Journal of the Research Institute of Industrial Technology. Vol. 23, 2006

웹 접근로그 분석을 통한 개선된 웹 문서 구조 추출

박철현* · 이성대* · 곽용원* · 전성환* · 박휴찬**

*한국해양대학교 대학원, **한국해양대학교 IT 공학부 교수

Improved Extraction of Web Document Structure augmented with Web Access Log

*C. H. Park** · *S. D. Lee** · *Y. W. Kwak** · *S. H. Jeon** · *H. C. Park***

**Graduate School of Korea Maritime University, Busan 606-791, Korea*

***Division of Information Technology, Korea Maritime University, Busan 606-791, Korea*

요 약 : 웹은 사용자가 원하는 정보를 쉽고 정확하게 검색할 수 있도록 웹 문서의 내용과 구조를 지속적으로 개선하고 사용자의 특성과 행동 패턴에 따라 개인화 하여야한다. 또한 웹으로부터 유익한 정보를 찾아내기 위한 웹 마이닝과 같은 문제가 대두되고 있다. 이러한 문제를 해결하기 위해서는 웹 문서들 간의 정확한 구조를 추출하는 것이 선행되어야 한다. 본 논문에서는 이러한 웹 문서 구조 추출을 위한 개선된 방법을 제안한다. 제안 방법은 기본적으로 웹 문서 태그의 하이퍼링크를 깊이 우선 탐색 알고리즘을 사용하여 방향그래프로 만든다. 하지만 이러한 웹 문서 태그 탐색 시 플래시나 스크립트 등에 숨어 있는 하이퍼링크를 찾는 문제와 '뒤로' 버튼 사용 시 웹 접근로그에 기록되지 않는 문제점이 보완되어야 한다. 이를 위해 클릭 스트림을 스택에 저장하여 이미 만들어진 방향그래프와 비교하여 새롭게 찾은 정점과 간선을 추가함으로써 보다 신뢰성 높은 방향그래프를 만든다.

핵심용어 : 웹 문서 구조, 웹 접근로그, 웹 마이닝, 방향그래프

ABSTRACT : *Web documents and structures should be continuously improved for users to access easily and exactly what they want, and need to be personalized by reflecting individuals' behavior pattern and characteristic. Furthermore, web-mining problems are emerging to extract useful information from the Web. To solve these problems, the exact structure of web documents must be extracted first. This paper proposes an improved method to extract the structure of web documents. This method basically constructs a directed graph by using the depth first search algorithm on hyperlinks of web document tags. However, two problems must be complemented when searching web document tags; Firstly, how to find hyperlinks implicitly contained in the documents such as Flashs and Scripts. Secondly, the not-recording problem on web access logs when using the 'Back' button of web browsers. To cope with the problems, new vertices and edges are gradually added to the previous directed graph by saving click streams on a stack.*

KEY WORDS : web document structure, web access log, web mining, directed graph

1. 서 론

웹은 사용자가 원하는 정보를 쉽고 정확하게 검색할 수 있도록 웹 문서의 내용과 구조를 지속적으로 개선하고 사용자의 특성과 행동 패턴에 따라 개인화 하여야한다. 또한 웹으로부터 유익한 정보를 찾아내기 위한 웹 마이닝과 같은 문제가 대두되고 있다. 이러한 문제를 해결하기 위해서는 웹 문서 분석뿐만 아니라 웹 접근로그(web access log) 분석도 병행하여 수많은 링크들의 구조를 효율적으로 추출하고 체계적으로 자료구조화하여야 한다.

본 논문에서는 웹 문서를 구조화하기 위한 방법으로 크게 두 가지를 제안한다. 첫째, 웹 문서의 태그 분석을 통해 하이

* arno78@bada.hhu.ac.kr 051)410-4895

** hcpark@hhu.ac.krr 051)410-4573

퍼링크를 탐색하고 이를 깊이 우선 탐색(depth first search) 알고리즘을 적용하여 정점과 간선을 추출하고 방향그래프 형태로 구조화한다. 둘째, 이렇게 구해진 방향그래프에는 플래시, 스크립트 등과 같은 웹 문서 태그에서 탐색되지 않은 하이퍼링크들이 존재한다. 이는 웹 접근로그 분석을 통해 추출할 수 있다. 먼저 웹 접근로그를 정제과정을 거쳐 사용자별 클릭 스트림(click stream)을 추출한다. 또한 '뒤로' 버튼 사용 시 접근한 페이지는 브라우저의 캐시에만 저장되고 웹 접근로그에는 기록되지 않는다. 이를 해결하기 위한 방법으로 클릭 스트림을 스택에 저장하여 이미 만들어진 방향그래프의 정점과 간선을 비교하여 새로운 정점이나 간선이 발생할 경우 패턴후보들을 생성하여 트랜잭션의 크기가 가장 작은 정점과 노드를 방향그래프에 추가한다. 이렇게 구해진 방향그래프는 높은 신뢰성을 갖는 방향그래프가 된다.

2. 관련연구

2.1 웹 마이닝

웹 마이닝은 웹 문서와 서비스들로부터 알려지지 않은 유용한 정보를 자동으로 검색하고 추출하기 위한 과정으로 3가지 영역으로 분류될 수 있다[1]. 첫째, 웹 내용 마이닝(Web Content Mining)은 온라인상에서 이용 가능한 정보를 자동으로 찾아주는 기법이다. 둘째, 웹 구조 마이닝(Web Structure Mining)은 웹 환경에서 참조한 페이지와 참조된 페이지 사이의 관계구조에 대한 정보 및 웹 사이트나 웹 페이지에 대한 요약된 구조를 생성시키는 기법이다. 마지막으로, 웹 사용 마이닝(Web Usage Mining)은 접속 경향과 패턴을 이해하기 위하여 웹 접근로그에 기록된 내용 중에서 웹 사이트의 하이퍼링크 경로를 통해 패턴을 분석하여 정확한 항해경로를 찾아내는 것이다[2].

2.2 웹 접근로그 분석

웹 접근로그를 분석하기 위한 단계로 데이터 정제(Data Cleaning), 사용자 구분(User Identification), 세션 구분(Session Identification), 세션 보정(Session Completion) 등이 필요하다. 데이터 정제 과정은 방문시간, 사용자, IP주소, 요청시간, HTTP 방식, 요청된 파일, HTTP 버전, 상태코드, 전송된 바이트 수 등이 기록되어 있는 웹 접근로그에서 필요한 항목을 추출하여, 페이지 뷰당 중복적인 내용을 제거하고 불필요한 노이즈들을 제거하는 과정이다[3].

사용자 세션구분은 한 사용자가 웹 사이트에 접속하여 웹 탐색을 수행한 후 접속을 종료할 때까지의 일련의 행위이다[4]. 사용자 구분에 사용되는 방법은 IP주소와 에이전트(Agent), 쿠키(Cookie)를 사용하는 방법, 캐시 버스팅을 이용하는 방법, 페이지에 에이전트나 애플릿을 삽입하는 방법 등 여러 가지 방법들이 쓰이고 있다. 본 논문에서는 IP주소의 에이전트 구분, 타임아웃 시간 등으로 사용자 세션을 구분한다.

2.3 순회패턴 탐사 및 방향그래프

정제된 데이터로부터 사용자의 웹 접근 패턴을 분석하는 것을 '순회 패턴 탐사'라 한다[5]. 이는 사용자가 원하는 정보를 탐색하기 위해 이동하는 경로를 말한다. 사용자의 행위의 특성을 파악하여 그 서비스의 질을 개선하고 사용자 요구를 극대화시킨다.

순회 패턴 탐사를 위해서 웹 문서의 태그를 분석하여 하이퍼링크들을 추출하고 웹 페이지 구조를 방향그래프로 표현한다[6]. 플래시, 스크립트 등에 대해서는 하이퍼링크 경로를 추출할 수 없기 때문에 완전한 그래프를 생성하지 못한다. 본 논문에서는 웹 접근로그 분석을 통해 정점과 간선을 추출하여 방향그래프에 추가한다.

3. 웹 문서 구조 추출

웹 문서를 구조화하여 관리하기 위해서는 적합한 자료구조 형태로 표현하여야 한다. 본 논문에서는 깊이우선 탐색 알고리즘을 적용하여 웹 문서의 하이퍼링크를 방향그래프로 표현한다. 이렇게 얻어진 방향그래프는 웹 문서의 플래시, 스크립트 등 때문에 하이퍼링크를 완전히 표현하지는 못한다. 이를 보완하기 위하여 웹 접근로그를 추가적으로 사용하여, 각 사용자의 클릭 스트림 분석을 통하여 새로운 정점(Vertex)과 간선(Edge)들을 추가하여 보완된 방향그래프를 만든다. Fig. 1은 본 논문에서 제안하는 시스템의 구성도이다

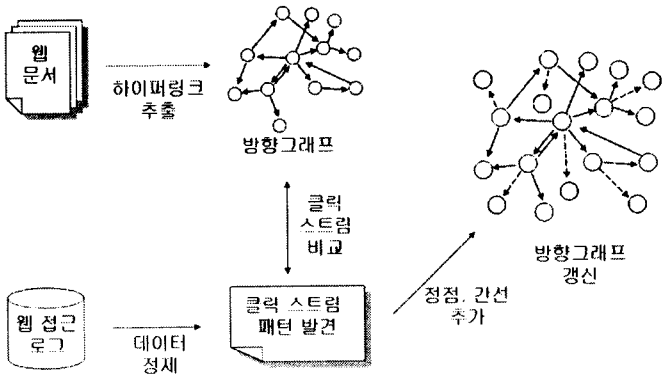

Fig. 1 Overview of System

3.1 웹 문서의 하이퍼링크 추출

웹 페이지를 구조화하기 위해서는 웹 문서의 태그를 분석하여 하이퍼링크를 추출하고 적합한 자료구조 형태로 바꾸어 주어야 한다. 이를 위해 방향그래프가 사용될 수 있다. 웹 페이지가 주어졌을 때 출발하는 첫 페이지로부터 연결된 모든 페이지 집합을 깊이우선탐색 알고리즘을 적용하여 모든 경로를 끝까지 탐색함으로서 웹 문서에 대한 하이퍼링크의 모든 경로를 찾을 수 있다. 이렇게 찾은 경로를 방향그래프로 표현하게 된다. 전체적인 탐색 과정을 살펴보면 Fig. 2와 같다.

① 최초의 정점(웹 페이지)에서 출발한다.
② 방문된 정점은 '방문기록배열'(visited record array)에 표시한다.
③ 선택된 정점에 연결된 여러 정점들을 검사한다.
④ 검사한 정점들 중에 아직 방문하지 않은 정점이 있으면 그 정점을 새롭게 선택하고 원래 정점을 스택에 넣는다.
⑤ 검사한 정점들 중에 방문하지 않은 정점이 하나도 없으면 최종적으로 삽입된 정점을 꺼내어 새롭게 선택한다.
⑥ Stack이 빌 때까지 ②-⑤의 과정을 반복한다.

Fig. 2 Search Procedure

Fig. 3의 알고리즘은 최초의 정점에서 출발하여 새로운 정점을 순회하면서 순환한다. 이 알고리즘은 깊이우선탐색을 하기 때문에 탐색 시 정점 'v'가 방문 될 때마다, 그 정점과 인접하면서 방문되지 않은 모든 정점들을 순환적으로 찾을 수 있다.

이런 과정을 통하여 정점과 간선을 추출하면 Table. 1과 같은 결과를 얻을 수 있고, 결과를 방향그래프로 표현하면 Fig. 4와 같이 나타낼 수 있다.

```
Visited_Record_DFS(int v)
    visited[ ] ← 0    // 방문여부저장 배열
    stack[ ] ← 0      // 방문경로저장 배열
    visited[v] ← 1;  // 정점v 방문
    for(each vertex w adjacent from v) {
        if(visit[w] is 0) {
            push(w);
            Visited_Record_DFS(w);
        }
        else {
            Cycle_Detection()
            Output_Cycle_Path()
        }
    }
    temp ← pop( );
    visited[temp] ← 1;
```

Fig. 3 Search Algorithm

Table 1 Hyperlink Extracted from Web Documents

부모노드	-	자식노드
A.html	-	B.html
A.html	-	C.html
A.html	-	D.html
A.html	-	E.html
B.html	-	F.html
B.html	-	G.html
B.html	-	H.html
C.html	-	I.html

부모노드	-	자식노드
C.html	-	J.html
D.html	-	A.html
D.html	-	J.html
D.html	-	K.html
E.html	-	L.html
F.html	-	A.html
I.html	-	B.html
L.html	-	A.html

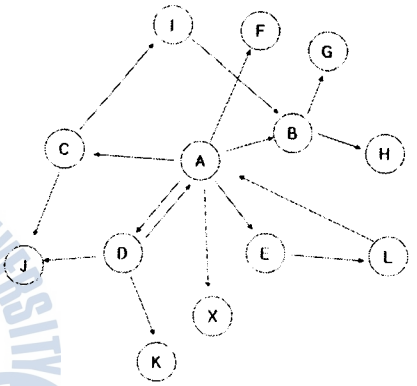

Fig. 4 Directed Graph

웹 문서에서 미처 발견하지 못한 정점, 간선이 포함 될 수 있기 때문에 Fig. 4는 신뢰할 수 있는 그래프가 아닐 수 있다. 사실 플래시를 사용한 경우는 웹 문서를 통해 하이퍼링크를 알 수가 없다. 본 논문에서는 웹 접근로그에서 탐색한 클릭 스트림을 이용해 새로운 정점과 간선들을 추가하여 보다 신뢰성 있는 방향그래프로 보완한다.

3.2 웹 접근로그 분석을 통한 정점과 간선 추가

본 논문에서는 웹 접근로그 데이터를 이용하여 데이터 정제를 하고, 사용자 구분과 세션 구분을 하는 전처리 과정을 거친다. 독립적인 세션을 구하기 위한 방법으로 IP주소, 같은 IP주소의 에이전트 구분, 타임아웃 시간 등을 고려하여 각각의 세션에 대한 페이지 접근을 구분하고 클릭 스트림을 추출하여 새로운 정점과 간선을 추가한다. 이때 도메인 주소가 다른 경우는 다른 사이트의 로그로 판단하여 탐색 과정에서 배제한다.

웹을 항해하는 사용자들은 '뒤로' 버튼을 자주 사용한다. 이때 이미 접근한 페이지는 브라우저 캐시에 저장되어지는데, 다시 요청이 일어나면 캐시에 있는 내용이 사용자에게 보내져 서버의 접근로그에는 기록되지 않는다는 문제점을 가지게 된다. 이를 해결하기 위한 방법으로 웹 접근로그의 전처리과정에

서 추출한 클릭스트림을 스택에 저장하여 이미 찾은 방향그래프와 비교하고 '뒤로' 버튼을 알아낸 뒤 새로운 정점과 간선이 나타나면 후보노드를 생성하여 방향그래프를 갱신한다.

예를 들어, 하나의 세션에 (A, B, H, D, J, K, X) 클릭 스트림이 있다면 다음의 순서로 방향그래프를 보완한다.

① 클릭스트림을 아래와 같이 스택에 저장한다.

A	B	H	D	J	K	X

② 스택의 내용을 이미 찾은 방향그래프와 비교하여 아래(③)의 방법으로 간선이 있는지 확인한다.

③

A	B	H	D	J	K	X

④ 존재하지 않으면 스택에서 하나씩 꺼내어 다음 노드와 비교 한다. (웹 문서에서 얻은 하이퍼링크와 일치하는 간선이 존재한다면 '뒤로' 버튼 사용이고, 존재하지 않으면 새로운 노드이다.)

A	B	H	D	J	K	X

A	B	D	J	K	X	

A	D	J	K	X		

⑤ ②~④ 과정을 반복한다.

A	D	K	X			

⑥ 새로운 노드가 발견되었을 때 ⑤과정에서 얻은 간선들과 새로운 노드를 저장한다.

패턴 후보	새로운 노드
A, D K	X
A, D	X
A	X

⑦ 트랜잭션 크기가 가장 작은 패턴후보 중 단말노드와 단말노드 이전의 노드에 새로운 노드를 추가한다. (단, 트랜잭션 크기가 1일 경우 해당 노드에만 추가한다. 위 표에서는 A 단말노드에만 노드 추가)

위의 과정 중에서 스택 저장 과정을 그림으로 나타내면 Fig. 5와 같다

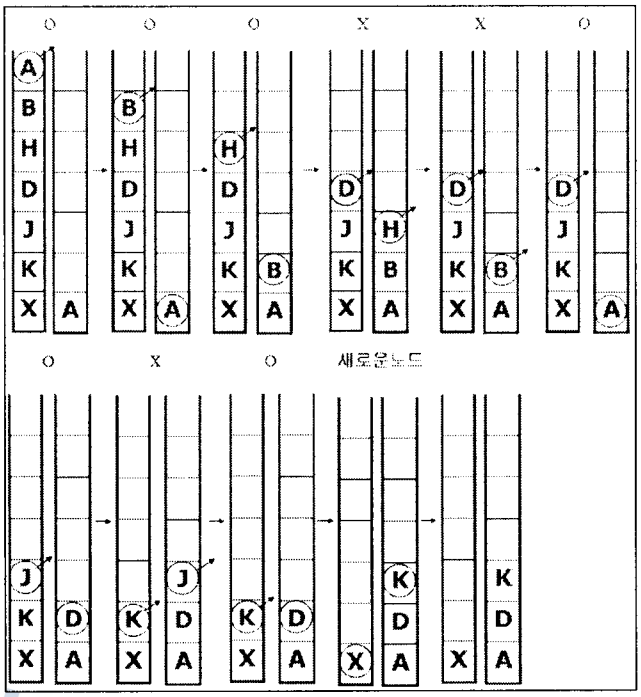

Fig. 5 Outline of Stack

새로운 노드가 나타났을 때의 추가 방법에 대해 알아보았고 존재하는 노드에 새로운 간선이 나타났을 경우에도 위와 같은 방법으로 간선을 추가한다. 결과적으로 웹 접근로그에서 얻어진 탐색경로를 웹문서에서 추출한 탐색경로와 비교/추가하여 보다 정확한 방향그래프를 완성한다.

3.3 실험 및 분석

본 논문의 실험 결과로 웹 페이지의 태그를 분석하여 하이퍼링크 탐색으로 만들어진 방향그래프에, 웹 접근로그 데이터로부터 추출된 클릭 스트림을 통해 새로운 정점과 간선들이 추가되었고, 추가된 정점과 간선들은 플래시나 스크립트 등을 사용한 정점과 간선들이 추출될 수 있었다.

새롭게 추가된 하이퍼링크나 노드들은 '뒤로' 버튼 사용으로 정점과 간선이 추가될 때 트랜잭션의 크기가 2 이상일 경우 단말 노드와 단말 노드 이전 노드에 새로운 노드를 추가하기 때문에 추가비용이 발생한다.

4. 결론 및 향후 과제

본 논문에서는 웹 문서의 하이퍼링크 분석뿐만 아니라 웹 접근로그의 분석을 통하여 보다 신뢰성 있는 웹 구조를 추출하는 방법을 제안하였다. 기본적으로 웹 페이지들의 태그 분석을 통하여 하이퍼링크를 추출하여 방향그래프를 만들었다. 추

가적으로, 웹 접근로그 분석을 통하여 하이퍼링크, 플래시, 스크립트 등을 찾아내어 방향그래프를 갱신하여 보다 신뢰성 있는 방향그래프를 만들어 내었다. 이렇게 만들어진 방향그래프는 다양한 웹 구조 개선 및 웹 마이닝을 위한 핵심적인 자료로 활용될 수 있다.

향후 연구 과제로는 웹 페이지 태그 추출로 만들어진 방향그래프를 보완하기 위해 추가적으로 발생하는 웹 접근로그 분석비용을 최소화하는 방법 등이 있다.

참 고 문 헌

[1] J. Huysmans, B. Baesens and J.Vanthienen, "Web Usage Mining: A Practical Study", KAM, pp.86-99, 2004.

[2] Thuraisingham, "Web Data Mining and Business Intelligence Analysis", CRC Press, 2003.

[3] M. Baglioni, U. Ferrara, A. Romei, S. Ruggieri and F. Turini, "Preprocessing and Mining Web Log Data for Web Personalization", Proceedings 8th Italian Conf. on Artificial Intelligence, LNCS 2829, pp.237-249, 2003.

[4] M. Koutri, N. Avouris and S. Daskalaki, "A Survey on Web Usage Mining Techniques for Web-Based Adaptive Hypermedia System", IRMA, pp.125-149, 2005.

[5] M. S. Chen, "Efficient Data Mining for Path Traversal Pattern", IEEE KDE, vol.10, no.2, pp.209-221, 1996.

[6] G. Nivasch, "Cycle Detection Using a Stack", Information Processing Letters, pp.135-140, 2004.

[7] Chakrabarti, Mining the Web, Morgan Kaufmann Pub, 2002.

[8] D. Embley, C. Tao and S. Liddle, "Automating the Extraction of Data from HTML Table with Unknown Structure", KDE, pp.3-28, 2005.

[9] 이성대, 박휴찬, "가중치 그래프에 기반한 순회 패턴 탐사", 한국해양정보통신학회 학술대회 논문집, vol.8 pp.433-437, 2004.

[10] 박상언 이우기, 차창일, "웹 그래프에서 순환 경로 나열 알고리즘", 한국경영정보학회, 단일호, pp.754-762, 2005.

원고접수일 : 2006년 1월 3일
원고채택일 : 2006년 1월 7일