DOI: 10.3745/KIPSTB.2009.16-B.3.225

저자명 모호성 해결을 위한 개념망 기반 카테고리 유틸리티

김 제 민[†] · 박 영 택[††]

요 약

동명이인의 저자를 구분하는 것은 웹에서 문서 색인과 검색의 성능을 향상시킨다. 동명이인의 저자 구분은 웹사이트 상에서 같은 이름을 갖는 여러 명의 사람이 존재했을 때 야기되는 여러 가지 문제점을 해결한다. 본 논문은 동명이인의 저자 구분을 위해 개념망 기반의 카테고리 유틸리티를 제안한다. 따라서 본 논문에서는 학술회의 웹 사이트를 대상으로 제안하고자 하는 방법을 설명한다. 제안된 방법은 저자가 가지고 있는 다양한 속성(제목, 요약, 공동저자, 소속)을 반영한 저자 온톨로지와 개념망을 활용한다. 저자 온톨로지는 OWL API와 휴리스틱한 방법을 사용하여 반자동으로 구축 되었다. 저자명 모호성 해결은 개념망 기반 카테고리 유틸리티를 사용하여 저자 온톨로지 내에 존재하는 동명이인 저자(Candidate Authors)들로부터 해당 논문에 관련된 정확한 저자를 결정한다. 카테고리 유틸리티는 각각의 저자간의 intra-class 유사성 와 inter-class 비유사성을 기본적인 개념으로 하는 평가 함수다. 이에 비해 개념망 기반 카테고리 유틸리티는 모호성 해결을 위해 개념망이 갖는 개념 정보를 추가로 활용한다. 실험 결과를 분석한 결과 개념망 기반 카테고리 유틸리티가 일반적인 카테고리 유틸리티에 비교해서, 저자명 모호성 해결에 있어서 10% 정도 우수한 성능을 보였으며, 전체적으로 98%의 정확도를 보였다.

키워드 : 온톨로지, 메타데이터, 카테고리 유틸리티, 저자명 모호성 해결

WordNet-Based Category Utility Approach for Author Name Disambiguation

Je-Min Kim[†] · Young-Tack Park[††]

ABSTRACT

Author name disambiguation is essential for improving performance of document indexing, retrieval, and web search. Author name disambiguation resolves the conflict when multiple authors share the same name label. This paper introduces a novel approach which exploits ontologies and WordNet-based category utility for author name disambiguation. Our method utilizes author knowledge in the form of populated ontology that uses various types of properties: titles, abstracts and co-authors of papers and authors' affiliation. Author ontology has been constructed in the artificial intelligence and semantic web areas semi-automatically using OWL API and heuristics. Author name disambiguation determines the correct author from various candidate authors in the populated author ontology. Candidate authors are evaluated using proposed WordNet-based category utility to resolve disambiguation. Category utility is a tradeoff between intra-class similarity and inter-class dissimilarity of author instances, where author instances are described in terms of attribute-value pairs. WordNet-based category utility has been proposed to exploit concept information in WordNet for semantic analysis for disambiguation. Experiments using the WordNet-based category utility increase the number of disambiguation by about 10% compared with that of category utility, and increase the overall amount of accuracy by around 98%.

Keywords : Ontology, Metadata, Category Utility, Author Name Disambiguation

1. 서 론

시맨틱 웹 성공을 결정짓는 중요한 요소 중 하나는 엄청난 양의 시맨틱 정보의 효율적인 생성이다[1, 3, 9]. 시맨틱 웹의 특징은 웹 콘텐츠에 대해서 기계가 프로세싱 할 수 있는 형식으로 정보가 구축된다는 점이다. 따라서 시맨틱 웹 페이지는 일반 정보에 정보가 갖는 의미를 추가함으로써 이용성이 향상된다. 많은 연구자들은 웹 페이지에 시맨틱 정보를 수동 또는 자동으로 추가 할 수 있는 방법을 연구해오고 있다[1, 2, 6, 11]. 수동으로 시맨틱 정보를 추가하는 것은 작업이 고되며, 많은 시간을 요구한다. 따라서 비용이 크고, 여러 사람이 작업하므로 일관적이지 못한 시맨틱 정보가 생성될 수 있기 때문에 자동으로 시맨틱 정보를 추가할 수 있는 방법이 필요하다.

객체의 이름은 사람명, 지역명, 기관명처럼 단어나 단어의

※ 본 논문은 숭실대학교의 지원을 받아 작성되었습니다.
† 준 회 원 : 숭실대학교 컴퓨터학과 박사과정
†† 정 회 원 : 숭실대학교 컴퓨터학과 교수
논문접수 : 2008년 10월 28일
수 정 일 : 1차 2009년 1월 30일
심사완료 : 2009년 1월 30일

조합으로 구성된다. 객체의 이름은 여러 어플리케이션에서 필요하며, 시맨틱 정보를 추가해야하는 작업에 있어서 중요한 요소가 되고 있다[1-3, 5, 7, 8]. 특히, 동명이인에 대한 시맨틱 정보 생성은 정보 검색과 웹 검색의 성능 향상에 중요한 요소다.

시맨틱 정보 생성에 있어서 해결해야할 문제점 중의 하나는 여러 가지 모호한 상황(동명이인, 동음이의)에서 올바른 시맨틱 정보를 결정하는 것 이다. 본 논문은 동명이인으로 발생하는 모호성 해결을 위해 개념망 기반의 카테고리 유틸리티를 제안한다. 본 논문의 목적은 논문을 작성한 정확한 저자를 판단하고, 논문의 타이틀과 요약 페이지를 저자의 홈페이지에 연결하는 것이다. 따라서 본 논문에서는 해외 학술회의 웹 사이트를 대상으로 제안하고자하는 방법을 설명하며, 저자를 판단하는 근거로 저자명, 논문 타이틀, 요약이 주어진다고 가정한다. 이러한 방법은 모호성이 야기하는 문제점들을 해결 할 수 있는 방안을 마련한다.

헤르도투스(Herodotus)는 동명이인의 저자를 구분하여 올바른 시맨틱 정보를 추가해주는 어플리케이션이다. 헤르도투스는 저자가 가지고 있는 다양한 속성(제목, 요약, 공동저자, 소속)을 반영한 저자 온톨로지와 개념망을 활용한다. 이러한 저자 온톨로지는 여러 학술회의 사이트를 참조하여 OWL API와 휴리스틱한 방법을 사용하여 반자동으로 구축되었다. 헤르도투스에 새로운 논문이 입력되면, 온톨로지에 존재하는 동명이인의 저자(Candidate Authors)들과 입력된 논문의 저자간의 유사성이 개념망 기반 카테고리 유틸리티 사용하여 평가된다. 카테고리 유틸리티는 각각의 저자간의 intra-class 유사성 와 inter-class 비유사성을 기본적인 개념으로 하는 평가 함수다. 이에 비해 개념망 기반 카테고리 유틸리티는 모호성 해결을 위해 온톨로지가 갖는 분류 정보(taxonomy)를 추가로 활용한다.

본 논문은 제안하는 개념망 기반 카테고리 유틸리티, 헤르도투스에 대한 전체적인 구조 및 사용된 알고리즘에 대한 설명과 헤르도투스의 실험 결과 분석으로 구성된다.

2. 관련 연구

현재 온톨로지를 기반으로 객체의 모호성을 해결하고자 하는 다양한 연구가 진행되고 있다. Hassel[1] 은 온톨로지를 기반으로 정확한 객체를 판단하기 위한 단서를 제공하는 방법을 제안했다. 이 방법은 웹 사이트인 DBLP로부터 추출된 온톨로지를 사용하여 비구조적인 문서 안에 존재하는 객체들의 모호성을 해결한다. 본 논문에서 제안하는 방법은 DBLP로부터의 정보를 이용하여 반자동으로 구축된 온톨로지를 사용한다는 측면에서는 Hassel의 방법과 유사하지만, 객체의 모호성 해결을 위해 Hassel의 방법이 온톨로지에 존재하는 객체의 속성 값들을 참조하는 반면에 개념망 기반의 평가 함수를 사용하여 시맨틱 유사도를 계산한다는 점이 다르다. 헤르도투스는 동명이인 저자 집합에 대해 각 저자를 그 저자의 논문을 기반으로 군집으로 분류하고 개념망 기반의 평가 함수를 사용하여 저자명 모호성을 해결한다.

Han[2] 은 논문 인용을 바탕으로 저자명 모호성 해결을 위해 교사 학습(Supervised Learning)을 적용한 방법을 연구하였다. 이 방법은 동명이인 저자 구분을 위한 요소로 논문 제목과 공동 저자를 바탕으로 나이브 베이지안 분류기를 사용하였다. 헤르도투스가 사용하는 개념망 기반 카테고리 유틸리티 역시 베이지안 규칙 기반의 확률 모델을 사용하지만 비교사학습을 적용하였으며 개념 분류, 개념 계층, 개념간의 관계와 같은 시맨틱 정보를 활용한다는 점에서 차이를 보인다.

SemTag[3]은 대용량의 온톨로지 안에 존재하는 인스턴스의 모호성을 해결하기 위해 개념 분류 기반의 모호성 해결 알고리즘을 사용한다. SemTag은 특정 정보(Context)가 특정 인스턴스에 적절한지 추측하기 위해 유사도 함수를 사용한다. SemTag은 온톨로지 내에 존재하는 개념과 인스턴스의 모호성을 해결한다. 헤르도투스 역시 비교사 학습 방법인 군집화를 적용하여 동명이인과 같이 모호한 개념이 존재하는 온톨로지에 새로운 인스턴스를 정확하게 연결한다.

이 외에도 헤르도투스처럼 온톨로지에 시맨틱 정보를 연결(Semantic Annotation)하기 위한 다양한 연구가 진행되고 있다[5, 7, 8, 10]. 이중에서 앞에 기술한 2개의 연구(Hassel, Han)는 본 연구의 목적과 적용 범위 및 모호성 해결을 위해 사용되는 기본 구분 요소(공동저자, 논문제목, 소속, 요약)가 거의 일치한다. 그러나 본 논문에서 제안하는 기법은 이전 연구방법과는 차이점을 보인다. 따라서 본 논문의 실험 부분에서 이러한 선행 연구와 제안하는 방법에 대한 비교실험 결과를 설명한다.

3. 개념망 기반 카테고리 유틸리티(WordNet-based Category Utility)

헤르도투스는 동명이인 저자를 찾아내고, 이중에 정확한 저자를 판별하기 위해 저자 및 연구 분야에 대한 온톨로지와 개념망을 사용한다. 개념망 기반 카테고리 유틸리티(WordNet-based Category Utility)는 개체군과 개체사이의 일반적인 유사도뿐 아니라 의미적 유사도를 측정하는데 유용한 함수이다. 본 논문에서는 서로 다른 동명이인 저자 중 입력된 논문을 작성한 저자를 찾기 위해 개념망 기반 카테고리 유틸리티를 활용했다.

3.1 카테고리 유틸리티(Category Utility)

사람의 경우 사전에 학습된 개념들은 새로운 개념을 구별하기 위한 원론적인 영역을 제공한다. 카테고리 유틸리티는 하나의 객체와 여러 개의 객체군 간의 유사도를 계산한다. 이 함수는 같은 군집 내에 속한 객체들의 유사도(Intra-Class Similarity)개념과 서로 다른 군집에 속한 객체들 간의 비유사도(Inter-Class Dissimilarity)개념을 사용한다. 각 객체는 속성-값 쌍들로 구성된다. 예를 들어 해르도투스의 경우 객체는 각 저자가 되고, 속성은 논문 제목, 논문 요약,

소속 등이 되며, 값은 속성에 대응되는 정보(해당 저자의 논문 제목, 논문 요약 및 소속등)가 된다. 같은 군집 내에 속한 객체들의 유사도에는 조건부 확률 P(Ai=Vij|Ck)가 반영된다.(Ai = Vij는 속성-값 쌍, Ck는 군집을 의미) 서로 다른 군집에 속한 객체들 간의 비유사도는 확률 P(Ck|Ai=Vij)을 반영한다.

다음 수식 1은 군집 내에 속한 객체들의 유사도와 다른 군집에 속한 객체들 간의 비유사도의 균형적인 관계를 표현하고 있다.

$$\sum_i \sum_j P(A_i = V_{ij}) P(C_k \mid A_i = V_{ij}) P(A_i = V_{ij} \mid C_k) \quad (1)$$

각각의 i, j, k에 대해서 베이지안 규칙을 적용했을 때 P(Ai = Vij)P(Ck|Ai=Vij)=P(Ck)P(Ai=Vij|Ck) 이므로 수식 1은 다음과 같이 치환된다.

$$P(C_k) \sum_i \sum_j P(A_i = V_{ij} \mid C_k)^2 \quad (2)$$

3.2 개념망 기반 카테고리 유틸리티

카테고리 유틸리티는 많은 어플리케이션에 성공적으로 적용되고 있다. 그러나 이 함수는 유사도 계산에 있어서 오직 속성 값들 간의 정확한 매치를 기반으로 하고 있다. 반면 제안하는 개념망 기반 카테고리 유틸리티는 유사도 계산에 있어서 개념망의 계층적인 구조를 추가로 반영한다. 다음은 개념망 기반 카테고리 유틸리티에 적용된 유사도 계산의 세 가지 정의다.

- Exact-Match similarity: 속성 값의 정확한 매치를 기반으로 두 객체간의 유사도 계산
- Sibling similarity: 두 객체의 속성 값이 속하는 공통된 개념을 기반으로 두 객체간의 유사도 계산
- Subsumption similarity: 두 객체의 속성 값 사이에 존재하는 포함 관계를 기반으로 두 객체간의 유사도 계산

개념망 기반 카테고리 유틸리티의 유사도 계산에 사용된 개념망은 WordNet[13]이다. WordNet의 분류된 개념 정보와 계층 정보는 개념망 기반 카테고리 유틸리티의 계산결과를 향상시키는데 중요한 역할을 한다.

P(Ck|Ai=Vij)P(Ai=Vij|Ck) 계산에 있어서, 일반적인 카테고리 유틸리티는 속성 값들(Ai=Vij) 간의 정확한 매치만을 적용한다. 개념망 기반 카테고리 유틸리티는 속성 값들(Ai=Vij) 간의 정확한 매치가 존재하지 않지만 Vij와 Vik간의 의미적 관계가 존재할 경우, 두 값이 상당히 유사하다고 가정하여 객체간의 유사도를 높인다.

따라서 개념망 기반 카테고리 유틸리티의 속성 값 유사도는 크게 Vij(normal)과 Vij(semantic)로 구성된다. Vij(semantic)는 다시 Vij(subsumption)와 Vij(sibling)으로 구성된다. Vij(subsumption)은 개념망 계층에서 부모-자식 관계를 갖는 속성 값의 집합을 의미하며, Vij(sibling)은 개념망 계층에서 형제 관계를 갖는 속성 값의 집합을 의미한다.

다음 수식 3은 제안된 개념망 기반의 카테고리 유틸리티를 나타내고 있다.

$$\sum_i \sum_j P(A_i = V_{ij}) \{ P(C_k \mid A_i = V_{ij(normal)}) P(A_i = V_{ij(normal)} \mid C_k) + P(C_k \mid A_i = V_{ij(subsumption)}) P(A_i = V_{ij(subsumption)} \mid C_k) + P(C_k \mid A_i = V_{ij(sibling)}) P(A_i = V_{ij(sibling)} \mid C_k) \} \quad (3)$$

베이지안 규칙에 의해 수식 3은 아래의 수식 4와 같이 표현된다.

$$P(C_k) \sum_w \sum_i \sum_j P(A_i = V_{ij(w)} \mid C_k)^2 \quad (4)$$
$$w \in \{normal, subsumption, sibling\}$$

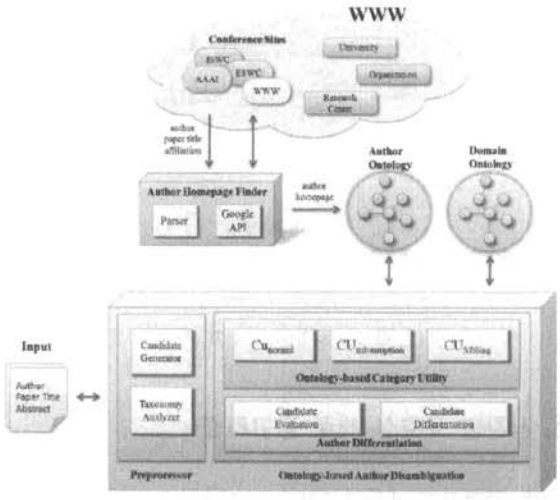
4. 헤르도투스(Herodotus)

헤르도투스는 저자가 수록된 논문을 입력 받은 후, 온톨로지를 기반으로 동명이인의 저자들을 찾아내고 제안된 개념망 기반 카테고리 유틸리티를 사용하여 이들 중에서 입력된 논문을 작성한 저자를 찾아낸다. 그리고 저자의 홈페이지에 입력 받은 논문을 연결한다.

4.1 헤르도투스의 구조

(그림 1)은 헤르도투스의 전체적인 구조를 보여준다. 헤르도투스는 개념망 기반 카테고리 유틸리티를 사용한 동명이인 저자 구분 모듈, Google API를 사용한 저자 홈페이지 탐색 모듈, 저자 온톨로지로 구성된다. 헤르도투스의 작업은 크게 3단계로 진행된다.

- 동명이인 저자 집합 생성 : 입력된 논문에 수록된 저자를 바탕으로 저자 온톨로지로부터 동명을 갖는 저자들



(그림 1) 헤르도투스의 구조

을 검색한다. 저자 온톨로지는 저자뿐만 아니라 해당 저자가 작성한 논문 타이틀과 요약에 대한 정보를 포함한다. 이 단계에서는 저자 온톨로지로부터 입력된 저자명과 매치되는 저자들의 집합을 구성하는 작업을 한다.

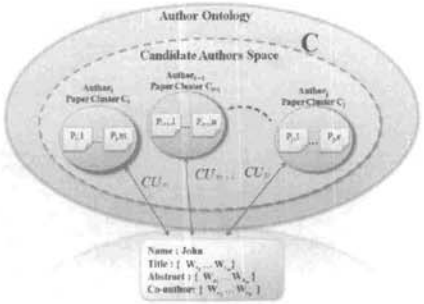
- 저자명 모호성 해결 : 정확한 저자를 찾아내기 위해, 제안된 개념망 기반 카테고리 유틸리티를 사용하여 입력된 저자와 동명이인 저자 집합들 간의 유사도를 계산한다. 보다 정확한 유사도를 계산하기 위해 속성 값들의 정확한 매치 뿐 아니라 개념적 분류 관계가 적용된다.
- 저자 홈페이지 검색 : 논문 제목과 요약문이 담긴 페이지를 해당 저자의 홈페이지에 자동으로 연결한다. 먼저 Google API를 사용하여 저자의 모든 웹 페이지를 수집한 후 휴리스틱 메소드를 사용하여 저자의 정확한 홈페이지를 찾는다.

4.2 모호성 해결

(그림 2)는 동명이인 저자 집합(Candidate Authors)을 보여준다. 동명이인 저자 집합은 각 저자가 작성한 논문들의 정보를 기반으로 각각의 군집으로 나뉜다. 각 논문은 제목, 요약, 공동 저자와 같은 속성을 가지고 있으며, 이러한 속성들이 가지고 있는 값들은 각 군집을 특징짓는 요소가 된다.

헤르도투스는 동명이인 저자 집합 C 중에서 입력된 저자 D와 C에 속하는 Ck중 가장 높은 확률을 고려한다. 따라서 입력된 저자 D와 매치된 Ck중 가장 높은 조건부 확률을 갖는 Ck를 찾게되며, 이러한 Ck를 MAP(*maximum a posteriori*)이라고 한다. 각 동명이인 저자들과 입력된 저자와의 MAP을 계산하기 위해 개념망 기반 카테고리 유틸리티를 사용한다. 다음 수식 5는 MAP을 계산하는 방법을 보여준다.

$$C_{MAP} \equiv \arg\max_{C_k \in C} P(C_k \mid D)\ P(D \mid C_k) = \arg\max_{C_k \in C} \sum_i \sum_j P(C_k \mid A_i = V_{ij})\, P(A_i = V_{ij} \mid C_k) \quad (5)$$



(그림 2) 동명이인 저자 집합

5. 저자명 모호성 해결 알고리즘

이번 장에서는 제안하고자하는 저자명 모호성 해결을 위한 알고리즘을 설명한다. (그림 3)은 헤르도투스에서 사용되는 저자명 모호성 해결을 위한 전체적인 알고리즘을 보여준다. 이러한 알고리즘을 구성하기 위해 몇 가지 기호를 정의하였다.

```
Algorithm Disambiguation (InputAuthor, AuthorOntology, DomainOntology) {
  Let C be the candidate space
  for(each candidate author Ci in C) {
    Create a paper cluster Pi
    for(each paper Pij) {
      for(each attribute in {title, abstract, affiliation, co-author} ) {
        Create value attribute vectors, Titleij, Authorij, Affiliationij, and CoAuthorij
  }}}
  for(each value attribute vectors) {
    With Title, Abstract, Affiliation, and CoAuthor
    Calculate Titleij(normal), Titleij(subsumption), Ttileij(sibling)
        Abstractij(normal), Abstractij(subsumption), Abstractij(sibling)
        Affiliationij(normal), Affiliationij(subsumption), Affiliationij(sibling)
        CoAuthorij(normal), CoAuthorij(subsumption), CoAuthorij(sibling)
  }
  for(each candidate author Ci in C) {
    Calculate CUi(normal), CUi(subsumption), CUi(sibling)
    CUi = Σk CUik, k ∈ {normal, subsumption, sibling}
  CUw = argmax CUi
  Return (URL of Cw)
}
```

(그림 3) 저자명 모호성 해결 알고리즘

- O : 동명이인 저자를 포함하는 저자 온톨로지
- I : 입력된 저자명과 저자의 논문이 가지고 있는 속성값(제목, 요약, 공동 저자)
- C : 동명이인 저자 집합
- Ci : 입력된 저자명과 동일한 저자
- Pi : 각 저자의 논문을 바탕으로 동명이인 저자 집합을 구성하는 군집
- Pij : 각 군집이 가지고 있는 속성

각 속성은 여러 개의 속성 값을 가지므로 백터 형식으로 구성되며, *Titleij, Abstractij, Affiliationij, CoAuthorAij*와 같이 표현된다. (그림 2)는 동명이인 저자 집합과 연관된 특징들을 보여준다.

개념망 기반 카테고리 유틸리티(*CUi*)는 I와 각 Pi를 바탕으로 계산된다. 개념망 기반 카테고리 유틸리티는 일반적인 카테고리 유틸리티에 개념망이 갖는 시맨틱 정보를 적용함으로써 각 속성 값의 정확한 매치 기반의 유사성*(Vijf(normal))* 뿐 아니라 공통 개념 유사성*(Vijf(sibling))*, 계층적 유사성*(Vijf(subsumption))*을 계산하게 된다. 따라서 속성 값이 정확하게 매치되지 않더라도 개념망의 의미적인 정보를 바탕으로 유사도를 측정할 수 있다.

5.1 동명이인 저자 집합의 표현

개념망 기반 카테고리 유틸리티는 엄밀하게 말해서 각 군집간 유사성을 계산하는 함수다. 이 함수를 동명이인 저자 간의 유사성을 계산하는 함수로 활용하기 위해, 헤르도투스는 카테고리 유틸리티의 Ci를 입력된 저자명과 동일하면서 동명이인 저자 집합을 구성하는 군집으로 표현하였다. 각 군집은 저자의 논문을 바탕으로 구성되며, 헤르도투스는 속성에 해당하는 논문 제목과 요약문에 대해 WordNet을 사용하여 필수 단어를 추출한다. 따라서 추출된 키워드는 논문 제목과 요약문에 대한 속성 값이 된다. 이와 같은 방법으로 또 다른 속성인 소속과 공동 저자에 대한 속성 값이 저자

온톨로지를 통해 추출된다. 따라서 *Pij*는 각 속성과 속성 값을 가지는 벡터로 표현된다.

$$C_i = \sum_j P_{ij}$$
$$P_{ij} = \sum_{Attribute} Attribute_{ij}, \quad Attribute \in \{Title, Author, Affiliation, CoAuthor\}$$
$$Attribute_{ij} = \bigcup_w Attribute_{ijw}, \quad w \in \{normal, subsumption, sibling\} \quad (6)$$

각 속성 벡터는 개념망 기반 카테고리 유틸리티를 활용하기 위해 세부 벡터로 다시 나뉜다. 먼저 각 군집의 *Pij*와 입력된 *I*의 *ij*가 일치한다면 *normal* 백터에 해당 속성 값을 삽입한다. 남은 군집의 *Pij*중 입력된 *I*와 의미적인 관계(개념적 포함관계, 개념적 형제관계)가 형성된다면, Subsumption 벡터, Sibling 벡터에 해당 속성 값을 삽입한다. 이러한 의미적인 관계는 저자 온톨로지와 WordNet에서 찾을 수 있다. 따라서 각 속성 벡터는 세부적으로 *normal* 벡터, Subsumption 벡터, Sibling 벡터로 구성된다. 수식 6은 속성 벡터의 구성을 표현한다.

5.2 전처리

헤르도투스는 먼저 논문 제목, 요약, 공동저자 속성에 대한 속성 값을 저자 온톨로지와 WordNet을 통해 추출한다. 논문 제목이나 요약은 보통 하나의 긴 문장과 문서로 구성된다. 따라서 헤르도투스는 제목과 요약에서 필수적인 키워드를 속성 값으로 추출하기위해서 TFIDF 방식을 사용한다. TFIDF는 TF를 보완하기 위해서 단어의 출현빈도인 TF와 그 단어가 출현한 문서의 빈도인 DF의 역수를 곱해준 통계치 이다. TFIDF가 의미를 가지는 이유는 적은 문서에 등장하지만 해당 문서들에서는 자주 등장하는 단어가 핵심단어일 것이라는 추론에 근거한다. 많은 실험에서 TFIDF는 TF보다 좋은 결과를 보여 왔다.[12] 두 번째 단계에서는 저자 온톨로지와 WordNet을 사용하여 *I*와 *Pij*의 속성 값의 일치성과 의미적 관계를 고려하여 속성 벡터를 구성한다.

5.3 개념망 기반 카테고리 유틸리티를 사용한 저자명 모호성 해결

본 절에서는 개념망 기반 카테고리 유틸리티를 평가하기에 앞서 간단한 예제를 통해 핵심 개념을 설명한다.

예제 1) 동명이인 집합은 전처리 과정을 통해 두 개의 군집으로 분류되었다. 두 개의 군집은 속성 $_a$를 가지며, 각 군집의 $_a$의 속성 벡터 C^1a와 C^2a는 다음과 같이 4개와 3개의 키워드로 구성된다. *Ina*는 입력된 저자정보를 의미한다.

C1a = {ontology(0.3), semantic web(0.2), DL(0.1), subsumption reasoning(0.2)}

C2a = {ontology(0.3), semantic web(0.2), semantic search(0.2)}

Ina = {ontology, semantic web, DL reasoning, realization}

개념망에는 개념간의 다음과 같은 관계가 존재한다.

ontology⊑semantic web, DL⊑semantic web, semantic search⊑semantic web, DL reasoning⊑DL subsumption reasoning⊑DL reasoning, realization⊑DL reasoning

따라서 MAP은 다음과 같이 계산된다.

$P(C^1) = 0.5$, $P(C^2) = 0.5$

$P(nomal|C1) = P(a=ontology\ nomal|C^1)^2 + P(a=semantic\ web\ nomal|C^1)^2 + P(a=DL\ reasoning\ normal|C^1)^2 + P(a=realization\ nomal|C^1)^2 = 0.3^2 + 0.2^2 + 0 + 0 = 0.13$

$P(sub|C1) = P(a=DL\ reasoning\ sub|C^1)^2 + P(a=realization\ sub|C^1)^2 = 0.1^2 + 0 = 0.01$

$P(sibling|C1) = P(a=realization\ sibling|C^1)^2 = 0.2^2 = 0.04$

$P(C^1)\Sigma\Sigma\Sigma P(a=Vij(w)|C^1)^2 = 0.0178$

$P(nomal|C2) = P(a=ontology\ nomal|C^2)^2 + P(a=semantic\ web\ nomal|C^2)^2 + P(a=DL\ reasoning\ normal|\ C^2)^2 + P(a=realization\ nomal|C^2)^2 = 0.3^2 + 0.2^2 + 0 + 0 = 0.13$

$P(sub|C2) = P(a=DL\ reasoning\ sub|C^2)^2 + P(a=realization\ sub|C^2)^2 = 0$

$P(sibling|C2) = P(a=DL\ reasoning\ sibling|C^2)^2 + P(a=realization\ sibling|C^2)^2 = 0$

$P(C2)\Sigma\Sigma\Sigma P(a=Vij(w)|C^2)^2 = 0.00845$

$\therefore$ cMAP = $\text{argmax}\{C^1, C^2\} = C^1$

6. 데이터 집합 구성

헤르도투스가 동작하기 위해서는 입력된 저자 정보의 유사도 계산에 필요한 군집들을 제공하는 데이터 집합(Test Data Set)이 필요하다. 이러한 데이터 집합은 저자 온톨로지를 통해 구성되는데, 저자 온톨로지는 여러 해외 학술회의로부터 얻어진 저자 정보를 기반으로 OWL API를 사용하여 반자동으로 구축된다. 즉, 온톨로지의 스키마 부분은 기존의 DBLP에서 사용한 온톨로지 스키마를 참조하여 수동으로 구축되었고, 온톨로지 인스턴스 정보는 OWL API와 학술회의 페이지 파서를 통해 자동으로 구축되었다. 본 논문에서 사용한 저자 온톨로지는 저자 클래스(Author Class), 논문 클래스(Paper Class), 공동 저자 클래스(Co-Author)로 구성되는데 이 중 저자 클래스는 저자의 이름, 저자의 논문, 소속, 홈페이지를 속성(Property)으로 갖게 된다. 저자 온톨로지와 함께 본 논문에서 사용되는 WordNet은 실세계의 용어들이 개념적으로 분류되어, 계층적으로 표현되었다.

헤르도투스에서 사용된 저자 온톨로지를 구축하기 위해 학술회의 사이트인 AAAI, ISWC, ESWC, WWW를 참조하여 6000개의 논문을 수집하였으며, 그 결과 데이터 집합을 구성하기 위한 3,000여명의 저자가 수집되어 온톨로지로 구축되었다. 다음 (그림 4)는 저자 온톨로지의 도식을 보여준다.

$\text{AUTHOR} \sqsubseteq \text{PERSON}$
$\text{COAUTHOR} \sqsubseteq \text{PERSON}$
$\top \sqsubseteq \forall \text{hasPublication}^{-1}.\text{AUTHOR}$
$\top \sqsubseteq \forall \text{hasPublication}.\text{PAPER}$
$\top \sqsubseteq \forall \text{name}^{-1}.\text{AUTHOR}$
$\top \sqsubseteq \forall \text{name.xsd:string}$
$\top \sqsubseteq \forall \text{affiliation}^{-1}.\text{AUTHOR}$
$\top \sqsubseteq \forall \text{affiliation.xsd:string}$
$\top \sqsubseteq \forall \text{homepage}^{-1}.\text{AUTHOR}$
$\top \sqsubseteq \forall \text{homepage.xsd:string}$
$\top \sqsubseteq \forall \text{isCoAuthorWith}^{-1}.\text{COAUTHOR}$
$\top \sqsubseteq \forall \text{isCoAuthorWith}.\text{AUTHOR}$
$\top \sqsubseteq \forall \text{ofPaper}^{-1}.\text{COAUTHOR}$
$\top \sqsubseteq \forall \text{ofPaper}.\text{PAPER}$
$\top \sqsubseteq \forall \text{hasAuthor}^{-1}.\text{PAPER}$
$\top \sqsubseteq \forall \text{hasAuthor}.\text{AUTHOR}$
$\top \sqsubseteq \forall \text{title}^{-1}.\text{PAPER}$
$\top \sqsubseteq \forall \text{title.xsd:string}$
$\top \sqsubseteq \forall \text{abstract}^{-1}.\text{PAPER}$
$\top \sqsubseteq \forall \text{abstract.xsd:string}$
$\top \sqsubseteq \forall \text{source}^{-1}.\text{PAPER}$
$\top \sqsubseteq \forall \text{source.xsd:string}$

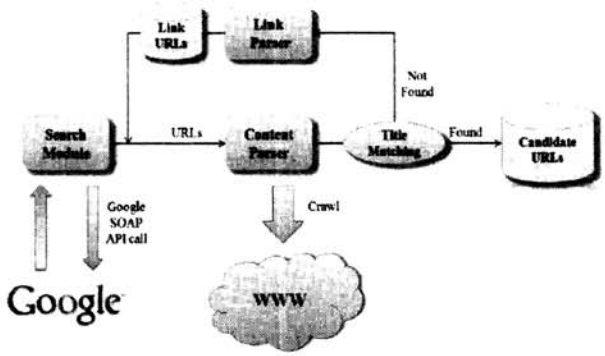
(그림 4) 저자 온톨로지의 도식

7. 저자 홈페이지 탐색

저자 온톨로지를 구축하기 위해 참조한 학술회의 사이트 대부분은 저자의 홈페이지 정보가 빠져있다. 따라서 저자 이름과 논문 제목을 가지고 저자의 정확한 홈페이지를 탐지하기 위한 모듈이 존재해야 한다. 헤르도투스에는 저자의 이름과 논문 제목을 이용하여 해당 저자의 홈페이지를 찾는 저자 홈페이지 탐색기가 구축되어 있다. 정확한 저자의 홈페이지의 탐색과 식별을 위해 Google API와 다양한 휴리스틱 메소드가 사용되었다. 헤르도투스의 저자 홈페이지 탐색기는 탐색과 식별 작업에 대해서 80%의 정확도를 보인다. 따라서 100%의 정확한 결과를 내기 위해서 헤르도투스 사용자는 홈페이지 탐색기를 통해 검색된 URL중 적절한 저자 홈페이지를 선택(식별 작업)해야 한다. 저자 홈페이지 탐색을 위한 전체적인 진행 과정은 (그림 5)와 같다.

저자 홈페이지 탐색기는 Google API 모듈과 파서 모듈로 구성된다. Google API 모듈은 Google에서 제공하는 검색 API(http://code.google.com/apis/soapsearch/)를 기반으로 구축되었다. 검색 단어로 저자명이 주어지면, 해당 저자명과 관련하여 pdf, ppt, ps와 같은 문서 형식을 제외한 모든 웹 페이지의 URL을 검색 결과로 출력한다.

파서 모듈은 콘텐츠 파서와 링크 파서로 구성되어 있다. 콘텐츠 파서는 주어진 URL에 존재하는 웹 페이지의 내용을 탐색하고, 홈페이지 탐색기에 입력된 논문 제목이 내용에 존재하는지 검사한다. 웹 페이지 내용에 논문 제목이 포함되어 있다면, 웹 페이지를 후보 URL 집합에 추가한다. 링크 파서는 웹 페이지 내에 존재하는 링크 정보를 수집하고, 연결된 모든 URL을 방문한다. 이때 같은 페이지에 반복되는



(그림 5) 저자 홈페이지 탐색 과정

방문을 예방하기 위해 링크 정보 집합을 이용하여 이전에 방문했던 URL에는 다시 방문하지 않는다. 링크 파서를 통해 링크 정보 집합이 만들어지면, 링크 파서는 링크 정보 집합 내에 존재하는 모든 웹 페이지의 링크 정보를 반복적으로 수집하고, 연결된 모든 URL을 방문한다. 링크 파서는 기본적으로 깊이 우선 검색(depth-first search)을 실행하며, 반복 검색 등급(Search Depth)을 2로 한 결과 81.4%의 정확도를 보였다. 그 이상의 반복 검색 등급은 더욱 정확한 검색 결과를 보이지만, 많은 시간을 소비하기 때문에 정확도와 시간의 균형을 고려하여 반복 검색 등급을 판단하였다.

따라서 헤르도투스는 데이터 집합을 위해 수집된 3,000여 명의 저자명과 6000개의 논문 제목을 저자 홈페이지 탐색기에 입력하여 해당 저자의 홈페이지 정보를 탐색 및 식별하고, 식별된 홈페이지 주소를 저자 온톨로지에 기록한다.

8. 실험 및 평가

본 논문의 핵심은 저자명 모호성을 해결하는 것이다. 이를 위해 본 논문에서는 개념망 기반 카테고리 유틸리티를 제안하였다. 따라서 제안한 개념망 기반 카테고리 유틸리티를 평가하기 위해 기존의 카테고리 유틸리티, Han이 제안한 나이브 베이지안 분류 기법, Hassel이 제안한 온톨로지 참조 기법과 비교하여 저자명 모호성 해결에 대한 실험을 실시하였다. 이 실험을 위해 학술회의 사이트인 AAAI, ISWC, ESWC, WWW를 참조하여 데이터 집합(Test Data Set)들을 만들었다. 이 데이터 집합은 239개의 동명이인 집합으로 구성되어 있으며, 이러한 집합들을 통해서 저자명 모호성 해결에 대한 정밀도(Precision)의 측정과 실험 결과에 대한 평가가 가능하다.

정밀도를 계산하는 방법은 아래의 수식 7과 같다. 먼저 t_pos는 입력된 저자 정보가 동명이인 집합 내 군집 중에서 정확한 저자의 홈페이지로 연결된 집합이고, *f_pos*는 동명이인 집합 내 군집 중에서 부정확한 저자의 홈페이지로 연결된 집합이다.

$$precision = \frac{t\_pos}{(t\_pos + f\_pos)} \tag{7}$$

제안한 개념망 기반 카테고리 유틸리티는 동명이인 저자들의 군집과 입력된 저자 정보의 공통 개념 유사도, 계층적 개념 유사도를 계산한다. 따라서 개념망의 구조가 전체 유사도 측정에 중요한 요소가 되며, 이는 올바른 개념망의 개념과 관계 정의가 중요하다는 것을 의미한다. 다음 <표 1>은 실험을 위해 사용된 동명이인 저자 집합과 저자 정보의 수다. 데이터 집합은 저자 온톨로지를 기반으로 239개의 동명이인 집합으로 구성되어 있다.

〈표 1〉 동명이인 저자 집합과 입력된 저자 정보

	동명이인 저자 집합	입력된 저자 정보
Number	239	819

개념망 기반 카테고리 유틸리티의 모호성 해결에 대한 실험 결과를 기존 카테고리 유틸리티의 실행 결과와 비교했을 때 10%, 나이브 베이지안 분류 기법 보다 5%, 온톨로지 참조 기법 보다는 7% 정도 더 나은 성능을 보였다. 개념망 기반 카테고리 유틸리티는 819개의 저자 정보 중 808건에 대해 정확한 저자명 모호성 해결을 보였다. 반면 카테고리 유틸리티는 723건, 나이브 베이지안 분류기법은 759건, 온톨로지 참조기법은 748건에 대해 정확한 저자명 모호성 해결을 보였다. <표 2>는 저자명 모호성 해결에 대한 실험 결과를 정리한 것이다.

<표 3>은 본 논문에서 제안하는 개념망 기반 카테고리 유틸리티와 기존 연구에서 제안된 방법을 통해 저자명 모호성 해결에 걸리는 시간을 측정한 것이다. 제안한 기법은 카테고리 유틸리티와 나이브 베이지안 기법에 비해 정밀도는 높지만 시간 소모가 2배 이상 크다. 그 이유로는 이 두 기법에 비해 Sibling similarity와 Subsumption similarity를 구하기 위한 추가적인 연산이 일어나기 때문이다. 또한 추가 연산을 위해 참조되는 데이터의 양도 두 기법에 비해평균 2.5배 정도 요구된다. 따라서 시스템이 자주 구동되지 않으면서 정확도가 중요한 저자명 모호성 해결 문제에 있어서는 제안된 방법이 더 효율성을 갖지만(일정한 기간마다 입력된 저자들의 모호성을 한꺼번에 해결한 후, 그 정보를 어플리케이션에 적용하는 경우), 인터넷 문서 색인처럼 항상 시스템이 구동 돼야 하는 어플리케이션(어플리케이션에서 요청할 때마다 저자의 모호성을 해결해야하는 경우)에서는 나이브 베이지안 분류기법이 더 효과적일 것이다.

〈표 2〉 개념망 기반 카테고리 유틸리티와 기존 연구에서 제안된 방법 적용에 따른 모호성 해결 정밀도 비교

	카테고리 유틸리티	개념망 기반 카테고리 유틸리티	나이브 베이지안 분류 기법	온톨로지 참조 휴리스틱
모호성 해결의 정확성	723	808	759	748
모호성 해결시도	819	819	819	819
입력된 저자정보	819	819	819	819
정밀도	0.88	0.98	0.93	0.91

〈표 3〉 저자명 모호성 해결에 소요되는 시간 비교

	카테고리 유틸리티	온톨로지 기반 카테고리 유틸리티	나이브 베이지안 분류 기법	온톨로지 참조 휴리스틱
소요 시간	324ms	519ms	217ms	2198ms
모호성 해결시도	819	819	819	819

9. 결 론

본 논문에서 저자명 모호성 해결을 위한 개념망 기반 카테고리 유틸리티를 제안하였다. 개념망 기반 카테고리 유틸리티는 유사도 함수인 카테고리 유틸리티의 성능을 증가시키기 위해 온톨로지가 담고 있는 시맨틱 정보를 유사도 측정에 적용하였다. 본 논문에서 제안하는 개념망 기반 카테고리 유틸리티는 서로 같은 군집 내에 속했을 때의 유사도(Intra-Class Similarity)와 서로 다른 군집에 속했을 때의 비유사도(Inter-Class Dissimilarity)의 개념을 사용하고, 이에 추가로 동명이인과 같이 모호한 정보를 해결하기 위해 개념망의 개념 분류, 개념 계층에 대한 정보를 추가로 적용하였다. 실험을 통해 기존 연구에서 제안한 방법들과 비교 평가한 결과, 제안한 개념망 기반 카테고리 유틸리티가 저자명 모호성에 대해 5~10%정도 더 나은 성능을 보였다.

본 논문에서 제안한 개념망 기반 카테고리 유틸리티는 개념간의 다중 관계 및 일반적인 데이터 타입 값과 관계를 가질 때의 상황에 대해 고려하지 않고 있다. 따라서 향후에는 이러한 점이 적용되어 성능이 개선된 개념망 기반 카테고리 유틸리티가 연구될 것이다.

참 고 문 헌

[1] Joseph Hassell, Boanerges Aleman-Meza, I.Budak Arpinar, "Ontology-Driven Automatic Entity Disambiguation in Unstructured Text", 5th International Semantic Web Conference, Athens, GA, USA, 2006.

[2] Hui Han, Lee Giles, Hongyuan Zha, "Two Supervised Learning Approaches for Name Disambiguation in Author Citations", 4th Joint Conference on Digital Libraries, Tucson, Arizona, USA, 2004.

[3] Stephen Dill, Nadav Eiron, David Gibson, Daniel Gruhl, R.Guha, Anant Jhingran, Tapas Kanungo, Sridhar Rajagopalan, Andrew Tomkins, John A. Tomlin, Jason Y. Zien, "SemTag and Seeker: Bootstrapping the semantic web via automated semantic annotation", 20th World Wide Web conference, Budapest, Hungary, 2003.

[4] Douglas H. Fisher, "Knowledge Acquisition Via Incremental Conceptual Clustering", Machine Learning, Vol.2, pp.139-172, 1987.

[5] Thamar Solorio, "Improvement of Named Entity Tagging by Machine Learning", Technical Report CCC-04-004, Coordinacin de Ciencias Computacionales, 2004.

[6] Michael Erdmann, Alexander Maedche, Hans-Peter Schnurr, Steffen Staab, "From Manual to Semi-automatic Semantic Annotation: About Ontology-based Text Annotation Tools",

Proceedings of the COLING 2000 Workshop on Semantic Annotation and Intelligent Content, Luxembourg, 2000.

[7] Norberto Fernandez Garcia, Jose Maria Blazquez del Toro, Luis Sanchez Fernandez and Ansgar Bernardi, "IdentityRank: Named Entity Disambiguation in the Context of the NEWS Project", 4th European Semantic Web Conference, Innsbruck, Austria, 2007.

[8] Hui Han, Hongyuan Zha, C. Lee Giles, "Name Disambiguation in Author Citations using a K-way Spectral Clustering Method", 5th Joint Conference on Digital Libraries, Denver, Colorado, USA, 2004.

[9] Alexiei Dingli, Fabio Ciravegna, Yorick Wilks, "Automatic Semantic Annotation using Unsupervised Information Extraction and Integration", K-CAP 2003 Workshop on Knowledge Markup and Semantic Annotation, 2003.

[10] Ziming Zhuang, Rohit Wagle, C. Lee Giles, "What's There and What's Not? Focused Crawling for Missing Documents in Digital Libraries", 5th Joint Conference on Digital Libraries, Denver, Colorado, USA 2004.

[11] Borislav Popov, Atanas Kiryakov, Angel Kirilov, Dimitar Manov, Damyan Ognyanoff, Miroslav Goranov, "KIM - Semantic Annotation Platform", Proceeding of the 2nd International Semantic Web Conference, Sanibel Island, Florida, 2003.

[12] Yiming Yang, and Jan O.Pedersen, "A comparative study on Feature Selection in Text Categorization", Proceedings of ICML-97, 14th International Conference on Machine Learning, 1997.

[13] WordNet, http://wordnet.princeton.edu/

김 제 민

e-mail : kimjemins@hotmail.com

2001년 숭실대학교 컴퓨터학과(학사)

2004년 숭실대학교 컴퓨터학과(석사)

2004년~현 재 숭실대학교 컴퓨터학과 박사과정

관심분야 : 인공지능, 시맨틱 웹, 유비쿼터스 컴퓨팅

박 영 택

e-mail : park@ssu.ac.kr

1978년 서울대학교 전자공학과(학사)

1980년 KAIST 전산학(석사)

1992년 Univ. of Illinois at Urbana Champaign (박사)

1981년~현 재 숭실대학교 컴퓨터학과 교수

관심분야 : 인공지능, 에이전트, 전문가 시스템, 시멘틱 웹