개인화 된 웹 네비게이션을 위한 온톨로지 기반 추천 에이전트

(An Ontology-based Recommendation Agent for Personalized Web Navigation)

정 현 섭 † 양 재 영 † 최 중 민 ††
(Hyunsup Jung) (Jaeyoung Yang) (Joongmin Choi)

요 약 온톨로지(ontology)란 객체(object)들과 이들 사이의 관계의 정의에 의하여 어떤 사실이나 상태를 표현는 지식 표현 방법이다. 본 논문에서는 온톨로지를 이용한 웹 문서 분류와 이를 바탕으로 사용자의 정보 요구에 대한 개인화 된 정보를 제공하는 에이전트를 제안한다. 에이전트는 웹 문서들이 가지는 의미 구조를 계층적 형태로 표현한 온톨로지를 바탕으로 웹 문서를 분류하게 된다. 본 논문에서 온톨로지는 개념(concept)과 개념에 대한 특징(feature), 개념간의 관계(relation) 그리고 문서 분류를 위한 제약조건(constraint)으로 이루어진다. 에이전트는 사용자 프로파일과 문서 식별의 결과를 이용하여 사용자의 정보 요구를 효율적으로 파악하고 사용자의 브라우징을 돕게된다. 또한 에이전트는 선행탐색(look-ahead)방법을 통해 문서를 획득, 문서를 개념으로 표현함으로써 사용자가 좀더 이해하기 쉬운 상위 단계의 웹 문서를 추천하게 된다.

Abstract Ontology is the artifacts for representing the truth or the states of objects by defining objects and their relations. In this paper, we propose an agent that classifies Web documents and provides personalized information towards user's information needs using ontology. The agent uses ontology in which semantic relations on Web documents are represented in a hierarchical form to classify Web documents. In this paper, ontology consists of concepts, features(describing concepts), relations(among concepts) and constraints(among elements in a feature). The agent can capture user's information needs efficiently by using ontology and assist Web navigation by using users profiles and the results of identification of semantic relations in Web documents.

Also, the agent obtains Web documents by a look-ahead search and represents them as concepts, therefore users can understand them easily by receiving recommendations expressed in the form of high-level concepts.

1. 서 론

급속도로 발전하는 인터넷이 가져온 정보의 과부하(information overload)로 인해 사용자는 양적 풍부함을 누리고 있지만 원하는 정보를 찾는데 많은 노력과 시간을 대가로 치러야만 하는 등의 불편함을 감수해야만 한다. 이러한 문제점은 비단 검색엔진 이용 시 뿐만 아니라 일반 웹의 브라우징 시에도 발생한다. 브라우징 시 특히 웹 사이트에 대한 특별한 사전 지식이 없는 사용자는 원하는 정보를 찾는데 상당한 어려움을 겪게 되었으며 이런 여러 문제들을 해결하기 위한 방법들이 관심사로 대두되게 되었다. 이의 해결방안으로 제시된 것이 웹의 "개인화" 이며 이를 위한 여러 방법들이 제시되었다.

하지만 고안된 대부분의 방법들은 다음과 같은 문제점을 안고 있다. 첫째, 웹 문서가 가지는 의미 구조를 고려하지 않고 단순히 단어에 기반하는 방법을 사용하였으며 사용자의 정보 요구가 고정적이라는 것을 가정하고 있다. 둘째, 사용자의 명시적인 관심 여부의 입력

† 비 회 원 : 한양대학교 컴퓨터공학과
hsj@posdate.co.kr
jyyang@csc.hanyangac.kr
†† 종신회원 : 한양대학교 컴퓨터공학과 교수
jmchoi@csc.hanyang.ac.kr
논문접수 : 2001년 12월 24일
심사완료 : 2002년 10월 25일

이 필요하므로 사용자에게 부담으로 작용하게 되며 극히 주관적이라 할 수 있다. 그러므로 명시적 입력이 없는 경우 시스템의 성능이 향상되지 않는다는 문제점을 안고 있다. 셋째, 웹 사용 마이닝 기법은 웹 사용 데이타로부터 필요한 정확한 정보의 추출이 어렵다는 것과 빈번히 변화하는 사이트에서는 효과를 발휘하지 못한다는 단점을 가진다.

따라서 본 논문에서는 이러한 문제의 해결책으로 온톨로지를 이용한 지식 기반 시스템을 제안하고자 한다. 본 논문에서 온톨로지는 "특정 주제(topic)에 대한 간단한 규칙들이나 의미적 연관관계와 단어들을 포함한 지식용어들(knowledge terms)의 집합"으로 정의한다. 온톨로지는 도메인에 관련된 개념(concept)과 개념을 표현하는 특징(feature), 개념간의 관계(relation) 그리고 특징들이 갖는 제약조건(constraint)으로 기술된다.

본 논문에서는 웹 문서들 사이의 의미구조가 명확하고 웹 사이트의 변화가 빈번히 발생하는 뉴스 사이트를 대상으로 개인화 된 정보를 제공하여 사용자의 브라우징을 돕는 에이전트를 제안한다. 에이전트는 웹 문서(기사)들이 가지는 의미 구조를 표현한 온톨로지를 바탕으로 웹 문서를 분류하게 되며 결과적으로 문서들이 가지는 의미론적 내용과 관계의 식별을 가능하게 해준다. 이를 통해서 브라우징 시 사용자의 정보 요구를 좀더 정확하게 파악할 수 있으며 사용자의 정보요구에 맞는 상위 단계(high-level)의 정보 제공이 가능하게 된다. 또한 선행탐색(look ahead)방법을 사용하여 문서를 획득, 분류하여 문서에 대한 사용자의 관심 사항에 부합되는 지역적으로 최적(locally optimal)의 문서를 제시함으로써 사용자의 브라우징을 돕게 된다. 여기서 지역적으로 최적의 문서라는 것은 사용자가 네비게이션을 통해 도달할 수 있는 전체 공간을 고려한 것이 아니라 일정 범위만을 고려할 경우 그 안에서 사용자의 관심사항에 가장 잘 부합되는 문서를 말한다. 그리고 이를 통해 유도된 사용자 프로파일(profile)을 유지함으로써 좀더 사용자의 요구를 잘 반영한 개인화 된 뉴스 기사를 제공하게 된다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구를 통하여 여러 개인화 된 정보 추천 기법을 알아보고 3장에서는 네비게이션이 가진 문제 공간과 추천에이전트가 갖추어야 할 조건에 대해서 살펴본다. 그리고 4장에서는 본 논문에서 사용한 온톨로지에 대해서 알아보며 5장에서는 온톨로지를 이용한 문서 분류와 정보 추천 방법을 살펴보고 6장에서는 시스템 구현과 웹 문서 분류 실험에 대한 결과를 살펴보고 마지막으로 7장에서는 결론 및 향후 연구 방향을 제시한다.

2. 관련 연구

오래전부터 사용자의 브라우징을 도와주는 방법에 대한 연구가 있어왔다. 다중의 사용자로부터의 경험을 통해 학습과 강화 학습(reinforcement learning)을 통한 정보 추천 능력을 향상시키는 WebWatcher[1] 와 사용자의 관심사항을 기록한 프로파일을 여러 기계학습 방법을 통해 학습하여 웹 문서에 대한 흥미 여부를 식별하는 Syskill & Webert[2] 라는 시스템 그리고 사용자의 행동을 추적하고 자발적으로 링크를 선행 탐색하여 사용자의 다음 브라우징 행위를 추측, 흥미가 있을 만한 아이템들을 추천하는 인터페이스 에이전트인 Letizia[3]가 있다. 그리고 WebMate[4]는 코퍼스(Cor pus)와 키워드 확장 방법을 이용하여 프로파일을 확장하여 개인화 된 정보를 제공한다. 또한 Mobasher는 [5, 6]에서 좀더 효과적인 개인화 된 정보 제공을 위해 웹 사용 마이닝(usage mining) 기법을 이용 하였으며 이 결과로 생기는 사용 프로파일과 사용자의 정보 요구 성향을 기반으로 웹 문서를 추천한다. OBIWAN[7]은 개인 온톨로지(개인의 브라우징 계층)을 통해 웹 사이트들에 대한 브라우징을 가능하게 하는 온톨로지를 기반으로 한 분산구조의 에이전트이다. 본 논문이 온톨로지의 개념을 사용한다는 것에 있어서는 OBIWAN과 같지만 온톨로지의 생성, 이용 그리고 추천 방식 등에서는 많은 차이를 가지고 있다.

하지만 대부분의 시스템은 웹 문서가 가지는 의미 구조를 고려하지 않았으며 사용자의 정보 요구가 고정적이라는 것을 가정으로 하고 있다. 또한 사용자의 명시적인 관심 여부의 입력이 필요 하므로 사용자에게 부담으로 작용하게 되며 극히 주관적이라 할 수 있다. 따라서 명시적 입력이 없는 경우 시스템의 성능이 향상되지 않는다. 또한 웹 사용 마이닝 기법은 웹 사용 데이타로부터 필요한 정확한 정보의 추출이 어렵다는 것과 빈번히 변화하는 사이트에서는 효과를 발휘하지 못한다는 단점을 가진다. 그리고 특히 OBIWAN의 경우 문서 변화에 민감하지 못하며 일괄된 형태의 출력을 제공하므로 자연스러운 브라우징이 되지 못한다는 단점을 가지고 있다.

따라서 본 논문에서는 웹 문서들 사이의 의미구조가 명확하고 웹 사이트의 변화가 빈번히 발생하는 뉴스 사이트를 대상으로 개인화 된 정보를 제공하여 사용자의 브라우징을 돕는 에이전트를 제안한다.

3. 네비게이션을 돕는 추천 에이전트

3.1 네비게이션 문제의 기술

웹 상의 문서들은 하이퍼링크로 서로 연결되어 있는 하이퍼 그래프(hyper-graph)형태로 이루어져 있다. 각각의 하이퍼링크는 하나의 문서를 가리키며 하나의 문서는 여러개의 외부 및 내부 문서에 존재하는 하이퍼링크에 의해 가리켜질 수 있다. 일반적으로 문서를 읽는다는 의미는 현재 위치에서 존재하는 하이퍼링크를 따라 네비게이션을 할 것인지 그렇지 않을 것인지의 여부를 결정한다는 의미로 해석될 수 있다. 그러므로 계속적인 네비게이션 시 어떤 경로로 이것이 이루어지는가가 중요한 사안이 된다.

네비게이션의 문제는 다음과 같은 특징을 갖고 있다. 첫째, 사용자의 네비게이션 시작 위치는 그래프 상의 어디라도 될 수 있다. 둘째, 사용자의 정보요구는 고정되어 있는 것이 아니라 그래프 상에서의 노드의 위치마다 변하게 된다.

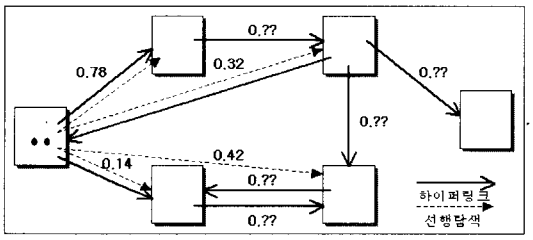


그림 1 네비게이션이 가지는 상태(state)들의 공간

그림 1은 네비게이션이 가진 상태들의 공간과 본 논문이 제시하는 방법을 도식화 한 것으로 사용자의 네비게이션 위치에서 선행 탐색 방법의 사용을 통하여 지역적으로 최적의 문서를 찾기 위한 과정을 나타낸 것이다. 수치는 사용자의 정보 요구와 문서간의 유사도를 나타낸다.

3.2 추천 에이전트의 요구사항

기존의 사용자 네비게이션을 돕는 에이전트들이 가진 문제점들을 극복하고 보다 나은 웹 문서의 추천을 위해서 추천 에이전트가 가져야할 요구사항은 다음과 같다. 첫째, 웹 문서들이 가지는 의미적 관계(semantic relations)의 고려를 통한 사용자 정보 요구의 획득과 이에 따른 프로파일의 유도가 이루어져야 한다. 둘째, 빈번히 문서의 내용이 변화하는 사이트에 대한 유연한 대처가 필요하다. 셋째, 사용자 정보 요구는 고정적인 것이 아니라 점진적이고 지속적으로 변화하는 것으로 이에 대한 반영이 필요하다. 넷째, 웹 사이트에서 사용자의 자연스러운 브라우징을 가능하게 하는 인터페이스를 통한 추천이 이루어져야한다. 다섯째, 사용자의 유사 피드백의 명시적인 입력을 통한 정보요구 획득이 아닌 암시적인 방법을 통한 자연스러운 사용자 정보요구의 입수가 이루어져야하며 마지막으로 시간이 경과에 따른 지속적인 학습으로 추천의 성능 향상을 가져와야 한다.

4. 지식기반으로서의 온톨로지

4.1 온톨로지의 정의

온톨로지는 과거에는 철학분야에만 국한되어 사용되었으나 근래에는 컴퓨터공학 분야에 적용되어 널리 사용되고 있다. 특히 최근에는 지식공학, 지식 표현, 데이타베이스 디자인, 정보 모델링, 정보 통합/관리/조직, 에이전트 기반 시스템 등 다양한 분야에 적용되고 있다.

인공지능(AI) 학문에 있어서 온톨로지는 "개념화의 명세(specification of a conceptualization)"로써 정의된다. 이는 "engineering artifact"로써 어떤 사실을 기술하기 위해 필요한 object(객체)의 집합인 "vocabulary (universe of discourse)"와 이의 객체들간의 관계인 relation과 function들의 집합으로 이루어진다[8, 9, 10, 11]. 쉽게 말해서 온톨로지는 객체의 집합과 객체들간의 관계의 정의에 의해 어떤 사실이나 상태를 표현하고자 하는 지식 표현 기법이다.

4.2 개념화(Conceptualization)

개념화는 세상에 존재하는 것으로 간주되거나 추정되는 모든 객체나 객체간의 상호 관계를 포함한다. 여기서 객체란 실제적인 것이나 추상적인 것 혹은 단일체나 복합체 그리고 허구적인 것들까지 포함한다. 간단히 말해서 객체란 우리가 말하고자 하는 것에 대한 모든 것일 수 있다[12].

예를 들어 그림 2과 같은 블록 세계의 한 장면를 고려해 보자. 이 경우 다섯 개의 블록으로 이루어진 다음과 같은 집합이 개념화에 필요한 "universe of discourse"가 된다.

$\{a,b,c,d,e\}$

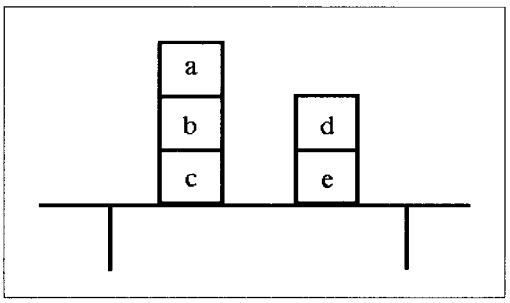


그림 2 블록 세계에서의 한 장면

Universe of discourse내 객체간의 관계를 나타내는 것으로 function과 relation이 있다. 예를 들어 한 블록을 자기 위에 위치하는 블록에 매핑시키는 function "hat"이 존재할 경우 이 function에 해당하는 tuple들은 다음과 같다.

$$\{<b,a>,<c,b>,<e,d>\}$$

또한 블록 세계의 공간적인 개념화를 고려할 경우, 한 블록이 어떤 한 블록의 위에 있다는 의미의 "on"의 경우 tuple은

$$\{<a,b>,<b,c>,<d,e>\},$$

자기 블록이 다른 어떤 블록위에 있다는 의미의 "above"의 경우

$$\{<a,b>,<b,c>,<a,c>,<d,e>\},$$

자기 블록위에 블록이 없다는 의미의 "clear"의 경우

$$\{a,d\}$$

그리고 테이블 위에 놓여있는 블록일 경우만 유용한 "ontable"의 relation일 경우

$$\{c,e\}$$

로 기술이 된다.

다시 말하면 유한의 universe of discourse가 있고 n-ary의 relation이 존재할 경우, b 크기의 universe of discourse에서는 최대 b^n개의 유일한 n-tuple이 생기며 n-ary relation의 가능한 집합중의 하나가 된다.

형식적으로 개념화는 다음과 같이 universe of discourse와 개념화에서 중요시된 function들의 집합인 "functional basis set", 그리고 relation의 집합인 "relational basis set"으로 구성된 triple의 형태로 이루어진다[12].

$$<\{a,b,c,d,e\},\{hat\},\{on,above,clear,ontable\}>$$

4.3 본 논문에서의 온톨로지 구조

본 논문에서는 온톨로지를 "특정 주제에 대한 간단한 규칙들이나 의미적 연관관계와 단어들을 포함한 지식 용어들의 집합"으로 정의하기로 한다. 본 논문은 웹 문서상의 의미적 계층구조의 표현과 웹 문서의 분류를 용이하게 하기 위해서 온톨로지를 개념과 특징, 관계 그리고 제약조건으로 구성된 노드(node)들로 표현한다<그림 3>.

온톨로지에서 특징은 개념을 표현할 수 있는 단어나 구의 집합으로 이루어지고 관계는 노드(개념)간의 관계를 표현하며 "isA", "partOf", "hasPart"로 정의된다. "isA"는 개념간의 일반화(generalization)의 의미로 모든 링크는 "isA"관계를 가지며 "partOf"와 "hasPart"는 서로 상반되는 의미로써 노드간의 포함관계를 나타낸다. 그리고 제약조건은 "isRelatedTo", "followedBy"로 표현되며 밀접한 관계를 가지는 특징들이나 요소들을 묶어 놓음으로써 단어기반의 분류가 가지는 모호성을 해결하고 분류의 정확성을 기하기 위해 중요한 의미를 지닌다. 온톨로지에서 한 노드의 표현을 보면 그림 4와 같다. 그림 4에서 온톨로지는 개체를 설명하기 위해 사용되는 특징을 나타내는 "features"태그와 다른 개체들과의 관계를 나타내는 "realtions"태그 그리고 각 개체가 활성화되기 위해서 필요한 제약조건을 나타내는 "constraints"로 구성되어 있다.

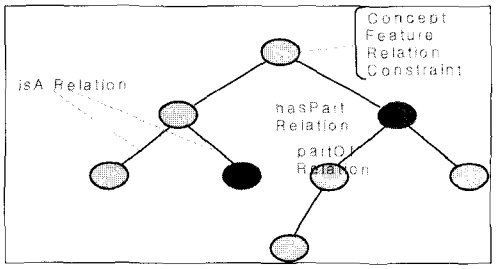


그림 3 온톨로지의 전반적인 형태

```
<Ontology>
    <OntoDef>
    <OntoName>history</OntoName>
    <Features>basebal,histori,world,seri,
            award,record,hall,fame,star,post,mvp
    </Features>
    <Relations> hasPart (baseball-major league-history)
    </Relations>
    <Constraints> (star) followedBy (all)
    </Constraints>
    </OntoDef >
    . . .
</Ontology>
```

그림 4 baseball-history 노드의 표현

본 논문에서 온톨로지는 XML로 표현되었으며 구축된 온톨로지는 약하게 구조화된 온톨로지(weakly structured ontology)라 할 수 있다. 약하게 구조화된 온톨로지에서는 개념 값(class value)이나 클래스 인스턴스(class instance), superclass-subclass, part-whole 등의 개념적인 관계가 명확히 구분되지 않는 특성을 갖고 있다.

5. 온톨로지를 이용한 개인화 된 문서의 추천

5.1 온톨로지를 이용한 문서 분류

웹 문서를 분류하는 과정은 개념 계층(온톨로지)상의 노드에 문서를 매핑(mapping)하는 것을 말한다. 분류시 문서의 매핑 절차는 온톨로지의 루트(root)노드에서부터 시작하여 리프(leaf)노드의 방향으로 진행되며 문서는 가장 유사하다고 판단된 노드에 매핑되게 된다. 그리고

부모 노드를 만족시킨 문서에 한해서 자식 노드들을 고려하게 된다[13]. 즉, 부모 노드에서 유사도의 계산이 이루어질 경우만 자식 노드로 분류과정이 진행되게 되며 그렇지 않을 경우 마지막으로 문서가 지나간 노드에 매핑되게 되므로 항상 리프 노드에 문서가 매핑되지는 않는다. 이 방법을 통해 적은 수의 노드 지식으로 온톨로지를 표현할 수 있으며 계층적인 분류를 통해 좀더 정확한 분류가 가능하게 된다.

분류를 위한 유사도 계산은 수식 (1)을 이용하였으며 문서는 가장 큰 유사도를 가지는 하나의 노드에 할당하게 되므로 한 문서는 최종 하나의 클래스(class)로 분류되게 된다.

$$Sim(Node, d) = \frac{\sum_{i=0}^{N} freq_{i,d} / \max_{i,d}}{N} \times \frac{V_d}{V} \qquad (1)$$

여기서 N은 한 노드에서의 총 특징의 수 이며 freqi,d는 문서 d에서 매칭되는 특징 i의 빈도수를 말하고 maxI,d는 문서 d에 의해 가장 많이 매칭되는 특징의 빈도수를 나타낸다. V는 제약조건의 수를 그리고 Vd는 문서 d에 의해서 만족되는 제약조건의 수를 말한다. 문서 분류과정에서 관계의 사용은 다음과 같다. 문서 분류시 노드가 "partOf"나 "hasPart"에 의해 다른 노드와 관련이 있을 경우 관련된 노드를 분류과정에 포함시켜서 유사도 계산을 행하게 된다. 또한 프로파일 생성 시에도 문서가 최종 온톨로지에 매핑된 경로 뿐만 아니라 관련된 노드에 이르는 경로까지 동시에 저장되게 된다. 이로써 문서의 좀더 정확한 분류와 사용자의 관심사항을 반영한 상위 단계 문서의 추천이 가능하게 된다. 전체적인 웹 문서 분류 과정은 그림 5와 같다.

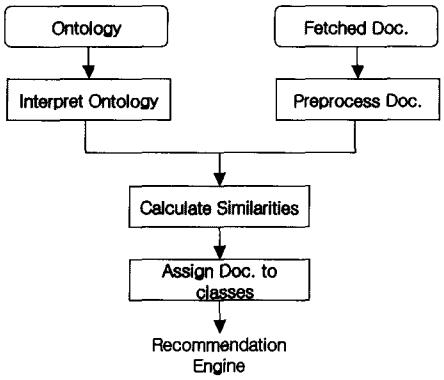


그림 5 온톨로지를 이용한 웹 문서의 분류 과정

5.2 추천 에이전트 시스템 구조

본 논문이 제안하는 온톨로지 기반 추천 에이전트의 구조는 그림 6와 같다. 사용자의 일반 웹 브라우져를 통해 문서를 요구하게 되고 이 요구는 Proxy Server를 통해 시스템에 전달된다. 요청한 문서는 HTML 테그의 제거와 불용어(stopword) 제거 그리고 스테밍 처리의 전처리 과정을 거친다. Interpreter와 Classifier는 구축된 온톨로지를 바탕으로 온톨로지 해석과 전처리 과정을 거친 문서들과의 유사도 계산을 통한 분류작업을 행하게 된다. 분류작업을 거친 문서들은 추천엔진의 입력으로 사용된다. 한편 현재 온톨로지는 수동으로 구축되며 추가, 삭제, 갱신 등에 관련된 온톨로지 관리는 Ontology Editor 그림 7를 통해서 가능하다. 이렇게 분류과정이 끝난 문서들에 대해서 Recommendation

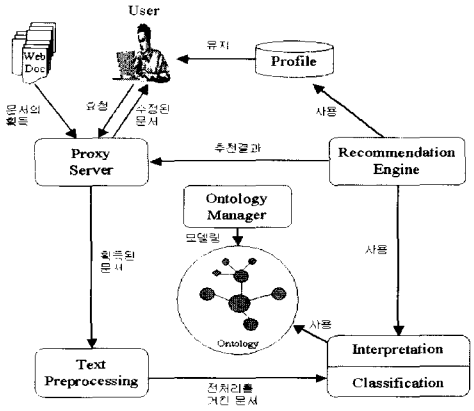


그림 6 온톨로지 기반 추천 시스템 구조

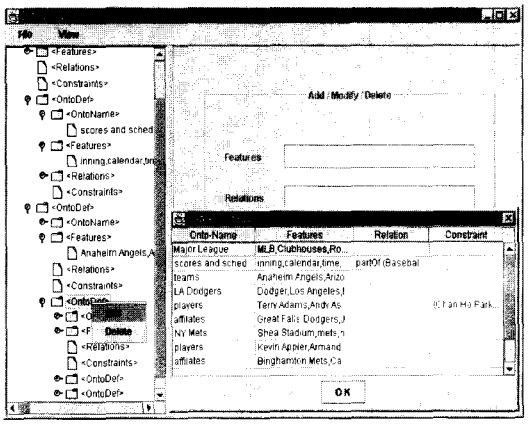


그림 7 온톨로지 에디터 및 뷰어

Engine 문서들에 대한 클래스 즉, 경로(path)와 사용자의 관심사항이 학습된 프로파일의 경로를 이용하여 추천 점수(Recommendation Score)를 계산하게된다. 그 결과 상위 랭크된 최대 10개의 문서들의 URL과 추천점수를 추가한 수정된 웹 문서가 프락시 서버를 통해 다시 웹 브라우져로 보낸다. 또한 추천 결과에 대한 사용자의 행위 즉, 문서의 선택과 관련한 정보를 프로파일로 유지하게 된다.

5.3 하이퍼링크의 분석

웹 문서의 추천 과정이전에 우선적으로 고려되어야 할 사항으로는 웹 문서에서의 하이퍼링크 분석이다. 프레임으로 이루어진 웹 사이트에서는 사용자가 요청한 문서가 프레임의 특정 부분에서만 변화하므로 주위의 메뉴에 해당하는 부분이나 기타 광고 부분을 고려할 필요가 없다. 하지만 불행하게도 대부분의 뉴스 기사를 제공하는 사이트는 프레임을 갖지 않는다. 따라서 웹 문서가 가진 모든 하이퍼링크를 선행 탐색한다는 것은 많은 시간을 요구하며 실시간 문서 추천에있어서 큰 장애가 아닐 수 없다.

따라서 본 논문에서 이를 해결하기 위해 사용자가 방문한 사이트에 대한 URL 제거 리스트(list)를 계속 유지하게 된다. 그리고 이를 이용하여 선행 탐색 시 저장된 URL제거 리스트에 존재하지 않는 하이퍼링크만 탐색하게 된다. 또한 이미 사용자가 요청한 사이트 즉, 사용자가 이전에 브라우징한 문서의 하이퍼링크 또한 추천대상에서 제외된다.

5.4 프로파일을 이용한 문서 추천

개인화 된 정보 추천을 위해 본 논문은 사용자의 관심사항이 반영된 프로파일을 이용한다. 프로파일은 그림 8와 같이 사용자가 요청한 문서가 분류된 후 매핑되었던 노드에 이르는 경로와 그 문서에 대한 사용자의 요청 횟수의 쌍으로 이루어진다. 따라서 사용자는 온톨로지에서 유도된 경로들과 그에 해당하는 문서의 요청 횟수를 프로파일로 가지게 되므로 사용자 각자의 가중치가 부여된 온톨로지(weighted ontology)[14, 15]를 가진다고 할 수 있다.

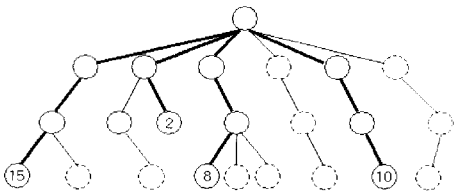


그림 8 사용자 프로파일 구조

굵은 선으로 표현된 경로는 온톨로지에서 유도된 프로파일 상에서 사용자의 문서 요청에 대한 경로가 존재함을 의미하며 숫자는 노드에 매핑된 문서에 대한 요청 횟수로써 문서 요청 시 계속적으로 증가하게 된다. 추천 대상이 되는 문서들은 사용자의 네비게이션 위치에서 선행 탐색방법을 사용하여 획득한 문서들이 된다.

현재 사용자의 위치에서의 문서를 d라고 하고 선행탐색 결과 획득된 문서들의 집합을 N이라 한다. 문서 거리(Document Distance)의 계산은 프로파일을 고려하지 않고 문서와 문서간에 이루어진다<수식 2>. 즉 현재 사용자가 읽고 있는 문서가 매핑된 개념과 선행탐색의 결과로 획득된 문서들이 매핑된 개념간의 거리를 말한다.

$$DD_{(d,d_i)} = \frac{1}{\log_2(dist(d,d_i)+1)+1} \quad \text{where } d_i \in N \qquad (2)$$

dist(d,di)는 문서 d와 di간의 경로의 길이차이를 말한다. 프로파일을 p라고 할 경우 프로파일 거리(Profile Distance)는 선행탐색으로 획득된 문서들에 대해 프로파일을 기반으로 계산된다.

$$PD_{(d,N)} = \max_i \left(\frac{\frac{freq_{p,i}}{\max freq_{p,j}}}{\log_2(Cdist(d,p)+1) \times \log_2(Wdist(d,d_i)+1)+1} \right) \qquad (3)$$

where $d_i \in N$

여기서 $freqp.i/maxfreq_{p,j}$은 프로파일에서 해당 문서가 노드에 이르는 경로에 대한 사용자의 방문 횟수를 최대 방문회수로 정규화 한 점수를 말한다. $Wdist(d,di)$는 웹 거리(Web distance)로써 하이퍼링크로 연결된 실제 웹 문서들간의 거리, 즉, 사용자가 현재 보고있는 문서와 선행 탐색의 결과로 획득한 문서들간의 거리를 말하며 물리적인 웹 문서간의 거리가 가까울수록 사용자가 원하는 정보일 가능성이 높다는 것을 가정한 것이다. $Cdist(d,p)$는 개념 거리(Concept distance)로써 사용자 프로파일의 계층상에서 문서와 사용자의 관심사항이 반영된 노드(문서 요청이 발생했던 노드)와의 거리를 말한다.

최종적으로 문서에 대한 추천 점수의 계산은 다음과 같다.

$$RS = \sqrt{DD \times PD} \qquad (4)$$

이 경우 프로파일의 계층에 문서가 매핑되는 단계에 따라 차등적으로 수치를 부여한다. 이로써 실제 사용자의 관심사항이 반영된 노드의 상위 레벨에 위치한 문서까지도 추천이 가능하게 된다.

마지막으로 사용자는 시스템이 관련 있다고 판단된 문서들에 대한 링크와 관련 정도를 나타내는 추천 점수가 추가된 수정된 웹 문서를 일반 브라우져를 통해서

최종 결과로 받게 된다. 프로파일을 이용한 웹 문서 추천 과정은 그림 9와 같다.

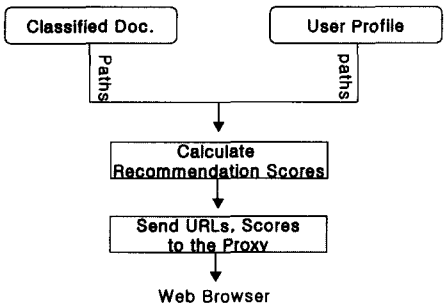


그림 9 프로파일을 이용한 웹 문서 추천 과정

6. 구현 및 실험

6.1 시스템 구현

현재는 야구에 대해서만 온톨로지를 구축하였으며 온톨로지를 바탕으로 한 웹 문서의 분류와 추천에 대해서 살펴보면 다음과 같다. 예를 들어 그림 10의 웹 문서[16]를 사용자가 요청한 경우를 고려해 보자. 우선 에이전트는 요청한 문서가 가지는 하이퍼링크들을 추출한 후 페이지마다 동일하게 존재하는 하이퍼링크들을 제거하게 된다. 이 과정을 거친 후 필요한 하이퍼링크에 대한 선행 탐색이 이루어지게 된다. 그림 11은 사용자가 요청한 페이지에서 선행탐색을 하여 획득한 문서들에 대한 분류 결과(URL, 클래스, 유사도)를 나타낸 것이다.

어느 사용자의 프로파일이 표 1와 같이 존재한다고 할 경우 문서추천에 대해 살펴보자. 그리고 사용자가 그림 10의 웹 문서를 요청했다고 가정한다. 사용자 프로파일

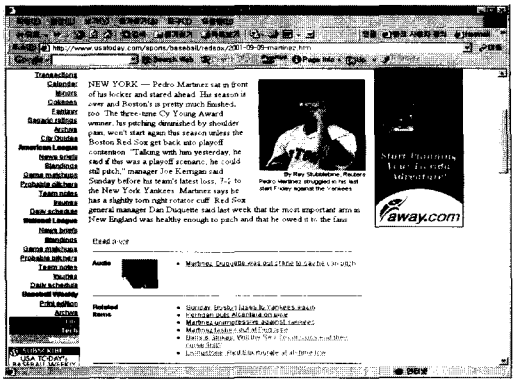


그림 10 사용자가 요청한 야구 관련 기사 예제

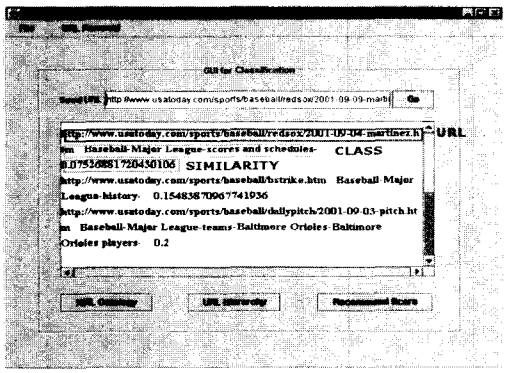


그림 11 웹 문서 분류 실험을 위한 GUI

과 웹 문서의 분류작업을 통해 분류된 문서들 그리고 사용자가 현재 브라우징하고 있는 문서와의 비교를 통해 상위 랭크된 문서들에 대한 링크를 사용자에게 추천

표 1 사용자 프로파일 예제

Path	Count
Sport-Baseball-Major League-history	1
Sport-Baseball-Major League-scores and schedules	11
Sport-Baseball-Major League-teams	5
Sport-Baseball-Major League-teams-LA Dodgers	8
Sport-Baseball history	1
Sport-Baseball-Major League-teams-LA Dodgers-players	14
Sport-Baseball Major League-teams-NY Mets-players	2
Sport-Baseball-Major League-teams-NY Mets	1

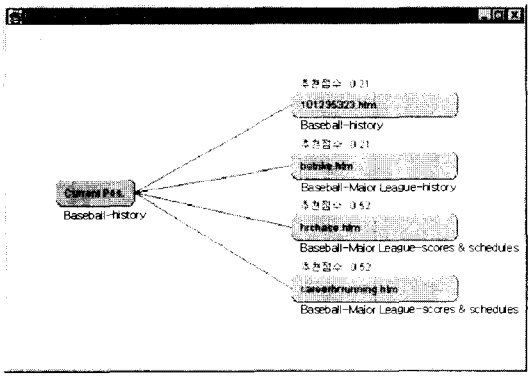


그림 12 프로파일을 이용한 상위 4개의 문서 추천 예제

하게 된다<그림 12>.

6.2 웹 문서 분류 실험

모든 웹 문서는 온톨로지상의 개념에 매핑(문서의 분류)되므로 온톨로지의 루트에서 매핑된 노드에 이르는 하나의 경로를 가지게 된다<그림 13>. 이 경로는 프로파일의 경로와 함께 추천 시 사용된다. 그리고 본 논문이 추구하고자 하는 상위 단계 문서를 추천하기 위해서는 정확한 문서의 분류, 즉 온톨로지의 노드에 이르는 경로의 정확한 식별이 전제되어야 한다. 달리 말하면, 문서가 매핑되는 개념 각각을 표현하는 특징의 선택이 추천과정에 선행되는 핵심 요소인 것이다. 문서 추천의 성능의 측정은 사용자가 추천된 문서의 올바름의 여부를 판단하게 되므로 지극히 주관적이라 할 수 있으며 따라서 성능의 측정이 어렵다. 따라서 본 논문에서는 추천의 정확도를 평가하는 대신 추천의 정확도를 가져오는데 있어서 핵심 요소인 문서 분류의 정확성을 평가하였다. 그리하여 수동으로 선택된 특징들로 구성된 온톨로지와 IDF[13]로 추출된 특징들로 구축된 온톨로지를 이용한 분류의 정확도 실험을 통하여 특징선택의 방법을 강구해 보았다.

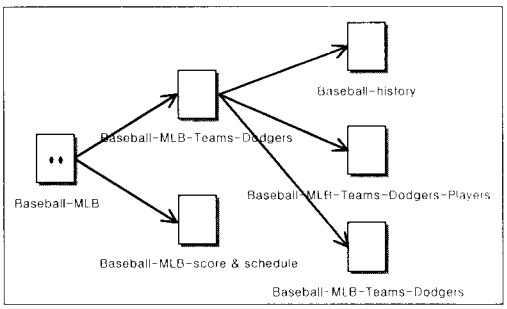


그림 13 온톨로지를 이용한 문서의 추천 주석(annotation)

일반적으로 문서 분류에 좋은 성능을 발휘한다고 알려진 기계학습 방법으로는 Naive Bayes classifier나 SVM(Support Vector Machine)이 있다[17]. 하지만 이들 방법은 명시적으로 학습의 목표가 주어지는 감독학습(supervised learning) 방법으로서 미리 분류된 실험 데이타가 존재하여야만 하며 분류 시 많은 시간을 요구하므로 본 논문이 제안하는 에이전트에는 적절하지 않은 방법이다. 반면 본 논문에서 제안하는 방법은 분류에 있어서 적은 시간을 요구하므로 실시간 분류가 가능할 뿐만 아니라 온톨로지의 내용에 따라 정확한 분류가 가능하며 시간이 지날수록 성능이 향상된다는 장점을 가진다.

본 실험에서는 온톨로지를 이용한 문서 분류의 정확도 측정을 위해 야후 검색엔진[18]을 이용하였다.

Yahoo!내 recreation/sports/baseball에 존재하는 카테고리들내의 문서들을 실험 대상으로 하였으며 온톨로지의 개념명(클래스)과 동일한 카테고리들(history, major league, minor league, ...)내에 존재하는 문서들을 이용하였다. 그리하여 문서들이 온톨로지의 다소 큰 범주인 야구 클래스로 분류되는지 그리고 온톨로지내의 동일한 클래스로 분류되는지의 여부를 실험하였다.

실험에 사용된 데이타의 개수는 총 415개로써 history카테고리에서 53개, amateur카테고리에서 78개, college&university에서 122개, minor league에서 108개 그리고 major league에서 54개의 문서를 사용하였다.

6.3 IDF를 이용해 수동 구축된 온톨로지를 통한 웹 문서 분류

다음은 IDF를 통해 문서들에서 단어들을 추출하고 이로부터 수동으로 선택된 특징으로 구축된 온톨로지를 이용하여 문서 분류의 정확도를 실험하였다. 야후 검색엔진에서 온톨로지의 클래스과 같은 카테고리내 존재하는 웹 문서들을 기반으로 단어들의 IDF를 구하게 되며 이 결과 하위 랭크된 50개의 단어에서 다시 10개씩의 단어를 특징으로 하여 온톨로지의 리프 노드를 구성하였다. 여기서 IDF의 하위 랭크된 단어들을 사용한 이유는 수식 (1)을 적용하기 위하여 빈번히 여러 문서에서 출현하는 단어들을 특징으로 선택하기 위함이다. 구축 순서는 온톨로지에서의 최하위 노드부터 시작되며 하위 노드들의 특징을 이용하여 상위 노드를 구성하는 방식을 사용하였다<그림 14>. 그러므로 상위노드는 하위 노드들을 대표한다고 할 수 있다. 리프 노드는 문서들에서 IDF를 이용해 구성되며 리프 노드를 제외한 상위 노드들은 하위 노드의 수를 고려하여 특징들이 선택된다. 즉, 특징이 가진 IDF값을 노드의 수로 나누어 하위 랭크된 10개의 특징들을 선택하게 된다<표 2>.

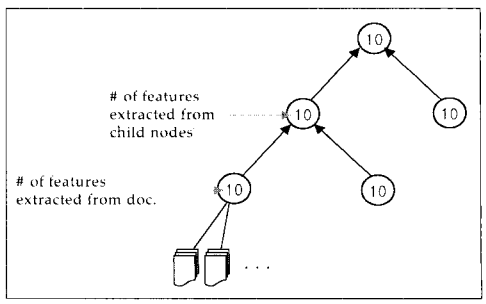


그림 14 정보검색 기법을 이용한 온톨로지 구축

표 2 IDF를 기반으로 수동 선택된 10개의 특징

카테고리	하위 랭크된 특징 10개
Baseball/	basebal,team,leagu,season,player,game,nation,mlb,major, schedul
Baseball/College Baseball/	athlet,coach,schedul,univers,field,colleg,ncaa,countri, student,camp
Baseball/Amateur Baseball/	basebal,sponsor,tournament,coach,school,plai,organ,top, champion,club
Baseball/History/	leagu,home,player,histori,game,hit,bat,run,picther,mark
Baseball/Minor League/	leagu,ticket,minor,fan,box,stadium,club,manag,start,affili
Baseball/Major League/	league,mlb,major,world,nation,seri,nl,al,top,roster

6.4 웹 문서 분류 실험 결과

표 3, 4는 각 각 문서 분류 후 문서가 야구 클래스로 분류된 비율과 Yahoo!의 카테고리내 문서가 온톨로지상의 동일한 클래스로 분류된 비율을 나타낸다. 야구 클래스로 분류된 비율이 82.7%, 82.5%로 분류 정확도가 거의 같게 나타났다. 그리고 동일 클래스(야구 클래스 바로 하위 레벨)로의 분류된 비율이 각 각 44.2%, 42.6%로써 IDF를 기반으로 수동 구축한 경우가 다소 높게 나타났다. 이를 통해 다른 사이트들을 참고로 하여 수동 구축된 온톨로지와 IDF를 통해 추출된 단어를 기반으로 구축된 온톨로지의 분류 정확도가 거의 같게 나타났음을 알 수 있다.

문서들이 동일 클래스로 분류된 비율이 낮음은 다음의 이유들에서 기인한 것이다. 첫째, Yahoo!의 카테고리내 문서들은 서퍼(surfer)에 의해 내용 판단 후 수동으로 분류된 것이다. 둘째, Yahoo!를 포함한 검색엔진에서 문서의 분류의 기준 혹은 단위는 페이지 각각이 아니라 사이트내의 전체 문서들이다. 셋째, 본 실험은 소수의 텍스트로 이루어진 문서나 이미지를 많이 포함한 문서까지도 실험 대상으로 하였다. 따라서 본 논문에서의 제안한 단어에 기반한 분류 방법이 Yahoo!의 문서에 대해 낮은 분류의 정확도를 가져왔다. 그리고 특히 history와 minor league에 해당하는 문서의 분류 비율이 낮게 나타난 이유는 다른 클래스와 이들을 구분 지울 수 있는 특징의 추출이 어려웠음을 의미한다.

7. 결론 및 향후 연구방향

본 논문에서는 온톨로지를 이용하여 개인화 된 네비게이션을 돕는 추천 에이전트를 제안하였다. 온톨로지를 이용하여 웹 문서들이 가지는 의미적 관계를 개념 구조로 표현하고 에이전트는 구축된 온톨로지를 이용하여 웹 문서를 분류하게 된다. 또한 에이전트는 온톨로지를 바탕으로 사용자 정보요구를 효과적으로 파악하며 사용자가 브라우징 시 프로파일을 유지함으로써 사용자 관심사항을 좀더 잘 반영한 상위 단계의 정보를 추천하게 된다.

표 3 야구 클래스로의 분류 여부 실험 결과

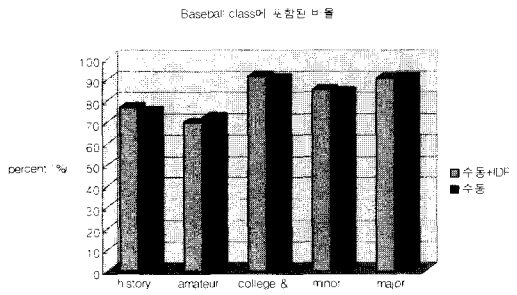
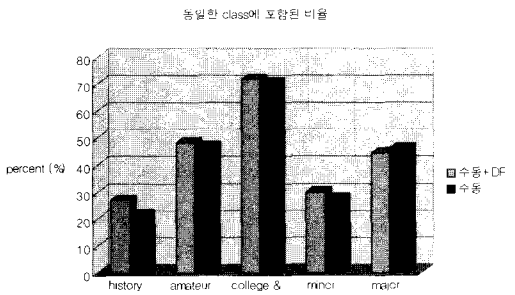


표 4 동일 클래스로의 분류 여부 실험 결과



웹 문서 분류 실험결과 Yahoo!에 존재하는 문서에 대해서는 정보 검색의 IDF와 수동을 결합한 특징의 선택 방법과 수동을 통한 특징선택 방법이 거의 비슷한 성능을 보였다. 이는 결론적으로 IDF에 의한 특징의 선택이 온톨로지 구축에 도움이 될 수 있으며 문서 분류 성능의 향상을 위해서는 각 클래스를 특징 지울 수 있는 특징의 선택이 필수적이라는 것을 말한다. 비록 본 논문에서는 많은 실험 데이타를 사용하지 않고 특정 도메인의 실험 데이타를 사용하였지만 어느 도메인에 국한되지 않고 전 도메인에 걸쳐 좋은 성능을 가져올 것으로 생각된다.

본 논문이 제안하는 에이전트가 가지는 가장 큰 문제점으로는 선행탐색 시 소요되는 시간으로 인한 실시간 추천의 지연을 들 수 있으며 잘못된 제약조건 및 특징의 사용으로 그릇된 문서 분류 결과를 가져는 경우 온톨로지 편집자의 끊임없는 노력이 필요하다는 것이다.

향후 연구는 크게 여섯 가지로 나누어 볼 수 있다. 첫째 온톨로지 구축의 자동화 혹은 반 자동화를 위하여 웹 문서에서부터 의미적 개념이나 관계를 자동으로 추출하는 방법에 대한 연구와 이에 관련된 인터페이스의 개발을 들 수 있다. 둘째로는 현재의 단일 클래스 문서 분류를 확장하여 다중 클래스 분류를 가능하게 하는 방법에 대한 연구이다. 셋째로는 좀더 효율적인 온톨로지 표현과 문서 분류 방법에 관한 연구이다. 넷째로는 온톨로지의 기본 목적인 정보 공유의 차원에서의 협동(collaborative) 시스템의 구축을 들 수 있다. 다섯째로 위에서 언급한 선행탐색 시 걸리는 시간 문제를 해결하기 위한 방안으로 링크된 하이퍼텍스트의 분석을 통한 의미 있는 하이퍼링크의 추출이며 마지막으로 온톨로지와 사용자 프로파일을 이용한 개인화 된 문서 필터링(filtering) 시스템에 대한 연구가 있을 수 있다.

참 고 문 헌

[1] T. Joachims, D. Freitag, and T. Mitchell. Webwatcher: A Tour Guide for The World Wide Web. *In IJCAI'97*, pp. 770-777, 1997.

[2] M. Pazzani, J. Muramatsu and D. Billsus. Syskill & Webert: Identifying Interesting Web Sites. In *Proc. of 13th Natl. Conf. on Artificial Intelligence*, pp. 54 61, 1996.

[3] H. Lieberman. Letizia: An Agent that Assists Web Browsing. *In IJCAI'95*, pp. 475-480, 1995.

[4] L. Chen and K. Sycara. Webmate : A Personal Agent for Browsing and Searching. In *Proc. of 2nd Intl. Conf. on Autonomous Agents*, pp. 132-139, 1998.

[5] B. Mobasher, H. Dai, T. Luo, Y. Sung, and J. Zhu. Integrating Web Usage and Content Mining for More Effective Personalization. In *Proc. of First Intl. Conf. on E Commerce and Web Technologies (ECWeb2000)*, pp. 165-176, 2000.

[6] B. Mobasher, R. Cooley, and J. Srivastave. Automatic Personalization Based on Web Usage Mining. Technical Report TR99010, Dept. of Computer Science, DePaul University, 1999.

[7] J. Chaffee and S. Gauch: Personal Ontologies for Web Navigation. In *Proc. 9th Intl. Conf. on Information and Knowledge Management (CIKM'00)*, pp. 227-234, 2000.

[8] J. Hendler. Agents and the Semantic Web. *IEEE Intelligent Systems, 16(2)*, pp. 30-37, 2001.

[9] N. Guarina and P. Giaretta. *Ontologies and Knowledge Bases*. In N. Mars (ed.) Towards Very Large Knowledge Base, Amsterdam: IOS Press. pp. 25-32, 1995.

[10] N. Guarino. Formal Ontology in Information Systems. In *Proc. of FOIS'98, IOS Press*, pp. 3-15, 1998.

[11] N. Guarino. *Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology*, chapter Semantic Matching : Formal Ontological Distinctions for Information Organization, Extraction, and Integration. pp. 139-170. Springer Verlag, 1998.

[12] M. Genesereth and N. Nilsson, *Logical foundation of Artificial Intelligence, Morgan Kaufmann*, 1987.

[13] H. Suryanto and P. Compton: Learning Classification Taxonomies from a Classification Knowledge Based System. *Ontology Learning ECAI-2000 Workshop*, 2000.

[14] W. Koh and L. Mui. An Information Theoretic Approach to Ontology-based Interest Matching. In *IJCAI'01 Workshop on Ontology Learning OL-2001*, Seattle, August, 2001.

[15] Y. Kalfoglou, J. Domingue, E. Motta, M. Vargas-Vera, and S. Buckingham Shum, my Planet: An Ontology-driven Web-based Personalised News Service. *In Proc. of the IJCAI'01 workshop on Ontologies and Information Sharing*. 2001.

[16] USA today, http://www.usatoday.com, 2001.

[17] T. Mitchell. 1997. *Machine Learning*. The Mc Graw Hill Companies, Inc.

[18] Yahoo!, http://www.yahoo.com, 2001.

[19] R. Baeza Yates and B. Ribeiro, editors. Modern Information Retrieval. Addison Wesley, 1998.

[20] R. Kosala and H. Blockeel. Web Mining Reseach: A Survey. SIGKDD Explorations, 2(1), pp. 1 15, 2000.

정 현 섭

1993년~2000년 삼척대학교 컴퓨터공학과. 2000년~2002년 한양대학원 컴퓨터공학과. 2002년~현재 포스데이타(주) 솔루션개발 1팀 (DW그룹). 관심분야는 웹 에이전트, 온톨로지, 인공지능, CRM

양 재 영

1998년 한양대학교 전자계산학과 졸업(학사). 2000년 한양대학교 대학원 전자계산학과 졸업(석사). 2000년~현재 한양대학교 대학원 컴퓨터공학과 박사과정. 관심분야는 지능형 에이전트 시스템, 기계학습, 데이타 마이닝, 정보검색, 정보추출

최 중 민

1984년 서울대학교 컴퓨터공학과 졸업(학사). 1986년 서울대학교 대학원 컴퓨터공학과 졸업(석사). 1993년 State University of New York at Buffalo, Computer Science 졸업(박사). 1993년~1995년 한국전자통신연구원 인공지능연구실 선임연구원. 1995년~현재 한양대학교 컴퓨터공학과 부교수. 관심분야는 지능형 에이전트 시스템, 인공지능, 정보검색, 데이타베이스, HCI