OWL 속성을 이용한 온톨로지 간 의미 유사도 측정 방법

안우식°, 박정은, 오경환
서강대학교 컴퓨터학과
{asheed°, jepark, kwoh}@sogang.ac.kr

Similarity Measure between Ontologies using OWL Properties

Woosik Ahn°, Jung-Eun Park, Kyung-Whan Oh
Dept. of Computer Science and Engineering, Sogang University

요 약

인터넷이 보다 대중화되고 광범위해지면서 의미적 관계에 따라 정보를 저장하는 온톨로지 시스템이 미래의 지능적인 컴퓨터를 위한 적절한 수단으로 각광받고 있다. 하지만 온톨로지와 같은 메타 데이터를 사용한 방법은 그 사용 목적 또는 작성자의 개인적인 관점에 따라 다양한 이질적인(heterogeneous) 형태를 띠게 된다. 이러한 이질적인 정보들은 데이터가 다른 시스템에서 처리되는 것을 어렵게 한다. 정보의 상호 운용성을 보장하기 위해서는 서로 다른 온톨로지 시스템 간의 개체에 대한 유사도를 평가할 수 있어야 한다. 따라서 두 개의 다른 OWL 언어로 정의된 온톨로지 사이에서 두 개의 엔티티의 유사도를 측정하기 위한 새로운 유사도 척도(similarity measure)를 제안하였다. 이는 온톨로지 상의 이질적인 정보를 통합하는데 사용되며, 온톨로지 비교(comparison), 정렬(alignment), 매칭(matching) 그리고 병합(merging)의 기반이 되는 중요한 기법이다. 새로운 유사도 척도는 특정한 매핑 정보를 사용하지 않고 온톨로지 언어의 속성을 기반으로 하므로 OWL을 사용한 온톨로지 간의 유사도 검색에 곧바로 적용될 수 있는 장점을 지닌다.

1. 서 론

월드 와이드 웹(WWW)이 사람들 사이의 정보 전달(communication)에 있어 혁명을 일으킨 뒤, 웹의 잠재성은 웹 상의 정보나 서비스에 빠르게 접근할 수 있는 또 다른 기술의 탄생을 요구하였다. 이러한 요구를 충족하기 위해 웹의 창시자 팀 버너스 리(Berners-Lee, T.)와 그의 동료들은 시맨틱 웹(semantic web)을 창시하였다.[1]

시맨틱 웹은 기존의 웹과는 달리 컴퓨터가 정보의 의미를 이해하고 조작할 수 있는 메커니즘을 제공한다. 이러한 시맨틱 웹은 온톨로지(ontology)를 바탕으로 구성되는데 이로 인해 많은 정보들을 의미적 관계에 따라 구분하여 사용할 수 있게 되었다. 하지만 온톨로지의 가장 큰 문제점은 온톨로지를 작성하는 사람들이 자신의 사용 목적에 따라 제각기 다른 형태로 작성하는 것이다. 이로 인해 실제로는 같은 개념을 뜻하지만 온톨로지 상에서는 다른 표현 형태로 나타나는 문제점이 있다.[2]

본 논문에서는 두 개의 다른 OWL DL 로 정의된 온톨로지 사이에서 두 개의 엔티티의 유사도를 측정하기 위한 새로운 유사도 척도를 제안한다. OWL DL 온톨로지 내의 속성과 구조를 바탕으로 유사도를 측정할 것이다. OWL DL의 모든 속성 중에서 가능한 한 관련이 있는 속성을 추출하여 그 추출된 속성을 바탕으로 부분 유사도를 계산한 후, 속성별로 미리 정의된 가중치를 적용하여 속성에 기반한 유사도를 계산한다.

본 논문의 나머지 부분은 다음과 같이 구성되어 있다. 2장에서는 시맨틱 웹과 온톨로지의 정의와 구조에 대해 살펴본다. 3장에서는 본 논문에서 제안하는 OWL 의 속성과 가변 가중치를 기반으로 한 유사도 측정 방법에 대해 살펴보고 4장에서는 실험을 통해 두 개의 서로 다른 온톨로지 간의 유사도 측정 결과와 속성의 개수, 초기 가중치 할당이 미치는 영향에 대해서 살펴본다. 마지막 장에서는 결론과 함께 향후 연구되어야 할 과제들에 대해서 기술하고자 한다.

2. 관련연구

시맨틱 웹과 관련되어 XML 문법을 사용하는 이유는 이러한 구조화된 문서의 생성을 유도한다는 점과 태그 이름을 사용자가 임의대로 정의할 수 있기 때문에 의미정보를 손쉽게 태그에 적용할 수 있다는 점이다.[3]

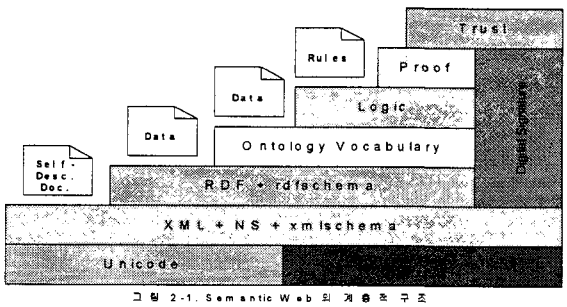

그림 2-1. Semantic Web 의 계층적 구조

그림 1. 시멘틱 웹의 계층적 구조

Semantic Web은 그림 1 과 같은 계층 구조(layered structure)를가진다. 가장 하위 레벨에서 웹 프로토콜에서 자원을 지칭하기 위한 주소지정(addressing) 방법인 URI가 밑받침되고 이를 기반으로 XML과 Namespace, RDF와 RDF 스키마, 온톨로지의 순서로 연구가 진행되고 있으며 그 위의 계층인 Logic에 대해서는 인공지능의 추론연구를 밑받침으로 일부 연구가 시작되었다. 또한 보다 더 상위 계층인 Proof와 Trust는 시맨틱 웹 정보의 신뢰성과 보안에 관한 내용으로서 아직 개념 정도만 얘기되고 있으며 차후 연구과제로 제시되고 있다.[4]

최근에는 시맨틱 웹을 표현하는데 중요한 언어로 인공 지능 분야에서 온톨로지가 널리 사용되고 있다. 일반적으로 온톨로지는 대상을 범주화하여 명칭을 붙이거나 개념화된 구조를 총칭하는 의미로 사용되며 각각의 범주명은 개념이나 범주로 언급된다. 구조적으로 온톨로지는 분류적인 계층 구조(taxonomic hierarchies)로 여겨지기도 한다. 어떤 개념이나 클래스들을 계

층적으로나 또는 격자(lattices)로 표시할 수 있다.

3. OWL 속성을 이용한 유사도 측정

이번 장에서는 OWL DL의 속성들을 기반으로 가중치를 적용하여 새로운 Similarity Measure를 정의하고자 한다. 본 논문에서 사용될 엔티티는 이름을(URI) 가지거나 또는 가지지 않은(Anonymous) 클래스와 관계(Relation)로 한정하였다. 또한 엔티티는 rdf:id, rdfs:range, rdfs:domain, owl:subClassOf, owl:equivalentClass 등과 같은 RDF(S) 또는 OWL DL 어휘로 한정하였다. 각각의 속성들은 엔티티의 전체 의미의 한 부분이므로, 이 속성들을 사용하여 측정된 local similarity를 사용하여 전체 엔티티의 유사도를 측정할 수 있다.

3.1 정의

OWL DL 온톨로지는 RDF 문서이고, RDF triple의 집합으로 가정하였다. O를 온톨로지, (s, p, o)를 triple이라 두자. s는 subject, p는 predicate, o는 object를 의미한다. 즉, O = {(s, p, o)} 이다.

엔티티 e를 subject로 하는 RDF triple의 집합을

$$T(e) = \{(e,p,o) \mid (e,p,o) \in O\}$$ 라고 표현한다.

엔티티 e를 subject로 하는 predicate의 집합

$$P(e) = \{p \mid \exists o, (e,p,o) \in T(e)\}$$

엔티티 e를 subject로 하고 p를 predicate로 하는 object의 집합을 $O(e,p) = \{o \mid (e,p,o) \in T(e)\}$ 로 정의하였다.

마지막으로 엔티티 e를 subject로 하는 predicate-object 쌍의 집합을 $E(e) = \{(p,o) \mid (e,p,o) \in T(e)\}$ 로 정의하였다.

온톨로지 O1 의 엔티티 e1과 온톨로지 O2 의 엔티티 e2 사이의 Similarity Measure는 Sim(e1, e2) 으로 표현하고 다음 두 값에 의해 결정된다.

3.2 엔티티 간의 유사도 측정

엔티티 간의 유사도를 구하기 위해서는 triple에서 관련된 속성들을 얻어와야 한다. 즉, 두 엔티티 사이의 유사도는 E(e1)과 E(e2) 쌍의 집합 사이의 유사도 이다. 정규화된 유사도 값을 얻기 위해 다음과 같은 절차를 수행한다.

1) predicate에 따른 부분 유사도를 측정하기 위해, 두 엔티티 e1, e2가 가지는 object의 집합을 얻어낸다. 즉 O(e1,p), O(e2,p)를 얻는다. 두 개의 object 집합 O(e1,p), O(e2,p)에 대하여 아래 알고리즘을 적용한다.

a) o1i⊂O(e1,p), o2j⊂O(e2,p) 인 두 집합 내의 object의 집합을 각각 o1j, o2j라 하자.

b) Simobj(o1i,o2j)를 최대화하는 o1i, o2j를 찾는다. (o1i,o2j)는 개체쌍이다.

c) O(e1,p) 집합에서 o1i 를, O(e2,p) 집합에서 o2j 를 각각 제거한다.

d) (o1i,o2j) 쌍이 더 이상 발견되지 않을 때 까지 b)를 반복한다.

2) predicate p에 대한 e1과 e2의 부분 유사도는 다음과 같다.

$$Sim_{partial}(e_1,e_2,p) = \frac{\sum_{(o_1,o_2)\in Pairing(O(e_1,p),O(e_2,p))} Sim_{obj}(o_1,o_2)}{\max(|O(e_1,p)|,|O(e_2,p)|)}$$

$$Sim_{obj}(o_1,o_2) = \begin{cases} 1 & , o_1 = o_2 \\ 0 & , o_1 \neq o_2 \end{cases}$$

그림 2. 의 두 온톨로지에서 나타난 ANIMAL 개체에 대해 rdfs:label predicate에 대한 부분 유사도를 구하는 예를 들어 보겠다. 두 개체 집합 O(ANIMAL1, rdfs:label)과 O(ANIMAL2, rdfs:label)은 표 1. 과 같다.

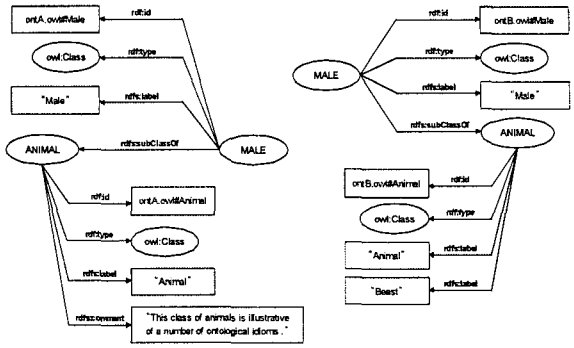

그림 3-1. 두 개의 다른 온톨로지의 축약된 RDF Graph 부분

그림 2. 두 개의 다른 온톨로지의 축약된 RDF Graph 부분

표 1. 온톨로지 A,B사이의 rdfs:label predicate에 대한 부분 유사도

	온톨로지 A	온톨로지 B
개체 집합	{"Animal"}	{"Animal", "Beast"}

구해진 집합으로부터 rdfs:label에 대한 부분 유사도를 계산하면 다음과 같다.

$$Sim_{partial}(ANIMAL_1, ANIMAL_2, rdfs:label)$$

$$= \frac{1}{\max(|\{"Animal"\}|, |\{"Animal","Beast"\}|)}$$

$$= \frac{1}{\max(1,2)} = \frac{1}{2}$$

이렇게 구해진 각각의 predicate에 대한 부분 유사도를 사용하여 전역 유사도를 구할 수 있는데, 이를 위해서 가중치 합을 사용한다. 하지만 서론에서 제기하였듯이 온톨로지는 그 사용 목적이나 작성자에 따라 다르기 때문에 엔티티의 속성이 존재할 수도, 그렇지 않을 수도 있다. 따라서 고정된 가중치 합을 적용하는 방법은 비효율적일 수가 있다. 이 문제를 해결하는 방법은 3.4 절에서 살펴보도록 하겠다.

3.3 그래프 구조 상의 유사도 측정

OWL 언어는 RDF 그래프로 표현될 수 있다. 이 절에서는 두 엔티티 간의 유사도를 속성 뿐 아니라 RDF 그래프 상에서의 구조적인 의미를 추가하였다.

본 논문에서는 그래프 상의 유사도 Sim_{graph}(e1,e2)를 두 엔티티가 가지는 tripple의 최대 크기에 비례한 유사한 predicate의 수의 비율로 정의 하였다. 수식으로는 다음과 같다.

$$Sim_{graph}(e_1,e_2) = \frac{\varphi(P(e_1),P(e_2))}{Max(|P(e_1)|,|P(e_2)|)}$$

φ 는 두 엔티티의 predicate중 유사한 predicate의 개수

유사한 predicate란 owl:cardinality, owl:minCardinality, owl:maxCardinality 등의 유사한 의미를 가지는 predicate를 의미한다. 여기에서 유사한 predicate를 사용한 이유는 회소성의 문제를 해결하고 속성의 의미를 유사도 값에 포함하기 위해서 이다.

3.4 가변 가중치를 사용한 전체 유사도

예를 들어, 세 가지의 속성(lang:id, lang:label, lang:comment)만을 제공하는 간단한 온톨로지 언어 lang이 있다고 가정하자. 엔티티의 속성은 이 세 가지 속성 중 임의의

조합이 된다. 각각의 속성에 대한 가중치가 0.5, 0.35, 0.15 순으로 미리 지정되어 있다고 하고 만약 두 엔티티가 오직 lang:id라는 속성만 가지고 있다고 하면 이 두 엔티티의 최종 유사도는 0.5가 된다. (이 경우 실제로 두 엔티티는 같다.)

이를 해결하기 위해 본 논문에서는 유사도를 계산하는 중간에 유사도를 변경할 수 있는 가변 가중치 개념을 추가하였다. 3.3 절의 그래프 유사도를 포함한 최종적인 유사도 척도는 다음과 같다.

$$Sim_{total}(e_1,e_2)=\sum_{p_i \in P_c}\phi(\omega_{p_i})*Sim_{partial}(e_1,e_2,p_i)+Sim_{graph}(e_1,e_2)$$

$\phi(\omega)$: 가중치를 수정하기 위한 적응 함수

본 논문에서는 여러 가지 가중치 변경 알고리즘 중에서 가중치의 총합이 1이 되어야 하는 점을 염두에 두어 다음과 같은 알고리즘을 사용하였다.

(1) 25개의 OWL 속성에 대해 미리 정해진 초기 가중치를 할당한다. 25개의 가중치의 총합은 1이 되어야 한다. 이 가중치들은 처음에 속성이 의미하는 바를 감안하여 수동적으로 각각 다른 값으로 할당한다. 실제로 rdf:id로부터 얻어진 정보는 owl:versionInfo로부터 얻어진 정보보다 더욱 중요하다. 그래서 더 많은 초기 가중치를 rdf:id에 할당하게 된다.

(2) 각각의 속성 pi 에 대해 엔티티 간의 부분 유사도를 결정할 때, 두 엔티티에 동시에 표현되지 않는 속성 pi 의 가중치를 0으로 변경한 후 자동적으로 0이 아닌 다른 모든 가중치에 대해 ωi / [0이 아닌 가중치의 수] 만큼 더해준다. 이는 가중치의 총합이 1이 됨을 보장한다.

4. 실험

실험은 Jena API와 J2SDK 1.5를 사용하여 수행되었으며, Jena API는 OWL DL 온톨로지를 메모리에 로드하기 위해 사용되었다. 실험에 사용된 온톨로지는 동물에 대한 온톨로지이며 이 온톨로지와 비교하기 위해 원본 온톨로지(Ontology A)에 약간의 수정을 가한 온톨로지(Ontology B)를 사용하였다.

4.1 서로 다른 두 온톨로지 에서의 개체 유사도 측정 결과

전체 수행시간은 1.24초 이고 세 개의 엔티티에 대해 1.0의 값을 얻었다. 가장 좋지 않은 값은 TwoLeggedThing - BipedalThing 에 대한 결과로 0.0의 값을 얻었다.

표 2. 온톨로지 A,B 사이의 유사도 값(상위 5 개, 하위 5개의 값)

순서	온톨로지 A의 엔티티	온톨로지 B의 엔티티	유사도 값
1	Shoesize	shoesize	1.0
2	Shirtsize	Shirtsize	1.0
3	Male	Male	1.0
4	Woman	Woman	0.81
5	Animal	Animal	0.78
...	...	...	...
45	hasFemaleParent	hasWife	0.037
46	hasMaleParent	hasFather	0.034
47	hasHusband	hasSpouse	0.021
48	hasParent	hasWife	0.015
49	TwoLeggedThing	BipedalThing	0.0

4.2 가변 가중치를 사용하지 않았을 경우의 측정 결과

이번 절에서는 본 논문에서 제시한 가변 가중치의 개념이 아닌 실제로 존재하는 predicate에 대해서만 같은 가중치 값을 할당하여 실험하였다. 3.3 절에서 예로 든 언어를 가정하자. 엔티티의 속성의 가중치 값은 미리 정해져 있지 않고, 두 엔티티가 공유하는 속성이 오직 lang:id 만 존재한다면, 이 속성의 가중치는 1이 된다. 즉 각각의 엔티티에 대한 유사도를 측정할 때마다 속성에 대한 가중치 값은 다르게 정해진다. 위의 예에서 lang:id와 lang:label의 속성을 공유한다면, lang:id와 lang:label의 가중치 값은 각각 0.5가 된다.

각각의 속성의 가중치 값은 다음과 같다.

$$\text{가중치 값} = \frac{1}{\text{엔티티간 공유하는 속성의 갯수}}$$

위와 같은 전제 하에서 실시한 실험의 결과는 표 3 과 같다.

표 3. 가변 가중치를 사용하지 않았을 경우의 유사도 값

순서	온톨로지 A의 엔티티	온톨로지 B의 엔티티	유사도 값
1	Shoesize	shoesize	1
2	Shirtsize	Shirtsize	1
3	Male	Male	1
4	hasFather	Woman	0.77
5	hasMother	hasParent	0.74
...	...	...	...
45	hasFemaleParent	hasWife	0.031
46	hasMaleParent	hasHusband	0.022
47	hasHusband	hasSpouse	0.021
48	biologicalMotherOf	hasMother	0.016
49	TwoLeggedThing	BipedalThing	0.0

5. 결론 및 향후 연구 과제

본 논문에서는 서로 다른 온톨로지 간의 유사도를 측정하는 방법에 대해 연구하였다. 기존의 연구방법들이 구조적인 측면, 용어적인 측면에서 접근한 것과 달리 W3C에서 권고안으로 제안된 웹 온톨로지 언어인 OWL를 기반으로 한 온톨로지 언어의 속성을 바탕으로 유사도를 구하므로 OWL을 사용하여 작성된 온톨로지 간의 유사도 검색에 즉시 적용될 수 있는 장점을 지닌다. 또한 전역 유사도 척도를 계산할 때 가변 가중치를 사용함으로써 속성에 의미를 부여하면서 정규화 된 유사도 값을 얻을 수 있었다.

추후 연구에서는 원본 온톨로지에서 변경된 온톨로지가 아닌 실제로 다르게 작성된 온톨로지를 가지고 테스트를 하여 속성들이 유사도에 미치는 영향을 더욱 자세하게 분석해야 할 것이며 object 간의 비교를 simple matching만을 사용하였기 때문에 rdfs:label, rdfs:comment와 같은 문자열에 대한 비교가 제대로 이루어지지 않았다. 여기에 좀 더 다양한 비교 방법을 적용한다면 좀 더 좋은 결과를 얻을 수 있을 것이다.

감사의 글

이 연구는 산업자원부 지원으로 수행하는 21세기 프론티어 연구개발사업(인간기능 생활지원 지능로봇 기술개발사업)의 일환으로 수행되었습니다.

참고문헌

[1]Berners-Lee, T., Hendler, J. and Lassila, O., The Semantic Web, Scientific American, 2001

[2]Gruber, T.R, A Translation Approach to Portable Ontologies, Journal on Knowledge Acquisition, Vol. 5(2), 199-220, 1993

[3]Dave Beckett, Connecting XML, RDF and Web, Technologies for Representing Knowledge on the Semantic Web, XML Europe 2002, Barcelona, 2002

[4] 박사준, 시맨틱 웹에서 온톨로지를 이용한 전문가 지식 추출 모델, 중앙대학교 대학원, 제 74 회, 박사학위 논문, pp.11-12, 2003