

온토로지 학습 방법을 활용한 (반)자동화된 웹 서비스 발견 및 조합 시스템

이용주
경북대학교 컴퓨터정보학부
e-mail:yongju@knu.ac.kr

Semi-Automated Web Services Discovery and Composition System using Learning Ontology Methods

Yong-Ju Lee
School of Computer Information, Kyungpook National University

요 약

시맨틱 웹 서비스 기술의 성공을 보장하기 위해서는 품질 좋은 온톨로지의 사용이 필수적이다. 하지만 온톨로지 사용의 중요성에도 불구하고 현재 웹 서비스를 위한 온톨로지는 거의 존재하지 않으며 이들의 구축도 쉬운 일이 아니다. 이러한 문제는 오늘날 웹 서비스의 확산과 발전을 가로막는 큰 저해 요인이 되고 있다. 본 논문에서는 웹 서비스를 개발할 때 자동 생성되는 WSDL 문서만 가지고 항목 간 숨어있는 시맨틱 정보를 찾아내어 온톨로지를 자동 구축하고, 이를 활용한 (반)자동화된 웹 서비스 발견 및 조합 시스템을 구현하는 것이다.

1. 서론

최근에 웹(web)은 정적인 웹 문서에서 지적이고 동적인 데이터 통합 환경으로 발전되고 있으며 이러한 진전을 위해 두 가지 혁신적인 변화가 제시되고 있다. 첫 번째로 웹 서비스(web services) 기술은 웹 표준을 통해 다양한 플랫폼과 서로 다른 언어로 개발된 소프트웨어 컴포넌트들에 대하여 동일한 액세스를 허용함으로써 이기종 소프트웨어 컴포넌트들 간의 상호운용성(interoperability)을 증가시키고, 다양한 소프트웨어 모듈 사이의 재사용(reuse)과 조합(composition)을 가능하게 한다. 두 번째로 시맨틱 웹(semantic web) 기술은 정보의 의미를 개념으로 정의하고 개념간의 관계성을 표현함으로써 정보를 공유시키고, 웹 상의 정보를 수집하고 처리하기 위해 더 이상 인간의 전적인 개입이 요구되지 않는다. 따라서 각종 자동화된 에이전트를 통해 정확한 정보 검색, 새로운 지식의 생성, 최적의 서비스 제공 등을 가능하게 한다.

현재 웹 서비스 기술의 주된 단점은 서비스의 발견 및 조합이 아직까지 수작업으로 수행되고 있는 것이다. 이는 오늘날 웹 서비스들이 수없이 증가되고 있는 상황에서 큰 부담이 되고 있다. 따라서 최근의 여러 연구들[1, 2]에서 시맨틱 웹 기술을 이용하여 자동적인 웹 서비스 발견 및 조합을 실현하기 위해 기계 가독형으로 웹 서비스 기능을 묘사할 수 있는 메카니즘, 즉 온톨로지(ontology)를 활용하고 있다. 이에 따라 호환 가능한 웹 서비스들 간의 실시간 발견 및 조합이 가능하다. 하지만 이러한 시맨틱 웹 서비스 기술의 성공을 보장하기 위해서는 품질 좋은 온톨로지의 사용이 필수적이다.

최근 온톨로지 사용의 중요성에도 불구하고 현재 웹

서비스를 위한 온톨로지는 거의 존재하지 않으며 이들의 구축도 쉬운 일이 아니다. 왜냐하면 현재 이들 온톨로지는 대부분 전문가의 수작업으로 구축되고 있으며 시간 및 인적 제약 때문에 실용적인 온톨로지를 구축하기가 쉽지 않다. 또한 현시점에서 매일 수백에서 수천 건씩 증가되고 있는 웹 서비스 전체에 대해 온톨로지를 수동 구축한다는 것은 거의 불가능하게 보이며, 이러한 문제는 오늘날 웹 서비스의 확산과 발전을 가로막는 큰 저해 요인이 되고 있다.

본 논문에서는 웹 서비스를 개발할 때 자동 생성되는 WSDL(Web Services Description Language) 문서만 가지고 항목 간 숨어있는 시맨틱 정보를 찾아내어 온톨로지를 자동 구축하고, 이를 활용한 (반)자동화된 웹 서비스 발견 및 조합 시스템을 구현한다.

2. 시맨틱 웹 서비스

시맨틱 웹(Semantic Web)의 근본적인 전제는 정보를 획득하는 과정에서 현재처럼 사람이 직접 수행해야 하는 인간 중심적인 인터페이스를 컴퓨터 프로그램이 이해하고 처리할 수 있는 형태로 확장시키는 것이다. 따라서 시맨틱 웹 개념을 기존의 표준 웹 서비스 기술에 확장시키면 자동화된 서비스 발견 및 실행이 가능해진다. OWL-S(Web Ontology Language-Service)는 가장 최근에 발표된 시맨틱 기반 웹 서비스 언어이다. OWL-S 온톨로지는 기계 가독형으로 웹 서비스 기능을 묘사할 수 있는 메카니즘을 제공한다. 이 메카니즘은 자동화된 웹 서비스 발견 및 조합을 가능하게 만든다.

OWL-S는 서비스를 기술하기 위해 presents, describedBy, supports의 세 가지 속성을 선언하고 있으며, 이는 서비스 프로

파일(Service Profile), 서비스 모델(Service Model), 서비스 그라운드(Service Grounding)의 세 가지 클래스로 구성된다. 서비스 프로파일은 서비스의 역할을 기술하고 있다. 이는 사용자가 서비스 탐색 에이전트를 이용하여 원하는 서비스를 찾을 때 이를 위한 필요한 정보를 제공하고 있다. 서비스 모델은 서비스 작업 프로세스를 기술하고 있다. 프로세스는 구성 개수에 따라 원자 프로세스와 복합 프로세스로 구분된다. 마지막으로 서비스 그라운딩은 에이전트가 서비스를 액세스할 수 있도록 자세한 사항을 기술하고 있다. 요약하면, 서비스 프로파일은 에이전트가 서비스를 탐색하는데 필요한 정보를 제공하고 있으며, 서비스 모델 및 그라운딩은 에이전트가 서비스를 사용할 수 있도록 충분한 정보를 제공하고 있다. 그러므로 웹 서비스 발견 및 조합 문제를 해결하기 위해서는 주로 서비스 프로파일을 이용하게 된다. <표 1>은 서비스 프로파일의 내용을 요약 기술한 것이다.

<표 1> 서비스 프로파일의 내용

구분	항목	설명
제공자에 대한 정보	serviceName	서비스 이름
	textDescription	서비스 내용을 요약한 텍스트
	contactInformation	서비스 관련 연락처
기능에 대한 정보	hasInput	입력항목
	hasOutput	출력항목
	hasPrecondition	서비스 사용 전 만족해야할 조건
	hasEffect	서비스 실행 후 변화
서비스 특성을 구체화한 특징	serviceCategory	
	· categoryName	분류명칭
	· taxonomy	분류체계
	· value	분류체계에서의 값
	· code	분류체계 코드
	qualityRating	품질등급
	serviceParameter	
	· ParameterName	매개변수 이름
· sParameter	매개변수 값	

기존의 전문가에 의한 수작업으로 온톨로지를 구축하는 방법에서는 <표 1>과 같은 내용을 온톨로지 구축 툴(tool)을 사용하여 만든다. 이러한 정보들이 충실히 기술만 될 수 있다면 지금까지 제안된 시맨틱 매칭 알고리즘들은 훌륭히 적용될 수 있을 것이다. 문제는 이러한 온톨로지 정보가 현재로서는 거의 존재하지 않으며 이들의 구축도 쉬운 일이 아니다. 일반적으로 웹 서비스 제공자는 UDDI 저장소 내에 자신의 웹 서비스와 관련된 WSDL 파일, 그리고 그 서비스에 대한 간단한 설명을 등록함으로 공개한다. 따라서 UDDI로부터 다음과 같은 정보를 얻을 수 있다.

- 웹 서비스 정보: 하나의 웹 서비스는 WSDL 파일 내에 서비스의 이름과 이에 대한 텍스트 설명이 기술된다. 그리고 UDDI 저장소 내에 연락처나 카테고리화 같은 비즈니스에 관한 정보도 입력된다.
- 오퍼레이션 정보: 각 오퍼레이션은 WSDL 파일 내에서 오퍼레이션에 대한 이름과 이에 대한 텍스트 설명이 기술된다.
- 매개변수 정보: 오퍼레이션의 입출력은 각각 매개변수들의 집합으로 구성되어 있다. WSDL 파일에서는 각 매개변수에 대한 이름과 데이터타입이 기술되어 있다.

위와 같은 UDDI 정보로부터 <표 1>에 나와 있는 OWL-S 서비스 프로파일 내용을 취득할 수 있다. 다만 WSDL 파일에서는 구문(syntactic) 정보만 있을 뿐 온톨로지에서 요구되는 동일(equivalent) 또는 계층(hierarchy) 관계와 같은 객체지향 클래스에 대한 정보(예, Customer equivalent Person 또는 Sedan subclassOf Car)가 없다. 따라서 본 연구에서는 이러한 WSDL 구문 정보만 가지고 항목 간의 숨어 있는 시맨틱 정보를 찾아내어 온톨로지를 자동 구축하고 이를 이용한 (반)자동화된 웹 서비스 발견 및 조합 시스템을 구현한다.

3. 온톨로지 학습 방법

사용자들이 시도하기를 원하는 웹 서비스에 대한 검색 유형은 다음과 같이 세 종류로 요약될 수 있다.

- 유사 오퍼레이션 발견: 하나의 오퍼레이션이 주어졌을 때 이와 유사한 오퍼레이션들을 찾는다.
- 유사 입출력 발견: 비슷한 입력(또는 출력)을 가진 오퍼레이션을 찾는다.
- 오퍼레이션 조합: 하나의 오퍼레이션으로는 다양하고 복잡한 사용자 요구사항을 충분히 만족시켜줄 수 없을 때 오퍼레이션 조합을 통해 새롭고 유용한 솔루션을 얻는다.

본 논문에서는 위의 요구사항을 쉽게 해결할 수 있도록 다음과 같은 두 가지 문제에 초점을 맞춘다. (1) 오퍼레이션 매칭: 오퍼레이션에 관한 질의 템플릿(query template)이 주어졌을 때 이와 유사한 오퍼레이션 리스트를 반환한다. 참고로 유사 입출력 발견 문제는 이 매칭 방법에 포함될 수 있다. (2) 오퍼레이션 조합: 자동화된 메카니즘으로 오퍼레이션 워크플로우를 생성한다. 워크플로우의 시작 단계에서부터 점차적으로 하나씩 오퍼레이션이 추가된다. 각 단계에서 사용자는 유사 오퍼레이션을 탐색하고 그가 의도한 바를 잘 이룰 수 있는 가장 적합한 오퍼레이션을 선택한다.

3.1 매개변수 클러스터링

본 연구의 핵심 내용은 웹 서비스 매개변수들에 대해 의미적으로(semantically) 같은 개념들을 묶고(clustering), 각 단어들 간의 계층관계를 구축하여 단어들 사이에 숨겨져 있는 시맨틱 개념을 활용하는 것이다.

보다 효율적인 웹 서비스 탐색을 위해서는, (1) 오퍼레이션 입출력 매개변수들은 일반적으로 다수의 단어가 연결된 복합단어로 이루어져 있으므로(예, ClientName), 이들에 대한 토큰화(tokenization)가 요구된다. (2) 올바른 매치를 발견하기 위해서는 스템밍(stemming) 뿐만 아니라 시소러스(thesaurus)를 통한 단어의 뜻도 고려되어야 한다. (3) 다수의 매개변수들에 대한 몇 개의 부분 일치도 고려해야만 한다. (4) 기존의 전통적인 클러스터링 알고리즘과는 다른 새로운 클러스터링 알고리즘의 적용이 요구된다.

매개변수들을 토큰화하여 텀으로 분리한 후, 관련성이

많은 텀들에 대해 클러스터를 형성하면 이 클러스터는 각각의 단어가 아닌 하나의 의미있는 개념(concept)을 나타낸다. 이러한 클러스터는 “매개변수들이 동시에 자주 나타난다면, 그것들은 같은 개념을 나타내는 경향이 있다”는 가정 하에 하나의 특별한 연관규칙(association rules)[2]에 따라 만들어 진다.

연관규칙 R 은 조건부와 결과부로 구성되며 텀 t_1 이 일어나면 t_2 가 일어난다는 의미로 $R: t_1 \rightarrow t_2$ 와 같이 표현될 수 있다. 따라서 연관규칙을 탐사하는 것은 적절한 텀 t_1 과 t_2 를 선택하는 문제로 볼 수 있으며 이를 위해 규칙 R 에 대한 지지도(support)와 신뢰도(confidence)를 고려하고 있다. 여기서 신뢰도가 임계치 δ 보다 크면(즉, $t_1 \rightarrow t_2$ (confidence $>$ δ)) 텀 t_1 과 t_2 는 밀접하게 연관되었다고 말할 수 있다. 이 알고리즘은 결과적으로 높은 점수(score)를 갖도록 클러스터를 형성하는 것이 목표이다. 높은 점수는 cohesion(한 클러스터 내의 텀들과의 응집력)은 높고, correlation(다른 클러스터 텀들 간의 상호관계)은 낮은 것을 의미한다.

이러한 클러스터링 기법을 이용한 웹 서비스 탐색에서는 단순히 입출력 매개변수 텀들의 빈도수에 의존하는 것이 아니라 각 텀들 간의 상호연관성을 이용해 관련된 단어들끼리 클러스터링 함으로써 보다 효과적인 웹 서비스 탐색을 가능하게 한다. 그러나 이 기법은 연관성 높은 단어들을 한 클러스터에 묶어서 단지 동일한(equivalent) 개념처럼 취급할 뿐 계층관계에 따라 사용자의 요구사항을 정확하게 표현하는 시맨틱 기능은 제공하지 못하고 있다.

3.2 계층관계 온톨로지 구축 기법

본 기법의 주된 아이디어는 매개변수 내의 각 단어들 사이의 상관관계를 취득하고, 비교되는 단어들이 서로 유사하고 상관관계가 조건에 일치한다면 그 비교를 매치하는 것이다. 이러한 구축 기법은 “사람들이 단어를 조합하여 복합단어로 된 매개변수를 만들 때 일반적으로 비슷한 패턴을 사용한다[3]”는 관찰로부터 시작한다. 일반적으로 사람들이 어떤 한 개념을 표현할 때 다양한 방법들이 있을 수 있지만 공간적인 제약 하에서는 비슷한 패턴을 사용하는 경향이 있다. 이러한 패턴들은 명사+명사, 접두사+명사+명사, 형용사+명사, 명사+전치사+명사, 그리고 동사+명사의 형태를 취한다.

첫 번째 단계로 각 단어들 간의 상관관계를 취득하여 그들을 온톨로지에 저장한다. 패턴 관찰로부터 WSDL 과일의 매개변수에 대한 변환 규칙(rule)은 다음과 같다.

- 규칙1(Noun1+Noun2)은 매개변수가 명사1의 속성(property)이 된다.
 - Parameter **isProperty** Noun1
- 규칙2(Prefix+Noun1+Noun2)는 두 개의 상관관계가 형성된다. 즉 매개변수가 접두사+명사1의 속성이 되는 것과 매개변수가 명사2의 자식관계(subclass)가 되는 것이다.
 - Parameter **isProperty** Prefix+Noun1
 - Parameter **subClassOf** Noun2
- 규칙3(Adjective+Noun)은 매개변수가 명사의 자식관계가 된

다.

- Parameter **subClassOf** Noun
- 규칙4(Noun1+Preposition+Noun2)는 매개변수가 명사2의 자식관계가 된다.
 - Parameter **subClassOf** Noun2
- 규칙5(Verb+Noun)는 매개변수가 명사의 자식관계가 된다.
 - Parameter **subClassOf** Noun

위와 같은 규칙을 사용하여 온톨로지가 구축되고 나면, 다음 단계는 질의문에 의해 두 개념 간 매칭을 시키는 것이다. 즉 두 개의 온톨로지 개념이 다음 조건을 만족하면 매치된다.

- 어떤 개념이 다른 개념의 속성일 경우
(예, CompanyID **isProperty** Company)
- 어떤 개념이 다른 개념의 자식관계인 경우
(예, virtualAccount **subClassOf** Account)

이러한 조건을 적용함에 따라 관련 없는 개념들 사이의 매칭을 피할 수 있다. 예를 들면, companyID와 countryID는 각각 다른 개념의 속성이므로 매치에서 배제된다. 계층관계 온톨로지 구축 기법은 관련 없는 개념들의 매칭을 피할 수 있으므로, 매치되는 후보 집합들은 클러스터링 기법에 의해 생성되는 결과보다 더욱 정확한 매칭을 얻을 수 있다.

4. 웹 서비스 발견 방법 및 조합

4.1 유사 오퍼레이션 발견

질의 템플릿 Q는 <표 1>로부터 다음과 같이 6개의 튜플로 표현될 수 있다.

$$Q = \langle name, description, Is, Os, category, location \rangle$$

여기서, *Is*는 입력항목, *Os*는 출력항목을 표시하고 *category*는 NAICS와 UNSPSC와 같은 분류시스템을, *location*은 ISO 3166 Geographic 코드시스템을 사용한다.

유사도를 측정하기에 앞서 정확성을 향상시키기 위해 각 단어들에 대해 다음과 같은 전처리 과정을 수행한다. 먼저 복합단어로 구성된 매개변수들을 파싱(parsing)하여 텀으로 분리하고 스테밍과 불용어(stop word) 필터링을 수행한다. 그 다음 단어 내 약어들이 확장되고 동의어 리스트를 발견하기 위해 시소러스가 사용된다.

본 연구에서 유사 오퍼레이션 발견 과정은 다단계 매칭 방법에 의해 수행된다. 즉 계산량이 많은 온톨로지 매칭은 다른 항목들보다 늦게 처리하고, 분류 코드나 텍스트의 경우 적은 계산량으로 높은 질의 처리 효율을 가지고 있으므로 다른 항목들보다 먼저 처리한다.

(1) 코드 일치 방법: 질의 템플릿에 카테고리(*category*)와 위치(*location*) 정보가 있으면 이는 분류코드로 이루어져 있으므로 코드 일치(exact) 방법을 사용하여 매칭을 수행한다.

(2) 이름 및 설명 유사도: 오퍼레이션의 이름과 설명은 텍스트로 구성되어 있다. 텍스트에 대해 완전 일치 방법을 사용하면 너무 제한된 탐색이 이루어질 수 있기 때문에 좀 더 유연한 매칭을 위해 IR 분야에서 널리 사용되고 있

는 TF/IDF 방법을 사용하여 매칭의 유사도를 구한다.

(3) 매개변수 유사도: 입출력 매개변수에 대해서는 3장에서 서술된 바와 같이 클러스터링과 계층관계 온톨로지가 구축된다. 오퍼레이션 내에는 매개변수들이 몇 개 존재하지 않으며, 하나의 오퍼레이션 내에 같은 매개변수는 거의 발생되지 않기 때문에 기존의 TF/IDF 방식을 그대로 적용하기는 어렵다. 따라서 관련성이 많은 텀들을 개념으로 클러스터링하고 이런 텀들을 단어의 백으로 재배치한 후 TF/IDF 방법을 적용한다. 또한 계층관계 온톨로지를 활용함으로써 검색 결과의 정확률을 향상시킬 수 있다. 매개변수 유사도를 계산할 때 계층관계 온톨로지 개념에 위배되는 매개변수 쌍들을 배제하여 사용자가 만족할 수 있는 품질 좋은 것만 정제한다. 즉 매치되는 매개변수 쌍들 중에서 3.2절에서 설명한 패턴 상관관계를 조사하여 부모 클래스가 일치하지 않을 경우 이 쌍들을 검색 결과에서 배제한다.

4.2 오퍼레이션 조합

본 연구에서는 (반)자동화된 웹 서비스 조합 시스템을 구현하기 위해 순차적 목표 지향 접근방법을 제안한다. 워크플로우의 시작 단계에서부터 점차적으로 하나씩 새로운 오퍼레이션들이 추가된다. 각 단계에서 새로운 오퍼레이션을 워크플로우에 첨가시킬 때 사용자는 먼저 질의 템플릿 Q 를 생성한다. 일단 Q 가 생성되면 이는 웹 서비스 탐색 모듈로 보내지고, Q 와 기존 웹 서비스들 간의 유사도 측정에 따라 우선순위가 매겨진 오퍼레이션들이 반환된다. 이때 사용자는 그가 의도한 바를 가장 잘 이룰 수 있는 적합한 오퍼레이션을 하나 선택한다. 일단 워크플로우가 완성되고 나면 워크플로우 엔진에 의해 이들 프로세스들이 자동으로 수행되어야 한다.

본 논문에서 구현된 (반)자동화된 웹 서비스 조합 시스템은 클라이언트-서버 구조로 되어 있다. 클라이언트 프로그래밍 기술은 워크플로우 디자이너를 구현하기 위해 자바 애플릿(applets)을 사용하였고, 서버는 워크플로우를 WS-BPEL 스펙으로 변환하고 이를 실행하기 위해 자바 서블릿(servlets)으로 구현되었다.

(1) 클라이언트 프로그래밍: 워크플로우를 표현하는 가장 자유스러운 방법은 사이클이 없는 방향성 그래프(DAG: Directed Acycle Graph) 방식이다. 여기서 정점(vertex)은 태스크를 표현하고 간선(edge)은 데이터 흐름을 나타내며 이 그래프는 $DAG=(V, E)$, 즉 정점들의 집합 V 와 간선들의 집합 E 로 구성된다. 구현된 디자이너는 "drag & drop" 방식으로 워크플로우를 작성함으로써 사용자에게 고수준의 인터페이스를 제공한다. 이러한 인터페이스를 구현하기 위해 클라이언트 측 프로그래밍은 자바 애플릿을 사용하는데, 애플릿은 사용자가 웹 브라우저로 웹 사이트를 접속할 때 사용자의 PC로 프로그램이 자동 다운로드된 후 실행되기 때문에 서버의 로드를 줄여줄 수 있다. 하나의 문제는 애플릿은 클라이언트 측 샌드박스(sandbox) 환경에서 작동되기 때문에 파일을 읽을 때 극히 제한된 권한을 가진다. 따라서 WSDL 파일의 읽기/쓰

기 작업은 모두 서버 측에서 수행된다. 이를 위해 서블릿은 애플릿과 동일한 codebase에 존재해야 한다.

(2) 서버 프로그래밍: 서버는 서블릿으로 구현되어 있으며 서블릿은 플랫폼 독립적인 서버 측 자바 프로그램이다. 클라이언트 측에서 워크플로우가 일단 완성되고 나면 이를 서버 쪽에 전송하고 서버에서는 워크플로우를 WS-BPEL 스펙으로 변환하고 BPWS4J 엔진에 배치(deploy)시켜야 한다. 이를 위해 먼저 클라이언트 프로그램에서는 완성된 워크플로우를 binary 형태로 서블릿에 전달한다. 서블릿은 binary 형태로 된 그래프를 받아서 그래프 파서인 GraphParser를 호출한다. 그래프 파서는 워크플로우를 파싱하여 관련된 WSDL과 WS-BPEL 문서를 자동으로 생성한다. 그 다음 서버 쪽의 BPEL 배치함수 Deploy()를 부른다.

WSDL과 WS-BPEL 문서가 생성되고 나면 배치함수에 의해 이들 문서가 BPWS4J 엔진에 배치될 수 있다. 먼저 WS-BPEL 파일의 partnerLink에서 partnerLinkType을 찾은 다음 WSDL 파일에서 동일한 이름을 가진 partnerLinkType들을 찾는다. 이로부터 role, portType, 그리고 실제 웹 서비스 URL을 얻은 후 서블릿을 호출한다. 서블릿은 IBM BPWS4J 엔진을 대신해서 기동된다. 이는 현재의 BPWS4J 엔진이 단지 웹 기반 관리자 툴을 통해서만 등록을 허용하기 때문에 엔진 그 자체로는 자동적인 BPEL 등록이 불가능하다. 따라서 프로그래밍 인터페이스가 지원되는 엔진 포크시를 만들어야만 한다. 처리과정은 IBM WSGateway 컴포넌트를 초기화한 후 WSDL과 BPEL 파일을 불러온다. 그리고 SOAP RPC(Remote Procedure Call) 채널을 통해 WS-BPEL 문서를 BPWS4J 엔진에 배치한다.

5. 결론

본 논문에서는 온톨로지 학습 방법을 활용한 (반)자동화된 웹 서비스 발견 및 조합 시스템을 구현하였다. 본 논문의 핵심 내용은 웹 서비스 매개변수들에 대해 의미적으로 같은 개념들을 클러스터링으로 묶고, 각 텀들 간의 계층관계 온톨로지를 구축하여 텀들 사이의 숨겨져 있는 시맨틱 개념을 활용하는 것이다.

참고문헌

- [1] Khalid Belhajjame and Marco Brambilla, "Ontology-Based Description and Discovery of Business Processes," BPMDS 2009 and EMMSAD 2009, LNBP 29, pp. 85 - 98, 2009
- [2] X. Dong, A. Halevy, J. Madhavan, E. Nemes, J. Zhang, "Similarity Search for Web Services," VLDB 2004
- [3] P. Velardi, P. Fabriani, M. Missikoff, "Using Text Processing Techniques to Automatically Enrich a Domain Ontology," ACM Int. Conf. on Formal Ontology in Information Systems, 2001