

# 이질적인 쇼핑몰 환경을 위한 온톨로지 기반 상품 매핑 방법론

김우주

연세대학교 컴퓨터산업시스템공학부  
(wkim@yonsei.ac.kr)

최남혁

연세대학교 컴퓨터산업시스템공학부  
(cera8413@yonsei.ac.kr)

최대우

연세대학교 E-비즈니스 연구센터  
(qorwkr@nate.com)

의미 웹(Semantic Web)과 이에 대한 관련기술들은 웹을 통해 자유롭게 정보를 공유할 수 있는 새로운 지평을 마련해 주었다. 이를 토대로 온톨로지(Ontology)에 대한 연구 역시 활발히 진행되어 왔으며, 결과적으로 온톨로지는 여러 가지 측면에서 가시적인 성과들을 이룩해 낼 수 있었다. 의미 웹에 대한 기대가 고조되면서, 온톨로지는 많은 연구 학자들로부터 각광을 받는 분야로 떠오를 수 있었으며, 오늘날, 이러한 온톨로지에 대한 연구는 지식표현(Knowledge-Representation)등과 같은 학술적인 단계에서 벗어나 전자상거래 분야까지 그 영역을 넓혀 나가고 있다. 그러나 일부에서는 이러한 온톨로지의 양산으로 인해 발생할 문제들에 대한 우려의 목소리가 적지 않다. 즉, 현재 온톨로지들은 서로간의 이질성으로 인해 그 효용 가치를 위협받고 있다는 것이다.

예를 들어, 현재 Yahoo나 DMOZ Open Directory(www.dmoz.org)등으로부터 제공 중인 상품 카테고리의 경우, 시각적으로는 유사해 보일지 모르나 그들이 지닌 카테고리 구조라든지 사용된 용어 등을 살펴보면 상당한 차이점이 존재한다는 사실을 발견해 낼 수 있다. 따라서 현재 온톨로지가 지향하는 본래의 효용 가치를 극대화시키기 위해서는 이러한 이질성을 극복할 수 있는 방법론이 반드시 필요하다고 할 수 있다.

본 연구에서는 현재 온톨로지 분야에서 가장 커다란 이슈로 떠오르고 있는 정보통합에 대한 해결책을 제시해 보고자 한다. 정보통합이란, 서로 이질적인 프로그래밍 언어나 형식으로 이루어진 데이터들을 통합하여 요약한 형태로 제공하는 기술이라 정의내릴 수 있다. 이러한 정보 통합의 대표적인 분야로 Merging, Alignment, Translation, Articulation을 들 수 있는데, 이 가운데, 본 연구에서는 “Ontology Alignment”에 대해 집중적으로 다루어 보고자 한다. 이와 더불어, 현재 의미 웹이 지향하는 대표적인 특성 중 하나로써 상호운영성을 꼽을 수 있다. 온톨로지 매핑(Ontology mapping)은 이러한 상호운영성을 지원하기 위한 대표적인 기술로써 각광을 받고 있다. 따라서 우리는 이러한 온톨로지 매핑에 대한 방법론을 쇼핑몰 환경에 적용함으로써 서로 이질적인 상품 카테고리 구성된 두 쇼핑몰간의 상품에 대한 매핑 알고리즘(Mapping Algorithm)을 제안하고자 한다. 또한, 효용성 검증을 위하여 온톨로지 매핑에 있어 이미 효용성을 검증 받은 PROMPT와의 비교 분석도 수행하여 본다.

논문접수일 : 2005년 11월

게재확정일 : 2006년 6월

교신저자 : 김우주

## 1. 서론

의미 웹과 이에 대한 관련기술들은 웹을 통해

자유롭게 정보를 공유할 수 있는 새로운 지평을 마련해 주었다. 이를 토대로 온톨로지에 대한 연구 역시 활발히 진행되어 왔으며, 결과적으로 온

톨로지는 여러 가지 측면에서 가시적인 성과들을 이룩해 낼 수 있었다. 일반적으로 온톨로지는 분산 환경하에서 사용자나 어플리케이션 간의 정보 공유를 지원하기 위한 도구로 인식되고 있다. 의미 웹에 대한 기대가 고조되면서, 온톨로지는 많은 연구 학자들로부터 각광을 받는 분야로 떠오를 수 있었으며, 이에 따라 온톨로지 개발에서부터 시작해 재사용에 이르기까지 수많은 관련 연구들이 진행되어 왔다. 의미 웹에 대한 기대가 고조되면서, 온톨로지는 많은 연구 학자들로부터 각광을 받는 분야로 떠오를 수 있었으며, 이에 따라 온톨로지 개발에서부터 시작해 재사용에 이르기까지 수많은 관련 연구들이 진행되어 왔다. 오늘날, 이러한 온톨로지에 대한 연구는 지식표현(knowledge-representation)등과 같은 학술적인 단계에서 벗어나 전자상거래 분야까지 그영역을 넓혀 나가고 있다. 전자상거래상의 여러 기업들은 온톨로지를 이용하여 서로 간의 정보를 공유할 뿐만 아니라 자신의 고객들에게까지 유용한 정보를 제공해 주고 있다. 자신의 웹 사이트 자체를 카테고리 형식으로 제공하는 Yahoo나 필요한 상품 정보만을 선별적으로 카테고리화시켜 제공하는 Amazon.com과 같은 사이트들이 이에 대한 대표적인 예라 할 수 있다. 이러한 추세에 따라 가까운 미래에는 다양한 분야를 통해 수 많은 온톨로지들이 구축될 것으로 예상된다. 그러나 일부에서는 이러한 온톨로지의 양산으로 인해 발생할 문제들에 대한 우려의 목소리가 적지 않다. 즉, 현재 온톨로지는 서로간의 이질성으로 인해 그 효용 가치를 위협받고 있다는 것이다. 예를 들어, 현재 Yahoo나 DMOZ Open Directory 등으로부터 제공 중인 상품 카테고리의 경우, 시각적으로는 유사해 보일 지 모르나 그들이 지닌 카테고리 구조라든지 사용된 용어 등을 살펴보면 상당한 차이가 존재한다는 사실을 발견

해 낼 수 있다. 따라서 현재 온톨로지가 지향하는 본래의 효용 가치를 극대화시키기 위해서는 이러한 이질성을 극복할 수 있는 방법론이 반드시 필요하다 할 수 있다. 따라서 본 연구를 통해 우리는 현재 온톨로지 분야에서 가장 커다란 이슈로 떠오르고 있는 정보통합에 대한 해결책을 제시해보고자 한다. 정보 통합의 관점에 있어 서로 다른 온톨로지 간의 유사성을 측정하고, 그들 간의 매핑을 지원하는 작업은 핵심적인 부분으로 여겨지고 있다. 또한, 현재 의미 웹이 지향하는 대표적인 특성 중 하나로써 상호운영성을 꼽을 수 있는데, 온톨로지 매핑은 이러한 상호운영성을 지원하기 위한 대표적인 기술로써 각광을 받고 있다.

우리는 이러한 온톨로지 매핑에 대한 방법론을 쇼핑몰 환경에 적용함으로써, 서로 이질적인 상품 카테고리로 구성된 두 쇼핑몰간의 상품에 대한 매핑 알고리즘을 제안하고자 한다. 또한, 효용성 검증을 위하여 온톨로지 매핑에 있어 이미 효용성을 검증 받은 PROMPT와의 비교 분석도 수행하여 본다. 2장에서는 본 연구를 제안하게 된 동기와 필요성에 대해 언급할 것이며, 3장에서는 본 연구를 통해 우리가 제안하고자 하는 방법론에 대해 서술 하도록 하겠다. 이어지는 4장에서는 3장에서 언급한 방법론에 대한 실험 결과를 토대로 PROMPT 방법론과의 비교 분석을 수행해 보고자 하며, 마지막으로 5장에서는 본 연구를 통해 얻은 결론을 맺으며 마무리 하도록 한다.

## 2. 연구의 배경 및 필요성

본 장에서는 현실 세계에서 온톨로지 매핑의 필요성과 함께 연구 수행 동기 관해 간략히 언급해 보고자 한다. 전술한 바와 같이 현재 활동중인

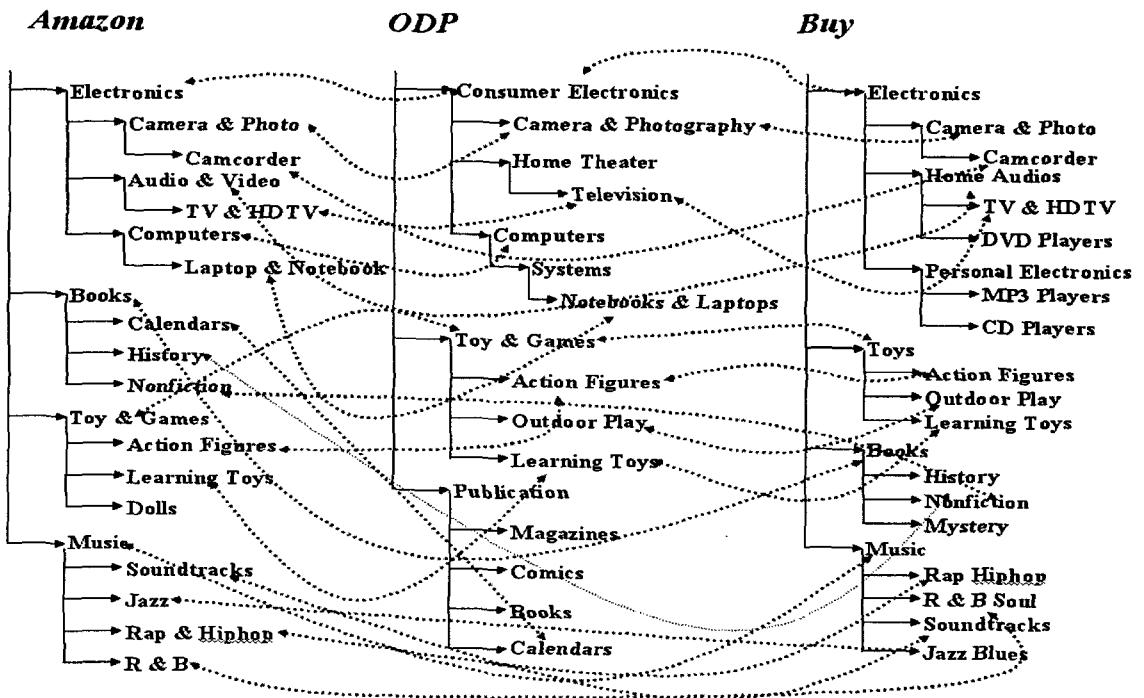
대다수의 쇼핑물 사이트들은 사용자가 원하는 상품에 대한 효율적인 검색을 지원해 주지 못하고 있다. 일례로 보스턴 컨설팅 그룹에 따르면, 현재 전자상거래를 이용 중인 모든 사용자 가운데 절반에 가까운 48%의 사용자가 원하는 상품을 검색하지 못했던 경험이 있는 것으로 밝혀졌으며, 실제 상품을 구매하고자 했던 고객들 가운데 28%는 원하는 상품에 대한 검색 실패로 인해 결국 구매를 포기해야만 했던 것으로 조사 결과 밝혀졌다. 이는 현존하는 대다수의 쇼핑물들이 서로 간의 약정을 통해 상품 카테고리를 제공하기 있기 때문인 것으로 해석되며, 이는 사용자들에게 통합적인 상품 정보를 제공하는데 있어 커다란 방해 요소로 작용하고 있다. 또한, E-Marketplace상에서 역시 효과적인 상품 카테고리 관리가 이루어 지지 못하

고 있는 것이 현실이다. 따라서 본 연구에서는 이러한 상품 정보들에 대한 효과적인 매핑을 통해, 추후 통합적인 상품 정보 제공을 위한 기반을 마련하고자 한다.

그럼 앞서 언급한 문제점들에 대해 좀 더 자세히 살펴보도록 하자. 아래 [그림 1]은 서로 다른 쇼핑물상에서 제공 중인 상품 분류 체계 간의 매핑 방식을 표현하고 있다.

위의 [그림 1]에서 살펴보는 바와 같이, 현존하는 쇼핑물들에서 제공 중인 상품 정보들은 다음과 같은 문제점들을 내포하고 있다.

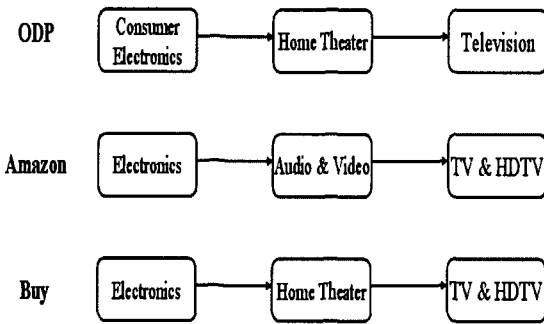
첫째, 동일한 상품일 지라도 각 쇼핑물마다 그 상품을 표현하는 어휘는 다를 수 있다. 일례로, 동일한 상품임에도 불구하고 Amazon.com에서는 “TV & HDTV”, 그리고 ODP 같은 경우는 “Televisions”



[그림 1] Product Category Mapping

로 서로 다른 어휘를 이용하여 각 상품을 분류하고 있다. 이와 같은 경우는 단순한 문자열 Matching만으로는 동일성을 판단한다는 것이 불가능하다고 할 수 있으며, 따라서 보다 진화된 형태의 방법론을 필요로 한다.

둘째, 동일한 상품일 지라도 각 쇼핑몰마다 해당 상품을 표현하는 분류체계는 다를 수 있다는 것이다. 다음의 [그림 2]를 살펴보자.



[그림 2] Difference of Conceptualization

위 그림에서 살펴보는 바와 같이, 각 쇼핑몰들은 “Television”이라는 동일한 상품을 표현하는데 있어 서로 다른 분류 체계를 사용하고 있다. 즉, 동일한 상품이라 할지라도 쇼핑몰마다 해당 상품을 표현하는 분류방식에는 차이가 있기 때문에 서로 다른 계층구조를 지니게 된다는 것이다. 따라서 쇼핑몰 상에서 사용자가 요청한 상품 분류 카테고리 와 가장 유사한 카테고리를 선택하고자 하는 경우, 상품 카테고리에 대한 계층 구조 역시 반드시 고려되어야 할 사항이라 할 수 있다.

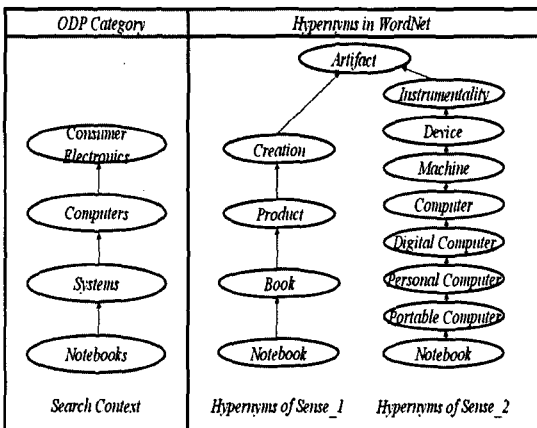
### 3. 온톨로지 기반 상품 매핑 방법론

앞서 2절에서는 서로 이질적인 카테고리 구조

로 상품 정보를 제공하는 쇼핑몰들 간의 온톨로지 매핑에 있어 발생 가능한 문제점들에 대해 분석해 보았다. 본 장에서는 앞서 지적된 문제점들을 극복하기 위해 제안된 방법론과 이를 통한 해결 방안을 모색해 보고자 한다. 전술한 바와 같이 현존하는 쇼핑몰들의 경우, 서로 이질적인 카테고리를 기반으로 상품 정보를 제공하고 있기 때문에, 사용자들로 하여금 효율적인 검색을 지원해 주지 못하고 있다. 또한, 보다 근본적인 문제는 각 쇼핑몰마다 동일한 상품이라 할지라도 그 상품에 대한 명칭은 제각각 이라는 것이다. 예를 들어, 같은 상품임에도 불구하고, Amazon.com에서는 “TV & HDTV”로 그리고 ODP에서는 “Television”으로 각각 다르게 표현되어 있다. 따라서 서로 다른 카테고리 구조상에 존재하는 상품들 가운데 서로 동일한 상품들을 매핑시키기 위해 우리는 다음과 같은 3가지 방법론을 제안하고자 한다. (1) 우선 상품 분류 온톨로지들마다 사용된 어휘가 다르기 때문에 이를 해결하기 위한 방안으로 어휘 확장단계가 필요하다. 그러나 사용된 어휘에 대해 무분별적으로 동의어나 유사어를 추출한다는 것은 사용자의 검색 의도에 부합하지 못한 상품을 검색해 낼 가능성이 크다고 할 수 있다. 따라서 우리는 어휘 확장과 더불어 상품분류 카테고리에 대한 의미를 파악하는 단계를 함께 고려해 보고자 한다. 이를 위하여 어휘의 의미 파악 및 어휘 확장 방법론을 제안 한다. (2) 다음으로 어휘 확장 단계를 통해 생성된 어휘 및 쇼핑몰의 상품 분류 온톨로지가 이용하여 검색을 수행하는 매핑 클래스 검색 단계를 제안하며, (3) 마지막으로 다수의 유사 상품이 검색되었을 경우, 사용자가 선택한 상품과 가장 유사한 상품을 선택해 주는 매핑 클래스 선택 방법론을 제안한다. 그럼 지금부터 각각의 단계에 대해 보다 구체적으로 살펴보도록 하자.

### 3.1 어휘의 의미 파악 및 확장 방법론

서로 이질적인 구조로 이루어진 상품 카테고리 간의 매핑을 수행하기 위해, 우리는 우선적으로 주어진 카테고리의 용어들에 대한 정확한 의미를 파악해 볼 필요가 있다. 우리말과 마찬가지로 영어 역시 시각적으로는 동일한 단어처럼 보이지만, 그 의미를 따져보면 전혀 다른 의미를 지닌 어휘들이 상당수 존재한다. 바로 이러한 다의어들에 관한 의미 파악 및 확장을 위해 우리는 WordNet을 이용하기로 한다. WordNet의 Synset은 각각 자신의 상위 개념들의 집합인 hypernym과 하위 개념들의 집합인 hyponym에 대한 정보를 포함하고 있는데, 본 연구에서는 이러한 특정 어휘들간의 관계들로 인해 형성된 경로와 사용자가 선택한 상품 카테고리간의 유사성 분석을 통해 어휘의 의미를 파악해 보고자 하였다. 그럼 지금부터 어휘의 의미 파악 및 확장 방법론에 대해 아래 예제를 통해 살펴보도록 하자.



[그림 3] Comparison between Category Hierarchies of WordNet and ODP

위의 [그림 3]에서 왼편에는 ODP상에서 제공 중

인 상품 “Notebook”에 대한 카테고리 정보를 나타내고 있으며, 오른쪽에 나타나는 그림은 WordNet 상에서 표현하고 있는 “Notebook”에 대한 hypernym정보이다. 이러한 정보를 토대로 지금부터 어휘 의미 파악 및 확장 방법론에 대해 구체적으로 살펴 보도록 하자.

$$pathProximity(p) = \frac{\sum_{x \in ancestor(cs_x)} hypernymProximity(x, p)}{n} \quad (1)$$

$$hypernymProximity(x, p) = \begin{cases} \frac{1}{distance(cs(x, p), cs(cat, p))} & \text{if } cs(x, p) \neq \phi \text{ and } cs(x, p) \in ancestor(cs(cat, p)) \\ 0 & \text{o.w} \end{cases} \quad (2)$$

$$cs(x, p) = \begin{cases} y & \text{if } y \in SYNSETS(x) \text{ and } y \in path(p) \\ \phi & \text{o.w} \end{cases} \quad (3)$$

[그림 4] Word Sense Disambiguation and Term Extension Algorithm

위의 그림은 해당 어휘의 정확한 의미를 파악하기 위해 정의된 수식이다. 이와 같은 수식을 정의하기 위해 우리는 다음과 같은 가정을 했다.

가정 : 사용자가 선택한 상품의 상위 노드들이 WordNet으로 인해 형성된 Path(해당 어휘의 Hypernym정보)상에 많이 존재하면 할수록, 그리고 그 노드들 간의 거리가 가까우면 가까울수록 해당 상품에 대한 정확한 의미로 선정될 가능성이 높다.

그럼 지금부터 위와 같은 가정을 바탕으로 [그림 4]에서 정의한 수식이 수행되는 과정에 대해 구체적으로 살펴보도록 하자.

우선, cs() 함수는 사용자가 요청한 상품의 상위

노드들에 대한 Coordinate Term들을 반환하는 과정이다. 즉,  $cs()$  함수는 사용자가 요청한 상품의 상위 카테고리들에 대한 Coordinate Term들과 WordNet상에서 형성된 Path와의 유사성을 측정하게 된다. Coordinate Terms란, 각 어휘들에 관한 동의어들의 집합이라 정의 할 수 있다. 따라서 만약 WordNet의 Path상에 이러한 Coordinate Term이 존재하는 경우에는 해당 상품에 대한 Coordinate Term들을 모두 반환하게 되며, 그렇지 않은 경우에는 무시된다. 아래 수식은 사용자가 “Notebook”이라는 상품 검색을 요청했다는 가정하에, 표현된 것이다. 따라서 “Notebook”상품에 대한  $cs()$  함수를 표현해보면 다음과 같다.

$$cs(x, p) = \begin{cases} y & \text{if } y \in SYNSETS(x) \text{ and } y \in path(p) \\ \phi & \text{otherwise} \end{cases}$$

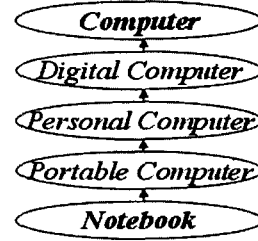
where  $x$  is a term belonging to a search context  
 $y$  is a term belonging to coordinate terms of  $x$   
 $p$  is a term belonging to a product category path

$$cs(system, path2) = \{\phi\}$$

$$cs(computers, path2) = \{computer, computing machine...\}$$

$$cs(consumer electronics, path2) = \{\phi\}$$

다음으로 이러한  $cs()$  함수로 인해 도출된 결과를 바탕으로 WordNet 경로(path)상에서 사용자가 요청한 상품과 경로상에 나타난 상위 상품들 간의 거리를 측정해내는 과정이 필요하다. 이를 위한 과정으로 [그림 4]의 hypernym Proximity() 함수를 이용하도록 한다. 우리 예제의 경우, [그림 3]상에서 살펴 보는 바와 같이, “computer”라는 상품이 있어 사용자 카테고리 와 WordNet hypernym 카테고리 간의 중복이 이루어지고 있을 확인해 볼 수 있다. 따라서 우리는 hypernym Proximity()에서 정의한 수식에 따라 이들간의 거리를 측정해 보았다.



위 그림은 [그림 3]에 나타나는 WordNet hypernym의 2번째 계층 구조 가운데 일부를 발췌한 것이다. hypernymProximity() 함수에서 노드 간의 거리는 아크(arc) 수를 바탕으로 측정이 이루어 지도록 정의하고 있다. 따라서 그림 상에서 “Notebook”과 그 상위 노드에 해당하는 “Computer”간의 거리는 4이며, 이들 간의 hypernymProximity()을 수행해 보면, 최종적으로 1/4이라는 값이 도출된다. 다음 수식은 “Notebook” 상품에 관한 hypernymProximity() 함수의 수행 과정을 나타내고 있다.

$$hypernym\ Proximity(x, p) = \frac{1}{distance(cs(x, p), cs(cat_n, p))}$$

where  $x$  is a term belonging to a search context  
 $distance()$  returns the number of arcs of a set  $cs()$

$$hypernymProximity(system, path2) = 0$$

$$hypernym\ Proximity(computer, path2) = \frac{1}{4}$$

$$hypernym\ Proximity(consumerelectronics, path2) = 0$$

다음으로 어휘 의미 파악 및 측정을 위한 마지막 과정으로 pathProximity() 함수에 대해 살펴해보도록 하자. pathProximity()는 앞서 hypernymProximity()를 통해 도출된 값을 사용자가 선택한 카테고리의 총 노드 수로 나누어 주는 과정이라 할 수 있다. 우리 예제의 경우, ODP의 “Notebook”상품 카테고리의 총노드 수는 4개임을 확인해볼 수 있으며, 따라서 앞서 hypernymProximity()를 통해 도출된 값들의

합을 총 노드 수에 해당하는 4로 나누어 주게 된다. 이에 대한 과정을 수식으로 표현해 보면 다음과 같다.

$$path\ Proximity(path1) = \left( \frac{0 + 0 + 0}{4} \right) = 0$$

$$path\ Proximity(path2) = \left( \frac{0 + \frac{1}{4} + 0}{4} \right) = 0.0625$$

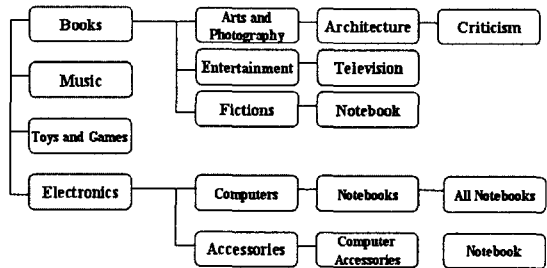
위 예제에서 살펴보았듯이, 우리 예제의 경우, WordNet을 통해 생성된 2개의 경로 가운데, WordNet상에서 “a small compact portable computer”의 의미로 사용된 2번째 의미가 사용자로부터 요청된 상품에 대한 올바른 의미로 선택 되었음을 알 수 있다. 따라서 “Notebook”의 2번째 의미에 대한 Coordinate term들이 검색을 위한 확장 어휘로써 사용된다.

### 3.2 매핑 클래스 검색 방법론

사용자가 선택한 상품 카테고리의 의미 파악 및 어휘 확장이 완료되면, 이렇게 확장된 어휘를 이용하여 쇼핑몰 상에서 사용자가 요청한 상품과 유사한 카테고리들만을 검색해 내는 과정이 필요하다. 이를 위해 본 절에서는 매핑 클래스 검색 방법론을 제안하고자 한다. 매핑 클래스 검색 방법론은 다음과 같은 2가지 단계로 구성된다. (1) 우선 순수하게 사용자가 선택한 상품명만을 가지고 검색을 수행하는 경우이다.

[그림 5]는 현재 Amazon.com에서 제공 중인 상품 카테고리 가운데 일부를 발췌한 것이다. 이와같은 경우, 우리는 단계를 1을 통해 “Notebook”, “Notebooks”, 그리고 “All Notebooks”등을 검색해 낼 수 있다. (2) 그러나, “Books - Fiction -

Notebook”으로 이어지는 카테고리 하에 존재하는 “Notebook”은 본래 사용자가 의도한 상품의 의미가 아닌, “Notebook”이라는 제목을 가진 소설책의 의미로 사용되고 있다. 따라서 이렇게 사용자의 의도에 부합하지 않는 상품들을 선별해 내기 위해서는, 그 상품이 지닌 상위 카테고리 정보를 파악해 볼 필요가 있다. 이에 대한 예를 한가지 살펴보자. 우선, 아래와 같이 선택된 상품들에 대한 상위 카테고리 정보를 파악한다. 다음으로 검색된 상품들의 상위 클래스에 해당하는 “Electronics”, “Accessories”, 및 “Computer Accessories”, “Computer”등과 사용자가 선택한 ODP의 상품 카테고리상에서 “Notebook”의 상위 카테고리에 해당하는 “System”, “Computer” 그리고 “Consumer Electronics”와의 비교를 수행한다.



[그림 5] Category Example of Amazon.com

- Books/Fictions/Notebook
- Electronics/Accessories/Computer
- Accessories/Notebook
- Electronics/Computer/Notebooks

위의 카테고리 정보를 통해 살펴볼 수 있듯이, “System”과 같은 경우, 유사한 의미를 지닌 카테고리가 존재하지 않고 있다. 그러나 “Computer”, “Consumer Electronics”와 같은 경우는 Amazon

카테고리상의 “Computer”와 “Electronics”와 유사한 의미를 지니고 있다는 사실을 확인해 볼 수 있다. 따라서 이 과정을 통해, “Computer” 및 “Electronics”의 의미가 포함되지 않은 1번째 카테고리리는 고려 대상에서 제외된다. 즉, 선택된 3개의 상품 카테고리 가운데 다음의 2개에 해당하는 카테고리만 남게 된다.

*Electronics/Accessories/Computer  
Accessories/Notebook  
Electronics/ Computer /Notebooks*

### 3.3 매핑 클래스 선택 방법론

3.2절의 매핑 클래스 검색 방법론을 통해 우리는 ODP의 “Notebook”과 유사한 카테고리 구조를 지닌 2개의 카테고리를 선정했다. 본 절에서는 가장 마지막 단계에 해당하는 매핑 클래스 선택 방법론에 대하여 살펴보도록 한다. 매핑 클래스검색을 통해 추출된 2개의 카테고리 가운데 사용자가 요청한 상품과 가장 유사한 상품 카테고리를 선정하기 위하여 우리는 Co-Occurrence와 Order-Consistency 라는 개념을 정의하고 이를 정량화시켜 보았다. 그럼 지금부터 앞서 매핑 클래스 검색 방법론을 통해 검색된 카테고리 가운데 사용자가 요청한 ODP의 상품 카테고리라 가장 유사한 카테고리만을 선별해 내는 과정에 대해 살펴보도록 하자. 다음 <표 1>은 각각의 경우에 따른 어휘 처리 방법론을 도식화 시켜 나타내주고 있다.

보다 효과적인 유사성 측정을 위해서는 현재 카테고리 상에서 제시된 어휘들에 대한 표준화 작업이 필요하며, 이렇게 표준화 과정을 거쳐 생성된 어휘를 토대로 유사성을 평가하게 된다. 아래 함수 (1)은 각 용어들에 대한 유사성 측정을 위해 정의되었으며, 이를 통해 정의된 용어를 바탕으로

함수 (2)의 과정을 수행하게 된다.

<표 1> Structure Classification of Category name

Features of Category Name	Transformation	Example	
①One Word	Use as it is	“Notebook”	
②more than two words and existing in WordNet	Use as it is	“Home Theater”	
more than two words and existing not in WordNet	③existing conjunction between words	Use as it is	“Computer Accessory”
	④existing conjunction such as “and”, “&” between words	Words partition or use Max Value	“Notebook & Laptop”→“Notebook”, “Laptop”
	⑤existing preposition “for” between words	Remove after “for”	“Gift for child” →“Gift”

*TermMatch (term<sub>1</sub>, term<sub>2</sub>)*

$$= \begin{cases} \frac{\text{strlen}(\text{term}_1)}{\text{strlen}(\text{term}_2)} & \text{if } \text{term}_1 \text{ is substring of } \text{term}_2 \\ \frac{\text{strlen}(\text{term}_2)}{\text{strlen}(\text{term}_1)} & \text{if } \text{term}_2 \text{ is substring of } \text{term}_1 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

*Similarity (Context<sub>-</sub>Cate, Cate)*

$$= \frac{\text{Max}_{\text{Cterm} \in \text{ExtendedTermSet}(\text{Context}_{-}\text{Cate})} \text{TermMatch}(\text{Cterm}, \text{Cate})}{\text{TermMatch}(\text{Cterm}, \text{Cate})} \quad (2)$$

예를 들어, 사용자가 선택한 “Notebook”과 Electronics/Accessories/Computer Accessories/ Notebook의 카테고리 상에 존재하는 “Notebook”에 대한 유사성은 <표 1>의 ①번째 경우에 해당하며, 따라서 “Notebook”은 아무런 변형 과정 없이 측정이 이루어진다.

또 다른 예로 “Notebook”과 Electronics/Computer/ Notebook으로 구성된 카테고리 상의 “Notebook”에 대한 유사성을 측정해 보면, 다음과 같다.



<표 2> Example of Similarity Calculation

"Television"	Notebook	TermMatch
notebook	Notebook	8
notebook computer	Notebook	0
Similarity(notebook, Notebook)		8/8

<표 3> Example of Similarity Calculation

"Television"	Notebook	TermMatch
notebook	Notebook	8
notebook computer	Notebook	0
Similarity(notebook, Notebook)		8/8

이와 같은 측정 방법을 통해 우리는 사용자가 선택한 카테고리화 가장 유사한 카테고리를 판별해 낼 수 있게 되며, 결과적으로 가장 큰 유사도 값을 지닌 카테고리가 선택된다. 따라서 앞선 예제의 경우 다음 <표 4>와 같은 결과를 얻게 된다.

그럼 앞서 살펴보았던 값들이 어떠한 과정을 통해 도출되었는지 정의된 수식을 바탕으로 좀 더 구체적으로 살펴보도록 하자. 우선, 비교 대상으로 선정된 두 카테고리간의 용어 중복성 정도를 측정하기 위한 Co-Occurrence에 관한 수식이다.

$Co\_Occurrence(Search\ Context, path)$

$$= \left( \frac{\sum_{Context\_Cate \in Search\ Context} Similarity(Context\_Cate, path)}{ns(Search\ Context)} \right)$$

where  $ns(x)$  is the number of elements of a list  $x$ .

(3)

<표 4> Calculation Example of Similar Category and its Similar Degree

Search Context	Electronics/Accessories/ Computer Accessories/Notebook		Electronics/Computer/ Notebook	
	Similar Category	Degree	Similar Category	Degree
Consumer Electronic	Electronics	11/19	Electronics	11/19
Computer	Computer Accessories	8/17	Computer	8/8
System	X	0	X	0
Notebook	Notebook	8/8	Notebook	8/8

$Co\_Occurrence(Search\ Context, \dots,$

$Electronics/Accessories/Computer/Accessories/Notebook)$

$$= \left( \frac{\frac{11}{19} + \frac{8}{17} + \frac{8}{8}}{4} \right) \left( \frac{11}{19} + \frac{8}{17} + \frac{8}{8} \right) = 0.512$$

$Co\_Occurrence(Search\ Context, \dots,$

$Electronics/Computer/Notebook)$

$$= \left( \frac{\frac{11}{19} + \frac{8}{8} + \frac{8}{8}}{4} \right) \left( \frac{11}{19} + \frac{8}{8} + \frac{8}{8} \right) = 0.555$$

다음으로 *Order-Consistency*는 비교 대상으로 선정된 두 카테고리 상에 존재하는 노드들의 순서가 얼마나 일치하는지에 대한 여부를 측정하는 과정이다. 측정에 앞서 이 과정에서 사용된 몇 가지 용어들에 대해 정의해보자. 먼저  $common(Search\ Context, Path)$ 은 서로 다른 두 계층구조상에서 중복되는 클래스들에 대한 선, 후행 관계 수를 고려하기 위해 정의되었다. 즉, 선택된 쇼핑물 클래스의 계층구조와 사용자가 선택한 클래스의 계층구조 사이에 중복되는 클래스들의 집합을 의미하는 것이다. 예를 들어, {Consumer Electronics, Computer, System, notebook} (사용자가 선택한 카테고리)과 {Electronics/Computer/Notebook} (쇼핑물 카테고리)의 경로를 비교해 볼 경우,  $common$  값으로는 {Consumer Electronics, Computer, Notebook}이 도출될 수

있다. 다음으로  $prelset(Category\ set, Category\ path)$ 은 클래스 계층 구조상에서 발생 가능한 모든 선, 후행 관계들의 집합을 정의한다. 예를 들어,  $prelset(\{(A,B,C), A/B/C\})$ 의 경우에는  $\{(A,B), (A,C), (B,C)\}$ 의 총 3가지 관계가 정의될 수 있다. 다음으로  $MaxSimNode(Context\_Cate, path)$ 는 가장 유사한 카테고리 검색해 내는 기능을 한다. 다음으로  $cons$ 는 두 카테고리상에 동일한 노드가 존재하는 경우는 1의 값을, 그렇지 않은 경우는 0의 값을 반환하게 된다.

$$cons((cat_1, cat_2), path) = \begin{cases} 1 & \text{if } n_1 = MaxSimNode(cat_1, path), \\ & n_2 = MaxSimNode(cat_2, path) \\ & \text{and } n_1 \text{ precedes } n_2 \text{ in path} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

$$cons\left(\left(\begin{matrix} (ConsumerElectronics, Notebook), \\ Electronics / Computer / Notebooks) \end{matrix}\right), \begin{matrix} Electronics = \\ MaxSimNode(ConsumerElectronics, Electronics / Computer / Notebooks), \\ Computer = \\ MaxSimNode(Computer, Electronics / Computer / Notebooks) \\ Notebook = \\ MaxSimNode(Computer, Electronics / Computer / Notebook) \text{ and} \\ Electronics \text{ precedes } Notebooks \text{ in } Electronics / Computer / Notebook \end{matrix}\right)$$

함수 (4)에서 살펴보는 바와 같이 우리가 제시한 카테고리 예에서는 동일한 노드가 존재하기 때문에 1의 값을 되돌려준다. 마지막으로  $Constc$ 은  $cons$ 에 대한 평균값을 의미하는 것으로 이를 정량화 시켜보면 다음과 같다.

$$constc(Search\ Context, path) = \frac{\sum_{pr} cons(pr, path)}{n}$$

where  $pr \in prelset\left(\begin{matrix} common(Search\ Context, path), \\ Search\ Context \end{matrix}\right)$

$$n = \text{number of element of } prelset\left(\begin{matrix} common(Search\ Context, \\ Search\ Context \end{matrix}\right) \quad (5)$$

최종적으로 사용자가 선택한 카테고리 정보와 메뉴 경로 간의 유사성은 앞서 제안한 Co-Occurrence와 Order-Consistency를 바탕으로 아래의 수식을 통해 도출 가능하며, 각 대안 경로들 가운데 가장 큰 Simpath값을 지닌 클래스가 선택된다.

$$SimPath(cat\_info, path) = \frac{Co\_Occurance(SearchContext, path) + constc(SearchContext, path)}{2} \quad (6)$$

이상의 과정을 통하여 앞서 예들 든 ODP의 “Notebooks”과 같은 경우, Amazon.com의 “Notebooks”와 최종적으로 매핑이 이루어지게 된다.

#### 4. 실험 평가

본 장에서는 앞서 3장에서 언급한 방법론을 통해 얻은 실험 결과들을 바탕으로 이에 대한 분석을 수행하고자 하며, 또한 이를 PROMPT의 결과값과 비교해 보기로 한다. PROMPT란, 온톨로지 간의 매핑을 지원하기 위해 개발된 도구로써 미국 스탠퍼드 대학으로부터 처음 제안되었다. 그럼 PROMPT가 지닌 기능에 대해서 간략히 살펴볼도록 하자. PROMPT는 서로 이질적인 계층 구조로 이루어진 두 온톨로지에 대한 비교를 수행하여 동

일하거나 비슷한 의미를 지닌 어휘들에 대해서는 추천을 수행함으로써 서로 간의 매핑을 지원하고 있으며, 또한, 사용자 판단 하에 추가적으로 매핑이 필요하다고 생각되는 어휘들에 대해서는 사용자가 직접 매핑을 수행할 수 있도록 하는 기능도 제공한다. 그러나 쇼핑몰 상품 검색의 특성상 사용자가 직접 상품에 대한 매핑을 수행한다는 것은 의미가 없으므로, 본 연구에서는 PROMPT로부터 추천된 값만을 비교 대상으로 고려했다. 따라서 본 장에서는 우리가 제안한 방법론과 PROMPT간의 실험 결과를 토대로 비교, 분석을 수행해 보기로 한다. 우선 4.1절에서는 실험을 위해 사용된 데이터에 대해 간략히 서술 하도록 하고, 이어지는 4.2절에서는 4.1절에서 소개된 데이터를 통해 도출된 결과들을 토대로 PROMPT와의 비교, 분석을 수행해 보도록 한다.

#### 4.1 실험 데이터

본 연구의 실험을 위하여 우리는 대표적인 쇼핑몰 몇 개를 선정하여 그들이 제공하고 있는 상품들에 대한 카테고리 정보를 온톨로지로 구성해 보았다. 온톨로지 구성을 위한 언어로는 시맨틱 웹 관련 언어인 OWL(Web Ontology Language)을 사용하였다. 실험 데이터 구성을 위해 선정된 쇼핑몰 사이트들은 Amazon.com과 Buy.com 그리고 보다 객관적인 실험을 위해 현재 일반적인 쇼핑몰에서 제공하는 상품들과는 별도로, 추가적인 상품에 대한 정보까지 포괄적으로 제공하고 있는 ODP의 쇼핑 카테고리를 포함시켜 보았다. 따라서 우리는 이렇게 선정된 3개 사이트에 대한 카테고리 정보를 OWL 형태의 온톨로지로 구성하여 우리가 제안한 방법론과 PROMPT의 비교 실험을 위한 데이터로 사용함으로써 데이터 일관성을 유지하였다.

#### 4.2 실험결과 비교 및 분석

실험 결과를 확인하기에 앞서 우선 우리가 어떠한 평가 방식을 기준으로 실험을 수행하였는지 살펴보기로 하자. 앞서 4.1절에서 이미 언급하였듯이, 우리는 선정된 3개의 온톨로지를 바탕으로 실험을 수행하였다. 온톨로지 매핑을 위해서는 우선적으로 비교대상이 될 수 있는 소스(Source) 온톨로지와 타겟(Target) 온톨로지에 대한 데이터가 필요하다고 할 수 있는데, 본 실험에서 우리는 각각의 온톨로지를 소스 와 타겟 온톨로지에 대한 데이터로 사용하였다. 다시 말해서 하나의 온톨로지는 한 번의 실험에서는 소스 온톨로지에 대한 모델로 그리고 나머지 한번의 실험에서는 타겟 온톨로지의 모델로 사용될 수 있다는 것이다.

예컨대, Amazon.com에 대한 온톨로지가 소스 온톨로지로 사용될 경우에는 Buy.com과 ODP가 타겟 온톨로지의 역할을 수행하게 되며, 반대로 Buy.com과 ODP가 각각 소스 온톨로지 모델로 사용될 경우, Amazon.com 온톨로지는 타겟 온톨로지 모델의 역할로써 재사용이 가능하다. 이러한 방식으로 소스와 타겟 온톨로지 모델에 대한 경우의 수를 생각해 보면, 6가지의 경우의 수가 나타나게 되며, 이러한 경우의 수를 PROMPT의 경우에도 동일하게 적용시켜 본다면 결과적으로 총 12개의 경우의 수가 도출된다. 이렇게 도출된 경우의 수를 바탕으로 우리는 각각의 경우의 수에 대한 결과 값들을 “제 1종 오류”, 그리고 “제 2종 오류”로 구분하여 각 경우에 대한 Precision을 산출해 보았다. 그럼 지금부터 이를 통해 산출된 결과 값들에 대해 분석해보도록 하자.

먼저 아래 <표 5>는 소스 온톨로지와 타겟 온톨로지 간에 동일한 의미를 지닌 노드가 존재하는 경우, 즉, Sensitivity에 대한 결과를 나타내고 있

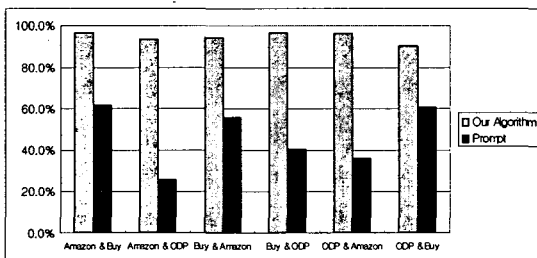
으며, <표 6>은 두 온톨로지 간에 동일한 의미를 지닌 노드가 존재하지 않는 경우에 해당하는 Specificity에 대한 결과를 나타낸다. 또한, [그림 6]은 각 <표 5>에 대한 결과를 그래프 형식으로 나타내고 있다.

<표 5> Performance Results terms of Sensitivity

Source \ Target	ODP	Amazon.com	Buy.com
ODP		96.4%	90.5%
Amazon.com	93.5%		96.9%
Buy.com	96.7%	94.4%	

<표 6> Performance Results terms of Specificity

Source \ Target	ODP	Amazon.com	Buy.com
ODP		51.9%	69.6%
Amazon.com	82.4%		56.4%
Buy.com	76.8%	64.4%	



[그림 6] Comparison of Precision

위의 표에서 살펴볼 수 있듯이 우리 방법론의 경우, Sensitivity에 대한 경우가 Specificity에 대한 경우보다 좋은 결과를 나타내었다. Sensitivity와 Specificity의 각 결과 값들에 대한 평균치를 살펴보면, 우리 방법론의 경우, 각각 94.7%와 66.9%로 나타났으며, 동일한 방식으로 실험 해본 결과, PROMPT는 Sensitivity가 46.8% 그리고 Specificity가 89.6%인 것으로 확인되었다.

PROMPT의 경우, 사용자 판단 하에 추가적으로 매핑이 필요하다고 생각되는 어휘들에 대해서는 사용자가 직접 매핑을 수행할 수 있도록 해주는 기능을 제공하고 있기 때문에, 실제적으로 많은 매핑이 이루어지지 않는 것이다. 따라서 Specificity에 대한 결과는 우리 방법론에 비해 비교적 좋은 결과 값을 지닌 것으로 나타났으나, Sensitivity의 경우는 그렇지 못했다. 즉, 자동화라는 측면에서 생각해 볼 때, 우리 방법론은 적극적인 추천을 통해 결과 값을 되돌려주고 있기 때문에 PROMPT 비해 많은 매핑을 수행할 수 있었으며, Precision 역시 상대적으로 우수함을 입증해 보였다. 또한, 이러한 적극적인 추천은 사용자가 요청한 정보에 대해 적극적인 응답으로 이어질 수 있기 때문에 쇼핑물의 검색이라는 측면에서도 효율성을 언급해 볼 수 있겠다.

PROMPT의 경우는 동일한 한가지 노드에 대해서 여러 가지의 매핑을 수행함으로써 방법론상의 약점을 드러내기도 했다. 예를 들어 “Game”이라는 한 상품에 대해서 PROMPT는 “Gadgets”, “Gamecube”, “PC Games”, “Games” 등 총 4개의 상품에 대한 매핑을 수행함으로써 매핑 수행 능력 측면에 있어 오류를 범하고 있는 것으로 확인되었다. 앞선 [그림 6]상에 나타난 결과는 이러한 사항들을 고려하지 않은 값이며, 이러한 부분들을 고려 대상에 포함시킬 경우, Precision은 더욱 더 낮아 질 위험이 있다. 이에 반해, 우리가 제안한 방법론 역시 문제점을 지적해 볼 수 있다. 우리 방법론의 경우, 전 자동화(Fully Automatic) 방식을 바탕으로 존재하는 모든 노드들에 대한 적극적인 매핑을 수행하고 있기 때문에 종종 잘못된 매핑 결과가 나타나기도 하였다. 따라서 추후 이러한 부분들이 보완된다면 현재 보다 견고한 방법론을 구상해 볼 수 있을 것으로 기대된다.

## 5. 결론

본 연구를 통해 우리는 서로 이질적인 분류 체계로 이루어진 온톨로지들 간에 보다 효율적인 매핑을 지원해 줄 수 있는 방법론을 제안해 보았으며, 이와 함께 동일한 조건하에서 PROMPT와의 비교 실험도 수행하였다. 또한, 대표적인 쇼핑물 사이트에 대한 상품 정보 카테고리를 온톨로지화시켜 실험 데이터를 구성함으로써 실험의 객관성 확보는 물론, 데이터 일관성도 유지하였다. 이러한 결과를 통해 본 연구가 가지는 의의는 아래와 같다.

첫째, 본 연구에서는 온톨로지 매핑 이전에 어휘의 의미를 파악하기 위한 과정으로 Word Sense Disambiguation에 관한 방법론을 제안함으로써, 보다 효과적인 온톨로지 매핑이 이루어 질 수 있도록 지원하는 방법론을 고안하였다. PROMPT를 포함한 기존의 온톨로지 매핑 방법론들의 경우, 온톨로지 상의 어휘 자체나 온톨로지 구조 자체에 대한 사항들을 토대로 매핑을 수행하여 왔기 때문에 보다 정확한 매핑을 수행하는데 있어 한계가 있었던 것이 사실이다. 하지만, 앞서 실험결과를 통해 이미 살펴보았듯이, 본 연구에서는 단순한 문자열 매칭(matching)방식 보다는 상품을 표현하는 어휘 자체가 지닌 의미를 중심으로 서로 간의 매핑을 수행하고 있기 때문에 매핑 수행 능력 측면에 있어 기존 방법론들보다 우수함을 입증해 보일 수 있었다. 이와 함께 PROMPT와의 비교 실험 결과, Sensitivity에 대한 매핑 수행 능력에 있어 우리 방법론이 PROMPT에 비해 우수하다는 사실도 확인해 볼 수 있었다.

둘째, 본 연구에서는 전자동화(fully automatic)된 방식을 바탕으로 온톨로지 매핑을 수행함으로써, 온톨로지를 통한 쇼핑물 검색이라는 본래 취지에

있어 효율성을 극대화시키고자 하였다. iPROMPT에서도 확인해 보았듯이, 현존하는 대부분의 방법론들의 경우, 반자동화(semi-automatic)된 방식으로 온톨로지 매핑을 수행하고 있기 때문에, 이를 쇼핑물 환경에 적용시키고자 하는 경우에는 한계가 있었다. 하지만, 본 연구에서는 전자동화된 방식을 바탕으로 매핑 과정에 있어 사용자의 역할을 최소화시킴으로써 사용자들이 보다 편리하고 효율적으로 상품 검색을 수행할 수 있도록 지원하였다.

본 연구에서는 기존연구에 대한 심도 있는 분석을 통해 기존연구가 지닌 한계점을 보완하고자 하였으며, 이와 같은 보완작업을 통해 온톨로지 매핑이라는 분야를 쇼핑물이라는 현실 세계에 적용시켜 보고자 하였다. 본 연구에 대한 한계점을 분석해 보면 다음과 같다.

첫째, 2개 이상의 단어가 모여 형성된 합성어들에 관한 처리 방법을 표준화 시킬 필요가 있다. 예를 들어, “Musical Instrument”나 “Gourmet Food” 등과 같은 어휘의 경우, 현재 방법론상에서는 해당 상품의 주된 의미를 표현하는 어휘에 대한 판단과정 없이 의미 파악을 수행하기 때문에 종종 잘못된 의미를 선택하는 경우가 있다. 앞서 예로 든 “Gourmet Food”와 같은 경우, 본래 상품에 대한 의미를 담고 있는 어휘는 “Food”이며, “Gourmet”은 단지 “Food”를 수식해 주기 위한 어휘로써 사용이 되고 있다. 따라서 이러한 경우, 상품을 표현하는 어휘 가운데 해당 상품에 대한 주된 의미를 지니고 있는 어휘를 판단해 이에 대한 의미를 파악할 수 있어야 한다. 따라서 추후 본 연구에서는 이러한 합성어들을 처리하기 위한 방안이 마련되어야 할 것이다.

둘째, 4.3절의 실험을 통해 살펴보았듯이, 본 연구에서 제안한 방법론의 경우, Sensitivity에 대한

결과는 상대적으로 우수하였으나, Specificity에 대한 결과는 그렇지 못했다. 이는 매핑 수행 과정에서 일부 어휘들에 대하여 잘못된 부분 매핑 (partial mapping)이 수행되었기 때문인 것으로 판단된다. 따라서 추후 Specificity 측면에 대한 보완 작업이 이루어져야 할 것이다.

셋째, 본 연구는 Amazon.com, Buy.com 등과 같은 해외 쇼핑몰들로부터 제공 되는 상품 검색에 주로 초점을 맞추어 왔다. 이는 현재 본 연구에서 어휘의 의미 파악 및 확장 과정에 사용된 WordNet이 영문화 버전이기 때문에 한글에 대한 처리를 함에 있어 한계가 있었기 때문이다. 그러나 현재 국내 대학 으로부터 WordNet의 한글화 버전이 구축 완료 되어 상용화되고 있는 것으로 확인되었다. 따라서 이를 이용해 추후에는 국내 쇼핑몰 상에서도 효과적인 상품 검색이 이루어질 수 있도록 방법론을 마련해 볼 수 있을 것이다.

## 참고문헌

- [1] A. Gal and G. Modica and H. Jamil. "Improving Web Search with Automatic Ontology Mapping", Mississippi State Univ, 2003.
- [2] Benetti, H., D. Beneventano, S. Bergamaschi, F. Guerra, and M. Vincini, "An Information Integration Framework for E-Commerce", *IEEE Intelligent Systems*, Vol.17, No.1 (2002).
- [3] Boston Consulting Group David Pecaut, Michael Silverstein, Peter Stanger "Winning the Online Consumer Insights into Online Consumer Behavior", 2000, avail. From <<http://www.bcg.com>>
- [4] Decker, S., M. Erdmann, D. Fensel, and R. Studer. "ONTOBROKER : Ontology Based Access to Distributed and Semi-Structured Information", *Database Semantics : Semantic Issues in Multimedia Systems*, New Zealand, 1999.
- [5] Ehrig, M., and Y. Sure, "Ontology Mapping - An Integrated Approach", *Lecture Notes in Computer Science*, No.3053(2004), 76-91.
- [6] Fensel, D., et al., "Ontobroker : Semantic-Based Access to Information Sources at the WWW", *Proceedings of the World Conference on the WWW and Internet (WebNet 99)*, Honolulu, Hawaii, USA, 1999.
- [7] Guarino, N., et al., "OntoSeek : Content-based Access to the Web", *IEEE Intelligent Systems*, Vol.14, No.3(1999).
- [8] Jiawei Han and Micheline Kamber, "Data Mining : Concepts and Techniques", (2001), 325-326.
- [9] Kalfoglou, Y. and M. Schorelmmmer, "Ontology mapping: the state of the art", *The Knowledge engineering review*, Vol.18. No.1(2003), 1-32.
- [10] Kerschberg, L., W. Kim, and A. Scime, "A Personalizable Agent for Semantic Taxonomy Based Web Search", *Lecture Notes in Computer Science*, Vol.2564 (2003), 3-34.
- [11] Lawrence, S. and C. L. Giles, "Accessibility of Information on the Web", *Nature*, Vol. 400(1999), 107-109.
- [12] Lawrence, S. and C. L. Giles, "Context and Page Analysis for Improved Web Search", *IEEE Internet Computing*, Vol.2, No.4 (1998), 38-46.
- [13] Marc Ehrig and Steffen Sraab, "QOM : Quick Ontology Mapping", *Institute AIFB*,

- Karlsruhe Univ, 2004.
- [14] Miller, G. A., "WordNet a Lexical Database for English", *Communications of the ACM*, Vol.38, No.11(1995), 39-41.
- [15] Minsky, M., "A framework for representing knowledge", In J. Haugeland, editor, *Mind Design*, The MIT Press, 1981.
- [16] Natalya F. N., "Semantic Integration : A Survey of Ontology Based Approaches", *Stanford Medical Informatics*, Stanford Univ, 2004.
- [17] Noy, N. F. and M. A. Musen, "The PROMPT Suite: Interactive Tools for Ontology Merging and Mapping", *International Journal of Human-Computer Studies*, Vol.59, No.6(2003), 983-1024.
- [18] O'Keefe, R. M. and T. McEachern, "Web Based Customer Decision Support Systems", *Communications of the ACM*, Vol. 41(1998), 71-78.
- [19] Philip Resnik, "Semantic Similarity in a Taxonomy : An Information Based Measure and its Application to Problems of Ambiguity in Natural Language", Dept. of Linguistics and Institute for Advanced Computer Studies, MarylandUniv, 1999.
- [20] Scime, A. and L. Kerschberg, "WebSifter : An Ontology-Based Personalizable Search Agent for the Web", *International Conference on Digital Libraries: Research and Practice*, Kyoto Japan, 2000.
- [21] Siddharth Patwardhan, "Incorporating Dictionary and Corpus Information into a Context Vector Measure of Semantic Relatedness," Minnesota Univ, 2003.
- [22]mith, M. K., C. Welty, and D. McGuinness, *OWL Web Ontology Language Guide*, W3C Recommendation. <<http://www.w3.org/TR/owl-guide/>>, 2004.
- [23] S, Castano, A. Ferrara, and S. Montanelli. "H-MATCH : an Algorithm for Dynamically Matching Ontologies in Peer-based Systems", *degli Studi di Milano Univ*, 2003.
- [24] Veltman, K. H., "Syntactic and Semantic Interoperability: New Approaches to Knowledge and the Semantic Web", *New Review of Information Networking*, Vol.7 (2001), 159-184.
- [25] Wooju Kim, Daewoo Choi, Sangun Park, "Agent Based Intelligent Search Framework for Product Information using Ontology Mapping", 2005.
- [26] Wooju Kim, Daewoo Choi, Sangun Park, "Product Information Meta-Search Framework for Electronic Commerce through Ontology Mapping", 2005.

Abstract

## An Ontology-Driven Mapping Algorithm between Heterogeneous Product Classification Taxonomies

Wooju Kim\* · Namhyuk Choi\* · Daewoo Choi\*\*

The Semantic Web and its related technologies have been opening the era of information sharing via the Web. There are, however, several huddles still to overcome in the new era, and one of the major huddles is the issue of information integration, unless a single unified and huge ontology could be built and used which could address everything in the world. Particularly in the e-business area, the problem of information integration is of a great concern for product search and comparison at various Internet shopping sites and e-marketplaces. To overcome this problem, we proposed an ontology-driven mapping algorithm between heterogeneous product classification and description frameworks. We also performed a comparative evaluation of the proposed mapping algorithm against a well-known ontology mapping tool, PROMPT.

**Key words** : E-Commerce, Information Integration, Ontology, Semantic Web, Semantic Similarity

---

\* School of Industrial and Information Engineering, Yonsei University

\*\* E-business Research Center, Yonsei University