

지능형 웹 서비스를 위한 확장된 시맨틱 웹서비스 검색 모델

최 옥 경[†] · 한 상 용^{††} · 이 준 기^{†††}

요 약

웹서비스는 e-비즈니스에서 반드시 필요한 핵심 기술로 인터넷을 이용한 오픈 네트워크를 통하여 단일 또는 다수의 비즈니스 간 기존 어플리케이션 시스템을 표준화된 기술로 결합 시킴으로써 언제, 어디서나, 동일 환경에서 원하는 정보나 서비스를 제공해 주는 총체적 서비스이다. 그러나 현재의 웹서비스 검색 시스템들은 단순한 텍스트 위주의 검색 서비스 방식으로 단어의 유사성 및 상관관계 등을 파악한 신뢰성 있는 결과를 제공하지 못하며 현재 이러한 시맨틱 웹 기능이 포함된 웹서비스 검색 모델은 거의 존재하지 않고 있다. 본 논문은 이러한 기존 웹서비스 검색 모델을 보완하기 위해 일반 웹 문서 검색, UDDI 검색, 시맨틱 문서 검색이 모두 가능한 확장된 시맨틱 웹서비스 검색 모델을 설계 및 구현하였다. 마지막으로 제안 시스템의 실행 결과를 제시하여 그 효율성과 정확성을 검증해 보이고자 한다.

키워드 : 시맨틱 웹서비스, 온톨로지, 매치메이킹, 정보 검색

Extended Semantic Web Services Retrieval Model for the Intelligent Web Services

Okkyung Choi[†] · Sangyong Han^{††} · Zoonky Lee^{†††}

ABSTRACT

Recently Web services have become a key technology which is indispensable for e-business. Due to its ability to provide the desired information or service regardless of time and place, integrating current application systems within a single business or between multiple businesses with standardized technologies are realized using the open network and Internet. However, the current Web Services Retrieval Systems, based on text oriented search are incapable of providing reliable search results by perceiving the similarity or interrelation between the various terms. Currently there are no web services retrieval models containing such semantic web functions. This research work is purported for solving such problems by designing and implementing an extended Semantic Web Services Retrieval Model that is capable of searching for general web documents, UDDI and semantic web documents. Execution result is proposed in this paper and its efficiency and accuracy are verified through it.

Key Words : Semantic Web Services, Ontology, Matchmaking, Information Retrieval

1. 서 론

기존 웹 서비스에 대한 문제점을 보완하기 위해 각 업계에서는 시맨틱 웹 서비스 구성을 위해 DAML-S(Semantic Markup For Web Service), WSFL(Web Service Flow Language), X-LANG과 BPELWS((Business Process Execution Language for Web Services)와 같은 언어들을 연구 및 개발 중이다. DAML-S는 이러한 기존의 웹 서비스 방식을 시맨틱 웹 환경에서 접근하는 방식으로 웹 서비스 자동화를 위해 기존 연구들의 부족한 면을

보완하여 효과적인 웹 서비스의 등록, 검색, 구성, 실행 및 조합이 가능하다. 하지만 현재의 시맨틱 웹 서비스 모델인 DAML-S는 자동화된 웹 서비스를 지원하기 위해서 몇 가지 부족한 점이 있다. 사용자들이 원하는 정보를 표현하는 방법 즉 제약 조건 만족 및 사용자 요구사항에 대한 반응이 미흡하고, 품질 요소(가용성, 준수성, 신뢰성, 성능, 가격)에 처리가 미흡하다.

따라서 본 연구에서는 이러한 기존 시맨틱 웹 서비스 시스템에 대한 문제점을 분석하고 이에 대한 해결책을 제시하고자 한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 기존 시맨틱 웹서비스 시스템에 대한 비교 및 분석을 수행하고, 3장에서 매치메이킹 엔진에 대해 살펴보고, 4장에서 확장된 시맨틱 웹서비스 검색 모델을 제안하고, 마지막으로 5장에서 결론 및 향후 연구 과제를 서술하였다.

* 이 논문은 2006년도 중앙대학교 학술연구비 지원에 의한 것임.

† 준 회원 : 연세대학교 정보대학원 BK 박사후 연구원

†† 종신회원 : 중앙대학교 컴퓨터공학과 교수

††† 정 회원 : 연세대학교 정보대학원 교수

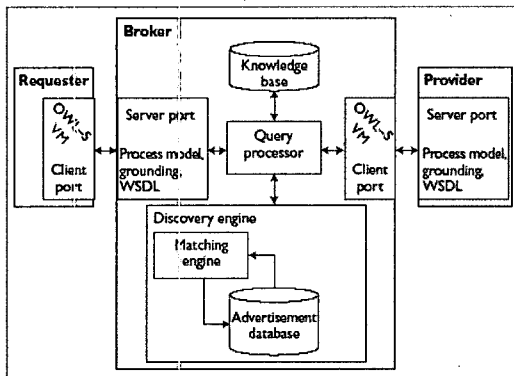
논문접수 : 2005년 8월 16일, 심사완료 : 2006년 8월 1일

2. 기존 시스템 비교 및 분석

InfoSleuth[5]는 에이전트 기반의 정보 검색 시스템으로 구문과 의미적 매치메이킹을 수행하기 위해서 “브로커 에이전트”를 이용한다. 브로커 에이전트는 서비스를 제공하는 에이전트 간의 연결을 지원하며 저장소를 통해 최근의 정보를 보유하고, 브로커는 적당한 서비스를 제공하기 위해 모든 가능한 에이전트들을 위치시키는 질의 에이전트를 제공한다. 구체적인 방식은 구문적 브로커링과 의미적 브로커링 방법을 이용한다. InfoSleuth는 멀티 브로커 에이전트가 서비스 요청자와 서비스 제공자 사이의 중간자 역할을 수행하고 Peer-to-Peer 시스템 방식으로 브로커들간에 상호 협력을 이용한 효율적인 검색 결과를 제공하는 장점이 있다. 그러나 LDL++[13], 즉 논리적 연역법 언어의 규칙을 이용하며 이는 표준화된 서비스 기술 언어가 아닐 뿐만 아니라 시맨틱 웹 기술을 기반으로 한 설계 방식이 아닌 단점이 있다. 또한 브로커 에이전트가 서비스 요청자와 서비스 제공자 사이의 중계자 역할을 수행하기에 데이터 전송시 병목 현상이 발생할 수 있다.

[6]은 OWL-S 기반의 브로커를 이용한 시맨틱 웹 서비스 시스템으로 웹 서비스의 효율적인 수행과 동기화를 위해 에이전트 기반의 브로커를 이용한다. 본 연구에서는 브로커의 기능을 명확하게 하기 위해 확장된 OWL-S를 제안한다. (그림 1)은 브로커 아키텍처로 Query processor와 Discovery engine을 이용해 서비스 요청자에게 효율적이고 정확한 서비스 광고를 제공한다. 그러나 이 방식도 위의 InfoSleuth처럼 브로커 에이전트가 중계자 역할을 모두 수행하여야 하므로 데이터 전송 시 병목 현상이 발생할 수 있는 단점이 있다.

시맨틱 웹 서비스 기술 문서를 기반으로 DAML-S로 작성된 [7]은 웹 서비스 검색의 일치 기준을 서비스 요청자와 서비스 제공자 사이의 요청과 광고 일치 여부에 따른다. 즉 요청과 광고 간의 일치 여부는 각각의 서비스 입출력 항목을 상호 비교하여 판단하고, 일치 등급을 Exact, Plug-In, Subsume, Fail의 4단계로 구분 하였다. 매칭 알고리즘을 이용한 서비스 등급 부여 방식으로 효율적인 웹 서비스 검색이 가능하나 동일한 등급에 대해서는 세부 순위가 주어지지 않아 정확한 서비스 검색 결과를 유도하기 어려운 단점이 있다.



(그림 1) 브로커 아키텍처

Carnegie Mellon 대학의 Robotics Institute에서 진행하고 있는 동적인 매치메이킹 방식을 중심으로 한 Larks[8]는 서비스 광고와 요청을 위해서 LARKS 언어를 이용하여 유연하고 효율적인 매칭 과정이 가능하도록 하였다. 매칭 프로세스는 다섯가지의 다른 매칭 필터(문맥 매칭, 프로파일 비교, 유사도 매칭, 기호 매칭과 제약 조건 매칭)를 사용하여 서비스 제공자, 서비스 요청자, 매치메이커 에이전트 간의 상호 작용으로 이루어진다. 이 방식 역시 다섯가지의 다른 매칭 필터를 사용하여 구문과 의미 매칭을 모두 지원하지만 실제적으로는 의미적인 매칭의 접근이 어려운 단점이 있다.

3. 매치메이킹 엔진

3장에서 매치메이킹의 정의, 요구사항을 살펴보고 매치메이킹 알고리즘을 통한 매치메이킹 엔진 아키텍처를 기술한다.

3.1 매치메이킹 정의 및 요구사항

매치메이킹은 서비스 요청자의 요구사항을 만족하는 서비스 제공자를 연결하기 위한 과정으로, 웹서비스 요청과 웹서비스 광고 간의 일치 여부에 의해 결정된다. 이러한 요청과 광고 간의 일치 여부는 기능적 측면의 기술 내용 중 서비스 입출력의 일치 여부에 의존한다. 또한 매치메이킹 시스템은 저장소를 통해 입출력을 지원하고 요청된 서비스의 검색, 수정, 취소가 가능하도록 하여야 한다.

3.2 매치메이킹 알고리즘

효율적인 시맨틱 웹서비스 검색을 위해서는 서비스 요청과 서비스 광고간의 정확한 일치가 이루어져야 한다. 이러한 요청과 광고 간의 일치 여부는 서비스 기능적 측면(Functional Description)의 입출력(Input & Output)의 일치에 따른다. 즉 서비스 요청의 입력과 서비스 광고의 입력을 비교하여 서로간에 요소들을 채울 수 있으면 둘 간에 입력이 일치된다고 본다. 반대로 서비스 요청의 출력과 서비스 광고의 출력을 비교하여 서로간의 요소들을 만족시킬 수 있으면 둘 간에 출력이 일치된다고 본다. 결과적으로 입출력 모두가 일치되면 해당 서비스는 서비스 요청을 올바르게 수행하여 만족스러운 결과를 제공해 주게 된다.

<p>[규칙 1] Exact If advertisement A and request R are equivalent concepts, we call the match Exact. ($R = A$)</p> <p>[규칙 2] PlugIn If request R is sub-concept of advertisement A, we call the match PlugIn. ($R \supset A$)</p> <p>[규칙 3] Subsume Request가 Advertisement의 SubClass일 경우 이를 Subsume이라고 명한다. ($R \subset A$)</p> <p>[규칙 4] Intersection Request와 Advertisement간의 교차점이 발생하면 이를 Intersection이라고 명한다. ($R \cap A$)</p> <p>[규칙 5] Fail Request와 Advertisement간의 일치 부분이 없으면 이를 Fail이라 명한다. ($R \neq A$)</p>
--

본 연구에선 이러한 입출력의 일치 여부를 판단하기 위해 Exact, PlugIn, Subsume, Intersection, Fail의 5등급으로 나누어 표시하였으며 [규칙 1]에서 [규칙 5]로 갈수록 낮은 순위가 부여된다.

여기서 [규칙 2]와 [규칙 3]이 혼동될 수 있는데 예를 들어, “남자” 타입의 출력을 내주는 서비스를 원하는 경우, “남자”가 “사람” 클래스의 하위 클래스라 하여 “사람” 타입의 출력을 결과로 찾아 준다면 이 서비스는 적합하지 않을 것이다. 따라서 [7, 11]의 일치 방식과는 다르게 본 연구에서는 [규칙 2]와 [규칙 3]방식을 기준으로 “남자”를 입력으로 받는 서비스를 원할 경우 요청의 입력 값이 광고의 입력 값보다 크거나 동일할 경우에 우선 등급을 부여하였다.

본 연구에서는 매치메이킹 방법의 효율성과 정확성을 위해 일치 등급 부여 방식을 1, 2단계로 나누었다. 2단계에서는 규칙2, 규칙3에 대해 기존 벡터 모델 방식을 수정한 새로운 순위 적용 알고리즘[12]을 적용시켜 순위를 보다 세분화 하였다.

제안하는 일치 등급 부여 알고리즘은 다음과 같다.

```

<1 단계>
First_Match_Compare(output.R, output.A)
{
    if output.R is equivalent to output.A then Level = Exact;
    return Exact;
    else if output.R is SuperClassOf output.A then Level = PlugIn;
    return PlugIn;
    else if output.R is SubClassOf output.A then Level = Subsume;
    return Subsume;
    else if output.R is not incompatable with output.A then Level = InterSection;
    return InterSection;
    else Level= Fail;
    return Fail;
}
R:request, A:Advertisement

2단계
Secound_Match_Compare(output.R, output.A)
{
    Switch(Level) {
        case Exact:
            Level_rank = 0;
            Break;
        case PlugIn:
        case Subsume:
        case InterSection:
            Call Function Ranking_Compare();
            Break;
        case Fail:
            Level_rank = 1;
            Break;
    }
}
    
```

여기서 2단계의 일치 등급 부여 방식은 일치 등급이 Exact, PlugIn, Subsume, InterSection 일 경우에 적용된다. 먼저 일치 등급이 Exact 일 경우엔 서비스 요청과 서비스 광고가 정확하게 일치하는 경우로 최상위 등급을 부여 한다. 그 다음에 서비스 요청이 서비스 광고를 포함할 경우 기존 연구 Semantic Management Module[12]의 Ranking_Compare() 함수를 호출한다. Ranking_Compare() 함수는 같은 등급의 서비스들간의 순위

부여를 위해 반드시 필요한 함수로 각 용어간의 관계성 즉 상속 계층간의 수직, 수평 근접도와 용어들간의 동의어 관계를 이용하여 순위를 적용하는 방식이다.

즉, Ranking_Compare() 함수는 용어들 간의 동의어 관계와 용어들 간의 관계성을 측정하여 매치메이킹의 랭킹을 보다 세분화 한다. 먼저 용어들 간의 동의어 관계 측정을 위해 각 용어가 가지는 유사도 범위를 이용하여 측정하는 데 범위는 1~9 사이의 값을 가지며 유사성이 높을수록 1에 가깝고 유사성이 떨어질수록 9에 가까워진다. 각 범위의 비교 대상 요소는 검색어, 추출된 문서에 포함된 용어, 온톨로지를 통해 추출된 검색어의 동의어, 온톨로지를 통해 추출된 문서에 포함된 용어의 동의어다. 또한 용어들 간의 관계성을 측정할 변수는 각 문서가 가지는 구조들의 각 수평 노드간의 근접도와 각 수직 노드간의 근접도를 이용하여 가중치를 결정한다. 따라서 1단계의 단순 일치 등급 부여 방식을 보완하기 위해 보다 세분화된 2단계 일치 등급 부여 방식을 적용시킴으로써 결과에 대한 확실한 우선 순위를 제공할 수 있어 보다 정확하고 효율적인 검색 결과를 가져오게 된다[12].

3.3 매치메이킹 알고리즘 적용 예

3.3.1 시맨틱 웹서비스 시나리오

웹서비스를 이용하여 핸드폰(Cellular Phone) 구매를 희망하는 한 고객이 있다. 다음은 웹서비스 요청 후 처리결과에 대한 사항이다. 구매자는 카메라 폰 기능이 있는 핸드폰 구입을 희망한다. 그러나 가격대가 20만원 이상이면 가격이 비교적 저렴한 대기업 OEM 제품으로 대체해서 구매하기를 원한다. 이러한 조건들을 만족하는 제품을 사기 위해 구매자는 웹서비스를 이용하였다. 여기서 확장 시맨틱 웹서비스 시스템의 웹 에이전트는 그의 개인 정보 프로파일을 통해 해당 구매자가 대기업 OEM 제품으로 H사 제품을 선호하고 제품을 구입할 때 가격대비 다양한 기능을 보유한 제품을 선호하는 경향이 있다는 정보를 얻게 된다. 따라서 해당 구매자에게 H사의 카메라폰 기능이 있는 20만원 미만의 가격대 제품을 추천하게 되고 만약 구매를 희망한다면 구매 처리까지 실행한다.

3.3.2 제품 구입을 위한 온톨로지 정의

본 절에서는 3.4.1의 시나리오를 바탕으로 온톨로지를 정의한 예이다. 온톨로지의 정의를 위해서 DAML S 0.9 버전의 서비스 프로파일 클래스를 이용하여 광고, 요청 서비스를 정의 하였다. 또한 구문 규칙을 표현하기 위해서 DAML+OIL 구문의 DL notions 을 사용하였으며 표현된 방식은 다음과 같다.

$$\begin{aligned}
 &ServiceProfile \subseteq \perp \\
 &Advertisement \subseteq ServiceProfile \\
 &Query \subseteq ServiceProfile
 \end{aligned}$$

또한 Sales 와 Provider, Product 온톨로지를 추가로 정의한다.

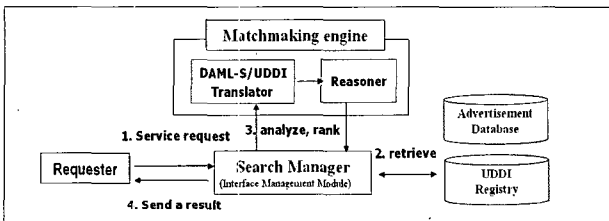
$$\begin{aligned}
 &Sales \subseteq (= | providedBy.Provider) \cap \\
 &(= | hasFeatureSelection.FeatureTyper) \cap
 \end{aligned}$$

$(=| hasQuantity.Integer) \cap$
 $(=| hasProductPrice.Integer)$
 $Provider \subseteq (=| hasName.ProviderName) \cap$
 $(=| hasCompanyName.CompanyName)$
 $Product \equiv hasBuilt-inCamera.Item$
 $Item \equiv Cellular-Phone \cup PDA$

3.4 매치메이킹 엔진 구조

(그림 2)는 매치메이킹 엔진을 이용하여 서비스 요청자가 요구하는 서비스 광고를 찾아주는 과정을 시스템 흐름도로 나타낸 것이다. 단계별 절차를 살펴보면 다음과 같다.

- 1단계 : 서비스 요청자는 Search Manger에게 서비스 검색을 요청한다.
- 2단계 : Search Manger는 광고 DB(Advertisement Database)와 UDDI 레지스트리(UDDI registry)를 통해 검색되는 서비스 결과를 추출한다.
- 3단계 : Search Manger는 DAML-S/UDDI 번역기(Translator)를 통해 UDDI 서비스 표현을 분석한다. 그 다음엔 추론기(Reasoner)를 통해 매칭의 등급을 부여한다.
- 4단계 : 그 결과를 서비스 요청자에게 반환한다. 모든 요청 결과들은 우선 순위대로 결과가 나타난다.



(그림 2) 매치메이킹 엔진 구조

4. 확장된 시맨틱 웹서비스 개선 모델

4.1 시스템 모델

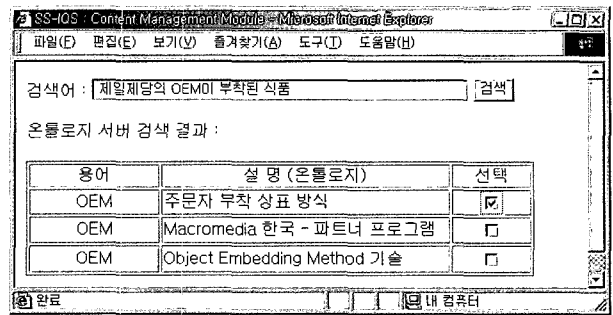
시스템 모델에 대한 전체 흐름도 및 방법론에 대한 설명으로 사용자가 검색 질의를 요구하고 이에 대한 결과를 사용자에게 제공해주기까지의 과정을 나타낸 것이다. 제안하는 시스템 모델은 10단계의 절차를 통해 웹서비스 검색을 수행하며 각각의 절차를 살펴 보면 다음과 같다.

4.1.1 사용자 요구 질의

사용자가 사용자 인터페이스 화면을 통하여 검색 질의를 한다.

4.1.2. 유사어 집합 검색 요구

반환된 검색 결과를 바탕으로 온톨로지 서버에 유사어 집합 검색을 요구한다. E-engine Ontology Server는 사용자가 올바르게 온톨로지를 선택할 수 있도록 온톨로지 용어와 설명을 함께



(그림 3) 온톨로지 정보 검색 결과

제공해 준다. 예를 들어 “제일제당의 OEM이 부착된 식품” 정보는 “주문자 부착 상표 방식”, “회사끼리의 파트너 프로그램인 Macromedia 한국-파트너 프로그램”, 또는 “Object Embedding Method 기술”의 OEM 중 어떤 것을 선택할 지를 사용자에게 물어 보게 된다. (그림 3)은 Interface Management Module의 온톨로지 정보 확인을 위한 인터페이스이다.

4.1.3 유사어 생성

생성된 유사어 집합 결과를 바탕으로 원하는 용어를 선택하기 위한 재질의를 수행한다. 예를 들어 음식, 식품, 요리 등은 식품과 관련이 있고, 주문자 부착 상표 방식, original equipment manufacturer 등은 OEM과 관련이 있다. 따라서 사용자의 재질의를 통해 원하는 결과를 도출한다.

4.1.4. 5 UDDI 검색 및 검색 결과 반환

사용자의 검색 질의를 Advertisement Database와 UDDI Registry를 이용해 검색을 수행하고 매치메이킹 엔진을 이용해 분석 후 그 결과를 반환한다.

4.1.6 규칙 기반 검색 요구

생성된 결과를 바탕으로 규칙 기반 검색을 요구한다.

4.1.7 규칙 기반 검색 수행

4.1.4에서 생성된 결과와 사용자 프로파일 레지스트리에서 가져온 사용자의 규칙을 비교하여 이와 적합한 서비스 정보만을 선정한다. 본 연구에서는 규칙 기반 검색을 수행하기 위해서 SweetJess[4]를 이용한다. SweetJess는 규칙을 DAML+OIL로 표현하고 JESS의 규칙 언어로 변환 후 추론을 시행한다. SweetJess를 통해 사용자 프로파일 레지스트리에 있는 기술 규칙들과 반환된 각 웹서비스 결과들을 XSLT를 통해 Jess 스크립트로 변환 시킨다.

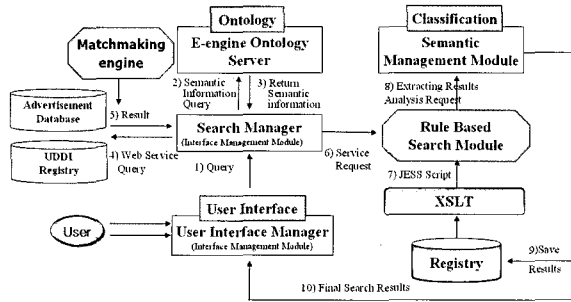
여기서 Jess 엔진[3]은 RuleML로 표현한 규칙들을 XSLT를 이용해서 기존의 자동 추론기가 사용할 수 있는 포맷으로 변환 후 전방향 추론을 통해서 결과를 도출한다.

4.1.8 추출된 결과 분석 요청

추출된 결과의 분석을 위해 기존 연구[12]의 분류 처리 모듈인 Semantic Management Module에 요청하여 유사도 측정 및 순위화를 수행한다.

4.1.9.10 추출 결과 반환 및 저장

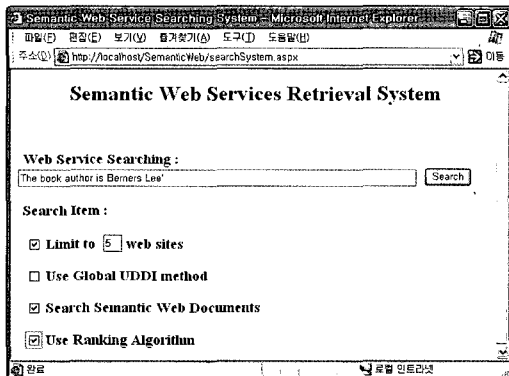
규칙 기반 검색을 이용해 사용자가 원하는 서비스 정보를 얻었다 하더라도 현재 웹서비스가 유효하게 존재하고 사용자가 바로 사용할 수 있는지는 알 수 없다. 따라서 유효하지 않은 서비스는 추출하고 기존 연구[12]의 유사도 측정 모델을 통해 사용자 검색 질의에 대한 순위화를 수행한 후 최종 결과를 반환하고 이에 대한 정보를 레지스트리에 저장한다.



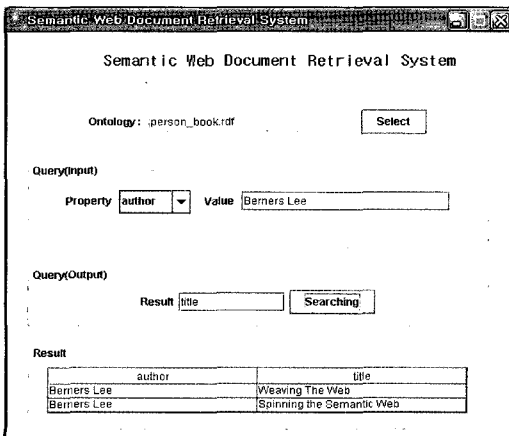
(그림 4) 시스템 구조

4.2 실행 결과

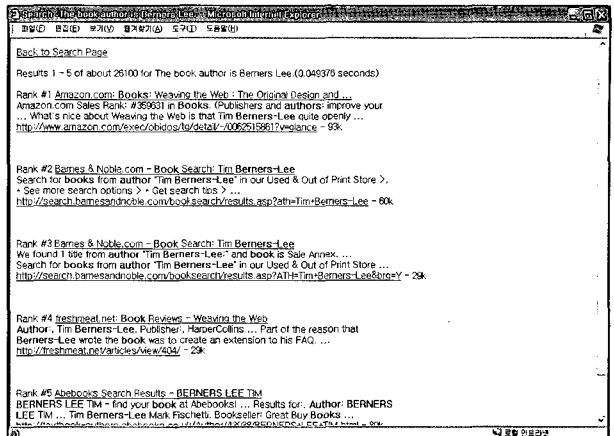
본 단원에서는 개선된 시맨틱 웹서비스 검색 모델의 구현 결과이다. 모듈 별 실행 화면 및 결과를 살펴 보면 다음과 같다.



(그림 5) 사용자 인터페이스 화면(초기 화면)



(그림 6) 시맨틱 웹 문서 검색 결과 화면



(그림 7) 순위화 기법을 적용 시킨 검색 결과 화면

(그림 5)는 시맨틱 웹서비스 검색을 위한 사용자 인터페이스 화면이다. 검색 방법은 일반 웹 검색 엔진, UDDI 방식, 시맨틱 웹 문서 검색 모두가 가능하다. 검색어에 'The book author is Berners Lee'를 입력한 후 검색 항목에서 5개의 검색 결과를 보여주는 것과 시맨틱 웹 문서 검색과 순위화 알고리즘 사용을 체크하고 검색을 실행하게 되면 다음 (그림 6)과 같은 결과를 보여 주게 된다.

(그림 6, 7)은 사용자가 검색어를 입력하고 질의를 보낸 후 시맨틱 웹 문서 검색 시스템과 일반 웹 검색 엔진으로부터 반환 받은 결과 화면으로 순위화 알고리즘을 적용시킴으로써 기존 시스템과의 차별화된 화면을 제공한다. 일반 웹 검색 방식은 구글 웹서비스의 WSDL 파일을 참조한 구글 API를 사용하였다. 즉 구글 API를 이용하여 질의에 대한 검색을 수행하고 검색된 결과를 result.html 파일에 저장한 후 웹 브라우저를 통해 사용자에게 제공하였다.

5. 결론 및 향후 연구 과제

최근 웹서비스의 등장은 기업이 자신들의 비즈니스 로직을 효율적으로 공개할 수 있는 기반을 제공하게 되어 협업과 통합으로 대변되는 기업간 통합을 불러일으키는 원동력이 되고 있다. SOAP, WSDL, UDDI라는 XML 기반의 3가지 공개 표준을 근간으로 하는 웹서비스는 XML을 이용하여 구조화한 메시지를 웹을 통하여 전송함으로써 이기종 시스템간의 상호 통합이 가능하다. 그러나 기존의 웹서비스 시스템은 사용자 요구 사항의 미반영, 신뢰성과 같은 품질 측정 미비, 검색결과에 대한 순위화 미지원 등의 문제점을 가지고 있다.

본 논문은 이러한 문제점을 해결하기 위한 방안으로 시맨틱 웹 기술과 웹서비스 기술을 융합한 확장된 시맨틱 웹서비스 검색 모델을 제안하였다. 제안된 모델은 매치메이킹 서비스를 통해 서비스 요청자와 서비스 제공자 간의 효율적인 매칭이 가능하도록 하였다. 향후 연구 과제로는 제안한 모델에서 사용자 제약조건에 대한 반영이 이루어진 시스템 개발이 요구되며 이를 바탕으로 보다 객관적인 성능 평가가 수행되어야 한다.

참 고 문 헌

[1] Cardoso, J. and A. Sheth. "Semantic e-Workflow Composition, Journal of Intelligent Information Systems (JIIS), 21(3), pp.191-225, (2003)

[2] C. Zaniolo. "The logical data language (ldl): An integrated approach to logic and databases", MCC Technical Report STP-LD-328-91, 1991.

[3] Jess, the Rule Engine for the Java Platform <http://www.jessrules.com/>, 2005.

[4] Mahesh Gandhe, Tim Finin, and Benjamin Grosf, "SweetJess: Translating DamlRuleML to Jess", International Workshop on Rule Markup Languages for Business Rules on the Semantic Web in conjunction with ISWC2002, Jun., 2002.

[5] M. Nodine, W. Bohrer, and A. Ngu, "Semantic multibrokering over dynamic heterogeneous data sources in InfoSleuth.", Proceedings of the Fifteenth International Conference on Data Engineering. Sydney, IEEE Computer Society Press, pp.358-365, 1999.

[6] M. Paolucci, J. Soudry, N. Srinivasan, and K. Sycara, "A Broker for OWL-S Web Services", in Cavedon, Maamar, Martin, Benatallah, (eds) Extending Web Services Technologies: the use of Multi-Agent Approaches, Kluwer, 2005.

[7] M. Paolucci, T. Kawamura, T. Payne, and K. Sycara. "Semantic matching of web services capabilities." In Proc. of the 1st International Semantic Web Conference (ISWC), 2002.

[8] Sycara, K., Widoff, S., Klusch, M., Lu, J.: LARKS: Dynamic Matchmaking Among Heterogeneous Software Agents in cyberspace, Autonomous Agents and Multi-Agent Systems, Vol.05, pp.173-203, (2002).

[9] 이경하, 이규철, "웹 서비스의 향후 발전 방향", 정보처리학회 제9권 4호, 2002. 7.

[10] 이경하, 이규철, "웹 서비스의 표준화 동향과 발전 방향", 데이터 베이스연구회 학회지 p.80-87, 2003. 3.

[11] Lei Li and Ian Horrocks, "A Software Framework For Matchmaking Based on Semantic Web Technology", In Proceedings International WWW Conference, Budapest, Hungary. 2003.

[12] 최옥경, 한상용, "자동화된 통합 프레임워크를 위한 시맨틱 웹 기반의 정보 검색 시스템", 한국정보처리학회논문지 제 13-C권 제1호, pp.129-136, 2006. 2. 28.

[13] C. Zaniolo. "The logical data language (ldl): An integrated approach to logic and databases", MCC Technical Report STP-LD-328-91, 1991.



최 옥 경

e-mail : okchoi20@gmail.com
 1996년 단국대학교 이과대학(이학사)
 1996-8년 삼성전자(주) 정보 통신 본부 근무
 1999년 CJ드림소프트(주) F/E팀 근무
 1999년 중앙대학교 대학원 컴퓨터소프트웨어학과(공학석사)

2006년 중앙대학교 대학원 컴퓨터공학과(공학박사)
 현재 연세대학교 정보대학원 BK 박사후 연구원
 관심분야 : Semantic Web Services System, e-Commerce, Auction System, Information Retrieval



한 상 용

e-mail : hansy@cau.ac.kr
 1975년 서울대학교 공과대학(공학사)
 1984년 Minnesota 공과대학(공학박사)
 1984년-1995년 IBM 책임연구원
 1995년~현재 중앙대학교 컴퓨터공학과 교수

관심분야 : Web Technologies(Web Services, Semantic Web, XML), Information Retrieval, Multimedia



이 준 기

e-mail : zlee@yonsei.ac.kr
 1985년 Computer Sciences and Statistics, Seoul National University(학사)
 1999년 Marshall School of Business, University of Southern California(박사)

현재 연세대학교 정보대학원 교수
 관심분야 : E-business Strategy, Information Technologies and Knowledge Management