온톨로지 기반 지능형 규칙 구성요소 추출에 관한 연구

(Ontology - Based Intelligent Rule Components Extraction)

김우주 [1], 채상용 [2], 박상언 [3]

[1,2] 연세대학대학교 정보산업공학과
서울 서대문 신촌동 134, 120 - 749
Tel: + 82 - 2 - 2123 - 7754, Fax: + 82 - 2 - 2260 - 8824, E - mail:
wkim@yonsei.ac.kr,slashap@yonsei.ac.kr
[3] 한국과학기술원 테크노경영대학원
서울 동대문구 청량리동 207 - 43

W. Kim[1], Sang - Yong,Chae[2], Sang - Un,Park[3]

Tel: + 82 - 2 - 958 - 3622, Fax: + 82 - 2 - 960 - 2102, E - mail: mascon99@kgsm.kaist.ac.kr
[1, 2] Department of Information Industrial Engineering College of Engineering, Yonsei University
134, Shinchonl - Dong, Soedaemun - Gu, Seoul, 120 - 749, Korea
Tel: + 82 - 2 - 2123 - 7754, Fax: + 82 - 2 - 2260 - 8824,
E - mail:wkim@yonsei.ac.kr, slashap@yonsei.ac.kr
[3] Maagement Engineering, KAIST Graduate School of Management
207 - 43 Cheongryangri - dong Dongdaemun - gu Seoul, Korea 13. - 012
Tel: + 82 - 2 - 958 - 3622, Fax: + 82 - 2 - 960 - 2102, E - mail: mascon99@kgsm.kaist.ac.kr

Abstract

시맨틱 웹 관련연구가 증가함에 따라 하나의 관련분야로 규칙기반 시스템 등의 지능적인 웹 환경에 대한 기대 역시 커지고 있다. 하지만 규칙기반 시스템을 활용하기에는 아직도 규칙습득이 많은 제약이 되고 있다. 규칙습득은 웹으로부터 필요한 규칙을 습득하는 일련의 방법인데, 이러한 규칙을 습득하기 위해서는 규칙구성요소를 먼저 식별해야만 한다. 그러나 이러한 규칙을 식별하는 작업은 대부분 지식관리자의 수작업에 의해 이루어지고 있다.

본 연구의 목적은 웹으로부터 규칙구성요소 식별을 최대한 자동화하고 지식관리자의 수작업을 최소화함으로써 그 부담을 줄여 주는 데 있다. 이러한 방법으로는 온톨로지를 근간으로 하여 웹 페이지와의 문자열 비교, 이러한 비교의 한계를 극복하기 위한 확장등의 방법이 있다.

첫 번째 방법은 온톨로지 기반으로 규칙식별 할 웹 페이지와 비교를 통해 지식관리자의 규칙식별 과정을 최대한 자동화하여 주는 것이다. 여기서 만약 현재 규칙을 식별하고자 하는 웹 사이트와 유사한 시스템의 규칙들을 활용하여 일반화 된 온톨로지가 구축되었다면, 이 온톨로지를 기반으로 규칙을 식별하고자 하는 웹 사이트와의 비교를 통해 규칙구성요소를 자동화하여 추출 할 수 있다. 이러한 온톨로지를 기반으로 규칙을 식별하기 위해서는 문자열 비교 기법을 사용하게 된다. 하지만 단순한 문자열 비교 기법만으로는 규칙을 식별하는 데에 자연어 처리에 대한 한계가 있다. 이를 극복하기 위해 다음의 두 번째 방법을 사용하고자 한다.

두 번째 방법은 정형화 되지 않은 정보들을 확장하여 사용하는 것이다. 우선 찾고자 하는 단어들의 원형을 찾기 위한 스테밍 알고리즘 기법,

WordNet 을 이용하여 동의어· 유의어등으로 확장을 하는 WordNet Expansion 기법, 의미 유사도를 측정하기 위한 방법인 Semantic Similarity Measure 등을 단계적으로 수행하여 자동화되고 정확한 규칙식별을 하고자 한다.

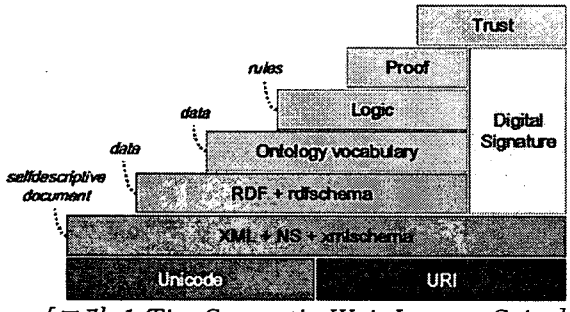
이러한 방법들의 조합으로 인하여 규칙구성요소 추출이 되지 않을 후보 단어들의 수를 줄여서 보다 더 정확하고, 지능적인 규칙구성요소 추출 방법론을 제시하고 구현하여 지식관리자의 규칙습득에 대한 부담을 줄여 주고자 한다.

Keywords:

Information Extraction, Information Integration, Semantic Web, Rule Acquiring.

1. 서 론

오늘날의 웹 안에는 무수히 많은 지식들이 분포되어 있으며 이러한 지식들은 단순한 데이터로부터 시작해서, 다양한 실시간 정보, 여러 가지 개념, 법률, 문제를 해결하기 위한 규칙에 이르기 까지 그 종류도 매우 다양하다. 따라서 이러한 웹 상의 지식을 효율적으로 습득하고 활용하기 위한 연구들이 활발하게 진행되고 있다. 그러나 대부분의 연구들이 온톨로지 수준의 사실이나 개념을 추출하는 데 초점을 두고 있으며, 규칙 기반 시스템이나 인텔리전트 에이전트에서 사용되는 추론 가능한 규칙 구성요소를 추출하는 연구는 많지 않다.



[그림 1 The Semantic Web Layers Cake]

웹으로부터 규칙 구성요소를 추출하여 지식을 추출하는 연구는 온톨로지를 활용하여 좀 더 쉽고 효율적인 지식 습득 방법론을 제안하고자 노력하고 있다. 따라서 이러한 온톨로지의 활용이 규칙 습득에도 도움을 줄 수 있을 것으로 기대 된다. 시맨틱 웹의 계층구조(*[그림 1]*)를 살펴보면 규칙은 온톨로지 위에 구축되기 때문에 규칙을 습득하고자 하는 도메인에 온톨로지가 이미 존재하고 있는 경우, 이러한 온톨로지는 규칙습득에 매우 유용하게 사용될 수 있다. 또한 웹 페이지로부터 규칙을 습득하는 방법은 순차적인 방법이 필요 하다. 이러한 연구 방법으로는 XRML 방법론 [Kang and Lee, 2005], [Park and Lee, 2006]이 있다. 이 방법은 우선 규칙 식별 요소인 변수(variables)와 변수값(values)을 추출해 오고, 추출된 규칙식별 요소들을 결합하고 조합하여 규칙을 구성하게 된다. 이러한 순차적인 방법을 통하여 본 연구에서는 규칙 구성요소를 자동적으로 추출해오는 방법론을 제안하고 구현하고자 한다.

1.1 연구의 목적

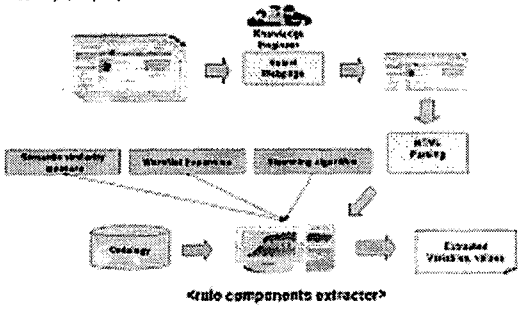
웹으로부터 지식이나 규칙식별 구성요소를 직접적으로 추출해 내는 방법은 매우 복잡하고 어렵다. 우선, 웹에 수없이 산재 되어 있는 규칙들은 자연어로 되어 있어 매우 복잡한 구조로 구성이 되어 있다. 또한, 이렇게 자연어로 되어 있는 규칙의 패턴을 식별하기란 의미 매칭등의 부분에서 매우 어려운 일이다. 이러한 이유에서 규칙식별을 할 때 순차적인 방법이 필요한 것이다. 순차적인 방법을 보면. 첫 번째로 규칙을 구성하고 있는 규칙구성요소(variables, values)를 식별해 와야 한다. 다음으로는 추출된 규칙구성요소의 조합을 통하여 규칙을 식별하는 방법이다. 이러한 방법론으로는 기존의 연구로 XRML 방법론 [Kang and Lee, 2005], [Park and Lee, 2006]이 있다. 이 연구에서 규칙식별을 위한 첫 번째 단계인 규칙구성요소 식별단계에서는 문자열 비교와 지식관리자가 수작업으로 규칙구성요소를 식별을 하게 되어 있다. 본 연구에서는 이러한 규칙구성요소의 식별을 자동화 시켜주어 지식관리자의 부담을 줄여주고, 찾아내지 못했던 규칙구성요소들을 자동적으로 찾아 주고자 하는 방법을 제안하였다.

1.2 문제점 및 해결방안

앞서 언급한 대로 단순한 문자열 비교만으로는 규칙구성요소 추출을 자동적으로 하는 것이 상당히 어려운 문제이다. 이에 대한 문제점으로는 유의어·동의어에 관한 문제(기준 온톨로지에 " book" 라는 인스턴스가 존재하고, 대상이 되는 웹 페이지에 그 유의어인 " script" 나 " volume" 이 존재 하면 각 단어들은 유의어 관계이어서 의미가 통하지만 단순한 문자열 비교로만으로는 추출이 되지 않음.), 단어들간에 현재형·과거형, 단수형·복수형등에 관한 단어 원형에 관한 문제(기준 온톨로지에 " book" 이 존재하고 있고, 대상이 되는 웹 페이지에 이 단어의 복수형인" books" 가 존재하면 단순한 문자열

비교만으로는 추출이 되지 않음), 의미적 유사도 문제점(기준 온톨로지에 " country" 가 존재하고 대상이 되는 웹 페이지에 " region" 이 존재하면 유사한 의미를 지니고 있는 두 단어이지만 단순한 문자열 비교만으로는 추출이 되지 않음)등이 있다.

이러한 문제점등을 해결하기 위하여 워드넷 확장, 스테밍 알고리즘의 적용, 의미유사도 측정을 수행하고자 한다. 다음의 그림은 이러한 방법들을 적용하여 본 논문에서 제안하는 자동적으로 규칙 구성요소를 추출해 오는 시스템인 <Rule Components Extractor>의 전체 개념도를 도식화해 놓은 것이다.



[그림 2]Rule Components Extractor

2. 관련 연구

본 장에서는 온톨로지의 활용, 스테밍 알고리즘의 활용, WordNet 확장, 의미유사도 측정등에 관한 연구를 조사하고 본 연구의 접근방법과 비교하여 설명하겠다.

2.1 온톨로지의 활용

본 연구에서 기술적인 차별성을 두기 위해서 시맨틱 웹에서 주요 근간을 이루는 온톨로지를 활용하였다.

오늘날 웹 자원이 폭발적으로 증가함에 따라, 정보의 양이 인간이 처리할 수 있는 한계를 초과함으로써, 인간 대신 기계가 웹 자원을 스스로 처리할 수 있도록 웹에 의미를 부여하는 시맨틱 웹 분야 연구에 대한 수요도 기하급수적으로 증가하고 있다.

본 연구에서는 온톨로지의 형태로 OWL 형태의 온톨로지를 채택하였고, 이의 활용을 위하여 Jena api 를 활용 하였다.

2.1.1 Jena - A Semantic Web Framework for Java

Jena 는 HP 연구소에서 만들어진 시멘틱 웹 프레임워크로 RDF, RDFS 및 OWL 등을 구현하기에 적당한 환경을 제공한다. 오픈 소스로 많은 사람들이 몇 가지 원칙하에서 자유롭게 소스의 수정 및 재 배포가 가능하다. 현재 안정화된 버전은 2.1 이고, 베타판으로 2.2 까지 나와 있는 상태이다.

RDF api 는 RDF 형태를 다루기 위한 다양한 API 들을 제공한다. 여기서는 RDF 의 기본 구조인 트리플 구조를 모델링한 Statement 클래스나 Model, Subject, Object, Property 각각을 독립된 클래스로 제공하여 RDF 모델을 직관적으로 생성 및 변경하는 코딩을 할 수 있다. 물론 이러한 api 에도 추론 개념이 포함되어 있어서 새로운 사실에 대한 정보도 유출해 낼 수 있다.

또한 이들 api 에는 RDF 모델을 RDF/XML 이나 N3 혹은 N-Triples 형태로 쓰거나 읽어 들이는 방법도 제공한다. 아주 간단한 메소드 호출 만으로 이러한 결과를 얻을 수 있게 되어 있다.

Jena 는 기본적으로 RDF 형태에 가장 강점을 보이고 여기에 OWL 과 같은 온톨로지 언어에 대한 지원으로 그 기능이 확장 되고 있는 방식을 채택하고 있다. 특히 현재 시맨틱 웹 추론 엔진 부분에서 가장 활발한 업데이트와 다양한 기능을 제공하고 있어 가장 많이 사용되고 있다. 이러한 차원에서 OWL api 는 RDF api 로 OWL 모델을 구현함에 비해 부가적인 편리함을 제공하고 다양한 Property 도 제공한다. 따라서 보다 복잡한 추론 형태를 제공할 수 있다. 또한 온톨로지 API 는 모델의 유효성 여부나 추론 규칙 추적과 같은 부가적인 기능도 제공하고 있다.

RDQL 은 RDF Data Query Language 의 약자로서 RDF 모델상에서 조건에 맞는(기본적으로 트리플 형태로 되어 있는) Subject, Object 혹은 Property 를 질의를 통해서 알아 낼 수 있다. 물론 복잡한 형태의 조건을 주어서 보다 정확한 결과를 얻게 할 수 있다.

Jena 는 자체적으로 위와 같은 기능을 제공하고 필요한 기능들에서는 버전업을 통해서 기능을 지원하는 현재도 발전하고 있는 프레임워크이다. Jena 는 자바로 구현되어 있어서 자바언어가 갖는 모든 잇점을 가지고 기본적인 RDF 파서도 제공하며 내부적인 추론은 그래프 매칭 방법을 이용하여 접근한다.

2.2 스테밍 알고리즘의 활용

스테밍 알고리즘이란 정보검색의 성능을 향상시키는 한가지 기술은 탐색자에게 탐색용어의 어형론적인 변형을 찾는 방법을 제공하는 것이다. 가령 탐색자가 질의어 일부로서 용어 " stemming" 을 입력한다면 그 조사자는 stem 이나 stemmed 와 같은 변형에 관심을 갖는다. 흔히 융합이나 결합의 행위를 의미하는 합성이라는 용어를 어형론적인 용어변형처리의 일반적인 용어로서 사용한다. 함성은 일반적인 몇 가지

수식을 사용하여 수동으로 처리하거나, 스테머라 불리는 프로그램을 통해 자동으로 처리할 수 있다. 용어대신 어간을 저장함으로써 정보 검색시에 완전한 용어와 일치 시킬 수 있다. 본 연구에서는 스테밍 알고리즘의 한 종류인 Porter 의 스테밍 알고리즘을 사용하였다.

2.3 WordNet 확장

워드넷은 영어를 기반으로 한 어휘적 지식을 모델링하기 위해 프린스톤 대학에서 진행하고 있는 프로젝트이다. 워드넷 시스템은 단어 사이의 의미론적 또는 사용 패턴에 관련된 정보로 단어 간의 연관성을 구축한 데이터베이스이라고 할 수 있다. 따라서 워드넷은 온톨로지 개념들 사이의 연관성을 알아내기 위해서 사용될 수 있다. 워드넷의 기본적 구조는 의미적으로 동일한 단어 리스트(이들 리스트 안에서 동의어 관계이다)를 가진 신셋(synset)이라는 논리적 그룹들과 이러한 신셋들 사이의 관계를 정의한 의미적 관계로 구성된다. 의미적 관계는 다음과 같은 타입들을 가질 수 있다.

- 상위어(Hyponym)와 하위어(Hypernym)
- 부분어(Meronym)와 전체어(Holonym)

워드넷의 명사 부분에서는 최상위 상위어로 *entity* 를 가지고 있으며 의미에 따라 이를 확장함으로써 하위어를 형성할 수 있다. 그러므로 워드넷도 또한 개념 어휘를 분류으로 정의하여 계층적 구조를 이룬 일종의 온톨로지이라고 할 수 있고 온톨로지 개념들 사이의 의미 유사도를 제공할 수 있는 표준 지침으로 사용될 수 있다.

본 논문에서 사용되는 워드넷은 워드넷 용어에 대해서 동사, 부사 그리고 형용사는 고려하지 않으며 오직 명사 계층만 고려한다. 그리고 개념들 사이의 의미 유사도를 측정하기 위해서 의미적 관계의 네 가지 타입 중에서 상위어와 하위어 관계만 이용한다.

2.4 의미유사도 측정

워드넷은 의미로 이루어진 데이터베이스이다. 이러한 워드넷을 사용하여 각 신셋들간의 의미 유사도를 측정을 할 수가 있다. 의미유사도란 각 단어들간의 의미가 얼마나 유사한가를 워드넷에 나타나 있는 상위어, 하위어개념, 부분어, 전체어 개념과 각 단어의 신셋들간의 노드상의 거리를 측정하여 수식으로 표현하고 나타내어 주는 방법이다. 이러한 의미유사도의 측정방법에는 단순히 노드들간의 거리를 측정해 표현해주는 pathlength measure, 정보내용(information contents)를 가지고 개념들사이의 의미유사도를 측정하는 Resnik measure, 그밖에 Lin measure, Jiang_Conrath measure, Leacok Chodorow measure 등이 있다.

본 연구에서는 본 연구의 목적과 맞게 새로운 의미유사도 측정방안을 제안하고자 한다. 이는 두 단어들이 상위어, 하위어 관계이 있으면 그 pathlength 를 구하여 의미유사도를 측정하고자 한다.

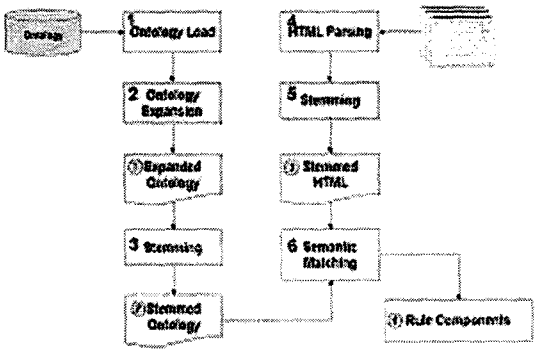
3. 규칙구성요소 식별 과정

본 장에서는 규칙구성요소 식별시 온톨로지의 역할 및 온톨로지를 활용한 규칙구성요소 식별과 관련된 접근 방법을 설명하고자 한다.

```
<Rule rdf:ID="Rule7">
    <IF_variables>
      <Variable rdf:ID="Purchased_from"/>
    </IF_variables>
    <IF_values>
      <Value rdf:ID="Imaginarium.com">
        <totalCount
rdf:datatype="http://www.w3.org/
2001/XMLSchema#int"
          >1</totalCount>
      </Value>
    </IF_values>
    <THEN_values>
      <Value rdf:ID="Amazon.com">
        <totalCount
rdf:datatype="http://www.w3.org/
2001/XMLSchema#int"
          >1</totalCount>
      </Value>
    </THEN_values>
```

[그림 3]Amazon.com 에서 추출한 기준 온톨로지

규칙구성요소 식별이란 규칙을 식별하는 순차적인 방법에서 첫 번째 단계이다. 이는 지식관리자가 규칙을 이루고 있는 기본 규칙구성요소들을 웹 페이지로부터 식별하는 과정이다. 규칙 구성요소 식별과 관련된 작업으로 웹 페이지로부터 규칙을 구성하고 있는 변수(variables)나 변수값(values)을 찾아내어서 이를 조합하여 규칙을 만들어 낸다. 이때, 현재까지 규칙구성요소를 식별해 올 때 단순히 문자열 비교로 이루어져 있던 식별과정을 2 장에서 설명하였던 관련연구들(owl 의 사용, 워드넷 확장, 스테밍 알고리즘 사용, 의미유사도 측정 방법)을 토대로 이를 자동화 시켜주고자 한다. 이를 위해서 다음과 같은 방법을 사용하였다.



[그림 4]규칙구성요소 식별 프로세스

위의 그림은 규칙구성요소 식별을 위한 전체의 과정을 나타낸 것이다. 것이다. 다음으로 단계적으로 이 과정에 대하여 설명을 하겠다. 우선 기준 온톨로지를 메모리상에 로딩 시킨 후 이 온톨로지를 워드넷을 통하여 유의어 등으로 확장을 시킨다.([그림 4]의 1,2 번 과정, 결과물 : ①) 그 후에 스테밍 알고리즘을 수행하면 스테밍된 온톨로지가 생성이 된다.([그림 4]의 3 번과정, 결과물 : ②) 다음으로는 웹 페이지에서 html 파일을 읽어 들여 스테밍 과정을 수행([그림 4]의 4,5 번 과정, 결과물 : ③)한 후 의미유사도 측정을 하면([그림 4]의 6 번과정, 결과물 : ④) 규칙 구성요소(variables, values)를 식별해 온다.

3.1 규칙구성요소 식별에서 온톨로지의 역할
<[그림 4]의 1 번과정>

우선 규칙을 식별하기 위하여서는 기준이 되는 기준 온톨로지가 필요하다. 이는 기존연구인 XRML 방법론 [Kang and Lee, 2005], [Park and Lee, 2006]에서 Amazon.com 에 관하여 추출한 규칙구성요소의 온톨로지를 이용하여 유사한 도메인인 웹 페이지(ex. Baens&Nobles.com)로부터 규칙구성요소를 추출하고자 한다.

이러한 온톨로지를 가지고 자동으로 규칙구성요소인 변수(variables)와 변수값(values)를 추출해 오기 위해서 Jena api 를 사용하였다. 온톨로지에서 S - P - O 관계를 추출해내어 규칙구성요소를 추출해 내기 위한 첫 번째 단계를 수행하였다. 추출된 온톨로지의 인스턴스들을 가지고 다음단계들을 수행해 나간다. 이 과정에서는 작성되어 있는 온톨로지를 가지고 그 각각의 인스턴스들을 추출해 온다.

3.2 워드넷 통한 온톨로지 확장
<[그림 4]의 2 번과정, 결과값 : ①>

단어 사이의 의미론적 또는 사용 패턴에 관련된 정보로 단어 간의 연관성을 구축한 데이터베이스이인 워드넷을 통하여 각 온톨로지의 인스턴스들에 관한 동의어· 유의어들을 확장해 준다. 이는 웹 페이지에 존재하는 수 많은 규칙구성요소의 후보들 중에 온톨로지에는 나타나 있지만 웹 페이지에 그 규칙 구성요소들의 후보가 온톨로지 인스턴스들의 유의어나 동의어등으로 존재 할때, 규칙 구성요소후보자들을 규칙 구성요소로 추출해 을 수가 없다. 이를 보완하기 위해서 워드넷을 가지고 각 인스턴스들을 확장하여 규칙구성요소 추출을 하고자 하는 것이다. 이를 위해서 워드넷 2.1 버전을 사용하였고, 이 워드넷을 유용하게 사용하기 위하여 JWNL api 를 사용하였다.

추출한 온톨로지 인스턴스들을 워드넷을 통하여 확장을 하면 다음과 같은 결과 값이 나타난다.

온톨로지 인스턴스	워드넷 확장을 통한 온톨로지 인스턴스 결과 값
cd	cadmium, Cd, atomic_number_48
	candle, candela, cd, standard_candle
	certificate_of_deposit, CD
	compact_disk, compact_disc, CD
book	book
	book, volume
	ledger,leger, account_book, book_of_account, book
	book
	record, record_book, book
	book
	script, book, playscript
	book, rule_book
	Koran, Quran, al - Qur'an, Book
	Bible,Christian_Bible, Book,Good_Book,Holy_Scripture, Holy_Writ, Scripture, Word_of_God, Word
israel	Israel, State_of_Israel, Yisrael, Zion, Sion
	Israel

[그림 5 워드넷 확장 결과값의 예시]

온톨로지 인스턴스들을 워드넷 확장을 통한 결과값을 가지고 웹 페이지와 비교를 하면 동의어 유의어 문제가 해결이 된다. 그럼 다음단계를 살펴보겠다.

3.3 스테밍 알고리즘을 통한 어근 찾기
<[그림 4]의 3 번 과정, 결과값 : ②>

확장된 온톨로지를 바탕으로 웹 페이지와 비교를 통하여 규칙 구성요소를 추출해 오는데에도 문제가 생긴다. 온톨로지 인스턴스들은 원형으로 이루어져 있는데, 웹 페이지에 나타나 있는 단어들은 단어의 원형만으로 이루어지지 않고 복수형, 단수형 또는 과거형, 현재형등으로 이루어져 있다. 이 문제의 해결을 위하여 탐색용어의 어형론적인 변형을 찾는 방법을 제공하는 스테밍 알고리즘을 사용하였다. 많은 스테밍 알고리즘들 중에 사용한 스테밍 알고리즘은 Porter 의 스테밍 알고리즘이다. Porter 의 스테밍 알고리즘은 가장 널리 사용되는 스테밍 알고리즘이며, 다양한 규칙들을 이용하여 접두사는 제거하지 않고, 접미사만을 제거하거나 새로운 스트링으로 대치한다. 스테밍한 결과값의 예시를 들어 보겠다.

```
Vector stem_return(String return_value)
{

//  IF_values_return IVo = new IF_values_return();
//  String stem_IVo = IVo.onto_str();
    String bbb = "";
    Vector a = new Vector();

    Stem_return s = new Stem_return();
    bbb = s.stemming(return_value);//스테밍 결과 값 어스트링 만들기


    StringTokenizer st = new StringTokenizer(bbb," ");
    while (st.hasMoreTokens()) {
      a.addElement(st.nextToken());
    }

/*  for(int i = 0; i<a.size(); i++){
      System.out.println("AAAA :" +a.get(i));
    }*/

//  System.out.println("원래 텍스트 = "+return_value);
//  System.out.println("스테밍 결과 = " +a);
  return a;

}
```

[그림 6]스테밍 알고리즘 구현 코드

위의 java code 를 가지고 온톨로지 인스턴스들을 스테밍 하면 다음과 같이 원형에 대한 온톨로지 확장이 이루어 진다.

온톨로지 인스턴스 결과 값	CD, Books, Israel, Canada, Standard_International_Shipping, Priority_International_Coury, vinyl, Japan, music_cassettes, VHS_videotapes, Experdited, International_ Shipping
스테밍한 결과 값	CD,Book,Israel,Canada, Standard_International_Ship, Priority_International_Couri, vinyl, Japan, music_casset, VHS_videotap, Experdited, International_ Ship

[그림 7]스테밍 알고리즘 수행 결과값

3.4 웹 페이지 파싱
<[그림 4]의 4,5 번 과정, 결과값 : ③>

규칙구성요소 추출을 위해서 웹 페이지를 본 연구의 시스템 안에서 읽어야 한다. 이는 웹 페이지의 html 파일을 읽어 오는 부분이다. 웹 페이지 파싱 부분에서는 크게 두 가지 부분이 있다. 첫 번째로 웹 페이지를 html 로 읽어 들이는 부분인데 이 부분에서는 Jericho - 2 - html api 를 사용하여 수행 하였다. 두 번째 부분에서는 읽어온 html 파일을 확장된 온톨로지와의 비교를 위해서 앞에서 언급한 Porter 의 스테밍 알고리즘을 사용하여 스테밍을 수행한다.

3.5 의미 유사도 측정(subclass path measure)
<[그림 4]의 6 번과정, 결과값 : ④>

단어들간의 의미가 얼마나 유사한가를 워드넷에 나타나 있는 상위어·하위어개념과 각 단어의 synset 들간의 노드상의 거리를 측정하여 수식으로 표현하고 나타내어 주는 방법이 의미 유사도 측정 방법이다. 이러한 의미 유사도 측정 방법은 pathlength 측정방법, Resinik measure 등 여러 가지의 측정방법이 있다. 본 연구에서 사용한 의미유사도 측정 방법으로 본 논문의 연구와 걸맞는 새로운 measure 를 제안 하였다. 그 유사도 측정 방법을 *subclass path measure* 라 하겠다.

subclass path measure 는 기본적으로 상위어·하위어 개념을 사용하여 비교하는 두 단어의 노드들 간의 거리를 계산한다. 스테밍된 온톨로지의 인스턴스들과 웹 페이지에서 파싱을 한 결과값이 대상이 되고 이 두 가지를 비교하여 추출해 내면 규칙구성요소가 추출이 된다. 그 과정은 다음과 같다.

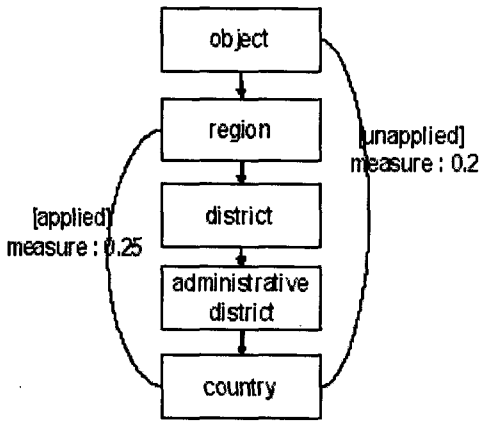
첫 번째, 비교대상이 되는 두 단어가 서로 subclass 관계인지 파악을 한다. 이 방법으로 처음 나온 단어가 두 번째 나오는 단어의 상위어 (Hypernym)이거나 하위어 (Hypo - nym)이면 두 단어는 subclass 관계가 성립을 하게 된다.

두 번째, 두 단어의 노드들 간의 거리를 구하여 역수를 취해준다.(자기 자신이나 동이어 끼리의 거리는 1 이 된다.) 역수를 취해준 값이 의미 유사도가 된다.

$$\text{subclass path measure} = \frac{1}{pathlength}$$

[그림 8]subclass path measure

이 측정 값의 결과가 0.25 이상이 되면 채택을 해주고, 측정 값이 그 미만이 나오면 채택을 하지 않는다.그 예를 보면 다음과 같다.



[그림 9]subclass path measure 의 예시

위의 그림은 region 과 country 두 단어들에 관하여 워드넷에 나타나 있는 노드를 도식화 해 놓은 것이다. 그림을 보면 region 과 country 는 subclass 관계이고 그 거리가 4 이므로 측정값이 0.25 가 나타난다. 측정값이 0.25 이상이고 두 단어가 subclass 관계이기 때문에 region 과 country는 채택이 되어 온톨로지에 country가 있고 웹 페이지에 region 이 있으면 의미 유사도가 높은 것으로 파악되어 규칙구성요소로 추출을 해 온다. 그러나 object 와 country 의 경우에는 두 단어가 subclass 관계에 있지만 측정값이 0.2 로 채택이 되지 않는다.

또 다른 예시로, rain 과 cd 를 측정 해 보면 우선 두 단어가 subclass 관계가 성립하지 않기 때문에 값을 측정하지 않고 채택이 되지 않아서 결과 값으로 subclass path measure 는 0 을 돌려 준다.

3.6 규칙 구성요소 추출

앞에서 제시한 방법을 하나의 플랫폼으로 통합해서 순차적으로 수행해 나가면서 기존 온톨로지와 비교 대상이 되는 웹 페이지를 비교하여서 규칙 구성요소(Rule Components : variables, values)를 추출해 낸다.

4. 기대 효과 및 활용 방안

앞에서 제시한 방법을 가지고 규칙 구성요소를 추출해 내오면 추출해 낸 규칙 구성요소를 조합하여 서론에서 설명한 바와 같이 규칙을 식별해 낼 수 있다.

규칙을 구성하는 기본요소인 변수(variables)와 변수값(values)들이 식별되면 이와 같은 구성요소들을 결합하고 조합하여 규칙을 만든다. 여기서 온톨로지는 주어진 변수(variables)와 변수값(values)들로 구성할 수 있는 가장 유사한 규칙을 자동적으로 제공하는 역할을 한다. 이러한 순차적인 규칙 식별 방법을 통하여 규칙을 식별하는데, 기존의 연구에서는 규칙식별이 단순한 스트링 매칭으로 기술이 되어 있고, 유의어·동의어, 의미유사도 문제에서는 지식관리자가 단순한 문자열 비교와 수작업으로 규칙 구성요소 식별 과정을 수행한다. 여기서 본 연구의 방법으로 규칙 구성요소를 추출해 내면 자동적으로 규칙 구성요소를 추출해 낼 수가 있다. 이로 인해서 지식관리자의 부담을 줄여 줄 수가 있다.

5. 결론

본 연구에서는 온톨로지, 워드넷 확장, 스테밍 알고리즘, 의미 유사도 측정방법등을 이용하여 규칙 구성요소 자동 추출기인 <Rule Components Extractor>를 구현 하였다. 온톨로지는 Amazon.com 을 기본으로 하여 규칙 구성요소로 구성이 되어 있으며, 워드넷 확장을 위하여 WordNet 2.1 과 이를 유용하게 사용하기 위하여 JWNL api 를 사용하였다. 스테밍 알고리즘은 가장 널리 보편적으로 쓰이고 있는 Porter 의 스테밍 알고리즘을 사용하였고, 의미 유사도 측정 방법으로는 subclass path measure 라는 새로운 의미 유사도 측정방법을 설계하여 제안하였다. 비교 대상이 되는 웹 페이지로는 Amazon.com 과 유사한 Domain 을 가지고 있는 Banes&Nobles.com 을 가지고 실험을 하였다. 본 논문에서 구현한 Rule Components Extractor 는 특정 주제에 관한 전문가 시스템을 구축하기 위해 웹 페이지로부터 규칙을 추출할 때 규칙구성요소를 자동적으로 추출하여 규칙을 추출할 때 매우 효율적인 도구가 될 수 있다.

본 연구는 다양한 분야에서 적용될 수 있을 것으로 기대 된다. 예를 들어, 쇼핑몰에서 약관으로부터의 규칙 추출, 각 보험 회사들의 보험 산정 규칙, 그리고 은행의 대출 평가 규칙등이 있다.

참고문헌

[1] *The Effect of Knowldedge Acqusition throuth OntoRule:XRML Approach, Sangun Park, Jae Kyu Lee, Juyoung Kang.*

[2] *Bonnie J. Dorr and Douglas Jones. 1996. Robust Lexical Acquisition: Word Sense Disambiguation to Increase Recall and Precision. Technical report, University of Maryland, College Park, MD*

[3] *International Conference on APL Proceedings of the 2002 conference on APL: array processing languages: lore, problems, and applications, Madrid, Spain pages: 7 - 16*

[4] *Corpus - based stemming using cooccurrence of word variants, Jinxi Xu, W. Bruce Croft, January 1998, ACM Transactions on Information Systems (TOIS), Volume 16 Issue 1*

[5] *Strength and similarity of affix removal stemming algorithms, William B. Frakes, Christopher J. Fox, April 2003, ACM SIGIR Forum, Volume 37 Issue 1*

[6] *Stemming Indonesian, Jelita Asian, Hugh E. Williams, S. M. M. Tahaghoghi, January 2005, Proceedings of the Twenty - eighth Australasian conference on Computer Science - Volume 38 CRPIT '38*

[7] *Semantic interfaces and OWL tools: Parsing owl dl: trees or triples?, Sean K. Bechhofer, Jeremy J. Carroll, May 2004, Proceedings of the 13th international conference on World Wide Web*

[8] *Berners - Lee, T., The Semantic Web, Scientific American, Vol. 501, 2001*

[9] *RDF Primer, ttp://www.w3.org/TR/rdf - primer/*

[10] Web ranking and retrieval: Semantic similarity methods in wordNet and their application to information retrieval on the web, Giannis Varelas, Epimenidis Voutsakis, Paraskevi Raftopoulou, Euripides G.M. Petrakis, Evangelos E. Milios, November 2005, Proceedings of the 7th annual ACM international workshop on Web information and data management WIDM '05