■ 2006년 정보과학 논문경진대회 수상작

# 의미 중의성을 고려한 온톨로지 기반 메타데이타의 자동 생성

## (Ontology-based Automated Metadata Generation Considering Semantic Ambiguity)

최 정 화 † 박 영 택 ††

(Jung-Hwa Choi) (Young-Tack Park)

**요 약** 인터넷의 발전으로 방대해진 정보를 컴퓨터가 이해하고 효율적으로 관리하기 위해서는 시맨틱 웹 기반의 메타데이타가 반드시 필요하다. 그러나 메타데이타 생성 시 의미 중의성을 가진 정보가 존재하며 이 문제의 해결책이 필요하다. 본 논문에서는 순차적으로 존재할 수 있는 단어들의 확률 모델을 이용하여 문서와 같은 정보에 포함된 의미가 애매한 단어를 관련성이 높은 모델의 개념으로 메타데이타를 생성하는 방법을 제안한다. 제안한 방법에서 메타데이타를 생성 할 때, 온톨로지에 정의된 개념들 간의 중의성을 고려하고 명칭(named entity)의 일부 단어에 대한 인식을 위해 은닉 마르코프 모델(Hidden Markov Model)을 사용한다. 먼저 온톨로지에 정의된 각 클래스(class)의 인스턴스(instance)를 인식하기 위한 마르코프 모델을 생성한다. 다음으로 문서로부터 의미가 애매한 단어의 의미를 파악할 수 있는 상황정보(context)를 생성하고, 상황정보에 포함된 단어들의 순서에 대응하는 최적의 마르코프 모델을 찾아 메타데이타 생성시의 중의성 문제를 해결한다. 제안한 방법으로 전산학관련 논문에 대해 의미가 애매한 7개의 단어를 추출하여 실험하였다. 그 결과 상황정보에 존재하는 개체(entity)의 의미부류들 중 가장 빈번한 의미 부류로 애매한 단어의 의미를 선정한 SemTag보다 정확도 면에서 18%정도의 나은 성능을 나타내었다.

**키워드** : 메타데이타, 메타데이타 자동 생성, 컨텍스트, 시맨틱 웹, 온톨로지, 의미 중의성

***Abstract*** There has been an increasing necessity of Semantic Web-based metadata that helps computers efficiently understand and manage an information increased with the growth of Internet. However, it seems inevitable to face some semantically ambiguous information when metadata is generated. Therefore, we need a solution to this problem. This paper proposes a new method for automated metadata generation with the help of a concept of class, in which some ambiguous words imbedded in information such as documents are semantically more related to others, by using probability model of consequent words. We considers ambiguities among defined concepts in ontology and uses the Hidden Markov Model to be aware of part of a named entity. First of all, we construct a Markov Models a better understanding of the named entity of each class defined in ontology. Next, we generate the appropriate context from a text to understand the meaning of a semantically ambiguous word and solve the problem of ambiguities during generating metadata by searching the optimized the Markov Model corresponding to the sequence of words included in the context. We experiment with seven semantically ambiguous words that are extracted from computer science thesis. The experimental result demonstrates successful performance, the accuracy improved by about 18%, compared with SemTag, which has been known as an effective application for assigning a specific meaning to an ambiguous word based on its context.

**Key words** : Metadata, Automated Metadata Generation, Context, Semantic Web, Ontology, Semantic Ambiguity

· 본 연구는 숭실대학교 교내 연구비 지원으로 이루어졌습니다.

† 학생회원 : 숭실대학교 컴퓨터학과
cjh79@ailab.ssu.ac.kr
†† 종신회원 : 숭실대학교 컴퓨터학부 교수
park@comp.ssu.ac.kr

논문접수 : 2006년 5월 18일
심사완료 : 2006년 9월 21일

## 1. 서 론

메타데이타(metadata)란 이미 존재하는 문서 또는 텍스트에 대해 추가적인 설명을 덧붙이는 것으로 주로 정보 검색의 정확도를 높이는 데 크게 기여할 수 있다[1]. 최근 들어 인터넷의 발전으로 웹상에 정보의 양이 많아짐에 따라 자신이 원하는 정보를 발견하는 데 매우 많은 시간을 투자해야만 하는 현상이 생기게 되었고, 이러한 이유를 극복하기 위해 Google과 같은 웹 검색 프로그램의 소프트웨어 에이전트가 사람을 대신하게 되었다. 하지만 웹에 있는 정보는 사람은 이해할 수 있으나 소프트웨어 에이전트는 이해 할 수 없다. 이와 같은 문제를 극복하기 위해 웹을 주창하였던 Tim Berners-Lee는 1999년에 W3C를 중심으로 차세대 웹 기술인 시맨틱 웹(Semantic Web)을 제안하였다. 시맨틱 웹 공간에서는 웹에 있는 정보를 에이전트가 이해할 수 있는 구조를 만들어 줌으로써, 에이전트를 이용하여 대량의 정보를 효율적으로 관리할 수 있다. 따라서 이를 위해 에이전트가 이해할 수 있는 온톨로지 언어를 이용하여 사람이 만든 문서의 내용을 메타데이타의 형태로 표현하는 연구가 반드시 필요하다[2-4].

본 논문에서는 시맨틱 웹 기반의 온톨로지를 이용하여 웹 문서에 대해 메타데이타를 자동으로 생성하는 방법을 소개하고, 이 과정에서 여러 개념을 가진 어휘에 대해 하나의 개념을 결정하는 온톨로지 기반 의미 중의성 해결 알고리즘(OSD: Ontology-based Semantic Disambiguation)을 제안한다. 온톨로지는 메타데이타를 표현하기 위한 해당 분야의 개념이다. 예를 들어 문서가 인공지능 분야 논문이라면 문서 내용에는 인공지능 용어가 나타날 것이고, 소프트웨어 에이전트는 그 인공지능 용어에 대한 개념을 알고 있을 때 그 문서를 이해할 수 있다. 따라서 메타데이타 생성 과정을 통해 문서 내용에 대한 추가적인 설명의 메타데이타를 생성하고, 메타데이타에 있는 어휘(vocabulary)의 개념이 온톨로지에 표현되어 있을 때 소프트웨어 에이전트는 문서의 내용을 이해할 수 있다. 이와 같이 시맨틱 웹에서는 문서가 가지고 있는 의미 있는 정보를 소프트웨어 에이전트가 이해할 수 있도록 메타데이타를 필요로 한다. 그러나 온톨로지에 표현된 개념들은 단어 의미의 중의성(ambiguity)이 존재한다. 단어 의미의 중의성이란 용어에 대한 개념이 온톨로지에 하나 이상 존재할 때 그 문서의 단어를 "ambiguity"라고 한다. 예를 들면, 전산학 분야의 한 논문에 "a network of many simple processors"라는 텍스트가 존재하고, "network" 용어가 전산학 분야 온톨로지의 인공지능 클래스와 컴퓨터 통신 클래스에 개념이 정의되어 있다면, 메타데이타 생성 시 단어의 중의성이 발생한다. 이 문제는 문서 내용에 정확한 메타데이타를 생성하기 위해 반드시 해결되어야 한다.

본 논문에서는 메타데이타 생성의 정확도를 높이기 위하여 단어 의미의 중의성을 해결하기 위한 알고리즘으로 순차적으로 존재할 수 있는 온톨로지 기반의 단어 확률 모델을 이용한 방법을 제안한다. 단어의 순서 모델을 이용한 방법은 은닉 마르코프 모델을 사용한다. 메타데이타 생성 시 중의성을 가진 단어는 어느 클래스에 해당하는지 알지 못한다. 이 의미가 애매한 단어의 앞, 뒤에 존재하는 온톨로지에 표현된 단어들, 즉 상황정보는 중의성을 가진 단어의 의미를 결정지을 수 있으며, 이 단어들 간의 유사성과 상관관계를 이용하여 정확한 메타데이타를 생성할 수 있다. 또한 제안한 방법은 온톨로지에 클래스로 정의되지 않은 어휘에 대해서도 인식이 가능하며, 의미를 부여하고자 하는 상황정보의 범위에 따른, 즉 한 문장, 한 문단, 한 문서 단위로의 메타데이타 생성이 가능하다.

본 논문은 다음과 같은 순서로 구성된다. 2장에서는 기존의 의미 중의성을 고려한 메타데이타 생성 연구들에 대해 설명하고 3장에서는 제안한 단어들의 순서 모델을 이용한 의미 중의성 해결 알고리즘을 통해 메타데이타 생성의 정확도를 높인 방법을 설명한다. 4장에서는 제안한 방법의 타당성과 정확성을 검증하기 위해 기존 연구 SemTag과 비교한 실험결과를 기술한다. 마지막 5장에서는 결론을 맺고 향후 연구를 제시한다.

## 2. 관련 연구

웹이나 개인 PC에 존재하는 방대한 정보를 유용하게 관리하기 위한 메티데이타 생성 연구는 크게 생성 방식에 따라 수동과 반자동, 그리고 자동 생성으로 나눌 수 있다. 메타데이타를 자동으로 생성하는 방법은 수동과 반자동 보다는 신뢰도가 비교적 낮지만 대량의 문서처리에 적합하고, 메타데이타의 통일성이 높다. 따라서 메타데이타 자동 생성 연구가 활발히 진행되고 있으며, 다음은 그 중 메타데이타 생성 시 문제점인 의미의 중의성까지 고려한 대표적인 방법으로 KIM과 SemTag 연구에 대해 설명한다[5,6,8,9].

### 2.1 Ontotext 연구소의 KIM(Knowledge and Information Management)

KIM은 온톨로지를 기반으로 GATE[7-9]의 정보추출(Information Extraction)기법을 사용하여 문서로부터 의미 있는 정보를 추출하고 메타데이타를 생성하는 플랫폼(platform)이다. KIM 온톨로지(KIMO)는 기관(Organization), 사람(Person), 날짜(Date), 장소(Location) 등과 같은 일반적인 클래스를 정의하였으며, 문서의 각각의 개체를 타입(type) 속성을 갖는 하나 이상의 클래스의 인스턴스로 정의할 수 있도록 설계되었다. 이 온톨로

지는 300개의 클래스와 100개의 속성(property), 즉 관계들로 구성된다. KIM은 GATE의 정보 추출 엔진을 이용하여 문서 내용의 메타데이타를 생성하므로 다른 연구와 비교하여 자연어 정보추출 엔진으로 우수하다. 이 연구는 중의성을 가진 단어에 대해 온톨로지에 표현된 모든 개체를 추출하고 리스트로 제공하여 사용자가 선택하도록 한 후, 가장 선호하는 일반적인 의미를 부여하는 방법을 사용하여 단어 의미의 중의성을 해결한다. 하지만 중의성을 가진 단어가 존재하는 상황정보의 의미를 고려하지 않으므로 중의성을 가진 단어에 대한 메타데이타는 정확성이 떨어진다.

### 2.2 GATE(General Architecture for Text Engineering)

KIM에서 사용한 GATE는 온톨로지 또는 말뭉치(corpus)를 기반으로 문서의 내용으로부터 개체를 인식하기 위한 프레임워크(framework)이다. 인식할 수 있는 온톨로지의 개체로는 사람(Person), 기관(Organizations), 날짜(Date), 장소(Location)등의 일반적인 타입이다. 예를 들어 "The shiny red rocket was fired on Tuesday. It is the brainchild of Dr. Big Head. Dr. Head is a staff scientist at We Build Rockets Inc." 라는 텍스트가 있다면, 인식되는 정보는 "rocket", "Tuesday", "Dr. Head" 그리고 "We Build Rockets" 등의 단어이다. 이 단어들은 온톨로지에 정의된 인스턴스 들이며, 기계(또는 에이전트)가 이해 가능한 온톨로지의 개체로 변환한다. 이 개체들 중 "it"은 "rocket"과 동일하고, "Dr. Head"와 "Dr. Big Head"는 동일 인물임을 인식하며, "Dr. Head"는 "We Build Rockets"에서 근무한다는 것도 인식한다.

GATE는 영어, 스페인어, 일본어, 중국어 등의 다국어 처리가 가능하고, 정보추출의 정확성이 97%로 우수하다. 하지만 개체간의 관계인식은 60~70%의 정확률을 보이며, 개체의 타입이 애매한 경우 사람들의 선호도를 조사하여 가장 일반적인 의미를 선택한다. 예를 들어 위의 예제에서 "rocket"이 사람 이름으로도 존재한다면, 일반적인 의미인 "분사식 엔진"으로 메타데이타를 생성한다. 따라서 중의성만 존재하지 않는다면 개체명 추출로 우수한 프레임워크이다.

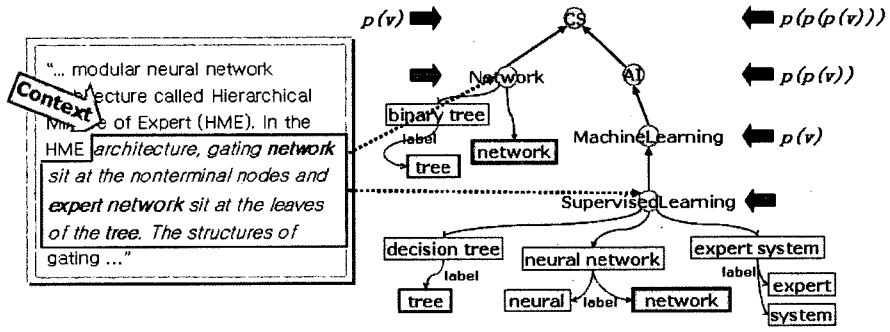
### 2.3 IBM의 SemTag(A Semantic Tagger)

SemTag은 264백만 웹페이지를 수집하여 434백만 개의 중의성을 고려하고 시맨틱 태그(tag)를 생성한 큰 규모의 자동 메타데이타 생성 플랫폼이다. SemTag은 대량의 문서를 대상으로 TAP[7]의 어휘와 개체들의 Taxonomy를 이용한다. TAP은 Music, Movies, Authors, Sports, Places 등의 인기 있는 도메인의 어휘를 포함하는 큰 규모의 온톨로지이다.

SemTag은 KIM과는 달리 TBD(Taxonomy-Based Disambiguation)라는 중의성 해결 알고리즘을 제안하여 상황정보의 의미를 고려하여 메타데이타를 생성한다. 그러나 다른 연구와 마찬가지로 개체의 일부 단어가 온톨로지에 인스턴스로 정의되어 있지 않으면 인식이 불가능한 단점이 있다. 예를 들어 사람이름을 하나의 인스턴스로 나타낸다면, 성(last name)과 이름(first name)을 따로 정의해 주던지 하나의 인스턴스의 속성의 값(value)로 정의해 주어야만 인식이 가능하다. 또한 의미가 애매한 단어의 개념을 파악하고자하는 상황정보에 두 가지 이상의 애매한 단어가 포함되어 있을 경우를 고려하지 않았고, 상황정보에 애매한 단어와 관계가 없는 단어가 특정 클래스와 높은 유사도를 가질 경우 부정확한 메타데이타를 생성하게 된다.

### 2.4 SemTag의 TBD 알고리즘

TBD 알고리즘은 Taxonomy를 기반으로 의미가 애매한 단어의 개념을 선택하는 알고리즘이다. 정보검색의 벡터 공간 모델을 사용하여 상황정보와 클래스간의 가장 높은 가중치를 갖는 클래스의 개념을 애매한 단어의 의미로 채택한다.




**(a)** 문서 내용의 예 **(b)** Taxonomy 예

그림 1 TBD 알고리즘 예

SemTag에서 제안하는 상황정보는 애매한 단어를 기준으로 앞, 뒤의 10개 단어를 포함하는 총 21개의 단어의 집합이다(전치사, 관사 포함). 예를 들어 그림 1의 (a)는 SupervisedLearning 클래스에 관련된 논문 내용의 일부이다. 내용의 두 번째 "network"에 대한 메타데이타를 생성한다면, 상황정보는 (a)의 네모상자의 21개 단어가 된다. 그림 1의 (b)에서 "network"는 Network 클래스와 Supervised Learning 클래스에 개념이 정의되어 있고, 따라서 중의성이 발생한다. TBD알고리즘은 상황정보의 단어들이 Taxonomy에 속하는 노드(node)들의 가중치를 계산하여 가장 높은 가중치를 갖는 노드의 개념으로 "network"의 중의성을 해결한다.

SemTag에서 제안하는 중의성 해결 방법은 기존의 정보검색 엔진에서 사용하는 방법이다. tf-idf로 단어의 가중치를 계산하고, 코사인 기법(cosine measure)으로 상황정보와 노드간의 유사도를 계산하여 82%정도의 정확도를 보였다. Taxonomy에 표현된 개체 수의 차원 벡터를 만들고, 애매한 단어의 개념이 표현된 노드 Network와 SupervisedLearning에 대해 상황정보에 포함된 단어들의 가중치를 벡터 값으로 채운다. 대상 노드를 $v$로 보고 상위 노드 $p(v)$들의 벡터 값도 계산하여 가장 높은 유사도를 갖는 노드의 개념을 애매한 단어의 의미로 채택한다.

그림 1의 예를 통해 SemTag 알고리즘의 단점에 대해 살펴보면, 첫째, 개체 "network"가 Supervised-Learning 클래스를 타입으로 갖는 인스턴스 "neural network"의 서브개념으로 정의되지 않았으면 이 클래스에 관련된 개념으로 인식되지 못한다. 둘째, 상황정보에 포함된 단어 중 "tree"도 두 클래스를 도메인으로 가져서 중의성이 증가하지만 이 경우에 대해 고려하지 않았다. 셋째, 실제로 Supervised Learning에 관련된 문서이지만, "tree" 단어가 Network에서 높은 가중치를 나타낸다면 "network"의 의미가 잘못 부여되는 오류가 발생한다.

## 3. 의미 중의성을 고려한 온톨로지 기반 메타데이타 자동 생성

본 논문은 온톨로지 기반의 메타데이타 생성 시 문제점인 의미가 애매한 단어의 중의성을 해결하기 위한 알고리즘을 이용하여 메타데이타를 생성하는 방법을 제안한다. 제안한 방법은 전처리 과정, 학습 과정, 그리고 의미결정 과정으로 구분된다. 그림 2는 단어 의미의 중의성을 고려하여 메타데이타를 생성하기 위한 전체적인 개요를 나타내고 있다.

시스템 구조를 살펴보면, 첫째, 전처리 과정을 통해 메타데이타로 생성할 개체들을 정의할 온톨로지를 구축하고, 온톨로지에 표현된 개체 정보를 추출하여 용어 사전을 구축한다. 상황정보 추출기는 용어 사전을 이용하여 문서로부터 상황정보를 추출한다. 둘째, 학습 과정은 상황정보 추출기를 이용하여 학습될 문서들의 상황정보를 추출하고, 온톨로지를 기반으로 각 클래스별 포함되는 개체 간의 확률을 계산하여 마르코프 모델을 생성한다. 셋째, 의미결정 과정은 문서 내용에 온톨로지를 기반으로 메타데이타를 생성한다. 이 과정에서 의미가 애매한 단어의 중의성을 해결하는 알고리즘을 사용한다. 이와 같이 제안한 방법은 순차적으로 존재할 수 있는

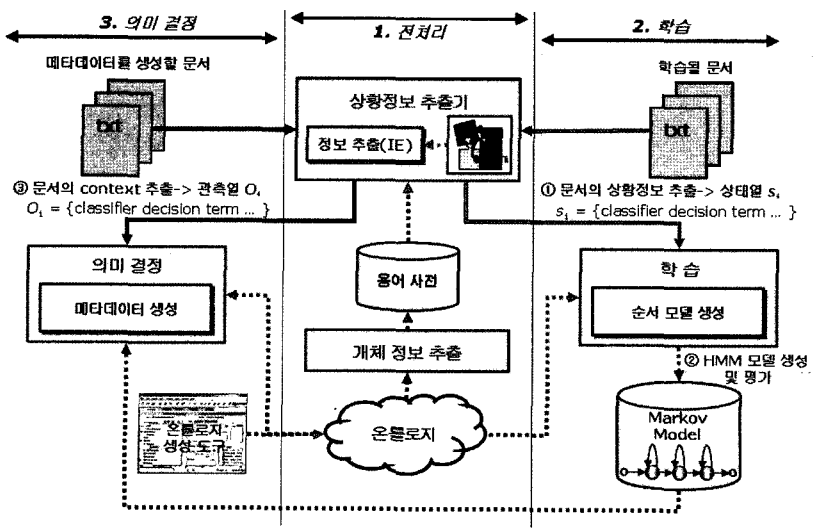



그림 2 온톨로지 기반 의미 고려 메타데이타 자동 생성 시스템 구조

단어의 확률 모델을 이용하기 위해 은닉 마르코프 모델을 사용한다. 메타데이타 생성 시 문제점인 의미가 애매한 단어는 어느 클래스에 해당하는 단어인지 알지 못한다. 그러나 이 의미가 애매한 단어의 앞, 뒤에 존재하는 단어가 온톨로지의 인스턴스로 표현되어 있다면, 중의성을 가진 단어의 의미를 결정지을 수 있다. 따라서 이 단어들을 상황정보로 생성하고 각 단어 간의 유사성과 상관관계에 의존해서 메타데이타를 생성한다.

은닉 마르코프 모델은 이러한 제약조건에서 효과적인 모델링 도구로 사용될 수 있다. 이 모델은 단지 관측열에만 의존해서 확률적으로 대상을 모델링한다. 이렇게 구축된 모델들을 이용하여 특정 관측열이 생성되었을 확률 및 최적의 상태전이를 구할 수 있다. 이러한 은닉 마르코프 모델의 특성은 관측열을 애매한 단어의 의미를 판별할 수 있는 상황정보로 보고, 상황정보에 포함되는 단어들의 순서에 대응하는 최대 확률을 갖는 구축된 모델을 찾아서 그 모델에 정의된 개념을 애매한 단어의 의미로 메타데이타를 생성한다.

본 논문은 온톨로지에 정의된 인스턴스에 포함된 단어들의 순서모델을 이용함으로써 개체로 인식해야 할 일부 단어가 온톨로지에 정의되어 있지 않더라도 인식이 가능한 장점을 갖는다. 또한 상황정보의 범위에 따라 한 문장, 한 문단 또는 한 문서 단위로의 메타데이타 생성에 정확성을 높인다.

### 3.1 전처리 과정

전처리 과정은 크게 개체 정보 추출 단계와 상황정보 추출 단계로 구분된다. 개체 정보 추출 단계는 본 논문에서 구축한 온톨로지로부터 인스턴스로 존재할 수 있는 단어 정보를 추출한다. 이 과정에서 온토롤지에 표현된 인스턴스의 일부분을 인식하기 위해 인스턴스의 일부분인 의미 있는 단어를 추출하여 용어사전을 구축한다. 상황정보 추출 단계는 개체 정보 추출단계에서 구축한 용어사전을 이용하여 학습 단계 또는 의미결정 단계에서 문서가 들어오면 용어사전의 단어정보만을 추출하여 상황정보를 생성한다.

3.1.1 온톨로지 구축 및 개체 정보 추출 단계

본 논문에서는 사람이 작성한 문서를 소프트웨어 에이전트가 이해 가능한 언어로 변환하기 위해 OWL (Web Ontology Language)을 사용한다. OWL은 풍부한 어휘와 형식적 의미론(formal semantics)을 포함하고 있기 때문에 기계 해석이 가능한 웹 콘텐츠를 저작하는데 있어 XML, RDF 및 RDF 스키마(RDF-S)보다 뛰어나다[11]. 웹 문서로부터 추출된 정보는 단어 형태로 된 것이기 때문에 소프트웨어 에이전트가 이해 가능한 언어로 변환이 필요하며, 이때 소프트웨어 에이전트 언어의 기본 단위는 온톨로지의 개체가 된다.

본 논문에서 온톨로지 역할은 도메인 내에서 사용되는 용어들의 의미를 정의함으로써, 문서 내용에 의미 있는 메타데이타 생성의 정확도를 향상시키는 데 있다. 즉, 의미적 연관성을 통한 소프트웨어 에이전트로 하여금 이해할 수 있는 문서를 만들기 위한 시맨틱 웹 환경에서의 온톨로지이다[12]. 온톨로지는 기본적으로 클래스, 서브클래스(subClass), 속성(property), 도메인(domain), 레인지(range), 그리고 인스턴스(instance) 등의 구성요소를 가지고 도메인내의 정보를 표현한다.

표 1의 OWL문은 SupervisedLearning 클래스의 "neural network"라는 인스턴스의 종류(type)와 이름(name)의 두 가지 특성에 대한 정보를 표현하고 있다. *온톨로지에서 하나의 인스턴스는 개체의 집합을 의미한다.* "neural network"라는 정보는 SupervisedLearning 클래스의 인스턴스이며, 온톨로지의 계층구조 의미에 따라 ComputerScience와 Artificial Intelligence, 그리고 MachineLearning의 서브 클래스로 선언되어 상위 클래스는 하위 클래스의 정보를 모두 포함한다. Supervised Learning 클래스의 속성 중 정보의 종류를 나타내는 속성 type의 도메인은 ComputerScience 클래스가 되고, 레인지는 type값이 속할 수 있는 범위를 나타내는 Type 클래스로 정보의 의미를 표현한다. Type 클래스는 전문용어와 사람으로 분류된다. 속성은 두 가지 종류가 존재하며, 이와 같이 클래스나 인스턴스를 레인지로 갖는 속성을 ObjectProperty라 하고, 문자열이나 숫자 등의 데이타 형을 레인지로 갖는 속성을 Datatype Property라 한다.

개체 정보 추출 단계에서는 소프트웨어 에이전트가 문서의 내용 중 의미 있는 정보를 추출하기 위해 온톨

표 1 OWL로 표현된 온톨로지

```
<SupervisedLearning rdf:ID="neuralNetwork_13">
    <type> <Type rdf:ID="term"/> </type>
    <name rdf:datatype="http://www.w3.org/2001/
      XMLSchema#string"
    >neural network</name>
</SupervisedLearning>
<owl:Class rdf:ID="SupervisedLearning">
    <rdfs:subClassOf rdf:resource="#MachineLearning"/>
</owl:Class>
<owl:Class rdf:ID="MachineLearning">
    <rdfs:subClassOf rdf:resource="#ArtificialIntelligence"/>
</owl:Class>
<owl:Class rdf:ID="ArtificialIntelligence">
    <rdfs:subClassOf rdf:resource="#ComputerScience"/>
</owl:Class>
<owl:ObjectProperty rdf:ID="type">
    <rdfs:range rdf:resource="#Type"/>
    <rdfs:domain rdf:resource="#ComputerScience"/>
</owl:ObjectProperty>
```

로지에 포함된 개체 정보를 인식한다. 개체 정보는 도메인 온톨로지의 인스턴스를 가장 작은 단위로 나눈 단어를 의미한다. 위의 예에서 SupervisedLearning 클래스의 인스턴스 중 "neural network"의 개체 정보는 "neural"과 "network"이며, 개체 정보 추출 엔진을 통해 추출된다. 본 논문에서는 OWL로 작성된 온톨로지를 그대로 사용하여 개체 정보를 추출할 수 없기 때문에 온톨로지의 변환이 필요하다. 기존의 RDF와 DAML+OIL에서 정의된 공리는 일차논리(First Order Logic) 표현방식을 따르고 있다. 즉, 온톨로지의 인스턴스를 인식하기 위해서는 <property><subject><object>형태의 트리플 형식으로 변환해야 한다. 본 개체 정보 추출 엔진의 트리플 변환기는 Jena의 RDF 파서를 사용하여 구현하였다[13-15]. 온톨로지는 이러한 트리플 변환기를 통하여 PSO형태의 트리플로 변환되고, 온톨로지의 클래스별로 개체 정보를 추출하여 용어사전에 저장한다. 따라서 용어사전에는 클래스별로 존재할 수 있는 개체의 일부 단어들이 분류되어 저장된다.

3.1.2 상황정보(context) 추출 단계

그림 2의 상황정보 추출 단계에서의 정보추출엔진은 GATE[8,9]의 API를 이용하여 문서로부터 용어사전에 정의된 단어를 추출한다. 소프트웨어 에이전트는 단어의 의미를 이해하기 위해 온톨로지의 개체 정보를 이용해서 문서로부터 상황정보를 추출한다. 상황정보는 학습단계 또는 의미결정단계에서 문서가 들어오면 정보추출엔진을 통해 생성된다. 상황정보는 애매한 단어의 의미를 판별할 수 있는 애매한 단어를 포함한 근접한 단어들의 집합이며, 애매한 단어란 메타데이타를 생성하고자 하는데 온톨로지의 두 개 이상의 클래스에 개념이 정의되어 있어 의미가 명확치 않은 단어이다. 다음은 본 논문에서 제안하는 중의성과 상황정보에 대해 수식을 통해 자세히 살펴본다.

3.1.2.1 중의성(ambiguity)의 정의

자연어는 사람과 사람사이의 의사소통의 수단이기 때문에 정보를 입력한 사람이 의도한 의미는 문장으로 변환되고, 입력이 되는 언어 표현에 대해서 단일의 올바른 해석이 되었을 때 정보 전달이 잘 되었다고 볼 수 있다. 그러나 수신자 측에서 이해한 의미가 두 가지 이상의 해석이 가능할 때도 있으며, 단어가 의미하는 범위가 불명확 할 수도 있다. 이것은 언어 표현에 있어서의 중의성의 발생에 해당하며 온톨로지에서도 이와 같은 문제점이 발생한다.

온톨로지에서의 중의성이란, 단일 언어 표현에 대해서 상이한 복수의 해석이나 불명확한 해석이 대응하는 경우를 말한다. 예를 들면 그림 3과 같이 개체 "network"는 클래스 SupervisedLearning 와 Network, 즉 두 클래스에 개체 정보가 존재하여 의미가 애매한 경우이다.

본 논문에서는 온톨로지 $Ot$를 다음의 네 가지 요소로 표현한다. 클래스의 집합을 $C$, 서브클래스 $Sb$의 관계를 $Sb \subseteq C \times C$, 인스턴스의 집합을 $I$, 그리고 인스턴스가 속하는 도메인의 타입을 $T \subseteq I \times C$로 표현한다. 온톨로지 $Ot$는 $Ot = \{d_1, d_2, \cdots, d_n\}$, 각 클래스 $d_1, d_2, \cdots, d_n$의 집합으로 표현되며 문서를 $D$라 할 때, 식 (1)과 같이 애매한 단어 $W_A$를 정의한다.

$$W_A = \{w_a \mid w_a \in D, w_a \in C_i \cap C_j\}, (i \neq j, 1 \leq i,j \leq n) \quad (1)$$

애매한 단어 $W_A$는 온톨로지 $Ot$에 표현된 두 개 이상의 클래스에 개체 정보를 갖고 있는 단어이다. 단어 $w$는 온톨로지의 개체로 정의되어 있어서 소프트웨어 에이전트가 문서에서 인식한 인스턴스 $I$의 부분집합, 즉 $w \subseteq I$이다. 이 때, $w$가 포함된 인스턴스 $i$가 $t(i,d_1)$과 $t(i,d_2)$의 타입으로 관계가 정의되어 있으면 $w$를 애매한 단어 $w_a$로 인식한다.

3.1.2.2 상황정보(context)의 정의

상황정보는 사용자와 컴퓨팅 환경 사이에 관련된 사용자의 환경, 객체, 상태에 관한 상황을 특징지을 수 있

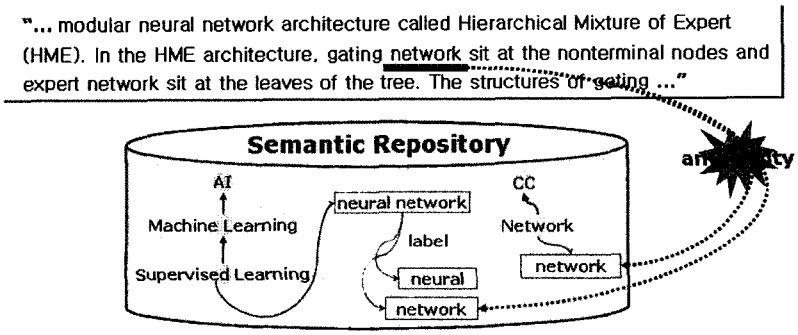



그림 3 온톨로지에서의 단어 의미의 중의성

는 모든 정보를 말한다[16]. 상황정보라는 용어는 여러 분야에서 다양한 의미로 사용되고 있으나, 본 논문에서는 상황정보를 "문서의 내용 중 온톨로지에 표현된 개체와 관련된 모든 정보"라고 정의한다.

소프트웨어 에이전트가 문서의 내용을 정확히 이해하고 메타데이타를 생성하기 위해서는 문서의 작성자가 의도한 내용의 의미를 파악하는 상황정보 인지 기술이 중요한 역할을 담당한다. 메타데이타 생성에서 상황정보 인지 기술을 이용하면, 문서의 내용에 포함되는 개체의 명확한 의미 정보 분석을 통한 상황정보를 이용하여 문맥에 맞는 정확한 메타데이타를 생성할 수 있다.

그림 4는 상황정보를 지식베이스와 실제 문서의 두 가지 측면에서 살펴본 그림이다. 먼저, 소프트웨어 에이전트는 개체 정보 추출 단계를 통해 온톨로지에 정의된 개체 정보들을 클래스별로 구분하여 용어사전을 구축한다 (a). 다음으로 (b)와 같은 애매한 의미의 단어 "network"가 포함된 문서가 입력되면, 용어사전에 포함된 개체정보를 추출하여 이 집합을 상황정보라 한다. 따라서 (b)문서의 상황정보는 "neural network Expert network expert network tree"가 되며, 이 상황정보는 (a)의 온톨로지에 표현된 관련 있는 개체의 계층구조와 개체에 정의된 의미를 포함한다.

상황정보는 애매한 단어 $W_A$를 포함하는 문서 $D$의 단어 $w$중에 온톨로지 $Ot$에 표현된 인스턴스 $I$의 집합이다. 식 (2)는 상황정보 $C$를 수식으로 표현한 것이다. 상황정보 $C$는 문서 $D$에 포함된 애매한 단어 $W_A$를 포함한 단어 $w$들의 집합이며, 단어 $w$는 온톨로지에 표현된 클래스에 존재하는 개체 정보들이다.

$$C = \{w_a, w \mid w \in D,\ w \in C_i,\ C_i \in Ot,\ w_a \in W_A\} \quad (2)$$

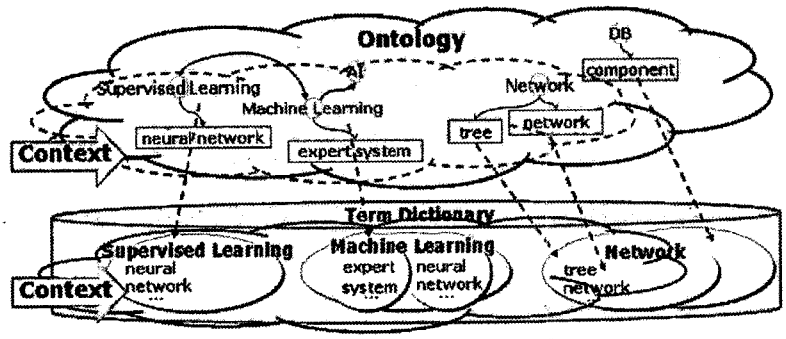
본 논문에서 제안한 상황정보는 메타데이타 생성 시 의미가 애매한 단어가 포함된 정보에 존재하는 모든 단어를 대상으로 한다. 정보는 하나의 문서, 문단 또는 문장이 될 수 있다. 애매한 단어의 의미를 판단하기 위해 상황정보의 범위를 정보의 전체 단어를 대상으로 하는 이유는 각 클래스에 포함될 수 있는 개체의 확률을 높여 애매하지 않은 단어 의미의 정확성을 좀 더 높여주기 위함이다.

### 3.2 학습 과정

학습과정에서는 온톨로지의 각 클래스별로 관련문서를 수집하여 문서에 포함된 개체 정보를 추출하여 마르코프 모델을 생성한다. 마르코프 모델은 의미결정 과정에서 메타데이타 생성 시 단어 의미의 중의성을 해결하기 위한 확률 모델이다. 이 과정에서는 전처리 과정에서 구축된 온톨로지 기반 용어사전을 이용하여 각 클래스에 존재할 수 있는 개체를 추출한다. 즉, 한 문서, 한 문단, 또는 한 문장 내에 각기 다른 두 단어가 나타날 확률을 구하는 것으로, 이것이 곧 단어 사이의 연관관계를 결정짓는 요소가 된다.

상황 정보 추출기는 학습할 문서에서 상황정보를 추출하고, 학습엔진은 먼저 상황정보에 존재하는 단어들의 순서대로 상태 열 $S$, $S=\{s_1, s_2, \cdots, s_n\}$, 즉 단어들의 순서집합을 만든다. 다음으로 식 (3)을 이용한 단어 간 확률을 통해 각 클래스별로 문서에 나타난 개체들에 대하여 그림 5와 같은 마르코프 모델을 만들게 된다. $a_{ij}$는 상황정보에 나타난 두 단어 간의 확률이다.

식 (3)은 $a_{ij}$확률을 계산하는 식이다. $P(S_j \mid S_i)$의 계




(a) 지식 베이스 (Knowledge Base)

(b) 문서 (Document)

그림 4 상황정보(context)

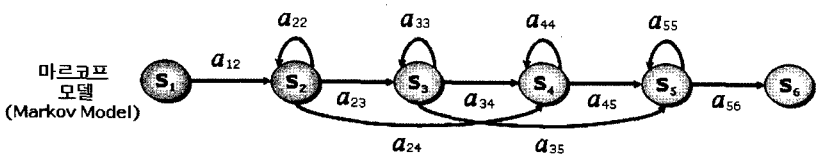



그림 5 마르코프 모델

산식은 식 (4)와 같으며 임의의 단어 $S_i$가 주어지면 $S_i$와 $S_j$가 클래스 상에 근접하여 나온 확률을 구하여 해당 클래스 문서에 나온 $S_i$의 전체 확률로 나누어 $S_i$와 $S_j$가 특정 클래스에서 나타난 확률을 구한다.

$$P(S_j \mid S_i) = a_{ij} \tag{3}$$

$$P(S_j \mid S_i) = \frac{P(S_j \cap S_i)}{P(S_i)} \tag{4}$$

식 (3)을 통해 생성된 마르코프 모델은 그림 6과 같이 용어사전의 전체 단어 수만큼의 차수를 가진 행렬로 온톨로지에 표현된 클래스 개수만큼 전이 확률 행렬이 생성된다.

그림 6의 $a_{ij}$는 단어$i$와 단어$j$가 연속해서 나온 확률이다. 각 클래스는 순서적으로 존재할 수 있는 단어 간의 확률을 가지고 클래스에 포함될 수 있는 개체들을 학습하게 된다. 각 클래스의 전이 확률 행렬은 하나의 모델로서 생성된다.

### 3.3 의미결정 과정

의미결정 과정은 구성된 마르코프 모델로부터 최적 확률을 탐색한 후, 최적 확률을 갖는 클래스에서 정의된 개념으로 해당 의미가 애매한 단어에 의미를 할당하는 과정이다. 이 때, 단어 의미의 중의성을 해결하기 위해 본 논문에서는 그림 7과 같은 알고리즘을 제안한다.

알고리즘을 살펴보면, 먼저 메타데이타를 생성하고자 하는 문서의 텍스트가 입력된다. (1)에서 문서의 상황정보를 구한다. 상황정보를 구하는 식은 3.1.2.2절의 식 (2)이다. 의미를 결정할 상황정보를 관측열로 보며 $o_0$은 시작 단어, $o_k$는 관측열의 마지막 단어를 의미한다. 다음으로 관측열에 포함되는 단어 의미의 중의성을 판단하여 온톨로지에 정의된 최적의 클래스의 개념을 메타데이타로 생성한다. (2)에서는 (1)에서 구한 관측열의

입력 : 문서의 텍스트
출력 : $M$

(1) 관측열 $O = \{o_0, o_1, o_2, \ldots, o_{k-1}, o_k\}$, $O$는 문서의 상황정보

*while*$(o_i \neq o_k)$ {
(2) *Let* $x = funWordAmbiguity(o_i)$
(3) $x = 0$ 일 때,
*find* $M_i \in \text{argmax}\{P(O \mid M_i)\}$
(4) $x = 1$ 일 때,
*find* $M_i$
(5) *return* $M_i$
$i = i + 1$
}

그림 7 온톨로지 기반 의미 중의성 해결(OSD) 알고리즘

단어 의미가 애매한지를 판단한다.

$funWordAmbiguity()$함수는 3.1.2.1절의 식 (1)을 이용하여 단어의 의미가 애매하면 0, 그렇지 않으면 1을 반환한다. (3)은 단어의 의미가 애매한 경우로써, 주어진 관측열에 대해 학습과정에서 생성한 각 모델의 전이 확률 중 최적의 값을 갖는 모델을 구한다. 이 모델이 애매한 단어의 의미를 결정짓는 클래스의 개념이다.

모델 $M_i$는 $M_i = \{a_{i(0)i(1)}, a_{i(1)i(2)}, \ldots, a_{i(n-1)i(n)}\}$, 즉 각 클래스에 존재하는 단어 간 전이 확률 집합이다. $M_i$의 원소 $a_{i(m)i(n)}$ 는 $M_i$의 단어 $m$에서 단어 $n$으로 가는 전이 확률이다. 학습 과정에서 구한 각 클래스의 모델 $M_i$가 주어지고, 관측열에 대응하는 확률 값 중 최대 확률을 갖는 모델 $M_i$를 구한다. $P(O \mid M_i)$의 계산 과정은 식 (5)와 같다.

$$P(O \mid M_i) = a_{i(0)i(1)} \cdot \prod_{j=1}^{k-1} (a_{i(j)i(j+1)}) \tag{5}$$

(4)는 단어 의미가 애매하지 않은 경우로써, 단어의 개념이 표현된 클래스 모델을 찾아 반환한다. 이와 같은

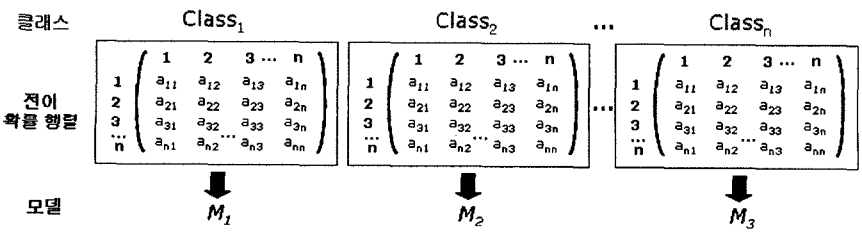



그림 6 클래스별 전이 확률

과정을 통해 중의성을 고려하여 문서 내용에 적합한 메타데이타를 생성하는데 정확도를 높이게 된다. 예를 들면, 그림 4의 예와 같은 문서가 입력되면, (2)에서 메타데이타의 생성이 애매한 "network" 단어에 대해 (3)이 적용된다. (3)에서는 (1)에서 추출한 관측열 "neural network Expert network expert network tree"에 대해 각 모델의 전이확률을 구한다.

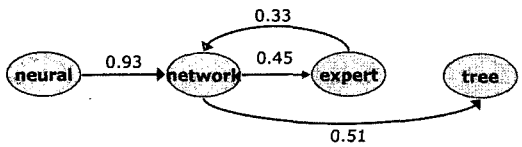



그림 8 전이 확률의 예

그림 8은 그림 4의 상황정보에 대한 전이확률을 나타낸 것이다. 이러한 전이확률은 각 모델별로 계산되어 식 (5)에 의해 최대값을 갖는 모델을 구하게 된다. 그리고 그 모델에 해당하는 클래스에 정의된 개념을 애매한 단어 "network"의 의미로 선택하고 메타데이타를 생성한다. 만약, 예의 "network"가 SupervisedLearning 클래스에서 최대 확률을 갖게 되면 다음과 같은 메타데이타가 부착된다.

<SupervisedLearning rdf:ID="neuralNetwork_13">
network </SupervisedLearning>

메타데이타에는 OWL 언어로 표현된 레이블이 부착되고 해당 레이블이 가지는 계층관계와 의미를 모두 포함하여 의미 있는 데이타로 활용된다.

## 4. 실험 및 결과

제안한 단어 의미의 중의성을 고려한 온톨로지 기반 메타데이타 생성 방법은 순차적으로 존재할 수 있는 단어의 확률 모델을 이용한다. 문서의 상황정보를 생성하고 상황정보가 포함되는 가장 확률이 높은 클래스의 개체 의미로 메타데이타를 생성한다. 제안한 방법의 정확성 검증 실험을 위해 기존연구 SemTag [6]에서 82%정도의 정확성을 입증한 정보검색(IR)기법의 tf-idf를 이용한 방법과 본 연구의 은닉 마르코프 모델을 이용한 방법을 동일한 온톨로지의 도메인과 실험 데이타를 통해 비교 분석하여 본 논문의 우수성을 확인한다.

### 4.1 전산분야 온톨로지를 이용한 실험 데이타

기존연구와 본 연구에서 제안하는 방법을 동일한 조건에서 정확도를 평가하기 위해 전산분야 도메인 온톨로지를 구축하여 실험하였다. 전산분야 온톨로지에 정의된 클래스는 여러 대학의 전공분류를 참조하여 구성하였으며, 각 전공 분야의 용어는 정보 기술 분류별 용어사전[17]과 인공지능 전공분야 홈페이지 AI Study[18]의 분류별 용어를 참조하여 구축하였다. 구축된 온톨로지에 표현된 인스턴스는 전문용어와 각 용어에 관계되는 수학자나 과학자 이름이 포함되며, 사람에 관계된 소속 기관과 기관 등에 해당하는 클래스와 인스턴스를 제외하고 전공분야 온톨로지는 24개의 클래스와 약 1,200개의 인스턴스를 포함한다.

학습 데이타는 온톨로지에 정의된 전공분야에 해당하는 논문으로 총 406개의 문서를 수집하였으며, 메타데이타 생성에 사용된 실험 데이타는 학습 데이타에 포함되지 않은 270개의 논문을 수집하여 학습된 모델의 성능을 검증하는데 사용하였다. 실험 데이타는 모두 의미가 애매한 단어를 하나 이상 포함하는 문서이며, 실험 대상이 되는 애매한 의미의 단어는 표 2와 같다.

전처리 과정을 통해 온톨로지에 표현된 인스턴스는 개체로 인식될 수 있는 최소의 단위인 단어로 구분하여 788개의 의미 있는 단어를 추출하였다. 이 단어 수는 중복을 제거한 개수이며 용어사전에 분류별로 저장 된다. 따라서 학습 과정에서의 전이 확률 행렬은 788 × 788로

표 2 두 개 이상의 클래스에 포함되는 의미가 애매한 단어

| 의미가 애매한 단어 | 의미가 정의된 클래스 (상위 클래스*) |
|---|---|
| clustering | Machine Learning(AI), Database(SE) |
| domain | Machine Learning(AI), Network(CC) |
| frequency | Machine Learning(AI), Fuzzy(AI), Linguistics(AI), Network(CC) |
| semantic | Linguistics(AI), Database(SE) |
| network | Machine Learning(AI), Fuzzy(AI), Linguistics(AI), Pattern Recognition(AI), Reasoning(AI), Robot(AI), Supervised Learning(Machine Learning(AI)) Network(CC) |
| model | Machine Learning(AI), Neural Network(AI), Database(SE) |
| tree | Machine Learning(AI), Heuristic(AI), Network(CC), Database(SE) |

*AI:Artificial Intelligence, SE:Software Engineering, CC:Computer Communication

써 조합을 이룰 수 있는 인스턴스의 마르코프 모델을 생성한다.

### 4.2 성능 평가 방법

제안하는 방법의 성능을 평가하기 위하여 실험 대상 단어의 메타데이타에 대하여 정확도를 계산하였다. 정확도의 식은 식 (6)과 같다.

$$\text{정확도}(accuracy) = \frac{T}{M} \tag{6}$$

$M$은 실험 데이타에 포함된 애매한 단어에 대해 해당 메타데이타를 생성해야하는 예제 단어 수이며, $T$는 제시된 예제 단어 중에서 본 시스템이 정확히 의미를 분별하여 메타데이타를 생성한 예제 단어 수이다.

### 4.3 비교 시스템

본 논문에서 제안하는 단어 의미의 중의성을 고려한 메타데이타 생성 방법의 성능을 객관적으로 비교하기 위해 기존 연구인 SemTag에서 제안하는 알고리즘을 구현하여 성능 평가 기준으로 사용하였다. SemTag은 본 논문의 2.3절과 2.4절에서 설명한 바와 같이 애매한 단어의 의미를 앞, 뒤의 10개 단어, 즉 21개의 단어가 Taxonomy의 노드에 포함되는 빈도로써 의미를 결정한다. 노드에 나타난 단어의 빈도 계산식은 정보검색에서의 tf-idf 계산 시 가장 많이 사용하고 정확도가 높은 식 (7)를 사용하였다.

$$V_i = \frac{(0.5+0.5\frac{tf(i)}{tf_{\max}})(\log\frac{n}{df(i)})}{\sqrt{\sum_{d_i \in D}((0.5+0.5\frac{tf(i)}{tf_{\max}})^2(\log\frac{n}{df(i)})^2)}} \tag{7}$$

$V_i$는 단어 $d_i$가 문서 $D$에서 가지는 중요도이다. $tf(i)$는 문서 $D$에 단어 $d_i$가 나타난 수이며, $df(i)$는 단어 $d_i$를 가지고 있는 문서의 수이다. $tf_{\max}$는 문서 $D$에 있는 모든 단어들의 출연 빈도 값 중에서 가장 큰 값으로 정규화에 이용된다. 본 비교실험은 문서 $D$를 하나의 노드에 포함되는 문서집합으로 보았으며, 따라서 각 Taxonomy의 노드별로 벡터 공간 모델을 생성하고 애매한 단어와 가장 거리가 가까운 노드를 메타데이타로 부여하였다. 애매한 단어의 의미를 결정짓는 상황정보와 노드와의 유사도는 식 (8)을 사용하였다.

$$\cos\vartheta = \frac{\vec{D}\cdot\vec{T}}{|\vec{D}||\vec{T}|} = \frac{\sum(W_{di}\times W_{ti})}{|\vec{D}||\vec{T}|} \tag{8}$$

유사도 계산은 코사인 기법을 사용하며, $W_{di}$는 학습 문서 $d_i$의 가중치이고, $W_{ti}$는 실험할 테스트 문서 $t_i$의 가중치이다. 이와 같이 실험 문서로부터 추출한 상황정보와 노드 벡터간의 유사도를 판단하여 가장 높은 유사도를 가지는 노드로 메타데이타를 부여하였다. 그러나 전산분야 온톨로지에 표현되는 개체 수에 비해 SemTag에서 제안하는 상황정보에 포함되는 개체의 수가 너무 적다. 실험결과 정확도가 너무 낮게 나왔고, 따라서 SemTag에서 제안하는 불용어를 포함한 21개의 단어는 상황정보에 애매한 의미의 단어가 하나 이상 존재하지 않을 때 적용이 가능했다. 상황정보에 포함된 애매한 단어 이외의 단어들이 명확히 Taxonomy 노드에 포함될 때 해당 노드로 유사도가 높아지기 때문이다. 그러므로 SemTag에서 제안한 알고리즘은 SemTag에서 사용한 도메인 온톨로지에 적용되었을 때 정확도가 높으며, 도메인에 한정된다는 단점을 입증할 수 있다.

본 비교 실험에서는 객관적인 평가를 위해 SemTag의 상황정보를 본 논문에서 제안한 동일한 방법으로 추출한다. 따라서 본 논문의 성능 평가를 위한 실험 결과는 도메인 온톨로지, 학습 데이타, 실험 데이타, 그리고 실험 데이타에서 추출한 상황정보를 동일하게 하여 실험하였다.

### 4.4 실험 결과

표 3은 SemTag과의 비교를 통한 의미 중의성을 가진 실험 대상 단어에 대한 정확한 메타데이타 생성 개수에 따른 정확도를 나타낸 표이다.

표 3 비교 실험을 통한 메타데이타 생성의 정확도

| | 알고리즘 | 평균 정확도 (%) |
|---|---|---|
| SemTag | IR (tf-idf) | 70.27 |
| Our Study | HMM | 88.02 |

표 3에서 볼 수 있듯이 제안한 알고리즘이 SemTag에 비하여 약 18%정도 더 나은 성능을 보였다. 18%의 만족할 만한 성능 향상을 보인 이유는 애매한 단어의 의미를 명확히 결정짓기 위해 SemTag에서 제안한 상황정보 보다 범위를 확장했고 상황정보에 포함된 개체간의 순서 확률을 이용하여 독립된 개체가 해당 클래스 영역에 자주 사용되는 의미로 분포가 치우치는 것을 방지하였기 때문이다. 그림 9는 한 문서 내에서 의미 중의성을 가진 실험 대상 단어에 대한 정확한 메타데이타 생성 개수에 따른 정확도를 나타낸 그래프이다.

그림 9에서 각 의미가 애매한 단어별로 성능 향상도가 일정한 것을 볼 수 있는데, 이 이유는 의미가 애매한 해당 단어가 온톨로지 클래스별로 존재할 확률이 일정하기 때문이다. 즉, 일반적으로 여러 분야에서 많이 사용되는 단어인 frequency, network, 그리고 tree에 대해서는 SemTag과 본 연구에서 제안하는 방법의 정확도가 비교적 낮음을 볼 수 있다. 그러나 특정 분야에서 전문적인 의미를 갖는 clustering, domain, semantic, 그리고 model 등의 단어는 비교적 높은 정확도로 차이가 남을 확인할 수 있다.

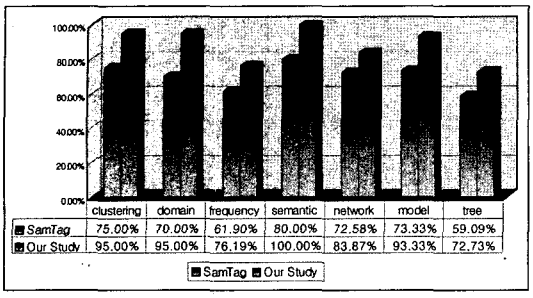



그림 9 의미가 애매한 단어의 메타데이타 생성에 대한 정확도

표 4는 각 의미가 애매한 실험 대상 단어 당 정확도를 나타낸다. 표4의 의미가 애매한 단어는 실험 데이타 중 해당 의미의 출현 빈도가 높아 성능 평가에 정확성을 높여준 단어를 추출한 것이다. 순서적으로 존재할 수 있는 단어란 온톨로지에 인스턴스로 존재할 수 있는 단어 집합에 순서를 매긴 것이다. 온톨로지를 이용한 메타데이타 생성 연구들을 보면, 대부분의 연구들이 도메인에 한정되어 연구의 활용도가 적다. 즉, 온톨로지에 개체를 정의하지 않으면 해당 정보를 소프트웨어 에이전트는 인식할 수 없다. 본 시스템에서는 이러한 문제점을 조금이나마 해결하고자 온톨로지에 정의된 인스턴스를 최소 단위의 단어로 나누고, 학습 데이타를 통해서 인스턴스로 존재할 수 있는 단어 간의 순서 모델로서, 개체들의 조합으로 생성될 수 있는 정보들을 추가하였다. 이 결과 표 4와 같이 여러 분야에서 사용되는 일반적인 단어에 대해 정확한 메타데이타를 생성할 수 있었다.

여기서 실험 데이타는 학습 데이타와 메타데이타가 생성되는 영역, 즉 클래스가 동일하다. 따라서 학습 데이타의 의미 결정 양상은 실험 데이타와 동일하다고 가정 할 수 있다. 실험결과 전체적으로 실험 데이타에 빈도가 높은 의미일수록 그 정확도가 높게 나타났으며, SemTag과 본 연구에서 비슷한 비율의 차이로 정확도를 보인 것을 표 4를 통해 확인할 수 있다. 그러므로 본 시스템은 학습 데이타에 대하여 충분히 학습을 하였으며, 학습한 내용에 대하여 적절한 성능을 보이고 있다고 결론을 내릴 수 있다. 이는 전산분야 온톨로지 도메인이 아닌 다른 도메인에 대하여 메타데이타를 생성 할 경우, 해당 영역에 대한 순서적으로 존재할 수 있는 단어 확률 모델을 생성하는 것만으로 해당 영역에 적합한 메타데이타 생성의 성능을 가질 것으로 예측 할 수 있다. 또한 빈도가 높은 의미에 대하여 정확도가 높은 이유는 학습 단계에서 클래스에 무관하게 사용빈도가 높은 단어에 대하여 효과적으로 학습했다는 데에서도 찾을 수 있다.

표 4 의미가 애매한 단어 당 메타데이타 생성에 대한 정확도

| 의미가 애매한 단어 | 메타데이타 클래스 | SemTag | Our Study |
|---|---|---|---|
| | | 정확도 (%) | 정확도 (%) |
| clustering | Machine Learning | 80.0 | 100.0 |
| | Database | 70.0 | 90.0 |
| domain | Machine Learning | 60.0 | 90.0 |
| | Network | 80.0 | 100.0 |
| frequency | Machine Learning | 50.0 | 100.0 |
| | Fuzzy | 70.0 | 70.0 |
| | Linguistics | 60.0 | 60.0 |
| | Network | 66.6 | 75.0 |
| semantic | Linguistics | 60.0 | 100.0 |
| | Database | 100.0 | 100.0 |
| network | Machine Learning | 83.3 | 100.0 |
| | Fuzzy | 50.0 | 75.0 |
| | Linguistics | 66.6 | 100.0 |
| | Pattern Recognition | 75.0 | 50.0 |
| | Reasoning | 87.5 | 62.5 |
| | Robot | 100.0 | 100.0 |
| | Supervised Learning | 50.0 | 87.5 |
| | Network | 70.0 | 100.0 |
| model | Machine Learning | 60.0 | 100.0 |
| | Neural Network | 80.0 | 80.0 |
| | Database | 80.0 | 100.0 |
| tree | Machine Learning | 60.0 | 70.0 |
| | Heuristic | 62.5 | 100.0 |
| | Network | 50.0 | 70.0 |
| | Database | 70.0 | 40.0 |

SemTag의 알고리즘에서 적용한 21개의 단어로 구성된 상황정보는 애매한 단어의 의미를 결정짓기에는 부족하다. 사용빈도가 높은 단어는 온톨로지의 각 클래스에 포함되는 빈도 역시 높기 때문에 사용빈도가 적은 단어가 특정 클래스에서 높은 빈도를 갖지 않으면 정확한 메타데이타를 생성하기 어렵다. 표 4에서 본 시스템보다 높은 정확도를 가진 클래스 Pattern Recognition의 "network", 클래스 Reasoning의 "network", 클래스 Database의 "tree"는 해당 영역에 자주 사용되는 의미로 분포가 치우쳐서 나타난 결과로 분석된다. 이 세 클래스에서는 이 단어들이 독립적인 개체로 사용되지 않는다. 예를 들어 클래스 Pattern Recognition에서의 "network"는 주로 "neural network"라는 용어로 개념을 정의하고 있으며, 이 용어는 다른 클래스(예: Machine Learning, Fuzzy, Linguistics, Reasoning 등)에서도 많이 이 순서로 사용되므로 본 연구의 방법으로는 특정한 한 클래스에서 높은 가중치를 보이지 못한다. 즉 순서 모델의 특징을 부각할 수 없는 개체들은 가중치가 여러 순서 모델로 분산 되므로 본 연구에서 더 낮은 정

확도를 보임을 알 수 있다. 하지만 SemTag에서는 이 단어가 많은 문서에서 특정 클래스로만 빈도가 제일 높게 되면 그 클래스로 분류된다. 따라서 본 연구에서 정의한 상황정보를 통해 SemTag은 다른 도메인에 대해서도 70%라는 정확도를 보일 수 있었으며, 두 개 이상의 단어가 연속으로 출현할 수 있는 확률 적용이 본 논문에서 제안하는 방법의 우수성을 입증하였다.

본 시스템은 시맨틱 웹을 구현하기 위해 온톨로지에 존재하는 개체들 간의 의미관계를 확률로써 나타냈고, 온톨로지의 문제점 중 하나인 정의하지 않은 개체의 일부에 대해서도 인식이 가능하게 하였다. 또한 의미가 애매한 단어에 대해 정확한 의미를 부여하기 위해 상황정보의 적합한 범위를 제안하여 의미가 애매한 단어가 여러 개 연이어 존재하는 것과 상관없이 의미를 결정할 수 있었다. 따라서 애매한 의미의 단어와 관계없는 단어가 특정 클래스에서 높은 빈도를 나타내더라도 실제로 사용빈도가 높은 단어에 대하여 그 정확도가 높아지게 된다.

## 5. 결론 및 향후 연구

본 논문은 순차적으로 존재할 수 있는 단어의 확률 모델을 이용하여 애매한 의미의 단어에 대해 온톨로지를 기반으로 정확한 메타데이타를 생성 할 수 있는 방법을 제안한다. 제안한 방법은 은닉 마르코프 모델을 사용하여 애매한 단어의 의미를 결정하는데 적합한 상황정보의 모델을 생성한다. 이렇게 제안된 시스템은 SemTag의 알고리즘에 비하여 약 18%정도의 성능향상을 보였다.

기존의 메타데이타 생성 방법들은 문서에 포함된 정보 추출에만 중점을 두어 메타데이타 생성 시 문제점에 대해서는 다루지 못했다. 하지만 본 논문에서 제안한 단어 의미의 중의성을 고려한 메타데이타 자동 생성 연구는 온톨로지를 기반으로 각각의 단어와 관계된 단어에 대한 클래스별 학습을 통해 메타데이타의 정확성을 높일 수 있도록 설계되었다. 제안한 방법은 도메인 온톨로지만 구축하면, 다른 도메인에 대해서도 단어 의미의 중의성을 고려하여 문맥에 맞는 메타데이타를 생성할 수 있다.

향후 연구로는 주어진 문장에서 상황정보의 범위를 축소하여 메타데이타를 생성하는 방법의 개선이 요구된다. 실제로 실험 시에 제안한 상황정보는 정확도는 우수하나 도메인의 범위가 커지고, 순서적으로 존재하는 단어의 수가 증가 하였을 때 복잡도가 증가한다. 또한 대명사로 시작하는 단어의 고유명사처리나 복수형 단어들의 구별된 의미 부여를 통해 자연어의 특성에 따른 메타데이타 생성 시에 자동 인식의 정확성을 향상 시킬 수 있는 방법의 개선이 필요하다.

## 참 고 문 헌


[1] Euzenat, J., "Eight questions about Semantic Web annotations," IEEE Intelligent Systems, Vol. 17, No. 2, pp.55-62, 2002.

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

[3] Fensel, D., Hendler, J., Lieberman, H. and Wahlster, W., Spinning the Semantic Web, MIT Press, 2003.

[4] Antoniou, G. and Van Harmelen, F., A Semantic Web Primer, MIT Press, 2004.

[5] Popov, B., Kiryakov, A., Ognyanoff, D., Manov, D. and Kirilov, A., "KIM - a semantic platform for information extaction and retrieval," Journal of Natural Language Engineering, Vol. 10, Issue 3-4, pp. 375-392, 2004.

[6] Dill, S., Eiron, N., Gibson, D., Gruhl, D., Guha, R., et al., "Semtag and Seeker: Bootstrapping the semantic web via automated semantic annotation," WWW 2003, 2003.

[7] Guha, R. and McCool, R., Tap: Towards a Web of Data. http://tap.stanford.edu/.

[8] Cunningham, H., Maynard, D., Bontcheva, K. and Tablan, V., "GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications," Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02), 2002.

[9] Bontcheva, K., Maynard, D., Cunningham, H. and Saggion, H., "Using Human Language Technology for Automatic Annotation and Indexing of Digital Library Content," ECDL'2002, 2002.

[10] Miller, D., Leek, T. and Schwartz, R., "A Hidden Markov Model Information Retrieval System," Proceedings on the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 214-221, 1999.

[11] Gruber, T., "Toward Principles for the Design of Ontologies Used for Knowledge Sharing," Standford Knowledge Systems Laboratory, 1993.

[12] Smith, M., Welty, C., Deborah, L. and McGuinness, D., "OWL Web Ontology Language Guide," W3C Recommendation, 10 February 2004. http://www.w3.org/TR/2004/REC-owl-guide-20040210/.

[13] Fikes, R., Jenkins, J. and Zhou, Q., "Including Domain-Specific Reasoners with Reusable Ontologies," Proceedings of the 2003 International Conference on Information and Knowledge Engineering, 2003.

[14] Manola, F. and Miller, E., "RDF Primer," W3C Working Draft 23 January 2003.

[15] Seaborne, A., "Jena Tutorial : A Programmer's Introduction to RDQL," April 2002.

[16] Ranganathan, A. and Campbell, R., "A Middle-ware for Context-Aware Agents in Ubiquitous Com-

puting Environments," In ACM/IFIP/USENIX International Middleware Conference 2004, 2004.
[17] terms, http://www.terms.co.kr/.
[18] AI Study, http://www.aistudy.co.kr/.

최 정 화

2004년 2월 숭실대학교 정보과학대학 컴퓨터학부 졸업(학사). 2006년 2월 숭실대학교대학원 컴퓨터학과 졸업(석사). 2006년 3월~현재 숭실대학교대학원 컴퓨터학과 박사과정. 관심분야는 유비쿼터스 컴퓨팅, 시맨틱 웹, 온톨로지 추론, Semantic Annotation, 다중 에이전트 시스템 등

박 영 택

정보과학회논문지 : 소프트웨어 및 응용
제 33 권 제 3 호 참조