웹 마이닝의 개념과 기술

김 인 철*

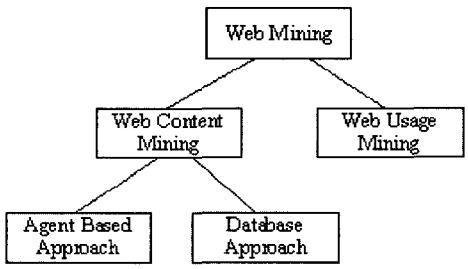
◈ 목 차 ◈

1. 서 론

오늘날 인터넷 월드와이드웹(World Wide Web)은 뉴스, 광고, 소비자 정보, 자산 관리, 교육, 정부, 전자상거래 분야 등을 총 망라하는 하나의 거대한 분산 정보서비스 센터의 역할을 수행한다. 이밖에도 웹은 많은 하이퍼링크 정보와 웹 서버 로그와 같은 웹 페이지 이용 정보들을 포함하고 있으므로, 데이터 마이닝(data mining)을 위한 풍부한 자원을 제공한다. 웹 마이닝(Web mining)은 웹으로부터 유용한 정보를 발견하고 분석하는 작업으로 폭넓게 정의할 수 있다. 하지만 웹 마이닝이란 용어는 그동안 두 가지 서로 다른 작업을 나타내는데 사용되어 왔다. 웹 내용 마이닝(Web content mining)은 온라인상의 정보자원들에 대한 자동화된 검색을 통해 정보를 발견하는 과정을 나타낸다. 이에 반해, 웹 이용 마이닝(Web usage mining)은 웹 서버로부터 이용자들의 접근 패턴을 찾아내는 과정을 나타낸다. 본 논문에서는 웹 마이닝의 개념에 대해 정의하고, 웹 마이닝에 관련한 다양한 연구주제와 기술, 그리고 도구들에 대해 간략히 고찰해본다. 그리고 끝으로 웹 이용 마이닝 기술을 이용하여 적응형 웹 사이트를 구축하는 방법에 대해 설명한다.

2. 웹 내용 마이닝

웹 마이닝(Web mining)은 그림 1과 같이 크게 웹



(그림 1) 웹 마이닝의 분류

내용 마이닝(Web content mining)과 웹 이용 마이닝(Web usage mining)으로 나눌 수 있다. 본 절에서는 먼저 웹 내용 마이닝에 관한 최근 연구와 기술들을 간략히 살펴본다. 일반적으로 웹상의 정보 자원들은 명확한 구조가 없으므로 자동화된 방법으로 유용한 정보를 찾기 어렵다. 'Lycos, Alta Vista, WebCrawler, MetaCrawler'와 같은 전통적인 검색엔진들은 웹 정보 이용자들에게 어느 정도의 편의성을 제공하지만, 대부분 구조정보를 제공하지 못할 뿐 아니라 웹 문서를 자동으로 분류하거나 여과하거나 해석하는 기능을 제공하지 못한다. Srivastava의 연구[21]에서는 가장 보편적으로 이용되는 검색엔진들의 기능과 성능에 대한 포괄적인 비교평가를 다루고 있다. 최근 들어 이와 같은 검색엔진들의 제한성을 극복하기 위한 노력의 일환으로서, 웹 에이전트(Web agent)와 같이 정보검색을 위한 보다 지능적인 도구를 개발하려는 연구와 데이터마이닝 기술들을 확장하여 웹 정보자원을 보다 높은 수준으로 조직화하려는 연구가 활발하다.

* 경기대학교 정보과학부 전자계산학전공 부교수

2.1 에이전트 기반의 접근법

일반적으로 에이전트 기반(agent-based)의 웹 마이닝 시스템들은 다시 아래와 같이 지능형 검색 에이전트들(intelligent search agents), 정보 여과 및 분류 에이전트들(information filtering /cate gorization agents), 그리고 개인화된 웹 에이전트들(personalized Web agents)로 크게 나눌 수 있다.

2.1.1 지능형 검색 에이전트

영역특성과 이용자 프로파일을 이용하여 관련 정보를 검색하고, 발견된 정보를 해석하는 지능형 웹 에이전트들이 최근에 많이 개발되었다. Harvest[3], FAQ-Finder[12], Information Manifold[16], OCCAM[19] 등이 이러한 지능형 검색 에이전트의 대표적인 예들이다. 이들은 관련 웹 문서들을 검색하고 해석하기 위해, 특정 문서들에 대한 영역 정보나 고정된 모델들을 이용한다. ShopBot[8]와 ILA(Internet Learning Age nt)[28]와 같은 에이전트들은 잘 알려져 있지 않은 정보 자원들(information sources)과 상호작용하면서 그 구조를 학습한다. ShopBot는 상품영역에 관한 일반적인 정보만을 이용하여 다양한 판매사들의 사이트로부터 제품 정보를 검색한다. ILA는 다양한 정보 자원들에 대한 모델을 학습하고, 이들을 하나의 개념 계층(concept hierarchy)으로 변환하는 기능을 제공한다.

2.1.2 정보 여과 및 분류 에이전트

한편, 다양한 정보 검색 기술들과 개방형 웹 문서의 특성들을 이용함으로써 웹 문서에 대한 여과(filtering)와 분류(categorization)를 자동화한 웹 에이전트들도 많이 개발되었다[5,11].

HyPursuit[34]는 하이퍼텍스트 문서들에 대한 계층적 군집화와 정보 자원의 구조화를 위해 링크 구조와 문서 내용에 내재된 의미론적 정보(semantic information)를 이용하였다.

BO(Bookmark Organizer)[22]는 개념 정보에 따라 웹 문서들을 조직화하기 위해 계층적 군집화 기술과 이용자와의 상호작용을 함께 이용하였다.

2.1.3 개인화된 웹 에이전트

이러한 유형의 웹 에이전트들은 주로 이용자의 선호도(preference)를 학습한 다음 이것에 기초한 맞춤형 웹 정보를 찾아주거나, 협력적 여과(coll borative filtering) 기술을 이용하여 관심분야가 비슷한 다른 이용자들의 기호에 맞는 웹 정보를 찾아준다. 이러한 웹 에이전트들의 대표적인 예로는 WebWatcher[1], PAINT[25], Syskill & Webert[27], GroupLens[32], Firefly[33] 등이 있다. 예컨대 Syskill & Webert는 이용자 프로파일과 베이지안 분류기(Bayesian classifier)를 이용하여 임의의 웹 문서에 대한 이용자의 관심도를 판정한다.

2.2 데이터베이스 접근법

웹 마이닝을 위한 데이터베이스 접근법들은 주로 웹상의 반-구조화된 데이터(semi-structured data)를 좀더 구조적인 자원들로 조직화하고, 이러한 데이터를 분석하기 위해 표준 데이터베이스 질의 메커니즘(database querying mechanism)과 데이터 마이닝 기술들을 이용하는데 초점을 맞추고 있다.

2.2.1 복수 계층 데이터베이스

복수 계층 데이터베이스(multilevel database)들의 맨 하위 계층은 하이퍼텍스트 문서들과 같이 다양한 웹 저장소에 저장된 반-구조화된 정보들을 포함한다. 반면에, 상위 계층들에서는 하위 계층으로부터 메타 데이터나 보다 일반화된 데이터를 추출하여 관계형 데이터베이스나 객체지향 데이터베이스와 같은 구조화된 데이터로 조직화한다. 예컨대, Han의 연구[35]에서는 각 계층이 하위 계층에 대한 일반화(generalization)와 변형(transoration)을 통해 얻어지는 하나의 복수 계층 데이터베이스를 제안하였다. Kholsa의 연구[14]에서는 각 정보 제공 영역별로 메타 데이터베이스를 생성하고 유지할 것을 제안하였고, 메타 데이터베이스를 위한 하나의 전역 스키마(global schema)를 제안하였다. King과 Novak의 연구[15]에서는 하나의 이질적인 전역 데이터베이스 스키마에 의존하기 보다, 각 정보 자원들로부터 한 부분씩 스키마를 점진적으로 통합해 나가기를 제안하였다. ARANEUS시스템[26]은 하이퍼텍스트 문서들로부터 관련 정보를 추출한 다음, 이들을 통합하여 데이터베이스 뷰(view) 개념을 일반화한 소위 'Derived Web Hypertext'들을 생성하였다.

2.2.2 웹 질의 시스템

많은 웹-기반의 질의 시스템(query system)들은 주로 SQL과 같은 표준 데이터베이스 질의 언어(query language)들과 웹 문서들에 관한 구조 정보, 그리고 웹 검색 질의를 위한 자연어 처리 기술들을 이용하고 있다. W3QL[17]은 하이퍼텍스트 문서의 구조에 기초한 구조 질의들과 정보 검색 기술들에 기초한 내용 질의들을 하나로 통합하였다. WebLog[20]는 웹 정보 자원들로부터 원하는 정보를 추출해내기 위한 논리-기반의 질의 언어이다. Lorel[31]과 UnQL[4]은 하나의 그래프 데이터 모델을 이용하여 웹상의 이질적이고 반-구조화된 정보에 대한 질의를 표현한다.

TSIMMIS[6]은 이질적이고 반-구조화된 정보 자원들로부터 데이터를 추출한 다음 이들을 서로 연계하여 하나의 통합된 데이터베이스 표현을 생성한다.

3. 웹 이용 마이닝

앞서 언급한 바와 같이 웹 이용 마이닝은 웹 서버들로부터 이용자 접근 패턴을 자동으로 발견해내는 것을 말한다. 일반적으로 웹 서버들은 많은 양의 서버 접근 로그(server access log) 데이터를 파일에 보관한다. 이 데이터는 어떤 이용자가 어떤 웹 정보를 이용했는지를 기록하고 있다. 그 밖에 이용 가능한 이용자 정보로는 각 웹 페이지에 대한 참조 페이지 정보를 포함하고 있는 참조 로그(referrer log) 데이터와 CGI 스크립트를 통해 수집된 이용자 등록 정보나 프로필 데이터 등이 있다. 각 기업이나 기관은 그러한 데이터를 분석함으로써 고객의 성향을 파악할 수 있고, 따라서 상품판매를 위한 타겟 마켓팅 전략(target mark eting strategy)을 수립할 수 있으며, 각종 프로모션 캠페인의 효과를 평가할 수도 있다. 또 이용자 접근 패턴 분석 결과에 따라 좀더 정보 접근이 용이하도록 웹 사이트를 재 구성할 수 있으며, 특정 이용자 그룹을 겨냥하는 웹 광고를 가능하게 한다.

기존의 웹 분석 도구들은 서버에서의 각 이용자의 활동에 관한 보고서를 출력해주거나 다양한 형태의 데이터 여과 기능을 제공하는 것들이 대부분이다[10, 13,24]. 그러한 도구를 사용하면, 서버와 각 파일들에 대한 접근 횟수와 방문 시간, 그리고 이용자의 도메인 이름(domain name)과 URL 등을 쉽게 알 수 있다. 그러나 이러한 도구들은 접근된 파일들과 디렉토리들 사이의 데이터 관계를 분석하는 기능은 제공하지 못한다. 최근에 와서는 이용자 접근 패턴의 발견과 분석을 위한 보다 지능적인 방법들이 개발되고 있다.

3.1 전처리 작업

마이닝 알고리즘을 적용하기 앞서 데이터 전처리 과정(preprocessing)을 통해 해결해야 할 일들로는 접근 로그 데이터에 대한 모델 생성, 관련 없는 데이터나 잡음을 삭제하는 데이터 정제와 여과, 개별적인 로그 데이터를 트랜잭션 단위로 묶는 그룹화, 이용자 등록 정보와 같은 다른 데이터와의 통합 등이 있다. 이러한 데이터 전처리 과정 중에서도 첫번째 수행되는 작업이 데이터 정제(data cleaning)이다. 관련 없는 항목들을 지워 하나의 서버 로그를 정제하는 기술은 어떤 유형의 웹 로그 분석을 위해서도 중요하다. 관련 없는 항목의 삭제는 URL 이름의 접미사(suffix)를 검사함으로써 가능하다. 예컨대, 파일 이름의 접미사가 'gif, jpeg, GIF, JPEG, jpg, JPG'인 모든 로그 데이터는 삭제될 수 있다.

로컬 캐쉬(local cache)와 프록시 서버(proxy server)를 사용하면, 이용자 접근이 있어도 접근 로그 파일에 기록되지 않는 경우가 발생하여 정확한 이용자 접근 운행을 짐작하기 어렵다. 이 문제를 해결하기 위한 현재 방법들로는 쿠키(cookies)를 이용하는 방법, 캐쉬를 해제하는 방법, 명시적인 이용자 등록을 이용하는 방법 등이 있다. 하지만 이러한 방법들 모두 문제점을 가지고 있다. 쿠키는 이용자에 의해 삭제될 수 있고, 캐쉬 해제는 캐쉬를 이용할 때 얻을 수 있는 속도상의 잇점을 얻을 수 없고, 이용자 등록은 이용자의 자발적인 협조가 필요하며 간혹 이용자가 잘못된 정보를 입력할 수도 있다. 또 프록시 서버를 사용할 때 발생하는 다른 문제는 이용자 식별(user identifica tion)이 어렵다는 점이다. 단순히 클라이언트 컴퓨터의 이름으로만 이용자를 식별하면 동일한 컴퓨터를 이용하는 서로 다른 이용자들을 동일한 한명의 이용자로 잘못 그룹화하는 오류를 범하기 쉽다. Pirolli의 연구[29]에서는 이용자 식별을 위해 새로이 요청된 페이지가 직전에 방문한 페이지로부터 직접 링크로 연결되어 있는지 검사를 하

여 만약 연결되어 있지 않으면 동일한 컴퓨터에 여러 이용자가 존재한다고 가정하였다. Srivastava의 연구에서는 항해 패턴을 기초로 한 이용자의 세션길이를 판정하고, 이것을 기초로 이용자를 식별하였다.

두 번째로 중요한 전처리 작업은 웹 페이지 참조 레코드들을 논리적 단위인 세션 또는 트랜잭션으로 그룹화하는 트랜잭션 식별(transaction iden tification) 작업이다. 하나의 이용자 세션은 한명의 이용자에 의해 한 서버에 대한 한번의 방문동안 이루어진 모든 페이지 참조들을 말한다. 이용자 세션을 식별하는 일은 각 개별 이용자를 식별하는 일과 유사하다. 하나의 트랜잭션은 트랜잭션을 식별하는 기준에 따라 단 한 페이지의 참조에서부터 한번의 이용자 세션동안 이루어진 모든 페이지 참조들에 이르기까지 트랜잭션의 길이가 달라질 수 있다는 면에서 이용자 세션과는 다르다. 하지만 전통적인 데이터마이닝 응용영역들과는 달리 웹 이용 마이닝에서는 이용자 세션보다 더 작은 트랜잭션들로 나누는 편리한 방법을 찾기가 어렵다.

3.2 접근 패턴 발견

일단 이용자 트랜잭션들을 식별해내고 나면, 분석자의 필요에 따라 경로 분석(path analysis), 연관 규칙(association rule)과 순차 패턴(sequential pattern) 발견, 군집화(clustering)와 분류(class ification)와 같은 다양한 접근 패턴 마이닝 작업이 수행될 수 있다. 경로분석을 위해서는 웹 페이지간의 관계를 나타내는 여러 가지 유형의 그래프를 생성할 수 있다. 그 중에 가장 대표적인 것은 한 웹 사이트의 물리적인 레이아웃을 그대로 나타내는 그래프로서, 각 노드(node)는 웹 페이지를, 각 에지(edge)는 웹 페이지간의 하이퍼링크를 나타내는 것이다. 그 밖에도 웹 페이지의 유형에 따라 유사한 페이지간에 에지를 가지는 그래프, 각 에지가 한 웹 페이지에서 다른 페이지로 옮겨간 이용자들의 수를 나타내는 그래프 등이 많이 이용된다[29]. 경로 분석은 한 사이트 내에서 가장 빈번하게 방문한 경로들을 찾아내는데 이용될 수 있다.

일반적으로 연관 규칙 발견 기술들은 각 트랜잭션이 아이템들의 집합으로 이루어진 트랜잭션 데이터베이스들에 적용된다. 그러한 경우에 문제는 한 트랜잭션에 한 아이템 집합이 등장하면 곧 다른 아이템들의 등장을 암시하는 데이터 아이템들 간의 모든 연관성을 찾아내는 것이다. 웹 이용 마이닝에서 이 문제는 한 서버에서 한 이용자에 의해 이루어진 페이지 참조들간의 상호연관성을 찾는 것에 해당한다. 예컨대, 연관 규칙 발견 기술을 적용하면 "URL이 /company/product1인 웹 페이지를 접근한 이용자의 40%는 /company/product2의 페이지도 접근한다" 와 같은 연관 규칙을 찾아낼 수 있다.

순차 패턴을 찾는 문제는 시간에 따라 순서화 된 트랜잭션들의 집합에서 한 아이템들의 집합이 등장하면 뒤이어 다른 아이템들의 집합이 등장하는 패턴을 찾는 것이다. 일반적으로 웹 서버 로그 트랜잭션들에는 방문시간도 함께 기록된다. 이러한 데이터를 분석함으로써 웹 마이닝시스템은 예컨대 "/company/product1에서 온라인 주문을 한 고객 중 60%가 15일 이내에 /company/product4에서 역시 온라인 주문을 한다"와 같은 데이터 아이템들간의 시간적 관계를 알아낼 수 있다. 데이터 아이템들의 시간적 특성을 고려하여 찾아낼 수 있는 또 다른 데이터 패턴으로는 유사 시간 순차(similar time sequence)가 있다. 예컨대, 특정 시간대 $[t_1, t_2]$동안 특정 웹 문서를 방문한 모든 이용자들의 공통된 특성을 찾아내거나, 역으로 특정 웹 문서를 가장 많이 접근한 시간대를 찾아내는 등이 이러한 패턴 발견에 속한다.

분류 규칙(classification rule)들을 찾는 일도 웹 이용 마이닝의 중요한 패턴 발견작업이다. 이것은 데이터 아이템들의 공통적인 속성에 따라 특정 그룹에 속하는 데이터 아이템들의 프로파일을 생성하는 일이다. 이러한 프로파일은 나중에 데이터베이스에 추가되는 새로운 데이터 아이템을 분류하는데 이용될 수 있다. 웹 이용 마이닝에서는 웹 이용자들의 신상정보나 접근 패턴에 기초하여 특정 웹 서버에 접근하는 이용자들의 프로파일을 생성하는데 분류 기술들을 적용한다. 예컨대, 웹 서버 접근 로그에 분류 기술을 적용하면 "정부기관이나 관공서에 근무하는 이용자들은 주로 /company/product1의 페이지에 관심을 가지는 경향이 있다"와 같은 관계를 발견해낼 수 있다. 군집화(clustering)는 유사한 특성을 가진 이용자들이나 데이터 아이템들끼리 그룹으로 묶는 일을 말한다. 이용자 정보나 웹 트랜잭션 로그상의 데이터 아이템들에 대한 군집화는

향후 온라인 및 오프라인 영업 전략을 개발하고 실현하는데 큰 도움을 줄 수 있다.

3.3 접근 패턴 분석

패턴 발견 도구를 이용하여 일단 접근 패턴들이 발견되면, 이번에는 이 패턴들을 이해하고, 시각화하고, 해석하기 위한 적절한 분석도구가 필요하다. 시각화(visualization)는 사람들로 하여금 다양한 현상을 쉽게 이해하도록 하는데 큰 도움을 준다. 따라서 웹 이용자들의 행위를 이해하는데도 이러한 시각화 기술은 반드시 필요하다. Pitkow 등은 웹 접근 패턴들을 시각화하기 위한 WebViz시스템[30]을 개발하였다. 이 시스템은 웹 사이트를 구성하는 웹 페이지들 중에서 관련없는 부분에 대한 여과 기능을 제공함으로써 선택적인 분석이 가능하다. 그리고 이 시스템에서는 하나의 웹 사이트를 사이클을 포함하는 하나의 유향 그래프로 시각화하여 보여준다.

온라인 분석처리(On-Line Analytical Processing), 즉 OLAP는 기업환경에서 전략적 데이터 분석을 위한 효과적인 방법이다. Dyreson의 연구[9]에서는 서버 접근 로그 데이터로부터 통계적 분석을 쉽게 하기 위한 목적으로 OLAP의 기술을 적용하여 데이터 큐브(data cube)를 구성하였다. 서버의 접근 로그 데이터의 양은 매우 급속히 증가하므로, 그 모든 데이터를 온라인 방식으로 분석하기 불가능한 경우도 있다. 따라서 온라인 분석이 가능하도록 다양한 방법으로 데이터를 요약하는 기술이 필요하다.

오늘날 관계형 데이터베이스(relational database) 기술이 성공하게 된 이유 중에 하나는 선언적(declarative), 비-절차적(non-procedural), 고급 질의 언어(high-level query language)가 있었기 때문이다. 이러한 질의 언어는 데이터가 만족해야 할 조건만을 표현할 뿐, 요구되는 데이터를 어떤 방식으로 구할 지에 대해서는 아무것도 나타내지 않는 특징을 가지고 있다. 발견될 패턴의 수가 많을 경우, 분석의 초점을 분명히 명시할 수 있는 기능이 필요하다. 이러한 기능은 두 가지 방식으로 제공될 수 있다. 첫째는 마이닝의 대상이 되는 데이터베이스의 범위를 제한하기 위한 제약조건(constraint)을 명시하는 방식이다. 둘째는 마이닝 프로세스를 통해 지식을 추출한 다음, 이 추출된 지식들에 대해 질의를 하는 방식이다. 이 경우에는 데이터가 아니라 지식에 대해 질의할 수 있는 질의언어가 필요하다. WEBMINER 시스템에서는 발견된 지식에 대한 질의 기능을 제공하기 위해 SQL과 같은 질의 메커니즘을 제안하였다. 그림 2는 1997년 1월1일 이후 ".edu" 도메인에서 최소 지지도(support)가 1%, 최소 신뢰도가(minimum confidence)가 90%인, URL A로부터 시작해서 B, 그리고 C를 포함하는 모든 연관 규칙들을 찾아달라는 WEBMINER시스템의 한 질의 예를 나타낸다.

```
SELECT association-rules (A*B*C)
FROM log.data
WHERE date>=970101 AND domain="edu"
  AND support=1.0 AND confidence=90.0
```

(그림 2) 연관 규칙 탐색을 위한 질의 예

4. 적응형 웹 사이트를 위한 웹 이용 마이닝

본 논문에서는 웹 이용 마이닝의 한 응용으로서, 웹 서버 접근 로그를 분석하여 적응형 웹 사이트(adaptive web site)를 구현하는 방법을 소개한다. 일반적으로 적응형 웹 사이트는 이용자의 접근 패턴을 학습하여 스스로 구조나 외형을 자동적으로 개선시켜 나가는 웹 사이트를 의미한다. 초기의 웹 사이트는 각 웹 문서들이 지닌 의미와 문서들 간의 상호관계 등을 고려해 최상의 사이트를 구현하고자 하는 웹 마스터의 의도가 반영된 것이다. 그러나 동적인 이용자들의 요구를 반영하여 보유 정보를 효과적으로 제공하기 위해서는 웹 마이닝 과정을 통한 지속적인 웹 사이트의 변경과 갱신 작업이 요구된다. 특히, 이용자들의 일반적인 정보 접근 패턴을 알아낼 수 있는 중요한 자료가 되는 웹 서버 로그 데이터를 마이닝함으로써 웹 사이트의 구조나 표현 방식을 개선시킬 수 있다.

전체적인 과정을 개략적으로 살펴보자, 먼저 웹 전체적인 과정을 개략적으로 살펴보면, 웹 서버 로그 데이터와 웹 문서의 하이퍼링크 구조 정보를 적용하여 최후전진 문서(last forward document)만을 갖는 데이터 시퀀스를 구성한다. 데이터 시퀀스를 대상으로 새로운 순차 접근 패턴 알고리즘인 TPA(Traversal Path Analysis)를 적용함으로써 연관성은 높으나 접근 경로가 긴 문서들의 시퀀스를 생성한다. 이러한 빈발 시퀀스들을 구성

```
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 3) 웹 서버 로그 데이터

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

⇒

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

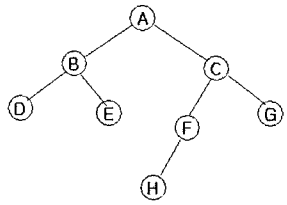
(그림 4) 데이터 시퀀스 생성

요소로 하는 색인 페이지(index page)들을 자동 생성하여 기존 웹 문서들에 추가한다. 이와 같이 자주 이용되는 웹 문서들에 대한 색인 페이지의 자동 생성과 추가를 통해 다수의 이용자들에게 원하는 웹 문서를 빠르게 접근 할 수 있는 서비스를 제공할 수 있다.

4.1 데이터 시퀀스 생성

그림 3과 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 위한 데이터 시퀀스를 생성하는 과정은 다음과 같다. 로그 데이터를 대상으로 어구분석과정과 이용자의 IP 주소, 이용자의 ID, 웹 문서에 접근한 날짜와 시간, 접근한 문서의 URL들을 제외한 항목들을 제거하는 정화과정을 거친다. 적용 가능한 접근 시간 내에 한 이용자 세션동안 거쳐간 웹 문서 운행 경로를 알아낸다. 즉, 그림 4의 왼쪽 표에서 보여주는 바와 같이 다수 이용자가 접근한 웹 문서들의 집합들을 구한다.

전 단계에서 구한 각각의 접근 웹 문서들을 대상으로



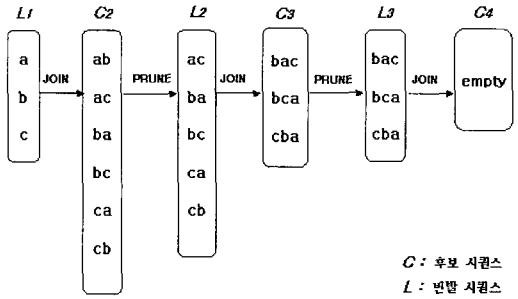
(그림 5)하이퍼링크 구조

로 전문서에서 바로 이어지는 후문서가 그림 5와 같은 하이퍼링크 구조상 동일 가지 아래에 존재하지 않는 조건을 만족하는 전문서들만을 추출하여 데이터 시퀀스가 될 최후 전진 문서(last forward document) 집합을 구한다. 이러한 최후 전진 문서 집합이 갖는 의미는 이용자가 순차적으로 접근한 문서들을 대상으로 하이퍼링크 구조의 깊이로 보면, 이용자가 원하는 정보를 담은 문서일 가능성이 가장 높은데 반해서 하이퍼링크 구조의 가지별로 나누어 보면, 두 문서간 접근 경로가 가장 멀어서 접근 경로를 단축 시켜줄 필요성을 갖는다는데 있다. 즉, 그림 4의 왼쪽 표의 집합들을 가지고 최후 전진 문서들만을 선택하면 오른쪽 표에 나타나는 형태의 데이터 시퀀스들이 생성되는데, 생성 과정은 표 1 LFD 알고리즘에서 보여주고 있다.

(표 1) LFD 알고리즘

```
Algorithm LFD
/*
· s : source page
· d : destination page
· t : traversal path
· k : the number of traversal paths
· a : array of ordered pairs (referring page, referred page)
· DF : database to store all the resulting last forward
       documents
*/
For (i=1; i<=k; i++) do
  Begin
  Express tk as {(s1, d1)...(sn, dn)};
  For (j=1; j <=m; j++) do
    Begin
    If Both sn and dn do not exist in aj then
        Append sn to DF;

     Else If dn is not placed back of sn
          Append sn to DF;
    End
  End
```



(그림 6) TPA 결합 알고리즘 형태

4.2 빈발 시퀀스 탐색

순차 접근 패턴 탐색 알고리즘인 TPA는 색인 페이지의 생성 목적에 적합한 AprioiriAll 알고리즘의 변형이다. AprioiriAll 알고리즘과 구별되는 특징으로 TPA 알고리즘에서는 그림 6과 같이 동일 문서들의 조합을 기본적으로 고려하지 않으며 데이터베이스의 스캔 과정도 매 단계마다 수행되지 않는다.

표 2는 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 1-시퀀스의 집합 C_1을 생성하고 C_1에서 최소 지지도를 만족하는 빈발 1-시퀀스의 집합 L_1을 구한다. 다음 후보 2-시퀀스의 집합 C_2를 생성하기 위하여 순서는 고려하지만 동일 문서 조합을 고려하지 않는다는 조건 아래 L_1*L_1을 구하고 C_2에서 최소 지지도를 만족하는 빈발 2-시퀀스의 집합 L_2를 구한다. 다음 패스에서, 후보 3-시퀀스의 집합 C_3를 생성하기 L_2를 기반으로 순서는 고려하지만 동일 문서 조합을 고려하지 않는다는 조건 아래 등장 가능성이 있는 C_3를 구한다. 물론, 이 과정에서 후보 시퀀스를 대상으로 L_2안에 존재하지 않는 서브시퀀스(subsequence)를 갖는 후보 시퀀스의 집합을 전정(pruning)하는 과정이 포함된다. 세 번째 패스에서 데이터베이스를 스캔 한 후, C_3에서 최소 지지도를 만족하는 빈발 3-시퀀스의 집합 L_3를 구한다. 이러한 방식으로 여러 패스를 거치면서 후보 시퀀스 집합 혹은 빈발 시퀀스 집합이 생기지 않을 때까지 반복 수행한다. 이와 같은 방식으로 모든 패스의 후보

(표 2) TPA 알고리즘

```
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 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 Li, i ( k ;
      Foreach  data-sequence c in the database DT do
         Increment the count for all candidates
            in Ck that are contained in c
      Lk=Candidates in with minimum support.
      End

      Else  /* Lk already known */
         Delete all sequences in Lk contained in
            some Li, i ( k ;
      Answer =Uk Lk ;

   Function  next(k : integer)
   Begin
      If (hitk >0.66) Return k + 2 /* hitk = |Lk|/|Ck| */
                        End
```

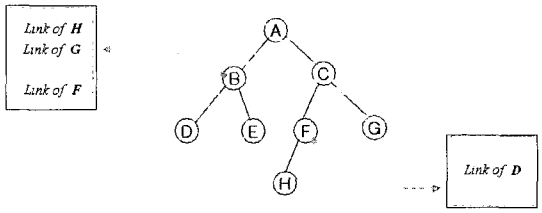
시퀀스와 빈발 시퀀스들을 구하는데, 전 패스의 빈발 시퀀스와 후보 시퀀스의 비가 0.66이상이 되는 경우가 발생하면 현 패스에서의 빈발 시퀀스를 구하지 않고 다음 패스로 바로 스킵(skip)하게 된다. 역진 단계에서는 전진 단계에서 스킵한 패스들을 대상으로 후 패스의

L_1	sup
B	7
D	4
F	6
G	3
H	4

L_2	sup
BF	4
FD	3

L_3	sup
BHG	3

(그림 7) 빈발 시퀀스



(그림 8) 생성된 색인 페이지의 구조

(표 3) IPG 알고리즘

```
Algorithm IPG:

For (i=1; i <n; i++) do
  If C i is not null then Begin
     Create Wc at C i1;
     Call Elements(C i);
     For (j= i +1; j =<n; j++) do
        If C i1=C j1 then Begin
             Draw dashline in Wc;
             Call Elements(C j);
             Set null to Cj;
        1.          End
     Set null to C i;
     End

Function Elements(C c)
   For (k=2; k=<m; k++) do
   Begin
      Insert C ck in Wc;
End
```

후보 시퀀스 혹은 빈발 시퀀스 내에 존재하는 서브 시퀀스를 제외시킨 빈발 시퀀스들을 구해 나간다. 예를 들어, 그림 4에서 생성한 데이터 시퀀스를 대상으로 지지도(support)가 3인 TPA알고리즘을 적용하면 그림 7과 같은 각 단계의 빈발 시퀀스들이 구해지고 이들은 색인 페이지의 구성요소가 된다.

4.3 색인 페이지 생성

색인 페이지는 기존 웹 상에서 하이퍼 링크로 직접 연결되어 있지는 않지만, 이용자 접근 패턴을 분석한 결과를 토대로 관련성이 있다고 추정되는 페이지들을 그 구성요소로 한다. 표 3 IPG 알고리즘에서 보여주는 바와 같이, 색인 페이지의 생성 구조는 앞 과정에서 구한 최대 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 8은 생성된 색인 페이지의 구조를 보여 주고 있다.

5. 결 론

웹 마이닝이란 용어는 그동안 매우 포괄적인 의미로 사용되어 왔다. 본 논문에서는 분석 대상과 도출하려는 지식의 유형에 따라 웹 마이닝의 개념을 웹 내용 마이닝과 웹 이용 마이닝으로 나누어 정의하였다. 그리고 이 각 분야에 연관된 연구주제와 대표적인 기술들을 간략히 정리하여 보았다. 끝으로 웹 이용 마이닝의 한 응용으로서, 웹 서버 접근 로그를 분석하여 적응형 웹 사이트(adaptive web site)를 구현하는 방법을 소개하였다. 동적으로 변화하는 이용자들의 요구를 반영하여 보유 정보를 효과적으로 제공하기 위해서는 이용자들의 정보 접근 패턴을 알아낼 수 있는 웹 이용 마이닝을 통한 자동화된 웹 사이트의 변경과 갱신 작업이 효과적이다.

참 고 문 헌

[1] R. Armstrong, D. Freitag, T. Joachims, and T. Mitchell, "Webwatcher : A Learning Apprentice for the World Wide Web", Proceedings of AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments, 1995.

[2] A.Z. Broder, S.C. Glassman, M.S. Manasse, and G. Zweig, "Syntactic Clustering of the Web", Proceedings of 6th International World Wide Web Conference,

1997.

[3] C.M. Brown, B.B. Danzig, D. Hardy, U. Manber, and M.F. Schwartz, "The Harvest Information Discovery and Access System", Proceedings of the 2nd International World Wide Web Conference, 1994.

[4] P. Buneman, S. Davidson, G. Hillebrand, and D. Suciu, "A Query Language and Optimization Techniques for Unstructured Data", Proceedings of 1996 ACM-SIGMOD International Conference on Management of Data, 1996.

[5] C. Chang and C. Hsu, "Customizable Multi-Engine Search Tool with Clustering", Proceedings of the 6th International World Wide Web Conference, 1997.

[6] S. Chawathe, H.G. Molina, J. Hammer, K. Irland, Y. Papakonstantinou, J. Ulman, and J. Widom, "The TSIMMIS Project: Integration of Heterogenous Information Sources", Proceedings of IPSJ Conference, 1994.

[7] M.S. Chen, J.S. Park, and P.S. Yu, "Data Mining for Path Traversal Patterns in a Web Environment", Proceedings of the 16th International Conference on Distributed Computing Systems, pp.385-392, 1996.

[8] R.B. Doorenbos, O. Etzioni, and D.S. Weld, "A Scalable Comparison Shopping Agent for the World Wide Web", Technical Report 96-01-03, University of Washington, Dept. of Computer Science and Engineering, 1996.

[9] C. Dyreson, "Using an Incomplete Data Cube as a Summary Data Sieve", Bulletin of the IEEE Technical Committee on Data Engineering, pp.19-26, 1997.

[10] Software Inc. Webtrends, http://www.webtrends.com, 1995.

[11] W.B. Frakes and R. Baeza-Yates, Information Retrieval Data Structures and Algorithms, Prentice Hall, Englewood Cliffs, NJ, 1992.

[12] K. Hammond, R. Burke, C. Martin, and S. Lytinen, "FAQ-Finder: A Case-Based Approach to Knowledge Navigation", Working Notes of the AAAI Spring Symposium: Information Gathering from Heterogenous, Distributed Environments, AAAI Press, 1995.

[13] Open Market Inc., Open Market Web Reporter, http://www.openmarket.com, 1996.

[14] I. Khosla, B. Kuhn, and N. Soparkar, "Database Search Using Information Mining", Proceedings of 1996 ACM-SIGMOD International Conference on Management of Data, 1996.

[15] R. King and M. Novak, "Supporting Information Infrastructure for Distributed, Heterogenous Knowledge Discovery", Proceedings of the SIGMOD-96 Workshop on Research Issues on Data Mining and Knowledge Discovery, Canada, 1996.

[16] T. Kirk, A.Y. Levy, Y. Sagiv, and D. Srivastava, "The Information Manifold", Working Notes of the AAAI Spring Symposium: Information Gathering from Heterogenous, Distributed Environments, AAAI Press, 1995.

[17] D. Konopnicki and O. Shmueli, "W3QS: A Query System for the World Wide Web", Proceedings of the 21th VLDB Conference, pp.54-65, 1995.

[18] M. Koster, "Aliweb-Archie-like Indexing in the Web", Proceedings of the 1st International Conference on the World Wide Web, pp.91-100, 1994.

[19] C. Kwok and D. Weld, "Planning to gather Information", Proceedings of the 14th National Conference on AI, 1996.

[20] L. Lakshmanan, F. Sadri, and I.N. Subramanian, "A Declarative Language for Querying and Restructuring the Web", Proceedings of the 6th International Workshop on Research Issues in Data Engineering: Interoperability of Nontraditional Database Systems (RIDE-NDS'96), 1996.

[21] H. Vernon Leighton and J. Srivastava, "Precision among WWW Search Services (Search Engines): Alta Vista, Excite, Hotbot, Infoseek, Lycos, http://www.winona.msus.edu/is-f/library-f/webind2/webind2.htm, 1997.

[22] Y.S. Maarek and I.Z. Ben Shaul, "Automatically Organizing Bookmarks per Content", Proceedings of the 5th International World Wide Web Conference, 1996.

[23] B. Mobasher, N. Jain, E. Han, and J. Srivastava, "Web Mining: Pattern Discovery from World Wide Web Transactions", Technical Report TR 96-050, University

of Minnesota, Dept. of Computer Science, Minneapolis, 1996.

[24] net.Genesis, net.analysis desktop, http://www.netgen.com, 1996.

[25] K.A. Oostendorp, W.F. Punch, and R.W. Wiggins, "A Tool for Individualizing the Web", Proceedings of the 2nd International World Wide Web Conference, 1994.

[26] P. Merialdo, P. Atzeni, G. Mecca, "Semistructured and structured Data in the Web: Going Back and Forth", Proceedings of the Workshop on the Management of Semistructured Data, 1997.

[27] M. Pazzani, J. Muramatsu, and D. Billsus, "Syskill & Webert: Identifying Interesting Web Sites", Proceedings of AAAI Spring Symposium on Machine Learning in Information Access, 1996.

[28] M. Perkowitz and O. Etzioni, "Category Translation: Learning to Understand Information on the Internet", Proceedings of the 15th IJCAI, pp.930-936, 1995.

[29] P. Pirolli, J. Pitkow, and R. Rao, "Silk from a Sow's Ear: Extracting Usable Strctures from the Web", Proceedings of 1996 Conference on Human Factors in Computing Systems(CHI-96), 1996.

[30] J. Pitkow and K.K. Bharat, "WebViz: A Tool for World-Wide Web Access Log Analysis", Proceedings of the First International WWW Conference, 1994.

[31] D. Quass, A. Rajaraman, Y. Sagiv, J. Ullman, and J. Widom, "Querying Semistructured Heterogenous Information", Proceedings of the International Conference on Deductive and Object Oriented Databases, 1995.

[32] P. Resnik, N. Iacovou, M. Sushak, P. Bergstrom, and J. Riedl, "Grouplens: An Open Architecture for Collaborative Filtering of Netnews", Proceedings of the 1994 CSCW Conference, pp.432-443, 1995.

[33] U. Shardanand and P. Maes, "Social Information Filtering: Algorithms for Automating Word of Mouth", Proceedings of 1995 Conference on Human Factors in Computing Systems (CHI-95), pp.210-217, 1995.

[34] R. Weiss, B. Velez, M.A. Sheldon, C. Namprempre, P. Szilagyi, A. Duda, and D.K. Gifford, "Hypursuit: a Hierarchical Network Search Engine That Exploits Content-Link Hypertext Clustering", Proceedings of the 7th ACM Conference on Hypertext, 1996.

[35] O.R. Zaiane and J. Han, "Resource and Knowledge Discovery in Global Infromation Systems: A Preliminary Design and Experiment", Proceedings of the First International Conference on Knowledge Discovery and Data Mining, pp.331-336,

◐ 저 자 소 개 ◑

김 인 철

1987년 서울대학교 대학원 전산과학과(이학석사)
1995년 서울대학교 대학원 전산과학과(이학박사)
1989년~1995년 경남대학교 전산통계학과 조교수
1996년~현재 : 경기대학교 정보과학부 전자계산학전공 부교수
관심분야 : 지능형 에이전트, 분산인공지능, 테이터마이닝