

온톨로지 모듈화 평가 프레임워크

오선주

서울대학교 컴퓨터 공학과
(ohsunju7@snu.ac.kr)

.....

최근 온톨로지 모듈을 이용하여 효과적으로 온톨로지를 구축하려는 시도들이 증가하고 있으며 이에 따라 다양한 온톨로지 모듈화 기법들이 제안되어왔다. 이들 기법들은 각각의 모듈화 목표와 기준을 바탕으로 개발되어 왔으나 모듈화 기법들을 평가할 수 있는 공통된 기준 및 평가 프레임워크가 정립되지 않아 서로 비교하기 어려운 실정이다. 또한 온톨로지와 관련된 평가에 대한 연구들이 지금까지 활발히 이루어져 왔지만 주로 온톨로지 단위의 평가가 중심이 되어왔으며 온톨로지 모듈화 기법의 평가에 대한 가시적인 연구들이 부족하다. 그러나 획일적이고 대용량인 온톨로지 사용의 문제점과 모듈화 필요성에 대한 인식과 함께 온톨로지를 모듈화하는 작업에 대한 평가 필요성이 제기되고 있다. 그러므로 본 연구에서는 온톨로지 모듈화 작업 과정과 모듈화 결과물인 모듈 그리고 모듈화 도구에 대한 총괄적인 평가를 제공하는 프레임워크를 제안한다. 또한, 본 연구에서 제시한 평가 프레임워크를 실증적 실험을 통해 SWOOP, PATO, Prompt 등 모듈화 기법에 적용하였다. 본 연구의 결과를 통하여 온톨로지 엔지니어들이 온톨로지 모듈화 작업의 품질을 개선시킬 수 있고 온톨로지 사용자들이 사용 목적에 맞는 온톨로지 모듈화 기법을 선택할 수 있도록 한다.

.....

논문접수일 : 2010년 02월 03일 논문수정일 : 2010년 02월 14일 게재확정일 : 2010년 02월 23일 교신저자 : 오선주

1. 서론

온톨로지는 시맨틱 웹의 지식표현의 핵심 기술로서 다양한 분야에서 관련 연구들이 진행되어 왔다. 또한 좀 더 강력한 표현력과 처리 성능을 위해 온톨로지의 크기가 점점 커지고 복잡해지는 추세이다. 그러나 획일적이고 크기가 대용량인 단일 온톨로지를 사용하는 데서 발생하는 문제점들이 이미 지적되어 왔다(d'Aquin et al., 2007). 그 중 대표적인 문제점은 추론의 비효율성과 온톨로지를 시각적으로 보여주는 도구들에서 작업이 어렵다는 점이다. 또한 온톨로지를 사용하는 응용 시스템에서 필요로 하는 것은 대용량 온톨로지가 아닌

온톨로지의 특정 부분이므로 융통성있게 사용할 수 있는 적은 규모의 온톨로지 모듈의 사용이 응용 시스템의 효율을 높일 수 있을 것이다. 그러므로 유지 보수와 사용의 편리성을 위해 온톨로지를 모듈화하여 사용하는 것이 필요하다.

현재까지 온톨로지를 모듈화하기 위한 다양한 기법들이 연구되어 왔으며 온톨로지 모듈화 기법의 평가에 대한 필요성이 이미 언급되어 온 바 있다(D'Aquin et al., 2006; Cuenca Grau, 2005; Noy et al., 2004; Seidenberg and Rector, 2006). 모듈화 기준에 대한 연구는 일부 진행되어 왔으나 모듈화 작업의 결과물인 온톨로지 모듈이 잘 생성되었는가에 대한 구체적이고 정형화된 평가 지표와 모듈

화 과정의 논리적 완전성에 대한 연구는 미흡한 실정이다(D'Aquin et al., 2006).

본 연구에서는 온톨로지 모듈화 기법을 평가하기 위한 평가 기준 및 평가 척도를 제안한다. 즉, 다양한 온톨로지 모듈화 방법 중 어떤 온톨로지 모듈화 기법이 사용에 적합한지 결정하기 위해 평가 지표와 구체적인 척도를 개발할 필요가 있다. 이를 위해 온톨로지 모듈화 도구와 모듈화 과정 그리고 모듈화 결과물인 모듈에 대한 평가 지표들로 구성되는 총괄적인 프레임워크를 제안한다. 또한 제안하는 온톨로지 평가 프레임워크를 현재까지 연구 발표된 대표적인 온톨로지 모듈화 방법인 SWOOP(Cuenca Grau, 2005), PATO(Stuckenschmidt et al., 2004), Prompt(Noy et al., 2000)에 적용해 봄으로써 실증적으로 타당성을 검증하고 온톨로지 모듈화 기법의 선택에 대한 가이드라인을 제시하고자 한다.

본 연구의 구성은 다음과 같다. 다음의 제 2장에서는 온톨로지의 모듈화에 관련된 선행 연구들을 살펴보았으며 제 3장에서는 온톨로지 모듈화 기법의 평가를 위한 프레임워크를 제시하며, 제 4장에서는 본 연구에서 제안하는 온톨로지 모듈화 평가 프레임워크를 적용하는 실험을 수행하고 온톨로지 모듈화 기법들을 서로 비교하여 보았다. 제 5장에서는 연구 성과에 대해 토의하며 향후 연구 방향에 대해 논의한다.

2. 관련 연구

2.1 모듈 평가에 대한 연구

2.1.1 소프트웨어 모듈 관련 연구

Page-Jones(1980)는 분할과 정복(Divide and conquer) 원리를 이용하여 소프트웨어가 적당한 크기

로 분할되고 독립적 기능을 가지는 것이 소프트웨어 재사용성(reusability), 이해도(understandability), 유지보수 용이성(maintainability), 확장성(modifiability)을 위해 필요하다고 하였다. 한편, Constantine과 Yourdon(1979)은 소프트웨어 모듈화는 결합도와 응집도에 의해 측정될 수 있다고 하였다. 이와 같이 소프트웨어 모듈화는 소프트웨어의 유지보수와 생산성을 높일 수 있는 수단으로 인식되어왔다.

또한 소프트웨어 모듈의 품질을 측정하는 메트릭에 대해서 활발히 연구가 이루어졌다. 그 중 Chidamber Kemerer(1994)의 CBO(Coupling Object Classes)와 RFC(Response For a Class)가 널리 사용되어왔다. CBO는 한 클래스와 결합되어 있는 다른 클래스들의 수로 정의된다. 두 클래스가 결합되어 있다는 것은 한 클래스 안에 선언된 메소드가 다른 클래스에 정의된 메소드 또는 인스턴스 변수를 사용하는 경우를 말한다. 반면 RFC는 한 클래스에서 다른 클래스의 메소드를 호출하는 수를 계산한다. CBO나 RFC가 높으면 결합도가 높아 모듈의 재사용을 어렵게 만든다.

2.1.2 온톨로지 모듈 평가에 대한 연구

온톨로지 평가에 대한 연구는 대부분 온톨로지 단위의 평가에 집중되어 왔으며(Brewstar et al., 2004; Burton-Jones et al., 2004; Gomez-Perez, 2001) 온톨로지의 모듈화 기법에 관한 공통의 평가기준과 평가 방법에 대한 연구는 아직까지 활발하지 않으며 잘 정립되지 못하였다. d'Aquin(2009)은 온톨로지 모듈화 기법에 대한 평가 기준으로 논리적 기준, 구조적 기준, 모듈의 질적 기준, 모듈 간 관계, 응용 기준 등을 제시하였다. 그 중 논리적 기준은 원본 온톨로지의 공리가 추가되거나 손실되지 않고 그대로 온톨로지 모듈화에 보존되는지 검사하는 기준이며 구조적 기준은 온톨로지 모듈

의 크기와 모듈 내 특정 엔터티간의 거리를 측정하는 지표이다. 또한 질적 기준은 응집도(Cohesion)와 풍족도(Richness) 그리고 도메인 커버리지(Domain Coverage)를 측정하며 모듈간 관계 기준으로는 모듈간 거리와 링크 수를 반영하였다. 응용기준으로는 온톨로지에 대한 가정, 사용자 상호 작용 정도, 모듈의 유용성, 모듈화 성능 등을 들었다. d'Aquin(2009)의 연구는 모듈화 기법을 평가하는 유용한 가이드라인을 제시하였으나 정량적이고 구체적인 평가방법을 제공하지 못하였을 뿐 아니라 이론적으로 검증되지 않은 측면이 있다. 그러나 평가 방법이 실제 적용되기 위해서는 이론적 실증적으로 검증된 정형화된 평가 지표가 필요하다.

2.2 온톨로지 모듈화 기법에 대한 연구

온톨로지 모듈화에 대한 연구는 모듈화에 적용되는 기법의 공통적 특성을 기준으로 다음과 같이 분할법과 추출법으로 분류되고 있다.

2.2.1 분할법

분할법은 온톨로지의 내부 구조에 따라 다수의 모듈로 분할하고 분할되는 모듈 내에 클래스와 이와 관련된 공리들을 포함시켜 모듈화한다. 사용자의 모듈화 의도보다 온톨로지의 내부적 구조와 공리의 보존을 중요시한다.

Stuckenschmidt and Klein (2004)는 독립형 온톨로지 분할 툴인 PATO를 사용하여 온톨로지를 분할하였다. 이 방법은 본래 대용량 온톨로지의 관리와 유지보수를 위해 온톨로지의 일부를 검토하기 위해 만들어졌다. 또한 개념간 종속성이 온톨로지의 구조에 의해서 결정된다고 가정하고 온톨로지를 개념간 종속성 그래프(Dependency Graph)로 변환한 뒤, 그래프상의 링크에 중요도를 배정하

여 높은 중요도를 가진 부분부터 먼저 모듈에 포함시킨다. 그러므로 모듈화의 기본 사상인 결합도를 모듈 분할의 기본 개념으로 한다. 이 방법은 온톨로지를 사용하는 응용 프로그램의 목적에 따라 링크 중요도(weight), 링크의 깊이(depth) 등의 파라미터를 튜닝할 수 있다. 예를 들면, 종속성 그래프에 반영할 온톨로지 내 클래스간의 링크의 종류를 선택할 수 있고 모듈 크기를 제한할 수 있다. 또한 응용 프로그램에 따라 링크의 가중치를 서로 다르게 배정할 수 있다. 응용 프로그램의 특수성을 반영할 수 있는 반면, 이 방법은 온톨로지를 구성하는 개념과 공리에 대한 의미적 고려 보다는 온톨로지의 구조적인 측면을 주로 고려하여 모듈로 분할한다.

Cuenca Grau(2005, 2007)의 모듈화 방식은 관심 대상 개념과 그와 관련된 공리로부터 추론을 통해 지역적으로 완전한(locally complete) 최소한의 노드와 속성을 구한 뒤 모듈화한다. 이 방식은 서로 관련된 클래스와 공리가 새로 만들어지는 모듈에 완전하게 보존된다는 장점은 있으나 사용자의 의도와는 동떨어진 모듈이 생성될 수 있다. 즉, 논리의 보존을 위해서 본래 사용자가 의도하지 않았던 개념과 공리가 함께 모듈에 포함될 수 있기 때문이다. 분할된 각각의 모듈은 논리적으로 완전하며 각기 독립적으로 사용될 수 있다. SWOOP 온톨로지 편집 도구에 모듈 분할 기능이 포함되어 있으며 모듈을 독립된 파일로 분리해 낼 수 있다. 각 모듈이 논리적 추론의 완전성(completeness of local reasoning)을 추구함으로써 분할한 모듈이 다른 모듈들과 독립적으로 사용될 수 있도록 한다. 이 방법은 모듈의 크기에 대한 고려 보다 모듈 분할 뒤에 공리가 완전하게 보존되는 것을 모듈화의 중요한 기준으로 한다.

2.2.2 추출법

추출법은 전체 온톨로지 중 일부를 모듈로 분리해 내는 방식이다. 추출법은 사용자가 지정한 특정 시작점에서 출발하여 온톨로지를 순회하며 관련 노드와 링크를 추출한다. 추출법의 장점은 사용자의 의도에 맞게 순회할 링크의 종류와 거리를 미리 지정하고 순회 시 관련된 링크를 모듈에 포함할지 여부를 조절할 수 있다. 즉, 원본 온톨로지가 보유한 클래스, 공리 등 정보를 완전히 보존하기보다 사용자의 의도에 맞게 간략하고 생략된 형태로 온톨로지의 일부를 모듈로 추출할 수 있다. 그러나 간략화되는 추출 과정에서 온톨로지에 포함된 의미 있는 정보를 잃을 수 있다. 추출법의 대표적인 시도로 다음과 같은 연구들이 있다.

Noy et al.(2004)의 방식은 사용자와 상호 작용을 통해 탐색의 시작 노드와 관련 속성(property), 탐색의 깊이(depth)를 사용자로부터 입력 받는다. 탐색 시작 노드에서 시작하여 지정된 탐색의 깊이에 도달할 때까지 클래스의 속성을 따라 반복적으로 계층 구조를 순회한다. 순회가 진행되는 과정에 사용자의 의도에 따라 특정 속성을 배제시킬 수 있다. 일반적으로 분할법에 속하는 모듈화 방식들은 완전 자동화 특성이 강한 반면, 추출법은 사용자와 상호 작용이 필요하여 순회의 대상이 되는 관계의 종류와 관계의 반복(recursion) 수준을 모듈화 과정 중에 상호 작용을 통해 지정할 수 있다. 상향 혹은 하향 순회를 지정할 수 있으며 뷰(view) 형식으로 모듈을 표현한다. 즉, 모듈이 물리적인 화일로 분리되지 않고 모듈화를 통해 추출된 뷰가 생성된다. 프토티지 온톨로지 에디터의 플러그인인 프롬프트(Prompt)를 통해 모듈화가 지원된다.

Seidenberg and Rector(2006)는 모듈화를 세그먼트이션(segmentation)이란 용어로 표현한다. Galen 의학 온톨로지에 적용할 목적으로 연구되었지만

일반 온톨로지에 적용이 가능하도록 일반화되어 있다. 이 방법은 하나 혹은 그 이상의 클래스로부터 순회를 시작하며 상향 순회와 하향 순회를 모두 적용할 수 있으며 특히 시작 노드의 모든 슈퍼 클래스들을 모듈에 포함한다. 순회 시 특정 속성을 필터링하거나 순회 깊이(boundary depth)를 제한하여 모듈의 크기를 줄일 수 있다.

D'Aquin et al.(2006)은 독립형 모듈화 도구인 KMi(D'Aquin et al., 2006)를 이용하여 사용자로부터 하부 어휘(Sub-vocabulary)나 관계를 입력으로 받아 온톨로지의 계층 구조를 반복적으로 순회하면서 관련된 요소를 추출하여 모듈화한다. 하부 어휘는 클래스, 속성, 그리고 개체가 될 수 있다. 완전 자동으로 모듈화가 진행되며 다양한 레벨의 온톨로지, 즉, 단순한 형태에서 복잡하고 풍부한 온톨로지까지 다룰 수 있도록 설계되었다. 또한, 완전 자동화 모듈 추출을 통해 사용자와의 상호 작용을 최소화하였다.

모듈화 과정에 추론 기능을 사용할 수 있으며 다른 기법들과 달리 순회 시작 노드의 모든 상위 노드가 모듈에 포함되지 않는 특징이 있다. 즉, 가장 근접한 상위 노드(Least Common Subsumer)만을 모듈에 포함한다. 또한 이 방법은 온톨로지의 계층구조를 원형 그대로 모듈로 이전하지 않고 계층 구조상의 중간 노드들을 제거하여 보다 단축된 형태로 계층구조를 변형하기도 한다.

Doran et al.(2007)은 온톨로지 모듈화 도구인 ModTool(Doran et al., 2007)을 설계하고 모듈화에 사용하였다. 사용자가 지정한 시작 노드에서부터 순회를 시작하여 관련된 관계와 속성을 추상화 그래프(abstract graph)로 변환하고 변환된 그래프로부터 모듈을 추출한다. 그러므로 온톨로지 구축 시에 사용한 온톨로지 언어의 종류에 관계없이 모듈화를 적용할 수 있다. 또한 특정 관계를 순회

의 대상에서 배제시키는 등 다양한 모듈화 방식을 지원하기도 한다. 온톨로지 모듈화의 목적은 모듈의 재사용이며, 사용자의 모듈화 의도에 따라 커스터마이징될 수 있도록 설계되었다.

3. 온톨로지 모듈화 평가 프레임워크

온톨로지 모듈화에 대한 필요성은 이미 널리 인지되어 모듈화 알고리즘에 대한 많은 연구들이 진행되어 왔다(Noy, 2004). 그러나 온톨로지 모듈화에 관한 많은 연구들이 각기 다른 목적과 상황에서 모듈화를 적용하여 왔다. 즉, 모듈화 범칙 및 기준, 모듈화에 대한 가정, 모듈화 과정에 수반되는 사용자와의 상호 작용 정도, 추론지원, 모듈의 사용 용도 등이 각각의 연구에서 서로 상이하게 적용되어 왔다. 따라서 모듈에 대한 정의도 상이하고 모듈이 잘 만들어졌는지 평가하는 방법에 대한 통일된 평가 기준과 모듈화 평가 프레임워크가 없어서 모듈화 기법을 서로 비교하기 힘든 실정이다(d'Aquin et al., 2007).

본 절에서는 모듈이 갖추어야 할 일반적 특성을 바탕으로 온톨로지 모듈을 평가할 수 있는 평가 지표를 정의하고 온톨로지 모듈화 과정 및 도구를 평가할 수 있는 평가 지표를 제시하고자 한다. 이를 위해 객체 지향과 컴포넌트 기반 연구 분야에서 주로 연구되어온 소프트웨어 모듈의 정의와 소프트웨어 모듈의 품질을 측정하는 방법에 대한 기존 연구들을 참조하고, 이를 온톨로지 연구 분야에 적용하였다. 또한 현재까지 이루어진 온톨로지 모듈화에 대한 연구를 바탕으로 온톨로지 모듈화 과정의 일치성과 도구의 기능을 평가할 수 있는 평가 프레임워크를 제안한다.

3.1 모듈화 평가 프레임워크

모듈화 평가는 모듈화의 결과물인 모듈에 대한 평가뿐만 아니라 모듈화 과정 중에 클래스와 공리의 보존에 대한 평가, 또한 모듈화 과정을 지원하는 도구의 기능에 대한 평가가 모두 종합되어야 한다. 그러므로 본 연구에서는 온톨로지 모듈화 기법의 평가 지표들을 다음의 <그림 1>에서 보는 바와 같이 세 단계로 구분하였다. 첫 번째는 모듈화 기법을 적용하여 생성된 모듈의 품질을 판단하는 모듈 평가지표로서 온톨로지 모듈 크기, 모듈 응집도 그리고 모듈 결합도로 구성된다. 모듈의 평가 지표는 모듈로서 기본적으로 갖추어야 할 응집도와 결합도 기준을 만족하는가를 판단하는데 도움을 줄 수 있다. 두 번째 관점은 모듈화 과정에 대한 평가지표이다. 이것은 온톨로지의 중요한 정보인 클래스와 관련된 공리들이 모듈화 과정 중에 손실되지 않고 잘 보존되었는지 측정한다. 마지막은 모듈화를 지원하는 도구의 평가 지표이다. 모듈화 도구 평가지표는 모듈화 도구들의 모듈화를 지원하는 다양한 기능에 대해 평가한다. 이 세 단계의 평가 항목들이 모두 반영될 수 있도록 모듈화 평가 프레임워크를 설계하였다.

3 단계 모듈화 도구 평가	모듈화 방법, 자동화 지원, 추론지원, 지원언어, 완전성 지원
2 단계 모듈화 과정 평가	논리적 일치성
1 단계 모듈 평가	모듈크기, 응집도, 결합도

<그림 1> 모듈화 평가 프레임워크

3.1.1 모듈 평가 지표

온톨로지 평가에 대한 기존의 연구들에서 온톨로지 수준의 평가 지표들이 개발되었으나 온톨로지로서의 일반적 특성 이외에 모듈로서 갖추어야 할 특성에 대해 다루어지지 못했기 때문에 본 연구에서는 온톨로지 모듈로서 갖추어야 할 특성을 중심으로 모듈 평가 지표를 제안한다.

2.1.1 소프트웨어 모듈 관련 연구에서 살펴 본대로 소프트웨어 모듈의 측정을 위해서 소프트웨어 결합도와 응집도가 이용되어 왔다(Constantine and Yourdon, 1979). 응집도는 하나의 단위 안에 있는 요소 간의 관계의 조밀도를 측정하는 척도이며 결합도는 단위들 간의 의존도를 측정한다. 온톨로지는 소프트웨어 프로시저에 비해 데이터위주로 구성되지만 공리화(Axiomatization)부분은 소프트웨어의 코드 부분과 유사한 측면이 있으므로(Loebe F., 2006) 온톨로지 모듈 요건으로 소프트웨어 모듈의 요건인 모듈크기, 응집도, 결합도를 도입하여 적용한다.

(1) 온톨로지 모듈 크기

소프트웨어 모듈과 유사하게 온톨로지 모듈이 여러 사용자들에 의해 쉽게 공유될 수 있고 재사용되기 위해서는 모듈이 재사용에 적합한 크기여야 하고, 재사용을 위해 수정이 필요한 경우에 수정을 최소화하고 용이하게 할 수 있도록 하여야 한다. 그러므로 온톨로지 모듈의 크기는 모듈의 재사용과 유지 보수를 위해 중요하다. 모듈 크기가 작으면 사용 효율성이 떨어지며 모듈이 크면 관리하기 어렵고 사용자 측면에서 직접적으로 필요하지 않은 부분도 포괄적으로 포함하여야 하므로 비효율이 발생한다. 본 연구에서의 모듈의 크기는 모듈이 포함하고 있는 클래스 수를 기준으로 한다.

본 연구에서 모듈의 크기를 측정하기 위한 평가 척도인 SoM(Size of Module)을 다음과 같이 정의한다.

$SoM(M) = |M|$, 이때 $|$ 는 카디널리티, M 은 모듈을 의미한다.

위 정의는 본 연구 내에서 동일한 의미로 적용된다.

(2) 온톨로지 모듈 결합도

온톨로지 모듈 결합도는 온톨로지 모듈이 다른 모듈과 가지는 관련성을 나타낸다. 다른 모듈과 연관성이 많을 경우 높은 결합도를 가진다고 정의한다. 온톨로지 모듈이 재사용되기 위해서는 모듈이 외부 참조를 최소화하고 모듈내의 클래스들은 모듈내의 자원을 이용해서 정의되는 것이 바람직하다. 따라서, 임의의 모듈 내부의 변경이 다른 모듈에 영향을 최소화하기 위해 모듈 결합도를 낮추어야 한다. 본 연구에서는 모듈 결합도를 측정하기 위한 평가척도 NoSR(Number of Separated Relation)을 다음과 같이 정의한다.

$$NoSR(M) = \sum_i \sum_j nosr(c_i, c_j)$$

$$nosr(c_i, c_j)$$

$$= \begin{cases} 1 & \text{클래스 } c_i \text{와 } c_j \text{ 사이에 임의의 관계가 있는 경우} \\ 0 & \text{그 외의 경우} \end{cases}$$

이때 $c_i \in M, c_j \in \{O-M\}$, O 는 온톨로지임.

(3) 온톨로지 모듈 응집도

온톨로지 모듈 응집도는 하나의 온톨로지 모듈을 구성하는 클래스들간의 관련 정도를 나타낸다. 클래스간의 관련성은 클래스가 개념적 혹은 의미적으로 모듈내의 다른 클래스들과 연관된 정도를

나타낸다. 모듈 내 클래스간 연관성이 많을수록 응집도가 높다고 정의한다.

온톨로지 모듈이 응집도가 높기 위해서는 서로 관련되는 개념들과 관계들을 한 모듈 내에 포함시키는 것이 바람직하다. 즉, 온톨로지 모듈을 구성하는 클래스 간에 조밀한 관련성이 있어야 한다. 모듈 응집도는 모듈 내에서 클래스가 다른 클래스와 공유하는 임의의 관계의 수를 기준으로 한다. 본 연구에서는 모듈 응집도를 측정하기 위한 평가 척도 NR(Number of Relation)을 다음과 같이 정의한다.

$$NR(M) = \sum_i \sum_j \frac{nr(c_i, c_j)}{|M|}$$

$$nr(c_i, c_j) = \begin{cases} 1 & \text{클래스 } c_i \text{와 } c_j \text{ 사이에 임의의 관계가 있는 경우} \\ 0 & \text{그 외의 경우} \end{cases}$$

이때 $c_i \in M, c_j \in M$ 이다.

3.1.2 모듈화 과정 평가 지표

모듈화 과정 평가 지표는 모듈화 이전의 온톨로지와 모듈화 이후에 생성된 모듈들을 비교함으로써 모듈화 과정의 논리적 일치성을 평가하는 지표이다. 온톨로지는 모듈화되기 이전과 이후에 논리적으로 일치(Logical Consistency)하여야 하는데 이를 위해 Cuenca Grau(2005)가 제안한 지역적 정확성(Local Correctness)과 지역적 완전성(Local Completeness) 개념을 반영하여 논리적 일치성을 평가한다.

지역적 정확성이란 모듈에 정의된 클래스와 공리들은 반드시 원본 온톨로지에 포함되어 있어야 하며 원본에 없는 새로운 클래스와 공리들이 모듈에 첨가되어서는 안 되는 것을 의미한다. 지역적

완전성이란 원본 온톨로지에서의 클래스와 내재적인 공리 정보가 온톨로지 모듈에 그대로 보존되어야 하며 기존에 보유하고 있던 클래스와 공리가 모듈화에 따라 손실되어서는 안 되는 것을 의미한다. 지역적 완전성은 지역적 정확성의 다른 시각에서의 해석이라 할 수 있는데 많은 온톨로지 모듈화 관련 연구들에서 가장 중요한 모듈 평가 기준으로 인식되어 왔다(d'Aquin et al., 2009). 일부 연구에서는 보존적 확장(Conservative extension)이라는 용어로 인식되어 왔다(Cuenca Grau, 2007).

본 연구에서는 지역적 완전성과 정확성을 측정하기 위한 구체적인 모듈화 평가지표를 다음과 같이 정의하였다. 지역적 완전성은 모듈화 이전의 클래스 수와 공리의 수가 모듈화 이후의 각 모듈의 클래스 수의 합계와 공리 수의 합계와 일치하여야 한다. 그러므로 온톨로지 모듈의 클래스들이 보유한 관계 수와 원본 온톨로지 상에 동일한 클래스들이 보유한 관계 수간의 차이로 모듈화 과정의 논리적 일치성을 평가한다. 본 연구에서는 모듈화 과정의 일치성을 측정하기 위한 평가척도 DR(Difference of Relation)을 다음과 같이 정의한다.

$$DR(O) = \sum_k \frac{NR(O) - NR(O \cap M_k) + NR(M_k - (O \cap M_k))}{NR(O)}$$

M_k 는 온톨로지 O 를 모듈화한 온톨로지 모듈.

3.1.3 모듈화 도구 평가지표

모듈화 도구 평가지표는 온톨로지 모듈화를 지원하는 도구들을 평가하는 지표이다. 모듈 및 모듈화 과정을 지원하는 특성은 모듈화를 위해 사용되는 도구의 특성에 따라 영향을 받는다. 그러므로 도구의 특성을 파악하고 적합한 도구를 선

택하는 것은 중요하다. 모듈화 도구 평가 항목으로 도구에서 제공하는 모듈화 방법의 다양성과 모듈화 과정의 자동화, 논리적 완전성 지원 그리고 지원하는 언어, 추론 기능 지원 등으로 평가지표를 제안한다.

- 모듈화 방법의 다양성은 모듈화 과정에서 다양한 모듈화 방법을 선택하여 적용할 수 있는 정도를 나타낸다. 즉, 해당 모듈화 기법이 분할법과 추출법 모두를 지원하는 지 또는 모듈화 시 특정 연관 관계를 필터링하거나 모듈화를 진행하기 위해 순회의 깊이를 조절할 수 있는지? 모듈화 결과인 모듈이 파일 형태로 생성되는지?와 같이 모듈화 방법의 다양성을 지원하는 정도를 나타낸다.
- 자동화 지원은 모듈화 과정의 자동화 지원 정도를 나타낸다. 즉, 모듈화 과정이 완전 자동화되는지 아니면 사람과의 상호 작용을 통하여 모듈화 작업이 이루어지는 지를 나타낸다.
- 지원 언어는 모듈화 과정에서 다양한 온톨로지 언어를 지원하는 정도가 평가 기준이 된다. 다양한 온톨로지 언어를 지원하는 도구는 사용자에게 다양한 온톨로지에 대한 모듈 평가와 이용 기회를 제공한다.
- 추론 기능 지원 여부는 도구에서 모듈화 시 추론을 고려하는지 여부를 나타낸다.
- 논리적 완전성은 원본 온톨로지의 클래스와 공리 등이 모듈화 이후에도 생성된 모듈에 내용이 보존되는 정도를 측정한다.

4. 평가

본 장에서는 새로이 제안한 모듈화 평가 프레임

워크의 타당성을 검증하기 위해 모듈, 모듈화 과정, 모듈화 도구에 대한 평가 지표를 적용하였다. 실험 대상이 될 온톨로지 모듈을 얻기 위해 Swoogle 검색엔진을 이용하여 웹 상에서 “Student”, “University” 검색어로 검색하여 Koala.owl,¹⁾ Iswc.owl,²⁾ Univ-bench.owl³⁾ 등 세 개의 온톨로지를 얻었다. 이들 온톨로지에 모듈화 도구인 Prompt factor, PATO와 SWOOP 등 세 가지 도구를 적용하여 온톨로지를 모듈화하고 본 연구에서 제안한 평가지표들을 적용하여 다음과 같은 결과를 얻었다.

4.1 모듈평가

Iswc, Koala, Univ-bench 등의 세 개의 온톨로지에 모듈화 도구를 사용하여 모듈화 작업을 한 결과 <표 1>과 같이 총 30여 개의 모듈이 생성되었다.

SWOOP을 적용하였을 경우, Iswc 온톨로지는 4개의 모듈로 분할되었는데 총 46개의 클래스 중 19개가 한 모듈로 할당이 되었다. Koala 온톨로지는 다양한 크기의 4개의 모듈로 구분되었는데 전체 20개 노드 중 13개가 하나의 모듈로 집중적으로 할당되었고 Univ-bench도 43개 노드 중 21개까한 모듈로 할당되었다.

SWOOP을 사용하여 분할된 모듈을 크기측면에서 비교할 때, 세 개의 온톨로지 모두가 한 모듈이 다른 모듈에 비해 상대적으로 크기가 크고 내부에 많은 공리를 보유하여 응집도가 높게 나타났다. 나머지 모듈들은 크기가 상대적으로 적게 나타났다. 이와 같은 크기의 비대칭성과 응집도의 비대칭성은 원래의 온톨로지가 보유하고 있던 법칙 그리고

1) <http://protege.stanford.edu/plugins/owl/owl-library/koala.owl>.

2) <http://annotation.semanticweb.org/ontologies/iswc.owl>.

3) <http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl>.

<표 1> 모듈화에 의해 생성된 온톨로지 모듈

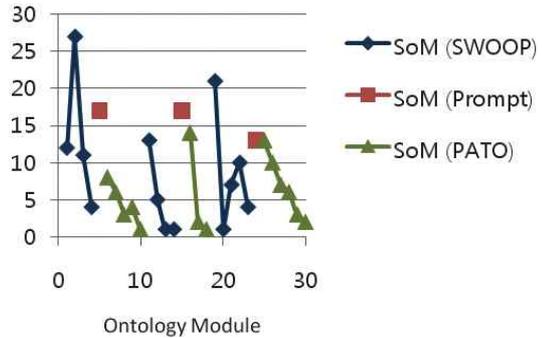
모듈 번호	온톨로지	사용 도구	모듈 번호	온톨로지	사용 도구
1	lswc	SWOOP	16	koala	PATO
2	lswc	SWOOP	17	Koala	PATO
3	lswc	SWOOP	18	Koala	PATO
4	lswc	SWOOP	19	Koala	PATO
5	lswc	Prompt	20	Univ bench	SWOOP
6	lswc	PATO	21	Univ bench	SWOOP
7	lswc	PATO	22	Univ bench	SWOOP
8	lswc	PATO	23	Univ bench	SWOOP
9	lswc	PATO	24	Univ bench	Prompt
10	lswc	PATO	25	Univ bench	PATO
11	koala	SWOOP	26	Univ bench	PATO
12	Koala	SWOOP	27	Univ bench	PATO
13	Koala	SWOOP	28	Univ bench	PATO
14	Koala	SWOOP	29	Univ bench	PATO
15	Koala	Prompt	30	Univ bench	PATO

추론 관련 정보 등을 가장 크기가 큰 주 모듈에 대부분 보존하기 때문인 것으로 추정된다(<그림 2> 참조).

한편, PATO를 사용했을 경우는 SWOOP을 적용하였을 때와 달리 모듈 크기의 차이가 비교적 적은 여러 개의 모듈로 분할되었다(<그림 2> 참조). 이와 같은 결과는 PATO에 모듈 크기 조정 옵션인 하위 클래스 연결 강도(strength of subclass link) 옵션과 분리(isolation) 옵션이 있어 모듈의 크기를 비교적 균일하게 제한하였기 때문인 것으로 추정된다. 본 실험에서는 하위 클래스 연결 강도 옵션과 분리 옵션을 각각 5.5와 55로 지정하였다.

Prompt는 추출법의 일종으로서 주어진 시작 노드에서 순회를 시작하여 서로 관련성이 있는 노드들과 공리들을 모아 하나의 모듈로 추출해내었다. 추출된 모듈은 관련 노드와 공리를 모두 포함하므로 모듈의 크기가 상대적으로 커지게 된다.

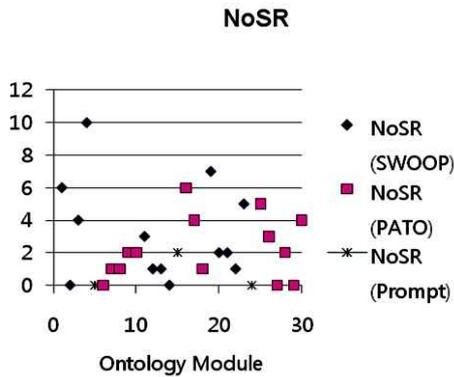
Module Size



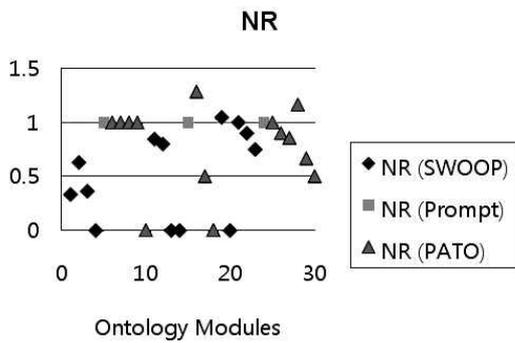
<그림 2> 모듈 크기

결합도 측면에서 SWOOP을 사용하였을 경우 노드와 노드간의 계층 관계의 대부분은 단절되지 않고 하나의 모듈에 포함되었으나 연관 관계들은 많은 경우 단절되었다. 모듈화로 생성된 상대적으로 큰 모듈이 다른 모듈들을 참조하는 참조 링크를 대부분 보유하고 있으므로 결합도가 다른 모듈에 비해 상대적으로 높게 나타났다. Prompt의 경우 전체 온톨로지에서 서로 관련성이 있는 노드들이 추출되어 모듈이 구성되는데 모듈에 포함된 노드들은 모듈로 추출되지 않고 남아있는 다른 노드들과의 관련성이 적고 이들 노드들에 대한 참조를 유지하지 않으므로 결합도가 낮게 나타났다. PATO에 의해 분할된 모듈은 모듈화 순회의 깊이를 지정하는 옵션값이 적은 경우, 많은 계층 관계와 비 계층 관계들이 단절되기도 하였다. 그러므로 PATO의 경우 일부 모듈에서는 결합도가 높고 일부는 낮은 다양한 분포를 나타내었다(<그림 3> 참조).

응집도 측면에서 세 가지 모듈화 방식에 의해 분할된 각각의 모듈이 각기 응집도가 다르게 구분되는 현상을 나타내었다. SWOOP은 비대칭적으로 모듈을 분할하는데 그 중 한 모듈이 상대적으로 많은 클래스와 공리들을 보유하고 있으며 다른 모



<그림 3> 결합도 지표



<그림 4> 응집도 지표

둘보다 높은 응집도를 나타내었다. 대부분의 나머지 모듈들은 적은 수의 클래스만 보유하는 등 극히 단순한 모듈로 구성되었다. Prompt를 적용한 경우, 하나의 모듈 안에 관련된 관계나 속성들을 대부분 보유하여 응집도가 다른 방법에 비하여 상대적으로 높았다. PATO의 경우도 SWOOP과 유사하게 온톨로지의 클래스와 관계, 속성들이 일부 모듈에 집중되어 응집도가 높아지는 현상을 보여주었다(<그림 4> 참조).

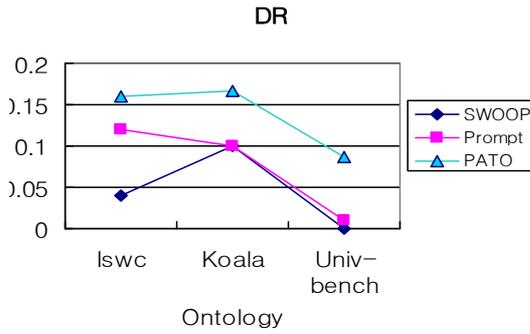
4.2 모듈화 과정 평가

모듈화 과정의 논리적 일치성은 지역적 정확성

과 지역적 완전성으로 평가하였다. 그런데 모듈의 노드와 공리가 모듈화 이전의 온톨로지로부터 반드시 근거해야 하는 지역적 정확성은 개념적으로 중요한 의미가 있으나 본 연구에서 살펴본 온톨로지 모듈화 기법들이 모듈화하는 과정에 새로운 노드나 공리를 추가하지 않고 기존의 온톨로지로부터 노드와 공리를 그대로 가져왔다. 그러므로 본 연구의 실험에서 온톨로지 모듈이 지역적 정확성을 만족함을 알 수 있었다. 한편, 지역적 완전성 요건은 모듈화된 이후에도 모듈화 이전에 보유하고 있던 공리와 제약사항(Constraint)이 보존되는지 평가하는 기준이다. 이와 같이 지역적 완전성과 지역적 정확성을 바탕으로 모듈화 정보의 논리적 일치성이 잘 준수되었는지 다음과 같이 평가하였다.

SWOOP을 사용하여 모듈화하였을 경우, 모듈화에 의해 클래스와 관련 공리가 모듈로 분할되고 서로 다른 모듈에 위치함에 따라 단절되는 관계 정보들이 외부 링크(foreign type)로 전환되었다. SWOOP의 경우 다른 도구를 사용하였을 때 보다 DR값이 작게 분포하는데 이는 Cuenca Grau (2005)의 보존 법칙(conservative extension)의 준수를 위해 관련성 있는 노드들과 공리들을 손실 없이 분할되는 모듈들에 포함시킨다는 것을 알 수 있었다. 특히 Univ-bench 온톨로지의 경우 모듈화 이전부터 온톨로지에 연관관계가 정의되어 있지 않아서 모듈화 과정에 손실되는 연관관계가 없고 모든 클래스와 계층 관계가 분할된 모듈에 보존되었다. 그러므로 모듈화 과정의 일치성이 잘 지켜졌다(<그림 5> 참조).

Prompt를 사용하였을 경우, 원본 온톨로지에 있던 일부 노드와 관련 링크들이 생성되는 모듈에 포함되지 않아 결과적으로 온톨로지의 완전성 측면에서 불완전한 결과를 나타내었다. DR값은 SWOOP과 PATO의 중간 분포를 나타내었다.



<그림 5> 논리적 일치성 지표

PATO를 적용한 경우, 세 온톨로지 모두에서 원본 온톨로지가 보유하고 있던 계층, 비 계층 관계들이 모듈 분리를 통해 단절되어 결과적으로 완전성 측면에서 불완전한 결과를 나타내었다.

4.3 도구 평가

모듈화 도구 평가는 <표 2>에서 보는 바와 같이 모듈화 방식 지원의 다양성, 모듈화 과정의 자

동화 척도, 지원 언어의 다양성, 추론 지원 여부, 그리고 모듈화 과정의 완전성 지원 척도로 측정되었다.

모듈화 방법의 다양성 측면에서 살펴보면 SWOOP은 모듈 분할이 완전 자동화하는 분할법과 시작 노드를 지정하여 온톨로지를 순회하여 모듈을 추출해내는 추출법을 모두 지원한다. PATO와 Prompt는 모듈을 화일로 분할하지 않고 분할 결과를 뷰로 보여주도록 되어 있다. Prompt의 경우 시작 노드를 지정하면 시작 노드와 관련된 노드와 공리를 추출하고 추가적으로 포함할 노드에 대하여 제안하도록 제시한다. 이때 사용자의 응답에 따라 추가적으로 노드를 추출하게 된다.

자동화 지원 측면에서 SWOOP과 PATO 모두 사용자와의 상호 작용 없이 완전 자동화하여 모듈을 추출할 수 있도록 지원한다. 또한 PATO의 경우 추가 옵션을 제공하여 사용자가 옵션을 지정할 수 있도록 하였다. 이와 같이 자동 추출 기능은 온톨로지의 전문적 지식이 없는 도메인 전문가가 편리하

<표 2> 모듈화 도구 평가

도구 기능	항목	SWOOP	Prompt	PATO
모듈화 방법의 다양성 지원	상호작용	상호 작용 없이 가능함	상호 작용을 수시로 할 수 있고 순회 시작점을 지정할 수 있음.	상호 작용 없이 가능하며 모듈화 선택 사항을 지정할 수 있음.
	결과형태	분할된 파일, 뷰	뷰	뷰
	모듈화 기법	분할법, 추출법	추출법	분할법
자동화 지원		완전 자동화	반 자동	반 자동
지원 언어		OWL, RDF, XML	RDF, OWL	RDF, OWL
추론 지원		지원됨	지원됨	지원안됨
논리적 완전성		엄격함	공리의 보존이 불완전함	공리의 보존이 불완전함
강점		논리적 완전성 추구	사용자 요구 사항에 따른 모듈화 옵션(관계, 순회 깊이 등) 조정 가능.	대 용량 온톨로지 관리용. 선택사항을 이용한 튜닝 지원.

게 모듈을 추출하기 위해 적합하다고 볼 수 있다. 한편, Prompt는 추출 과정 중에 사용자와 상호 작용을 통해 사용자의 요구에 적합한 온톨로지를 추출하도록 한다. 예를 들면 사용자가 지정한 클래스와 연결된 노드들과 관련 공리들을 추출한다.

지원언어는 SWOOP의 경우 OWL, RDF, XML을 지원하며, Prompt의 경우 RDF, OWL, PATO의 경우 RDF, OWL을 지원한다.

추론 기능은 SWOOP의 경우 Pallet 추론 엔진의 사용을 지원하며 Protégé Prompt의 경우 온톨로지 편집 환경에서 Pallet을 지원하고 자체적으로 추론 기능 지원을 위한 추론 엔진을 제공한다. PATO의 경우 노드들 간의 관계의 구조적 측면을 고려하여 모듈화하고 노드들 간의 관계의 의미나 추론을 고려하지 않는다.

논리적 완전성 지원 측면에서는 SWOOP의 모듈화 방법은 기본적으로 계층 관계와 공리 등이 단절되지 않고 유지된다. 지역적 완전성과 정확성을 기반으로 모듈화를 통한 논리의 보존을 우선시한다. 한편 Prompt의 경우, 온톨로지를 구성하는 클래스간 관계가 새롭게 추출되는 모듈에 포함되지 않는 경우도 있었다. 예를 들면 Prompt를 Koala 온톨로지에 적용하였을 때 시작 노드로 선택한 Animal 노드의 동위 클래스(equivalence class)로 선언된 Parent 등 몇몇 노드들은 추출되는 모듈에 포함되지 않았는데 이는 Protégé Prompt에서 동위 클래스를 모듈 추출 시 관계 추출에서 제외하기 때문으로 추정된다. 이와 같이 Prompt의 경우 연관성이 높은 노드들과 공리들을 모두 하나의 모듈로 추출하지만 모든 공리들을 완전하게 보존하지는 못하였다. 한편 PATO의 경우, 생성되는 모듈의 크기가 대부분 다른 기법에 비해 적는데, 이는 논리적 일치성을 보존하기 보다는 모듈의 크기에 더욱 비중을 두는 것을 알 수 있었다. 즉, PATO

가 논리적 일치성을 보존하기 위해 관련된 노드와 관계를 모두 포함시켜 모듈을 크게 만들기보다는 사용자에게 의해 지정된 깊이 옵션 사항에 따라 제한된 깊이의 모듈로 분할하는 것에 비중을 두는 것을 모듈화 결과 분석을 통해 알 수 있었다.

5. 시사점

본 연구에서 개발한 평가 지표를 적용한 결과를 통하여 다양한 모듈화 방식에 따라 생성된 모듈의 크기, 응집도, 결합도 차이와 모듈화 과정의 일치성에 대한 평가 결과를 알 수 있었다. 모듈화 기법 중 추출법과 분할법을 비교하였을 때 모듈화의 완전성 측면에서 분할법은 노드의 손실이나 겹침 없이 분할되는 모듈로 노드들을 배분시킨다. 그러나 노드와 관련된 공리의 보존 특성은 동일한 분할법에 속하더라도 모듈화 도구에 따라 결과가 일정하지 않았다. 즉, SWOOP을 적용하였을 경우 생성된 모듈이 논리적으로 완전하며 다른 모듈로의 링크 정보를 유지하였다. 그러므로 다른 모듈로의 링크를 보유함에 따라 일부 모듈에서 결합도가 상대적으로 강하게 나타났다. 또한 정보의 손실을 막기 위해 연관된 노드와 공리를 대부분 한 모듈에 포함하여 해당 모듈이 상대적으로 커지게 되는 크기의 비대칭 현상을 보였다. 한편, Prompt는 사용자가 지정한 노드에서부터 순회를 시작하여 관련된 노드들과 공리들을 모아 하나의 모듈을 추출하므로 모듈로 추출되지 않는 나머지 부분에 대한 연관성이 비교적 낮고 참조 정보도 유지하지 않았다. 그러므로 결합도가 상대적으로 낮게 나타난다. 그러나 관련된 모듈과 공리들을 대부분 추출하여 모듈의 크기는 상대적으로 큰 것으로 나타났다. 이에 반하여 PATO는 노드 간의 관계가 쉽게 단절되어 모듈의 크기는 상대적으로 작고 결합도가 높아지

는 것을 알 수 있었다.

본 연구에서 제안한 온톨로지 평가 프레임워크는 다양한 온톨로지 모듈화 기법들을 평가할 수 있는 기준으로 활용될 수 있을 것이다. 따라서 향후 적합한 모듈화 기법의 선택 사용을 통한 온톨로지 구축과 관리가 용이해지고 보다 활성화되리라 기대된다. 특히 모듈화 목표와 사용자의 관심 대상, 중요도에 따라 평가 지표에서 고려하는 항목과 항목의 중요도를 융통성있게 조절하여 활용할 수 있을 것이다. 예를 들면 사용자의 관심 대상이 클래스간의 계층 관계인 경우, 계층 관계에 대하여 중요한 가치를 부여하여 결합도와 응집도에 반영할 수 있다.

본 연구는 학문적 측면에서 온톨로지 모듈 및 모듈화 평가 지표를 제시하고 웹 상에 공개된 온톨로지를 모듈화하고 적용해 봄으로써 온톨로지 모듈화 평가에 대한 실증적 타당성을 보였다. 이와 같이 온톨로지 모듈화 평가 작업의 목적은 온톨로지에 대한 기술적 지식 없는 온톨로지 사용자들이 쉽게 활용할 수 있는 온톨로지 모듈화 평가 프레임워크를 제시하여 사용 목적에 적합한 온톨로지 모듈화 방법을 선택할 수 있도록 하고, 온톨로지 공학자들에게 온톨로지 모듈화 기법을 평가하는 기준 및 측정방식을 제시하여 온톨로지 모듈화 작업의 품질을 높일 수 있도록 하기 위함이다.

6. 결론

본 논문에서는 소프트웨어 모듈의 품질 측정을 위해 주로 사용되었던 모듈 크기, 응집도와 결합도 지표를 온톨로지 모듈 평가 분야에 적용하였다. 또한 온톨로지 평가에서 중요하게 인식되어 온 모듈화 과정의 논리적 일치성을 측정하는 구체적인 평가지표를 개발하고 모듈화 도구에 대한 평가를 포

함시켜 총체적으로 모듈화 기법을 평가할 수 있는 프레임워크를 제시하였다는데 의의가 있다.

본 연구에서 제시한 평가 프레임워크는 도구평가의 정성적인 방법과 메트릭을 사용한 정량적인 방법을 모두 사용하여 상호 보완적으로 평가한다. 또한 프레임워크의 검증을 위하여 온톨로지 모듈화 기법들을 대상으로 평가 프레임워크를 적용하는 실증적 방법을 사용하였다.

향 후 과제로는 본 논문에서 정의한 온톨로지 모듈 평가 프레임워크의 평가지표들을 지속적으로 개선시키며 온톨로지가 활발하게 적용될 수 있는 응용 분야의 온톨로지를 대상으로 그 유용성을 입증하는 것이다.

참고문헌

- Brewster, C., H. Alani, S. Dasmahapatra, and Y. Wilks, "Data driven ontology evaluation", *Proceedings of Int. Conf. on Language Resources and Evaluation*, Lisbon, (2004).
- Burton Jones, A., V. C., Storey, V. Sugumaran, and P. Ahluwalia, "A semiotic metrics suite for assessing the quality of ontologies", *Data and Knowledge Engineering*, (2004).
- Chidamber S. R. and C. F. Kemerer, "A metrics Suite for Object Oriented Design", *IEEE Trans. on Software Engineering*, Vol.20, No.6(1994).
- Constantine and Yourdon, *Structured Design*, Prentice Hall, Englewood Cliffs, NJ, 1979.
- Cuenca Grau, B., H. Ian, K. Yevgeny, and S. Ulrike, "Just the right amount : extracting modules from ontologies", *Proceedings of the 16th international conference on World Wide Web*. ACM Press, Banff, Alberta, Canada (2007).
- Cuenca Grau, B., B. Parsia, E. Sirin, and A.

- Kalyanpur, "Automatic Partitioning of OWL Ontologies Using E-Connections", *In: Proc. of Description Logic Workshop, (DL)*(2005).
- d'Aquin, M., M. Sabou, and E. Motta, "Modularization : a Key for the Dynamic Selection of Relevant Knowledge Components", *In : Proc. of the ISWC 2006 Workshop on Modular Ontologies*, (2006).
- d'Aquin, M., P. Doran, E. Motta, and V. Tamma, "Towards a Parametric Ontology Modularization Framework Based on Graph Transformation," *Workshop : International Workshop on Modular Ontologies*, K-CAP(2007).
- D'Aquin, M., A. Schlicht, H. Struckenschmidt, and M. Sabou, "Criteria and Evaluation for Ontology Modularization Techniques", *Lecture Notes in Computer Science : Modular Ontologies*, (2009), 67~89.
- Doran, P., V. Tamma, and L. Iannone, "Ontology module extraction for ontology reuse : an ontology engineering perspective", *CIKM* (2007).
- Gomez-perez, A., "Evaluation of Ontologies", *International Journal of Intelligent Systems*, Vol.16(2001), 391~409.
- Loebe F., "Requirements for Logical Modules", *In: Peter Haase, Vasant Honavar, Oliver Kutz, York Sure and Andrei Taminin (ed.) : Proceedings of the First International Workshop on Modular Ontologies*, (WoMO 2006).
- Noy, N. and M. Musen, "Specifying Ontology Views by Traversal", *In : Proc. of the International Semantic Web Conference*, (ISWC) (2004).
- Noy, N. and M. Musen, "PROMPT : Algorithm and Tool for Automated Ontology Merging and Alignment", *In : Proc. of the 17th National Conference on Artificial Intelligence*, (2000).
- Page-Jones, M., *The practical guide to structured systems Design*, YOUDON Press, New York, NY, 1980.
- Schlicht, A., H. Stuckenschmidt, "Towards Structural Criteria for Ontology Modularization", *In : Proc. of the ISWC 2006 Workshop on Modular Ontologies*, (2006).
- Seidenberg, J., and A. Rector, "Web Ontology Segmentation : Analysis, Classification and Use", *In : Proc. of the World Wide Web Conference, (WWW)*(2006).
- Stuckenschmidt, J., and M. Klein, "Structure Based Partitioning of Large Concept Hierarchies", *In : Proc. of the International Semantic Web Conference*, (ISWC)(2004).

Abstract

Ontology Modularization Evaluation Framework

Sunju Oh*

Several techniques and methods for ontology modularization have been proposed recently. However, there are few ontology evaluation frameworks to evaluate these techniques and methods. Most researches on ontology modularization have not been focused on ontology modularization evaluation but ontology modularization process itself. In this paper, we devise a novel ontology modularization evaluation framework to measure the quality of ontology modules, logical integrity during modularization process and modularization tools. Experiments were conducted to validate the proposed framework. Three representative modularization approaches SWOOP, Prompt, and PATO were chosen and used to partition or extract modules from an ontology. Then the proposed evaluation framework is applied to these modules. The experiment results indicate that the modularization framework works well. The proposed framework would help ontology engineers improve ontology module quality, anticipate and reduce future maintenance as well as help ontology users to choose ontology modules that best meet their requirements.

Key Words : Ontology, Module, Modularization, Metrics, Coupling, Cohesion

* School of Computer Science and Engineering, Seoul National University

저자 소개



오선주

서울 대학교 계산 통계 학과에서 학사와 석사를 취득하였으며 서울대학교 경영학
과에서 박사 학위를 취득하였다. 삼성SDS에서 책임연구원으로 일했으며 현재 서
울대학교 컴퓨터 공학과에서 박사 후 연구원으로 근무하고 있다. 관심 분야는 온
톨로지 엔지니어링, 인터넷 비즈니스, 지식관리, 시스템 인테그레이션 등이다.