

자동항해를 지원하는 적응형 웹 서핑 시스템

(An Adaptive Web Surfing System for Supporting Autonomous Navigation)

국 형 준 ^{*}

(Hyung Joon Kook)

요 약 사용자 적응형 웹 서핑 시스템 설계를 위해 전체 시스템을 사용자 데이터 수집, 데이터 처리를 통한 사용자 프로파일 구축 및 개선, 그리고 사용자 프로파일의 적용을 통한 적응 등 세 단계로 나누어 접근할 수 있다. 본 연구가 설계한 세 부분의 에이전트들은 이들 각 단계에서 작업하며 상호 유기적인 협동을 통해 적응형 웹 서핑을 지원한다. 이들은 각각 대화식 인터페이스 에이전트(Interactive Interface Agent), 사용자 프로파일 에이전트(User Profile Agent), 그리고 자동항해 에이전트(Autonomous Navigation Agent)이다. 대화식 인터페이스 에이전트는 사용자 인터페이스를 제공하며 이를 통해 데이터를 수집하고 기계적 항해 지원을 수행한다. 사용자 프로파일 에이전트는 수집된 사용자 데이터를 처리하여 사용자 브라우징의 실시간에 사용자 프로파일을 동적으로 구축하고 갱신한다. 자동항해 지원 에이전트는 사용자 프로파일에 기초하여 사용자 관심과 가까운 웹 문서를 자동으로 선별하여 추천하는 자동 항해 모드를 제공한다. 본 연구가 제시하는 접근과 설계 방식은 향후 확장과 보안을 통해 실용 가능한 수준의 사용자 적응형 웹 서핑 시스템 구축에 활용될 수 있다.

키워드 : 적응형 웹 서핑, 자동항해 지원, 에이전트 시스템

Abstract To design a user-adaptive web surfing system, we may take the approach to divide the whole process into three phases; collecting user data, processing the data to construct and improve the user profile, and adapting to the user by applying the user profile. We have designed three software agents. Each privately works in each phase and they collaboratively support adaptive web surfing. They are IIA(Interactive Interface Agent), UPA(User Profile Agent), and ANA(Autonomous Navigation Agent). IIA provides the user interface, which collects data and performs mechanical navigation support. UPA processes the collected user data to build and update the user profile while user is web-surfing. ANA provides an autonomous navigation mode in which it automatically recommends web pages that are selected based on the user profile. The proposed approach and design method, through extensions and refinements, may be used to build a practical adaptive web surfing system.

Key words : Adaptive Web Surfing, Automatic Navigation Support, Agent System

1. 서 론

최근 인터넷의 발전과 함께 웹상의 다양한 정보를 사용자 적용 방식에 의해 접근할 수 있도록 지원하는 시스템에 관한 연구가 활발하다[1-3]. 이들 시스템은 사용자마다 고유한 관심을 시스템이 명시적 또는 묵시적인 방식에 의해 추적하고 이를 토대로 사용자의 웹 검색과

항해를 직접 또는 간접적인 방식에 의해 지원하는 것을 목표로 한다.

본 논문은 사용자 적응형 웹 서핑 지원을 위한 효과적인 방안으로서 수행한 다중 에이전트 시스템 연구의 결과를 제시한다. 본 연구가 구축한 시스템은 대화식 인터페이스 에이전트(IIA: Interactive Interface Agent), 사용자 프로파일 에이전트(UPA: User Profile Agent) 그리고 자동 항해 에이전트(ANA: Autonomous Navigation Agent)의 세 에이전트 부분으로 구성된다. 이 가운데 IIA 및 UPA 부분의 설계 내용은 [4]에서 상세히 소개된 바 있으므로, 본 논문에서는 이들에 관해서 간략히만 언급하고 ANA 부분의 설계 내용과 시스템 전체에

· 본 연구는 한국과학재단의 특장기초연구과제(과제 번호: 97-0100-0801-3) 연구비 지원에 의해 수행되었음

* 종신회원 : 세종대학교 컴퓨터공학과 교수

kook@sejong.ac.kr

논문접수 : 2003년 10월 28일

심사완료 : 2004년 1월 6일

초점을 맞추어 서술한다.

논문의 구성은 다음과 같다. 먼저 2장에서는 본 연구의 접근 방식을 관련 연구와 함께 소개한다. 3장에서는 본 연구에서 구현한 시스템 AutoNaver의 설계 원리와 구현 내용을 전체와 부문별로 나누어 시스템 개요를 제시한다. 4장에서는 ANA 부문의 설계 원리와 작업 내용을 소개한다. 이어 5장에서 몇 가지 연구 이슈에 관한 고찰과 시스템의 향후 개선 방안을 논한 후, 6장에서 결론을 제시한다.

2. 관련 연구 및 접근 방식

적용형 웹 서핑 시스템의 구축은 사용자 데이터 수집, 데이터 처리를 통한 사용자 모델 구축 및 개선, 그리고 사용자 모델의 적용을 통한 적용의 세 단계로 나누어 접근할 수 있다. 이들 각 단계에서의 연구 동향과 본 연구의 접근 방식을 설명한다.

첫째, 사용자 데이터 수집은 사용자가 보유하는 어떤 요소(feature)를 대상으로 시스템이 어떤 방식으로 적용할 것인가 하는 문제이다. 이 요소들은 궁극적으로 사용자 모델을 구성하는 요소가 되며 시스템은 사용자 모델에 표현된 각 요소의 현재 값(수치, 기호, 또는 구조적 값)에 의존하여 개별 사용자에게 대한 시스템 행위(즉, 적용)를 결정한다. 적용 대상으로는 주로 사용자의 지식[2,5,6], 목표[7,8], 배경 및 경험[1,9,10] 등이 사용되어 왔다. 데이터 수집의 방식으로는 크게 사용자 모델 구축을 위한 사용자-시스템간의 직접적인 협동에 의한 방식과 시스템의 명시적 개입이 없는 간접적인 방식이 존재한다. 직접적인 방식은 사용자 데이터 수집을 위해 브라우저의 초기에, 또는 데이터의 갱신을 위해 브라우저의 중간이나 완료 시점에서 시스템이 개입함으로써 사용자에게 시간 비용과 인지적 부담을 부과하는 단점이 있다. 따라서 이러한 부담을 최소화할 수 있는 보다 간접적인 방식의 개발에 관한 연구의 필요성이 증가하고 있다 [11-14].

본 연구에서의 사용자 데이터 수집은 별도의 대화식 인터페이스 에이전트에 의해 시스템의 명시적 개입이 없는 간접적이고 자동적인 방식에 의해 작동한다. 즉, 에이전트는 사용자의 브라우징 행위를 백그라운드에서 지속적으로 모니터링한 데이터를 사용하여 사용자 데이터를 수집해 나간다. 모니터 대상이 되는 사용자 행위는 브라우징 과정에서 특정 웹 문서를 사용자가 '관심있음(Mark As Cool)'으로 표시하는 행위, 시스템이 자동으로 추천한 웹 문서를 거부하는 행위, 그리고 사용자가 검색용 키워드들을 제공하는 행위 등이 포함된다. 한편 북마킹, 저장, 인쇄 등 웹 브라우저 자체의 명령 버튼을 이용한 관심 표시는 MS Explorer 등 특정 브라우저에

의존적인 구현을 필요로 하는 관계상 본 연구의 구현 범위에서 제외되었다.

둘째, 사용자 모델 구축 및 개선은 사용자로부터 수집한 데이터를 어떤 형태로 저장, 표현하고 갱신해 나가는지에 관한 문제이다. 본 연구는 이를 위해 사용자 프로파일 에이전트라는 별도의 에이전트가 사용자로부터 수집한 데이터를 처리하여 관심 벡터(Interest Vector)로서 저장하는 방식을 취한다. 관심 벡터란 가중치가 부과된 관심 키워드들의 목록이다. 구체적으로, 하나의 관심 벡터는 (키워드, 가중치) 쌍의 집합인데, 하나의 벡터에 속한 가중치의 총합은 100 정도가 되도록 유지한다. 초기 사용자인 경우, 초기 브라우징에 기반한 최초의 관심 벡터가 생성되며, 사용자의 브라우징이 계속됨에 따라 관심 벡터의 갱신 또는 추가가 이루어진다. 여기서 갱신은 사용자의 관심이 구체화되는 경우 기존의 관심 벡터가 갱신됨을, 추가는 사용자의 관심이 이동하는 경우 이것으로부터 새로운 관심 벡터가 생성됨을 말한다. 따라서 사용자 프로파일의 주 내용은 일련의 관심 벡터 집합이라 할 수 있으며, 이를 통해 시스템은 특정 분야에 대한 사용자의 관심이 증진되는 양상과, 새로운 분야로 관심이 이동하는 양상을 자연스러운 방식으로 표현하게 된다.

마지막으로, 사용자 모델의 적용을 통한 적용은 시스템의 어떤 요소를 어느 시점에 적용에 사용하는가 하는 문제이다[15,16]. 웹과 같은 하이퍼미디어 인터페이스에서 사용 가능한 요소는 크게 텍스트, 멀티미디어 항목과 같은 내용(contents) 요소와 링크의 존재, 링크의 순서, 링크 주석, 맵 구조 등 항해(navigation) 요소로 구분하여 생각할 수 있다[1,3,9]. 시점에 관해서는 사용자의 요구가 있을 때만 작동하는 기계적인 방식, 그리고 시스템이 적절한 시점을 포착하여 작동하는 자동적인 방식이 가능하다.

효과적인 적용 수단으로서 본 연구는 이 가운데 링크 순서와 주석을 통한 기계적 항해 지원과 시스템에 의한 자동 항해 지원을 합동 구상한다. 먼저, 기계적 항해 지원은 대화식 인터페이스 에이전트를 통해 사용자에게 의해 입력된 키워드를 기반으로 한 검색을 거쳐 얻어진 웹 문서들의 링크 목록을 키워드와의 관련도가 높은 순서로 제시하는 것을 말한다. 또한 사용자의 필요에 의해, 관심있는 웹 문서의 주요 어휘들을 추출, 화면에 표시함으로써 사용자로 하여금 현재 문서의 대강을 인지하도록 하는 동시에 키워드 리스트로부터 관심 키워드(들)를 손쉽게 선택하여 다음 검색의 질의어로서 사용될 수 있도록 한다. 다음, 자동 항해 지원은 별도의 자동 항해 에이전트에 의해 웹 문서 선별과 추천이 자동화됨을 말한다. 사용자 요구에 의해 구동하는 자동 안내 모

드에서, 시스템은 사용자 브라우징의 백그라운드에서 사용자 프로파일의 내용을 토대로 웹 문서 분석을 실행한다. 분석의 대상 범위는 기계적 항해 지원 체계에서 제시된 링크들이며 이 가운데 사용자가 현재 브라우징하고 있는 웹 문서보다 사용자 모델과 더욱 부합하는 웹 문서가 있는 경우, 이를 브라우징 윈도우에 자동으로 불러들이는 방식을 취한다.

국내에서도 인터넷을 기반으로 한 적응형 에이전트에 관한 연구가 활발하다[17-22]. 이 연구들은 정보의 추출 [17,18], 사용자 적응[19,20], 학습 및 이동 에이전트 [21,22] 등 다양한 부문에서 진행되고 있다. 이 가운데 [19,20]은 본 연구와 유사한 부문이랄 수 있는 적응형 웹 검색 에이전트에 관한 연구 내용을 제시한다. 먼저, [19]에서는 사용자가 브라우징하고 있는 현재 웹 문서에 대해 문서 변경 감시 또는 자연 검색을 요구할 경우, 이를 양의(positive) 피드백으로 받아들여 사용자 모델을 갱신해 나가며, 갱신된 모델은 차후에 브라우징되는 문서를 시스템이 추천할지 여부를 판단하는 기초가 된다. 다음, [20]은 사용자 프로파일에 기초한 개인 웹 정보 검색 및 추출용 에이전트를 제시한다. 이 에이전트는 감독 학습(supervised learning) 방식을 동원하여 개인의 관심 키워드들로부터 결정 트리를 구축하고 이를 정련해 나간다. [19,20]에서 특정 웹 문서가 사용자의 관심 대상인지 판단하는 것은 결정 트리 형태로 구축된 사용자 모델에 의거한다. 또한, 이러한 사용자 모델의 구축은 일정 수준의 사용자 피드백에 주로 의지한다. 이에 비해 본 연구는 사용자 모델을 일련의 관심 벡터로써 구축하고, 특정 웹 문서의 사용자 관심 여부를 '있다/없다' 만이 아닌, 사용자의 현재 관심 벡터와 문서 간의 연관성을 정량화된 개념으로 판단한다. 그로 인해 다수의 문서 간에 연관 정도를 상호 비교할 수 있고, 이 정보를 웹 문서 추천에 활용할 수 있게 된다는 점에서 유리하다.

3. 시스템 설계 내용

앞 장에서는 본 연구가 제안하는 적응형 웹 서핑 시스템 설계의 대략적인 접근 방식을 제시하였다. 이 장에서는 앞서의 접근 방식에 의해 본 연구가 구축한 시스템(AutoNaver)의 내용을 설명한다. 그림 1은 시스템의 전체적 설계 내용을 나타내는 구성도를 보인다.

그림에서 보는 바와 같이, 사용자 적응에 관련된 모든 작업은 세 개의 독립된 세부 에이전트의 협력으로 수행된다. 이들은 각각 대화식 인터페이스 에이전트(IIA: Interactive Interface Agent), 사용자 프로파일 에이전트(UPA: User Profile Agent) 그리고 자동 항해 에이전트(ANA: Autonomous Navigation Agent)이다. 따

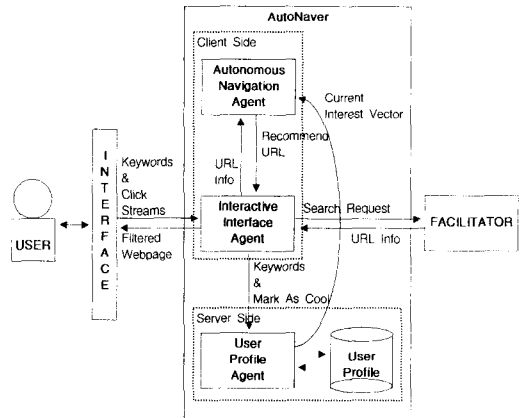


그림 1 AutoNaver 시스템 구성도

라서 AutoNaver는 다중의 세부 에이전트로 구성된 에이전트 시스템이라 할 수 있다. AutoNaver 서버에 접속하면 이 가운데 IIA와 ANA는 서버로부터 다운로드되어 클라이언트에서 작동하는 에이전트로, UPA는 서버에서 작동하는 에이전트로서 상호 연계 수행하여 사용자 적응형의 웹 브라우징을 가능하게 한다. 이어지는 장에서는 본 연구의 핵심적인 내용을 구성하는 이들 에이전트 각각에 대한 설계 및 구현 내용을 제시한다.

3.1 대화식 인터페이스 에이전트(IIA)

AutoNaver 시스템의 중심에 위치하는 IIA는 시스템 전면(foreground)에서 사용자 인터페이스를 제공하고, 후면(background)에서 시스템 내외부의 여타 에이전트들(UPA, ANA, Facilitator)과의 통신 채널을 유지한다. 이 가운데 협동 에이전트(Facilitator)는 본 연구와는 별도의 연구로 수행된 에이전트 시스템으로서, 본 연구의 AutoNaver 시스템에서 발생하는 검색 요청을 처리한다.

그림 2는 IIA의 사용자 인터페이스 화면을 보인다. 인터페이스 화면은 화면 오른쪽의 현재 웹 문서(Current Webpage) 출력 영역, 왼쪽 상단의 키워드 목록(Keyword Index) 입출력 혹은 검색된 URL 목록(Suggest Link)의 출력을 위한 탭 영역, 그리고 왼쪽 하단의 자동 항해와 관련된 애플릿 영역으로 구성된다. 각 영역의 기능과 설계 내용을 간략히 기술하기로 한다.

현재 웹 문서(Current Webpage) 출력 영역은 현재 사용자가 브라우징하고 있는 웹 문서를 표시한다. 여기에는 사용자가 직접 선택한 웹 문서 혹은 후에 설명할 ANA에서 추천된 웹 문서가 출력된다. 키워드 색인(Keyword Index) 탭 영역에서는 현재 문서에 대한 사용자의 관심을 명시적으로 표현하거나('Mark As Cool'), 현재 웹 문서의 어휘 벡터를 추출하여 어휘의 가중치 순으로 나열하거나('Extract Keywords'), 원한

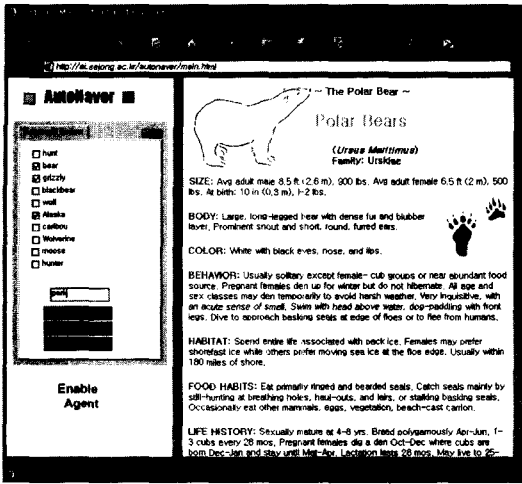


그림 2 AutoNaver의 인터페이스 화면

다면 현재 웹 문서에 기초한 심층적인 추가 검색('Search')을 수행하도록 지원한다. 어휘 벡터는 (어휘, 가중치) 쌍의 집합으로 해당 웹 문서가 가지는 주요 어휘를 빈도 수에 의거하여 나열한 자료 구조로서 협동 에이전트로부터 입수되는 데이터이다. 즉, 현재 웹 문서에서 사용되고 있는 주요 어휘들을 일목요연한 형태로 제시해 줌으로서 사용자가 현재 웹 문서의 성격을 신속히 판단할 수 있도록 도와주는 동시에, 이들 가운데 일부 어휘를 선택하여 선택 어휘 집단에 기초한 추가적 검색을 가능케 한다. 제안 링크(Suggest Link) 탭 영역에는 키워드 색인 탭 영역에서 'Search' 요청에 대한 응답으로서 검색된 웹 문서들의 URL 목록이 제시된다(그림 3 참조). 여기서 URL들은 단일 계층 구조의 목록이 아닌 트리 형태의 복합 계층 구조의 목록으로 제시됨으로서 사용자 브라우징의 경로를 입체적인 형태로 보여준다. 마지막으로 ANA 애플릿 영역에서는 AutoNaver의 자동 항해 지원 기능을 활성화하는 버튼('Enable Agent')이 제공된다.

IIA는 시스템 내의 다른 세부 에이전트들과 필요시에 통신을 수행한다. 먼저, 사용자의 검색 질의('Search') 요구나 관심 표시('Mark-As-Cool')는 AutoNaver 시스템의 서버에서 작동하는 UPA에게 메시지로 전달하여 사용자 관심 정보를 구축하고 갱신하는데 활용되도록 한다. 검색의 결과는 URL 정보(URL Info)의 형태로 전달받는데, 이는 URL 목록과 각 URL 웹 문서의 어휘 벡터들로 구성된다. IIA는 이 URL 정보를 다시 IIA와 동일 클라이언트에서 작동하는 ANA에게 전달하여 자동 항해 지원의 자료로 활용되도록 한다. 이 같은 전달은 계속적인 검색 요구에 따라 URL 정보가 변경될 때마다 이루어짐으로써 차후 설명할 ANA의 자

동 항해 지원이 최신의 자료에 근거하도록 보장한다. 활성화된 ANA는 자동 항해 지원을 위한 작업을 수행하고 그 결과 IIA에게 웹 문서를 추천하게 된다.

3.2 사용자 프로파일 에이전트(UPA)

UPA는 특정 사용자의 브라우징이 진행됨에 따라 동적으로 변화하는 사용자 관심을 후면(background)에서 처리하여 사용자 프로파일을 구축하고 갱신한다. 사용자 프로파일은 사용자별 텍스트 파일로서 사용자의 관심 분야를 나타내는 정보가 복수개의 관심 벡터(Interest Vector) 형태로 저장된다. 관심 벡터는 사용자가 제시한 검색 질의어 집합 또는 관심이 표시된 현재 웹 문서의 어휘 벡터로부터 초기화되며, 다음에서 보듯이 관심 벡터는 고유 번호, 갱신 회수, (키워드, 가중치) 쌍의 집합으로 구성된다.

```
((Interest_Vector 13)
(Number_of_Updates 7)
((bear 46)(alaska 25)(arctic 12)(food 11)(life 6)...
)
```

사용자의 시스템 사용 초기에는 단 하나의 관심 벡터만을 가지게 되지만 해당 사용자의 브라우징과 검색이 거듭되면서 관심 벡터의 수가 늘어나고 구체화되는 양상으로 사용자 프로파일이 갱신된다. 갱신 과정은, 먼저 새로운 검색 질의어 집합이 주어지거나 특정 웹 문서가 관심 웹 문서로 알려진 경우, 해당 검색 질의어 집합 또는 해당 웹 문서로부터 어휘 벡터를 추출한다. 다음, UPA는 사용자 프로파일에 저장된 기존의 관심 벡터 가운데 이것과 근접한(similar) 벡터가 있을 경우, 이와 합병(merge)하여 기존의 관심 벡터를 갱신하여 기존 관심에서 더욱 구체화된 관심을 표현하는 효과를 얻거나, 근접한 벡터가 없을 경우, 이로부터 새로운 관심 벡터를 생성하여 넓혀진 관심의 폭을 수용하는 효과를 얻는다. 또한, 합병이나 생성의 마지막 과정에서는 언제나 유효한 가중치 값을 가지는 키워드들만으로 관심 벡터의 길이를 단축함으로써 벡터가 불필요하게 커지는 것을 방지한다. 마지막으로, 이렇게 하여 합병 또는 생성되는 관심 벡터는 현재 관심 벡터로, 즉 사용자의 실시간 현재의 관심사로 취급되는데, 이는 후에 설명할 자동 항해 지원의 근거로서 활용된다.

사용자 프로파일은 AutoNaver 시스템 서버에 사용자별로 유지 관리됨으로서 사용자 접근과 관리의 용이성을 추구함과 동시에 향후 협동 에이전트 등 외부 에이전트가 필요시 그 내용을 참조할 수 있게 했다. 한편, 로컬 파일 접근의 필요성과 작업 효율을 고려하여 UPA는 동일 서버에서 작동하는 서블릿으로 구현되었다. UPA와 사용자 프로파일에 관한 세부 내용은 근접도(similarity) 평가 방식과 관심 벡터로부터 유효 키워드

집단을 선택하는 방식과 함께 [4]에 상세히 제시되어 있다.

4. 자동 항해 에이전트(ANA)

시스템과의 직접적인 상호 작용을 통한 가이드에 의한 정보 검색 방식은 때로는 사용자의 인지적 부담을 가중시킴으로써 검색의 흥미와 효과를 저하시킬 수 있다. 이런 관점에서, 본 시스템에 있어서 키워드 색인 및 제안 링크를 통한 항해는 사용자 관심도를 직접 반영하는 검색을 상당히 지원하지만, 학습자가 저연령이거나, 시스템 사용에 익숙치 않거나 또는 최선보다는 편리한 것을 선호하는 사용자에게는 다소 불편한 방식일 수도 있다. 본 연구는 이러한 경우에 매우 유용할 수 있는 자동 항해 에이전트 ANA(Autonomous Navigation Agent)에 의한 자동 항해를 지원한다.

ANA에 의한 안내는 현재 웹 문서 영역 하단의 'Enable Agent'를 선택함으로써 시작된다(그림 2 참조). 이 시점부터 ANA는 사용자의 브라우징 행위와 독립적으로 사용자 관심에 따른 항해 안내를 수행하게 된다. 앞서 IIA가 대화 화면의 전면(foreground)에서 수행하는 것과는 대조적으로, ANA는 사용자와의 대화를 강요하지 않는 방식으로 항해를 후면(background)에서 지원한다. 자동 항해 지원은 제안 링크 목록 가운데 사용자가 브라우징하고 있는 현재 웹 문서보다, 현재 관심 벡터(Current Interest Vector)로 표현되는 사용자의 현재 관심과 연관도가 더 높은 웹 문서를 선별해 내어 이를 현재 웹 문서로 대체하도록 추천하는 방식으로 이루어진다. 이어지는 절에서는 ANA와 관련된 세부 연구 내용과 웹 문서 추천 사례를 제시한다.

4.1 ANA 작업 메커니즘

ANA는 자동 항해 에이전트로서 사용자의 활성화 요구가 있어야만 작동하여 웹 문서를 선별하여 추천한다. 선별의 기반이 되는 정보로는 현재 웹 문서에 대한 정보, 현재 관심 벡터, 그리고 제안 링크에 대한 정보가 포함된다. 이 가운데 현재 웹 문서와 제안 링크 정보는 동일 클라이언트내의 IIA로부터, 현재 관심 벡터는 서버의 UPA로부터 최신의 것을 전달받아 사용한다. 비활성(disabled) 상태의 ANA는 웹 문서에 대한 선별 작업을 멈추고 선별의 기반이 되는 정보만을 유지한다. 다시 말해 ANA는 비활성화 되더라도 작동을 아주 멈추는 것이 아니라 장래 활성화에 대비해 최소한의 정보 유지를 계속하는 것이다.

ANA에 의한 웹 문서 선별 작업은 다음 네 가지 경우에 구동된다. 첫째 'Enable Agent' 버튼 선택시, 둘째 활성 상태에서 현재 관심 벡터가 변경된 경우, 셋째 활성 상태에서 현재 웹 문서가 변경된 경우, 넷째 역시 활

성 상태에서 제안 링크가 변경된 경우이다. 첫째 경우는 ANA가 비활성 상태에서 활성 상태가 된 것으로, 선별 기반 정보의 의거하여 웹 문서 선별을 수행하게 된다. 둘째 경우는 ANA가 이미 활성인 상태에서 사용자 관심이 변경이 된 것으로, 선별과 추천이 이미 이루어졌다 하더라도 이것과 관계없이 웹 문서에 대한 재선별을 시도하게 된다. 사용자 관심이 변경되었으므로 선별 기준도 변경되었기 때문이다. 사용자 관심의 변화 즉, 현재 관심 벡터의 변경은 서블릿으로 수행하는 UPA와의 정기적인 통신으로 알아낸다. 셋째 경우는 사용자 브라우징 행위로 현재 웹 문서가 변경된 경우인데 이때는 선별의 기준이 되는 현재 웹 문서가 변경된 것이므로 선별 작업을 재개하게 된다. 넷째 경우는 사용자의 브라우징에 의해 제안 링크 목록이 변경된 경우이다. 이때는 사용자가 제안 링크 중 하나의 링크를 선택하여 브라우징함과 동시에 새로운 제안 링크 목록에 기반하여 웹 문서 선별을 다시 시작하게 된다. 다시 말해서, 첫째 경우에서 구동된 ANA의 웹 문서 선별 작업은 둘째, 셋째, 넷째 경우가 발생하면 작업을 중단하고 새로운 선별 기준에 의거하여 재실행하게 된다.

4.2 웹 문서 선별

활성 상태의 ANA는 제안 링크 목록 가운데 사용자 관심과 가까운 웹 문서를 선별해내는 작업을 수행한다. 선별 작업에 들어가면 ANA는 맨 먼저 현재 웹 문서의 어휘 벡터와 현재 관심 벡터 간의 연관도(Current Affinity)를 계산한다. 그 다음, 현재 제안 링크 목록에 포함된 각 URL의 어휘 벡터를 차례로 추출, 현재 관심 벡터와의 연관도를 계산한다. 계산 결과, 연관도가 현재 웹 문서의 연관도보다 크면 해당 웹 문서 추천을 시도한다. 그리고는 연관도를 갱신한 후 나머지 URL들에 대한 비교 작업을 계속한다.

본 선별 작업의 특징이라면 제안 링크의 URL 중 최대 연관도를 가지는 URL을 찾아 이를 추천하는 방식이 아니라, 현재 웹 문서의 연관도보다 조금이라도 높은 연관도를 가지는 URL이 발견되는 즉시 이를 추천한다는 점이다. 이는 사용자의 항해 지원 요구를 신속하게 반영함으로써 시스템의 사용성을 극대화하는 이점이 있다. 또한 이러한 선별 작업은 한번에 끝나는 것이 아니라 나머지 URL에 대해서도 지속적으로 이루어지기 때문에, 언제나 당시의 현재 웹 문서보다는 연관도가 높은 웹 문서를 선별 시도함에 따라 시간이 경과할수록 최적의 웹 문서는 더욱 빨리 찾아지게 되는 이점도 있다.

연관도(affinity)는 특정 웹 문서가 현재의 사용자 관심, 즉 현재 관심 벡터와 얼마나 연관이 높은지를 나타내는 척도이다. 연관도는 다음 수식을 사용하여 계산된다.

$$Affinity = \sum(w_{iv} \times \min(w_{iv}, w_{tw})) - \sum(w)$$

where w_{iv} and w_{tw} is the weight in

CurrentInterestVector and the weight in TermVector, respectively, of t in $\{T_{iv} \cap T_{tw}\}$, and w is the weight of t in $\{T_{iv} \cup T_{tw}\} - \{T_{iv} \cap T_{tw}\}$

T_{iv} = terms in CurrentInterestVector

T_{tw} = terms in TermVector

즉, 특정 웹 문서의 사용자 관심과의 연관도는 해당 웹 문서의 어휘 벡터와 현재 관심 벡터의 공통 키워드들의 가중치를 상호 곱한 것들의 총합에서 비공통 키워드들의 가중치들을 뺀 것으로 정의된다. 따라서 연관도는 가중치가 큰 공통 키워드들이 많을수록 높게, 그리고 가중치가 높은 비공통 키워드들이 많을수록 낮게 나오게 된다. 위 식에서 특기할 점은, 공통 키워드 가중치 간의 곱셈에 있어서 어휘 벡터 쪽의 가중치가 현재 관심 벡터 쪽의 가중치보다 클 경우, 이를 작은 쪽 가중치, 즉, 현재 관심 벡터 쪽의 가중치로 평가 절하하여 곱셈에 적용한다는 점이다. 이는 사용자가 비교적 낮은 관심(예: 8)을 갖는 공통 키워드가 특정 웹 문서에서 상대적으로 큰 가중치(예: 25)를 가진 경우, 웹 문서 쪽의 가중치를 그대로 반영하는 것이 아니라 현재 관심 벡터의 가중치와 같은 수준으로 낮춰 적용함으로써, 해당 키워드의 연관도를 $8 \times 25 = 200$ 이 아닌, $8 \times 8 = 64$ 로 평가되도록 하고자 함이다. 이와 반대의 경우, 즉, 어휘 벡터 쪽의 가중치(예: 8)가 현재 관심 벡터 쪽의 가중치(예: 25)보다 작을 경우, 양쪽의 가중치를 그대로 곱하여 해당 키워드의 연관도($25 \times 8 = 200$)를 구함으로써 사용자 쪽의 가중치를 최대한 반영한다.

4.3 자동 항해 지원 시나리오

이 절에서는 자동 항해 지원의 시나리오를 제시한다. 그림 2에 주어진 상태에서 출발하여 사용자 행위마다 순번을 붙이고, 각 행위에 대한 시스템의 대응을 예시 화면과 함께 설명한다.

① 사용자 : 'Enable Agent'

사용자의 요구에 의해 ANA가 활성화된다.

② 사용자 : 'Search'

사용자가 제공한 검색 키워드들(bear, grizzly, Alaska)을 기반으로 한 웹 문서 검색이 행해진다.

주어진 키워드들에 기반하여 현재 관심 벡터가 갱신된다.

검색의 결과가 제안 링크 탭 영역에 웹 문서들의 URL 목록으로 제시된다(그림 3의 왼편 탭 참조).

ANA는 현재 관심 벡터에 기반하여 제안 링크 웹 문서들에 대한 선별 작업을 개시한다.

선별 작업에 의해 현재 웹 문서보다 연관도가 높은 웹 문서(www.yellowstonebearworld.com)가 발견되어

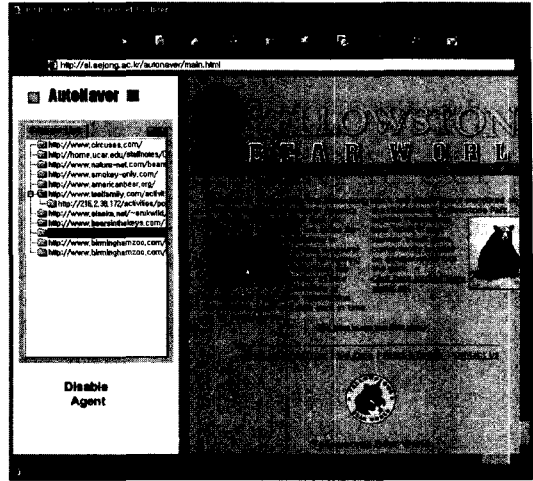


그림 3 ANA의 추천이 이루어진 화면

추천 작업이 시작된다. 추천시, ANA는 일정 시간(현재 5초) 동안 인터페이스 화면의 ANA 애플릿 영역에 알람 막대(alarm bars)를 발행하여, 사용자로 하여금 이를 받아들일지 거부할지에 대한 선택의 시간적 여유를 제공한다.

③ 사용자 : (5초간 무응답으로 시스템의 웹 문서 추천을 받아들임)

알람 막대는 막대 수를 늘이면서 진행되며 막대 수가 팍 찼을 때 즉, 5초가 경과한 후에 추천된 웹 문서를 현재 웹 문서 영역으로 불러들인다(그림 3의 오른편 화면 참조).

④ 사용자 : 'Mark as Cool'

사용자가 추천된 웹 문서를 관심 문서로 표시한다.

이 행위로 현재 관심 벡터가 갱신된다.

ANA는 갱신된 현재 관심 벡터에 기반하여 제안 링크 웹 문서들에 대한 재선별 작업을 개시한다.

재선별 작업에 의해 현재 웹 문서보다 연관도가 높은 웹 문서(www.americanbear.org)가 발견되어 추천 작업이 시작된다. 추천은 앞서와 같이 5초간의 알람 막대 방식으로 진행한다.

⑤ 사용자 : 'Disable Agent'

이 시점에서 사용자는 현재 웹 문서에 흥미를 느낌으로 해서 새로이 추천된 웹 문서로의 이동을 원치 않는다. 시스템의 추천을 거부하기 위해 알람 막대 진행 중에 ANA 애플릿 영역의 'Disable Agent' 버튼이나 알람 막대를 클릭해주면, ANA가 비활성화 되어 자동 항해 지원이 정지되는 동시에 추천 웹 문서도 불러오지 않게 된다.

5. 고찰 및 향후 개선

이 장에서는 본 연구가 구축한 자동 항해 지원 에이전트의 설계에 수반된 문제들에 관해 고찰하고 향후 개선의 여지에 대해서도 생각하기로 한다.

첫째, 웹 문서 추천의 기준에 관한 문제이다. 현재의 설계 방식에서 웹 문서 추천의 기준은, 사용자 관심과 후보 웹 문서의 상호 연관도를 측정하여 그 값이 사용자가 현재 브라우징하고 있는 웹 문서의 그것보다 크나에 달려 있다. 비교 결과 전자가 우세하면 후보 웹 문서의 추천에 들어간다. 그렇지 않을 경우 다음 후보 웹 문서에 대해 이를 반복한다. 본 연구가 이 방식을 채택한 이유는 현재의 모든 후보 웹 문서 중 최적의(best) 웹 문서를 추천하기 위해 실시간 성능을 희생하기 보다는, 더 나은(better) 문서를 추천하는 방식을 취함으로써 어느 정도 실시간 성능을 유지하면서 사용자의 항해를 측면 지원하도록 하기 위해서이다. 이 방식이 가지는 또 하나의 장점은 사용자 관심의 변화에 보다 순발력 있게 적용한다는 점이다. 즉, 잠시의 브라우징 과정에서도 사용자 프로파일은 변화할 수 있는데, 이런 경우 이미 과거가 된 사용자 관심에 기초한 최적의(best) 웹 문서를 고르기 위해 소비한 시간은 무의미하게 될 것이기 때문이다. 그러나 향후 실시간 성능의 희생이 문제가 되지 않을 정도로 시스템 수행시간이 현저히 향상되는 경우 최적을 추구하도록 개선할 여지도 충분히 있겠다.

둘째, 웹 문서 추천 과정에서의 시간 지연에 따른 문제이다. 즉, 웹 문서 추천 과정이 일정 시간(현재 5초)의 알람 발생 후 완료되기 때문에, 그 사이에도 이보다 연관도가 높은 웹 문서가 하나 또는 그 이상 찾아질 수도 있다. 이 상황에 대해 여러 가지 대처 방법이 있을 수 있겠다. 먼저, 가장 단순한 접근 방식으로서, 웹 문서가 찾아질 때마다 다시 같은 시간의 알람을 발생시키는 방식이다. 하지만 알람이 누적될 경우 사용자가 과연 흥미를 가질지 알 수 없는 자동 추천 웹 문서 하나를 보는데 5초의 몇 배가 되는 s시간을 대기해야 한다면 시스템의 사용성이 극히 저하될 것이다. 따라서 사용자에게 일정한 시간(5초) 이상의 대기를 요구하지 않아야 하는 전제 아래 생각할 수 있는 다음 두 가지 대처 방안을 생각할 수 있다. (대안 1)은 처음 선별된 웹 문서를 5초 대기 후 일단 불러오는 방식이다. 즉, 나중에 선별된 웹 문서들에 대해서는 선별된 순서대로 다시 5초 대기에 의한 추천 과정을 반복한다. (대안 2)는 몇 개의 웹 문서가 선별되더라도 5초라는 고정된 시간 내에서 마지막에 선별된 웹 문서를 추천한다. 즉, 그 사이 선별된 것중에선 가장 연관도가 높은 웹 문서가 화면에 불러오게 된다.

위의 (대안 1)은 웹 문서를 자주 교체함으로써 사용자로 하여금 풍부한 브라우징과 'Mark As Cool' 등을

이용한 취사선택을 가능케 한다는 장점이 있으나, 제안 링크 목록이 긴 경우 많은 웹 문서들이 연관도의 미세한 오름차순으로 약 5초 간격으로 불러지게 되므로 시스템 사용 효과가 저하된다는 문제를 가진다. (대안 2)는 현재 채택된 방식으로서, (대안 1)이 선별되는 웹 문서 모두에게 의미를 부여하는 것과 달리, 그 중 마지막 문서 즉, 가장 높은 연관도를 가지는 웹 문서에게만 의미를 부여하는 방식이다. 중간에 몇 개의 웹 문서가 추천되었다고 해도 사용자의 입장에서는 큰 의미가 없다. 사용자는 단지 현재 웹 문서보다 자신의 관심에 더 근접한 웹 문서를 빨리 볼 수 있으면 된다.

현재 웹 문서가 교체되기 전까지 5초 동안 알람 막대가 진행되는 시간은 사용자에게 선택의 여유를 주기 위한 최소한의 시간이라 생각된다. 이 시간이 너무 짧으면 사용자는 알람 막대가 움직이는 것을 보지도 못하거나, 보더라도 추천을 거부하기 위한 행동을 취할 충분한 시간이 없게 된다.

마지막으로, UPA와 ANA 간 통신 구현에 관한 문제이다. 현재 UPA는 AutoNaver 시스템의 서버에서 서블릿으로 작동한다. UPA를 자바 서블릿으로 작성한 가장 큰 이유는 사용자 프로파일을 서버에 저장시키기 위해서이다. 사용자 프로파일을 서버에서 작동하는 UPA와 별도로 사용자의 컴퓨터에 직접적으로 저장시키는 방법도 연구했으나, 아직까지 서블릿은 보안상의 문제로 이를 허락하지 않는다. 또한 UPA를 애플릿으로 작성하더라도 보안상의 문제로 직접적인 사용자 프로파일 접근이 불가능하다. 이러한 구현 방식때문에 문제가 되었던 부분은 ANA의 웹 문서 선별 작업에 필요한 현재 관심 벡터를 UPA가 능동적으로 보내주지 못한다는 점이다. 서블릿은 요청에 의해서만 정보를 제공한다는 보안 문제가 걸려 있기 때문이다. 따라서 현재는 ANA의 반복 요청 방식에 의해 통신하지만, 이 부분은 가능하다면 UPA가 Vector 변경이 있을 경우에만 능동적으로 보내는 방식으로 개선할 필요가 있다.

6. 결론

인터넷 웹 문서들을 개별 사용자의 관심과 흥미에 맞는 방식으로 항해하도록 지원하는 적응형 웹 서핑 시스템은 여러 가지 관점에서 초점을 달리 하여 설계되고 구축될 수 있다. 본 연구는 시스템의 요건을 사용자 데이터 수집, 사용자 모델 구축 및 개선, 그리고 사용자 모델 적용을 통한 적용의 세 부분으로 분리하고, 각 부분에 대해 작용하는 별도의 에이전트 모형을 설계하고 이들을 유기적으로 결합한 시스템을 제안하였다.

구체적으로, 대화식 인터페이스 에이전트는 사용자 데이터 수집과 기계적 항해 지원을 수행하는 체계를 제시

한다. 사용자 프로파일 에이전트는 사용자 데이터에 기초하여 사용자의 관심을 브라우저의 실시간에 동적으로 프로파일하는 체계를 제시한다. 그리고 자동항해 지원 에이전트는 사용자 프로파일에 기초하여 사용자 관심과 가까운 웹 문서를 자동으로 선별하여 추천하는 체계를 제시한다.

본 연구가 제시한 부문별 기능 정의 및 주요 설계 원리들은 그 미비점을 향후 평가와 보완을 통하여 개선해 나간다면 실용 가능한 수준의 적응형 웹 서핑 시스템으로 발전할 수 있을 것으로 본다.

참 고 문 헌

- [1] Boyle, C. and Encarnacion, A. O., "MetaDoc: An Adaptive Hypertext Reading System," *UMUAI* 4(1), 1-19, 1994.
- [2] Kobsa, A., Muller, D. and Nill, A., "KN-AHS: An Adaptive Hypertext Client of the User Modeling System BGP-MS," *UM'94*, pp.31-36, 1994.
- [3] Maybury, M. T. (ed), *Intelligent Multimedia Interfaces*, MIT Press, 1993.
- [4] 국형준, "적응형 웹 서핑 지원을 위한 에이전트 시스템", 정보처리학회 논문지 9-B-4, pp.399-406, 2002.
- [5] Greer, J. and MaCalla, G. (eds.), *Student Modeling: The Key to Individualized Knowledge-Based Instruction*, NATO ASI Series F, vol.125, Springer-Verlag, 1993.
- [6] Hohl, H., Boecker, H. D. and Gunzenhauser, R., "Hypadapter: An Adaptive Hypertext System for Exploratory Learning and Programming," *UMUAI* 6, 1996.
- [7] Grunst, G., "Adaptive Hypermedia for Support Systems," In Schneider-Hufschmidt, M., Kuhme, T. and Malinowski, U. (eds.), *Adaptive User Interfaces: Principles and Practice*, North Holland, pp.269-283, 1993.
- [8] Micarelli, A. and Sciarone, F., "A Case-Based Toolbox for Guided Hypermedia Navigation," *UM'96*, pp.129-136, 1996.
- [9] Beaumont, I., "User Modeling in the Interactive Anatomy Tutoring System ANATOM-TUTOR," *UMUAI* 4(1), 21-45, 1994.
- [10] de Rosis, F., De Carolis, N. and Pizzutilo, S., "User Tailored Hypermedia Explanations," *UM'94*, 1994.
- [11] Chin, D., "Intelligent Interfaces as Agents", In Sullivan, J. W. and Tyler, S. W., *Intelligent User Interfaces*, Academic Press, pp.177-206, 1991.
- [12] Cook, R. and Kay, J., "The Justified User Model: A Viewable, Explained User Model", *UM'94*, pp.145-150, 1994.
- [13] Dieterich, H., Malinowski, U., Kuhme, T. and Schneider-Hufschmidt, M., "The State of Art in Adaptive User Interfaces", In Schneider-Hufschmidt, M., Kuhme, T. and Malinowski, U. (eds.), *Adaptive User Interfaces: Principles and Practice*, North Holland, pp.269-283, 1993.
- [14] Hayes-Roth, B., "An Architecture for Adaptive Intelligent Systems," *AI* 72(1/2), 329-365, 1995.
- [15] Moore, J. D. and Swartout, W. R., "Pointing: A Way toward Explanation Dialogue", *AAAI'89*, pp. 457-464, 1989.
- [16] Perez, T., Gutierrez, J. and Lopisteguy, P., "An Adaptive Hypermedia System," *AI-ED'95*, pp.351-358, 1995.
- [17] 강보영, 이상조, "정보검색에서 어휘체인을 이용한 효과적인 색인어 추출 방안", 정보과학회 논문지 29-B-8, pp.584-594, 2002.
- [18] 최중민, "인터넷 정보 추출 에이전트", 정보과학회지 18-5, pp.48-53, 2000.
- [19] 김태훈, 최중민, "사용자 편의의 인터넷 정보검색을 위한 지능형 웹 브라우저 에이전트", 정보과학회 논문지 25-B-7, pp.1064-1078, 1998.
- [20] 소영준, 박영택, "사용자 프로파일 기반 개인 웹 에이전트", 정보과학회 논문지 27-SA-3, pp.248-256, 2000.
- [21] 박재복, 이광용, 조근식, "분산 이형 환경에서의 이동 에이전트를 이용한 정보 검색 시스템", 정보과학회 논문지 29-B-1, pp.30-41, 2002.
- [22] 장병탁, 이종우, 서영우, "학습 에이전트", 정보과학회지 18-5, pp.26-35, 2000.



국형준

1979년 서울대학교 공과대학 졸업(학사)
1983년 Univ. of South Carolina(Computer Science 석사). 1989년 Univ. of Texas at Austin(Computer Science 박사, Post-Doc). 1989년~현재 세종대학교 컴퓨터공학과 부교수. 관심분야는 인공지능, 전문가 시스템, 지능형 교수시스템, 지능형 에이전트