웹 문서 검색을 위한 한글 MG 시스템의 확장

김범수°, 나연묵
단국대학교 컴퓨터공학과
{toto, ymnah}@dankook.ac.kr

An Extension of Hangul MG System for Web Document Retrival

Bumsoo Kim° Yunmook Nah
Dept. of Computer Engineering, Dankook University

요 약

최근 국내 인터넷 발전과 더불어 검색엔진들의 수요가 많아지고 있어 크고 작은 검색엔진들이 많이 개발되고 있다. 기존의 디지털 라이브러리에 사용되고 있는 정보 검색 엔진인 한글 MG 시스템을 웹 문서 검색에 적용하는데는 어려움이 있었다. 본 논문은 한글 MG 시스템을 기반으로 웹 사이트의 내부 문서 검색이 가능한 소형검색엔진으로 확장하는데 필요한 웹 로봇에 의한 문서 수집, 수집된 문서의 가공, 메타 데이타의 데이타베이스화, 단락 대 문서 사상, 문서 검색을 위한 질의 루틴의 수정과 웹 검색 및 시스템 관리 인터페이스에 대한 방안들을 제안하여 확장 시스템을 설계하고 구현하였다.

1. 서론

최근 국내에는 웹 문서 검색을 위한 대형 검색 엔진들의 개발이 활발해 지고 있어 1995년부터 시작된 국내 정보 검색 엔진의 시대가 전성기를 맞이하고 있다고 해도 과언이 아니다. 이에 힘입어 웹사이트 내부의 문서만 전용으로 색인하는 소형 정보 검색 엔진들도 그 수가 증가하고 있는 실정이다. 본 논문에서는 한글 MG 시스템을 웹 문서 검색을 위한 소형 검색 엔진으로 확장하는데 필요한 방안들을 제안하였다.

한글 MG 시스템은 텍스트 데이타베이스의 검색을 위해 역 리스트(inverted list) 기법을 사용하며 역 리스트 기법의 단점인 기억 공간 오버헤드를 인덱스 압축과 스킵 기술을 이용하여 대폭 감소시킬 뿐만 아니라 디스크 입출력량의 감소로 고속 한영 검색을 제공하는 정보 검색 시스템이다[1].

웹 검색 엔진으로 확장하는데 있어서 기존의 한글 MG 시스템의 문제점들은 알아보면 웹 로봇의 부재, 단락기반색인(index on paragraphs of a documents), 웹 문서의 가공, 웹 검색 인터페이스 및 관리자 인터페이스의 부재를 들 수 있다. 웹 로봇의 부재는 시스템에서 가장 선행되어야 하는 웹 문서 수집 작업의 부재를 의미한다. 단락기반색인이란 로컬 드라이브의 특정 디렉토리 내 텍스트 문서들을 압축하고 색인하는 과정에서 각각의 문서마다 고유지정자를 부여하는 것이아니라 문서 내의 각각의 단락마다 고유지정자(identifier)를 부여하게 되는 것을 말한다.색인어를 검색하게 되면 색인어를 포함하는 단락의 고유지정자들을 검색결과로 가져오게 되지만 그 단락을 포함하는 문서는 알아낼 수 없다. 그 밖에 웹 문서의 메타데이타를 추출/저장하거나 웹 문서의 가공을 통하여 웹 문서 자체의 태그 필터링을 수행하는 과정이 없으며, 웹에서 검색을 할 수 있는 웹 검색 인터페이스와 시스템 제어를 위한 시스템 관리 인터페이스도 없다.

2장에서는 한글 MG 시스템을 웹 검색 엔진으로 확장하는 방안을 설명하고, 3장에서는 세부적인 설계 및 구현에 대해 설명한다. 4장에서는 결론과 향후 연구 과제를 제시한다.

2. 한글 MG 시스템의 확장 방안

한글 MG 시스템을 웹 검색 엔진 시스템으로 개선하는 방안 중 우선적으로 선행되어야 하는 것은 웹 로봇의 추가 설계, 웹 로봇에 의해 수집된 웹 문서로부터 메타 데이타를 추출하여 저장하는 모듈의 개발, HTML 태그 및 클라이언트측 스크립

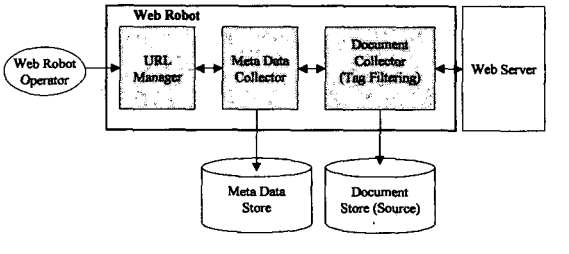

그림 1. 메타 데이타 추출/저장 및 웹 문서의 저장

본 연구는 한국과학재단의 특정기초연구과제 (과제번호 : 98-0102-06-01-3) 연구비 지원에 의한 것임.

트 제거 모듈의 추가를 통해 웹 문서의 수집과 수집된 웹 문서를 색인 대상 원시 텍스트 문서(source)로 변환하는 과정이다. 메타 데이타는 웹 문서의 URL, 문서크기, 최종변경시각, 문서 타입 등을 나타내며 위 과정을 거치면 데이타베이스에 저장된다. 웹 로봇의 추가로 인한 장점은 HTTP 프로토콜 기반의 GET 방식을 사용하기 때문에 웹 상에서 스크립트 페이지를 접근하여 페이지를 요청하면 응답된 결과는 스크립트 실행결과인 HTML 형태를 가지게 되는 것이다. 이는 색인 대상 문서에서 스크립트 형태의 페이지도 포함시킬 수 있다는 의미로써 게시판 등의 스크립트 실행으로 얻어지는 문서를 색인할 문서에 포함시킬 수 있게 된다.

단락기반색인을 위한 개선 방안은 단락지정자를 문서지정자로 사상시키는 자료구조를 추가 설계하고 한글 MG 시스템의 질의 처리 루틴에 문서지정자목록을 획득하는 모듈을 추가 구현하는 것이다. 구체적으로 설명하면 문서는 단락들을 포함한다. 단락기반색인 시스템은 원시 텍스트 문서 내의 단락마다 고유한 단락 지정자를 부여한다. 단락 대 문서 사상 개념은 고유한 지정자를 가진 단락들이 어떤 문서에 포함되는가라는 것이다. 그림 2는 한글 MG 시스템에 추가적으로 구현한 단락 대 문서 사상을 간단하게 도식화한 것이다. 원시 텍스트 문서는 색인 대상이 되는 문서들의 목록을 나타낸다. 문서들은 단락들로 이루어져 있으며 단락마다 고유지정자를 가지고 있다. 먼저 처리되는 문서 내의 단락들이 앞선 지정자를 가지게 된다. 원시 텍스트 문서가 색인과정을 거치면 색인어별 단락지정자의 목록으로 구성된 역 리스트가 생성된다. 단락 대 문서 사상 과정을 거치면 단락지정자별 문서지정자의 목록으로 구성된 단락 대 문서 사상 파일(이하 사상파일)이 생성된다.

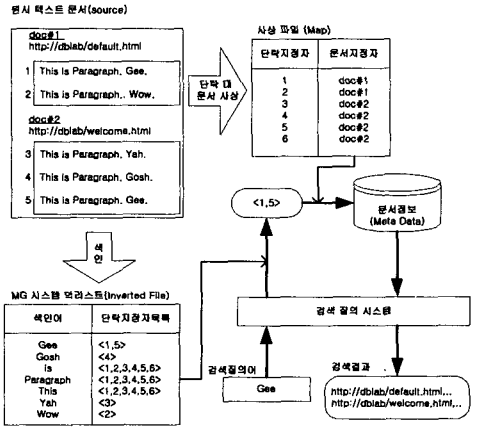

그림 2. 단락 대 문서 사상

한글 MG 질의 시스템에서 "Gee" 라는 검색질의어를 처리하는 과정은 역 리스트 파일을 통해 색인어 "Gee"를 포함하는 단락지정자목록 <1>, <5>를 검색하고, 사상파일에서 <1>, <5>를 포함하는 문서지정자 <doc#1> ,<doc#2>를 검색한 후, 마지막으로 문서정보 테이블에서 <doc#1>, <doc#2>의 관한 문서정보를 검색하여 결과로 출력하는 순으로 진행된다.

검색 인터페이스는 WWW를 통한 서비스를 제공하기 위해 CGI 기술을 이용하여 구현하고 시스템 관리 인터페이스는 유닉스 쉘 상에서 동작하도록 구현한다.

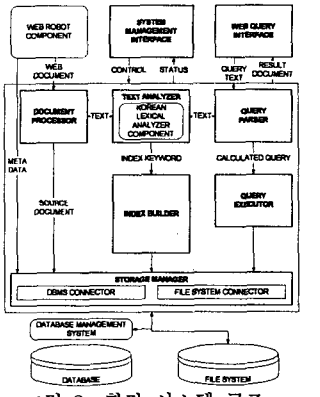

그림 3. 확장 시스템 구조

그림 3은 한글 MG 시스템을 확장하여 설계한 확장 시스템의 최종 구조이다. 웹 로봇은 웹 문서를 수집하여 태그 필터링을 통해 원시 텍스트 문서로 변환하고 수집된 웹 문서로부터 메타 데이타를 추출한다. 생성된 원시 텍스트 문서와 추출된 메타 데이타는 저장관리(storage manager) 모듈을 통해 데이타베이스나 파일 시스템에 저장된다. 문서 처리(document processor) 모듈은 한글 MG 시스템의 텍스트 압축 과정을 수행하는 기능을 한다. 텍스트 분석(text analyzer) 모듈은 비단어, 영어, 한국어를 구분해 내고 언어에 따른 색인어 스태밍을 수행하는 모듈이다. 인덱스 생성(index builder) 모듈은 한글 MG 시스템의 인덱스 생성 과정을 수행한다. 질의 파서(query parser) 모듈은 LALR(1) 파서 생성기인 YACC에 의해 생성된 파서 모듈이고 질의 실행(query executor) 모듈은 파서 모듈과 함께 웹 검색 인터페이스로부터 입력받은 검색질의를 처리하여 검색결과를 출력하는 기능을 한다. 시스템관리 인터페이스는 위에서 언급한 전반적인 과정을 제어하며 시스템 상태를 모니터링하는 기능을 수행한다.

3. 확장 시스템 설계 및 구현

3.1 웹 로봇

```
GET /dbm2/members.asp HTTP/1.1          // 메타 데이타
Accept: text/plain,text/html,*/*;q=0.3
Host: 203.237.228.115
Referer: ./menu.asp
User-Agent: W3CRobot/5.2.8 libwww/5.2.8
HTTP/1.1 200 OK
Server: Microsoft-IIS/5.0
Date: Mon, 03 Jul 2000 17:19:02 GMT
Content-Length: 4168
Content-Type: text/html
<html>                                  // 웹 문서
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ks_c_5601-1987">
<title>단국대학교 데이타베이스 &amp;amp; 멀티미디어 연구실 소개</title>
<meta http-equiv="Content-Type" content="text/html; charset=EUC-KR">
<h4><font color="teal">지도 교수</font></h4>
~ 중략 ~
</html>
```

그림 4. 웹 로봇에 의해 수집된 members.asp 문서

웹 로봇 모듈에서는 웹 문서 수집과 동시에 태그 필터링을 통해 원시 텍스트 문서를 생성하며 웹 문서로부터 추출된 메타 데이타를 기반으로 문서정보를 생성한다. 웹 로봇 모듈에 태그 필터링 과정을 수행하기 위한 매크로 모듈과 문서정보를 생성하는 부분을 추가로 설계하고 구현한다. 그림 4는 웹 로봇이

수집한 웹 문서의 예이다. 여기서 메타 데이타 부분을 추출하여 데이타베이스에 저장하게 된다. 표 1은 문서정보를 저장한 데이타베이스 테이블이다. 테이블로부터 검색되어진 문서정보는 질의 처리 모듈에서 DocEntry 구조체에 저장되어 검색결과로 출력된다.

표 1. 문서정보 테이블

필드	데이타타입	설명
DocID	int	문서지정자
DocURL	varchar(128)	문서URL
Title	varchar(128)	문서제목
Length	int	문서길이
LastUpdate	datetime	최근변경시각
Charset	varchar(16)	문자셋
Content_Type	varchar(16)	문서타입

3.2 단락 대 문서 사상

한글 MG 시스템은 앞에서 언급했던 텍스트 압축 과정과 색인 생성 과정을 가진다. 간단한 역 리스트를 사용하는 시스템에서는 역 리스트 파일과 원시 텍스트 파일만을 접근해 검색을 하는 반면에, 한글 MG 시스템은 인덱스 공간 압축과 고속 검색 수행을 위해 추가적인 파일을 사용하고 있다. 이러한 파일들은 색인어에 대한 사전 파일들, 랭킹 질의 시에 사용되는 가중치 파일들과 위의 두 과정을 수행하는 프로그램 파일들이다. 이에 더하여 단락 대 문서 사상을 수행하려면 추가적인 파일들이 필요하다. 단락 대 문서 사상 관련 파일들 생성하는 과정은 텍스트 압축 과정 중에 진행된다. 표 2는 사상 파일 포맷이다.

표 2. 단락 대 문서 사상 파일 포맷

필드	길이	데이타타입	설명
MagicNo	4 Bytes	unsigned long	매직키
DocID	4 Bytes	unsigned int	문서지정자

MagicNo는 사상파일을 인식하기 위한 데이타이고, DocID는 문서지정자를 나타내는 고유한 4바이트 정수형 데이타이며 단락지정자 수만큼 반복하여 기록된다. 단락지정자와 문서지정자는 1부터 시작하여 1씩 증가하는 데이타이다. DocID는 파일 내의 단락지정자에 해당하는 오프셋에 기록되게 된다. DocID를 기록하는 오프셋 계산공식은 다음과 같다.

오프셋 = sizeof(MagicNo) + (sizeof(DocID) × 단락지정자)

예를 들어, 단락지정자의 해당 문서지정자가 기록되는 오프셋은 MagicNo (4Bytes)를 지나 단락지정자 크기만큼 4×5=20 바이트 간격을 더 간 위치로 파일 시작 포인터로부터 총 24바이트 뒤이다. 이러한 고정된 레코드 구조를 이용하여 단락 대 문서 사상에 소요되는 디스크 I/O 비용을 줄일 수 있다.

3.3 질의 처리 루틴 수정

한글 MG 시스템의 질의 시스템은 불리언(boolean) 질의, 랭킹(ranking) 질의와 단락지정자로 직접 해당 단락을 검색하는 질의를 지원한다. 그림 5는 한글 MG 시스템의 질의 처리 루틴을 문서 검색이 가능하도록 변경한 것이다.

볼드체로 표시된 GetDocList() 부분은 단락 대 문서 사상을 검색에 적용하기 위해 구현한 부분으로 단락정보목록을 인수로 전달받아 문서정보목록를 획득한다. 하단을 보면 출력결과가 문서정보목록(DocList)과 단락정보목록이라는 것을 알 수 있다. 검색된 단락정보목록은 ParaList형 구조체 포인터 PL에 저장된다. 실제로 DocList와 ParaList는 query_data라는 구조체의 일부분이다. 그림 6은 query_data와 DocList 구조체이다.

```
Loop
  Get query
   Case query-type of
     Boolean : do Boolean query --> ParaList  // 불리언 질의
     Ranked : do Ranked query --> ParaList    // 랭킹질의
     ParaID : do ParaID query --> ParaList    // 단락지정자 질의
   End-Case
End-Loop
GetDocList(ParaList) --> DocList              // 추가된 부분
Display DocList, ParaList
```

그림 5. 확장 시스템의 질의 처리 루틴

그림 6에서 볼드체로 표시된 DocList형 구조체 포인터 PL은 추가된 자료형이며 Num은 검색되어진 문서의 수를 나타내고, DocEntry형 구조체 DE는 연결리스트(Linked-List)의 첫 번째 문서정보노드를 연결하는데 이는 문서정보를 읽어들인 308바이트 크기의 내용을 포함한다.

```
typedef struct query_data {
        DocList *DL;          //문서정보목록 포인터(추가된 부분)
        ParaList *PL;         //단락정보목록 포인터
} query_data;
typedef struct DocList {
        int Num;
        DocEntry DE[1];
} DocList;
```

그림 6. query_data 구조체

3.4 구현환경

사용된 환경으로는 SunOS 운영체제가 탑재된 SPARC 시스템이며, 컴파일러는 gcc를 사용하고, MG 시스템 1.2.1 버전과 형태소 분석기 HAM 4.8 버전이 결합된 한글 MG 시스템, MySQL 3.22 버전과 W3C에서 공개한 wwwlib 5.2.8 기반의 웹 로봇인 Webbot을 사용하였다.

4. 결론 및 향후 연구 과제

본 논문은 한글 MG 시스템 기반해서 웹 문서 검색을 위한 확장 시스템을 설계하고, 시스템 확장 방안으로 제안한 웹 로봇 모듈, 웹 문서의 메타 데이타 추출 및 저장 관련 모듈, 웹 문서 가공 모듈, 단락 대 문서 사상 관련 모듈, 질의 처리 모듈들을 구현하였다. 향후 과제로는 한글 MG 시스템 1.0 버전에서 보다 좀더 효율적인 한글코드처리 기법 연구가 요구된다.

5. 참고 문헌

[1] 박미란, 나연묵, "대용량 한글 텍스트 검색엔진 HMG의 구현", 멀티미디어학회 논문지, 제1권 제2호, 한국멀티미디어학회, 1998년 12월, pp.162-172.

[2] A.N. Vo, A. Moffat, "Compressed Inverted Files with Reduced Decoding Overheads," SIGIR' 98, Melbourne, Australia, 1998.

[3] J. Zobel, et al., *Indexing Technique for Advanced DataBase System*, Kluwer Academic Publishers, 1997.

[4] 오세만, *[개정판] 컴파일러 입문*, 정익사, 1994.

[5] I.H. Witten, A. Moffat, and T.C Bell, *Managing Gigabyte Second Edition: Compressing and Indexing Documents and Images*, Morgan Kaufmann, 1999.

[6] MG Homesite, http://www.mds.rmit.edu.au/mg.

[7] Webbot - The Libwww Robot, http://www.w3c.org/Robots.