사이트간 웹 사용 마이닝을 위한 데이터 전처리의 성능 향상

현우석°
한국성서대학교 정보과학부
wshyun°@bible.ac.kr

Performance Improvement of Data Preprocessing for Intersite Web Usage Mining

Woo-Seok Hyun°
Dept. of Information Science, Korean Bible University

요 약

매일 새롭게 생기는 웹 페이지 수가 수천만 개, 온라인 문서들의 수가 수십억 개에 이르게 되자, 웹 사이트를 설계함에 있어서 웹 서버 로그 파일에 기록된 사용자의 행동을 분석하는 것이 중요한 부분이 되어 가고 있다. 분석가들은 전체 웹 사이트에서 사용자 행동의 완전한 개요를 알기 원하기 때문에 고객이 방문했던 모든 다른 웹 서버를 통하여 사용자의 패스(path)를 다시 수집해야만 한다. 본 연구에서는 모든 로그 파일을 연결해서 방문했던 곳을 재구성하는 향상된 데이터 전처리 방법에 의하여 실험을 하여 로그 파일 크기를 감소시키게 되어 데이터 전처리의 성능이 향상되었음을 보였다.

1. 서 론

월드 와이드 웹(world wide web)이 사람들 사이의 정보 전달(communication)에 있어 혁명을 일으키게 된후, 매일 만들어지는 새로운 웹 페이지의 수는 수천만 개, 온라인 문서들은 수십억 개에 이르고 있다[1]. 이러한 자료들이 증가함에 따라 웹 사이트 발행자들은 사용자들이 특정 웹 사이트를 계속 사용하게 하는데 있어서 어려움을 호소하고 있다. 보편적인 웹 사이트를 설계하기 위하여 발행자들은 사용자의 요구를 이해해야만 하기 때문에, 웹 서버 로그 파일에 기록된 사용자의 행동을 분석하는 것이 설계에 있어서 중요한 한 부분이 되고 있다.

WUM(web usage mining)은 웹 사이트의 사용자 접근을 분석하기 위한 데이터 마이닝 절차에 응용된다. KDD(knowledge discovery and data mining) 과정에 따르면 웹 사용 마이닝은 전처리(preprocessing), 지식 추출(knowledge extraction), 결과 분석(result analysis) 등의 세 단계로 이루어진다. 본 논문은 데이터 전처리에 대하여 연구한다. 분석가들은 웹 사이트에 접속했던 사용자들의 정확한 목록을 결정하는 것과 웹 사이트에서 각 사용자가 수행했던 일련의 행동들인 사용자 세션(user session)을 재구성하는 것을 목표로 한다. 이것을 위하여 분석가들은 로그 파일만을 사용하기도 하고 어떤 경우에는 사이트 맵까지 사용하기도 한다.

사이트간 WUM은 몇 개의 웹 사이트로부터 웹 서버 로그를 다루는데, 일반적으로 같은 조직(organization)에 속하게 된다. 오늘날 중요한 조직은 웹 사이트를 위해서 몇 개의 웹 서버를 가지고 있다. 웹 페이지들이 다른 서버와 상호 연결되어 있기 때문에 사용자들은 이러한 모든 서버들을 통하여 인터넷상에서 항해하게 된다. 고객들이 브라우저의 주소창을 보아야만 웹 서버가 변경되었다는 것을 알 수 있기 때문에, 웹 사이트를 방문한 고객들은 웹 서버가 변경되었다는 것에 자주 주목하지 못하고 있는 실정이다.

그러나 분석가들은 전체 웹 사이트에서 사용자 행동의 완전한 개요를 알기 원하기 때문에 사용자 중심의 분석을 수행하는 WUM 분석가들을 위해서 이러한 변화는 중요하게 되었다. 분석가들은 고객이 방문했던 모든 다른 웹 서버를 통하여 사용자의 패스(path)를 다시 수집해야만 한다. 이것을 위하여 본 연구에서는 모든 로그 파일을 연결해서 방문했던 곳을 재구성하는 방법[2]을 사용한다.

전통적인 데이터 전처리는 데이터 융합(data fusion), 데이터 클리닝(data cleaning), 데이터 구성(data structuration) 등의 세 가지 단계를 포함한다. WUM을 위하여 본 연구에서는 향상된 데이터 전처리 방법[2]을 추가한다. 이것은 분석가들이 관심 있는 정보만을 선택할 수 있도록 하는 데이터 요약 단계(data summarization step)로 구성된다. 본 연구에서는 한국성서대학교 웹 사이트의 로그 파일들을 가지고 실험을 하여 데이터 전처리의 성능이 향상되었음을 보였다.

2. 정의와 문제 정형화

웹 특성 용어에 관한 World Wide Web Consortium의

연구에 따라서, 주요한 WUM 용어들의 정의를 다시 정리하고 방문, 에피소드와 웹 서버 로그 파일 등의 용어에 대하여 정의[2]를 내리며, WUM 데이터 전처리 문제를 정형화[2]한다.

2.1 정의

자원(resource)이란 W3C에 따르면 고유성을 지니는 어떤 것을 말한다. 그러므로 URI(Uniform Resource Identifier)란 추상적인 혹은 물리적인 자원을 인식하기 위한 문자열이 된다. 가능한 예로서 HTML 파일, 이미지, 웹 서비스 등을 포함한다.

웹 자원(web resource)이란 HTTP 1.1 혹은 HTTP-NG 등 HTTP 프로토콜의 어떤 버전을 통해서도 접근할 수 있는 자원이다.

웹 서버(web server)란 웹 자원에 대한 접근을 제공하는 서버이다.

웹 페이지(web page)란 URI에 의해서 인식될 수 있는 몇몇 웹 자원 혹은 하나의 웹 자원을 구성하는 데이터의 집합이다.

페이지 뷰(page view)란 웹 브라우저가 웹 페이지를 표시할 때, 일정한 시간 내의 특정 순간에 발생한다.

웹 브라우저 혹은 웹 클라이언트는 웹 요청을 하고 응답을 해 주며 요청된 URI를 보여주는 클라이언트 소프트웨어이다.

사용자(user)란 웹 브라우저를 사용하는 사람이다.

웹 요청(web request)이란 웹 클라이언트가 웹 자원을 만들게 하는 요청이다. 이것은 사용자에 의해서 시작될 수도 있고 웹 클라이언트에 의해서 시작될 수도 있다.

사용자 세션(user session)이란 한 개 이상의 웹 서버에 대한 사용자의 명백한 웹 요청의 명확한 수(delimited number)를 말한다.

방문(visit)이란 사용자 세션에서 충분히 발생하는 연속적인 페이지 뷰들의 부분집합이다.

에피소드(episode)란 관련된 클릭으로부터 구성된 방문의 부분집합이다. 예를 들면 kr.yahoo.com에 대한 사용자의 방문은 스포츠 용품을 주문하고 재고 가격을 확인하며 사진을 찾는 등의 세 가지의 에피소드들로 구성된다.

웹 서버 로그 파일은 연대순으로 기록된 웹 서버에 대한 요청으로 구성된다. 가장 보편적인 로그 파일 형식은 Common Log Format(www.w3.org/Daemon/User/Config/Logging.html#common-logfile-format)과 Extened CLF이다. ECLF에서 한 라인에는 클라이언트의 host name 혹은 IP 주소, 사용자 로그인, 요청된 날짜와 시간, 운영 유형(GET, POST, HEAD 등), 요청된 자원의 크기, 요청된 상태, 요청된 페이지의 크기, 사용자 에이전트와 조회인(referrer) 등을 포함한다.

2.2 문제 정형화

다음과 같이 문제를 정형화한다. 웹 사이트의 웹 자원의 모든 집합인 R 은 수식 (1)과 같이, 그 사이트에 접근한 모든 사용자의 집합인 U 는 수식(2)와 같이 정의한다.

$$R = \{r_1,\ r_2,\ \ldots\ r_{nR}\} \quad \ldots\ldots(1)$$

$$U = \{u_1,\ u_2,\ \ldots\ u_{nU}\} \quad \ldots\ldots(2)$$

로그 엔트리(entry)는 수식(3)과 같이 정의하는데, 여기에서 $u_i \in U$ 하고 r_i, $ref_i \in R$ 하며 t는 접근시간, s는 요청 상태, ref는 임의로서 초기 로그 파일에 없다면 추측되어질 수 있다. s는 세 자리 숫자코드로서 요청에 대한 성공이나 실패를 나타낸다.

$$l_i = <\ u_i;\ t;\ s;\ r_i;\ [ref_i]\ > \quad \ldots\ldots(3)$$

$$L = \{\ l_1,\ l_2,\ \ldots,\ l_{nL}\ \} \quad \ldots\ldots(4)$$

웹 서버 로그는 수식(4)와 같이 시간값 t에 의해 오름차순으로 정렬된다. 몇몇 웹 서버들인 1,2, ..., N은 이러한 경우에 수식 (5)와 같이 로그 파일의 집합이 된다.

$$Log = \{\ L_A,\ L_B,\ L_C,\ \ldots\ \} \quad \ldots\ldots(5)$$

앞에서 언급했던 것처럼 사이트 맵은 페이지 뷰, 방문과 에피소드들을 인식하기 위해 전처리 동안에 유용하다. 웹 사이트 맵은 Rovert Cooley에 의하면 수식(6)[4]과 같이 정형화되었는데, 여기에서 M은 사이트 맵, F는 프레임을 나타낸다.

$M=<F_1;\ \ldots;\ F_n>$, $F=\{h_1, <L_1, \ldots, L_m>\}$,

$L=<r, (h_1, g_1)|\ldots|(h_p, g_p)>$, ... (6)

전처리에서는 가능하다면 로그와 사이트 맵을 사용하여 로그 엔트리들을 페이지뷰로 그룹화한다. 페이지 뷰는 수식 (7)과 같이 구성되는데, 여기에서 $r_{ij} \in R$의 관계가 성립한다.

$$p_i = \{\ r_{i1},\ r_{i2},\ \ldots,\ r_{np}\ \} \quad \ldots\ldots (7)$$

시간간격 Δt를 고려하면, 사용자 u_i에 대한 방문 v_i는 수식(8)과 같이 정의되며, 여기에서 $pv_i=<(t_1,\ p_1);\ (t_2,p_2);\ \ldots;\ (t_n,\ p_n)>$, $t_{i+1} \geq ti$, $t_{i+1}-t_i<\Delta t$, $i = \overline{1..n-1}$ 이다.

$v_i = <u_i,\ t,\ pv_i>$ .....(8)

본 연구에서는 주어진 몇몇 웹 사이트를 위해 로그파일의 집합 Log와 웹 사이트 맵 M를 가지고 주어진 Δt 동안에 웹 사이트 사용자의 방문들을 추출해 내는 것으로 문제를 정형화하였다.

3. 데이터 전처리

그림 1은 데이터 융합, 데이터 클리닝, 데이터 구성과 데이터 요약 등의 데이터 전처리의 네 가지 단계[2]를 보여주고 있다.

3.1 데이터 융합(data fusion)

데이터 전처리 초기에 몇몇 웹 서버뿐만 아니라 웹 사이트 맵으로부터 웹 서버 로그 파일을 가지고 처리를 하게 된다. 본 연구에서는 먼저 로그 파일을 결합하고 사적인 이유 때문에 결과 로그 파일을 익명으로 만든다.

3.1.1 로그 파일 합치기

먼저 Log 로부터 다른 로그 파일을 결합하게 되는데, 그림 2는 이 과정을 하기 위한 알고리즘[2]을 보여준다.

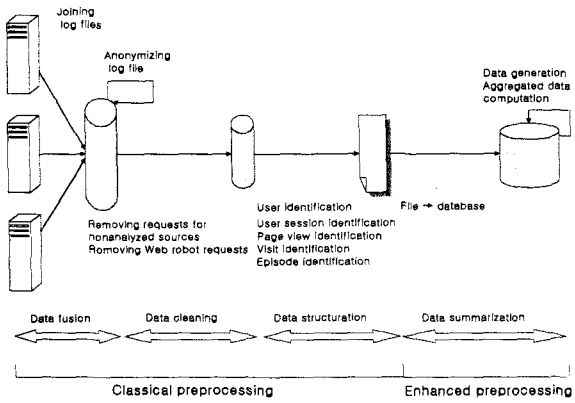

그림 1 데이터 전처리의 네 가지 단계

```
Procedure JoinLogs (Log)
  T = ∅
  for        i = 1, N
    Li.c=1
    t = Li.l.time
    InsertT (i, t)
  while T.length >1
    While T[1].t > T[2].t
       id = T[1].id
       l'=S[id] + Lid.l
       L^J = L^J ∪ l'
       Lid.C = Lid.c + 1
       if EOF(Lid)
         RemoveT(T[1])
         break
       end if
    end while
    if not EOF(Lid)
      OrderT(1, T[1])
    end if
  end while
  while not EOF(LT[1].id)
     id = T[1].id
     l' = S[id] + Lid.l
     L^J = L^J ∪ Lid.l
     Lid.c = Lid.C + 1
  end while
End JoinLogs
```

그림 2 로그 파일을 합치기 위한 알고리즘

3.1.2 로그 파일을 익명으로 만들기

본래 호스트 이름 혹은 IP 주소를 암호화한다. 그래서 본래 호스트 이름을 .com, .ac, .co 등 도메인 확장에 관한 정보를 가지고 있는 인식자로 대치시킨다.

3.2 데이터 클리닝(data cleaning)

이 단계에서는 로그 파일로부터 필요 없는 요청들을 제거하는 일을 한다. 일반적으로 본 과정에서는 이미지와 멀티미디어 파일과 같은 분석되지 않은 자료와 관련된 요청들을 제거하게 된다. 또한 웹 로봇을 인식하고 그들의 요청을 제거한다.

본 연구에서는 불필요한 데이터를 제거함에 의해서 기억 장치를 적게 사용하게 되었고 로그 파일 크기를 감소시키게 되었다. 예를 들면, 이미지 요청을 제거함에 의해서 한국성서대학교 웹 서버 로그 크기를 본래 크기보다 45%정도 감소시킬 수 있었다.

3.2.1 분석되지 않은 자원을 위한 요청을 제거하기

이미지를 위해서 로그 파일을 유지하거나 제거하는 것은 웹 사용 마이닝의 목적에 의존하게 된다. 웹 저장(caching)이 목적이라면 분석가들은 이미지나 멀티미디어와 관련된 로그 엔트리들을 제거해서는 안 된다. 만약 분석가들이 사이트 구조의 부족한 점을 찾기를 원한다면 사용자의 행동을 표현하는 명백한 요청들을 유지되어야 한다.

3.2.2 로봇의 요청을 제거하기

웹 로봇은 적절한 내용을 추출하기 위해서 웹 사이트를 자세히 조사하는 소프트웨어 도구이다. 웹 로봇 호스트를 인식하기 위해서 다음과 같은 세 가지 귀납법(heuristics)을 사용한다. 첫째, "robots.txt" 페이지를 요청한 모든 호스트들을 찾는다. 둘째, 로봇으로 알려진(www.robotstxt.org/wc/robots.html) 사용자 에이전트의 목록을 사용한다. 셋째, 호스트가 웹 로봇인지 추측한다. 모든 웹 로봇이 인식이 된다면, 웹 로봇이 만들어 낸 요청을 제거하게 된다.

3.3 데이터 구조화(data structuration)

이 단계에서는 사용자, 사용자 세션, 페이지 뷰, 방문, 에피소드 등에 의한 로그 파일의 비구조적 요청을 그룹화한다.

3.3.1 사용자 인식(user identification)

대부분의 경우에 로그 파일은 IP 등의 컴퓨터 주소와 ECLF 파일 등의 사용자 에이전트만을 제공한다. 웹 사이트가 사용자 등록을 요청했을 경우에 로그 파일은 사용자 로그인을 포함하게 되는데 이 경우에는 사용자 인식을 위해서 이 정보를 사용한다.

3.3.2 사용자 세션 인식(user session identification)

로그 파일로부터 사용자 세션을 인식하는 것은 도서관 등 동일한 컴퓨터를 여러 명의 사용자가 사용하는 경우와 한 명의 사용자가 여러 대의 컴퓨터를 사용하는 경우가 있는 경우에 단순하지가 않다. 그러나 몇몇 기술들은 추가적인 정보를 제공할 수 있다. 가장 보편적인 것으로는 쿠키, 동적 웹 페이지(URL에서 세션 ID), 사용자 등록 등을 들 수 있다.

한국성서대학교 웹 서버 로그 파일에 대해서 사용자 로그인 정보를 사용하지 않고 (Host, User Agent) 정보만을 사용하였다.

3.3.3 페이지 뷰 인식(page view identification)

사이트 맵 M을 사용하여 다음과 같은 알고리즘에 의해 페이지 뷰에 대한 요청을 그룹화한다. 첫째, 페이지 뷰 p_i에 대한 요청이 로그 파일 안에 있으면, 내포된 자

원에 해당하는 로그 엔트리들을 p_i에서 제거하고 p_i를 위한 요청을 유지한다. 둘째, p_i에 대한 요청이 없고 대응하는 자원에 대한 몇몇 엔트리들만 있다면, 대응하는 자원에 대한 엔트리를 p_i에 대한 요청으로 대체한다. 본 연구에서는 이 요청에 대한 시간을 t_i = min{time(l_i)}로 보았다. 여기서 l_i 는 자원 r_i를 위한 대응하는 로그 엔트리이다. 이 단계를 마치게 되면 로그 파일은 한 명의 사용자 활동을 위한 요청을 포함하게 된다.

3.3.4 방문 인식(visit identification)

사용자를 위해서 본 연구에서는 일련의 페이지 뷰를 획득한다. 이것은 일정 기간동안 웹 사이트에서 연속적인 일련의 클릭을 나타내는데 다음과 같은 귀납법이 방문과 사용자 세션을 분리하기 위해서 사용되었다. 첫째, H_{page}는 연속적인 요청들 사이에서의 시간 차를 측정하기 위해서 한계(threshold) Δt를 사용한다. 둘째, H_{visit}은 전체 방문을 위해서 시간 한계를 사용한다[5-6]. 셋째, H_{ref}는 방문 기록과 참고인을 사용한다. 넷째, MF(maximal forward)는 사용자가 전에 방문을 했던 페이지에 대한 두 번째 요청은 유지하지 않는다[7].

3.3.5 에피소드 인식(episode identification)

에피소드 인식은 웹 페이지의 시멘틱 정의에서의 거리 측정과 전체 웹 사이트의 시멘틱 정의를 필요로 하기 때문에 어려운 문제이다. 본 연구에서는 시멘틱 주제의 계층구조와 에피소드를 인식하기 위한 시멘틱 거리를 사용한다. 이러한 거리가 정의된 한계를 초과했을 때에는 새로운 에피소드가 시작된다.

3.4 데이터 요약(data summarization)

이 단계 초기에는 파일을 관계형 데이터베이스로 바꾸게 된다. 데이터 요약 단계에서는 요청 수준에서 데이터 일반화, 방문과 사용자 세션을 위한 집단화된 데이터 계산을 적용한다.

3.4.1 구조적 로그 저장하기

본 연구에서는 고전적 전처리에서 인식되는 각 객체를 위한 관계형 데이터베이스에서 다른 테이블을 설계한다. 또한 존재하는 테이블에 새로운 속성 혹은 새로운 테이블을 첨부함에 의해서 이 모델을 확장할 수 있다. 이러한 데이터를 마이닝할 때, 분석가들은 관심 있는 정보만을 선택할 수 있다. 이러한 테이블을 만들기 위해서 데이터 일반화와 집단화된 데이터 계산을 적용한다.

3.4.2 데이터 일반화

이 과정은 지속적으로 수를 감소시키기 위해서 문법적으로 혹은 의미적으로 URL 집합으로 변형시킨다.

문법적인 일반화 대신에 본 연구에서는 같은 의미의 주제하에 유사한 내용을 지니는 페이지들로 그룹화한다.

3.4.3 집단화된 데이터 계산

이 과정에서는 방문 혹은 사용자 세션을 위한 새로운 파라미터들을 계산한다. 데이터 마이닝에서는 나중에 이러한 파라미터들을 사용하게 된다[8]. 이 파라미터들은 분석화된 객체를 특징짓는 통계적인 값들을 표현한다.

4. 실험 및 평가

본 연구에서는 서로 연결되어 있는 두 개의 웹 서버로 구성되어 있는 한국성서대학교 웹 서버를 사용하였다.

두 개의 한국성서대학교 웹 서버(www.bible.ac.kr, mission.bible.ac.kr)와 두 개의 검색 엔진 사이트를 가지고 3개월 동안 로그 파일의 데이터 집합을 대상으로 실험을 하였다. 데이터 집합의 크기는 3,432 Mbyte였다. 표 1 은 분석되지 않은 자원을 제거한 결과를 보여준다. 이 과정에서 융합된 로그 파일이 초기 데이터 집합 크기와 비교했을 때 46%정도 감소되었다.

표 2 는 웹 로봇 요청을 제거한 각 방법에 대해 상세한 정보를 보여준다. 이 표에서는 분리되어 각 방법에 적용하고 연속적으로 3개월 동안 적용함에 의해서 감소가 되었다는 것을 보여주고 있다. 웹 로봇 요청을 제거함에 의해서 융합된 로그 파일이 초기 크기와 비교했을 때 52.6% 정도 감소되었음을 보여준다.

표 1 분석되지 않은 자원에 대한 요청을 제거함에 의해 로그 파일 크기를 감소시킨 결과

	Jun. 2006	Jul. 2006	Aug. 2006	Total
Logs for www.bible.ac.kr	897	896	1,068	2,862
Logs for mission.bible.ac.kr	507	449	597	1,554
Total logs	1,407	1,352	1,673	4,431
Logs cleaned	602	680	760	2,042
% of initial size	42.8	50.2	45.5	46.0

표 2 웹 로봇 요청을 제거함에 의해 로그 파일 크기 (Mbytes)를 감소시킨 결과

	Jun.	Jul.	Aug.	Mbytes	Total %
initial size	603	679	759	2,042	100
robot.txt	415	431	519	1,364	66.8
Known user agent	536	590	691	1,816	89.0
browsing speed	528	605	691	1,825	89.3
All three methods	314	339	440	1,093	52.6

5. 결론 및 향후 과제

본 연구에서는 데이터 융합(data fusion), 데이터 클리닝(data cleaning), 데이터 구성(data structuration) 등의 세 가지 단계를 포함하는 전통적인 데이터 전처리뿐만 아니라 WUM을 위하여 데이터 요약을 포함하는 향상된 데이터 전처리 방법을 사용하여 한국성서대학교 웹 사이트의 로그 파일들을 가지고 실험을 하여 로그 파일 크기를 감소시키게 되어 데이터 전처리의 성능이 향상되었음을 보였다.

향후 연구과제로는 XGMML 소스 파일 등의 사이트 구조의 다른 버전들을 효과적으로 다룰 수 있는 방법에 대한 향후 연구가 남아있다.

참고문헌

[1] I. Turner, "The One-Stop Portal," Line56,8 Oct. 2002, www.line56.com/articles/default.asp?ArticleID=4075.

[2] Doru Tanasa and Brigitte Trousse, "Advanced

Data Preprocessoing for Intersites Web Usage Mining", IEEE Intelligent Systems, March/April, pp. 59-65, 2004.

[3] T. Berners-Lee, R.Fielding, and L.Masinter, "Uniform Resource Identifiers(URI): Generic Syntax," Network Working Group RFC2396, Aug. 1998, www.rfc-editor.org/rfc/rfc2396.txt.

[4] R. Cooley, Web Usage Mining: Discovery and Application of Interesting Patterns from Web Data, PhD thesis, Dept. of Computer Science, Univ. of Minnesota, 2000.

[5] R. Cooley, B. Mobasher, and J. Srivastava, "Data Preparation for Mining World Wide Web Browsing Patterns," J. Knowledge and Informations Systems, vol. 1, no. 1, pp.5-32, 1999.

[6] Y. Fu, K. Sandhu, and M. Shih, "A Generalization-Based Approach to Clustering of Web Usage Sessions," Proc. 1999 KDD Workshop Web Mining, LNCS 1836, Springer-Verlag, pp. 21-38, 2000.

[7] M.S. Chen, J.S. Park and P.S. Yu, "Data Mining for Path Traversal Patterns in a Web Environment," Proc. 16th Int'l Conf. Distributed Computing Systems(ICDCS 96), IEEE CS Press, pp.385-392, 1996.

[8] M. Arnoux et al., "Automatic Clustering for the Web Usage Mining," Proc. 5th Int'l Workshop Sumbolic and Numeric Algorithms for Scientific Computing(SYNASC 03), Editura Mirton, pp.54-66, 2003.