인터넷에서 정보 탐색에 대한 연구 조사
A Survey of Information Searches on Internet

강병주, 백혜승, 최기선
전산학과 한국과학기술원

Byung-Ju Kang, Hae-Seung Paik, Key-Sun Choi
Dept. of Computer Science
Korea Advanced Institute of Science and Technology

The huge size of Internet does not allow ordinary information seekers to search information with ease. Now, it is almost impossible to navigate the ocean of information without effective search tools. Web search engine has been the most effective technology for information retrieval on WWW. But recently, the need for new search tools on WWW or Internet has increased drastically. Currently, there are many on-going researches on the related topics. In this survey, we categorize the new search tools into four types: monitoring systems, filtering systems, browsing assistant systems, recommending systems. These example systems are examined. We are especially interested in WWW information filtering. It is studied how to apply the information filtering techniques to WWW. The application is not so straightforward like Email, Newswire filtering systems. As a result of this study , a simple WWW information filtering system is proposed.

1. 서론

인터넷 또는 World Wide Web에는 엄청난 양의 정보가 있으며 하루가 다르게 그 양이 증가하고 있다. 이제 WWW에서 필요한 정보를 찾는 것은 더 이상 간단한 일이 아니다. 아무리 많은 정보가 있더라도 원하는 정보를 정확히 찾을 수 있는 방법이 없다면 그 많은 정보는 아무런 쓸모가 없을 것이다.

현재 가장 대표적인 WWW 검색 도구는 웹검색엔진이다. 웹검색엔진은 WWW의 대부분의 정보가 HTML 문서이고 HTML 문서가 텍스트라는 점에 착안하여 기존의 정보검색(Information Retrieval) 또는 문서검색(Document Retrieval) 기술을 WWW에 적용한 시스템이라고 할 수 있다. 따라서 웹검색엔진은 WWW 상의 HTML 문서를 색인하여 가지고 있다가 사용자의 질의와 색인을 비교하여 적합한 HTML 문서들의 URL을 제공해준다. 또 다른 유용한 검색 도구는 웹디렉토리(Web directory) 서비스이다. 대표적인 것이 야후[Yahoo]인데 유명 웹사이트(Web site)를 주제별로 계층적으로 분류해두고 있어 사용자가 해당 주제에 관련된 웹사이트를 쉽게 찾을 수 있게 도와준다.

하지만 하루가 다르게 크지는 WWW, 더욱 다양해지고 있는 WWW 상의 정보의 내용, 정보의 유형, 그리고 동적 HTML 문서를 지원하는 다양한 웹애플리케이션(Web application)의 증가로 기존의 WWW 검색 도구들인 웹검색엔진, 웹디렉토리 서비스, 등 만으로는 원하는 정보를 찾기가 어렵게 되었다. 최근 인터넷 또는 WWW에서 정보 검색을 도와주는 새로운 검색 도구들이 속속 제안되고 있다.

인터넷 상의 정보를 사용자의 관심(interest)에 따라 필터링해서 사용자의 관심에 부합하는 정보만을 사용자에게 제공해주는 필터링 시스템(filtering system), 사용자가 관심을

가지는 웹사이트를 정기적으로 방문, 확인하여 변경된 사항이 있으면 이를 사용자에게 알려주는 모니터링 시스템(monitoring system), 사용자가 필요로 하는 정보를 효과적으로 찾을 수 있게 WWW에서 브라우징(browsing)을 도와주는 브라우징 보조 시스템(browsing assistant system), 그리고 사용자들의 비슷한 검색 요구 및 정보에 대한 평가를 모아두었다가 사용자의 검색 요구에 대해 다른 사용자들의 평가 또는 주석(annotations)에 기반하여 유력한 정보를 추천해주는 추천 시스템(recommending system), 등이 대표적인 새로운 검색 도구들이다.

Email 필터링에서 시작된 정보필터링(Information Filtering)은 대량의 컴퓨터 네트워크 상의 정보의 필터링의 현실적인 필요성의 높아짐에 따라 최근 많은 연구자들의 꾸준한 관심을 끌고 있다 [Oard 96]. 정보필터링의 대상은 Email, Newswire, Usenet News, 등에서 최근 WWW으로 확대되고 있다 [Falk 96][Pazzani.95]

WWW 정보필터링이란 무엇인가라는 질문에 대한 정확한 해답은 아직 없는 것 같다. 사실 WWW에 정보필터링을 어떻게 적용할 수 있는 지가 분명하지 않다. WWW은 Email, Newswire, Usenet News와는 성격이 많이 다르다. Email이나 Newswire처럼 유입되는 정보가 아니고 Usenet News 처럼 정보 저장장소(repository)가 집중되어 있지 않고 WWW 정보는 전세계에 걸쳐 독립적으로 분산되어 있다. 여하턴 이런 어려움을 무시한다면 WWW에서 정보필터링(information filtering)은 WWW의 웹오브젝트(Web object)를 사용자들의 관심 프로파일(interest profile)과 비교하여 일치되는 사용자에게만 웹오브젝트나 URL을 제공하는 과정(process)이라고 볼 수 있다.

WWW 정보필터링은 지능형 인터페이스 에이전트(Intelligent Interface Agent)로 구현하는 것이 여러 가지로 편리하다 [Lieberman 96]. WWW 정보필터링 에이전트는 WWW을 정보탐색 공간으로 해서 정보필터링(information filtering) 기술을 사용하여 사용자의 정보 검색을 도와주는 소프트웨어 에이전트(software agent)라고 정의할 수 있다. 좀 더 구체적으로 WWW 필터링 에이전트는 WWW을 탐색전략에 따라 이동하면서 웹오브젝트와 사용자 프로파일과 비교하여 일치하는 웹오브젝트만을 사용자에게 제공하는 에이전트이다.

2장에서는 기존의 WWW 검색 도구들의 현황과 문제점에 대하여 현재의 대표적인 검색 도구인 웹검색엔진을 중심으로 살펴보고, 3장에서는 현재 새롭게 제안되고 있는 WWW 검색 도구들을 유형별로 정리해서 내용을 살펴본다. 그리고 4장에서는 WWW 정보필터링 에이전트와 관련된 여러 가지 요소 기술에 대해서 알아보고 마지막으로 우리가 구현 중에 있는 WWW 정보필터링 에이전트를 간략히 소개한다.

2. WWW에서의 정보검색: 웹검색엔진 (Information Retrieval on WWW : Web Search Engine)

웹검색엔진은 World Wide Web을 가능하게 하는 현재 가장 중요한 기술 중의 하나이다. 웹검색엔진이 없다면 WWW의 존재 자체가 불투명할 것이다. 현재 많은 웹검색엔진이 공개되어 있으며 국내에도 한글 웹(Web) 문서의 검색을 도와주는 웹검색엔진이 다수 공개 되어 있다.

WWW에 있는 정보의 대부분이 텍스트 형태이다. 따라서 기존의 문서검색 (Document Retrieval) 또는 정보검색 (Information Retrieval) 기술을 그대로 WWW에 적용할 수 있다. 웹검색엔진은 WWW상의 HTML 문서들을 색인하여 그 색인 정보를 가지고 있다가 사용자의 질의가 들어오면 질의와 색인을 비교하여 유사도가 높은 HTML 문서들의 URL을 제공해주는 시스템이라고 할 수 있다.

2.1. 웹 탐색 공간 (Web Search Space)

웹검색엔진의 색인의 대상이 되는 WWW 오브젝트는 HTML 문서에만 국한되지는 않는다. Lycos[Lycos]의 경우 색인의 대상이 되는 웹탐색 공간은 HTTP 공간, FTP 공간, 그리고 Gopher 공간이다. 반면에 WAIS database,

Usenet News, MAILTO 공간, TELNET 공간, 그리고 Local file 공간은 포함되지 않는다. 그리고 색인이 가능하기 위해서는 텍스트 또는 텍스트로 변환될 수 있는 파일 형태이어야만 하기 때문에 Lycos는 파일 확장자가 AU, AVI, BIN, DAT, DVI, EXE, FLI, GIF, GZ, HDF, HQX, JPEG, LHA, MAC, MPEG, PS, TAR, TGA, TIFF, UU, UUE, WAV, Z, ZIP, 등인 파일은 무시한다 [Cheong 96]. 이렇게 색인 대상이 되는 웹 공간은 웹검색엔진에 따라 다르며 전체 WWW 공간의 부분 공간(sub-space)을 이룬다.

HTTP 공간만을 탐색 공간 또는 색인 대상 공간으로 잡더라도 그 크기는 엄청날 것이다. 대표적인 웹검색엔진인 AltaVista의 경우 627,000 웹 서버(web server)에서 발견된 31,000,000 웹 페이지(Web page)에 대한 색인을 구축하여 가지고 있다 [AltaVista]. 참고로 국내의 대표적인 웹검색엔진인 한글과컴퓨터의 심마니는 380,070 개의 웹 페이지에 대한 색인을 가지고 있다 [Simmany].

2.2. 어떤 정보를 추출할 것인가 (What Information To Keep)

HTML 문서에서 어떤 정보를 추출할 것인가를 결정하여야 한다. Lycos의 경우 제목(title), 헤딩(headings) 과 서브헤딩(subheadings), 처음 20 줄, 문서 크기(bytes), 단어 수, 가장 중요한 100 개의 단어, 등을 추출한다 [Cheong 96]. 처음 20 줄은 문서의 요약을 대신하기 위한 것이다. Lycos는 전문 색인 (full text indexing)을 하지 않고 *Tf*IDf* weighting 알고리즘[Salton 83]을 이용하여 100 개의 가장 중요한 단어를 키워드(keywords)로 추출한다. 재현율(recall ratio)은 떨어지지만 색인의 크기를 대폭 줄일 수 있다. 그리고 제목, 헤딩과 서브헤딩에 출현하는 단어에 높은 가중치를 둠으로써 HTML 문서의 구조적인 정보도 이용한다.

2.3. 문제점 (Problems)

웹검색엔진의 WWW 에서의 검색 도구로서의 훌륭한 장점에도 불구하고 다음과 같은 몇 가지 단점을 가지고 있다.

1) WWW은 그 기본 성격상 매우 동적이다. WWW 상의 HTML 문서는 수시로 그 내용이 바뀌기도 하고 문서 자체가 삭제되기도 하며 새로운 HTML 문서가 추가되기도 한다. 이러한 변경 사항이 즉각적으로 웹검색엔진의 색인 정보에 반영되는 것이 어렵다.
2) 모든 웹 문서를 색인하는 것은 거의 불가능하다. 예를 들면, 다른 HTML 문서에서 참조되지 않는 잘 알려지지 않은 웹 서버를 색인하는 것은 불가능하다.
3) 적절한 질의를 만드는 것이 어렵다. 이 문제는 전통적인 정보 검색 문제로 잘 알려져 있다. 대부분의 사용자가 정보 검색의 초보자이고 전문가라고 하더라도 검색 목적에 맞는 정확한 질의를 구성하는 것이 어렵다.

이러한 문제점들에도 불구하고 웹검색엔진은 현재로서는 가장 유용한 WWW 검색 수단임에 틀림 없고 앞으로 당분간 이 위치는 흔들리지 않을 것이다.

3. 인터넷에서의 정보탐색을 위한 새로운 도구들 (New Tools for Information Searches on Internet)

World Wide Web의 등장으로 이제 우리가 손쉽게 접근할 수 있는 정보의 양이 매일 급격히 증가하고 있다. 그동안 웹검색엔진(Web search engine)이 WWW에서 매우 효과적인 정보 검색 도구로써 자리 잡아 왔다. 사실 웹검색엔진이 없는 WWW이란 상상하기 어렵다. 하지만 검색엔진만을 가지고는 정보 요구자의 검색 목적을 효율적으로 달성하기 어렵다. 하루가 다르게 폭발하고 있는 인터넷 상의 정보의 양과 자유 분방한 WWW의 성격을 볼 때 분명히 새로운 검색 도구의 필요성이 대두되고 있다. 최근 인터넷에서의 정보 탐색을 돕기 위한 여러 가지 아이디어가 제안되고 있고 시스템이 발표되고 있다.

인터넷에서도 특히 WWW의 웹(HTML)문

서에 관심을 가지는 이유는 WWW의 HTML 문서가 오늘날 우리가 인터넷에서 가장 많이 참조하는 정보원(information source)이라는 데 있다. WWW은 그야말로 인터넷의 모든 컴퓨터와 사용자들을 하나의 거대한 디지털 도서관으로 묶어 놓았다. 사용자들은 단지 마우스의 포인팅(pointing)과 클릭킹(clicking)만으로 세계 어디에 있는 정보라도 손쉽게 액세스(access)할 수 있게 되었다. 그리고 더욱 중요한 사실은 WWW에 정보를 올리는 일은 문서 작성기로 문서를 만드는 일만큼 쉬워진 관계로 모든 WWW사용자가 동시에 정보제공자가 되었다는 것이다. 따라서 이제 WWW은 디지털 정보 시대에 주요 정보원으로 자리를 잡았다고 할 수 있다.

웹검색엔진의 유용성은 현재 운영 중인 많은 검색엔진들의 인기로 이미 검증을 받은 상태에 있다. 하지만 인덱스의 불완전성, 부정확성으로 인해 그리고 사용자의 부정확한 질의 및 검색 옵션 사용법의 미숙으로 번번히 검색에 실패하는 경우가 많다. 이러한 경우 찾는 정보가 정말 없는 것인지 아니면 찾지를 못하는 것인지 사용자는 판단할 수 없다. 정보검색에서 해결하기 어려운 문제 중의 하나인 색인어와 질의어의 불일치 문제도 웹검색엔진의 한계이다. 따라서 검색엔진만으로는 원하는 정보를 쉽게 찾는 데는 한계가 있다.

우리는 웹검색엔진과는 별도로 WWW에서의 정보검색을 도울 수 있는 새로운 도구에 관심이 있다. 최근 이 분야에 활발한 연구가 진행 중에 있고 시험적인 정보 탐색 시스템이 발표되고 있다. 이러한 때 최근 발표되고 있는 정보 탐색을 돕기 위한 도구들을 정리해 보는 것도 의미 있는 일이 아닌가 생각한다. 우리는 이러한 새로운 탐색 도구들이 어떻게 정보 요구자의 탐색 목적을 만족시킬 수 있게 도울 수 있는가에 초점을 맞춘다. 시스템 구현상의 차이는 별로 중요하지 않다. 그래서 어떤 역할을 수행하는가 또는 사용자에 어떤 서비스를 제공하는가 하는 관점에서 시스템을 분류하면 다음 4가지 유형으로 나눌 수 있다.

- 모니터링 시스템 (Monitoring System)
- 필터링 시스템 (Filtering System)
- 브라우징 보조 시스템 (Browsing Assistant System)
- 추천 시스템 (Recommending System)

최근의 시스템들이 정확하게 위의 4가지 형태 중 한가지로 분류될 수는 없지만 한 시스템의 성격은 이러한 4가지 시스템의 조합으로 이루어 진다고 볼 수 있다. 그리고 위의 4가지 형태가 엄밀하게 서로 직교하는(orthogonal) 개념은 아니다. 추천 시스템(Recommending System)에 관한 가장 최근의 조사(survey)는 [Resnick97]에 잘 정리되어 있다.

3.1. 모니터링 시스템 (Monitoring System)

WWW은 매우 동적인 디지털 도서관이라고 할 수 있다. 즉 정보의 내용이 계속해서 변경되며 웹 사이트 자체가 없어지기도 하고 주소(URL)가 변경되기도 하며 새로운 웹 사이트가 만들어지기도 한다. 이러한 변경 사항은 때때로 정보 검색자에게 매우 중요할 수 있다. 하지만 현재로서는 이 거대하고 자유분방한 WWW 전체를 모니터링(monitoring)하는 일은 거의 불가능하다. WWW전체를 모니터링하는 것은 어렵지만 WWW의 일부만을 모니터링하는 일은 가능할 수 있다. 전체 WWW의 어느 부분을 어떻게 제한하느냐 하는 데는 여러 가지 방법이 있을 수 있다.

현재 가장 많이 제안된 모니터링 시스템은 정보검색자가 미리 지정해둔 URL이 가르치는 자료의 갱신 여부를 주기적으로 모니터링하고 있다가 갱신이 발견되면 이를 사용자에게 알려주는 시스템이다 [조강래97] [Smart Bookmarks] [Web Buddy]. 지속적인 관심의 대상이 되는 웹 사이트를 정기적으로 방문해서 갱신된 내용이 있는지 일일이 확인하는 일은 정보 검색자에게 많은 부담이 된다. 따라서 이 일을 사람 대신 해줄 수 있는 에이전트가 있다면 매우 편리할 것이다. 모니터링 시스템은 미리 사용자가 지정해둔 URL 리스트를 가지고 있다가 정기적으로 그 리스트의 URL들이 가리키는 웹 사이트를 방문하여 정보의 갱신 여부를 점검해서 변경된 내용이 있으면 정보 요구자에게 알리게 된다.

상용화되어 있는 시스템을 살펴보면 먼저 Netscape 네비게이터의 What's New가 가장 대표적인 URL 모니터링 에이전트이다. What's New는 책갈피(*Bookmark*)의 URL들의 갱신 여부를 모니터링하는 기능을 한다. [Web Buddy]는 사용자가 지정한 URL만 모니터링 할 수 도 있지만 지정한 URL에서 링크를 추출하여 일정한 깊이까지 탐색하여 모니터링할 수 도 있다. 시스템의 형태는 기존의 웹브라우져에 플러그인(plug-in) 형태로 기능을 추가하기도 하고 [Smart Bookmarks], 웹브라우저와 독립적으로 동작하게 할 수도 있다 [Web Buddy].

모니터링 에이전트가 자료의 갱신 여부를 어떻게 판단할 지를 결정해야 하는데 여러 가지 방법을 생각할 수 있다. 변경 전과 변경 후의 파일의 크기, 최종 갱신 날짜, 비교하는 방법이 있고 실제 문서를 추출하여 비교해보는 방법이 있을 수 있다. 실제 문서를 비교하는 방법이 가장 확실한 방법이지만 변경 전의 문서를 보관하고 있어야 하는 부담 때문에 현실적으로 선택하기 어려운 대안이다.

인트라넷의 경우는 기업 단위의 WWW으로 구성되므로 그 크기가 다룰 수 있는 범위 내에 있다. 따라서 모니터링 범위를 기업 내 인트라넷에 국한시키면 가용한 자원 내에서 모니터링 서비스의 구현이 가능하다. 대개 기업 내에서 일어나는 일은 주요 관심의 대상이 될 확률이 높으므로 인트라넷에서의 WWW 모니터링은 효과적일 수 있다.

3.2. 필터링 시스템 (Filtering System)

우리는 바야흐로 엄청난 정보의 홍수 속에 살고 있다. 매일 수많은 정보가 새로이 디지털 형태로 공급되고 있다. 이러한 정보들은 대표적으로 Newswire, Email, BBS, Usenet News, 웹 페이지, 등의 형태로 제공된다. 이 많은 정보들을 일일이 조사하여 필요한 정보를 추리는 일은 이제 거의 불가능하다. 이제는 모든 사람이 개인 비서가 있어 중요한 정보만을 취사선택하여 제공해주는 것이 필요하게 되었다. 사실 개인 비서가 여러 명이 있어도 이일을 해 낼 수 있을지 의심스럽다.

매일 계속적으로 들어오는 정보들을 걸러서 사용자에게 필요한 정보만을 선택적으로 전달해 줄 수 있는 필터링 시스템이 필요하다. 이때 개인 비서가 주인이 어떠한 정보들을 필요로 하는 지를 알고 있듯이 필터링 시스템은 사용자의 관심 모델(*user interest model*)을 가지고 있어서 어떤 정보가 적합한지 아닌지를 판단하게 된다. 사용자의 관심 모델은 프로파일(profile)이라고도 하는데 사용자의 장기적인 관심을 표현하는 계속적으로 평가되어야 하는 질의를 표현한 것이라고 할 수 있다. 정보검색(information retrieval)에서는 문서 집합에 대해 일회성의 질의가 일어나지만 정보필터링(information filtering)에서는 반대로 유입되는 하나의 문서에 대해 많은 사용자들의 질의(profile)가 일어난다고 볼 수 있다 [Belkin 92].

필터링의 대상이 되는 정보 원(information source)은 계속적으로 사용자에게 유입되는 형태이다. 예를 들면 Email, Newswire, 등이 여기에 속한다. 또 한가지 중요한 필터링의 대상이 될 수 있는 정보 원의 조건은 정보 유입 창구의 집중성이다. WWW의 경우는 정보의 유입 창구가 분산되어 있고 어디로 유입되는지에 대한 정보도 없으며 일일이 확인하기에는 인터넷의 크기가 너무 크다는 문제점이 있다. 따라서 그동안 인터넷에서의 필터링 시스템은 주로 유입 창구가 정해져 있는 Usenet News, Email, 등을 정보 원으로 하는 경우가 대부분이었다 [Yan 95] [Sheth 93] [Sheth 94] [Resnick 94] [Lang 95].

Build Time	The profile index structure is built. Other auxiliary data structures are allocated and initialized.
Filtering Time	Documents are run against the index one by one. Document-profile matchings are written into a file.
Sorting Time	The document-profile matching file is sorted by user email

	address and subscription identifier, using Unix sort command.
Notify Time	The sorted matching file is read. Excerpts of matchings for each subscription are prepared into a email message. Unix sendmail is invoked to sent out each message.

<표 3.2> SIFT 처리 시간의 4 가지 단계

그리고 필터링 시스템의 최근에 부각되고 있는 가장 큰 문제점(issues)은 성능 문제(performance issue)이다. 규모가 큰 시스템은 매우 짧은 시간 간격을 가지고 계속적으로 유입되는 문서를 많은 사용자 프로파일과 비교하여야 한다. SIFT(Stanford Information Filtering Tool)은 Stanford 대학에서 개발된 Usenet News를 대상으로 한 정보산포시스템(information dissemination system)이다. 이 시스템의 성능 연구 결과 SIFT 처리 시간을 <표 3.2>와 같이 4 가지 단계로 나누었을 때 전체 처리 시간의 약 63%정도가 사용자에게 결과를 전자 우편으로 알려주는데 사용되고 있다. *Notify Time*을 줄이기 위해서는 보다 정교한 문서 전달 스킴(document delivery scheme)이 필요하다. 이 문제를 해결하기 위해 [Yan 94a]는 지리적으로 사용자들을 그룹화 시켜 네트워크 트래픽(network traffic)을 감소시키는 방법을 제안하고 있다.

*Filtering Time*도 전체 처리 시간(processing time)의 많은 부분을 차지하고 있고 실제로 전체 시스템의 성능을 좌우하는 부분이다. 기존의 IR(Information Retrieval)과 다르게 IF(Information Filtering)에서는 질의(IR 에서는 문서)는 매우 길고 문서(IR 에서는 질의)들은 매우 짧다. 많은 짧은 문서들에 대해 매우 긴 질의를 할 때는 기존의 IR 시스템의 가정과 틀리므로 기존의 색인(indexing) 방법과 질의 방법에 무언가 변화가 있어야 한다는 점을 인식할 수 있다. SIFT 에서는 이와 같이 IF 가 IR의 동전의 양면과 같은 문제(dual problem)라는 것을 인식하고 문서 색인(document index)대신에 프로파일 색인(profile index)를 만들고 이를 위한 다양한 색인 기법을 제안하고 있다 [Yan 94b] [Yan 94c].

WWW, 특히 HTML 문서, 에 대해 필터링을 직접 적용하기는 쉽지 않은 것 같다. 먼저 WWW을 대상으로 한 어떤 처리 결과로 나오는 결과물에 필터링을 적용하는 방법이 가능하다. 예를 들면 검색 결과로 얻은 URL 들을 필터링할 수 있다. 그리고 모니터링 시스템에서 사용자에 정보의 갱신 사항을 무조건 알려줄 것이 아니라 갱신 내용이 사용자의 관심을 가질 만한 내용인지를 판단해서 선별적으로 알려주는 방법을 생각해 볼 수 있다.

WWW 에서 정보 필터링(information filtering)라는 제목으로 최근에 발표된 연구들을 보면 기본적인 아이디어는 책갈피(bookmark)의 기능을 확장하는 방향을 취하고 있다 [Pazzani 95][Falk 96]. 각 관심 주제별로 책갈피 목록(bookmark list)를 유지하고 이들 책갈피를 이용하여 사용자 관심 모델(user interest model)을 학습시킨다. 일단 충분한 학습이 이루어지면 사용자 관심 모델을 기반으로 책갈피에 있는 정보들과 비슷한 정보들을 찾고, 추천하고, 책갈피에 추가한다 [Falk 96]. WWW 정보 필터링은 결국 WWW을 브라우징(browsing)하는 과정에서 일어나야 하므로 다음 절에서 설명할 브라우징 시스템과 개념에서는 공유하는 부분이 많다. 우리가 관심을 가지고 있는 WWW 정보필터링에 대해서는 앞으로 4 장에서 보다 자세하게 알아 본다.

3.3. 브라우징 보조 시스템 (Browsing Assistant System)

WWW 에서 정보를 검색하는 행태는 크게 2 가지로 나눌 수 있다. 원하는 정보가 위치해 있는 URL을 알고 있을 때 바로 그 URL로 바로 가는 방법과 원하는 정보가 어디에 있는지 모를 때 검색 엔진을 사용해 후보 URL 들의 리스트를 얻고 하나하나 확인하는 방법이다. 2 가지 경우 모두 더 이상의 검색 활동(activity) 없이 바로 검색 목적을 달성하는 경우는 드물다. 현재의 URL 에서 검색의 목적이 달성되지 못했을 경우 하이퍼링크(hyperlink)를

따라가며 브라우징(browsing)이라는 형태로 2차 검색을 하여야 하는 경우가 많다. 현재의 URL에서 검색의 목적이 달성되었더라도 새로운 검색 목적을 가지고 추가적인 검색이 일어날 수 있다.

브라우징이란 검색자가 직접 HTML 문서의 하이퍼링크를 따라감으로써 WWW 공간을 탐색하는 것이라고 정의할 때 뚜렷한 검색 목적을 가지고 브라우징을 할 수 도 있고 뚜렷한 검색 목적은 없지만 관심을 끌만한 정보가 혹시 있을지도 모른다는 기대에서 브라우징을 하는 경우도 있다. (전자의 경우를 적극적인 브라우징, 후자의 경우를 소극적인 브라우징이라고 하자.) 위 두 가지 형태의 브라우징 모두 현재 인터넷의 늦은 속도를 고려할 때 매우 시간 소모적인 일이 될 수 있다. 따라서 사용자가 효과적으로 브라우징할 수 있도록 보조해주는 소프트웨어 에이전트가 있다면 검색자에게는 매우 편리할 것이다. 하지만 어떻게 사용자의 효과적인 브라우징을 도울 수 있을지는 확실하지 않은 것 같다.

최근에 WWW 브라우징을 도와주는 브라우징 보조 에이전트에 대한 적지 않은 연구가 있어 왔다 [Joachims 96] [Lieberman 95]. 대부분의 브라우징 보조 에이전트는 사용자가 미답의 웹공간에서 방향을 잃지 않고 검색목적을 달성할 수 있도록 다음에 방문할 바람직한 링크를 제시하는 방식으로 사용자의 브라우징을 도운다. 사용자가 현재의 URL의 내용을 검토하고 있는 동안 브라우징 보조 에이전트는 백그라운드에서 현재 문서의 하이퍼링크를 탐색하게 된다. 에이전트는 사용자와 독립적으로 그리고 동시 병렬적으로 움직이게 된다. 따라서 사용자는 에이전트의 안내를 받아들일 수도 에이전트를 전혀 무시할 수도 있다.

에이전트가 어떤 HTML 문서가 적합한 문서인지 아닌지, 즉 사용자가 관심을 가지는 정보를 포함하고 있는지 아닌지를, 판단하기 위해서는 필터링 시스템과 같이 사용자의 프로파일(profile)을 가지고 있어야 하는데 사용자 관심 모델을 어떻게 구축하고 유지하는가가 매우 중요하다. 미리 사전에 정의해둘 수도 있지만 사용자의 관심이 장기적인 것일 지라도 계속 변화하기 마련이고 일회성의 검색의 경우 브라우징할 때마다 검색 목적이 다를 수 있다. 따라서 사용자 관심 모델을 자동으로 구축하고 상황에 따라서 자동적으로 적응시키는 것이 바람직하다. 사용자의 브라우징 행태를 관찰함으로써 어떤 정보에 관심이 있는지를 대강 추측할 수가 있다. 그리고 브라우징을 시작할 때 검색 목적을 지정 받거나 추천한 URL에 대해 적합성 여부에 대한 피드백(feedback) 등의 형태로 사용자로부터 도움을 받을 수 있다 [Joachims 96]. 사용자로부터 어떠한 개입도 없이 완전히 자율적으로 사용자의 브라우징 행태로부터의 단서 그리고 여러 가지 휴리스틱을 이용하여 사용자의 프로파일을 학습시키기도 한다 [Lieberman 95].

제한된 자원과 시간 내에 가능한 하이퍼링크를 모두 탐색하는 것은 불가능하다. 따라서 어떤 링크를 선호하여 먼저 탐색할 것인가를 결정하는 탐색 전략이 매우 중요하다. 사용자는 대개 depth-first 식으로 브라우징하는 경향이 있다. 어느 정도 깊이로 내려 갔다가 더 이상의 탐색이 필요 없다고 판단되면 다시 백트랙(backtrack)해서 depth-first 탐색을 계속한다. 하지만 HTML 문서의 구조상 원하는 정보는 현재의 위치로부터 얕게 위치해 있을 확률이 높다. 따라서 breadth-first 탐색이 보다 적합하며 신빙성 있는 휴리스틱에 기반한 best-first 탐색이면 더욱 바람직할 것이다 [Lieberman 95].

3.4. 추천 시스템 (Recommending System)

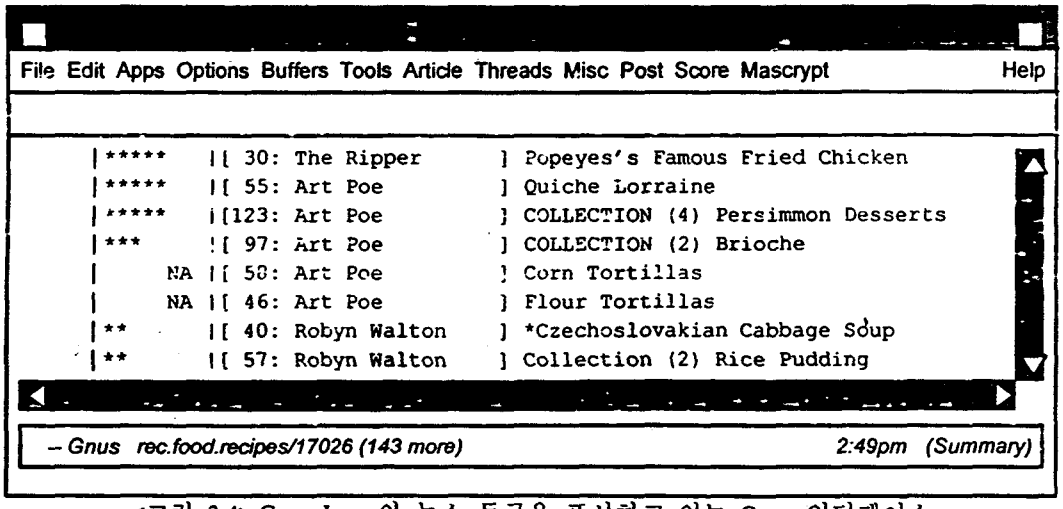
종종 우리는 선택 대안에 대한 충분한 경험이 없이 어떤 선택을 하여야 하는 경우가 있다. 이러한 경우 가능하다면 다른 사람의 경험을 참고하는 것이 바람직하다. 우리는 실제로 매일의 일상 생활에서 신문의 책 리뷰(book review), 영화 평론, 식당 가이드, 등의 형태로 다른 사람의 추천(recommendation)을 참고하고 있다.

인터넷은 거대한 정보의 창고이고 매일 수많은 사람들에 의한 검색이 일어나고 있다. 자기와 관심 분야가 비슷한 사용자가 많이 있

을 것이고 또한 일회성 검색의 경우도 누군가에 의해서 이미 유사한 검색이 일어났을 가능성이 높다. 검색의 결과에 대한 적합성 여부는 검색자의 개인적인 판단으로 끝나며 여기에 대한 정보는 즉시 사라져 버린다. 만일 이러한 개인적인 의견(annotation)을 잘 기록해 둔다면 다른 검색자가 같은 정보를 찾을 때 실제로 문서의 내용을 확인하지 않고서도 문서의 적합성 여부를 판단할 수 있을 것이다.

최초의 추천 시스템(recommending system)인 Tapestry[Goldberg 92]의 개발자가 처음으로 "협동적 필터링(collaborative filtering)"이라는 말을 처음으로 사용했는데 "협동적(collaborative)"라는 단어는 다른 사람의 추천에 의존한다는 것을 강조한 것이다. 이 글에서 우리가 추천 시스템이라 하면 "협동적"을 전제로 한 것이다. 추천을 협동적 추천(collaborative recommendation)과 내용 기반의 추천(content-based recommendation)으로 구분하기도 하는데 [Balabanovic 97] 내용 기반의 추천은 사용자 자신이 과거에 선호하였던 것과 비슷한 것을 추천하는 것을 말한다.

추천 시스템의 가장 적합한 적용 대상 애플리케이션 중의 하나는 Usenet News이다. 현재 수많은 뉴스 그룹(News Group)이 있으며 각 뉴스 그룹마다 매일 수많은 뉴스(news)가 포스팅(posting)된다. 검색 대상을 몇 개의 뉴스 그룹으로 제한하더라도 많은 뉴스들을 일일이 읽고서 원하는 정보를 찾는 일은 대우 많은 시간을 소비해야 하는 작업이다. GroupLens[Konstan 97]는 각 뉴스의 제목 옆에 1부터 5까지의 추천 등급(recommendation rating)을 제공한다 <그림 3.4>. 등급(rating)이 5이면 매우 유용한 정보이고 등급이 1이면 전혀 읽을 가치가 없는 뉴스임을 표시한다. 사용자는 등급이 높은 뉴스들만을 읽음으로써 시간을 절약할 수 있다.



<그림 3.4> GroupLens의 뉴스 등급을 표시하고 있는 Gnus 인터페이스

추천 시스템의 가장 심각한 문제 중의 하나는 등급 희소성(ratings sparsity) 문제이다 [Konstan 97]. 추천 시스템의 사용에 무임승차하기 위해서는 누군가에 의해서는 상당량의 정보 아이템(items)에 대해서 등급이 매겨져야만 한다. 누가 보수 없이 이러한 초기 투자를 감수하겠는가 하는 것이 문제이다. 사용자들은 한번의 키스트로크(keystroke)으로 끝나는 등급주기에도 매우 인색하다. 이유는 보편적으로 사람들이 적절한 등급을 결정하는 것에 대해 생각하는 것 조차도 귀찮아 하기 때문이다. 등급 희소성(ratings sparsity) 문제는 애플리케이션의 성격에 따라 정도의 차이는 있겠지만 추천 시스템이 필연적으로 직면해야 하는 문제이다. 사용자에 등급매기기에 대한 어떤 보수(reward)를 주는 것도 하나의 방법이 될

수 있지만 이상적인 해결책은 사용자의 행태를 관찰함으로써 간접적인(implicit) 등급을 획득할 수 있도록 사용자 인터페이스를 개선하는 방법이다. 또 다른 방법은 컴퓨터로 하여금 사람의 읽기(reading)이라는 고도의 지적인 프로세스를 대신할 수 있게 하는 것이다 [Konstan 97].

4. WWW 정보 필터링 (WWW Information Filtering)

현재 WWW에서 정보검색의 효과적인 유일한 도구는 웹검색엔진이라고 할 수 있다. 그리고 야후[Yahoo] 등에서 제공하는 사람의 수작업에 의한 주제별 분류 서비스가 보조적으로 사용될 수 있다. 웹검색엔진이 매우 유용한 도구이기는 하나 웹검색엔진만으로는 정보검색의 목적을 완벽하게 달성하기는 어렵다. 웹검색엔진은 기존의 정보 검색 시스템의 문제점을 그대로 안고 있을 뿐만 아니라 WWW 정보 공간(information space)의 특수한 상황이 더욱 문제를 악화시키고 있다.

WWW의 특수한 성격 때문에 생기는 웹검색엔진의 문제점은 인덱스의 불완전성이다. WWW은 통제가 불가능한 분산 시스템이고 정보의 내용이 계속 변하기 때문에 인덱스를 완벽하게 구축하는 것은 근본적으로 불가능하다. 정보검색시스템의 해결되지 못한 어려움인 사용자 질의와 문서 색인 언어의 불일치 문제가 있기 때문에 검색의 한계가 있을 수 밖에 없다. 또한 정보 검색 전문가가 아닌 이상 질의의 부정확성은 피할 수 없다.

이러한 웹검색엔진의 한계는 자연스럽게 보다 효과적인 WWW 검색 도구를 요구하고 있다.

4.1. 문서의 표현 (Document Representation)

WWW 상에는 많은 종류의 웹오브젝트(Web object)들이 있다. 그 중에서도 가장 중요한 웹오브젝트는 HTML 문서이다. 왜냐하면 HTML 문서는 WWW의 골격을 이루는 웹오브젝트이기 때문이다. 정확한 통계는 없지만 현재 WWW에서 HTML 문서가 전달하는 정보의 양은 다른 오브젝트들에 비해 압도적일 것이다.

HTML 문서가 담고 있는 정보를 처리하기 위해서는 시스템이 처리할 수 있게 시스템 내부 표현으로 변환되어야 한다. HTML 문서의 내용이 자연언어로 기술되어 있으므로 HTML 문서의 시스템 내부 표현 문제는 정보검색 분야에서 가장 중요한 문제 중의 하나인 문서 표현(document representation)의 문제와 동일하다.

대부분의 WWW 정보필터링시스템이 가중 키워드 벡터(weighted keyword vector) [Salton 87]를 사용하고 있다. 이유는 현재 정보검색 분야에서 가장 많이 쓰이는 문서 표현이기도 하고 간단하다는 큰 장점이 있다. 그리고 기계학습을 적용하기 위해서는 문서의 정보가 특징 벡터(feature vector)로 표현되어야 하는데 가중 키워드 벡터는 바로 특징 벡터로 사용될 수 있다.

또 다른 이유로는 대부분의 시스템 개발자의 학문적 배경이 기계학습(machine learning)이나 소프트웨어 에이전트(software agent)등이기 때문에 더욱 복잡한 자연언어처리 기법을 사용한 문서 표현을 시도하기에는 한계가 있었을 것으로 추측된다. 하지만 최근에는 자연언어처리를 통한 더욱 적절한 문서 표현을 정보필터링 에이전트에 적용하려는 시도도 있다 [Anikina 96].

4.2. 사용자 관심 모델링 (User Interest Modeling)

정보 필터링(information filtering)는 사용자의 정보 요구에 대해 유입되는 정보들 중에 관련 정보는 사용자에 제공하고 비관련 정보는 버리는 과정(process)이라고 할 수 있다. 어떤 정보가 사용자가 관심을 가지는 것인지 아닌지를 판단할 수 있는 근거가 되는 것이 필터링 에이전트(filtering agent)가 내부에 가지고 있는 사용자 관심 모델(user interest model)이다.

사용자의 관심 모델을 만드는 일은 쉽지 않다. 왜냐하면 사용자들 스스로가 그러한 관

심들이 무엇인지 구체적으로 표현하는 것이 쉽지 않기 때문이다. 사용자 관심 모델을 사용자로부터 직접적으로 얻기 힘들 경우 이를 간접적으로 획득할 수 있는 방법이 필요하다. 사용자의 행태를 관찰함으로써 사용자가 무엇에 관심이 있는지 추측하게 하는 것이 한가지 방법이다. 그리고 관심이란 계속 변화하기 마련이다. 사용자의 관심이 시간의 경과나 환경의 변화에 따라 변화하기 때문에 관심 모델은 이에 대처할 수 있게 스스로를 변화시켜 관심의 변화에 적응할 수 있는 능력이 필요하다.

또 관심(interest)은 도메인(domain)에대한 믿음, 정보 탐색 목적(information goals), 정보의 유형(information types), 정보의 성격과 특징(information characteristics), 등과 관련이 있을 수도 있다 [Stadnyk 92].

어떤 문서가 사용자의 관심에 부합하는지 또는 하지 않는지 결정하는 가장 간단한 방법은 키워드 매칭(keyword matching) 방법이다. 사용자의 관심이 어떤 단어들에 의해 기술되며 그러한 단어들이 포함된 문서는 사용자의 관심에 부합하는 문서라고 판단하게 된다. 이 방법은 초기에 많이 사용된 간단한 방법인데 단어 자체가 어떤 개념이나 토픽을 표현하는데 가지는 한계 때문에 문서 선택에 모호성이 증가하는 단점이 있다.

단순한 키워드 매칭을 개선할 수 있는 방법이 대표적인 정보검색 모델인 부울린 모델(Boolean model)과 벡터 공간 모델(vector space model) [Salton 89]을 이용하는 것이다.

벡터 공간 모델(vector space model)에서 질의는 사용자가 지정하는 키워드(keyword)들의 벡터로 표현된다. 질의가 m 개의 단어로 구성되고 각 키워드 t_i의 가중치가 w_i라고 하면 질의는 m-차원 벡터 $Q = <w_1, w_2, \ldots, w_m>$로 표현된다. 마찬가지로 문서도, 문서에 n 개의 키워드가 있다면, 벡터 $D = <w_1, w_2, \ldots, w_n>$으로 표현된다. 이제 사용자 프로파일 (질의)과 문서 사이의 유사 정도는 두 벡터 사이의 거리 (dot product, 등)로 계산될 수 있다. 사용자 프로파일을 구성하는 키워드들과 가중치는 사용자가 미리 적합하다고 주석을 달아둔 문서 집합에서 통계적인 방법으로 자동으로 추출될 수 있다. 키워드 프로파일은 모델 구축이 간단하고 기계학습 알고리즘을 적용하기에 용이하다는 장점 때문에 많이 사용되고 있다. 벡터공간 프로파일의 학습에는 주로 사용자의 적합성 피드백(relevance feedback)을 많이 사용한다.

부울린 모델에서는 사용자가 원하는 키워드와 원하지 않는 단어들을 부울린 연산자로 표현한다. 예를 들면 부울린 프로파일(Boolean profile) (정보 and 검색 and (not 문서))은 "정보"와 "검색"은 반드시 문서에 포함되어 있어야 하고 "문서"는 포함되지 않아야 한다는 것을 의미한다. 부울린 프로파일은 키워드 벡터에 비해 표현력이 높아 사용자의 관심을 보다 구체적으로 표현할 수 있는 장점이 있다. SIFT[Yan 95]에서는 키워드 프로파일과 부울린 프로파일을 같이 사용하여 두 가지의 장점을 모두 활용하고 있다.

또 한 가지 단순 키워드 프로파일을 개선할 수 있는 방법이 LSI(Latent Semantic Indexing) 방법이다. LSI 에서는 전체 문서 집합에 걸친 단어의 사용 패턴에 밑바닥에 어떤 보이지 않는(Latent) 구조가 있다는 가정에서 출발한다. 그리고 통계적인 방법으로 이러한 숨은 구조(Latent Structure)를 계산해 낼 수 있다는 것이다 [Deerwester 90]. 따라서 정보검색에서 단순한 단어를 사용하기 보다는 의미적 숨은 구조에 기반해서 사용자의 질의나 문서를 표현할 수 있다는 것이다. [Foltz 92]는 Technical Memo를 필터링하는데 LSI 방법과 벡터 공간 모델을 비교하였다.

Information Lens 시스템에서는 사용자가 Email 메시지를 필터링하기 위한 룰(rule)을 만들 수 있게 되어 있다. Email 메시지는 준구조적인 텍스트로 보낸 사람에 대한 정보라던가 키워드 필드의 키워드들과 같은 구조 정보를 가지고 있다. 따라서 룰을 만드는데 이러한 구조 정보가 유용하게 사용될 수 있고 컴퓨터에 대한 지식이 많이 없는 사용자도 필터링 룰을 쉽게 만들 수 있음을 발견하였다. 그리고 사용자들은 메시지 제목이나 본문보다는 보낸사람, 다른 받는사람, 등의 정보를 더 많이 사용하는 것으로 나타났다.

우리는 본 절에서 사용자들의 관심을 어

떻게 기술할 수 있을 것인가에 대해 관련 연구들을 알아보려고 하였다. 대부분의 방법이 정보검색에서 빌려온 방법이었고 어떤 방법이 사용자의 관심을 가장 효과적으로 표현할 수 있는지에 대한 체계적인 연구가 아직 미흡한 것 같다. 사용자 프로파일의 표현은 필터링 대상이 되는 정보의 유형에 의존적일 수 밖에 없고 그리고 프로파일의 학습 알고리즘에도 의존할 것이라는 것을 추측할 수 있다. 사용자의 관심을 정확하게 기술하는 것도 중요하지만 정보필터링 에이전트가 사용자 프로파일을 스스로 학습하여 사용자의 관심의 변화에 적응하게 하는 것도 매우 중요하다. 사용자 프로파일 학습에 대해서는 다음 장에서 살펴본다.

4.3. 적응형 인터페이스 에이전트: 학습 방법 (Adaptive Interface Agents: Learning Methods)

정보필터링 에이전트는 적응형 인터페이스 에이전트(adaptive interface agent)이다. 인터페이스 에이전트(interface agent)란 인공지능 기법을 이용하여 사용자가 특정 응용 프로그램을 사용하는데 있어서 도움을 주는 프로그램이다 [Maes 94]. 적응형 에이전트(adaptive agent)는 과거의 경험을 참고하여 사용자의 선호(preference)에 따라 그리고 환경의 변화에 따라 스스로를 자동적으로 적응해가는 에이전트이다 [Etzioni 95].

정보필터링 에이전트는 사용자의 프로파일을 가지고 있어 사용자가 관심을 가질만한 문서만을 필터링해서 제공하는 에이전트이다. 프로파일이란 에이전트가 내부에 가지고 있는 사용자의 관심 모델(interest model)이라고 할 수 있다. 사용자의 관심은 시간과 장소에 따라 변화할 수 있다. 따라서 관심 모델은 적응성을 가져야만 한다. 적응형 에이전트(adaptive agents)는 스스로 사용자의 관심 모델을 형성하고 사용자의 관심의 변화에 따라 모델을 변화시킬 수 있어야 한다.

4.3.1. 학습을 위한 지식 원 (Knowledge Sources for Learning)

인터페이스 에이전트(interface agent)는 4 가지 다른 소스(sources)에서 학습에 필요한 지식을 획득한다 [Maes 94].

① 사용자의 행위의 관찰을 통한 학습
② 사용자의 피드백(feedback)을 통한 학습
③ 훈련을 통한 학습
④ 다른 에이전트의 충고를 통한 학습

먼저 인터페이스 에이전트는 사용자가 어떤 일을 하는 동안 계속적으로 사용자의 행위를 어깨너머로 관찰함으로써 배운다. 인터페이스 에이전트는 사용자의 활동을 모니터하고, 장기간(몇 주 혹은 몇 달)에 걸쳐 사용자의 행위를 관찰함으로써 반복적인 패턴과 규칙성을 발견한다.

두 번째 소스는 직접적 또는 간접적 사용자의 피드백(feedback)이다. 간접적인 피드백은 에이전트가 제의하는 행위를 사용자가 무시하고 다른 행위를 취할 때 일어난다. 이것은 사용자가 유입되는 메시지를 읽는 순서를 바꾼다든지 에이전트에 의해서 제시된 기사를 읽지 않고 다른 기사를 읽는 다든지 하는 식으로 매우 미묘한 성격을 띠게 된다. 사용자는 또한 에이전트의 행위에 대해 명확하게 부정적인 피드백을 줄 수 있다 (예를 들면 "나는 이 기사를 좋아하지 않는다" 또는 "다시는 이러한 행위를 하지 마라", 등).

세 번째는 에이전트가 사용자에 의해서 구체적으로 주어진 예로부터 학습하는 방법이다. 사용자는 사건(events)이나 상황(situations)들의 가상적인 예를 주고 그러한 경우에는 무엇을 해라라고 말함으로써 에이전트를 훈련시킬 수 있다. 인터페이스 에이전트는 사용자의 행위를 기록하고 객체(object)들 사이의 관련성을 밝혀내고 보여진 예를 흡수할 수 있도록 example base 를 변경한다.

마지막으로 인터페이스 에이전트는 같은 성격의 일을 다른 사용자들을 위해 하고 있는 (더 많은 경험을 쌓았을 지 모르는) 다른 에이전트들로부터 충고를 요청하고 받음으로써 학습한다. 에이전트가 어떤 상황에서 어떤 행동이 적절한지 스스로 알 수 없을 때 다른 에이전트에게 현재 상황을 보여주고 이 상황에서

어떤 행동을 해야 할지 충고해 달라고 질문할 수 있다.

4.3.2. 학습 방법 (Learning Methods)

사용자 프로파일의 학습이라고 할 때 두 가지 의미가 있다. 사용자 프로파일을 구체적으로 기술하는 것이 어렵기 때문에 사용자 프로파일을 자동으로 만들기 위해 필요한 학습이 있고 사용자의 관심의 변화에 적응하기 위해 사용자 프로파일을 갱신하기 위한 학습이 있다. 여기서 문제가 되는 것은 프로파일을 갱신하기 위한 학습이다. 새로운 예제를 보고 이를 흡수할 수 있도록 현재의 프로파일을 갱신해야만 한다. 이를 위해 증분 학습(incremental learning)이 가능해야만 한다. 신경망 학습에서는 과거의 학습 예제들을 모두 저장하고 있다가 새로운 예제를 더해서 다시 처음부터 학습이 일어나야만 한다. 필터링 에이전트에게는 오랜 학습 시간과 과거의 학습 예제를 모두 저장해야 함으로써 발생하는 많은 기억공간 필요는 바람직하지 않다.

그동안 에이전트의 프로파일 학습에 많이 사용된 증분 학습이 가능한 대표적인 프로파일 학습 방법으로 유전자 알고리즘(Genetic Algorithm), TFIDF (Term Frequency Inverse Document Frequency) 학습 방법, 등이 있다. 사용자 프로파일에 기계학습을 적용하기 위해서는 일단 프로파일이 기계학습에 사용되는 자질(features) 집합으로 표현되어야 한다. 대부분의 에이전트는 이 자질들로 키워드(keywords)를 사용하고 있다.

최근 유전자 알고리즘(genetic algorithm)을 사용한 시스템이 늘고 있다. 이는 여러 에이전트들이 서로 협동하에 목적을 달성하는 다중 에이전트 시스템이 여러 에이전트들이 적자생존의 법칙에 따라 움직이는 생태계(ecosystem)와 같다고 보는 시각이 때때로 편리하기 때문이다. 사용자의 선호(preference)에 잘 적응하는 에이전트는 살아남아 계속 자식을 번식하고 그렇지 못한 에이전트는 죽는다 [Moukas 96] [Sheth 93].

에이전트의 진화는 시스템 적합성(fitness)에 의해 제어된다. 최고의 성능을 발휘하는 몇몇의 소수의 에이전트만 자식을 번식하도록 허용된다. 새로운 에이전트는 교차(crossover)와 변이(mutation)에 의해 만들어진다. 이 두 가지의 조작자(operators)가 부모 에이전트의 유전자에 적용된다. 교차는 부모 에이전트의 유전자를 반반씩 가져와 새로운 개체를 만드는 것이고 변이는 임의로 일부 유전자를 변화시켜 새로운 개체를 만드는 방법이다. 교차와 변이를 키워드 벡터에 적용하면 교차는 부모의 키워드들을 반반씩 가져와 (이때 어느 부분을 가져오는 가는 무작위로 결정된다) 새로운 키워드 벡터를 만든다. 변이는 키워드 벡터를 구성하는 몇몇 키워드를 무작위로 추출된 다른 키워드들로 치환하는 것에 해당된다. 교차와 변이는 새로운 에이전트를 만들 때 동시에 작용하게 된다.

TFIDF(*Term Frequency Inverse Document Frequency*)를 이용한 방법은 정보검색의 벡터 공간 모델에서 사용자의 적합성 피드백(relevance feedback)을 이용하여 질의를 재구성하는 방법을 사용자 프로파일의 학습에 적용하는 방법이다 [Balabanovic 97]. 사용자 프로파일과 문서(웹 페이지) 모두 N 차원의 가중 키워드 벡터로서 표현된다. 이때 가중치는 TFIDF 값으로 계산된다. 문서 W 에서 단어 d_i 의 가중치 w_i 는 다음의 식으로 주어진다

$$w_i = \left(0.5 + 0.5\frac{tf(i)}{tf_{\max}}\right)\left(\log\frac{n}{df(i)}\right)$$

여기서 *tf(i)*는 문서 W 의 단어 d_i 의 출현 빈도 (*term frequency*), *df(i)*는 전체 문서 집합에서 단어 d_i 를 포함하는 문서의 수 (*document frequency*), n 은 전체 문서 집합의 문서의 수, tf_{max} 는 W 에서 모든 단어에 대한 최대 출현 빈도이다.

사용자 프로파일 벡터를 **m**, 웹 페이지 벡터를 **w** 라고 할 때, 웹 페이지 **w** 가 얼마나 사용자 프로파일 **m** 과 일치하는 가를 계산하기 위해서 두 벡터의 내적(dot product)을 취할 수 있다.

$$r(\mathbf{w},\mathbf{m}) = \mathbf{w}.\mathbf{m}$$

또한 사용자 프로파일 **m**을 갱신하는 것은 정보검색에서 적합성 피드백 [Rocchio 71]을 적용하는 것과 같고 다음의 간단한 식이 사용될 수 있다.

$$u(\mathbf{w},\mathbf{m},s) = \mathbf{m} + s\mathbf{w}$$

여기서 s는 웹 페이지 **w**에 대한 사용자의 점수(평가)이다.

WebWatcher[Armstrong 95]도 TFIDF를 사용하지만 사용자 프로파일 벡터를 만드는 방법이 약간 다르다. WebWatcher가 학습해야 할 개념이, 적합(사용자가 선택한 링크)과 부적합(사용자가 선택하지 않은 링크), 두 가지의 클래스를 가진다. 적합 예제들과 부적합 예제들로 분리하고 각 클래스의 벡터들을 모두 더함으로써 목표 개념(target concept)의 원형 벡터(prototype vector)를 만든다. 그래서 새로운 인스턴스(instance)에 대한 평가는 적합 원형 벡터와 인스턴스 벡터 사이의 코사인(cosine)값에서 부적합 원형 벡터와 인스턴스 벡터 사이의 코사인값을 뺀 값으로 계산된다.

4.4. WWW 필터링 시스템의 설계 (A Design of WWW Filtering System)

4.4.1. 시스템 구조에 영향을 미치는 몇 가지의 설계 요소 (Some Design Factors Influencing the System Architecture)

전체적인 시스템 아키텍쳐는 여러 가지 설계 고려에 따라 다양하게 구성될 수 있다. 가장 크게 시스템 아키텍쳐를 결정하는 요소들 중 몇 가지는 다음과 같다.

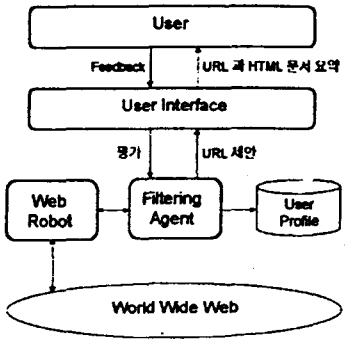
- 개인 사용자; 다수 사용자 (Single user; Multiple users)
- 하나의 프로파일; 다수 프로파일 (Single profile; Multiple profiles)
- 개인 프로파일; 공유 프로파일 (Private profile; Public profile)

개인 사용자를 대상으로 할 것인가 [Lieberman 5] 아니면 다수의 동시 사용자를 대상으로 할 것인가 [Armstrong 95]에 따라 크게 시스템 아키텍쳐가 달라질 수 있다. 다수 사용자의 경우 사용자당 에이전트를 배당할 것인가, 한 에이전트가 다수 사용자를 서비스할 것인가, 또는 에이전트 풀(pool)을 전 사용자들이 공유할 것인가, 등의 선택이 있을 수 있다. 다수의 동시 사용자를 대상으로 하는 경우는 개인 사용자를 대상으로 할 때 보다 여러 가지 추가로 고려해야 할 사항들이 많을 것이다. 시스템 모듈화, 제한된 자원의 효율적인 활용, 사용자들 사이의 안전 장치, 등에 더욱 신경을 써야 한다.

시스템 아키텍쳐는 또한 사용자당 한 개의 프로파일만 허용하는 것인가 아니면 여러 개의 프로파일을 허용하는 가에 따라 달라질 수 있다.

다수의 사용자들을 대상으로 하는 경우 사용자들간에 관심(interest)을 공유하는 경우가 있다. 에이전트는 한 가지 관심에 전문화되는데 같은 관심에 대해 사용자별로 에이전트를 소유하는 것이 비효율적일 수 있다. 따라서 다수의 사용자가 특정 관심에 전문화된 에이전트를 공유하게 할 수도 있을 것이다 [Moukas 96].

4.4.2. WWW 필터링 시스템의 구조 (WWW Filtering System Architecture)



<그림 4.4.2A> 전체 시스템 구조

다음에 개인 사용자, 다수 프로파일, 개인 프로파일에 기반한 간단한 WWW 정보필터링 시스템(information filtering system)을 원형을 제

안한다. 이 시스템 구조는 현재 개발 중인 WWW 정보필터링 시스템의 기본 구조에 기초하고 있다. 제안하는 정보필터링 시스템은 자율적인 인터페이스 에이전트(Autonomous Interface Agent) [Lieberman 96]이다. 정보필터링 에이전트는 사용자와 동시에 움직이면서 웹로봇(Web Robot)의 출력을 받아 사용자에게 필터링 결과를 사용자 인터페이스를 통해 제시하고 사용자로부터 피드백을 인터페이스를 통해 받는다. 필터링 에이전트는 사용자와 웹로봇 사이에서 지능적인 인터페이스 에이전트의 역할을 하게된다. 전체 시스템은 사용자 인터페이스, 필터링 에이전트, 웹로봇으로 구성된다 <그림 4.2.2A>

- 사용자 인터페이스 (User Interface): 사용자에게 필터링된 URL과 문서 요약을 제공하며 사용자의 피드백을 받는다.
- 필터링 에이전트 (Filtering Agent): 웹로봇에서 넘겨받은 HTML 문서와 사용자 프로파일을 비교하여 적합한 문서만을 사용자 인터페이스에 출력한다.
- 웹로봇 (Web Robot): WWW을 탐색 전략에 따라 이동하면서 HTML 문서를 필터링 에이전트에 공급한다.

가. 사용자 인터페이스 (User Interface)

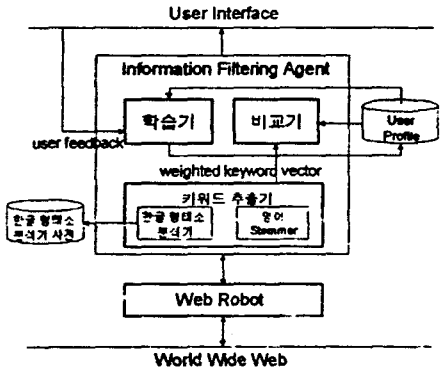
사용자 인터페이스는 필터링 에이전트로부터 필터링된 URL들을 받아서 이 URL들과 함께 URL이 가르키는 실제 HTML 문서의 제목과 요약 (처음 4-5 줄)을 사용자에 제시한다. 사용자는 제안된 각 URL에 대해 사용자의 관심에 부합하는 문서이다 또는 아니다 라고 평가를 하고 필터링 에이전트에 피드백한다.

HTML 문서의 요약을 제공하는 것은 매우 중요한데 사용자는 요약만 읽고 현재의 HTML 문서가 관심에 부합하는 문서인지 아닌지 판단할 수 있다. 요약으로 판단이 서지 않을 때는 실제 URL이 가르키는 HTML 문서를 가져와 읽어야 한다.

나. 필터링 에이전트 (Filtering Agent)

필터링 에이전트는 웹로봇으로부터 넘겨받은 HTML 문서를 내부 표현인 가중 키워드 벡터 (weighted keyword vector)를 생성하는 키워드 추출기(keyword extractor), HTML 문서의 가중 키워드 벡터와 사용자 프로파일을 비교하여 적합한 문서만을 사용자 인터페이스에 전달하는 비교기, 사용자 프로파일을 사용자의 피드백에 따라 적응시키는 학습기, 등으로 구성된다 <그림 4.4.2B>.

HTML 문서에서 HTML 태그를 제거하고 ASCII 텍스트로 변환하기 위해 HTML 파서(parser)가 필요하다. 그리고 한글 문서에서 키워드(명사)를 추출하기 위해서 한글 형태소 분석기가 필요하며 영어 문서는 어근 추출(stemming)를 위해 간단한 영어 어근 추출기(English stemmer)가 필요하다.



<그림 4.4.2B> 필터링 에이전트의 구조

다. 웹로봇 (Web Robot)

검색엔진에 사용되는 웹로봇과 하는 역할에서는 비슷하다. World Wide Web을 탐색 전략에 따라 이동하면서 HTML 문서를 가져와서 필터링 에이전트에 넘겨주는 일을 수행한다. 그리고 WWW 필터링 시스템의 WWW 인터페이스를 담당한다

5. 결론

웹검색엔진은 WWW을 가능하게 하는 가장

중요한 기술중의 하나이다. 그동안 인터넷(특히 World Wide Web)에서 정보 검색은 웹검색엔진에 전적으로 의존해 왔다고 할 수 있다. 인터넷이 WWW의 포함하는 공간이기는 하나 최근에는 모든 정보가 WWW을 위주로 분배, 공유되고 있는 실정이다.

정보 검색은 사용자의 구체적인 정보 요구에 의해서 이루어 지고 일회성으로 끝나게 된다. 하지만 최근의 상황은 개인의 정보 욕구를 만족시키기에는 종래의 정보 검색 방법은 너무나 미약한 방법이 되었다. 매일 수많은 정보들이 직접적 간접적으로 쏟아지고 있다. 개인에게 직접적으로 들어오는 정보들에 대해서는 정보 필터링 기술을 사용하여 필요한 정보들만을 골라내야 하고 간접적으로 인터넷 어딘가에 발생하는 정보는 자기에게 꼭 필요한 정보라면 찾아내어 가져올 수 있는 메커니즘(mechanism)이 필요하다.

최근에 기존의 웹검색엔진을 개선, 보조하거나 전혀 새로운 개념의 정보 검색 도구들이 활발히 제안되고 있다. 그 중에서 WWW 정보필터링 에이전트는 WWW상에서 발생하는 정보들을 적극적으로 찾아 다니면서 사용자의 장기적인 관심에 부합되는 정보들만을 필터링하여 사용사에게 제공하는 에이전트이다. WWW 정보필터링 에이전트는 웹검색엔진과는 달리 사용자가 예측하지 못한 정보들을 발견하는 에이전트로서 사용자가 브라우징(browsing)이라는 형태로 많은 시간을 소모하여 할 수 있는 일을 대신해줄 수 있는 소프트웨어 에이전트이다.

아직 웹검색엔진을 대체할 만한 어떤 새로운 검색 방법, 어떤 새로운 검색 시스템도 없는 상황이지만 현재의 상황을 획기적으로 개선 시킬 수 있는 새로운 검색 패러다임(paradigm)이 개발되어야 한다는 점에서는 모두가 동의하리라고 믿는다.

참고문헌

[조강래 97]
조강래, 김형근, 신봉기, 김영환. WWW에서 모니터링 에이전트. HCI'97 학술대회 발표 논문집. 한국통신 멀티미디어 연구실. 1997.

[AltaVista]
The AltaVista. http://www.altavista.digital.com/

[Anikina 95]
Natallia Anikina, Valery Golender, Svetlana Kozhukhina, Leonid Vaniner, Bernard Zagatsky. REASON: NLP-based Search System for the WWW. AAAI Spring Symposium on Natural Language Processing for the World Wide Web, Stanford University, March, 1997 http://www.lingosense.co.il/ai_symp_paper.html

[Armstrong 95]
Robert Armstrong and Dayne Freitag and Thorsten Joachims and Tom Mitchell, WebWatcher: A Learning Apprentice for the World Wide Web , 1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments, Stanford, March 1995., http://www.cs.cmu.edu/afs/cs/project/theo-6/web-agent/www/project-home.html

[Balabanovic 97]
Balabanovic, M. An adaptive web page recommendation service. In *Proceedings of the 1st International Conference on Autonomous Agents*, Marina del Rey, Calif., Feb. 1997.

[Belkin 92]
N. J. Belkin and W. B. Croft. Information filtering and information retrieval: Two sides of the same coin?. *Communicqtions of the ACM*, 35(12):29-38, Dec. 1992.

[Cheong 96]
Fah-Chun Cheong. "Internet Agents: Spiders, Wanderers, Brokers, and Bots", New Riders Publishing, Indianapolis, Indiana. 1996.

[Douglas 96]
Douglas W. Oard. A Condeptual Framework for Text Filtering. EE-TR-96-25, May, 1996.

[Etzioni 95]
Etzioni, O., Weld, D. S. Intelligent Agents on The Internet: Fact, Fiction, and Forcast.

[Falk 96]
Anders Falk and Ing-Marie Jonsson. Media Lab, Ericsson Telecom. PAWS: an agent for WWW-retrieval and filtering.

http://www.fek.su.se/forskar/program/imorg/dok/1996-08-8_1/erimedlab/paws/NewPAAM.doc.html

[Foltz 92]
Peter W. Foltz and Susan T. Dumais, Personalized Information Delivery: An Analysis of Information Filtering Methods. CACM, Vol.35, No.12, 1992.

[Goldberg 92]
Goldberg, D., Nichols, D., Oki, B. and Terry, D. Using collaborative filtering to weave an information tapestry. CACM , Vol.35, No.12, Dec. 1992.

[Joachims 96]
Thorsten Joachims, Dayne Freitag, Tom Mitchell. WebWatcher: A Tour Guide for the Wrld Wide Web. *Technical Report CMU-CS-96-*, Carneige Mellon University, Sep. 1996.

[Konstan 97]
Joseph A. Konstan, Bradley N. Miller, David Maltz, Jonathan L. Herlocker, Lee R. Gordon, and John Riedl. GroupLens: Applying Collaborative Filtering to Usenet News. CACM, Vol.40, No.3, March 1997.

[Lang 95]
Lang, K. Newsweeder: Learning to filter netnews. In *Proceedings of the 12th International Conference on Machine Learning*, Tahoe City, Calif., 1995.

[Lieberman 95]
Henry Lieberman, Letizia: An Agent That Assists Web Browsing, Proceedings of the International Joint Conference on Artificial Intelligence, Montreal, August 1995., http://lieber.www.media.mit.edu/people/lieber/Lieberary/Letizia/Letizia.html

[Lieberman 96]
Henry Lieberman. Autonomous Interface Agents.
http://lieber.www.media.mit.edu/people/lieber/Lieberary/Letizia/AIA/AIA.html

[Lycos]
The Lycos, the catalog of the internet. http://www.lycos.com/

[Maes 94]
P. Maes. Agent that reduce work and information overload. *Communications of the ACM*, Vol. 37, No. 7, 1994.

[Malone 87]
Thomas W. Malone, Kenneth R. Grant, Franklyn A. Turbak, Steven A. Brobst, and Michael D. Cohen. Intelligent information sharing systems. CACM, vol.30, No.5, May 1987.

[Mauldin]
Michael L. Mauldin. "Lycos: Design choices in an Internet search service", http://www.computer.org/pubs/expert/1997/trends/x1008/mauldin.html.

[Mladenic 96]
Mladenic, D., (1996) Personal WebWatcher: Implementation and Design Technical Report IJS-DP-7472, October, 1996. (work is a part of CMU Text Learning Groupwork)

[Moukas 96]
Moukas, A. Amalthaea: Information Discovery and Filtering using a Multiagent Evolving Ecosystem.

[Oard 96]
Douglas W. Oard. A Conceptual Framework for Text Filtering. EE-TR-96-25. University of Maryland, College Park, May 1996.

[Pazzani 95]
M. Pazzani, L. Nguyen & S. Mantik, Learning from hotlists and coldlists: Towards a WWW information filtering and seeking agent, In *Proceedings of AI Tools Conference*, Washington, DC, 1995. http://www.ics.uci.edu/~pazzani/Coldlist.html

[Pazzani 96]
M. Pazzani, J. Muramatsu, D. Billsus. Syskill & Webert: Identifying Interesting Web Sites. In *Proceedings of the National Conference on Artificial Intelligence (AAAI96)*, Portland, 1996.

[Resnick 94]
Paul Resnick, Neophytos Iacovou, et.al. , GroupLens : An open architecture for collaborative filtering of net news. In Proceedings of the Conference on Computer Supported Cooperative Work, pp.175-186. ACM, October 1994.
http :// www.cs.umn.edu /Research /GroupLens

/cscwpaper / paper.html.

[Resnick 97]
Paul Resnick and Hal R. Varian. "Recommender Systems", *Communications of the ACM*, 40(3):56-58, Mar. 1997.

[Rocchio 71]
Rochio, Jr., J. Relevance feedback in information retrieval. In The Smart System – Experiments in Automatic Document Processing. Prentice Hall Inc. 313-323.

[Salton 83]
G. Salton and M. J. McGill. McGraw-Hill Computer Science Series: Introduction to Modern Information Retrieval. McGraw-Hill, New York, 1983.

[Salton 87]
G. Salton and C. Buckley. Term Weighting Approaches in Automatic Text Retrieval. Cornell University Technical Report 87-881, 1987.

[Salton 88]
G. Salton and C. Buckley. Term-weighting approaches in atomatic text etrieval. *Information Processing and Mangement*, Vol. 24, No.5, 1988.

[Sheth 93]
B. Sheth and P. Maes. Evolving Agents for Personalized Information Filtering. In: *Proceedings of te Ninth Conference on Artificial Intelligence for Applications*. IEEE Computer Society Press, 1993.

[Sheth 94]
Sheth. B. A Learning Approach to Personalized Information Filtering, *SM Thesis*, Department of Electrical Engineering and Computer Science, MIT, Feburary 1994.

[Simmany]
The Simmany. http://simmany.hnc.com/

[Smart Bookmarks]
Firstfloor. http://www.firstfloor.com/

[Stadnyk 92]
Stadnyk, I., Kass, R. Modeling User's Interests in Information Filters. CACM , vol.35, No.12, Dec. 1992.

[Stevens 92]
Curt Stevens , Knowledge-Based Assistance for Accessing Large, Pooly Structured Information Space. Ph D Thesis, University of Colorado, Dept. of Computer Science, Boulder, 1992. http://www.cs.colorado.edu/homes/stevens/public_html/ mypapers/Thesis-tech-report.ps.

[Web Buddy]
DataViz Home Page. http://www.dataviz.com/

[Yahoo]
The yahoo index. http://www.yahoo.com/

[Yan 95]
T.W. Yan and H. Garcia-Molina. SIFT – A Tool for Wide-Area Information Dissemination. In *Proceedings of the USENIX 1995 Winter Technical Conference*. New Orleans, La. Jan. 1995.

[Yan 94a]
T.W. Yan and H. Garcia-Molina. Distributed selective dissemination of information. In *Proc. Parallel and Distributed Information Systems*, pages 89-98, 1994.

[Yan 94b]
T.W. Yan and H. Garcia-Molina. Index structures for information filtering under the vector space model. In *Proc. International Conference on Data Engineering*, pages 337-47, 1994.

[Yan 94c]
T.W. Yan and H. Garcia-Molina. Index structures for selective dissemination of information under the boolean model. *ACM Transactions on Database System*, 19(2):332-64, 1994.