제26회 한국정보처리학회 추계학술발표대회 논문집 제13권 제2호 (2006. 11)

웹 접근로그를 활용한 웹 구조 마이닝

박철현* · 이성대* · 전성환* · 박휴찬**
*한국해양대학교 대학원
**한국해양대학교 IT 공학부 교수
e-mail:arno78@bada.hhu.ac.kr

Web Structure Mining Using Web Access Log

C. H. Park* · S. D. Lee* · 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

웹의 급속한 성장으로 정보의 양이 많아졌지만 디자인의 비중이 커지면서 웹 문서에 대한 구조를 추출하는데 어려움이 있다. 웹은 사용자가 원하는 정보를 쉽고 정확하게 검색할 수 있도록 웹 문서의 내용을 구조화하여 지속적으로 개선하면서 사용자의 특성과 행동 패턴에 따라 개인화 하여야한다. 이러한 문제를 해결하기 위해서는 웹 문서들 간의 정확한 구조를 추출하는 것이 선행되어야 한다. 본 논문에서는 보다 웹 사이트의 정확한 구조를 추출하기 위한 방법을 제안한다. 제안 방법은 기본적으로 웹 문서 태그의 하이퍼링크와 플래시 파일을 2진 형태의 문서로 불러 하이퍼링크를 추출하고 이를 깊이 우선 탐색 알고리즘을 사용하여 방향그래프로 만든다. 하지만 이러한 웹 문서 태그 탐색 시 애플릿이나 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 추가 삭제함으로써 보다 신뢰성 높은 방향그래프를 만든다.

1. 서론

웹의 급속한 성장으로 정보의 양이 늘어나면서 웹 개발자들은 개발 및 유지보수가 어려워지고 사용자는 방대해진 정보를 쉽고 정확하게 찾기 힘들어 졌다. 근래의 웹 문서는 디자인에 많은 비중을 두어 플래시(Flash)를 사용하여 화려하게 만들고 있다. 그러나 플래시가 포함된 웹 문서는 해당 플래시 파일의 하이퍼링크만 존재하고 플래시 내의 하이퍼링크는 보이지 않는다. 이를 해결하기 위한 방법으로 플래시 파일을 2진 형태의 문서로 불러 하이퍼링크를 추출하고 웹 문서내의 하이퍼링크와 함께 깊이 우선 탐색을 하고 자료 구조를 지속적으로 개선하고 사용자의 특성과 행동 패턴에 따라 개인화 하여 웹으로부터 유익한 정보를 찾아내어 이를 웹 마이닝화 하여야 한다. 이러한 문제를 해결하기 위해 웹 문서를 체계적으로 구조화 하여야 한다.

본 논문에서는 웹 문서를 구조화하기 위한 방법으로 크게 세 가지를 제안한다. 첫째, 웹 문서의 태그 구조 분석되어 하이퍼링크를 탐색하고 탐색한 링크의 우선 순서로 깊이 우선 탐색(depth first search) 알고리즘을 적용한다. 이때 플래시 파일 하이퍼링크가 탐색되면 플래시 파일을 2진 형태의 문서로 불러와 하이퍼링크 집합을 추출하고 진행 중인 탐색 알고리즘에 추가하여 정점과 간선을 추출한다. 그리고 방향그래프 형태로 구조화한다. 둘째, 이렇게 구해진 방향그래프에는 애플릿, 스크립트 등 탐색되지 않은 하이퍼링크들이 존재한다. 이는 웹 접근로그 분석을 통해 추출할 수 있다. 먼저 웹 접근로그 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 웹 접근로그의 클릭 스트림을 스택에 저장하여 이미 만들어진 방향그래프의 정점과 간선을 비교한다. 새로운 정점이나 간선이 발생할 경우 패턴후보들을 생성하고 트랜잭션의 크기가 가장 작은 정점과 노드를 방향그래프에 추가한다. 셋째, 웹 접근로그 정재 과정에서 서비스 상태 코드가 에러인 간선에 대해 정점과 간선을 찾아내고 방향그래프에서 삭제하여 보다 높은 신뢰성을 갖는 방향그래프를 완성한다.

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]. 애플릿, 스크립트 등에 대해서는 하이퍼링크 경로를 추출할 수 없기 때문에 완전한 그래프를 생성하지 못한다. 본 논문에서는 웹 접근로그 분석을 통해 정점과 간선을 추출하여 방향그래프에 추가 삭제한다.

3. 웹 문서 구조화

웹 문서내의 하이퍼링크는 각 문서를 연결하는 자료의 흐름이다. 웹 문서를 구조화하여 자료의 흐름을 알아냄으로서 사용자의 행동 패턴을 파악하고 웹 사이트의 구조, 설계상의 문제점 등을 발견, 보완해서 적합한 자료구조 형태로 표현하여야 한다. 본 논문에서는 웹 문서를 적합한 자료구조 형태로 표현하기 위해 웹 문서의 구조를 하이퍼링크 깊이 우선 탐색 알고리즘을 적용하여 사이트의 모든 하이퍼링크를 추출한다. 이때 플래시 파일을 연결하는 하이퍼링크가 나타나면 해당 플래시 파일을 2진형태의 문서로 불러와 문서내의 하이퍼링크를 찾아내어 이를 함께 깊이 우선 탐색 알고리즘에 포함시켜 방향그래프로 표현한다. 이렇게 얻어진 방향그래프는 웹 문서 내의 애플릿, 스크립트 등에 완전히 표현하지는 못한다. 이를 보완하기 위하여 웹 접근 로그를 추가하여 각 사용자의 클릭 스트림 분석을 통한 새로운 정점(Vertex)과 간선(Edge)들을 추가하여 방향그래프를 만든다. 그리고 웹 접근 로그파일의 서비스 상태 코드를 참조하고 연결이 끊어져있는 정점에 대해서는 정점과 간선을 삭제하여 보다 보완된 방향그래프를 만든다.

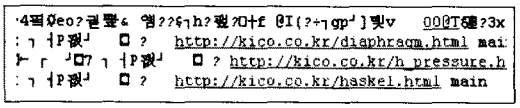
3.1 웹 문서의 하이퍼링크 추출

웹 사이트를 구조화하기 위해서는 웹 문서의 태그를 구조 분석하여 해당 웹서버 내의 하이퍼링크를 추출하고 적합한 자료구조 형태인 방향그래프가 사용된다. 그림 1은 방문기록 순환탐색 알고리즘은 웹 페이지가 주어졌을 때 최초의 정점에서 출발하는 첫 페이지로부터 연결된 모든 페이지 집합을 깊이 우선 탐색 알고리즘을 적용하여 모든 경로를 끝까지 탐색한다. 탐색 시 플래시 파일이 존재하면 해당 파일을 그림 2와 같이 2진 형태의 문서를 읽어 하이퍼링크들을 추출하여 진행 중인 탐색 알고리즘과 함께 순회하여 방문했던 정점들을 '방문기록배열' (visited record array)에 기록한다. 순환 탐색 시 정점을 방문하는 순서대로 스택에 넣고 방문기록 하여 이미 방문한 정점을 방문할 경우 방문기록에 있으므로 순환이 탐색되었음을 알 수 있고, 스택을 이용해서 순환경로를 출력할 수 있다. 방문기록이 되지 않은 정점을 탐색했을 경우에는 그 정점을 스택에 넣고 반복한다. 재귀호출을 이용하며 정점을 스택에 넣고 재귀순환이 끝나면 호출했던 함수로 돌아와서 다시 스택에서 제거한다. 스택이 모두 비워지면 순환 탐색은 종료된다. 이 알고리즘은 깊이 우선 탐색을 하기 때문에 탐색 시 정점 'v'가 방문 될 때마다, 그 정점과 인접하면서 방문되지 않은 모든 정점들을 순환적으로 찾을 수 있다.

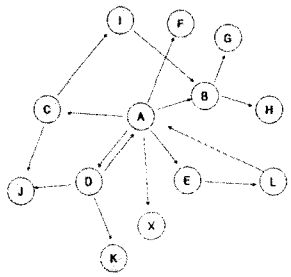
이런 과정을 통하여 정점과 간선을 추출하면 그림 3와 같이 방향그래프로 나타낼 수 있다.

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

(그림. 1) 방문기록 순환탐색 알고리즘



(그림. 2) 플래시 파일의 2진 형태 일부분



(그림. 3) 방향그래프

웹 문서에서 미처 발견하지 못한 정점, 간선이 포함 될 수 있기 때문에 그림 3은 신뢰할 수 있는 그래프가 아닐 수 있다. 사실 애플릿이나 스크립트를 사용한 경우는 웹 문서를 통해 하이퍼링크를 알 수가 없다. 본 논문에서는 웹 접근로그에서 탐색한 클릭 스트림을 이용해 새로운 정점과 간선들을 추가, 삭제하여 보다 신뢰성 있는 방향그래프로 보완한다.

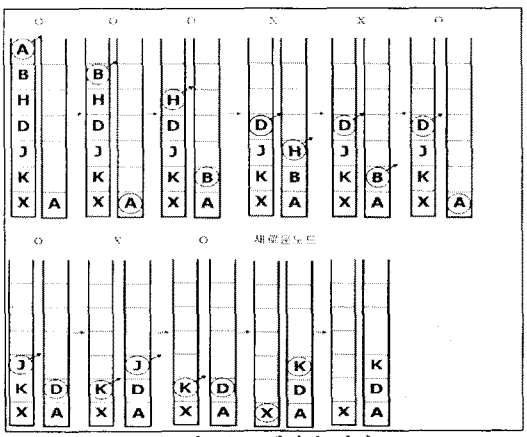
3.2 웹 접근로그 전처리과정

본 논문에서는 웹 접근로그 데이터를 이용하여 새로운 정점과 간선들을 추가, 삭제하여야 한다. 그러기 위해서는 웹 접근로그 데이터를 통해 사용자 구분과 세션 구분을 하는 전처리 과정을 거친다. 로그를 분석하는 항목으로는 독립적인 세션을 구하기 위한 방법으로 IP주소, 같은 IP주소의 에이전트 구분, 타임아웃 시간, 서비스 상태 코드 등을 고려하여 각각의 세션에 대한 페이지 접근을 구분하고 클릭 스트림을 추출한다. 그리고 서비스 상태 코드가 200번이 아닌 경우 웹 문서를 보는 동안 에러가 발생한 경우이므로 에러에 대한 로그정보를 저장한다. 그리고 도메인 주소가 다른 경우는 다른 사이트의 로그로 판단하여 탐색 과정에서 배제한다.

3.3 웹 접근로그를 통한 정점과 간선 추가

실제적으로 존재하지만 웹문서의 하이퍼링크에서 추출하지 못한 링크에 대해 웹 접근로그의 전처리과정에서 추출한 사용자별 클릭스트림을 이용하여 새로운 정점과 간선을 추가한다. 이때 '뒤로' 버튼에 따른 문제가 발생하게 된다. 이는 항해하는 사용자들은 '뒤로' 버튼을 자주 사용하는데 이미 접근한 페이지는 브라우저 캐시에 저장되어지고, 다시 요청이 일어나면 캐시에 있는 내용이 사용자에게 보내져 서버의 접근로그에는 기록되지 않는다는 문제점을 가지게 된다. 이를 해결하기 위한 방법으로 본 논문에서는 웹 접근로그의 전처리과정에서 추출한 클릭스트림을 스택에 저장하여 이미 찾은 방향그래프와 비교하고 '뒤로' 버튼을 알아낸 뒤 새로운 정점과 간선이 나타날 때 후보노드를 생성하여 방향그래프를 갱신한다.

예를 들어, 하나의 세션에 (A, B, H, D, J, K, X) 클릭스트림이 있다면 클릭스트림을 스택에 저장하고 정점을 쌓을 지어 이미 찾은 방향그래프와 비교하여 간선이 있는지 확인한다. 간선이 존재하면 패턴 후보 스택에 삽입하고, 간선이 존재하지 않을 때까지 비교한다. 존재 하지 않은 간선이 나타나면 패턴 후보 스택들과 다시 비교하고 존재한다면 '뒤로' 버튼 사용이고, 존재 하지 않으면 새로운 노드이다. 이 과정을 그림으로 나타내면 그림 4와 같다.



(그림. 4) 스택저장 과정

위의 예제에서 새로운 노드가 발견되었을 때 얻은 패턴 후보는 A, D, K가 되고 새로운 노드는 X가 된다.

세션구분	패턴후보	새로운 노드
세션1	A, D, K	X
세션2	A, D	X
세션3	A	X

<표. 1> 세션에 따른 새로운 노드 발견

세션별 클릭스트림을 통해 새로 발견된 노드에 대해서 패턴후보의 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)

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

3.4 웹 접근로그 분석을 통한 정점과 간선 삭제

웹 문서의 하이퍼링크에는 사용자의 접근이 가능한 링크로 존재하지만 웹 접근 로그의 서비스 상태 코드가 200번이 아닌 에러코드인 경우 정점 또는 간선이 존재 하지 않는 것으로 간주하고 해당 정점과 그에 속한 간선들을 방향그래프에서 삭제하여 보다 정확한 방향그래프를 완성한다.

3.5 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 애플릿이나 스크립트 등을 사용한 정점과 간선들이 추출되었다.

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. 먼저 웹 페이지들의 태그 분석을 통하여 하이퍼링크를 추출하여 방향그래프를 만들었다. 추가적으로, 웹 접근로그 분석을 통하여 하이퍼링크, 애플릿, 스크립트 등을 찾아내어 방향그래프를 갱신하여 보다 신뢰성 있는 방향그래프를 만들어 내었다. 이렇게 만들어진 방향그래프는 다양한 웹 구조개선 및 웹 마이닝을 위한 핵심적인 자료로 활용된다.

향후 연구 과제로는 플래시로 구조화된 내비게이션 웹은 구조를 전혀 알 수 없으므로 플래시에서 추출된 하이퍼링크를 사용자의 항해 패턴을 분석하여 하이퍼링크를 구조화 하여야 한다. 그리고 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, 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, F. Turini, "Preprocessing and Mining Web Log Data for Web Personalization", Proceedings 8th Italian Conf. on Artificial Intelligence, pp.237-249 Vol.2829 of LNCS, 2003.

[4] M. Koutri, N. Avouris, S. Daskalaki, "A Survey on Web Usage Mining Techniques for Web-Based Adaptive Hypermedia System", IRMA, pp125-149, 2005

[5] M. S. Chen, "Efficient Data Mining for Path Traversal Pattern", IEEE KDE Vol.10 Num2 pp209-221, 1996

[6] G. Nivasch, "Cycle Detection Using a Stack", Information Processing Letters, pp135-140, 2004

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

[8] D. Embley, C. Tao, S. Liddle, "Automating the Extraction of Data from HTML Table with Unknown Structure", KDE, pp3-28, 2005

[9] 이성대, 박휴찬, "가중치 그래프에 기반한 순회 패턴 탐사", 한국해양정보통신학회 학술대회 논문집, vol.8 pp433-437, 2004

[10] 박상언 이우기, 차창일 "웹 그래프에서 순환 경로 나열 알고리즘", 한국경영정보학회, 단일호, pp754-762, 2005