2001년 한국정보처리학회 춘계 학술발표논문집 제8권 제1호

# 적응형 웹 사이트를 위한 웹 로그 마이닝

고경자, 김인철
경기대학교 전자계산학과
e-mail : rain825@kuic.kyonggi.ac.kr

# Web Log Mining for Adaptive Web Sites

Kyong-Ja Ko, In-Cheol Kim
Dept of Computer Science, Kyonggi University

## 요 약

본 논문에서는 웹 사이트에 접근하는 이용자의 패턴을 분석하여 정보 제공이 보다 용이한 구조로 자동 개선시켜 나가는 적응형 웹 사이트의 구현 방안을 제시한다. 특히, 본 연구에서는 기존 웹 사이트의 구조를 가능한 파괴하지 않는 범위 내에서 웹 사이트를 변경하고자 이용자의 접근 패턴상 연관성은 높으나 접근 경로가 긴 문서들을 추출하여 색인 페이지를 추가 생성한다. 이를 위하여, 먼저 대용량의 웹 서버 로그 데이터를 대상으로 하이퍼 링크 구조에 따라 필터링된 최후 전진 문서만을 가지고 데이터 시퀀스를 구성한다. 이러한 데이터 시퀀스에 새로운 순차 접근 패턴 탐색 알고리즘인 TPA를 적용함으로써 웹 문서간 충분한 지지도를 갖는 연관성 있는 문서들의 시퀀스를 구한다. 이와 같은 빈발 시퀀스들에 대한 색인 페이지를 추가로 생성시켜주는 서비스를 통하여 이용자들의 효과적인 정보 접근을 지원할 수 있는 웹 사이트로의 변경이 가능하다.

## 1. 서 론

초기의 웹 사이트는 각 웹 문서들이 지닌 의미와 문서들 간의 상호관계 등을 고려해 최상의 사이트를 구현하고자 하는 웹 마스터의 의도가 반영된 것이다. 그러나, 동적으로 변화하는 이용자들의 요구를 반영하여 보유 정보를 효과적으로 제공하기 위해서 지속적인 웹 사이트의 변경과 갱신 작업이 요구된다. 이를 위하여, 이용자들의 일반적인 정보 접근 패턴을 알아내는데 중요한 자료가 되는 웹 서버 로그 데이터를 마이닝함으로써 웹 사이트의 구조나 표현 방식을 개선시킬 수 있다.

본 논문에서는 색인 페이지의 추가 생성을 통한 적응형 웹 사이트의 구현을 제안한다. 이를 위하여, 웹 서버 로그 데이터를 대상으로 하이퍼 링크 구조 정보를 적용하여 최후 전진 문서(last forward document)만을 갖는 데이터 시퀀스(data sequence)를 구성한다. 데이터 시퀀스를 대상으로 새로운 순차 접근 패턴 알고리즘인 TPA(Traversal Pattern Analysis)를 적용함으로써 연관성은 높으나 접근 경로가 긴 문서들의 시퀀스를 생성한다. 이러한 빈발 시퀀스들을 구성요소로 한 색인 페이지(index page)들은 기존 웹 사이트의 구조를 가능한 파괴하지 않도록 추가 생성이라는 방식으로 구현된다. 이와 같은 색인 페이지의 자동 생성은 다수의 이용자들에게 원하는 웹 문서를 빠르게 접근 할 수 있는 서비스를 제공한다.

본 논문의 2절에서 관련 연구들을 살펴보고, 3절에서 데이터 시퀀스 생성 모델과 빈발 시퀀스 탐색 과정을 기술한다. 4절에서 색인 페이지의 생성 구조를 제시하고 5절에서 결론을 맺는다.

## 2. 관련 연구

Etzioni의 연구[1]에서는 ' 적응형 웹 사이트란 이용자의 접근 패턴을 학습하여 웹 사이트의 구조나 외형을 자동적으로 개선시켜 나가는 웹 사이트' 라고 정의하고 있다. WebWatcher와 AVANTI 프로젝트에서는 이용자 개인의 접근 경로를 추적하여 학습한 후, 개인의 요구와 기호를 고려해서 성공적인 항해(navigation)가 가능하도록 웹 사이트를 개선시켜 나가는 맞춤화(customization) 방식을 취하고 있다. 반면, 이미 접근한 이용자들을 대상으로 접근 패턴을 학습하여 특정 개인이 아니라 접근할 가능성을 지닌 모든 이용자들을 위한 서비스 형태로서의 최적화(optimization) 방식을 따르는 시스템들도 다수 소개되고 있다. 이와 같은 적응형 웹 사이트의 구현을 위해서는 이용자들의 접근 패턴을 비롯한 다양한 웹 이용 정보를 토대로 이용자들의 요구가 반영되도록 웹 마이닝(web mining)과정을 거친다.

일반적으로, 웹 마이닝이라 함은 웹 사이트와 관련된 데이터를 분석함으로써 유용한 정보나 지식을 발견해 내는 기술을 말한다. 웹 마이닝은 분석 자료에 따라 웹 문서나 멀티미디어 자료와 같은 자원들을 분석하는 웹 컨텐트 마이닝(web content mining)과 웹 로그 데이터를 분석하는 웹 유시지 마이닝(web usage mining)으로 나눌 수 있다[2]. 전자를

대표하는 시스템으로는 STRUDEL, GroupLens 등이 있고, 후자로는 WebViz, WEBMINER등이 있다. 웹 유시지 마이닝에 적용 가능한 기술로는 탐사하고자 하는 지식의 형태에 따라 분류화(classification), 클러스터링(clustering), 연관 규칙(association rule)탐사, 순차 패턴 (sequential pattern)탐사 등이 있다. Perkowiz와 Etzioni의 연구[1]에서는 제안한 조건부 확률을 이용한 웹 문서 클러스터링 알고리즘을 제안하고 있다. 즉, 웹 문서 $P_1$과 $P_2$가 있다고 가정했을 때 $P_1$을 방문한 이용자가 $P_2$를 방문할 확률과 $P_2$를 방문한 이용자가 $P_1$을 방문할 확률 중 최소치를 선택하여 동시 발생(co-occurrence)빈도수로 정하고 이를 행렬과 그래프로 도식화해서 관련성 있는 웹 문서들의 클러스터를 생성하였다. 한편, 연관 규칙(association rule) 탐사 방법인 Apriori 알고리즘[3]을 응용함으로써 최소 지지도(support degree)를 만족하는 웹 문서들의 집합을 구할 수 있다. 또한, 순차 패턴 탐사 방법인 ApioriAll 알고리즘[6]을 응용함으로써 이용자들의 웹 운행 패턴(traversal pattern)을 알아낼 수 있다. Park의 연구[4]에서는 운행 패턴을 마이닝하기 위하여 이용자가 전진한 경우에 거쳐간 웹 문서 집합을 구하고 이를 대상으로 마이닝 알고리즘을 적용한 후 빈번히 등장한 웹 문서 시퀀스들을 구한다. 마지막 단계에서 빈번히 등장한 웹 문서 시퀀스들을 모두 포함할 수 있는 웹 문서 시퀀스들을 구하고 있다. Apriori을 개선한 알고리즘으로는 AprioriTid, AprioriHybrid, DHP 등이 있고, AprioriAll을 개선한 알고리즘에는 AprioriSome, AprioriAll, GSP, SPADE 등이 있다.

## 3. 순차 접근 패턴 분석

### 3.1 데이터 시퀀스 생성 모델

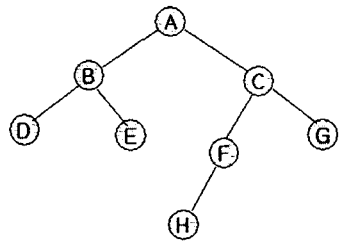
```
203.249.22.75 - - [25/Feb/2000:15:57:46 +0900] "GET / HTTP/1.1" 304 -
203.249.22.75 - - [25/Feb/2000:15:57:46 +0900] "GET /last1.jpg HTTP/1.1" 304 -
203.249.22.75 - - [25/Feb/2000:15:57:47 +0900] "GET /main.html HTTP/1.1" 304 -
203.249.22.75 - - [25/Feb/2000:15:57:47 +0900] "GET /menu.htm HTTP/1.1" 304 –
```

[그림 1] 웹 서버 로그 데이터

[그림 1]과 같이 웹 서버 로그 파일 안에는 이용자의 IP 주소, 이용자의 ID, 웹 문서에 접근한 날짜와 시간, 요청 방법, 접근한 문서의 URL, 데이터 전송에 사용된 프로토콜, 에러 코드, 전송 바이트 수 등에 대한 정보가 들어 있다. 이러한 서버 로그 파일을 대상으로 마이닝을 하기 위하여 데이터 시퀀스를 생성하는 과정은 다음과 같다.

① 로그 데이터를 대상으로 어구분석(parsing) 하는 과정과 이용자의 IP 주소, 이용자의 ID, 웹 문서에 접근한 날짜와 시간, 접근한 문서의 URL들을 제외한 불필요 항목들을 제거하는 정화(cleansing)과정을 거친다.

② 이용자가 접근한 웹 문서 각각을 트랜잭션으로 간주하고, 적용 가능한 접근 시간 내에 이용자별 한 세션(session)동안 거쳐간 웹 문서 운행 경로를 구하여 트랜잭션 시퀀스를 형성한다. 즉, [그림 3]의 왼쪽 표에서 보여주는 바와 같이 트랜잭션 시퀀스는 다수 이용자가 접근한 웹 문서들의 집합으로 구성된다.




[그림 2] 하이퍼 링크 구조

| ID | Traversal path |
|---|---|
| 1 | ABACF |
| 2 | ABACFHG |
| 3 | ABDBACFH |
| 4 | ABCF |
| 5 | ABACF |
| 6 | ABACFHFCG |
| 7 | ABCFHCG |
| 8 | ABCFABD |
| 9 | ACFBD |
| 10 | ACFCABED |

⇒

| ID | Sequence |
|---|---|
| 1 | BF |
| 2 | BHG |
| 3 | DH |
| 4 | BF |
| 5 | BF |
| 6 | BHG |
| 7 | BHG |
| 8 | BFD |
| 9 | FD |
| 10 | FED |

[그림 3] 데이터 시퀀스 생성

**Algorithm LFD:**

/* $s$ : source page
$d$ : destination page
$t$ : traversal path
$a$ : array of maximal sequences starting from the root node
$D_F$ : database to store all the resulting last forward document obtained. */

**For** ($i$=1; $i$ <=$k$; $i$++) **do** /* $k$: the length of traversal path*/
**Begin**
Express $t_k$ as $\{(s_1, d_1)\cdots(s_n, d_n)\}$;
**For** ($j$=1; $j$ <=$m$; $j$++) **do** /* $m$: the number of array*/
**Begin**
**If** Both $s_n$ and $d_n$ do not exist in $a_j$ **then**
Append $s_n$ to $D_F$;
**Else If** $d_n$ is not placed back of $s_n$
Append $s_n$ to $D_F$;
**End**
**End**

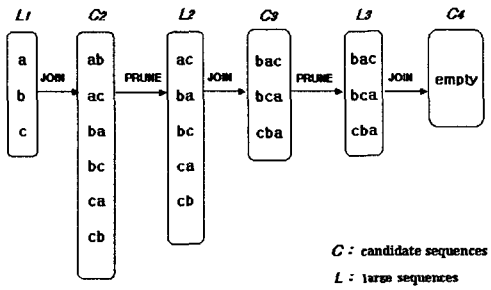
[표 1] LFD 알고리즘

③ 전 단계에서 구한 트랜잭션 시퀀스 안의 운행 웹 문서 각각을 대상으로 전문서에서 바로 이어지는 후문서가 [그림 2]과 같은 하이퍼 링크 구조상 동일 가지의 하위 레벨에 존재하지 않는 조건을 만족하는 전문서만을 추출하여 데이터 시퀀스가 될 최후 전진 문서(last forward

document) 집합을 구한다. 이러한 최후 전진 문서 집합이 갖는 의미는 하이퍼 링크 구조의 깊이(depth) 관점에서 보면, 이용자가 원하는 정보를 담은 문서일 가능성이 가장 높은데 반해서 하이퍼 링크 구조의 가지별로 구분해 보면, 두 문서간 접근 경로가 가장 멀어서 접근 경로를 단축시켜줄 필요성을 갖는다는데 있다. 즉, [그림 3]의 왼쪽 표의 집합들을 가지고 최후 전진 문서들만을 선택하면 오른쪽 표에 나타나는 형태의 데이터 시퀀스들이 생성되는데, 생성 과정은 [표 1] LFD 알고리즘에서 보여주고 있다.

### 3.2 빈발 시퀀스 탐색

본 논문에서 제안하는 순차 접근 패턴 탐색 알고리즘인 TPA는 색인 페이지의 생성 목적에 적합한 AprioiriAll 알고리즘의 변형이다. AprioiriAll 알고리즘과 구별되는 특징으로 TPA 알고리즘에서는




[그림 4] TPA 결합 알고리즘 형태

[그림 4]와 같이 동일 문서들의 조합을 기본적으로 고려하지 않으며 데이터베이스의 스캔 과정도 매 단계마다 수행되지 않는다. [표 2]은 TPA알고리즘의 적용 과정을 보여주고 있는데, 전체 처리 단계는 전진 부분 (forward phrase)과 역진 부분(backward phrase)으로 나누어 수행된다. 전진 단계를 살펴보면, 앞 과정에서 얻은 데이터 시퀀스들을 대상으로 웹 문서 각각으로 구성되는 후보 $i$-시퀀스의 집합 $C_i$을 생성하고 $C_i$에서 최소 지지도를 만족하는 빈발 $i$-시퀀스의 집합 $L_i$을 구한다. 두 번째 패스에서 후보 2-시퀀스의 집합 $C_2$를 생성하기 위하여 순서는 고려하지만 동일 문서 조합을 고려하지 않는다는 조건 아래 $L_1$* $L_1$을 구한 후, 데이터베이스를 스캔하여 $C_2$에서 최소 지지도를 만족하는 빈발 2-시퀀스의 집합 $L_2$를 구한다. 세 번째 패스에서, 후보 3-시퀀스의 집합 $C_3$를 생성하기 위하여 $L_2$를 기반으로 순서는 고려하지만 동일 문서 조합을 고려하지 않는다는 조건 아래 등장 가능성이 있는 $C_3$를 구한다. 물론, 이 과정에서 후보 시퀀스를 대상으로 $L_2$안에 존재하지 않는 서브시퀀스(subsequence)를 갖는 후보 시퀀스의 집합을 전정(pruning)하는 과정이 포함된다. 이후, 데이터베이스를 스캔하여 $C_3$에서 최소 지지 도를 만족하는 빈발 3-시퀀스의 집합 $L_3$를 구한다.

**Algorithm TPA:**

```
/ * Forward Phase */
L1={large 1-sequences};
C1= L1;   /*so that we have a nice loop condition*/
last=1;   /*we last counted Clast*/

For (k=2; Ck-1≠0 and Llast≠0; k++) do
  Begin
  If (Lk-1 known) then
    Ck=New candidates generated from Lk-1;
  Else
    Ck=New candidates generated from Ck-1;
  If (k==next (last)) then Begin
    Foreach data-sequence c in the database do
        Increment the count for all candidates
          in Ck that are contained in c
    Lk=Candidates in Ck with minimum support.
    last= k;
    End
  End

/ * Backward Phase */
For (k--; k>=1; k--) do
  If (Lk not found in forward phase) then Begin
    Delete all sequences in Ck contained in
        some Lp, i>k;
    Foreach data-sequence c in the database Dr do
        Increment the count for all candidates
          in Ck that are contained in c
    Lk=Candidates in Ck with minimum support.
    End
  Else  /* Lk already known */
    Delete all sequences in Lk contained in
      some Lp, i>k;
  Answer=∪k Lk;

Function next(k: integer)
Begin
  If (hitk > γ) Return k+2   /* hitk = |Lk|/|Ck| */
End
```

[표 2] TPA 알고리즘

이러한 방식으로 여러 패스를 거치면서 후보 시퀀스집합 혹은 빈발 시퀀스 집합이 생기지 않을 때까지 반복 수행한다. 이와 같은 방식으로 모든 패스의 후보 시퀀스와 빈발 시퀀스들을 구하는데, 전 패스의 빈발 시퀀스와 후보 시퀀스의 비가 임계값 γ이상이 되는 경우가 발생하면 현 패스에서의 빈발 시퀀스를 구하지 않고 다음 패스로 한 단계 건너 뛴다. 역진 단계에서는 전진 단계에서 빈발 시퀀스를 구하지 않고 넘긴 패스들을 대상으로 후 패스의 후보 시퀀스 혹은 빈발 시퀀스 내에 존재하는 서브 시퀀스를 제외시킨 빈발 시퀀스들을 구해 나간다. 예를 들어, 3.1절에서 생성한 데이터 시퀀스를 대상으로 TPA알고리즘을 적용하면 [그림 5]와 같은 각 단계의 빈발 시퀀스들이 구해지고 이들은 색인 페이지의 구성요소가 된다.

| $L_1$ | Support |
|---|---|
| B | 7 |
| D | 4 |
| F | 6 |
| G | 3 |
| H | 4 |

support ≥ 3

| $L_2$ | support |
|---|---|
| BF | 4 |
| FD | 3 |

| $L_3$ | support |
|---|---|
| BHG | 3 |

[그림 5] 빈발 시퀀스

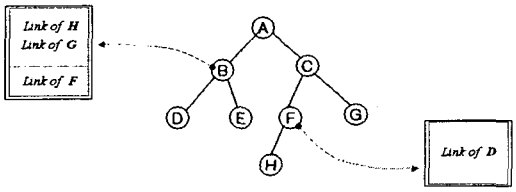
## 4. 색인 페이지의 생성구조

색인 페이지는 기존 웹 상에서 하이퍼 링크로 직접 연결되어 있지는 않지만,이용자 접근 패턴을 분석한 결과를 토대로 연관성이 있다고 추정되는 페이지들을 그 구성요소로 한다. [표 3] IPG 알고리

**Algorithm IPG:**

/* $S$: the set of maximal large sequences */

*While* ($S$ is not empty) *do*
*Begin*
  $S_1 \leftarrow$ the first sequence in $S$
  Create a new index page $P$
  *For* ($k$=1; $k$<=|$S$|; $k$++) *do*
    *If* (*First_Item*($S_1$) = *First_Item*($S_k$)) *then Begin*
      Insert into $P$ the links pointing to each item in *Rest-Item*($S_k$) in order
      Remove $S_k$ from $S$
    *End*
  Insert into the document of *First_Item*($S_1$) a link pointing to the index page $P$
*End*




[그림 6] 생성된 색인 페이지의 구조

즘에서 보여주는 바와 같이, 색인 페이지의 생성 과정은 앞 과정에서 구한 빈발 시퀀스를 대상으로 처음 등장한 문서를 색인 페이지의 생성 위치로 정하고,이어서 등장하는 문서들을 등장 순서대로 정렬하여 해당 문서의 링크를 색인 페이지에 삽입하는 형태를 취한다. [그림 6]는 생성된 색인 페이지의 구조를 보여 주고 있다.

## 5. 결론

본 논문에서는 웹 사이트에 접근하는 이용자 접근 패턴을 학습하여 사이트의 구조나 외형을 자동 개선시켜 나가는 적응형 웹 사이트의 구축 방안으로 색인 페이지의 자동 생성을 제시했다.웹 마이닝 과정에서 적절한 데이터를 얻기 위해 대용량의 서버 로그 데이터를 대상으로 이용자들이 거쳐간 최후 전진 문서만을 가지고 데이터 시퀀스들을 구성하였다. 다음 단계에서 데이터 시퀀스에 순차 접근 패턴 탐색 방법인 TPA 알고리즘을 적용하여 웹 문서간 충분한 지지도를 갖는 연관성 있는 문서들의 시퀀스들을 구해냈다. 이와 같은 빈발 시퀀스들을 이용하여 색인 페이지를 자동 생성시켜 주는 서비스는 이용자들에게 보다 효과적인 정보 접근을 제공할 수 있다.

## 참고문헌

[1] Perkowitz M. and Etzioni O., Adaptive Web Sites: an AI Challenge, In Proc. 15th Int. Joint Conf. AI, 1997.
[2] R. Cooley, B. Mobasher, and J. Srivastava, Web Mining: Information and Pattern Discovery on the World Wide Web, In Proc. of TAI-97, 1997.
[3] R.Agrawal and R.Srikant, Fast Algorithms for Mining Association Rules, In Proc. of the 20th VLDB Conference, pp.487-499, 1994.
[4] M. S. Chen, J. S. Park, and P. S. Yu, Data Mining for Path Traversal Patterns in a Web Environment, In Proc. 16th Int. Conf. Distributed Computing Systems, pp. 385-392, 1996.
[5] D.W. Cheung, Ben Kao, and Joseph Lee, Discovering User Access Patterns on the World-Wide Web, In Proc. of PAKDD-97, 1997.
[6] Srikant, R. & Agrawal, R. (1996). Mining sequential patterns. Research Report RJ 9910, IBM Almaden Research Center, San Jose, California, October 1994.
[7] Park, J. S., Chen, M-S, and Yu, P. S. "An Effective Hash Based Algorithm for Mining Association Rules", In Proc. of ACM SIGMOD-95, 1995.