

중요 여행 정보를 찾기 위한 지능 검색 시스템

윤은일*, 신현일**, 류근호*

Intelligent Retrieval System for finding important travel information

Unil Yun *, Hyeonil Shin **, Keun Ho Ryu *

요약

최근 주 5일 근무의 시행범위가 확대되면서 여가활동에 대한 관심이 증대되고 있다. 또한 인터넷과 모바일 인프라가 널리 보급되어 특정 정보를 필요로 하는 사용자들은 검색 엔진을 이용하여 원하는 정보를 얻을 수 있게 되었다. 그러나 공유되는 정보의 양이 급속하게 증가함에 따라, 검색되는 정보는 많지만 사용자가 원하는 정확한 정보만을 제공받기는 쉽지 않다. 단적인 예로, 여행 정보의 경우 사용자들은 자신이 필요한 정보를 검색하지만, 그 결과로는 수많은 여행 상품의 광고들을 보게 된다. 이 논문에서는 정보 수집 에이전트를 이용하여 여행 정보 검색 시스템(TIRS)을 설계하고 구현한다. 정보 수집 에이전트는 포털 사이트의 여행 관련 카테고리 페이지와 주요 언론사들의 여행 기사 페이지를 주기적으로 방문하여 여행과 관련된 정보를 수집하고, 수집된 정보를 데이터베이스에 저장한다. 이를 이용하여 사용자들은 여행에 관련된 정보를 얻기 위해 여러 광고들에 둘러싸일 필요 없이 TIRS의 질의 서버를 통하여 필요한 정보를 간편하게 검색할 수 있다.

Abstract

The increasing interest in leisure activities of a five-day work per week has been recently prevailed. Additionally, as internet and mobile infrastructures have been becoming widespread, the user can get specific information using a search engine. However, it is difficult for the user to get accurate information they really want as shared information has been rapidly increased and the information has been searched. For example, users can retrieve required travel information, but they also must see a huge number of travel advertisements. In this paper, we design and implement a retrieval system using travel information collecting agent. The information gathering agent regularly visits travel-related category pages of the portal sites and major media travel-article pages to collect information related to travel, and the agent stores the gathered information to a database. Then, users can search the travel information conveniently without the need to view advertisements.

• 제1저자 : 윤은일

• 투고일 : 2009. 07. 22, 심사일 : 2009. 08. 31, 게재확정일 : 2009. 09. 15.

* 충북대학교 전자정보대학 컴퓨터전공 교수 ** 충북대학교 전자정보대학 컴퓨터전공 석사

※ “이 논문은 2008년도 충북대학교 학술연구지원사업의 연구비지원에 의하여 연구되었음 (This work was supported by the research grant of the Chungbuk National University in 2008)”

▶ Keyword : 여행 정보 (Travel information), 검색 시스템 (Retrieval System), 웹 로봇 (Web robot), 검색 에이전트 (Retrieval Agent)

I. 서론

현존하는 웹 페이지들의 양은 실로 방대하여 세계 최대의 검색 엔진인 구글의 경우, 사용자의 검색 질의시 80억 개 이상의 웹 페이지에 접속하며, 하루 20PB이상의 웹 페이지 용량을 처리한다. 이처럼 인터넷상에는 엄청난 양의 정보가 있으며, 지금 이 순간에도 정보를 보유한 웹 사이트들의 양은 폭발적으로 증가하고 있다. 하지만 사용자가 필요로 하는 정보는 엄청난 양의 웹 데이터들 속에 산재되어 있기 때문에 일정한 제한조건을 바탕으로 하여 정보를 추출하는 작업이 필요하다. 정보를 획득하기 위하여 사용자들은 인터넷 정보 검색 엔진을 이용하는 방법을 사용한다. 하지만 일반적인 주제의 경우, 단순 검색으로도 만족할만한 결과를 얻을 수 있으나 상업적인 주제 검색의 경우는 정보보다는 광고와 판매에 목적을 두고 비용을 지불한 자료의 결과가 상위에 검색되기에 사용자의 만족을 얻어내지 못한다. 사용자는 그러한 키워드들에 대해서는 여러 사이트를 방문하여 상업성이 배제되었다고 생각되는 정보를 골라내야 하는 불편함이 따른다. 이러한 주제들 중 대표적인 것이 여행 정보이다. 주 5일 근무 시행과 여가 생활에 대한 관심 증대로 인하여 여행 정보를 필요로 하는 사용자들이 많아진 데 비해서, 대부분의 포털사이트의 여행 정보 카테고리의 경우 민간 여행사의 여행 상품 예약 판매를 주로 다루고 있다. 사용자들은 이러한 불필요한 정보 가운데서 필요한 정보만을 골라내기 위해서 오랜 시간을 소비해야만 한다. 본 논문에서는 이러한 상업적인 키워드의 하나인 여행 정보에 대한 검색 시스템을 설계하고 구현하여, 사용자의 여러 사이트를 방문해야 하는 번거로움을 줄일 수 있도록 한다. 여행 정보가 담겨있는 사이트와, 언론사의 여행 정보 기사들을 주기적으로 수집하여 저장하고, 검색페이지를 통하여 사용자에게 보여주게 된다. 본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 살펴보고, 3장에서는 전체적인 정보 검색 시스템의 구조에 대해 살펴본다. 4장에서는 본 논문의 여행 정보 검색 시스템의 설계 방법에 대해 기술한다. 5장에서는 설계된 시스템의 성능을 평가해보며, 마지막 6장에서는 결론을 맺도록 한다.

II. 관련 연구

월드와이드웹은 수백만 이상의 정보수집이 가능하기에 사용자들은 쇼핑, 연구, 오락, 웹 사이트 구축을 할 때 항상 인터넷을 사용한다. 이러한 인터넷에 흩어진 정보를 검색할 때 가장 먼저 사용하는 것이 검색 엔진이다. 검색 엔진은 동작의 특성에 따라 카테고리형 검색 엔진, 키워드 검색 엔진, 메타형 검색 엔진으로 구분된다. <표 1>은 각 검색 엔진의 특징을 보여준다 [8]. 이러한 검색 엔진을 이용하여 제공되는 일반적인 웹을 통한 여행 정보는 카테고리별로 분류하여 사용자에게 제공되므로 사용자는 간단한 키워드 검색이나 카테고리 검색을 통해 정보에 접근할 수 있게 되어 있다. 그러나 이러한 분류 방법은 사용자가 정보를 검색하기에 편리하지만 검색된 여행정보 모두가 사용자에게 유용한 것은 아니다. 또한 타 사이트와 교류가 거의 이루어지지 않기 때문에, 사용자가 사이트와 사이트를 이동하며 검색하는 과정을 반복해야하므로 정보 검색을 위해 드는 노력과 시간 및 비용을 무시할 수 없다. 따라서 여행자 개인의 특성에 적합한 개인화된 정보가 제공 [1, 3] 및 관리 [2] 되고 있다. 또한, 성능의 향상을 위해 여러 검색 엔진을 결합하여 이용하기도 하며 [4] 이러한, 이용자의 특성을 고려한 차별화된 정보는 모바일 디바이스를 통해서도 제공되고 있는등 [5, 6, 11] 다양한 정보검색기법이 이용되고 있다 [13, 14, 15].

표 1. 검색엔진의 분류
Table 1. Classification of Retrieval Engine

구분	특징
카테고리형 검색엔진	사이트들을 사전에 설정해 놓은 몇 가지 범주에 따라 분류해 두고, 사용자로 하여금 관련 대주제 ~소주제 순으로 범주를 구체화 시키도록 하여, 원하는 사이트에 도달할 수 있게 하는 방식이다. 찾고자 하는 정보의 정확한 분류만 알면 누구나 쉽게 정보를 찾을 수 있으나, 잘못된 분류 선택시 전혀 관계없는 결과가 나오며, 정보 검색에 비교적 많은 시간이 걸린다.
키워드형 검색엔진	로봇 에이전트를 사용하여 검색한 사이트들의 내용을 자동으로 인덱스 하여 데이터베이스를 구축하고, 사용자가 직접 질의어를 입력하여 원하는 정보를 얻는 방식이다. 정보의 보유량이 많고 직관적이지만, 연산자 및 옵션에 대한 사전 지식을 갖추어야 제대로 된 활용이 가능하다.
메타 검색엔진	하나의 사용자 입력으로부터 다수의 검색 엔진을 동시에 구독시키고, 각 검색 엔진에서 얻어진 결과를 통합하여 하나의 형태로 보여준다. 이용하기가 쉽고 광범위한 검색이 가능하지만, 더 많은 자원이 필요하고 검색 및 수행시간이 오래 걸린다.

일반적으로 여행정보란 여행지의 숙박시설이나 교통, 해당 지역의 볼거리, 먹거리 등의 정보를 사용자에게 제공하여 사용자가 해당 지역을 여행하기 전이나 여행시 불편하지 않도록 모든 정보를 사전에 전달하여 여행지에서의 편의를 도모하는 것을 목적으로 한다. 여행정보 서비스는 3세대로 구분하여 설명할 수 있다. 1세대는 독립적인 여행정보 서비스 형태로 사용자가 오프라인으로 여행사나 주변 친인척 또는 먼저 그 곳을 다녀온 사람들의 말을 듣고 여행지를 결정하고, 여행지에서도 해당 지역주민들의 도움을 구해야 했다면, 2세대는 인터넷의 보급으로 인하여 여행전이나 여행일정 중간에 한국관광공사나 민간차원의 항공사, 여행사, 호텔, 예약업체, 혹은 개인에게 정보를 얻는 것이다[7, 8, 9].

본 연구에서는 일반적인 컴퓨터와 인터넷을 이용한 2세대 여행정보 서비스와 모바일 기기의 웹 브라우저를 이용하여 접속할 수 있는 3세대 여행정보 서비스를 제공하게 될 것이다. <그림 1>은 사용자가 TIRS를 이용하여 여행 정보를 검색하는 전체적인 시스템의 구성을 보여주고 있다. TIRS는 사용자의 접속 방식을 체크하여 각각에 적합한 화면으로 검색 페이지를 제공한다. 검색 페이지는 로봇에 의해서 미리 수집되며 수집된 데이터는 분석되어 체계적으로 서버에 저장된다. 이후, 검색 페이지는 TIRS 서버에 저장된 자료를 바탕으로 하며, 이 자료들은 웹 로봇을 통하여 주기적으로 업데이트 되어 사용자에게 제공된다. 사용자 질의 시 TIRS 서버 질의 처리 에이전트에서는 저장된 여행 데이터를 조회하여 관련 데이터를 사용자에게 보여준다. 보다 자세한 시스템의 구조에 대해 다음장에서 설명된다.

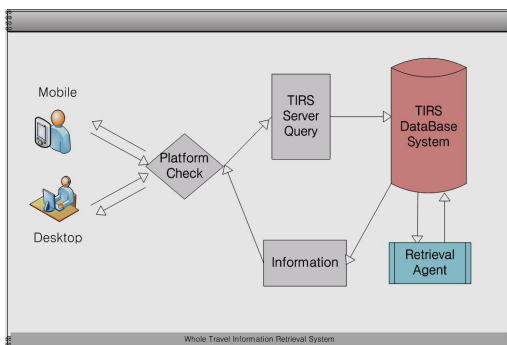


그림 1. 시스템 구성도
Fig 1. System Architecture

III. 검색 시스템 구조

검색 시스템은 <그림 2>와 같이 검색 에이전트(웹 로봇), 데이터베이스 시스템, 사용자 질의 서버의 세 부분으로 이루어져 있다. 검색 에이전트는 두 포털사이트의 여행 관련 카테고리인 '네이버 여행'과 '야후 거기', 언론사의 여행 정보 관련 기사의 웹 페이지들을 수집한다. 이렇게 수집된 웹 페이지들은 데이터베이스 시스템 상에 url address, keyword, title, pre-context, last modification date 등의 일정한 형식의 정보들로 인덱스 되어 저장된다. 이 때 keyword들과 url address 등의 자료들을 연관시켜주기 위해 역 인덱스 방법을 사용한다.

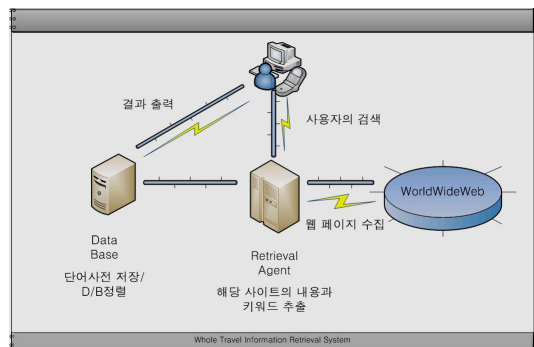


그림 2. 사용자 질의 서버
Fig 2. User query server

사용자 질의 서버는 이러한 일련의 과정들을 사용자들이 인지하지 않고도 일관된 인터페이스를 제공하여 손쉽게 정보를 검색할 수 있도록 해준다. 사용자의 접속 환경에 따라 일반 컴퓨터의 경우와 모바일 기기 접속(제한된 크기의 디스플레이)의 경우를 구별하여 인터페이스를 제공하게 된다.

IV. 여행 정보 검색 시스템의 설계

TIRS의 개발 환경은 <표 2>와 같다. 검색 에이전트의 경우 Python 2.5.2로 제작되었으며, D/B서버와 웹 서버는 MS Visual Web Developer 2005를 사용하였다. 검색 에이전트와 D/B서버의 자료 교환을 위하여 MS C# 2005가 사용되었다.

표 2. TIRS 개발 환경

Table 2. Implementation environment of TIRS

종 류	사 양
프로토콜	TCP/IP
운영체제	Windows XP SP3
D/B, 웹서버	MS Visual Web Developer 2005
사용언어	Python 2.5.2, MS C# 2005

4.1 웹 로봇

웹 로봇의 동작 원리는 <표 3>과 같다. 여기서 url주소의 이름들로 저장되는 파일들과, linklist.dat 파일은 웹 로봇의 다음 링크로의 진행을 위해 필요한 파일이며, url주소를 이름으로 하여 저장되는 파일들은 데이터베이스에 키워드 파일을 넘겨주기 위한 파일들이다.

표 3. 웹 로봇의 동작 원리

Table 3. Working principle of Web robot

단계	내 용
1	해당 페이지의 HTML소스를 읽어온다.
2	정규 표현식 함수를 이용하여 불필요한 부분 제거
3	키워드들이 해당되는 URL주소를 파일이름으로 저장
4	링크들이 해당되는 URL주소를 파일이름과 Linklist로 저장
5	Linklist를 따라서 계속해서 검색

프로그램의 모듈 구성은 다음과 같다. 모든 모듈은 utf-8 코딩 방식에 어긋나지 않게 한다. Main은 Robot모듈을 호출하고, Robot모듈과 그 하위 모듈인 Crawler모듈, Psave모듈 간에 호출과 리턴이 이루어진다. Main모듈은 시작 검색 페이지를 지정해주고, 그 페이지의 값을 Robot모듈로 넘겨주게 된다. Robot 모듈은 Main모듈에서 웹 페이지 주소를 넘겨받아서, 이를 바탕으로 Crawler모듈을 호출하여 그 결과 값을 변수 html에 저장한다. Crawler모듈은 사이트 주소에 대한 웹 페이지 소스를 읽어오는 함수이다. 그리고 사이트 주소를 파일이름으로 변환하기 위해, 사이트 주소에서 { /, ?, : }과 같이 파일이름으로 사용할 수 없는 단어들을 임시 단어로 치환한다. 그리고 강제로 리다이렉션 되는 웹 페이지의 경우는 올바른 페이지 주소를 가져오기 위해서, 리다이렉션의 표시인 *를 기준으로 앞부분을 삭제하는 작업도 필요하다. linklist.dat 파일에서 계속 연속된 링크를 불러오게 되며, 그 링크에 따라 계속해서 사이트의 웹 페이지 내용을 수집하게 되며, Psave모듈을 통하여 파싱하여 데이터를 저장할 수 있게 된다.

Pseudo code : Robot Module

```

Procedure Robot(site)
html:=Crawler module(site)
rename {?, *, /, :} in site
while 1
begin
file:=open(linklist.dat)
while 1
begin
site:=read per line(file)
if site=blank then break
rename {?, *, /, :} in site
start Psave module(site, html)
end
end
    
```

Crawler모듈은 Robot모듈에서 호출되어, 사이트 주소를 기본으로 해당하는 웹 페이지의 내용을 그대로 수집하는 역할을 한다. 해당 사이트를 불러오는데 에러가 없다면, 웹 페이지의 내용을 변수 filetext에 담아서 리턴하며, 타임아웃 에러 등의 경우에는 재접속하여 작업을 다시 수행하게 된다.

Pseudo code : Main Module

```

Procedure Main
begin thread
site:=startpage(each site)
start Robot module(site)
end
for i:=1 to 10
begin
start thread(i)
end
    
```

Pseudo code : Crawler Module

```

Procedure Crawler(site)
urlopentry:=true
while urlopentry
urlopentry:=false
filetext:=Crawling html(site)
if error then urlopentry:=ture
return filetext
end
    
```

Psave모듈은 수집해 온 웹 페이지의 내용을 구문 분석하여 이름이 url주소인 파일의 내용을 채우는 모듈이다. 먼저 내장 모듈인 BeautifulSoup를 불러와서 변수 html에 담겨 있는 웹 페이지 내용을 정리한다. 그 결과는 변수 soup에 저장시킨다. 그리고 키워드가 저장될 파일 이름이 될 변수 filename

과 링크들이 저장될 파일 이름이 될 변수 linkfile을 생성한다.

또한 데이터베이스에서 불러올 수 있도록 형식을 만들기 위해서 filename에 넣을 형식인 임시변수 content를 생성한다. 변수 html안에 들어있는 웹 페이지의 내용을 구문 분석한다. 줄바꿈, 수직탭, 수평탭, 불필요한 특수문자 및 태그들을 내장 함수인 정규식 변환 함수 're'를 이용하여 제거한다. 페이지의 제목을 저장하기 위해, 웹 페이지가 정리되어 저장된 변수 soup에서 <title> 태그가 있는 부분만 추출하여 변수 Title에 저장한다. 이 때, BeautifulSoup모듈에서 코딩 변환이 잘못된 경우를 대비하여, 다시 utf-8 코드로 변환시켜준다. 또한 자료형이 리스트 형식으로 추출되므로, 이를 문자열로 변환시키기 위해 join명령을 사용한다.

```
Pseudo code : Psave Module

Procedure Psave(site, html)
    filename:=str(site)+"txt"
    linkfile:=str(site)+"link"
    soup:=Structure analyzing html
    content:=The rest of the letter & number , except
    to delete in soup (ex) tag, special character, tab)
    title:=Select tag start with "<title>"
    fileopen(filename)
    filewrite(title, content)
    fileclose()
    links:=Select tag start with "<a href"
    if links not already exists then
        begin
            fileopen(linkfile, linklist)
            filewrite(links)
            fileclose()
        end
    end
```

이렇게 변환된 내용을 형식에 맞추어 이름이 url주소인 파일에 저장하게 된다. 또한 링크들을 위해서 <a> 링크가 있는 부분만을 추출한 뒤, 그 결과 중에서 <href> 링크를 다시 찾아내어 태그 뒤의 부분인 링크 주소를 같은 방식으로 추출한다. 이 결과를 url주소 파일 및 linklist.dat 파일에 저장하여 프로그램이 계속해서 실행 될 수 있게 한다. 단, 기존의 링크와 중복이 되지 않는 결과만을 저장하도록 한다. 이를 통하여 검색 로봇이 일정한 링크들 사이에서 무한 반복되는 것을 막을 수 있다.

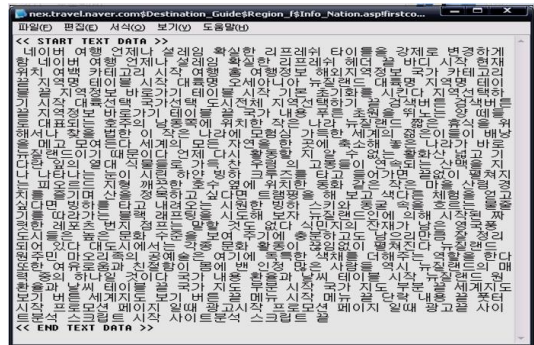


그림 3. 키워드 리스트
Fig 3. Keyword list

4.2 데이터베이스 시스템

DB의 구성은 다음과 같다. DB를 구성하기 위하여, Dictionary 및 URL Table를 생성한다. <그림 5>는 생성할 테이블의 구조를 보여준다.

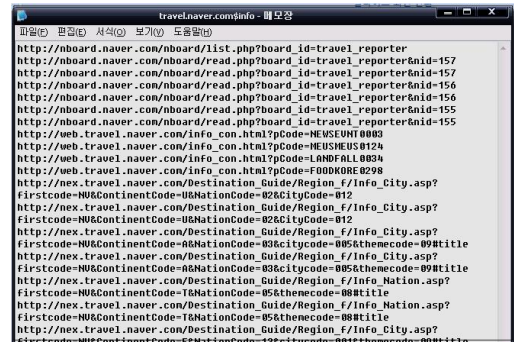


그림 4. URL 주소의 저장
Fig 4. Storing URL address

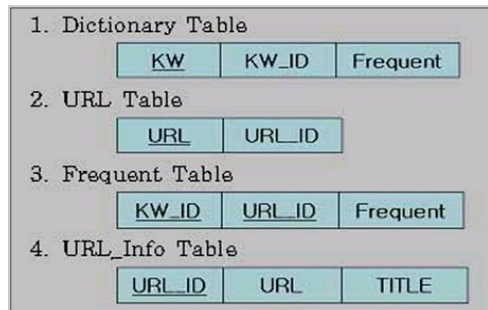


그림 5. 데이터베이스 스키마
Fig 5. Database schema

테이블을 생성하기에 앞서 먼저 로봇이 추출한 문서들의 파일명을 읽어온다. 각 파일의 이름은 URL주소로 되어있다. 그리고 URL에서 몇몇 특수문자는 다른 문자로 치환되어 있기 때문에 정상적인 URL을 얻기 위해 다음과 같이 치환해준다.

```
ex) / -> $, .txt 제거
치환 전 : 211.214.103.10$Service$weat-
             her$weather.asp.txt
치환 후 : 211.214.103.10/Service/weat-
             her/weather.asp
```

다음으로, 정상 URL을 URL Table에 넣고, URL마다 URL_ID를 부여해서 같이 저장한다. 파일의 세 번째 줄이 키워드 집합이므로 그것을 읽어서 Dictionary에 저장한다. 이때, 키워드마다 키워드ID를 부여해서 같이 저장한다. 또한 키워드마다 빈도수를 조사하여 같이 저장한다. 그리고 빈도수 1인 키워드는 Dictionary Table에서 삭제한다. 다음은 Frequent 및 URL Info Table을 생성한다. 먼저, 키워드와 URL을 ID를 이용하여 저장한다. 이는 실질적인 검색시 이용되는 Table이며, 해당 키워드가 있는 URL들을 검색할 수 있다. 각 키워드, URL 쌍에 대한 빈도수를 저장함으로써 랭킹을 구현하였다. 가령 '서울'(ID : 1)이라는 키워드를 검색해서 얻은 결과가 (1,2,4), (1,3,6), (1,4,2)라면 빈도수가 큰 순서로 3,2,4의 순서로 페이지를 보여준다.

URL	URL_ID
airbooking.tour.daum.net/insurance/instinfo/orm.bp	1
airoutlet.tour.daum.net/AEVE/verf/areListNEW_bp?FL...	2
airoutlet.tour.daum.net/AEVE/verf/areListNEW_bp?FL...	3
asset.hani.co.kr	4
asset2.hani.co.kr/board/board_view.php?board_id=10...	5
asset2.hani.co.kr/board/board_view.php?board_id=10...	6
asset2.hani.co.kr/board/board_view.php?board_id=10...	7
bbs.hani.co.kr/board/aj_hani_ajnu/Contents.asp?S7a...	8
bbs.hani.co.kr/board/aj_hani_ajnu/list.asp?S7abtema...	9
bbs.hani.co.kr/Board/aj_hkr_ajnu/Contents.asp?S7abl...	10
bbs.hani.co.kr/Board/aj_hkr_ajnu/Contents.asp?S7abl...	11
bbs.hani.co.kr/Board/aj_hkr_ajnu/list.asp?S7abtema...	12
bbs.tour.daum.net/bbs/view.php?d=daum_notice&pa...	13
bbs.tour.daum.net/bbs/view.php?d=daum_notice&pa...	14
bbs.tour.daum.net/bbs/zboard.php?d=daum_notice...	15
bbs2.hani.co.kr/Board/info_recrut_bbs/Contents.asp?...	16
bbs2.hani.co.kr/Board/info_recrut_bbs/list.asp?S7abl...	17
bizstart.yahoo.co.kr/changup_item/chain_view.asp?C...	18
blog.hani.co.kr/bonbon/10235	19
blog.hani.co.kr/bonbon/10894	20
blog.hani.co.kr/bonbon/10912	21
blog.hani.co.kr/gaunhan/17182	22
blog.hani.co.kr/nuhkr/13963	23
blog.hani.co.kr/westmin/14504	24
blog.hani.co.kr/yh.in/5789/17185	25
blog.kakimew.com/index.asp	26
blog.paran.com/blogindex/bloginfo.php?cate=1	27
blog.paran.com/blogindex/bloginfo.php?id=603&cate...	28
blog.paran.com/blogindex/bloginfo.php?id=605&cate...	29
blog.paran.com/blogindex/bloginfo.php?id=606&cate...	30
blog.paran.com/blogindex/bloginfo.php?id=607&cate...	31

그림 6. URL 테이블
Fig 6. URL table

페이지를 사용자에게 보여줄 때 URL_Info Table을 이용한다. Frequent Table에서 얻은 관련 URL_ID를 이용한 질의 연산을 통해 URL_Info 테이블에서 URL과 TITLE정보를 얻는다.

```
ex) URL_ID로 질의
- Frequent Table (1,2,10)
- URL_Info Table
(2,www.travel.com, 여행사이트)
```

사용자 질의시에는 불리언 질의를 기본으로 하였다. 이는 AND 와 OR 연산을 지원한다. 또한, 사용자가 검색한 키워드와 선택한 연산종류를 바탕으로 SQL 쿼리를 생성한다. 범위 검색을 위하여 와일드카드문자 '%'를 사용했으나, AND 쿼리시 검색속도가 상당히 저하되었다. 그래서 와일드카드문자 '_'를 사용하여 검색범위를 앞뒤 한글자로 제한하여 문제를 해결하였다. 수행성능이 향상되었으며, 검색결과가 다소 줄어들긴 하였지만 차이가 크지 않았다.

4.3 사용자 질의 서버

사용자 질의 서버는 사용자의 검색 작업이 일어나는 페이지로 이 사이트의 핵심 페이지라 할 수 있다. 사용자 질의 서버는 접속자의 브라우저 버전을 확인하여 사용자가 PDA나 핸드폰 등의 모바일 기기로 접속했는지, 일반 PC로 접속했는지에 따라 맞춤형 페이지를 제공하게 된다. 접속자의 브라우저 종류와 버전을 체크하여 모바일기기 접속자에게는 낮은 해상도의 화면을 보여주고, 일반 PC접속자에게는 해당하는 해상도의 화면을 보여주게 된다. HTML의 User-Agent 헤더 값에는 클라이언트의 브라우저를 비롯한 각종 정보가 담겨 있다. 이 User-Agent 헤더값은 접속되는 웹 브라우저마다 다르기 때문에 브라우저의 종류를 확인한 뒤, 낮은 버전을 사용하는 모바일 기기와 일반 PC를 구별하여 다른 시작페이지를 제공하면 된다. 텍스트 박스에 질의를 입력하고 Search 버튼을 누름으로써 검색 작업이 시작된다. 라디오 버튼으로 AND 와 OR을 두어 불리언 질의 검색을 지원하게끔 하였다. AND 를 선택하고 텍스트 박스에 두 질의어를 입력했을시 해당 키워드가 모두 나오는 문헌만 검색 작업이 일어나고, OR을 선택했을시에는 키워드 중 하나라도 들어가는 문헌을 찾아 아래 datalist 영역에 결과를 출력해준다.(디폴트는 OR로 설정)

출력된 결과의 URL은 하이퍼링크로 속성을 지정하여 특정 URL을 클릭하면 해당 웹페이지로 이동할 수 있고, 텍스트박스에 입력된 질의어에 따라 바로 아래부분에 위치한 레이

어에 '~에 대한 검색 결과입니다.'를 출력하도록 설정하였다. <그림 7>은 질의어에 대한 검색이 완료된 결과이다.

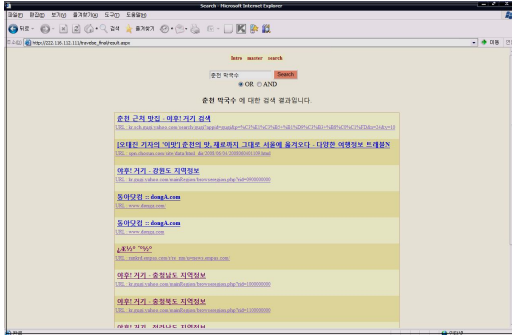


그림 7. TIRS의 검색 결과
Fig 7. Retrieval result of TIRS

VI. 성능 평가

본 논문에서는 중요 여행정보를 효율적으로 검색하고 관리하는 중요 정보 검색 엔진을 개발하였다. 본 섹션에서는 성능 평가를 위해 기존의 정보 검색엔진과 비교 평가한다. 먼저, 검색엔진이 갖추어야할 요건을 살펴보면 다음과 같다.[5]

- 많은 정보를 가지고 있을 것
- 데이터베이스 내용 및 갱신 주기가 빠를 것
- 검색 속도가 빠를 것.
- 검색 옵션의 다양성 및 편리한 사용자 환경을 제공할 것

검색 엔진의 주목적은 정확한 검색이다. 이를 위해서는 여러 검색 옵션이 제공되어야 한다. 각 검색엔진들은 다양한 검색 조건을 이용한 복합 검색 기법을 활용하고 있다. 불리언 연산은 AND, OR, NOT 각각의 옵션과 이들의 조합으로 검색이 가능하다. 완벽한 검색을 수행했다라도 그 결과가 보기 쉽게 나타나야 원하는 정보를 파악하기 쉽다. 카테고리 검색이 로봇 검색에 비해 나은 장점은 사람이 분류해 놓았기 때문에 간략한 설명이라도 볼 수 있고, 정렬 순서의 파악이 용이하다는 점이다. 또한 검색 결과를 한 화면에 몇 개까지 나타낼 것인가, 분류 항목과 검색 결과가 함께 표시되는가 하는 점 등도 주요 체크 포인트이다.

본 논문의 TIRS와 주요 포털사이트의 여행 정보 내용과의 기능별 성능 평가는 <표 4>와 같다. 네이버 여행 정보와 TIRS가 키워드 검색을 지원하고 다음과 엠파스 여행 정보는

카테고리형 검색만을 지원하였다. 본 논문에서 제시된 TIRS는 특정 여행 데이터에 대한 효율적인 검색에 초점을 맞추어서 디자인 및 구현을 하였다. 본 검색엔진은 상업적으로 개발된 검색엔진에서의 단점이 될 수 있는 부분을 극복하고자 개발되었으며, 다음, 네이버 그리고 엠파스등에서 제공하는 여행 정보 검색은 넘쳐나는 광고를 볼 필요 없이 필요한 여행 정보를 효율적으로 찾아볼 수 있다.

표 4. TIRS 성능 비교 평가
Table 4. Performance comparison of TIRS

	다음 여행정보	네이버 여행정보	엠파스 여행정보	TIRS
에이전트	X	O	X	O
카테고리	O	O	O	X
연산자	X	O	X	O
단어확장	X	O	X	X
통합검색	X	O	X	O
속도	O	O	O	O
일관성	O	O	O	O

그에 따라 연산자와 단어 확장 등의 키워드형 검색에서 가능한 옵션은 사용이 불가능하였다. 모든 여행 정보 사이트는 광고성이 짙었으며, 특히 다음과 엠파스 여행 정보의 경우는 여행 업체의 정보를 사용하고 있었다. 모든 사이트가 D/B에 저장된 내용을 보여주기 때문에 속도는 빠른 편이었으며, 일관성 있는 인터페이스도 갖추어있다.

V. 결론

인터넷의 급속한 성장과 보급으로 정보 스트림을 통해 제공되는 정보의 양은 엄청나며, 그 종류도 뉴스 정보, 상품 카탈로그, 영화 스케줄 등 매우 다양하다. 하지만 이처럼 웹을 통해 접근 가능한 테라바이트 수준의 정보량에 비해 실제 사용자 개개인이 필요로 하는 정보는 극히 일부분이다. 이로 인해 생산성 증대, 교육적 이득, 오락적 가치 등 인터넷을 이용하는 이점이 위협받을 수 있다. 이러한 현실 속에 인터넷의 많은 정보 중에서 필요한 정보만을 사용자에게 제공할 수 있는 에이전트의 필요성은 증대되고 있다. 이런 필요성에 따라 검색 엔진을 이용하는 사용자들의 수와 검색 엔진에 대한 상업적인 이용도 늘어나고 있다. 그로 인해 필요한 정보를 찾고자 하는 사용자들은 자신이 찾은 정보가 광고와 판매를 위한 상업적인 것인지 아닌지를 판단해야 하는 부담을 갖게 된다.

본 논문에서는 상업적인 주제인 여행 정보에 대한 검색 엔진인 TIRS를 설계하고 구현하였다. 포털 사이트의 여행 정보 카테고리화 주요 언론사의 여행 기사 페이지의 내용을 수집하여 자체 데이터베이스에 저장하여 사용자들이 그 내용을 검색하여 정보를 얻을 수 있게 하였다. AND나 OR의 불리언 검색도 지원하여 두 가지 이상의 키워드에 대해서도 동시에 검색이 가능하도록 하였다. 향후 연구 과제로는 검색된 자료의 내용을 분석하여 정보의 신뢰성을 더 높이고, DB를 분석하여 카테고리형 검색을 지원하며, AND, OR이 아닌 '와', '과', '또는' 등의 연산자 확장 검색을 지원하게 하며[12], 단어사이의 인접도 [10]를 고려한 질의 방법도 향상시키는 것이다.

참고문헌

- [1] S. Chelcea, G. Gallais, and B. Trousse, "A personalized recommender system for travel information," Proceedings of the 1st French-speaking conference on Mobility and ubiquity computing, pp. 143-150, 2004.
- [2] C. C. Liu, C. H. Huang, W. T. Chu, and J. L. Wu, "ITEMS: intelligent travel experience management system," Multimedia Information Retrieval pp. 291-298, 2007.
- [3] S. Shekhar, A. Fetterer, and D. R. Liu, "Genesis: An Approach to Data Dissemination in Advanced Traveler Information Systems," IEEE Data Eng. Bull. Vol 19, No. 3. pp. 40-47, 1996.
- [4] R. W. White, M. Bilenko, and A. P. Heath, "Enhancing web search by promoting multiple search engine use," Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, pp. 43-50, 2008.
- [5] 강기순, 윤용익, "여행정보 서비스를 위한 모바일 에이전트 미들웨어," 개방형지리정보시스템학회논문지, 제 6권, 제 2호, 85-95쪽, 2004년 12월.
- [6] 고은정, 김여정, 김운, 강지훈, "지역 온톨로지를 이용한 지능형 여행정보 제공 시스템," 한국정보과학회학술발표논문집, 제 31권, 제 1호(B), 610-612쪽, 2004년, 4월.
- [7] 김윤정, 방혜미, 김명주, "유비쿼터스 컴퓨팅 환경에서 여행 정보 제공 시나리오 및 개인 정보보호 지원 기술 연구," 한국정보보호학회지, 제 16권, 제 2호, 46-52쪽, 2006년 4월.
- [8] 전진욱, 배인한, "정보수집 에이전트를 사용한 어린이 교육정보 검색시스템의 설계 및 구현," 한국인터넷정보학회지, 제 3권, 제 2호, 97-108쪽, 2002년 4월.
- [9] 조미영, 최창, 김판구, "온톨로지를 이용한 맞춤형 여행정보 시스템," 한국컴퓨터종합학술대회논문집, 제 33권, 제 1(B)호, 187-189쪽, 2006년 6월.
- [10] 윤성운, 채기진, 이상훈, "질의 내부 단어 인접도를 이용한 검색 효율 향상 기법," 정보과학회논문지, 제 35권, 제 2호, 191-198쪽, 2008년 4월.
- [11] 이양원, "시맨틱 웹에 기초한 모바일 관광정보 서비스," 대한지리학회지, 제 42권, 5호, 788-807쪽, 2007년 12월.
- [12] 이재성, "메타 검색에서 외래어 질의 정제 효과," 한국정보처리학회논문지, 제 15권, 제 2(B)호, 171-178쪽, 2008년 7월.
- [13] 양형규, 이강호, 최종호, "국내 검색엔진을 이용한 개인 정보 해킹에 관한 연구," 한국컴퓨터정보학회논문지, 제 12권, 제 3호, 195-201쪽, 2007년 7월.
- [14] 송재원, 김덕환, 이주홍, "공간 위치 정보를 적합성 피드백을 위한 가중치로 사용하는 영역 기반 이미지 검색 시스템," 한국컴퓨터정보학회논문지, 제 11권, 제 4호, 1-7쪽, 2006년 9월.
- [15] 하창승, 정이상, "내용기반 검색을 이용한 선박매매 정보추출 에이전트의 구현에 관한 연구," 한국컴퓨터정보학회논문지, 제 12권, 제 1호, 43-50쪽, 2007년 3월.

저 자 소 개



윤 은 일

1997: 고려대학교 이학석사.
 1997 - 2006 :
 한국통신 멀티미디어연구소 전임/선임
 연구원.
 2005 :
 Texas A&M Univ. 공학박사
 2005 - 2006:
 Texas A&M Univ. 포스닥연구원.
 2006 - 2007:
 한국전자통신연구원, 선임연구원.
 2007 - 현재:
 충북대학교 전자정보대학 컴퓨터전공
 조교수
 관심분야 : 데이터마이닝, 정보검색,
 데이터베이스



신 현 일

2009. 8 :
 충북대학교 컴퓨터공학 학사.
 2009. 9 - 현재 :
 충북대학교 컴퓨터공학 석사
 관심분야 : 데이터마이닝, 정보검색,
 데이터베이스



류 근 호

1976 : 숭실대학교학공 공학사.
 1997 : 연세대학교 공학석사.
 1980 - 1983 :
 한국전자통신연구원 연구원.
 1983 - 1986 :
 한국방송통신대학교 조교수.
 1988: 연세대학교 공학박사
 1986 - 현재:
 충북대학교 전자정보대학 컴퓨터전공
 교수
 관심분야 : 데이터베이스, 데이터마이
 ning, 바이오인포매틱스