Incremental scan 방식을 이용한 사용자 웹페이지 추천

강귀영[U] 조동섭
이화여자대학교 컴퓨터학과
{Ryderbay, dscho}@ewha.ac.kr

User Web Page Recommendation Using incremental scan

Kwi-Young Kang[U] Dong-Sub Cho
Dept. of Computer Science and Engineering, Ewha Womans University

요 약

한 사이트 내에서 제공되는 정보가 많아질수록 사용자는 많은 실패를 거친 후 자신이 원하는 정보에 도달하게 된다. 사용자가 어떤 사이트에 자주 찾아오도록 하기 위해서는 적은 노력으로도 원하는 정보에 도달할 수 있도록 도움을 주는 웹 페이지 추천 기법이 필요하다. 기존의 연관규칙이나 순차패턴 기법은 모든 규칙을 찾으므로 필요한 개수 이상의 연산을 한다. 연산 개수가 많아지면 연산 시간이 길어져 갱신되는 데이터베이스를 매번 적용시켜 계산하기가 어렵다. 제안하는 기법은 현재 사용자의 경로 정보를 기준으로 데이터베이스를 변형시키고, 기존 사용자의 경로정보가 저장된 데이터베이스를 검색하여 경로 정보의 패턴을 분석한다. 분석된 결과 중 가장 연관성이 높다고 판단되는 웹 페이지를 현재 사용자에게 추천한다.

1. 서론

사용자가 어떤 사이트에 자주 찾아오도록 하기 위해서는 적은 노력으로도 원하는 정보에 도달할 수 있어야 한다. 사용자가 특정 웹 페이지에서 어디로 가야할지 결정[1]하지 못할 때, 사이트 내부에서 연관성이 있다고 판단되는 웹 페이지를 사용자에게 추천할 수 있다면 연관성 없는 웹 페이지까지 탐색하는 경우를 줄일 수 있을 것이다. 이에 따라 사용자는 탐색 실패 횟수를 줄일 수 있을 것이고, 또한 원하는 정보에 더 빨리 도달할 수 있을 것이다. 한 사이트 내에서 사용자들이 항해를 하게 되면 그 사이트에는 사용자들의 경로 정보들이 쌓이게 된다. 이렇게 쌓이게 되는 사용자들의 경로 정보 속에서 웹 페이지들 간의 연관성이라는 지식이나 패턴을 찾아내기 위해서는 데이터 마이닝 기술이 필요하다.

웹사이트에 접속하여 페이지들을 거쳐간 모든 사용자들의 경로 정보는 시간 순서대로 서버에 저장된다. 한 사용자가 한 번 로그인하여 로그아웃했을 때까지의 경로 정보를 하나의 시퀀스(sequence)로 본다면 이것을 마이닝 하기 위해서는 연관 규칙이나 순차 패턴을 이용하는 것이 가장 개념적으로 적합하다. 그러나 연관 규칙이나 순차패턴의 경우, 모든 규칙을 다 계산한 후에 결과를 제시하므로 매 순간 갱신되는 데이터들을 알고리즘에 적용시켜 추천하기에는 연산 시간이 너무 길어진다. 연산 시간이 길어지면 매번 갱신되는 DB를 적용시켜 웹 페이지를 추천하기가 어려워진다. 본 논문에서는 현재 사용자의 경로 정보를 기준으로 필요한 정보만을 찾아 DB를 생성하고 연산을 수행하는 알고리즘을 제시한다. 비교 분석하는 기준이 되는 시퀀스를 이용하면 DB와 연산 횟수가 줄어들게 되므로 연산 시간이 감소될 수 있다.

이 논문은 2001년도 두뇌한국21사업에 의하여 지원되었음.

2. 연관규칙

T를 현재까지 발생한 모든 트랜잭션이라 하고, I를 m개의 상품들 또는 구매자들이라 하면 T와 I는 다음과 같이 표현된다.

$$T = \{t_1,\ldots,t_n\}$$
$$I = \{i_1,i_2,\ldots,i_m\}$$

어떤 트랜잭션 t에 상품 또는 구매자 w가 포함되어 있을 때, t(w)는 다음과 같이 표현된다.

$$t(w) = 1$$

w는 트랜잭션 t에 존재하는 상품 또는 구매자 중에 하나이다. 또한, W가 상품들의 집합이라고 했을 때, t(W)는 트랜잭션 t에 대해 W에 있는 모든 상품 또는 구매자가 존재함을 말한다.

$$t(W) = 1 : t(w) = 1, \text{모든 } w \in W$$

W : I에 속하는 상품들 또는 구매자들의 부분집합

$$(X) = \{i \mid t_i(X)=1\}$$

(X) 는 X가 포함하는 모든 상품 또는 구매자들을 포함하고 있는 모든 트랜잭션의 집합이 된다. 여기서, X : $|(X)| \geq \sigma$ 이면 X를 σ-covering이라 부른다. 따라서, 모든 트랜잭션에 대해 다음과 같은 법칙을 발견할 수 있다.

W ⇒ B : T에 대해 W⊆R 이고 B⊆R\W 를 만족하는 연관 규칙

W : 연관 규칙의 왼쪽 집합

B : 연관 규칙의 오른쪽 집합

W ⇒ B는 지지도(support threshold)와 신뢰도(confidence threshold)에 의해 발견된다. 지지도는 이 법칙이 적용되는 트랜잭션의 수를 나타내고, 신뢰도는 연관 규칙의 왼쪽 집합이 가리키는 전체 트랜잭션 개수 중에서 이 법칙을 만족하고 있는 트랜잭션의 개수를 나타낸다. 따라서, 지지도와 신뢰도에 의해 다음의 연관 규칙이 발견된다.

T 는 W⇒B 을 만족한다.

단, W∪B : σ-covering **(σ : 지지도**

(i.e., $|(W \cup B)| \geq \sigma$) **γ : 신뢰도**

$|(W \cup B)|/|(W)| \geq \gamma$)

연관 규칙 발견은 간단히 2가지 알고리즘에 의해 쉽게 발견될 수 있다. 하지만, 이들 알고리즘은 모든 아이템에 대해 가능한 부분집합들을 구해야 하므로 NP-hard로 취급된다. 따라서, 성능을 개선하기 위해 병렬 알고리즘과 같은 다양한 방법에 대한 연구들이 활발히 진행되고 있다[5].

3. 제안하는 알고리즘

이 기법의 기본적인 개념은 현재 사용자가 지나온 경로를 되짚어 가면서, 현재 사용자가 지나온 페이지를 인접한 순서대로 가장 많이 지나온 트랜잭션(transaction)을 찾아 그들 중 Count가 가장 큰 다음 페이지 Next를 추천하는 것이다.

그림 1을 보면, 현재 사용자는 C 페이지에 있고 이 페이지에 오기 전에 B와 A를 역순으로 거쳤다. 초기 데이터베이스 DB는 기존 사용자들의 로그인부터 로그아웃까지의 경로정보를 하나의 트랜잭션으로 묶어 모은 것이다. Count는 Next가 나온 횟수이다. 초기 데이터베이스 DB에서 모든 트랜잭션에 대해 현재 페이지 C가 있는 항목들을 선택하여 발견된 C의 다음 페이지 정보까지 잘라 데이터베이스 D_0를 구성한다. 동시에 선택된 각 트랜잭션 내에서 C의 Next를 찾아 Count를 계산한 P_0를 만든다. 계산된 Next의 Count들 중 사용자에게서 입력받은 Support 이상의 것들만 선택해 결과 파일에 한 개의 경로를 포함한 결과로써 저장한다. 다음 반복에서는, D_0를 스캔하여 현재 페이지 C의 바로 이전에 거쳤던 페이지 B가 C의 이전에 있는 트랜잭션들을 선택하여 D_1을 구성한다. 동시에 선택된 각 트랜잭션 내에서 C의 Next를 찾아 Count를 계산한 P_1을 구성한다. 계산된 Next의 Count들 중 Support 이상의 것들만 두 개의 경로를 포함한 결과로써 결과 파일에 저장한다.

다음 반복에서도 이전 반복과 마찬가지로, D_1을 스캔하여 현재 페이지 C의 두 번째 이전 페이지인

표 1 웹 페이지 추천 알고리즘

```
k = 0;
ch = pop(stack); //stack:현재 사용자의 경로정보 저장
                 //ch:현재 사용자의 현재 페이지
//make transformed DB
forall transaction t of DB
    if ch ∈ t then
        첫 page부터 ch의 next page까지 잘라
transformed DB에 저장
        { Next = ch's next page
          Count 계산 } //Pk 계산
    end if
end forall
forall Next
    if Count >= Support then
       Save Next & Count in result.txt
    end if
end forall

// compare and select transactions
while ([Dk]! = 0 or stack underflow){
//stack underflow : 현재사용자 경로정보의
//                  첫 페이지까지 읽었을 때
    k++;
    ch = pop(stack);
    forall transaction t of Dk
        if ch ∈ transaction t of Dk then
            save t in Dk
            { Next = ch's next page
              Count 계산 } //Pk 계산
        end if
    end forall
    forall Next
        if Count >= Support then
           Save Next & Count in result.txt
        end if
    end forall
end while

select one result
```

페이지 A가 C의 이전에 있는 트랜잭션들을 선택하여 D_2를 구성한다. 동시에 선택된 각 트랜잭션 내에서 C의 Next를 찾아 Count를 계산한 P_2를 구성한다. 계산된 Next의 Count들 중 Support 이상의 것들만 세 개의 경로를 포함한 결과로써 결과 파일에 저장한다. 더 이상 탐색할 페이지가 없으므로 반복을 중지한다. 탐색할 페이지가 더 있더라도 데이터베이스 D_k가 더 이상 생성되지 않는다면 반복을 중지한다. 그림 3에서 구체적인 알고리즘 진행 과정을 보여준다.

k를 현재 페이지와 탐색하는 페이지 사이의 거리라고 볼 때 매 반복은 D_{k-1}을 스캔하여 D_k와 P_k를 만드는 과정으로 볼 수 있다.

조건을 만족하는 항목들 중 Support 이상의 P_k를 가진 항목 중 가장 큰 k 값을 가진 $P_{(max\ k)}$에 있는 Next를 추천한다. Next가 여러 개일 경우에는 Count가 가장 큰 페이지를 추천한다. 만일 추천하

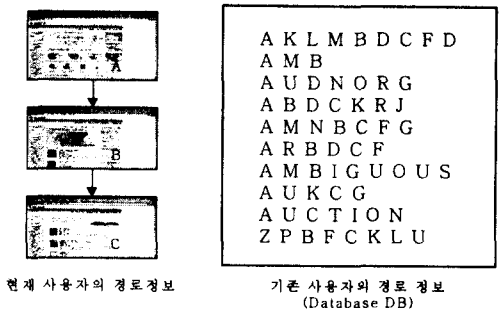

그림 1 사용자 경로정보와 초기 데이터베이스

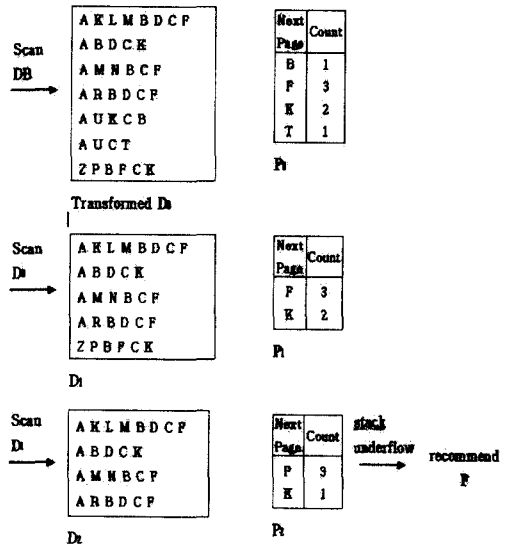

그림 2 추천 페이지 계산 과정

고자 하는 페이지가 삭제되었을 경우에는 다음 우선순위를 가진 페이지를 추천한다.

4. 결론 및 향후 과제

본 논문에서는 한 사이트 내에서 현재 사용자가 지나온 경로정보에 따라 동적으로 관련된 웹 페이지를 계산하여 추천하는 기법을 설명하였다. 기존의 웹 페이지 추천 시스템에 대한 연구는 한 사이트 내에서의 웹 페이지 추천이 아닌 월드 와이드 웹 상에서의 웹 페이지 추천에 대한 연구가 주류를 이루었다. 기존의 시스템들은 기계 학습 방법(Machine Learning)을 이용하여 나온 결과가 사용자에게 적합한 것이었는지를 인증 받고 다시 이 결과를 토대로 학습해야 하므로 전자상거래와 같이 한 사이트 내의 웹 페이지에 대해 추천을 할 경우에는 적합하지 않았다. 또한, 연관규칙과 같은 범용적인 데이터 마이닝 방식을 적용시킨다면 모든 가능한 경우들을 다 계산하므로 연산 횟수가 필요 이상으로 많아지게 된다. 본 논문에서 제시한 기법은 한 사이트 내에서 각 사용자가 지나온 경로정보를 이용하므로 사용자의 인증을 따로 받지 않아도 되고, 필요한 부분만 찾도록 계산하기 때문에 연산횟수가 감소된다. 따라서 변화하는 상황과 데이터에 좀 더 빠르게 대응하는 추천을 제공할 수 있다.

향후에 다른 추천 기법들과 비교해 얼마나 많은 시간 절약을 가져오는지를 실험해야 할 것이다. 데이터베이스 DB를 전체 사용자의 경로정보로 보지 않고, 사용자를 그룹화 하여 각 그룹의 경로정보로 본다면 현재 사용자에게 좀 더 개인화 된 페이지를 추천할 수 있을 것이다. 또한, 새 페이지, 최근 갱신된 페이지에 대해서도 우선순위를 적용시킬 수 있는 방법을 모색해야 할 것이다.

5. 참고 문헌

[1] Gabriela Polcicova, "Recommending HTML-documents using Feaure Guided Automated Collaborative Filtering," ACM SIGIR '99, August 1999.

[2] Thorsten Joachims, Tom Mitchell, Dayne Freitag, and Robert Armstrong, "WebWatcher: Machine Learning and Hypertext," Fachgruppentreffen Maschinelles Lernen, Dortmund, August 1995.

[3] Thorsten Joachims, Dayne Freitag, Tom Mitchell, "WebWatcher: A Tour Guide for the World Wide Web," IJCAI97, Aug. 1997.

[4] Robert Armstrong, Dayne Freitag, Tom Mitchell, and Thorsten Joachims, "Web Watcher: A Learning Apprentice for the World Wide Web," 1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments, March 1995.

[5] 문홍기, 이수원, "전자 상거래 에이전트를 위한 연관 규칙 발견 및 확장," 1999년 춘계정보과학회, August 1999.

[6] Jong Soo Park, Ming Syan Chen and Philip S. Yu, "An Effective Hash-Based Algorithm for Mining Association Rules," 1995 ACM SIGMOD Conference, pages 175--186, San Jose, California, USA, May 1995.

[7] Sanford Gayle, "The Marriage of Market Basket Analysis to Predictive Modeling," The Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 2000.