾는 것은 어려워질 수 있다. 많은 유저들이 서로 다른 기호를 가지고 있는 만큼, 사용자에 따라 다른 형태의 정보를 제공하는 것이 필요하다. 이러한 형태의 서비스를 제공하기 위해서는 다양한 프로그램들이 상호협력하는 것이 필요하다. 본 논문은 데이터베이스를 활용한 멀티 에이전트 시스템을 통하여 사용자가 원하는 정보를 쉽게 관리하고 찾는 것에 목적을 둔다.

1. 서론

인터넷은 지속적으로 성장하고 있다. 인터넷은 성장하는 만큼 방대한 정보를 포함하고 있으며, 사용자들은 인터넷에서 자신이 필요로 하는 정보를 찾아서 활용을 하게 된다. 사용자들은 정보를 요구하는 분야가 서로 다르고 취향 또한 다양하기 때문에 웹에서 얻고자 하는 정보도 서로 다르다. 따라서 웹에서 얻어지는 정보를 제공할 때 사용자에 따라 다른 형태의 정보를 가공하여 보여주는 것이 필요하다.

정보를 얻고자 하는 사용자의 다양한 요구에 응하기 위해서는 서로 다른 프로그램들간의 상호협력을 통하여 보다 나은 결과를 내는 것이 필요하다. 본 논문에서는 좀 더 합리적인 방법으로 웹 상에서 정보를 얻기 위하여 여러 종류의 프로그램, 즉 서로 다른 기능을 가진 에이전트들을 사용하여 위와 같은 문제점을 해결해보려고 한다.

따라서 본 논문에서는 데이터베이스를 이용하여 다양한 사용자들의 욕구를 동시에 충족시키기 위한 멀티 에이전트 시스템 모델을 제안한다. 데이터베이스를 기반으로 하여 사용자들이 원하는 웹페이지에 있는 정보들을 웹 로봇을 이용하여 모은 뒤에 데이터베이스에 저장한다, 그 후 사용자에 맞는 인덱스를 작성하여 사용자들이 자신이 원하는 정보를 손쉽게 찾고, 관리할 수 있도록 하는 것에 목적을 둔다.

* 본 연구는 방위사업청과 국방과학연구소의 지원으로 수행되었음

2. 관련연구

본 장에서는 기본적으로 널리 알려진 웹로봇과 멀티에이전트 시스템에 관한 내용들을 설명한다.

2.1 웹 로봇

웹 로봇은 웹사이트를 순회하며 HTML 정보를 읽어들여서 웹사이트가 가지고 있는 정보를 자동으로 수집하는 프로그램이다. 웹 로봇은 자신이 읽어들인 HTML의 정보 중에 다른 URL로 연결된 링크가 있을 경우에 해당 URL들로 이동하면서 정보를 수집하게 된다.

웹 로봇이 다른 URL로 이동하면서 정보를 얻게 될 때는 몇 가지 사항을 고려하게 된다. 이전에 방문을 했는가에 대한 여부, 처음 정보를 읽어들인 웹 페이지로부터 몇 단계에 걸쳐서 방문하게 되었는지 등을 기록하여 정보를 읽어들이게 된다. 웹 로봇은 브라우저가 아니라 프로그램이다. 따라서 웹 로봇이 읽어들이는 데이터는 HTML을 그대로 읽어들이게 된다. 해당 웹 페이지가 자바 스크립트를 활용했을 경우에는 링크되어 있는 다른 웹 페이지를 검색하지 못하고 그대로 종료되는 경우가 생길 수 있다.

URL 링크가 여러 개가 있을 경우에 웹 로봇은 해당 링크를 큐(Queue)에 저장하게 된다. 링크가 저장될 때 큐에는 해당 URL, 처음 웹 페이지로부터의 단계를 저장하게 된다. 이렇게 큐를 사용하여 링크들을 저장할 때 웹 로봇은 해당 URL이 만족해야 하는 조건들 - prefix, suffix, include, exclude -을 사전에 검사하게 된다. 위의 조건들이 일치하게 되면 웹 로봇은 큐에 링크들을 저장하

고 그 후에 큐에 있는 링크들을 차례대로 검사하게 된다.

웹 로봇 자체적으로 데이터를 검사할 수 있지만 사용자들의 다양한 기호를 만족시키기에는 충분하지 않다. 따라서 본 논문에서는 웹 로봇과 함께 다른 에이전트들을 사용하여 목적을 달성하려고 한다.

2.2 멀티 에이전트 시스템

멀티 에이전트 시스템은 복잡한 문제를 해결하기 위한 하나의 대안으로서 연구되었다. 하나의 에이전트만으로는 문제 해결이 어려울 경우에 여러 에이전트들이 상호협력을 함으로써 보다 더 나은 결과를 창출하는 것에 도움을 얻을 수 있게 된다. 즉, 하나의 에이전트가 복잡한 문제를 해결하기 위한 모든 기능을 포함하지 않고 여러 개의 여러 개의 에이전트들이 문제 해결 기능을 나눠가지게 되는 것을 멀티 에이전트 시스템이라고 한다.

멀티 에이전트 시스템은 서로 다른 에이전트들간의 상호 협력에 의하여 동작한다. 이러한 상호 협력은 분산 환경에서의 데이터를 다루는 데에 있어서 좀 더 나은 환경을 제공하게 된다.

웹 로봇에 여러 가지 설정을 더하여서 사용자가 원하는 데이터들만을 수집하는 것은 수많은 사용자들에게 각기 다른 프로세스를 제공해야 한다는 문제점을 안고 있다. 따라서 여기에서는 웹 로봇이 가지고 있는 기능과 사용자들이 필요로 하는 기능을 분리하여 각기 다른 에이전트 시스템을 구축한다. 여기에 데이터베이스를 추가하여 정보의 체계적인 관리가 가능해지고 수많은 사용자들을 동시에 관리하면서 사용자들의 요구에 대응하는 것이 가능해진다.

3. 멀티 에이전트 시스템의 구성

3.1 시스템 구조

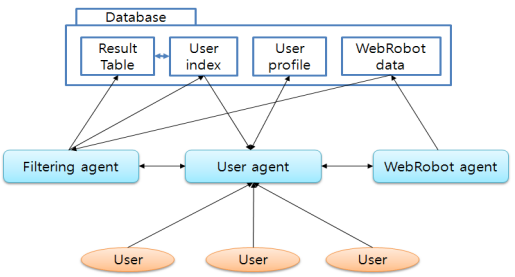

그림 1. 시스템 구조

본 논문에서 제시하는 멀티 에이전트 시스템은 그림 3.1과 같은 시스템 구조를 가지고 동작한다. 각 에이전트들은 자신들의 기능과 관련된 해당 데이터베이스와 연결되어 동작하게 된다. 사용자들은 에이전트를 통하여 자신이 원하는 정보를 획득하고, 확인할 수 있게 된다.

이 시스템은 크게 자료를 저장하는 데이터베이스 부분과 사용자가 원하는 서비스를 제공하는 에이전트들로 나눌 수가 있다.

데이터베이스는 사용자가 필요로 하는 정보들이 저장되는 공간이다. 여기에서는 데이터베이스의 몇 가지의 테이블을 기본적으로 구성하고 있다.

3.2 데이터 베이스 구조

Field	Type	Null	Key	Defaults
url	varchar(256)	NO	PRI	NULL
title	varchar(256)	YES		NULL
ctime	char(15)	YES		NULL
mtime	char(15)	YES		NULL
content	text	YES		NULL
gtime	char(15)	YES		NULL

표 1. Web Robot data table

Web Robot data 테이블은 웹 로봇 에이전트가 수집한 정보를 저장하는 테이블이다. 이 테이블은 정보를 url(수집한 URL), title(해당 URL의 제목), content(내용)을 기본적으로 기록하며 마지막으로 gtime(웹 로봇 에이전트가 해당 정보를 수집한 시간)을 기록하게 된다. ctime, mtime은 해당 정보가 URL에 기록된 시간을 나타낸다. 이 자료는 웹 로봇 에이전트가 아닌 다른 에이전트들이 분석하는 데에 이용이 된다. gtime은 사용자가 해당 자료를 검색한 뒤에 필요없다고 판단한 뒤 자료를 자신의 인덱스에서 삭제하게 될 경우에 필터링 에이전트가 다시 해당 자료를 불러들이는 것을 막기 위하여 사용된다.

Field	Type	Null	Key	Defaults
position	varchar(20)	YES		NULL
index_list	varchar(100)	YES		NULL

표 2. User profile table

User profile 테이블은 사용자의 정보를 기록한다. 해당 필드들은 position(사용자의 정보), index_list(사용자가 원하는 정보들)로 구성이 된다. 사용자의 다양한 기호를 기록하기 위하여 필드들은 중복되어 기록될 수 있다. 데이터를 분석할 때 사용자의 정보, 즉 position의 정보를 기준으로 하기 때문에 동시에 여러 개의 정보를 분석하는 것이 가능하다. position은 사용자 아이디와 같은 의미를 가지게 되지만 중복해서 기록이 되므로 서로 다른 레코드가 같은 position을 보유하게 되면 같은 사용자가 여러 개의 쿼리를 등록했다는 것을 알 수 있다.

Field	Type	Null	Key	Defaults
id	int(11)	NO	PRI	0
words	varchar(1000)	YES		NULL

표 3. User index table

User index 테이블은 사용자에게 제공되는 테이블이다. 즉, 웹 로봇이 모은 데이터를 분석한 뒤에 해당 데이터의 id와 데이터가 포함하고 있는 단어들의 목록을 보여주는 역할을 한다. 즉, 이 시스템에서는 하나의 결과 테이블을

가지고 있고 사용자들은 자신이 원하는 쿼리와 관련된 인덱스 테이블만을 보유하고 있다. 수많은 사용자들에게 자료테이블을 제공하는 것은 하드웨어적인 측면에서 불필요한 공간을 낭비하게 된다. 따라서 이 시스템에서는 사용자들에게 쿼리와 해당 쿼리에 대응하는 자료들의 아이디를 제공함으로써 데이터 공간을 효율적으로 사용한다.

Field	Type	Null	Key	Defaults
id	int(10) unsigned	NO	UNI	NULL
url	varchar(256)	NO	PRI	NULL
title	varchar(256)	YES		NULL
content	text	YES		NULL
summarize	varchar(1024)	YES		NULL
gtime	char(15)	YES		NULL

표 4. Result table

Result 테이블은 결과를 저장하는 테이블이다. 사용자의 기호에 따라 웹 로봇이 수집한 데이터를 분석하고 난 뒤에 필요 없는 데이터는 버리고 사용자가 필요한 데이터만을 저장하게 된다. 기본적으로 웹 로봇이 자료를 모은 Web Robot Data 테이블과 같은 형태를 취하고 있다. 여기에서 gtime은 해당 자료를 분석한 시간을 기록한다. Web Robot Data 테이블의 gtime과 Result 테이블의 gtime을 비교하게 되면 웹 로봇이 모은 자료를 중복해서 분석하는 상황을 예방하는 것이 가능해진다.

3.3 세부 에이전트

3.3.1 Web Robot agent

웹 로봇 에이전트는 기존의 웹 로봇과 같은 기능을 한다. 사용자가 원하는 데이터가 있는 웹페이지에서 정보를 가져온다. 가져온 정보들은 데이터베이스의 Web Robot data 테이블에 기록이 된다.

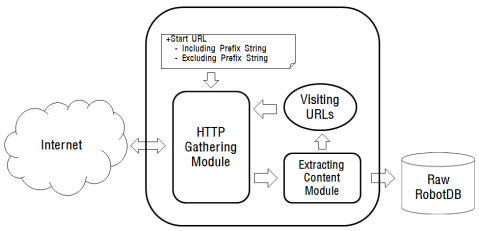

그림 2. 웹 로봇 에이전트 구조

웹 크롤링을 제외한 부가 기능들은 다른 에이전트 시스템이 가지고 있다. 따라서 해당 웹 로봇은 기본적인 웹 크롤링 기능만을 제공한다. 여기에서는 큐를 사용하여 사용자가 원하는 특정 사이트에서 일정 깊이만큼의 링크를 따라다니며 자료를 모으게 된다. 모은 자료는 Web Robot Data 테이블에 기록이 되며 기록된 자료는 다른 에이전트들에 의해서 분석이 되고 사용자가 원하는 형태로 가공이 되어 제공이 된다.

3.3.2 Filtering agent

필터링 에이전트는 웹 로봇 에이전트가 수집한 정보를 분석하는 작업을 한다. 데이터베이스에서 사용자가 요구하는 정보들을 읽어온 후에 분석하여 데이터베이스에 결과를 기록하고, 사용자 인덱스를 작성한다.

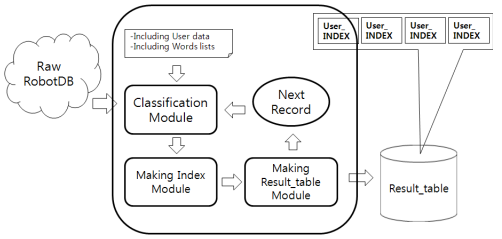

그림 3. 필터링 에이전트의 구조

필터링 에이전트는 자신의 역할을 수행하기 위하여 필수적인 몇 가지의 모듈을 보유하게 된다. 웹 로봇이 모은 자료를 사용자 쿼리를 사용하여 분석하는 작업을 하는 Classfication Module, 분석된 레코드를 사용하여 사용자에 맞는 인덱스를 작성하는 Making Index Module, 분석된 레코드들 중 사용자 요구에 맞는 레코드들만을 따로 저장하는 Making Result_table Module이 필수적인 모듈로서 존재하게 된다.

Classification Module은 데이터베이스를 차례대로 순회하며 필요한 정보들만을 추출한다. User profile 테이블이 가지고 있는 사용자들의 쿼리들과 비교하여 사용자들이 등록한 쿼리와 일치하는 데이터들만을 뽑아낸다. Classification Module이 추출한 자료들은 Result 테이블에 저장이 되며 이후 삭제된다. 즉, 이 모듈은 웹 로봇이 모은 자료들을 필요에 맞게 분류하고 삭제하는 역할을 수행한다.

Making Index Module은 Classification Module이 추출한 자료들에 대해 User Index 테이블에 해당 자료들의 ID를 기록한다. 해당 자료와 일치하는 쿼리를 보유하고 있는 모든 사용자들의 User Index 테이블에 해당 자료가 있음을 표시하는 작업을 진행한다.

Making Result_table Module은 위의 Making Index Module과 동시에 동작한다. 먼저 유일한 아이디를 부여받고 자료가 저장되면 이 아이디는 다른 테이블들에서 해당 자료를 식별하기 위한 기본 데이터가 된다.

위의 세 가지 모듈이 차례대로 웹 로봇이 모은 자료들을 분석하고 분석된 자료들은 결과 테이블에 저장이 된다. 하나의 레코드가 웹 로봇 데이터에 저장이 되면 위의 세 가지 모듈이 작동하게 되는데, 하나의 레코드를 다시 읽어들이는 일이 없이 한번에 모두 수행하게 된다.

3.3.3 User agent

유저 에이전트는 사용자의 편의를 제공한다. 사용자는 유저 에이전트를 통하여 프로그램에 접속하고, 또 사용할

수 있게 된다. 유저 에이전트는 사용자가 요구하는 기능들을 제공하게 된다.

웹 로봇 에이전트와 필터링 에이전트, 유저 에이전트는 해당 멀티 에이전트 시스템에 있어서 가장 기본적인 형태의 뼈대를 이루게 된다. 여기에 사용자들의 요구에 맞게 모은 자료들을 정기적으로 해당 사용자들에게 보내주는 메일링 서비스 등이 추가적으로 시스템에 포함될 수 있다.

4. 실행결과

본 논문의 에이전트 및 모듈들을 Java 환경에서 구축해 보았다. Swing Library를 사용하여 시각적으로 보여줄 수 있는 형태를 만들어 보았다.

그림 2. 로그인 화면

기본적인 형태의 로그인 화면이다. 사용자들은 여기에서 자신의 아이디와 해당 이메일을 기록하여 실질적인 유저 인터페이스를 사용하게 된다.

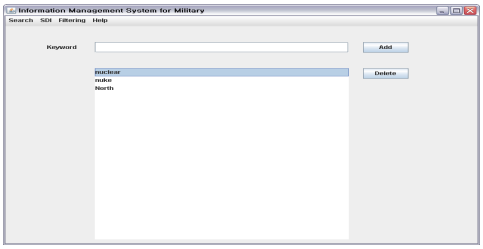

그림 3. 쿼리의 추가 및 삭제

위의 그림은 사용자가 자신의 쿼리를 관리하는 일종의 유저 인터페이스 화면이다. 해당 화면에서 사용자는 자신의 쿼리들을 편리하게 관리할 수 있게 된다.

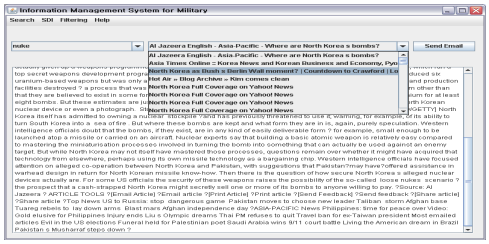

그림 4. SDI 서비스

위의 화면은 사용자가 자신의 쿼리에 맞는 내용들을 검색하여 자기 자신에게 이메일을 발송하는 그림이다. 좌측의 콤보박스에는 사용자가 등록한 쿼리들이 나타난다. 쿼리를 선택하면 우측의 콤보박스에 쿼리에 맞는 자료들의 제목이 나타나고 자신이 원하는 제목을 선택하면 아래의 화면에 해당 자료의 내용이 보인다. 제목을 선택한 뒤 이메일 발송 버튼을 누르면 사용자에게 해당 자료의 제목과 내용이 이메일로 발송이 된다.

웹로봇 같은 경우에는 화면 바깥에서 지속적으로 돌고 있다. 웹로봇은 사용자 로그인과는 별개이며 사용자는 웹로봇이 수집하는 정보에 대해서는 조작이 불가능한 형태로 디자인되어 있다.

5. 결 론

웹은 시간이 지날수록 보다 많은 데이터를 포함하며 더욱 방대해질 수 있다. 이 경우에 사용자들은 자신이 원하는 데이터를 찾기가 쉬워질 수 있겠지만, 반대로 너무 많은 정보들을 포함하고 있기 때문에 더욱 난해해 질 수도 있다.

본 논문에서는 데이터베이스를 이용한 멀티 에이전트 시스템으로 사용자의 편리를 추구하였다. 기존의 웹 로봇만으로는 사용자가 원하는 정보를 얻는 것에 부족함이 있기 때문에 다른 프로그램들과의 상호 협력을 통해 보다 더 나은 결과를 보일 수가 있다.

기본적인 기능들만을 제공하기 때문에 사용자가 원하는 정보를 제공하는데 있어 부족함이 있을 수 있다. 보다 더 나은 결과를 보이기 위해서는 단순히 키워드만이 아닌 사용자의 기호도 등을 사용하여 더욱 복잡한 연산을 처리하는 과정이 필요하다. 이러한 과정 또한 추가적인 에이전트의 사용으로 처리할 수 있다. 사용자가 더 많이 검색하는 단어 등을 구별한다면 인덱스 등의 제공에 있어서 보다 사용자 친화적인 구성이 가능할 것이다.

데이터베이스를 사용함으로써 사용자가 원하는 데이터들을 보다 잘 분류하는 것이 가능해질 수 있다. 사용자들에게는 보다 적은 공간을 활용하는 고유의 인덱스를 제공하게 되므로 자신의 원하는 정보의 관리에 있어서 편의성을 제공할 수 있게 될 것이다.

참고문헌

[1] Hongyu Liu "Probabilistic Models for Focused Web Crawling", Faculty of Computer Science, Dalhousie University

[2] Ying Xie, "Incorporating Agent Based Neural Network Model for Adaptive Meta-Search", The Center for Advanced Computer Studies University of Louisiana at Lafayette

[3] Hsin-Tsang Lee, ",IRLbot: Scaling to 6 Billion Pages and Beyond", Department of Computer Science, Texas A&M University

[4] BERNARD J. JANSEN, "Automated Gathering of Web Information: An In-Depth Examination of Agents Interacting with Search Engines", The Pennsylvania State University