익명 웹로그 탐사에 기반한 동적 링크 추천

윤 선 희[†] · 오 해 석[††]

요 약

웹 공간(Webspace)에서 사용자의 순회패턴을 포착하는 것을 '순회패턴 탐사(mining traversal patterns)'라 한다. 순회패턴 탐사에서는 사용자가 원하는 정보를 탐색하기 위해 정보 제공 서비스에 따라 이동하기 때문에 객체(예 : URL)의 내용보다는 위치 때문에 방문될 수도 있는 독특한 특징을 가진다. 따라서 순회패턴 데이터로부터 의미있는 정보를 추출하는 작업의 복잡도를 크게 증가시킨다. 그러나 이러한 정보 제공 서비스의 질을 개선하기 위한 요구가 증가하고 있기 때문에 데이터 탐사 분야에서 순회패턴 탐사 문제는 최근 중요한 문제로 대두되고 있다. 본 논문에서는 빈발 순회패턴을 탐사하여 웹 사이트 상에서 추천을 수행하는 동적 링크 추천(Dynamic Link Recommendation ; DLR) 알고리즘을 제안한다. 제안한 DLR 알고리즘은 방대한 자료를 포함하고 있는 대부분의 웹 사이트에 효과적으로 적용될 수 있다. 두 개의 실제 웹 사이트에 적용한 실험 결과는 제안한 방법의 성능이 우수함을 보여준다.

Dynamic Link Recommendation Based on Anonymous Weblog Mining

Sun-Hee Yoon[†] · Hae-Seok Oh[††]

ABSTRACT

In Webspace, mining traversal patterns is to understand user's path traversal patterns. On this mining, it has a unique characteristic which objects (for example, URLs) may be visited due to their positions rather than contents, because users move to other objects according to providing information services. As a consequence, it becomes very complex to extract meaningful information from these data. Recently discovering traversal patterns has been an important problem in data mining because there has been an increasing amount of research activity on various aspects of improving the quality of information services. This paper presents a Dynamic Link Recommendation (DLR) algorithm that recommends link sets on a Web site through mining frequent traversal patterns. It can be employed to any Web site with massive amounts of data. Our experimentation with two real Weblog data clearly validate that our method outperforms traditional method.

키워드 : 웹로그 탐사(Weblog Mining), 클러스터링(clustering), 동적 링크 추천(Dynamic Link Recommendation)

1. 서 론

웹 환경에서 분산된 정보를 접근하는 순회 패턴(traversal patterns)을 이해하는 것은 지식발견과 웹 탐사 분야에서 중요한 문제이다[1, 2]. 사용자는 관심이 있는 정보를 찾을 때, 사이트에서 제공되는 하이퍼링크(hyperlinks)와 같은 것을 통해서 한 웹 객체(예 : URL)에서 다른 객체로 이동한다. 구체적으로 온라인상에서 쇼핑하는 예를 들어보자. 사용자가 온라인으로 백화점에서 옷, 전자제품, 스포츠 용품 등을 구매하고 싶다고 하자. 이들 품목들은 각각 해당 부서에서 판매를 담당할 것이고, 이들 부서의 URL들이 직접 한 곳에 연결되어 있지 않다면 구매자가 원하는 품목들을 찾기 위해서는 여러 개의 중간 URL들을 거쳐야 할 것이다. 이때 이와 같은 구매 패턴을 가지는 사용자들을 하나의 클러스터(cluster)로 구분할 수 있다. 또한 유사한 구매 패턴을 가지는 다양한 클러스터들로 사용자를 분류할 수도 있다. 여기서 웹 사이트 설계자가 각 사용자 클러스터에 대한 접근 패턴을 알아낼 수 있는 어떤 방법이 있다면 사용자들이 그러한 URL들을 쉽게 접근할 수 있도록 함께 묶어서 제공할 수도 있다. 예를 들어, 전자제품 URL을 스포츠 용품 URL과 링크로 연결해 놓을 수 있다.

이런 환경에서 사용자의 접근 패턴을 포착하는 것을 '순회 패턴 탐사(mining traversal patterns)'라 한다[1]. 순회 패턴 탐사 문제에서는 사용자가 원하는 정보를 탐색하기 위해 정보 제공 서비스에 따라 이동하기 때문에 어떤 객체는 객체의 내용보다는 위치 때문에 방문되기도 한다. 순회 패턴 탐사 문제의 이런 독특한 특징은 순회 데이터로부터 의미있는 정보를 추출하는 작업의 복잡도를 크게 증가시킨다. 그러나 이러한 정보 제공 서비스는 최근 대중화되어 가고 있고, 사용자 행위 특성을 포착하고 그 서비스의 질을 개선하기 위한 요구가 증가하고 있기 때문에 순회 패턴 탐

† 정 회 원 : 미림 전산고등학교 교사
†† 종신회원 : 숭실대학교 컴퓨터학과 교수
논문접수 : 2002년 7월 24일, 심사완료 : 2003년 7월 14일

사 문제는 최근 중요한 문제로 대두되고 있다.

본 논문에서는 효율적인 웹 정보 서비스를 제공하기 위해 먼저 웹로그로부터 빈발 순회패턴들을 탐사하고, 이를 기반으로 추천 링크집합을 생성함으로써 웹 사이트 상에서 추천을 수행하는 동적 링크 추천(Dynamic Link Recommendation : DLR) 알고리즘을 제안한다. 제안한 DLR 알고리즘은 방대한 자료를 포함하고 있는 대부분의 웹 사이트에 효과적으로 적용될 수 있다. 먼저 웹로그에 기초하여 클러스터링(clustering) 기법을 적용하면 사용자 클러스터에 대한 접근 패턴들을 포착할 수 있다[3]. 이러한 클러스터링 기법들은 웹 사이트 설계자가 미리 고려하지 못한 클러스터들을 찾을 수 있다. 다음은 각 클러스터의 중심 벡터로부터 최소 지지도(s_{min})에 따라 빈발 링크집합을 얻을 수 있다. 이와 같이 생성된 빈발 링크집합은 정적인 하이퍼텍스트(hypertext) 구조의 시스템 설계를 개선시킨다거나(즉, 크게 연관된 객체들 사이에는 효율적인 접근을 제공하고 그 URL에 더 좋은 저작 설계를 제공)[4], 더 좋은 마케팅 전략을 이끌어 내는데(즉, 적절한 위치에 광고함으로써 더 좋은 소비자/사용자의 분류와 행위 특성 분석을 제공) 적용될 수 있다[5]. 궁극적으로 제안한 알고리즘의 목적은 획득한 빈발 순회패턴들을 적용하여 사이트를 더 좋은 형태의 새로운 페이지뷰(pageview)로 제공하는 것이다. 사이트에 접근하는 모든 사용자가 이러한 변형된 형태의 새로운 페이지뷰를 제공받을 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 순회패턴 탐사에 관한 관련연구들을 기술한다. 3장에서는 효율적인 웹 정보 서비스 제공을 위해 빈발 링크집합을 탐사하고 추천 링크집합을 생성하는 DLR 알고리즘에 대해서 구체적으로 설명한다. 4장에서는 두 개의 실제 웹 사이트에 대한 웹로그 데이터를 이용하여 제안한 알고리즘을 비교 분석한다. 마지막 5장에서는 결론 및 향후 연구과제에 대해 기술한다.

2. 관련 연구

익명 웹 탐사 기술은 전자상거래를 위한 개인화된 마케팅이나 Amazon.com과 같은 전자상거래 사이트 등의 웹 기반 응용분야에서 크게 요구된다. 최근, 웹로그(특히, 순회 패턴 정보)에 기초하여 이러한 익명 웹 탐사를 수행하기 위한 많은 연구들이 제시되었다[6-12]. Letizia[6]는 사용자가 관심있게 보았던 URL들을 기록하여 놓았다가 관심을 가질 수 있는 인접한 URL들을 미리 찾아서 그 사용자에게 제시해 주는 클라이언트 측의 에이전트이다. [7]에서 Yan 등(1996)은 자동 고객화 접근방법(automatic customization approach)을 이용한다. 이 시스템은 사용자들의 카테고리(categories)를 구분하기 위하여 웹 서버 로그 상에서 유사한 접근 패턴을 가지는 사용자들의 클러스터를 생성한다. 여기서 유사한 페이지들을 방문하는 사이트의 방문자들은 유사한 관심을 가진 것으로 가정한다. 이처럼 생성된 카테고리는 사이트를 방문한 새로운 사용자들을 고객화하기 위하여 사용자들을 분류하는데 이용된다. 예를 들어, 카테고리의 다른 사용자들이 자주 방문하는 링크들을 하이라이트(highlight) 시키는 것을 들 수 있다. Yan 등[7]은 세션을 벡터(vector)로 표현한다. 따라서 웹 탐사에서 중요한 정보인 세션에서의 페이지들이 접근되는 순서를 반영하지 못한다. 또한 이 방법은 프락시나 웹 브라우저에 의한 페이지들의 캐시(cache) 기능에 의한 효과를 무시한다. 일반적으로 이 문제는 사용자의 순회패턴 탐사의 정확도에 영향을 미친다. SiteHelper[8]는 각 사용자의 페이지 접근 정보에 의하여 사용자의 선호도를 학습한다. 사용자가 일정 시간 이상으로 소요한 페이지들로부터 키워드 리스트를 생성하여 그 사용자에게 제시한다. 키워드 리스트에 관한 피드백에 근거하여 사이트내의 다른 페이지들에 대한 추천 페이지들을 생성한다. WebWatcher[9]는 사용자가 웹을 순회하는 페이지들을 따라가면서 그 사용자에게 잠정적으로 관심있는 링크들을 발췌한다. 시스템은 사용자의 관심있는 페이지들에 대한 간략한 설명으로 시작한다. 각 페이지 요청은 여러 개의 웹 사이트에 분산된 세션을 쉽게 추적하여 관심있는 링크들을 표시하기 위해 WebWatcher 프록시 서버에 라우트된다. WebWatcher는 특정 사용자의 브라우징과 더불어 유사한 관심을 가지는 다른 사용자들의 브라우징에 기초하여 새로운 페이지들에 대한 관심도를 계산한다. Zarkesh 등[10]은 먼저 유사한 순회패스를 가지는 사용자들에 대한 프로파일을 생성하는 통계적인 방법을 제안한다. 다음 단계는 추천 항목들이 일종의 학습 알고리즘을 통하여 이들 프로파일에 사상된다. 마지막으로 새로운 사용자가 들어오게 되면 여러 프로파일들과 비교되어 가장 가까운 프로파일에 근거한 추천항목들을 추천하게 된다. WebPersonalizer[11, 12]는 웹 사이트의 서버 로그로부터 발견한 페이지들의 클러스터들에 기초하여 페이지를 추천한다. 시스템은 현재의 세션과 가장 잘 부합되는 클러스터로부터 페이지들을 추천한다.

3. 동적 링크 추천(DLR) 알고리즘

3.1 개 요

이 장에서는 웹로그 데이터를 이용하여 신뢰할 수 있는 빈발 링크집합을 탐사하는 효율적인 모델을 제안하고, 이를 웹 사이트에 적용하여 웹 사용자에게 신뢰할 수 있는 정보 서비스를 제공하는 동적 링크 추천(Dynamic Link Recommendation : DLR) 알고리즘을 제안한다. 일반적으로 링크 추천을 통하여 웹 정보 서비스를 개선하기 위한 알고리즘에서 가장 핵심적으로 고려해야 할 사항은 추천 링크집합의 내용(즉, URL)이다. 이것은 세션(session) 특징 표현 방법에 따라서 다양한 결과를 얻을 수 있다. 여기서 세션이란 W3C의 WCA[1]에서 정의한 용어로 사용자가 웹 사이트에

1) http://www.w3.org/WCA.

접근하여 한번에 순회한 웹 페이지들의 순차 리스트이다.

추천 링크집합 탐사를 위한 두 개의 DLR 알고리즘들은 먼저 순회패턴들로부터 빈발 링크집합을 생성한다. 각 알고리즘은 빈발 링크집합을 생성함에 있어서 세션에서의 페이지들간의 연관성을 고려하는 정도에서 차이가 있다. 첫 번째 알고리즘인 DLR(V)는 (여기서 "V"는 Vector를 의미한다) Yan 등[7]의 벡터모델을 수정 적용한 것으로 세션에서의 연관성을 고려하는 정도가 약하다. 이 알고리즘은 단순히 세션 데이터베이스 D에서 한 페이지의 빈발 횟수만을 고려한다. 따라서 빈발 1-링크들의 집합을 생성한다. 두 번째 알고리즘인 DLR(M)은(여기서 "M"은 Matrix를 의미한다) 세션에서 인접한 페이지들간의 연관성을 고려한다. 이 알고리즘은 빈발 2-페이지들의 집합을 생성한다. 두 개의 알고리즘에 의해 생성된 빈발 링크집합의 종속관계는 세션에서 페이지들간의 연관성을 얼마나 고려했느냐에 달려있다.

DLR 알고리즘들은 다음과 같은 세부 단계들로 이루어져 있다.

① 전처리 : 웹로그에 대하여 데이터 클리닝 등의 전처리를 수행하고, IP 주소, 시간 등을 이용하여 세션 데이터베이스 D를 생성한다.
② 빈발 링크집합 탐사 : 빈발 링크집합을 생성한다.
③ 추천 링크집합 생성 : 결속력있는 개념을 표현하는 링크집합을 생성한다.
④ 추천 적용 : 웹 사이트를 수정하여 직접 동적 추천을 수행하거나 생성된 추천 링크집합을 웹 마스터에게 제안한다.

다음 절에서 두 개의 알고리즘에서 모두 요구하는 과정인 전처리(preprocessing) 과정을 설명하고, 이어서 두 개의 DLR 알고리즘들의 각각에 대하여 빈발 링크집합을 생성하는 과정과 빈발 링크집합을 이용하여 웹 사이트에서 추천하는 과정을 상세하게 설명한다.

3.2 전처리

사용자 세션은 한 사용자에 의해 접근된 순서가 있는 페이지들의 집합을 의미한다. 그러나 웹로그 데이터는 웹 서버에 기록된 일련의 페이지 뷰(page views) 혹은 요청들(requests)로 이루어져 있다. 일반적으로 각 요청은 요청 시간, 요청된 URL, 그 요청이 발생한 곳의 IP 주소 등을 포함하고 있다. 또한 원시 웹로그 데이터에는 순회패턴 탐사 목적에는 불필요한 많은 웹로그 항목(Weblog entry)들이 존재한다. 이러한 웹로그 항목들은 HTTP 프로토콜의 정의에 의해 사용자가 명시적으로 요구하지 않았음에도 불구하고 웹로그 항목으로 기록된 것이기 때문에 사용자의 순회패턴을 탐사하는 데에는 도움이 되지 않는다. 따라서 이러한 불필요한 웹로그 항목들을 제거하는 과정이 요구된다(이러한 과정을 데이터 클리닝(data cleaning)이라고 한다).

데이터 클리닝을 통하여 사용자의 요구에 의해 웹로그 항목으로 기록된 웹로그 항목들로만 구성된 웹로그 집합을 얻었다면, 다음은 사용자별로 순회한 페이지들을 시간 순서에 따라 구성된 링크집합으로 구분하는 작업이 필요하다. (여기서 한 사용자가 웹 사이트에 접근하여 일정시간동안 순회한 링크집합을 세션이라고 한다.) 웹로그로부터 세션을 구분하기 위해서는 한 사용자가 웹 사이트의 순회를 종료한 시점을 알아야 한다. 그러나 HTTP가 연결이 지속되지 않는 프로토콜(stateless protocol)이기 때문에 사용자가 현재의 웹 사이트를 언제 떠났는가를 알 수 없다. 일반적으로 30분 시간제한(timeout)을 이용하는 방법을 취하고 있는데, 이것은 [13]의 실험 결과에 따른 것이다. 본 논문에서도 30분 시간제한을 이용하여 세션을 구분한다. 일반적으로 세션을 구분하는 방법은 일정 시간 간격(예 : 30분) 내에 있는 동일한 IP 주소에 의한 웹로그 항목들의 집합을 한 명의 사용자에 의한 기록으로 간주하여 하나의 세션으로 구분한다. 본 논문에서도 마찬가지로 위의 일반적인 방법을 이용하여 세션을 구분한다.

3.3 DLR(V) 알고리즘

DLR(V) 알고리즘은 순회패턴 탐사와 추천을 위해 Yan 등[7]의 벡터모델을 단순하게 수정한 것이다. DLR(V)는 후보 세션 s의 모든 페이지들에 대한 빈발 횟수를 계산한다. 이것은 세션에서 링크들간의 연관성을 전혀 고려하지 않은 방법으로 빈발 1-링크(즉, 한 페이지)들의 집합을 생성한다. DLR(V)는 빈발 1-링크집합 F_1의 집합을 생성하기 위해 세션 데이터베이스의 모든 세션들을 클러스터링하는 과정을 요구한다. 생성된 F_1 집합은 최소 지지도($s_{\min}$)에 따라 달라진다. 또한 추천과정에서 추천 링크집합 R은 F_1 집합으로부터 추천 페이지 개수 제한조건(N_{rp})에 따라 생성된다.

3.3.1 클러스터링

DLR(V)에서는 사용자 세션 $s \in D$가 주어지면 다음과 같이 사용자 세션을 벡터로 표현한다.

$$\mathbf{v}_s = \langle v_1, v_2, \cdots, v_n \rangle \quad (1)$$

$$\text{여기서, } v_i = \begin{cases} 1, & \text{if } s_i \in s \\ 0, & otherwise \end{cases}$$

여기서 주목할 점은 세션 s에서 페이지들의 빈발 횟수는 고려하지 않는다는 것이다. 즉, s_i는 세션에서의 i-번째 페이지의 존재 여부만을 나타내는 이진(binary) 변수이다.

DLR(V)에서는 K-means 알고리즘[14]을 이용하여 클러스터링을 수행한다. 먼저 세션을 n 차원(n : 전체 웹 페이지 개수) 벡터(vector) 공간으로 매핑시킨다. 일반적으로 클러스터링 알고리즘은 이 공간을 유사도(similarity measure)에 기반하여 서로 가까운 항목들의 그룹으로 분리해준다. DLR(V)에서는 세션들의 클러스터링을 위해 유사도(similarity

measure) Vector Angle(VA)와 Euclidean Distance (ED)를 사용한다. VA는 두 특징 벡터 사이의 각도거리(angular distance)를 사용하여 유사성을 계산한다.

$$VA(\mathbf{x}, \mathbf{y}) = \cos\varphi = \frac{\mathbf{x} \cdot \mathbf{y}}{|\mathbf{x}||\mathbf{y}|} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}} \quad (2)$$

여기서 $\varphi = \langle \mathbf{x}, \mathbf{y} \rangle$이고 $VA \in [0, 1] (x_i, y_i \geq 0)$이다. VA는 유사성을 표현해 주는데, $VA(\mathbf{x}, \mathbf{y})$의 값이 클수록 특징 벡터 $\mathbf{x}$와 $\mathbf{y}$는 더 유사하다고 할 수 있다. ED는 특징 벡터들의 비유사성(dissimilarity)을 정량화시켜 준다. 즉, 그 값이 클수록 비교되는 벡터들은 유사하지 않다. ED는 단순히 두 특징 벡터 사이의 유클리드 거리를 계산한다.

$$ED(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \quad (3)$$

여기서 $ED \in [0, \infty)$이다. ED의 값이 작을수록 두 벡터 $\mathbf{x}$와 $\mathbf{y}$의 유사도는 높다. *VA*와 비교해 보면, *ED*는 한 벡터와 그 벡터의 크기만을 변형시킨 벡터를 구분할 수 있다.

$$ED(\mathbf{y}, k\,\mathbf{y}) = |(k-1)\mathbf{y}| \neq 0 \ (\text{if } k \neq 1 \text{ and } \mathbf{y} \neq \mathbf{0}).$$

DLR(V)는 위에서 기술한 유사도 VA와 ED를 이용하여 클러스터링을 수행한다. 클러스터링의 목적은 유사한 순회패턴을 보이는 세션의 클러스터를 발견하는 것이다. DLR(V)에서는 잘 알려진 K-means 알고리즘을 적용한다. K-means는 처음에 무작위(random)로 클러스터의 중심(center)을 선택한 후, 선택된 중심과의 거리를 평균하여 다시 클러스터의 중심을 옮겨가는 방식으로 클러스터링을 수행하는 알고리즘이다. 이와 같이 생성된 각 클러스터의 중심 벡터집합 $CV = \{\mathbf{v}_{c_1}, \mathbf{v}_{c_2}, \cdots, \mathbf{v}_{c_k}\}$은 3.3.2절의 빈발 링크집합 탐사와 3.3.3절의 추천 링크집합 생성에 이용된다.

3.3.2 빈발 링크집합 탐사

DLR(V)는 K-means 알고리즘에 의해 생성된 각 클러스터의 중심 벡터를 이용하여 각 원소 값이 최소 지지도($s_{\min}$) 이하의 값을 가지는 원소들을 필터링(filtering)한다. 이와 같이 생성된 중심 벡터들이 빈발 *1*-링크집합 CV' (즉, F_1)이 된다. 이것은 세션에서 페이지들간의 연관성을 전혀 고려하지 않는다.

3.3.3 추천 링크집합 생성

(그림 3-1)은 DLR(V)의 추천 링크집합 생성 알고리즘을 보여준다. DLR(V)의 추천 링크집합 생성과정은 간단하다. 첫 번째 단계는 새로운 사용자의 카테고리(category)를 얻기 위해 3.3.2절에서 생성한 중심 벡터집합 CV와 액티브 세션[2] 벡터와의 유사도 계산을 통하여 가장 유사한 빈발 *1*-링크집합 CV'을 결정하는 것이다(Step 1). 다음은 CV'의 각 페이지 p에 대한 가중치를 가져와서 현재 액티브 세션 s에 대한 추천 링크집합 R을 생성한다(Step 3-7). 여기서 추천 링크집합의 크기가 매우 클 수 있다. 따라서 PR_V에서는 한번에 추천되는 링크집합의 크기에 대해 개수 제한을 둔다. 먼저 R을 가중치의 크기 순으로 정렬시킨다(Step 8). 다음은 사용자에 의해 미리 정해진 추천 링크집합 크기 제한조건 N_{rp}에 따라서 상위 N_{rp} 만큼의 페이지들이 추천된다(Step 9).

```
Procedure VectorRecommendation( CV, CV', s, N_rp )
// CV : 클러스터 중심 벡터 집합, CV' : 빈발 링크집합
   s : 액티브 세션
// N_rp : 각 추천단계에서의 추천링크 개수 제약조건
1. determine cluster c by computing match(s, CV)
2. R := ∅
3. for each page p in the frequent 1-linkset CV'(c) do {
4.      Rec(s, p) := weight(p, CV'(c))
5.      p.rec_score := Rec(s, p)
6.      R := R ∪ {p}
7. }
8. R' := sort( R )
9. select top N_rp pages from R'
end
```

(그림 3-1) DLR(V)에 의한 추천 링크집합 생성

한편, (그림 3-1)의 *match*(s, CV)은 3.3.1절에서 기술한 유사도(VA, ED)에 기반하여 계산된다. 그러나 유사도 VA와 ED는 유사한 길이의 세션들을 비교하는 경우에는 잘 적용될 수 있지만, 액티브 세션의 부분 정보와 한 클러스터의 전체 정보를 표현하는 클러스터 모델을 비교하는 경우에 과대평가(overestimation)될 수 있다. 따라서 본 논문에서는 이러한 과대평가 문제를 완화시킬 수 있는 유사도 PED(Projected ED)[2]를 적용한다. PED는 ED에서의 과대평가 문제를 해결하기 위한 변형된 유사도 측정 방법이다. 예를 들어, 두 벡터 $\mathbf{s}$와 $\mathbf{c}$를 각각 세션 벡터와 클러스터 벡터라고 하자. 각 벡터는 n개의 구성원소를 가진다. 벡터 $\mathbf{s}$와 $\mathbf{c}$의 비유사성을 계산하기 위하여 벡터 $\mathbf{s}$가 영(zero)이 아닌 구성원소를 가지는 좌표평면 상의 $\mathbf{c}$의 사상(projection)을 이용한다.

$$PED(\mathbf{s}, \mathbf{c}) = \sqrt{\sum_{i=1, s_i \neq 0}^{n} (s_i - c_i)^2} \quad (4)$$

여기서 $PED \in [0, \infty)$이고, 교환법칙은 성립되지 않는다. 따라서 PED가 여러 가지 길이로 세션들을 비교할 수 있기 때문에 주어진 시간에 세션의 일부만이 이용 가능한 실시간 클러스터링에 유용하게 사용될 수 있다.

2) 액티브 세션(active session)은 현재 사용자가 웹 사이트를 순회하는 경로(path)를 의미한다. 액티브 세션 벡터는 액티브 세션의 현재까지의 부분정보를 이진 벡터로 표현한 것이다.

3.4 DLR(M) 알고리즘

DLR(M)은 빈발 링크집합을 생성하기 위해 세션 데이터베이스로부터 빈발 *2*-링크집합을 생성한다. 이것은 세션의 인접한 페이지들간의 연관성을 고려한 것이다. DLR(M)은 F_2 집합을 생성하기 위해 세션 데이터베이스의 모든 세션들을 각 클러스터별로[3] 인접 행렬(adjacency matrix)을 생성한다. 이러한 인접 행렬을 이용하여 액티브 세션의 현재 페이지와 연관성이 높은 링크집합을 생성할 수 있다. 즉, DLR(M)에서 생성한 CV를 이용하여 현재 액티브 세션의 클러스터를 찾은 다음 액티브 세션의 현재 페이지와 연관성이 높은(즉, 평균 빈발횟수가 높은) 페이지들의 집합을 생성한다. DLR(M)의 F_2 집합은 최소 지지도($s_{\min}$)에 따라 달라진다. 또한 추천과정에서 추천 링크집합 R은 F_2 집합으로부터 추천 페이지 개수 제한조건(N_{rp})에 따라 생성된다.

3.4.1 클러스터링

DLR(M)에서의 클러스터링은 3.3.1절의 DLR(V)의 클러스터링을 그대로 적용한다.

3.4.2 빈발 링크집합 탐사

DLR(M)은 먼저 세션 데이터베이스에서의 모든 세션들에 대하여 클러스터별로 인접 행렬(adjacency matrix) AM(즉, 빈발 *2*-링크집합 F_2)을 생성한다. 여기서 클러스터별로 AM을 생성할 수 있는 것은 3.3.1절의 클러스터링의 결과로 얻은 각 세션에 대한 *cluster_id*를 세션 데이터베이스에 저장해 두었기 때문이다. DLR(M)에서는 이러한 *cluster_id*를 이용하여 클러스터별로 인접 행렬 AM을 생성한다. 예를 들어, 웹 사이트의 링크집합 $\{1, 2, \cdots, n\}$이 주어지면, 인접 행렬 AM을 $n \times n$ 행렬로 정의한다. 여기서 $AM(i, j)$는 세션에서 페이지 i로부터 j로 가는 링크가 존재하면 1의 값을 추가하고, 그렇지 않으면 무시한다. 이때 한 세션에서 페이지 i로부터 j로 가는 링크가 2회 이상 발생해도 1의 값만 추가된다. 이와 같이 클러스터별로 모든 세션들에 대해 $AM(i, j)$를 계산하였다면, 이제 각 클러스터별 $AM(i, j)$를 해당 클러스터의 크기로 나눈다. 따라서 모든 $AM(i, j) \in [0, 1]$이 된다. 마지막으로 모든 클러스터별 $AM(i, j)$에 대하여 최소 지지도($s_{\min}$) 이하의 값을 가지는 원소들을 0의 값으로 한다. 이러한 인접 행렬은 세션에서의 인접한 두 페이지 간의 연관성을 고려한 빈발 *2*-링크집합 F_2를 의미한다. 이것은 인접한 페이지 간의 연관성을 전혀 고려하지 않은 DLR(V)에서보다 한 단계 확장된 방법으로 볼 수 있다.

3.4.3 추천 링크집합 생성

(그림 3-2)는 DLR(M)의 추천 링크집합 생성 알고리즘을 보여준다. DLR(M)의 추천 링크집합 생성을 위한 첫 번째 단계는 새로운 사용자의 카테고리를 얻기 위해 3.4.2절에서 생성한 클러스터별 인접행렬 AM(즉, 빈발 *2*-링크집합 F_2)과 액티브 세션 벡터 s와의 유사도 계산을 통하여 가장 유사한 빈발 *2*-링크집합 $AM(c)$(즉, $F_2(c)$)을 결정하는 것이다(Step 1). 다음은 $AM(c)$로부터 액티브 세션 s의 현재 페이지에 대응되는 행 r을 추출한다(Step 2). 이러한 행벡터 $\mathbf{r} = \langle r_1, r_2, \cdots, r_n \rangle$의 각 페이지들이 후보 추천 링크집합이 된다. 즉, $AM(c)$의 행벡터 r의 각 페이지에 대한 가중치를 가져와서 현재 액티브 세션 s에 대한 추천 링크집합 R을 생성한다(Step 4-8). 여기서 추천 링크집합의 크기가 매우 클 수 있다. 따라서 DLR(M)에서도 마찬가지로 한번에 추천되는 링크집합의 크기에 대해 개수 제한을 둔다. 먼저 R을 가중치의 크기 순으로 정렬시킨다(Step 9). 다음은 사용자에 의해 미리 정해진 추천 링크집합 크기 제한조건 N_{rp}에 따라서 상위 N_{rp} 만큼의 페이지들이 추천된다(Step 10).

```
Procedure MatrixRecommendation( CV, AM, s, N_rp )
// CV : 클러스터 중심 벡터집합, AM : 인접 행렬, s : 액티브 세션
// N_rp : 각 추천단계에서의 추천링크 개수 제약조건
1. determine cluster AM(c) by computing match(s, CV) ;
2. AM(c)로부터 s의 현재 페이지에 대응되는 행 r 추출 ;
3. R := ∅ ;
4. for each page p of selected row vector r of AM(c) do {
5.     Rec(s, p) := weight(p, r) ;
6.     p.rec_score := Rec(s, p) ;
7.     R := R ∪ p ;
8. }
9. R' := sort(R)
10. select top N_rp pages from R'
end
```

(그림 3-2) DLR(M)에 의한 추천 링크집합 생성

4. 실험 및 고찰

4.1 실험 방법

DLR 알고리즘의 평가는 다음과 같은 세 가지 척도에 기반하여 평가할 수 있다[4] : Impact, Benefit.

- Impact : 얼마나 많은 사용자가 얼마나 자주 추천한 페이지를 이용하였나.
- Benefit : 웹 사이트를 방문한 사용자들이 얼마나 많은 노력을 절약하였나.

본 논문에서는 각 사이트의 웹 서버 로그 데이터를 두 개의 그룹으로 나누어 실험을 수행한다 ; 훈련 데이터와 테스트 데이터. 실험에 사용하는 척도는 위에서 언급한 impact/benefit을 기본적으로 사용한다. 3장에서 기술한 DLR 알고리즘들을 먼저 훈련 데이터에 적용하여 추천 링크집합을 생

3) DLR(V)에서 클러스터링 수행시 세션 데이터베이스에 각 세션에 대한 클러스터 ID를 저장한다. 따라서 클러스터 ID를 이용하여 클러스터별로 인접 행렬을 생성할 수 있다.

성한 후 테스트 데이터를 이용하여 impact/benefit을 비교함으로써 평가한다.

Impact는 세션 데이터베이스의 각 테스트 세션이 추천 링크집합 R의 $x(x \geq 1)$ 링크들을 접근한 세션들의 개수를 측정한다. 사이트를 변형시킬 수 없는 경우에 테스트 데이터의 각 세션에 대해 추천 링크집합에 있는 $x(x \geq 1)$ 링크들을 방문한 사용자 수를 카운트함으로써 측정할 수 있다.

Benefit은 측정하기가 조금 어렵다. 정보를 탐색하기 위한 사용자의 노력은 사용자가 클릭한 링크의 개수와 그러한 링크들을 찾는데 있어서의 어려움에 달려 있다. 즉, 사용자의 클릭수를 줄일 수 있고, 사용자가 순회해야만 하는 페이지의 복잡도를 감소시킬 수 있다면 제안한 방법은 사용자들에게 많은 혜택을 주는 것이다. 이러한 혜택을 정확하게 측정하려면 제안한 웹 정보 서비스 방법을 적용했을 때와 하지 않았을 때 사용자들이 웹 사이트를 순회하는데 있어서 드는 노력의 평균을 비교해야 한다. 그러나 이것은 현실적으로 어려운 일이다. 따라서 본 논문에서는 사용자가 추천 링크집합 내에서 선택한 페이지의 개수를 이용하여 근사시킨다. 사이트를 변형시킬 수 없는 경우에 사용자의 세션에 존재하는 추천 링크집합에 있는 페이지의 개수를 카운트함으로써 측정한다.

본 논문에서는 제안한 알고리즘과 다른 탐사 알고리즘과의 비교를 위해 impact와 benefit 척도를 이용한다. 각 클러스터에 대하여 각 사용자에 의해 순회된 추천 링크집합 내의 페이지 개수를 카운트하고, 적어도 한 페이지를 방문한 사용자들의 수, 적어도 두 페이지를 방문한 사용자들의 수 등을 계산한다. 다음은 특정 알고리즘에 의해 생성된 모든 클러스터에 대해 평균을 계산한다. 각 알고리즘에 대해 benefit(추천 링크집합에서 선택된 페이지 수) 대 impact(페이지 개수별로 순회한 사용자들의 수)를 그래프로 표현한다. 만일 특정 알고리즘에서 특정한 개수 이상의 페이지들(예 : 일곱 개 이상의 페이지들)을 순회한 사용자들이 없다면 해당 알고리즘의 그래프는 거기서 종료한다. 모든 실험에서 각 알고리즘은 조정할 수 있는 파라메타(parameters)를 갖는다. 모든 경우에 최적의 결과가 나오도록 파라메타를 조정한다.

4.2 실제 데이터

본 논문에서의 실험은 두 개의 웹 사이트로부터 얻은 로그 데이터를 이용하여 수행한다. 첫 번째 사이트는 아시아나 항공의 AsianaLady 사이트[4]이다. 이 사이트는 211개의 서로 다른 웹 페이지들로 구성되어 있으며, 그 외 많은 이미지와 텍스트 등이 포함되어 있다. AsianaLady 사이트로부터 총 84일간의 로그 데이터를 얻었다. 로그 데이터는 하루에 평균 69,833명의 사용자로부터 5,912개의 히트를 기록한다. 전체 히트 수는 4,913,731건이며, 총 접속자 수는 408,648명, 그리고 평균 세션 길이는 12.02로 나타났다. 실험에서 훈련 데이터(training data)로는 71일간(2002년 12월 2일~2003년 2월 11일)의 로그 데이터를 이용하였으며, 전처리[5] 한 결과로 유효 히트 수는 761,678건이고, 유효 세션 수는 117,893, 그리고 유효 세션 평균 길이는 6.46을 얻었다. 테스트 데이터는 13일간(2003년 2월 12일~2003년 2월 24일)의 로그 데이터를 이용한다. 테스트 데이터의 전체 히트 수는 928,454건이며, 이 기간 동안 총 접속자 수는 79,933명이고, 평균 세션 길이는 11.62로 나타났다. 전처리한 결과로 유효 히트 수는 161,345건이며, 유효 세션 수는 24,330, 그리고 유효 세션 평균 길이는 6.63으로 나타났다.

두 번째 사이트는 KBS 방송국의 미디어서버 사이트[6]로 이 사이트는 62개의 서로 다른 웹 페이지들로 구성되어 있으며, 그 외 많은 이미지와 텍스트 등이 포함되어 있다. KBS 미디어서버 사이트로부터 총 16일간의 로그 데이터를 얻었다. 로그 데이터는 하루에 평균 12058명의 사용자로부터 평균 300,457개의 히트를 기록한다. 전체 히트 수는 4,807,307건이며, 총 접속자 수는 192,934명, 그리고 평균 세션 길이는 24.92로 나타났다. 실험에서 훈련 데이터(training data)로는 12일간(2002년 2월 9일~2월 20)의 로그 데이터를 이용하였으며, 전처리 한 결과로 유효 히트 수는 425,725 건이고, 유효 세션 수는 34,576, 그리고 유효 세션 평균 길이는 12.31을 얻었다. 테스트 데이터는 4일간(2002년 2월 21일~2월 24일)의 로그 데이터를 이용한다. 테스트 데이터의 전체 히트 수는 1,073,934 건이며, 이 기간 동안 총 접속자 수는 41,617명이며, 평균 세션 길이는 25.81로 나타났다. 전처리한 결과로 유효 히트 수는 126,616건이며, 유효 세션 수는 10,120, 그리고 유효 세션 평균 길이는 12.51로 나타났다.

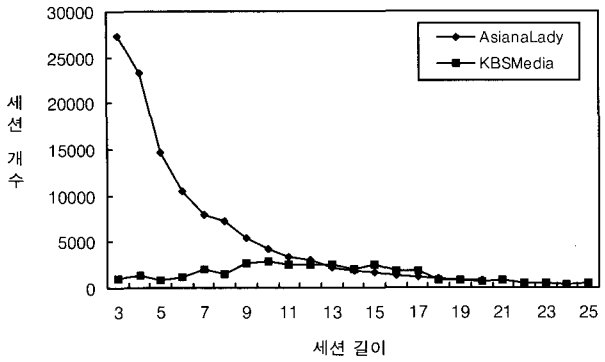
실험 환경으로 OS는 Windows 98, CPU는 Pentium III, RAM 512MB, 프로그래밍 언어는 JAVA를 사용하였다. (그림 4-1)은 두 개의 실제 사이트의 웹로그 데이터로부터 얻은 세션들의 길이에 대한 분포도를 보여준다. 가로축은 각 세션이 포함하는 페이지들의 개수를 표현하고, 세로축은 $x(x \geq 1)$ 페이지 이상을 접근한 세션들의 개수를 표현한다. 본 논문에서는 세션의 길이가 너무 짧거나 혹은 너무 긴 경우는 관심이 없다. 즉, 많은 사용자들의 순회패턴에 적용하기 위해 최소 페이지 개수(즉, 3페이지) 미만인 세션들과 최대 페이지 개수(즉, 20페이지 혹은 25페이지)를 초과하는 세션들을 제거한다. (그림 4-1)은 훈련 데이터와 테스트 데이터에 대하여 이와 같은 전처리 과정을 수행한 결과를 보여준다. 최소 페이지 개수는 두 개의 사이트 모두 3페이지 이상인 세션들로 제한하고, 최대 페이지 개수는 AsianaLady

4) http://rsvweb.asiana.co.kr/.

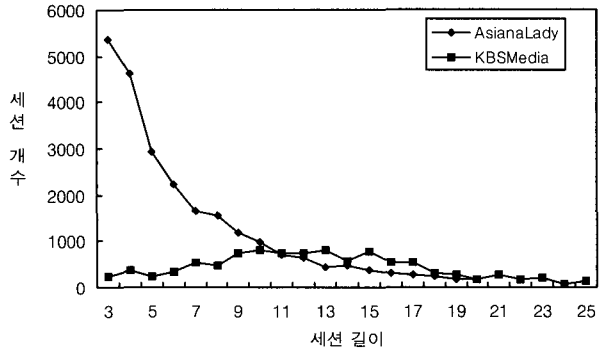
5) 실제 세션에서 추천 페이지로서의 의미가 없는 페이지(예 : 메뉴 페이지)들은 삭제된다.

6) http://www.kbsmedia.co.kr/.

사이트의 경우 20페이지, 그리고 KBSMedia 사이트의 경우는 25페이지로 각각 다르게 제한을 두었다. 이것은 두 개의 웹로그에 대한 세션길이의 분포도에 따른 것이다.



(a) Train data



(b) Test data

(그림 4-1) 세션 길이별 분포(아시아나 항공의 AsianaLady 사이트와 KBS의 KBSMedia 사이트)

4.3 실제 데이터를 이용한 DLR 알고리즘들의 성능 평가

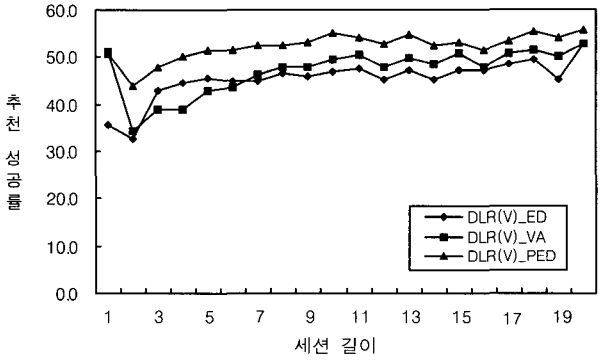
이 절에서는 제안한 DLR 알고리즘들을 4.2절에서 기술한 두개의 실제 데이터에 적용한 실험을 수행한다. 실험 방법은 4.1절에서 기술한 방법과 동일하다.

4.3.1 클러스터링

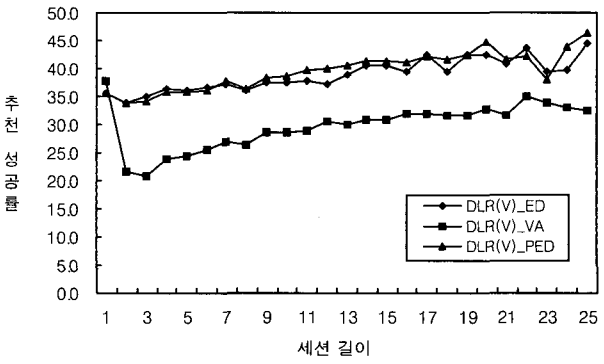
3.3절에서 언급한 바와 같이 DLR(V)와 DLR(M)에서는 클러스터링 과정이 요구된다. 본 논문에서는 잘 알려진 K-means 알고리즘을 이용하여 세션 데이터베이스의 세션들을 클러스터링한다. 유사도 ED를 적용한 경우 AsianaLady는 K = 42개, 그리고 KBSMedia는 K = 37개의 클러스터들을 생성하였다. 유사도 VA를 적용한 경우 AsianaLady는 K = 66개, 그리고 KBSMedia는 K = 39개의 클러스터들을 생성하였다. K-means 알고리즘을 수행할 때 초기 K의 개수는 각 사이트의 전체 페이지 개수로 정하여 시작하였다. 종료 조건은 각 클러스터의 세션들이 40개 이하로 변동될 때까지 수행하는 것으로 주었다. 대부분의 경우 6~9 사이클 내에서 수렴하였다.

4.3.2 DLR(V)_ED, DLR(V)_VA, DLR(V)_PED의 성능 평가

DLR(V)_ED, DLR(V)_VA, DLR(V)_PED 중에서 가장 성능이 좋은 모델을 선택하여 DLR(V)로 표기한다. 가장 좋은 모델 선택을 위해 먼저 각 모델들의 세션 길이별 추천 성공률을 조사하였다. (그림 4-2)와 (그림 4-3)은 각각 N_{rp} = 10인 경우와 N_{rp} = 5인 경우에 세 가지 모델들에 대한 추천 성공률을 보여준다. 그림에서 알 수 있듯이 N_{rp} = 10인 경우의 DLR(V)_PED 모델이 가장 우수함을 보여준다. 실제 N_{rp} = 10인 경우의 AsianaLady에 대하여 평균 추천 성공률은 DLR(V)_ED가 41.7%, DLR(V)_VA가 43.3%, 그리고 DLR(V)_PED가 49.9%를 보여주었다. 반면에 N_{rp} = 5인 경우의 평균 추천 성공률은 DLR(V)_ED가 26.9%, DLR(V)_VA가 28.5%, 그리고 DLR(V)_PED가 41.0%를 보여주었다. KBSMedia의 경우에도 이와 유사한 결과를 보여주었다. 따라서 이후부터 N_{rp} = 10인 경우의 DLR(V)_PED 모델을 DLR(V)의 대표 모델로 간주하여 DLR(V)로 표기한다. 또한 모든 실험에서 N_{rp} = 10인 경우와 N_{rp} = 5인 경우에 대해 실험하였으나 모든 실험 방법에서 N_{rp} = 10인 경우에 더 좋은 성능을 보였다. 따라서 이후 모든 실험 결과는 N_{rp} = 10인 경우만을 보여준다.

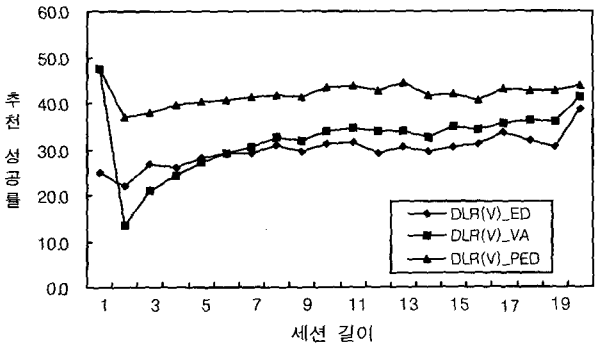


(a) AsianaLady

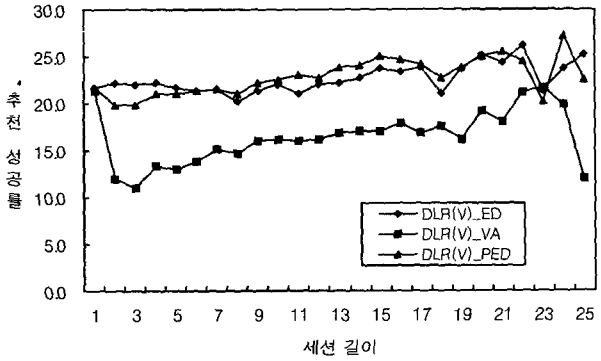


(b) KBSMedia

(그림 4-2) DLR(V)_ED, DLR(V)_VA, DLR(V)_PED의 세션 길이별 추천 성공률(N_{rp} = 10인 경우)

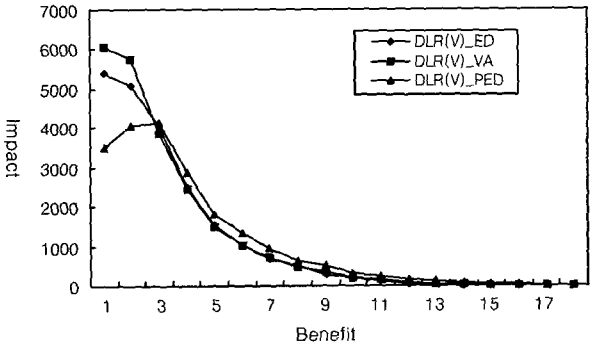


(a) AsianaLady

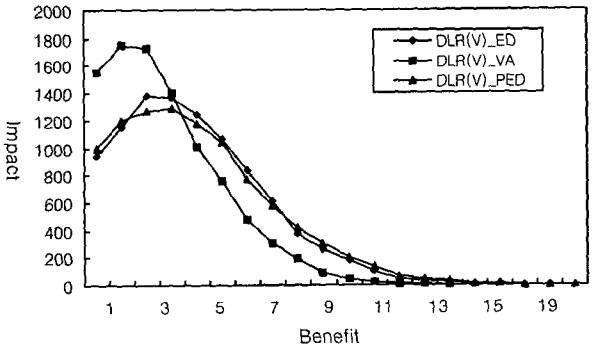


(b) KBSMedia

(그림 4-3) DLR(V)_ED, DLR(V)_VA, DLR(V)_PED의 세션 길이별 추천 성공률(N_{rp} = 5인 경우)



(a) AsianaLady



(b) KBSMedia

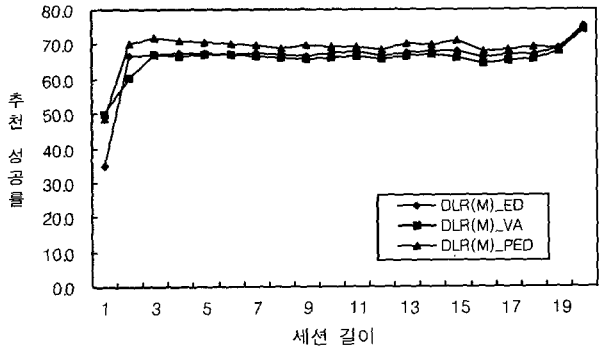
(그림 4-4) DLR(V)_ED, DLR(V)_VA, DLR(V)_PED의 성능 평가(N_{rp} = 10인 경우)

(그림 4-4)는 DLR(V)_ED, DLR(V)_VA, DLR(V)_PED의 Impact/Benefit기능 평가를 보여준다(N_{rp} = 10인 경우). 그림에서 DLR(V)_PED의 결과가 가장 우수함을 보여준다.

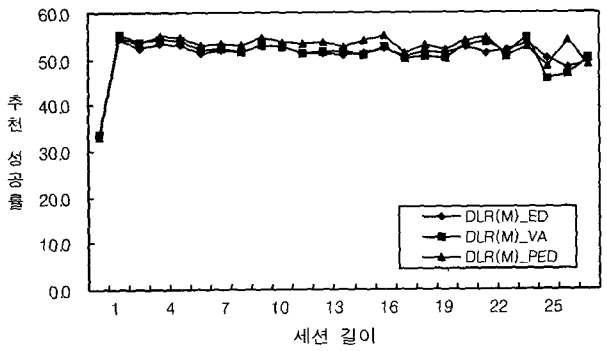
4.3.3 DLR(M)_ED, DLR(M)_VA, DLR(M)_PED의 성능 평가

DLR(M)_ED, DLR(M)_VA, DLR(M)_PED 중에서 가장 성능이 좋은 모델을 선택하여 PR_M으로 표기한다. 가장 좋은 모델 선택을 위해 먼저 각 모델들의 세션 길이별 추천 성공률을 조사하였다. (그림 4-5)는 N_{rp} = 10인 경우의 세 가지 모델들에 대한 추천 성공률을 보여준다. 실제 N_{rp} = 10인 경우의 AsianaLady에 대하여 평균 추천 성공률은 DLR(M)_ED가 62.2%, DLR(M)_VA가 63.2%, 그리고 DLR(M)_PED가 67.2%를 보여주었다. 반면에 N_{rp} = 5인 경우의 평균 추천 성공률은 DLR(M)_ED가 51.0%, DLR(M)_VA가 53.5%, 그리고 DLR(M)_PED가 57.0%를 보여주었다. KBSMedia의 경우에도 이와 유사한 결과를 보여주었다. 따라서 이후부터 N_{rpN} = 10인 경우의 DLR(M)_PED 모델을 PR(M)의 대표 모델로 간주하여 PR_M으로 표기한다.

(그림 4-6)은 DLR(M)_VA, DLR(M)_ED, DLR(M)_PED의 Impact/Benefit 성능평가를 보여준다(N_{rp} = 10인 경우). 그림에서 DLR(M)_PED의 결과가 가장 우수함을 보여준다.

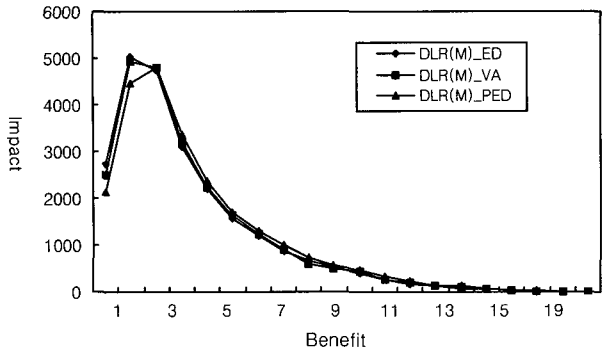


(a) AsianaLady

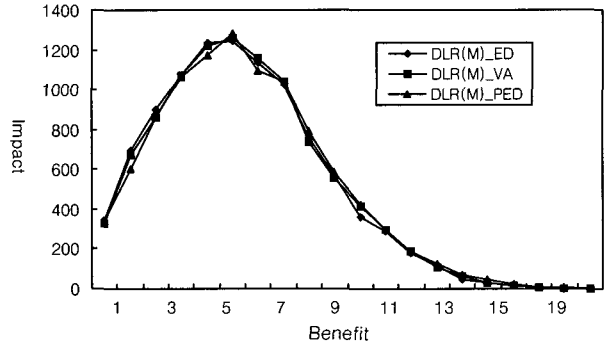


(b) KBSMedia

(그림 4-5) DLR(M)_ED, DLR(M)_VA, DLR(M)_PED의 세션 길이별 추천 성공률(N_{rp} = 10인 경우)

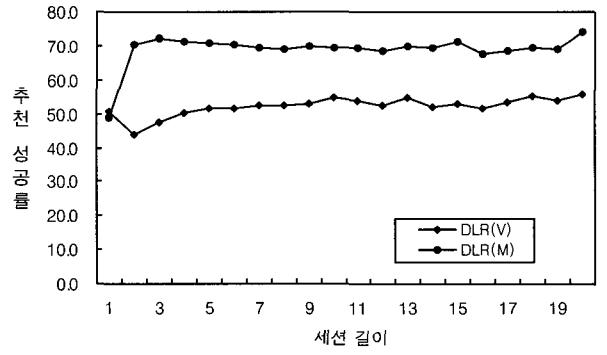


(a) AsianaLady

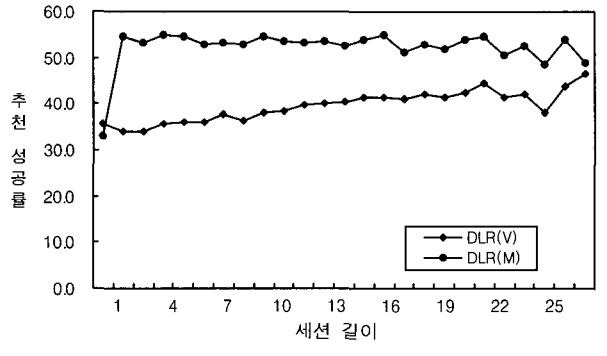


(b) KBSMedia

(그림 4-6) DLR(M)_ED, DLR(M)_VA, DLR(M)_PED의 성능 평가(N_{rp} = 10인 경우)



(a) AsianaLady



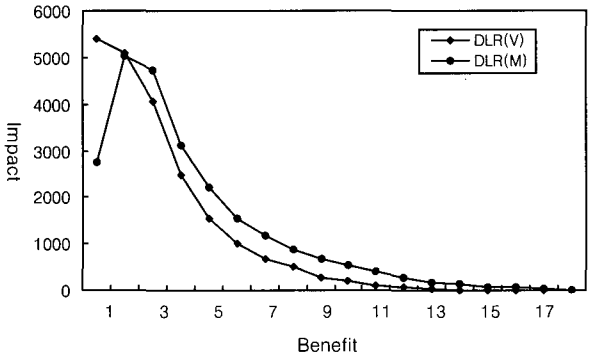
(b) KBSMedia

(그림 4-7) DLR 알고리즘들의 세션 길이별 추천 성공률 (N_{rp} = 10인 경우)

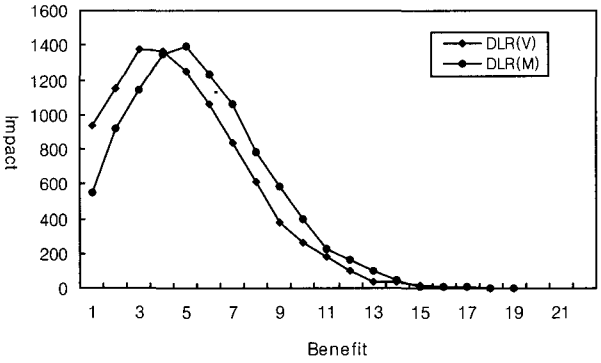
4.3.4 DLR(V), DLR(M)의 성능 평가

(그림 4-7)은 DLR(V), DLR(M)의 두 개의 실제 사이트에 대하여 적용한 세션 길이별 추천 성공률을 보여준다(N_{rp} = 10인 경우). AsianaLady의 경우 평균 추천 성공률은 DLR(V)가 49.9%, DLR(M)이 67.2%로 DLR(M)의 결과가 DLR(V)에 비해 월등한 추천 성능을 보여준다. KBSMedia의 경우 AsianaLady의 경우보다는 추천 성능이 약간 떨어진다. DLR(V)가 37.5%, DLR(M)이 52.0%를 보여준다.

(그림 4-8)은 DLR(V)와 DLR(M)을 두 개의 실제 사이트에 적용한 Impact/Benefit 성능평가를 보여준다(N_{rp} =10인 경우).



(a) AsianaLady



(b) KBSMedia

(그림 4-8) DLR(V), DLR(M)의 성능 평가(N_{rp} = 10인 경우)

7. 결 론

본 논문에서는 세션에 나타나는 링크간의 연관성 정보를 활용하여 빈발 링크집합을 생성하고, 이를 기반으로 하여 추천 링크집합을 탐사함으로써 효율적인 웹 정보 서비스를 제공할 수 있는 DLR(Dynamic Link Recommendation) 알고리즘을 제안하였다. 제안한 DLR 알고리즘은 빈발 링크집합을 탐사하기 위해 링크간의 연관성 정보를 이용한다. 실험 결과에서도 알 수 있듯이 링크간의 연관성 정보를 더 많이 이용하는 DLR(M)의 추천 성능이 우수하다. DLR 알

고리즘들은 웹로그로부터 빈발 링크집합을 탐사하고, 이를 기반으로 하여 추천 링크집합을 생성한다. 이와 같이 추천 링크집합에 기반하여 웹 사이트를 방문한 사용자에게 추천 링크집합을 포함하는 새로운 페이지뷰(page view)를 제공함으로써 궁극적으로 찾고자하는 목표 페이지에 효과적으로 접근할 수 있도록 한다.

향후 연구과제로는 페이지간의 연관성 정보를 활용하는 정도를 좀 더 체계적으로 기술하는 것과 다양한 실험을 통하여 DLR 알고리즘의 특성을 파악하는 것이다. 마지막으로 더 많은 실제 사이트에 적용하는 것과 데모 사이트를 구축하여 DLR 알고리즘을 적용하는 것이다.

참 고 문 헌

[1] Chen, M. S., Park, J. S. and Yu, P. S., "Efficient Data Mining for Path Traversal Patterns," IEEE Trans. on Knowledge and Data Engineering, Vol.10, No.2, pp.209-221, March, 1998.

[2] Shahabi, C., Banaei-Kashani, F., Faruque, J. and Faisal, A., "Feature Matrices : A Model for Efficient and Anonymous Web Usage Mining," EC-Web 2001, Germany, September, 2001.

[3] Shahabi, C., Zarkesh, A. M., Adibi, J. and Shah, V., "Knowledge discovery from users Web-page navigation," Proc. of the IEEE RIDE '97 Workshop, April, 1997.

[4] Perkowitz, M. and Etzioni, O., "Towards adaptive Web sites : Conceptual framework and case study," Artificial Intelligence, Vol.118, pp.245-275, 2000.

[5] Büchner, A. and Mulvenna, M. D., Discovering internet marketing intelligence through online analytical Web usage mining, SIGMOD Record, 27(4), 1999.

[6] Lieberman, H., "Letizia : An Agent That Assists Web Browsing," Proc. of the 1995 Int. Joint Conf. on AI, Montreal, Canada, 1995.

[7] Yan, T. W., Jacobsen, M., Molina, H. G., and Dayal, U., "From User Access Patterns to Dynamic Hypertext Linking," The 5th Int'l World Wide Web Conf., Paris, France, May, 1996.

[8] Ngu, D. S. W. and Wu, X., "SiteHelper : A Localized Agent that Helps Incremental Exploration of the World Wide Web," The 6th Int'l World Wide Web Conf., Santa Clara, CA, pp.691-700, 1997.

[9] Joachims, T., Freitag, D., Mitchell, T., "WebWatcher : A Tour Guide for the World Wide Web," IJCAI 97, Proc. of the 5th Int'l Joint Conf. on AI, Nagoya, Japan, pp.770-775, 1997.

[10] Zakesh, A. M., Adibi, J., Shahabi, C., Sadri, R., Shah, V., "Analysis and Design of Server Informative WWW-sites," Proc. of the ACM CIKM, 1997.

[11] Mobasher, B., Cooley, R. and Srivastava, J., "Creating adaptive web sites through usage-based clustering of urls," Knowledge and Data Engineering Workshop, 1999.

[12] Mobasher, B., Cooley, R., and Srivastava, J., "Automatic Personalization Based on Web Usage Mining," Communications of the ACM, 43(8), pp.142-151, 2000.

[13] Catledge, L. and Pitkow, J., "Characterizing browsing behaviors on the world wide web," Computer Networks and ISDN Systems, 27(6), 1995.

[14] Han, J. and Kamber, M., Data Mining : Concepts and Techniques, Morgan Kaufmann publishers, pp.349-351, 2001.

윤 선 희

e-mail : sunniyoon@hanmail.net

1986년 숭실대학교 전자계산학과(공학사)

1988년 숭실대학교 전자계산학과(공학석사)

1998년 숭실대학교 전자계산학과 박사과정 수료

1992년~현재 미림전산고등학교 교사

관심분야 : 데이터마이닝, 웹컴퓨팅, 멀티미디어 통신, 멀티미디어 응용 등

오 해 석

e-mail : oh@computing.soongsil.ac.kr

1975년 서울대학교 응용수학과(공학사)

1981년 서울대학교 계산통계학과(이학석사)

1989년 서울대학교 계산통계학과(이학박사)

1996년~1999년 숭실대학교 부총장 역임

1983년~현재 숭실대학교 컴퓨터학과 교수

관심분야 : 멀티미디어 통신, 웨이블릿 영상코딩, 멀티미디어 응용 등