

# 매치메이킹 알고리즘을 이용한 개선된 웹서비스 검색 시스템

최옥경  
연세대학교 정보대학원  
(okchoi20@gmail.com)

이정우  
연세대학교 정보대학원  
(jlee@yonsei.ac.kr)

한상웅  
중앙대학교 컴퓨터공학부  
(hansy@cau.ac.kr)

.....

향후 우리가 지향하는 차세대 웹은 보다 정확하고 신뢰성 있는 정보를 제공할 수 있는 의미론 중심의 시맨틱 웹과 XML이라는 표준 인터페이스를 통해 컴퓨터간의 자동화된 정보 교환 방식으로 향상된 고품질의 서비스를 제공하는 웹서비스의 기술이 결합된 시맨틱 웹서비스일 것이다. 이러한 시맨틱 웹서비스에서는 개념적으로 자동화된 웹서비스의 발견, 실행 및 조합이 가능하다. 그러나 기존에 개발된 시맨틱 웹서비스 검색 시스템들에서는 매치메이킹과 브로커링을 통해서 자동화된 웹서비스를 지원하는 에이전트들을 개발하고 있는 데 QoS(Quality of Services)에 관한 고려가 안 되어 있어서 실제 상황에서 적용에 어려움이 있다.

이에 본 연구에서는 이러한 기존 웹서비스 검색 모델을 보완하기 위해 QoS에 근거한 매치메이킹 알고리즘을 제안하고 이를 활용하여 일반 웹 문서 검색과 시맨틱 문서 검색이 모두 가능한 개선된 웹서비스 검색 시스템을 개발하였다. 여기서 제안된 시스템에서는 매치메이킹 알고리즘을 통해 서비스 요청자와 서비스 제공자 사이의 효율적인 매칭이 가능하고 사용자의 요구 사항 중 서비스 품질 정보를 고려하여 보다 정확한 매치메이킹을 가능케 한다. 구현된 시스템의 성능 평가를 수행하였고 그 결과도 여기에 보고하였다.

.....

논문접수일 : 2006년 10월

게재확정일 : 2007년 09월

교신저자 : 이정우

## 1. 서론

웹서비스는 XML(Extensible Markup Language)을 이용하여 구조화한 데이터를 HTTP(Hypertext transfer Protocol) 등의 기존 웹 기술을 이용하여 전송함으로써 상호 이질적인 플랫폼 사이의 상호 작용을 돕는 소프트웨어 시스템[이경하, 2002]이다.

그러나 기존 웹서비스 방식에서는 기존의 “웹 콘텐츠”와 새로운 “웹서비스 정보”가 분리되어 입력, 발견, 접근, 프로세싱 되어서 사용자가 실제로 원하는 의미를 포함한 시맨틱(의미론적) 검색이

힘든 단점이 있다[Haibo, 2005].

따라서 웹서비스의 자동화를 위해 기존 연구들의 부족한 면을 보완한 시맨틱 웹서비스가 등장하게 되었고 시맨틱 웹서비스 기술은 시맨틱 웹 기술을 웹서비스에 적용시킴으로써 웹서비스가 갖고 있는 기술적 한계들을 해결하고자 하는 시도이다. 이러한 자동화된 웹서비스의 발견, 실행 및 조합을 위하여 브로커링과 매치메이킹을 중심으로 시맨틱 웹서비스 시스템에 대한 연구들이 진행 중이다. 브로커링 방식의 [Nodine et al., 1999], [Sycara et al., 2004]는 서비스 요청자와 제공자 사이에서

브로커를 이용하여 웹서비스의 효율적인 발견과 중계를 수행하는 형태로서 이러한 브로커를 이용한 방식은 브로커 중심의 중앙 집중형 구조로 데이터 전송 시 병목 현상이 발생할 수 있는 단점이 있다. 한편 매치메이킹 방식의 [Paolucci et al., 2002], [Sycara et al., 2002]는 서비스 제공자와 서비스 요청자의 요구 사항을 매치메이커가 매칭을 시켜주는 방법이나 단순히 서비스 요청과 광고의 일치 여부에 대한 매칭 기법으로 서비스들 사이의 관계를 추론하는 의미론적인 매치메이킹이 불가능하고 사용자의 요구 사항을 반영한 QoS(Quality Of Services)가 제공되지 않는 단점이 있다.

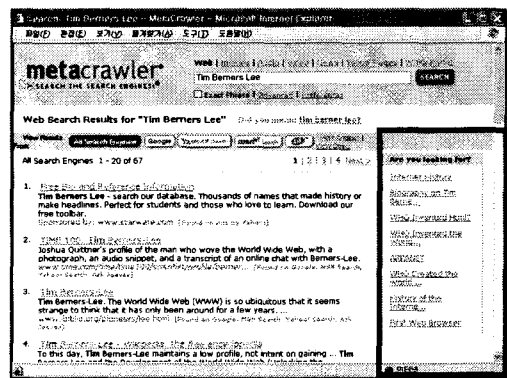
이에 본 연구에서는 이러한 기존 시맨틱 웹서비스 시스템에 대한 문제점을 분석하고 이에 대한 해결책을 위해 매치메이킹 알고리즘을 제안하고 이를 근간으로 일반 웹 문서 검색과 시맨틱 문서 검색이 모두 가능한 개선된 웹서비스 검색 시스템을 설계 및 구현하였다. 제안 시스템은 QoS 기반의 매치메이킹 알고리즘을 도입하여 서비스 제공자와 서비스 요청자 사이의 매칭을 높이고 양질의 서비스를 선택할 수 있도록 하였다.

본 논문의 구성은 다음과 같다. 먼저 제 2장에서는 기존 시맨틱 웹서비스 시스템들에 대한 비교 및 분석을 수행하고, 제 3장에서는 매치메이킹 알고리즘을 제안하고, 제 4장에서 시스템 구현 과정과 평가 결과를 서술하고 마지막으로 결론 및 향후 연구 과제를 제 5장에서 서술하였다.

## 2. 기반 연구

현재의 웹 환경에서 진행되고 있는 연구 방향을 살펴보면 자체적으로 검색 시스템을 운영하지 않고 Google[Google, 2006], Yahoo[Yahoo, 2006],

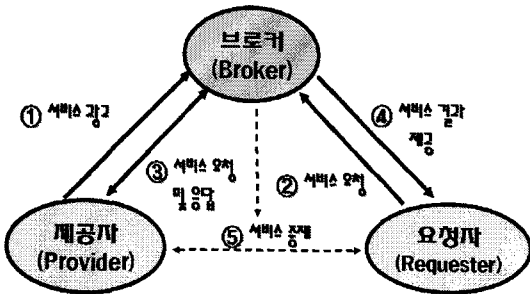
LookSmart[LookSmart, 2006]등과 같은 검색 엔진에 검색을 의뢰한 후 반환 받은 결과들을 웹상에 보여 주는 방식을 채택하고 있는 검색 기법인 Metacrawler[Metacrawler, 2006]가 있는데 이 방식은 [그림 1]에서 보는 바와 같이 "Tim Berners Lee"라는 검색어를 입력하였을 때 해당 결과 사이트를 좌측에 보여주고 우측에는 에이전트 기법을 이용하여 사용자가 관심 있어 하는 항목을 찾을 수 있도록 리스트를 보여준다. Metacrawler는 에이전트를 이용하여 관심 있어 할 만한 항목들을 리스트로 제시하는 방식으로 의미론적 검색이 가능한 것 같아 보이지만 실제로 제시해 주는 관심사 리스트는 사용자 의도와는 전혀 무관한 정보를 제공해 주는 경우가 많으며 실제 웹서비스 검색이 가능하다고는 하지만 단순한 정보 검색만 가능할 뿐 실제 웹서비스 제공자의 정보는 제공되지 않는 단점이 있다.



[그림 1] 검색 엔진 Metacrawler

검색 엔진 Metacrawler는 매치메이킹이나 브로커링 방식이 아닌 일반 웹서비스 검색 시스템 방식으로 실질적인 웹서비스 검색이 불가능하고 의미론적인 웹 검색도 불가능하다. 따라서 이에 대한 개선책으로 추후 연구들이 매치메이킹과 브로

커링 방식을 웹서비스에 도입 시킨 것입니다.



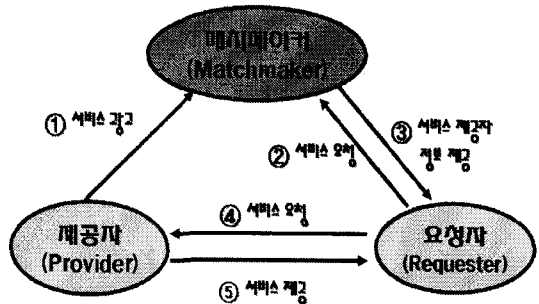
[그림 2] 브로커링

브로커링 방식 [그림 2]은 브로커가 모든 서비스를 주관하는 중앙 집중형 구조를 취하는데 각 단계별 프로세스를 살펴보면 다음과 같다.

- ① 서비스 제공자는 제공하고자 하는 서비스 기술서를 브로커에 등록한다.
- ② 서비스 요청자는 브로커를 통하여 검색하고자 하는 웹서비스를 요청한다.
- ③ 브로커는 적합한 서비스를 찾기 위해 관련 정보를 서비스 제공자에게 요청한다
- ④ 적합한 서비스를 찾은 브로커는 이러한 결과를 서비스 요청자에게 응답해 준다.
- ⑤ 브로커는 서비스 제공자와 서비스 요청자 사이에서 중계 역할을 수행한다.

위와 같이 브로커링 방식은 적합한 서비스 제공자를 찾기 위해 브로커가 서비스 발견에서 중계까지 모든 업무를 담당하므로 보다 정확한 서비스 매칭이 가능하지만 중앙 집중형 방식으로 데이터 전송 시 브로커에서 병목 현상이 발생할 수 있는 단점이 있다.

매치메이킹 방식 [그림 3]은 매치메이커가 서비스 요청자와 서비스 제공자 사이에서 효율적인 매칭을 담당한다. 브로커링 방식과 마찬가지로 서비



[그림 3] 매치메이킹

스 제공자는 처음에 매치메이커에게 정보를 등록해야만 한다. 서비스 제공자 에이전트는 적합한 메시지를 보냄으로써 그들의 정보를 광고하게 된다. 서비스 요청자는 서비스를 매치메이커에게 요청하고 매치메이커는 적합한 서비스 제공자를 찾은 후 그 서비스 제공자에 대한 정보를 서비스 요청자에게 제공해 주게 된다. 브로커링 방식과는 달리 매치메이커는 서비스 제공자와의 모든 서비스 수행 업무를 담당하지는 않으며 서비스 요청자에게 적합한 서비스 제공자에 대한 정보만을 제공해 주고 그 이후의 서비스 수행 업무는 서비스 제공자와 요청자 사이에 이루어지도록 한다. 즉 매치메이커는 서비스 발견의 업무만 수행하고 중계의 역할은 하지 않는다.

브로커링 방식을 이용한 웹서비스 시스템은 크게 구문과 의미 기반의 브로커 에이전트를 이용한 InfoSleuth[Nodine et al., 1999], OWL-S 기반의 브로커를 이용한 시맨틱 웹서비스 시스템인 [Sycara et al., 2004]이 있다. 매치메이킹 방식을 이용한 웹서비스 시스템은 5가지의 서로 다른 필터를 이용한 매치메이킹 기법인 LARKS[Sycara et al., 2002], DAML-S로 작성된 시맨틱 매칭을 중심으로 한 [Paolucci et al., 2002]등이 있다. 각각의 장단점을 분석하면 <표 1>과 같다.

&lt;표 1&gt; 기존 시스템 장단점 분석

방식	시스템	장 점	단 점
브로커링	InfoSleuth [Nodine et al, 1999]	· 멀티 브로커 에이전트가 중간자 역할 수행 · Peer-to-peer 시스템 방식으로 브로커들간에 상호 협력으로 결과 제공	· 표준화된 서비스 기술 언어를 사용하지 않음
	[Sycara et al, 2004]	· OWL-S 기반의 브로커를 이용한 시맨틱 웹서비스 시스템 · 브로커의 기능을 명확하게 하기 위해 새로운 OWL-S 제안	· 데이터 전송시 병목 현상 발생 우려
매치메이킹	[Paolucci et al, 2002]	· DAML-S로 작성된 시맨틱 매칭 · 매칭 알고리즘을 이용한 서비스 등급 부여 방식	· 동일한 등급에 대해서는 세부 순위가 주어지지 않아 정확한 서비스 검색 결과를 유도하기 어려움
	LARKS [Sycara et al, 2002]	· 서비스 광고와 요청을 위해 LARKS 언어를 이용한 유연하고 효율적인 매칭 제공 · 다섯 가지의 다른 매칭 필터를 사용하여 구문과 의문 매칭 모두 지원	· 실제로 의미적인 매칭 접근 방법이 어려움

### 3. 매치메이킹 알고리즘

제 3장에서는 시맨틱 웹서비스 시스템의 설계를 위해서 필요한 핵심 기술인 온톨로지, 매치메이킹 알고리즘에 대해 기술한다.

#### 3.1 매치메이킹을 위한 온톨로지 정의

본 절에서는 웹서비스 요청과 광고를 위한 DAML-S 기반의 온톨로지를 정의하고 웹서비스 광고와 웹서비스 요청간의 일치를 위한 DAML+OIL의 DL notions를 기술 하였다.

[그림 4]는 서비스 제공자의 광고와 서비스 요청자의 질의 온톨로지의 예제로 서비스 요청자는 카메라폰 기능이 있는 20만원 이하의 삼성 핸드폰을 원하고 서비스 제공자는 20만원 이하의 핸드폰을 제공한다. 여기서 서비스 요청자는 삼성 핸드폰을 원하지만 서비스 제공자가 삼성 제품을 취급하는지의 여부는 기술되지 않았다.

#### 3.2 매치메이킹 알고리즘

매치메이킹은 서비스 요청자의 요구 사항을 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 브라우징, 수정, 취소가 가능하도록 하여야 한다[최옥경, 2006]. 결국 효율적인 웹서비스 검색을 위해서는 서비스 요청과 서비스 광고간의 정확한 일치와 QoS 요소에 대한 반영이 이루어져야 한다.

기존 연구[Paolucci et al., 2002]에서 QoS 요소를 기반으로 한 매칭 매커니즘이 제안되고 있으나 다음과 같은 몇 가지 문제점을 가지고 있다. 첫째, 웹서비스의 일치 요소를 포괄적으로 다루지 못하고 부분적인 요소에만 치중하여 매칭을 수행한다. DAML-S을 기반으로 매치메이킹 알고리즘을 수행

## ■ 서비스 요청

```

<profileHierarchy : ProductSelling rdf : ID = "RequestProductSellingService">
  <profile : input>
    <profile : ParameterDescription rdf : ID = "Product_title">
      <profile : parameterName>Cellular Phone</profile : parameterName>
      <profile : restrictedTo rdf:resource = "http://www.w3.org/2001/XMLSchema # string" />
    </profile : ParameterDescription>
  </profile : input>

  <profile : input>
    <profile : ParameterDescription rdf : ID = "Product_price">
      <profile : parameterName>200,000</profile : parameterName>
      <profile : restrictedTo rdf : resource = "http://www.w3.org/2001/XMLSchema # Integer" />
    </profile : ParameterDescription>
  </profile : input>

  <profile : input>
    <profile : ParameterDescription rdf : ID = "FeatureSelection">
      <profile : parameterName> Built-in Camera </profile : parameterName>
      <profile : restrictedTo rdf:resource = "ProductProcess.daml # FeatureType" />
    </profile : ParameterDescription>
  </profile : input>

  <profile : input>
    <profile : ParameterDescription rdf:ID = "CompanyName">
      <profile : parameterName>Samsung</profile:parameterName>
      <profile : restrictedTo rdf : resource = "ProductProcess.daml # CompanyName" />
    </profile : ParameterDescription>
  </profile : input>

  <profile : precondition>
    <profile : ParameterDescription rdf : ID = "Price_range">
      <profile : parameterName>Price_range</profile : parameterName>
      <profile : restrictedTo rdf : resource = "ProductProcess.daml # Price_range" />
    </profile : ParameterDescription>
  </profile : precondition>

  <profile : output>
    <profile : ParameterDescription rdf : ID = "Product_Output">
      <profile : parameterName>Samsung Cellular Phone </profile : parameterName>
      <profile : restrictedTo rdf : resource = "ProductProcess.daml#ProductOutputType" />
    </profile : ParameterDescription>
  </profile : output>

```

## ■ 서비스 광고

```

<profileHierarchy : ProductSelling rdf: ID = "ProductSellingService">
  <profile : serviceName>ProductSelling_Agent</profile : serviceName>
  <profile : textDescription>browse a product and buy things</profile : textDescription>
  <profile : contactInformation>
    <profile : Actor rdf: ID = "Product_contacts">
      <profile : :name>ProductSelling</profile : name>
      <profile : title>Service Representative</profile : title>
      <profile : phone>82 2 824 1111</profile : phone>
    </profile : Actor>
  </profile : contactInformation>

  <profile : input>
    <profile : ParameterDescription rdf: ID = "Product_title">
      <profile : parameterName>Product_title</profile : parameterName>
      <profile : restrictedTo rdf: resource = "http://www.w3.org/2001/XMLSchema # string" />
    </profile : ParameterDescription>
  </profile : input>

  <profile : input>
    <profile : ParameterDescription rdf: ID = "Product_price">
      <profile : parameterName>200,000</profile : parameterName>
      <profile : restrictedTo rdf: resource = "http://www.w3.org/2001/XMLSchema # Integer" />
    </profile : ParameterDescription>
  </profile : input>

  <profile : input>
    <profile : ParameterDescription rdf: ID = "FeatureSelection">
      <profile : parameterName>Built-in Camera</profile : parameterName>
      <profile : restrictedTo rdf: resource = "ProductProcess.daml # FeatureType" />
    </profile : ParameterDescription>
  </profile : input>

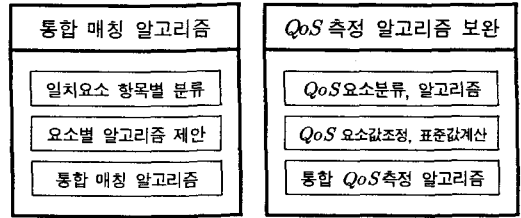
  <profile : input>
    <profile : ParameterDescription rdf: ID = "CompanyInformation">
      <profile : parameterName>Samsung</profile : parameterName>
      <profile : restrictedTo rdf: resource = "ProductProcess.daml # Company" />
    </profile : ParameterDescription>
  </profile : input>

  <profile : precondition>
    <profile : ParameterDescription rdf: ID = "Price_range">
      <profile : parameterName>Price_range</profile : parameterName>
      <profile : restrictedTo rdf: resource = "ProductProcess.daml # Price_range" />
    </profile : ParameterDescription>
  </profile : precondition>
  <profile : output>
    <profile : ParameterDescription rdf: ID = "Product_Output">
      <profile : parameterName>Cellular Phone</profile : parameterName>
      <profile : restrictedTo rdf: resource = "ProductProcess.daml # ProductOutputType" />
    </profile : ParameterDescription>
  </profile : output>

```

[그림 4] 핸드폰 구매를 위한 서비스 광고와 서비스 요청 서비스

하고 있는 [Paolucci et al, 2002]은 웹서비스의 매칭을 입출력 정보만으로 수행하여 정확한 웹서비스 검색이 힘들다. 둘째, 기존 연구들의 QoS 매칭 매커니즘은 특정 한 QoS 요소가 전체 유사도를 결정지어 결국 전체 QoS 점수가 높더라도 결코 좋은 서비스가 되지 못하는 문제점을 가지게 된다. 예를 들어 신뢰성이나 접근성은 지나치게 낮고 가격만 지나치게 높은 서비스가 전체 QoS 점수가 높아 다른 서비스에 비해 우선적으로 선택될



[그림 5] 매치메이킹 알고리즘

같다.

각각의 일치 요소들은 0-1사이의 값으로 표준화

<표 2> 웹서비스 일치 요소

분류	항목	설명
서비스 설명 정보	Service Name	서비스의 개요 기술
	Service Description	서비스 특징 및 설명 기술
입출력 정보	Input/Output Name	입출력 이름에 대한 일치
	Input/Output Datatype	입출력 데이터타입에 대한 일치
	Input/Output Constraint	입출력 제약조건에 대한 일치
서비스 기능적 표현 정보	Service Functional Description	input, output, precondition, effect을 이용하여 서비스의 기능 표현
품질 정보	QoS	가용성, 시스템 응답시간, 최대 처리량, 신뢰성, 접근성, 가격

수 있다.

결론적으로 본 연구에서는 기존 연구들의 매치메이킹 알고리즘에 대한 문제점에 대한 해결 방안으로서 QoS를 보완한 알고리즘을 [그림 5]와 같이 제안한다. 먼저 웹서비스 일치 요소를 <표 2>에서와 같이 항목별로 분류하여 정의하고 각각에 대한 알고리즘을 제안한 후 이를 바탕으로 통합 매칭 알고리즘을 개발하여 수행하였다.

두 번째 QoS 매칭 매커니즘을 보완하기 위해 QoS 요소의 값을 조정하고 각각에 대한 표준값을 계산한 후 전체 QoS 점수를 계산한 후 이를 통합 매칭 알고리즘에 반영하였다. 각각의 일치 요소들에 대한 알고리즘 측정 방식은 다음 <표 3>과

및 정규화 하였다. 각 일치 요소들의 합으로 이루어진 전체 유사도 값은 식 (1)과 같으며 각 정의별 알고리즘의 상세는 다음과 같다.

### 3.2.1 서비스 설명 정보

웹서비스 질의 시 가장 기본적인 정보는 서비스 이름과 서비스 설명 정보이다. 예를 들어 서비스 요청자가 서비스 이름이 “핸드폰 판매”이고 서비스 설명 정보가 “신용 카드 결제가 가능한 핸드폰 판매 서비스”라고 질의를 했을 때 문자열로 이루어진 서비스 이름과 서비스 설명 정보에 대한 유사도를 분석 후 적합한 서비스를 찾아 주어야

<표 3> 웹서비스 일치 요소 알고리즘

정의	분류	알고리즘
1	서비스 설명 정보	$N[name(s)] = \frac{name(s)}{100}, N[desc(s)] = \frac{desc(s)}{100}$
2	입출력 정보	$Similarity[Input(s)] = \sqrt[3]{Sim(i\_name(s)*i\_type(s)*i\_const(s))}$ $Similarity[Output(s)] = \sqrt[3]{Sim(o\_name(s)*o\_type(s)*o\_const(s))}$
3	서비스 기능적 표현 정보	$Function(S_{level}) = Sim(Input(s), Output(s))$ $0 \leq Function(S_{level}) \leq 1$
4	품질정보	$\sum(QoS) = N(A_u) * N\left(\frac{Th_s}{T_r}\right) * N(R_s) * N(Ac_s) * N(P_s)$

$$S_{total} = N[name(s)] + N[desc(s)] + Similarity[Input(s)] + Similarity[Output(s)] + function(S_{level}) + \sum(QoS) \quad (1)$$

한다. 본 연구에서는 서비스 이름과 설명 정보의 유사도를 구하기 위해서 서비스 이름 유사도를  $name(s)$ , 서비스 설명 유사도를  $desc(s)$  이라고 하고 TF-IDF(Term Frequency-Inverse Document Frequency Weighting)를 이용하여 유사도를 계산하였다. TF-IDF는 문서의 특징을 추출하기 위해서 단어의 빈도수(term frequency)를 이용하는 기존 방식의 문제점을 해결하기 위해 여러 문서에서 높은 빈도를 나타내는 용어는 관련성이 떨어진다고 판단하여 역 문서 빈도수를 단어의 빈도수와 같이 적용함으로써 그 문서를 대표하는 단어들을 효율적으로 찾는 알고리즘이다[Hayes, 2006][신진섭, 1999].

여기서  $name(s)$ 과  $desc(s)$ 은 0-100사이의 값을 가지게 되는데 이를 정규화(normalize) 하여 0-1사이의 값으로 표현 하면 다음과 같다.

[정의 1]

서비스 이름 유사도를  $name(s)$ 라 하고, 서비스 설명 정보 유사도를  $desc(s)$ 라 할 때 이를 정규화

한  $N[name(s)], N[desc(s)]$ 는 다음과 같다.

$$N[name(s)] = \frac{name(s)}{100},$$

$$N[desc(s)] = \frac{desc(s)}{100}$$

3.2.2 입출력 정보

입출력 정보에 대한 일치는 Input/Output Name, Input/Output Datatype, Input/Output Constraint 에 대한 일치를 기준으로 하며 유사도 측정은 기존의 TF-IDF 방식을 이용하여 계산하였다. 입출력 정보에 대한 값을 식으로 표현 하면 다음과 같다.

[정의 2]

서비스 입력 정보를  $Input(s)$ 라 하고, 서비스 출력 정보를  $Output(s)$ 라 할 때 이에 대한 유사도를 식으로 표현 하면 다음과 같다.

$$Similarity[Input(s)] = \sqrt[3]{Sim(i\_name(s)*i\_type(s)*i\_const(s))}$$



$$Similarity[Output(s)] = \sqrt[3]{Sim(o\_name(s) * o\_type(s) * o\_const(s))}$$

### 3.2.3 서비스 기능적 표현 정보

서비스 요청과 광고 간의 일치 여부는 서비스 기능적 측면의 입출력의 일치에 따른다. 즉 서비스 요청의 입력과 서비스 광고의 입력을 비교하여 서로 간에 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.

서비스 기능적 표현 정보에 대한 일치는 서비스 입력 정보인  $Input(s)$ 과 서비스 출력 정보인  $Output(s)$ 에 대한 일치를 기준으로 한다. 서비스 기능적 표현 정보에 대한 값을 식으로 표현하면 다음과 같다[Okkyung Choi et al, 2006].

### [정의 3]

서비스 입력 정보를  $Input(s)$ 라 하고, 서비스 출력 정보를  $Output(s)$ 라 할 때 이에 대한 서비스 기능적 표현 정보인  $Function(S_{level})$ 에 대한 일치치를 식으로 표현하면 다음과 같다.

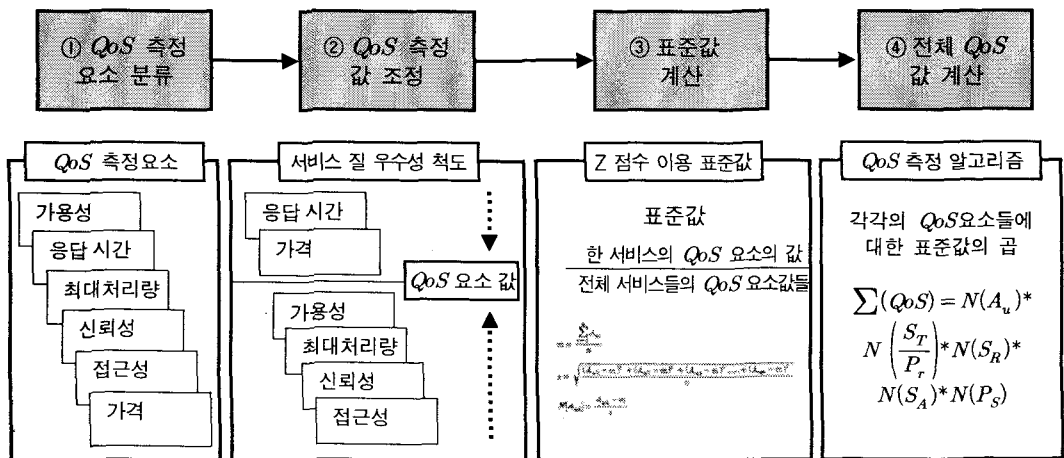
$$Function(S_{level}) = Sim(Input(s), Output(S))$$

$$0 \leq Function(S_{level}) \leq 1$$

### 3.2.4 품질 정보

보다 정확하고 신뢰할 수 있는 웹 서비스 결과를 제공 받기 위해서는 시간, 가격, 신뢰성, 성능 등의 QoS 요소들을 반영한 매칭 기법이 제공되어야 한다.

LARKS[Sycara et al, 2002]나 Infosleuth [Nodine et al, 1999]와 같은 기존 시스템들은 매치메이킹 방식에 이러한 QoS 요소들을 반영하지 못하고 있다. 또한 QoS 요소를 매치메이킹에 반영시킨 조지아 대학의 METEOS[Sycara et al, 2004]는 각각의 QoS 요소의 값에 대한 조정이 이루어지지 않고 특정 QoS 요소가 전체 QoS 유사도를 결정



[그림 6] 단계별 품질 정보(QoS) 측정 방식

지어 실질적으로 의미 있는 웹 서비스가 낮게 평가될 수 있다는 단점이 있다.

이에 본 연구에서는 품질 평가 요소를 분류하고 각각에 대한 알고리즘을 제안한다. 각각의 QoS 측정 요소별 값에 대해 알고리즘을 적용시킨 후 이에 대한 조정값과 표준값을 계산한다. 마지막으로 각각의 QoS 측정 요소별로 곱하여 전체 QoS 점수를 계산한다. 단계별 QoS 측정 방식은 다음 [그림 6]과 같으며 이에 대한 값을 식으로 표현하면 [정의 4]와 같다.

**[정의 4]**

단위 시간 동안 서비스를 사용 가능한 시간의 평균( $A_u$ ), 응답 시간( $T_r$ ), 최대 처리량( $Th_s$ ), 신뢰성( $R_s$ ), 접근성( $Ac_s$ ), 가격( $P_s$ )들과 같은 각각의 QoS 요소들에 대한 표준값의 곱으로 이루어진 QoS 측정 알고리즘은 다음과 같다.

$$\sum(QoS) = N(A_u) * N\left(\frac{Th_s}{T_r}\right) * N(R_s) * N(Ac_s) * N(P_s)$$

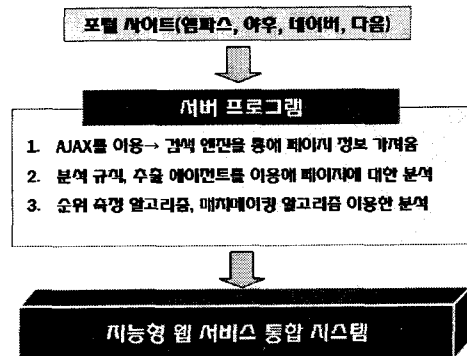
## 4. 시스템 구현 및 평가

### 4.1 시스템 구현

본 연구에서 구현한 개선된 웹서비스 검색 시스템은 국내 4개 포털 사이트(네이버, 야후, 다음, 엠파스)의 키워드, 키워드 설명, 키워드 가격, QoS 정보(네트워크 속도, 신뢰성, 가용성, 접근성), 업체 정보에 대한 정보를 데이터베이스에서 가져와서 보여 주는 시스템이다.

내부 구조상으로는 본 연구에서 제안한 알고리즘인 매치메이킹 알고리즘과 한글 검색 엔진, 온톨로지 서버를 모두 포함하고 있다. 구현 원리는

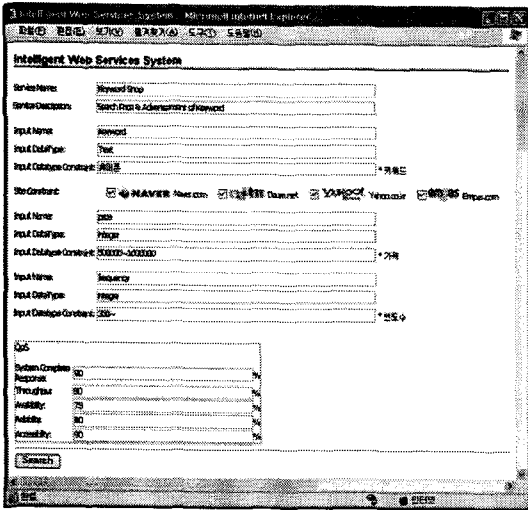
AJAX[Jesse James Garret, 2005]를 이용해 국내 주요 포털 사이트에서 서비스하는 실제 데이터를 가져와 웹서비스를 위해 정보를 가공했다. 오버추어, 다음, 네이버, 야후, 엠파스에 접속해서, 키워드 가격 정보, QoS 정보를 가져오는 ‘서버 프로그램’이 동작하고 있고, 이 ‘서버 프로그램’이 검색 엔진 프로그램과 같이 각 사이트에 접속해 웹 페이지를 가져와서 페이지에 대한 분석을 수행한 후 이를 데이터베이스에 저장시킨다. 페이지 분석시 제안한 추출 에이전트를 통해 연관성 있는 정보만 가져왔으며 정보의 정확도를 높이기 위해서 각 사이트마다의 ‘분석 규칙’을 이용하였다. 이때 사이트들이 가끔 페이지를 변경하는 현상이 있는데, 각 사이트마다 웹 페이지가 변경되면, 각 사이트 정보를 분석하는 ‘분석 규칙’도 같이 변경되도록 하였다. 마지막으로 순위 측정 알고리즘과 매치메이킹 알고리즘을 적용시켜 검색된 웹서비스에 대한 분석을 수행 후 그 결과를 반환해 준다. 시스템 구현 원리를 살펴보면 [그림 7]과 같다.



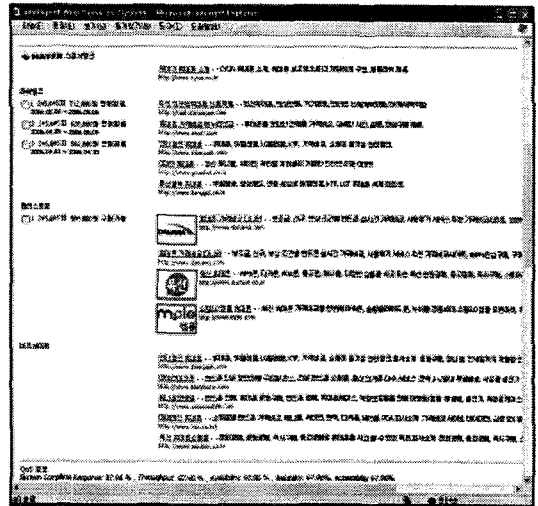
[그림 7] 시스템 구현 개념도

### 4.2 실행 결과

[그림 8]은 웹서비스 검색을 위한 사용자 인터



[그림 8] 사용자 인터페이스 화면(초기 화면)



[그림 9] 검색 결과 화면

페이스 화면으로 웹서비스 사이트, 키워드 가격 정보와 광고주 정보를 검색해주는 시스템이다. 여기서 검색에서 나오는 정보는 포털 사이트에서 서비스하는 실제 데이터를 가져와서 가공한 것이다. 내부 구조상으로는 매치메이킹 알고리즘, 순위 측정 알고리즘, 한글 검색 엔진을 포함하고 있다. 현재 시맨틱 웹의 구조를 반영하고 있다. Input Name (Keyword)에 '휴대폰'을 입력하고, Input Name (Price)에 '500000~1000000'(50만원 이상 100만원 이하 제품)을 입력하고, Input Name(frequency)에 '350~'(350이상)을 입력한 후 QoS 정보를 입력한 후 [Search]를 실행하게 되면 다음 [그림 9]와 같은 결과를 얻게 된다.

### 4.3 성능 평가

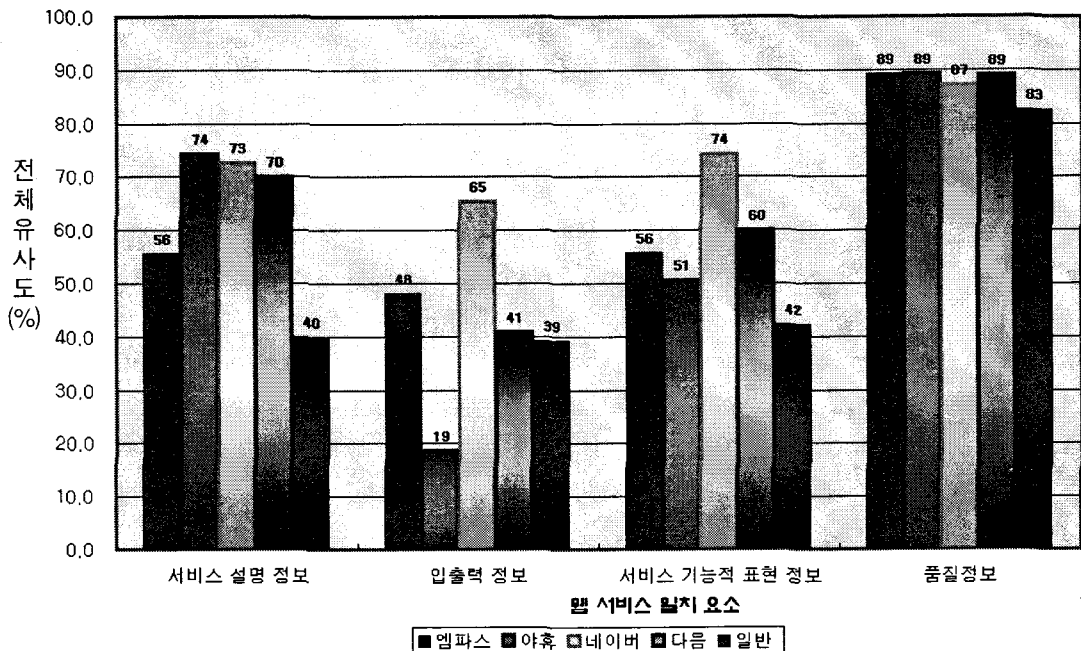
본 절에서는 개선된 웹서비스 검색 시스템의 검색 성능을 측정함으로써 본 시스템의 효율성과 유용성을 평가하고자 한다. 성능 평가의 신뢰성과

정확성을 위해 약 3개월간의 사전 운영을 통해 신뢰할 수 있는 통계 정보를 확보한 다음 실험을 진행 하였다. 또한 제안 시스템과의 비교를 위해 일반 사이트인 네이트온에서 실제 데이터를 가져와서 비교 및 분석을 수행하였다. 네이트온을 대상으로 만든 데이터는 제안 시스템에 있는 서버 프로그램을 거치지 않고, 직접 접속해서 데이터를 가져온 결과이며 비교 질의는 같은 검색어와 매칭 요소를 사용하여 객관적인 평가가 가능하도록 하였다. 실험 산정 기준은 현재 표준화된 테스트마크가 존재하지 않기에 UNIX에서 자체 개발한 서버 프로그램을 통해 2006년 6월 1부터 8월 30일까지의 백업 데이터를 기준으로 사용자가 검색한 키워드 집합에 대해서 서비스 설명 정보, 입출력 정보, 서비스 기능적 표현 정보, 품질 정보가 일치하는 비율 등을 통계 수치화 하였다. 실험에서 사용한 질의와 키워드명은 약 300개 정도의 검색 가능 키워드를 오름차순으로 정렬하여 얻은 결과를 분류한 것이다.

<표 4> 각각의 매칭 요소에 대한 전체 유사도

매칭 요소	제안 시스템				일반
	엠파스	야후	네이버	다음	네이트온
서비스 설명 정보	55.7%	74.4%	72.6%	70.2%	39.8%
입출력 정보	48.1%	18.7%	65.5%	40.9%	39.2%
서비스 기능적 표현 정보	55.8%	50.7%	74.3%	60.0%	42.3%
품질 정보	89.1%	89.1%	87.2%	89.1%	82.5%

<통합 매칭> 분석결과



[그림 10] 각각의 매칭 요소에 대한 전체 유사도

각각의 매칭 요소들에 대한 일치 점수를 비교 분석한 실험 결과인 <표 4>, [그림 10]을 살펴보면 본 연구에서 제시한 야후의 입출력 정보를 제외한 엠파스, 네이버, 다음의 매칭 요소의 유사도 점수가 일반 사이트의 유사도 점수 보다 20~30% 정도 결과가 향상된 것을 볼 수 있다.

## 5. 결론 및 향후 연구과제

현재 웹서비스는 서비스에 대한 정보를 비즈니스 레지스트리에 한정 시켜 검색의 표현력이 부족하며 의미론적 웹서비스 검색이 불가능 하다.

본 논문은 이러한 문제점을 해결하기 위해 웹

서비스에 시맨틱 웹 기술을 적용시켜 의미적이고 효율적인 웹서비스의 발견 및 검색이 가능하도록 하였다. 제안 시스템은 매치메이킹 서비스를 통해 서비스 요청자와 서비스 제공자 간의 효율적인 매칭이 가능하도록 하고 사용자 요구 사항에 대한 반영 및 검증이 가능하도록 하였다. 마지막으로 제안 시스템의 효율성과 정확성을 검증하기 위해 자체 개발한 서버 프로그램을 통해 사용자가 검색한 키워드 집합에 대해서 서비스 설명 정보, 입출력 정보, 서비스 기능적 표현 정보, 품질 정보가 일치하는 비율을 측정하는 정량적인 실험 평가를 시행하였고 그 결과 제안하는 시스템의 유사도 점수가 일반 사이트의 유사도 점수보다 20-30% 정도 향상된 결과를 가져온 것을 볼 수 있었다.

향후 연구 과제로는 사용자의 요구 사항, 제약 조건을 좀 더 세분화하여 반영할 수 있는 화면 인터페이스의 추가와 규칙 기반 검색 기법을 반영한 시스템의 개발이 요구되며 제안 시스템에 대한 효율성 및 정확성을 보다 정확하게 검증 할 수 있는 테스트 마크를 통한 비교 분석 및 평가가 수행되어야 한다.

## 참고문헌

- [1] 김제민, 박영택, “시맨틱 웹 환경에서의 개인화 검색”, 정보처리학회논문지 B, 13-B권, 5호(2006), 533~540.
- [2] 박미, 류근호, “공간객체 기반의 온톨로지와 규칙을 이용한 상황 정보 모델”, 정보처리학회논문지 D, 13-D권, 6호(2006), 789~795.
- [3] 신진섭, 이창훈, “단어의 연관성을 이용한 문서의 자동분류”, 정보처리학회논문지 A, 6-A권, 9호(1999), 2422~2430.
- [4] 이경하, 이규철, 웹서비스의 향후 발전 방향, 정보처리학회, 9권 4호, 2002.
- [5] 최옥경, 한상용, “자동화된 통합 프레임워크를 위한 시맨틱 웹 기반의 정보검색 시스템”, 정보처리학회논문지 C, 13-C권, 1호(2006), 129~136.
- [6] DARPA, The DARPA Agent Markup Language Homepage, <http://www.daml.org/>, 2006.
- [7] Haibo, Yu, Mine, T., and Amamiya, M., “An architecture for personal semantic Web information retrieval system integrating Web services and Web contents”, ICWS, (2005), 329~336.
- [8] Hayes, J. H. Dekhtyar, A., and Sundaram, S. K., “Advancing Candidate Link Generation for Requirements Tracing : The Study of Methods”, IEEE Trans on SE, Vol.32, No.1(2006), 4~19.
- [9] Jesse James Garrett, Ajax : A New Approach to Web Applications, <http://www.adaptivepath.com/publications/essays/archives/000385.php>, Feb. 2005.  
Google, <http://www.google.com/>, 2006  
LookSmart, <http://search.looksmart.com/>, 2006  
Metacrawler, <http://www.metacrawler.com/>, 2006.  
MSN Search, <http://search.msn.com/>, 2006
- [10] Nodine, M, Bohrer, W., and Ngu, A : Semantic Brokering over Dynamic Heterogeneous Data Sources in InfoSleuth, In Proc. of the International Conference on Data Engineering, 1999.
- [11] Okkyung, C., Sangyong, H, Ajith, A. Semantic Matchmaking Services Model for the intelligent Web Services. International Conference on Computational Science and Applications, UK, IEE Presss, 2006.
- [12] Paolucci, M, Kawamura, T., Payne, T., and Sycara, K., Semantic matching of web ser-

- vices capabilities, In Proc. of the 1st International Semantic Web Conference (ISWC), 2002.
- [13] Srinivasan, N., Paolucci M., and K. P. Sycara, K., Adding OWL-S to UDDI, implementation and throughput, In Proceedings of the First International Workshop on Semantic Web Services and Web Process Composition, USA, 2004.
- [14] Sycara, K., Widoff, S., Klusch, M., and Lu, J., "LARKS: Dynamic Matchmaking Among Heterogeneous Software Agents in cyberspace", *Autonomous Agents and Multi-Agent Systems*, Vol.05(2002), 173~203.
- [15] Sycara, K., Paolucci, M., Soudry, J., and Srinivasan, N., "Dynamic Discovery and Coordination of Agent-Based Semantic Web Services", *IEEE Internet Computing*, Vol.08, No.03 (2004), 66~73.  
Yahoo, <http://www.yahoo.com/>, 2006.

Abstract

## Advanced Web Services Retrieval System using Matchmaking Algorithm

Okkyung Choi\* · Jungwoo Lee\* · Sangyong Han\*\*

Recently, semantic web technology, represented by ontology building, is being combined with web services technology, creating 'Semantic Web Services' as a new promising field in information retrieval research. Accordingly, many brokering and matchmaking agents are being developed and used in the field. However, literature review revealed that most models do not take QoS(Quality of Services) into consideration.

In this study, a QoS-augmented matchmaking algorithm is developed based on service availability, response time, maximum transaction amount, reliability, accessibility and price as critical QoS items. A prototype for Intelligent Semantic Web Services System is developed using publicly available data. Performance test was conducted and reported at the end.

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

---

\* Graduate School of Information, Yonsei University

\*\* School of Computer Science and Engineering, Chung-Ang University