효율적인 홈페이지 관리를 위한 구조 분석 시스템의 설계 및 구현

최봉준, 박규석
경남대학교 컴퓨터공학과

Design and Implementation of A Structure Analyzer for Efficient Homepage Management

Bong Joon Choi Kyoo Seok Park
Dept. of Computer Engineering, Kyungnam University

요 약

네트워크기술의 발달로 인하여 인터넷 이용자의 급증과 함께 그에 따른 정보를 제공하기 위한 인터넷 홈페이지도 기하급수적으로 증가하게 되었다. 인터넷 홈페이지의 운영 기간이 증가함에 따라 홈페이지를 구성하는 웹 문서와 웹 컨텐츠의 수도 증가하게 된다. 따라서, 이러한 웹 문서와 웹 컨텐츠에 대한 관리를 위하여 WCMS(Web Contents Management System)이 도입되고 있지만, WCMS에서는 웹 컨텐츠의 생성, 출판, 관리가 주요 기능으로, 웹 컨텐츠의 사용 횟수, 웹 문서의 다운로드 속도, 웹 문서 다운로드 용량, 데드 링크 여부 등을 분석하고 관리해 주는 소프트웨어가 필요하게 되었다.

본 논문에서는 웹 문서를 시각적으로 분석하여 웹 컨텐츠를 추출하고 웹 문서 내에서의 웹 컨텐츠 위치 및 크기를 분석한 후, 웹 컨텐츠의 연결 가능 여부를 분석하여 데드링크일 경우, 시각화하고, 웹 컨텐츠가 얼마나 사용되는지 등 홈페이지를 효율적으로 관리할 수 있는 구조 분석 시스템을 설계 및 구현하였다.

1. 서론

인터넷의 사용이 급증하면서 그 사용자의 다양한 욕구를 충족시키기 위한 서비스를 제공하는 홈페이지의 경우, 2003년 현재 전 세계적으로 약 4000만 개의 웹서버와 약 30억 개 이상에 이르는 웹 문서로 서비스되고 있다[2][3].

인터넷 홈페이지의 개설과 함께 증가된 웹 컨텐츠를 효율적으로 관리하기 위해 WCMS(Web Contents Management System)을 도입하여 웹 컨텐츠의 생성, 출판, 관리를 효율적으로 수행하고 있으나, 이러한 WCMS는 웹 컨텐츠 자체에 대한 관리를 효율적으로 수행하기 위한 방법으로 웹 문서의 시각적인 크기, 웹 컨텐츠의 다운로드 속도, 웹 문서의 다운로드 용량, 데드 링크의 여부 등 홈페이지 이용자의 측면에서 분석하고 관리해 주는 시스템이 필요하게 되었다.

인터넷의 홈페이지의 개설과 폐쇄가 자유롭게 이루어지고 있어 웹 문서에서 하이퍼링크로 연결되어 있는 다른 홈페이지의 웹 컨텐츠가 웹 문서 생성 당시에는 연결이 가능하였으나, 일정 시간이 경과되면 홈페이지의 폐쇄로 인하여 연결할 수 없는 데드링크가 발생하게 된다.

따라서 홈페이지 관리자는 주기적으로 하이퍼링크를 분석하여 데드링크가 발생하였는지 확인하여야 하며 데드링크가 발생하였을 경우, 이를 웹 문서에 적용하여 해당 하이퍼링크가 존재하지 않는 링크임을 표시하거나 하이퍼링크를 삭제하는 등의 관리를 필요로 한다.

이러한 유지보수는 홈페이지 개설 초기에는 적은 비용으로 관리가 되어지나 시간이 경과함에 따라 웹 문서를 생성한 홈페이지 관리자도 해당 하이퍼링크를 찾아 수정하는데 상당한 시간이 소요되어 이로 인한 홈페이지 관리비용이 증가하게 된다.

본 논문에서는 웹 컨텐츠의 위치와 크기를 측정하고 하이퍼링크가 연결되지 않는 데드링크가 발생하였

을 경우, 이를 시각화하고 웹 문서 다운로드 속도, 웹 문서 다운로드 용량 등을 분석함으로서 홈페이지를 효율적으로 관리하기 위한 방법에 대해서 논의한다.

본 논문의 구성은 다음과 같다. 먼저, 2장에서는 홈페이지 구조를 분석하기 위한 기초 기술인 웹 로봇에 대해서 기술하고 3장에서는 웹 컨텐츠의 위치 및 크기를 측정하고 구조를 분석하는 시스템을 제안한다. 그리고, 4장에서는 구조를 분석하기 위해 웹 컨텐츠의 위치 및 크기 측정 방법을 구현하고 웹 문서의 구조를 시각화함으로서 홈페이지를 효율적으로 관리할 수 있는 시스템을 구축한다. 마지막으로 5장의 결론에서는 향후의 연구과제를 제시한다.

2. 웹 로봇

웹 로봇이란 원하는 정보를 얻기 위해 웹 상의 문서들을 검색하고, 참조되는 문서들을 재귀적으로 검색하면서 웹의 하이퍼텍스트 구조를 자동으로 추적하여 정보를 저장해 주는 프로그램을 말한다.

웹 로봇은 일반적으로 Spider, Web Crawler 등으로 불리기도 한다. 웹 로봇은 자동적으로 웹의 하이퍼텍스트 구조를 따라 다니며 문서를 추출하고, 재귀적으로 그 문서에 참조되는 다른 문서들을 추출하는 방식으로 동작하는 프로그램으로 정의된다. 인터넷에서 정보검색 서비스를 제공하려면 웹 문서를 수집해야 하는데, 웹 문서를 어떻게 수집하느냐에 따라 검색 결과도 크게 달라진다. 웹 로봇의 순회 방법에 따라 넓이 우선 순회(Breath-First Traversal)와 깊이 우선 순회(Depth-First Traversal)로 나뉘어 볼 수 있다.
웹 로봇들은 통계 분석, 유지 보수, 미러링, 리소스 발견 등에 이용된다. 또한 웹 로봇은 상대 시스템에 과다한 부하를 초래하거나, 웹 로봇의 동작 오류, 다녀온 URL 검사시 웹 로봇 성능의 저하 등의 문제를 발생시킨다[5].

3. 홈페이지 구조 분석 시스템 설계

그림 1은 본 논문에서 제안한 구조 분석 시스템의 구성도이다.

웹 문서 다운로더는 웹 컨텐츠 데이터베이스로부터 분석할 웹 문서 목록을 가져와서 웹 문서를 다운로드 한다. 다운로드한 웹 문서의 다운로드 속도를 측정한 후, 웹 문서 표현기에 의해 웹브라우저와 동일하게 표현한다.

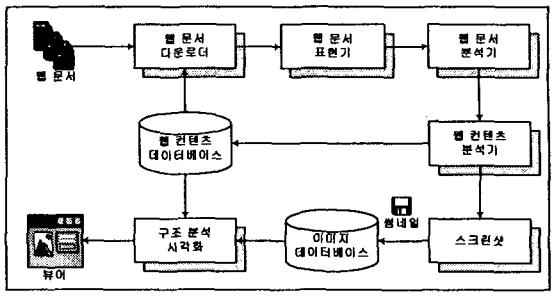

그림 1 시스템 구성도

웹 문서 분석기는 웹 문서를 parsing하여 이미지, 플래시, 애플릿 등의 웹 컨텐츠와 하이퍼링크를 추출하고 웹 컨텐츠와 하이퍼링크가 표현된 위치와 크기를 분석하고, 가로와 세로의 크기를 픽셀 단위로 분석한다.

웹 문서에 대한 분석이 완료되면 웹 컨텐츠 분석기에 의해 추출된 웹 컨텐츠와 하이퍼링크에 대한 연결 가능 여부 및 웹 컨텐츠의 파일 타입, 파일 크기에 대한 분석을 마친 후, 웹 문서에 대한 다운로드 용량을 분석한다.

웹 문서 분석기와 웹 컨텐츠 분석기에 의해 분석된 정보를 웹 컨텐츠 데이터베이스에 저장한다.

스크린샷은 웹 문서에 대한 썸네일 화면을 제공하기 위하여 웹 브라우저와 동일하게 표현된 웹 문서를 캡쳐하여 이미지 데이터베이스에 저장하고 데드링크와 같이 연결이 불가능한 웹 컨텐츠와 하이퍼링크에 대한 화면을 캡쳐하여 이미지 데이터베이스에 저장하는 기능을 수행한다.

구조 분석 시각화는 웹 문서의 구조를 시각적으로 표현하기 위하여 썸네일 이미지를 보여주고, 웹 문서의 다운로드 속도, 파일 크기, 다운로드 용량 등의 기초 정보와 가로 세로의 크기를 픽셀 단위로 분석한 웹 문서의 크기 정보를 제공한다.

분석된 웹 컨텐츠와 하이퍼링크의 위치 및 크기 정보를 이용하여 웹 컨텐츠의 파일 타입, 파일 크기 등의 분석 정보를 제공하고, 하이퍼링크로 연결된 웹 문서나 웹 컨텐츠에 대한 분석 정보도 함께 제공된다.

데드링크의 관리를 위하여 데드링크에 대한 위치 파악이 쉽도록 시각화된 정보와 웹 컨텐츠의 파일 타입, 파일 크기 등의 분석 정보를 제공한다.

그림 2는 홈페이지에 대한 구조 분석 과정을 나타낸 것으로 11 단계에 의해 홈페이지의 모든 웹 문서와 웹 컨텐츠에 대하여 반복적으로 분석함으로써 완료된다.

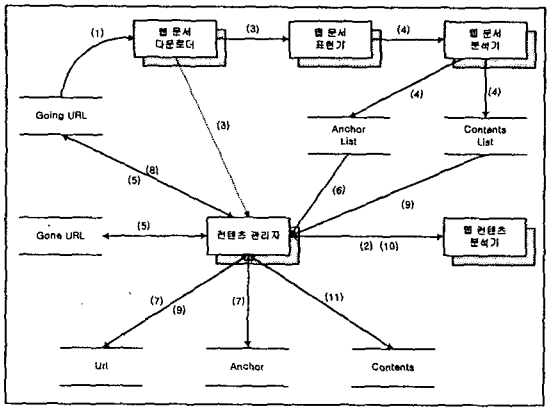

그림 2 홈페이지 구조 분석 과정

◆ 홈페이지 구조 분석 단계

단계 1. Get Going URL
→ Going URL을 읽어온다.

단계 2. Analysis Contents for URL and Save
→ URL에 대한 컨텐츠 타입, 파일크기, 존재 여부를 분석하고 저장한다.

단계 3. Download or Goto 1
→ URL이 text/html이면 다운로드한다. 만약 데드링크면 저장하고 1을 수행한다.

단계 4. DOM Analysis(Anchor List, Contents List, Exposure)
→ 다운로드가 완료되면 URL을 분석하여 Anchor List와 Contents List를 추출한다.

단계 5. Save Gone / Delete Going
→ URL 분석이 완료되면, 현재 분석한 URL을 Gone URL에 저장하고, Going URL에서 삭제한다.

단계 6. Registered URL, Get URL ID for Anchor List
→ URL Table에 없는 URL을 등록하고 ID를 할당받는다.

단계 7. Save Anchor List
→ Anchor List에 대하여 노출위치 및 노출 크기 정보를 Anchor Table에 저장한다.

단계 8. Add Going URL
→ Going URL, Gone URL에 존재하지 않는 Anchor를 Going URL에 추가한다.

단계 9. Registered URL, Get URL ID for Contents List
→ Contents List를 읽어와 URL Table에 없는 목록을 URL로 등록하고 ID를 할당받는다.

단계 10. Analysis Contents for Contents List
→ Contents List에 대하여 컨텐츠 타입, 파일크기 및 존재 여부를 분석한다.

단계 11. Save Contents List
→ 분석한 Contents List에 대하여 노출위치 및 노출 크기 정보를 Contents Table에 저장한다.

4. 구현

그림 3은 본 논문에서 제안한 구조 분석 시스템의 실행 화면으로 현재 분석중인 웹 문서의 파일 타입, 파일 크기, 접속 속도를 분석하고 웹 문서에 포함된 이미지, 플래시 등의 웹 컨텐츠에 대한 파일 타입, 파일 크기 등을 분석하며, 하이퍼링크의 접속 가능 유무를 분석하여 접속이 불가능한 데드링크의 경우 데드링크된 하이퍼링크의 위치 크기를 분석하여 화면을 캡쳐한 후, 데이터베이스에 저장하는 기능을 수행한다.

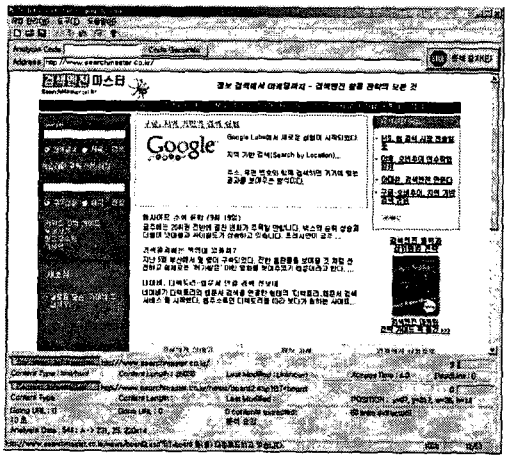

그림 3 구조 분석 시스템의 구현 화면

그림 4는 구조 분석 시스템에 의해 생성된 웹 문서 정보를 HTML로 나타낸 것으로 웹 문서 제목, 컨텐츠 형태, 파일 크기, 다운로드 크기, 다운로드 시간, 문서 크기, 최종 변경 날짜, 웹 컨텐츠 수, 하이퍼링크, 참조 횟수 등을 나타낸 것으로 웹 컨텐츠에 대한 개별 분석 및 하이퍼링크에 대한 개별 분석 등의 정보도 제공한다.

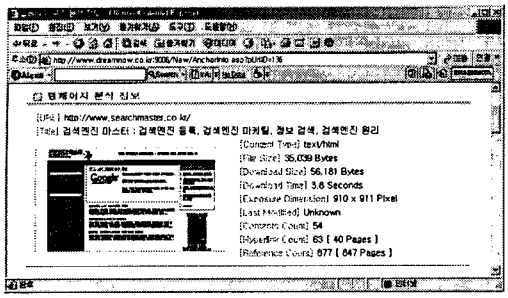

그림 4 웹 문서 분석 정보

그림 5는 구조 분석 시스템에 의해 분석된 정보 중 데드링크가 포함된 웹 문서의 목록을 나타낸 것으로 웹 문서의 파일 타입, 파일 크기, 다운로드 크기, 접속 속도, 노출 크기, 데드링크 수 등의 정보를 제공한다.

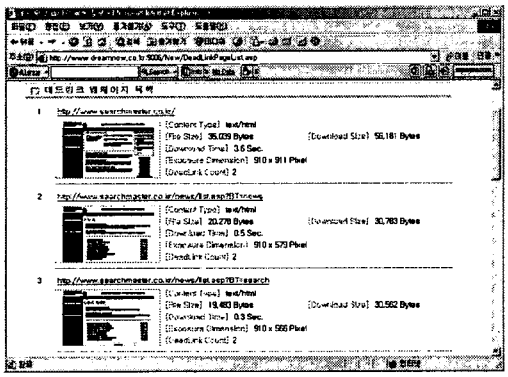

그림 5 데드링크가 발생된 웹 문서 목록

그림 6은 본 논문에서 제안한 구조 분석 시스템에서 분석한 데드링크 정보를 시각적으로 표현한 화면으로 데드링크의 위치 및 크기를 픽셀로 분석한 후 캡쳐할 화면의 위치 및 크기를 설정한 후 화면을 캡쳐하여 이미지 데이터베이스에 저장한 다음, 저장된 데이터베이스를 조회하여 캡쳐된 이미지와 함께, 데드링크의 위치 및 크기 정보를 이용하여 시각적으로 표현한 화면이다.

그림 5와 그림6에 의해 제공된 데드링크를 포함한 웹 문서의 목록과 웹 문서에 포함된 데드링크의 시각화 정보를 이용함으로서 데드링크의 효율적인 관리가 가능해 진다.

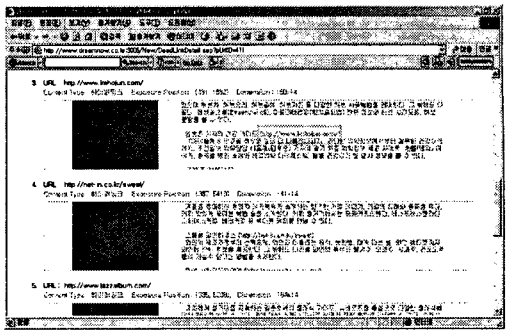

그림 6 데드링크의 시각적 표현

5. 결론

본 논문에서는 더욱 복잡하고 다양해져 가는 인터넷상의 홈페이지를 효율적으로 관리하기 위하여 웹 문서를 분석하고 웹 컨텐츠 및 하이퍼링크에 대한 정보를 구조적으로 표현하며, 웹 문서 내에 포함된 데드링크를 탐지하여 이를 시각화함으로써 기존의 텍스트 기반에 의존한 데드링크 관리 보다 효율적으로 관리가 가능하게 되었다. 따라서, 본 논문에서 제안한 구조 분석 시스템을 홈페이지의 기능적 관리에 대한 데드링크 시각화 및 효율적인 홈페이지 구조 분석이 가능하도록 개선하며, 향후 연구 방향으로는 페이지 랭킹 알고리즘에서 웹 문서 내의 하이퍼링크의 위치 및 크기에 따른 웹 문서의 인기도 분석에 대한 연구가 필요하다고 하겠다.

[참고문헌]

[1] http://msdn.microsoft.com/workshop/samples/author/dhtml/overview/measure.htm

[2] http://www.netcraft.com/

[3] http://www.google.com/

[4] Suhit Gupta, Gail Kaiser, David Neistadt, Peter Grimm, "DOM-based Content Extraction of HTML Documents", Proceedings of the twelfth international conference on World Wide Web, May 20-24, 2003, pp.207-214

[5] 김 일, "지역정보망을 위한 실시간 제어 검색엔진의 설계 및 구현", 경남대학교 석사학위 논문, 1999.2