http://www.jsebs.org
ISSN: 2288-3908

The Journal of Society for e-Business Studies
Vol.20, No.4, November 2015, pp.61-76
http://dx.doi.org/10.7838/jsebs.2015.20.4.061

위키피디아 링크를 이용한 랭크 기반 개념 계층구조의 자동 구축

Automated Development of Rank-Based Concept Hierarchical Structures using Wikipedia Links

이가희(Ga-hee Lee)*, 김한준(Han-joon Kim)**

초 록

흔히 대용량 텍스트 데이터의 분류를 위한 인덱싱 데이터 구조로서 계층 개념 트리가 활용된다. 본 논문은 개념 계층구조를 자동적으로 구축하기 위해 위키피디아를 이용한 일반성 랭크 기반 기법을 제안한다. 이것의 목적은 위키피디아 문서를 하나의 개념으로 정의하여 이들 간의 계층적 위상관계를 생성하는 것이다. 이를 위해 위키피디아 문서들 간의 링크 개수를 주요 인자로 하여 개념 일반성을 가늠하는 랭킹함수를 고안하였으며, 이를 활용하여 개념 간 확률적 포함관계를 산출함으로써 안정적인 개념 간 계층 구조를 생성한다. 결과적으로 계층적 관계를 담은 개념쌍은 DAG 구조로 시각화 된다. Open Directory Project 계층구조를 사용한 성능 분석을 통해 제안 기법이 기준 기법에 비해 성능이 우수하며 고품질 계층 관계를 안정적으로 추출할 수 있음을 확인하였다.

ABSTRACT

In general, we have utilized the hierarchical concept tree as a crucial data structure for indexing huge amount of textual data. This paper proposes a generality rank-based method that can automatically develop hierarchical concept structures with the Wikipedia data. The goal of the method is to regard each of Wikipedia articles as a concept and to generate hierarchical relationships among concepts. In order to estimate the generality of concepts, we have devised a special ranking function that mainly uses the number of hyperlinks among Wikipedia articles. The ranking function is effectively used for computing the probabilistic subsumption among concepts, which allows to generate relatively more stable hierarchical structures. Eventually, a set of concept pairs with hierarchical relationship is visualized as a DAG (directed acyclic graph). Through the empirical analysis using the concept hierarchy of Open Directory Project, we proved that the proposed method outperforms a representative baseline method and it can automatically extract concept hierarchies with high accuracy.

키워드 : 텍스트 마이닝, 정보검색, 개념 계층구조, 인덱싱, 개념, 위키피디아, DAG
Text Mining, Information Retrieval, Concept Hierarchy, Indexing, Concept, Wikipedia, DAG

이 논문은 2013년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (NRF-2013R1A2A2A01017030).

* First Author, School of Electrical and Computer Engineering, University of Seoul(larah1007@uos.ac.kr)
** Corresponding Author, School of Electrical and Computer Engineering, University of Seoul (khj@uos.ac.kr)

Received: 2015-08-18, Review completed: 2015-09-17, Accepted: 2015-10-17

1. 서 론

빅데이터란 디지털 환경에서 빠른 속도로 생성되는 대규모의 정형, 반정형, 비정형 데이터를 일컫는다[11, 19]. 빅데이터의 규모는 2011년 약 1.8제타바이트에 달하였고 10년 이후 그 규모가 50배를 초과할 것으로 추정되면서 기존의 데이터를 인덱싱하고 처리하는 방식으로는 관리와 분석이 어려워서 효과적인 빅데이터 관리 방안에 대한 연구가 활발하다[1, 12, 13].

특히 빅데이터에 포함되는 비/반정형 텍스트 데이터의 비중이 폭발적으로 증가함에 따라 텍스트 데이터를 효과적으로 인덱싱하는 연구가 매우 중요하다. 일반적으로 텍스트 문서는 개념 카테고리의 위상 수준에 따라 계층적으로 분류된다. 계층 분류라 함은 카테고리의 개념적 수준에 따라 일반적(general)인 의미를 가진 개념을 가지는 문서들은 상위 수준에 할당하고 좁은(specific) 의미를 가지는 문서들은 하위 수준에 할당함을 의미한다. 여기서 데이터 인덱싱에 근간이 되는 자료구조를 '개념 계층구조(concept hierarchy)'라 칭한다.

이러한 개념 계층구조의 좋은 예로서 ODP (Open Directory Project)[16]를 들 수 있다. ODP는 웹문서를 분류하기 위하여 모범적이면서 포괄적인 웹 디렉토리로 평가받고 있으며, 16개의 최상위 카테고리(개념)를 필두로 하여 그 하위에 1백만 개 이상의 카테고리가 계층적으로 구성되어 있다. 이는 심볼릭 링크 (symbolic link)를 사용하여 각 카테고리가 하나 이상의 상위 노드를 가지기 때문에 정확하게는 DAG(directed acyclic graph) 구조라 할 수 있다[17]. 이러한 개념 계층구조의 구축을 위해서는 각 카테고리를 정의하는 과정과 정의된 카테고리들을 계층적으로 구성하는 과정이 필요한데 이는 전문가에 의한 수작업에 크게 의존하게 된다.

본 논문은 world knowledge 수준의 영문 위키피디아(Wikipedia)를 활용하여 개념 계층구조를 자동 구축하는 기법을 제안한다. 위키피디아[22]는 웹 기반 개방형 백과사전으로서, 영문 위키피디아의 경우 현재 약 490여만 개의 문서를 포함하고 있다. 각 위키피디아 문서들은 관련 전문가들에 의해 편집, 수정되고 있기 때문에 그 품질이 매우 높은 편이다. 이 때문에 위키피디아 문서들은 개념 기반 인덱싱을 위한 개념의 원천으로 사용하기에 충분하다. 최근 텍스트마이닝 분야에서 고품질의 위키피디아 문서를 활용하여 개념 기반의 문서 인덱싱 또는 자동 문서 분류 기법에 관한 연구가 활발하게 진행되고 있다[5, 8, 9, 15, 20, 23].

본 논문은 개념 계층구조에 사용하기 적절한 개념을 선별하고, 개념간의 의미적인 계층 관계를 판별한 뒤 이를 시각화하는 기법을 제안한다. 본 논문의 구성은 다음과 같다. 제2장에서는 위키피디아 데이터와 개념 간 계층구조를 생성하는 기존의 연구 방법들을 제시하면서 본 연구의 배경이 되는 기술적 요소를 소개한다. 제3장에서는 제안 기법을 상세히 논의하고 제4장에서는 제안 기법에 의해 도출되는 개념 계층구조를 시각적으로 보여주고 그것의 정확도를 평가한다. 마지막으로 제5장에서는 결론과 향후 연구에 대해 기술한다.

2. 연구 배경

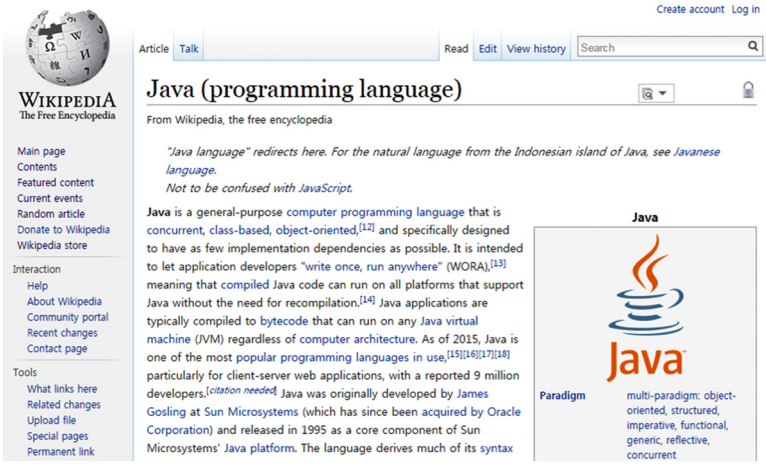
2.1 위키피디아의 구조 및 기능

위키피디아아는 위키미디아 재단이 운영하

는 하나의 백과사전이며, 이에 관여하는 모든 사용자들이 정보의 생산자 혹은 가공자로 참여하여 지속적으로 편집하고 있기 때문에 집단지성(collective intelligence)을 대표하는 온톨로지(ontology) 또는 지식베이스(knowledge base)라 할 수 있다. 위키피디아의 구조는 크게 제목과 본문으로 나눠지며, 본문 내에는 인포박스(infobox), 앵커텍스트(anchor text)등이 포함하고 있다.

본 논문에서는 하나의 위키피디아 문서를 하나의 개념으로 정의한다. 이때, 위키피디아 문서 내의 제목은 본문의 내용을 가장 잘 설명하는 대표적인 개념으로 애매모호함이 없고, 분명한 의미를 결정하는 명사로 표현되는 것이 일반적이다. 결국 위키피디아 문서의 제목은 개념의 명칭으로 사용된다. 이때 위키피디아 제목이 한 가지 이상의 의미를 가지는 경우에는 <Figure 1>에서 보는 바와 같이 제목 뒤에 괄호 문구를 추가하여 의미를 명확히 하게 된다.

결국 위키피디아 문서의 본문은 해당 개념을 정의하는데 기여하며, 추가적으로 인포박스와 앵커텍스트는 개념의 명확성을 높이는 역할을 한다. 인포박스는 개념에 대한 부가적인 정보를 문서 내에 테이블 형식으로 간략히 제시함으로써 개념에 대한 이해를 돕는다. 앵커텍스트는 본문에서 언급된 주요 용어를 설명하기 위해 해당 용어를 제목으로 하는 다른 위키피디아 문서와 연결함으로써 개념의 이해를 높인다. 여기서 문서들의 연결 수단인 하이퍼링크는 진입링크(incoming link)와 진출링크(outgoing link)로 구분한다. 진입링크는 해당 개념을 앵커텍스트로 하여 다른 위키피디아 문서들과 연결해주는 링크이며, 진출링크는 본문에 앵커텍스트로서 작성된 용어를 설명하기 위해 다른 문서로 진출하는 링크이다. 예를 들어, 위키피디아 문서 A, B가 있고 개념 A의 위키피디아 문서에 앵커텍스트로 작성된 개념 B가 존재할 때, 개념 B의 위키피디아 문서에는 개념 A로부터의 진입링크가 있고, 반대로 개념 A의 위키피디아 문서는 개념 B를 가리키는 진출링크가 있는 것이다.



〈Figure 1〉 Structure of Wikipedia Articles

한편 위키피디아는 단일 문서의 크기, 문서 편집에 참여한 편집자의 수, 편집 횟수, 문서 열람 횟수 등과 같은 통계적 정보를 함께 제공하고 있다. 본 논문에서는 이러한 메타정보가 고품질의 후보 개념을 선정하기 위한 필터링 휴리스틱에 활용된다.

2.2 개념 계층구조 생성 기법

개념 계층구조는 정보검색(information retrieval)과 텍스트마이닝(text mining) 분야에서 매우 중요한 자료 구조로 활용될 수 있다. 단어간의 계층적 관계를 의미 중심으로 정의하기 위해 다양한 방법들이 제시되어 왔다. 흔히 워드넷(WordNet)[14]과 같은 외부 시소러스(thesaurus)를 이용하여 단어간 관계가 도출하거나, 외부 지식 없이 단어간 계층적 관계 산출을 위해 문서 내 단어의 출현 빈도수, 기준 단어를 중심으로 한 주변 단어들과의 관계, 특정 단어가 출현하는 문헌 빈도수 등을 산출하여 그 관계를 도출 할 수 있다. 특히 Sanderson and Croft[18]에서는 단어를 개념으로 정의하고, 문헌빈도수에 기반하여 식 (1)에 따라 단어간 개념적 포함관계를 산출한다. 구체적으로 식 (1)을 만족하면 단어 t_i가 단어 t_j보다 개념적으로 상위 수준의 단어인 것으로 평가한다.

$$\Pr(t_i|t_j) \geq 0.8, \quad \Pr(t_i|t_j) > \Pr(t_j|t_i) \tag{1}$$

$\Pr(t_i|t_j)$는 단어 t_j가 출현하는 문서 집합에 단어 t_i도 출현하는 확률을 의미한다. 확률값 $\Pr(t_i|t_j)$와 $\Pr(t_j|t_i)$을 계산하여 $\Pr(t_i|t_j)$가 $\Pr(t_j|t_i)$보다 확률이 높으면서 임계값 0.8 이상의 확률 값을 가지면, t_i는 단어 t_j의 상위 수준의 단어로 정의하게 된다. 즉 상위 수준의 단어가 출현하는 대부분의 문서는 하위 수준의 단어가 출현한다는 것을 의미한다. 이 기법은 동일한 문서에 동시에 출현하는 단어 t_i와 t_j에 대해서는 의미있는 계층 관계를 도출할 수 있다.

Lee and Kim[10]에서는 진입 링크에 기반하여 개념 계층관계를 산출하는 기법을 제안하였다. 이 기법은 위키피디아를 이용하여 위키피디아 문서를 하나의 개념으로 정의하여 Sanderson and Croft[18]에서 제시된 기법을 변형하여 개념 계층구조를 구성하였다. 이것의 기본 아이디어는 '단어가 문서에 출현함'을 '단어가 문서에 인용됨'으로 간주하여, 위키피디아 문서내의 진입 링크 정보를 적극 활용하였다[2, 4]. 위키피디아 문서의 앵커텍스트에 해당하는 개념 c_i에서 다른 위키피디아 문서인 개념 c_j를 가리키는 링크가 존재한다면, 'c_i가 c_j를 인용함' 또는 'c_j가 c_i에 인용됨'으로 해석하여 식 (2)~식 (3)에 따라 진입 링크 기반 개념 간 계층 관계를 결정할 수 있다.

$$\Pr(c_i|c_j) = \Pr(I(c_i) \supset I(c_j)) = \frac{|I(c_i) \cap I(c_j)|}{|I(c_j)|} \tag{2}$$

$$|\Pr(c_i|c_j) - \Pr(c_j|c_i)| \geq 0.1, \quad \Pr(c_i|c_j) > \Pr(c_j|c_i) \tag{3}$$

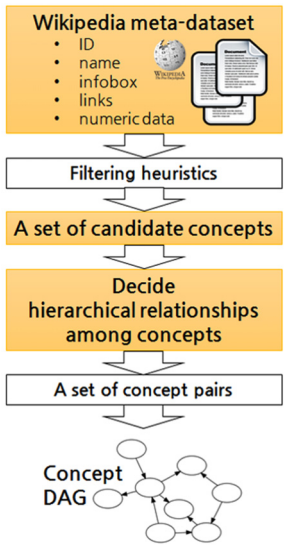
여기서 $I(c)$는 개념 c에 대한 진입링크 집합, $|I(c_j)|$는 개념 c_j의 진입링크 개수를 의미한다. $|I(c_i) \cap I(c_j)|$는 개념 c_i와 c_j의 진입 링크 집합인 $I(c_i)$와 $I(c_j)$의 교집합에 대한 크기를 의미한다. 식 (3)의 확률식을 만족하면 식 (2)에

의해 도출된 $\Pr(c_i|c_j)$에서 c_i가 c_j보다 개념적으로 상위 수준인 것으로 평가하게 된다. 내부적으로 각 개념들간 식 (3)의 확률값은 부분순서열 행렬(partial ordering matrix)로 표현되며, 이를 단순화하여 개념 계층 트리가 생성된다. 여기서 식 (3)은 상위개념의 영향을 받는 정도를 산출하기 위해 제시된 것이다. 정리하면, Lee and Kim[10]에서 소개한 기법은 개념 c_i와 c_j의 계층적 관계를 위키피디아 문서 간 링크 집합의 포함관계에 근거하여 개념 계층관계를 결정하는 기법이다. 이는 적정 수준 이상의 일반성을 가지는 개념들에 대한 계층구조를 구축하기 위해 고안된 것이다.

그러나 Lee and Kim[10]의 기법은 개념 간 진입 링크에 대한 상호 포함 정도와 전체 진입 링크의 개수에 민감하게 반응하는 단점을 가진다. 간단한 예를 들면, 개념 A와 B의 진입링크 개수가 각각 1,000개, 10,000개, $I(c_A \cap c_B) = 700$이라 할 때, $\Pr(c_A|c_B)$와 $\Pr(c_B|c_A)$은 각각 0.07과 0.7로 계산된다. 이 예는 전체 링크 개수의 크기에 따라 식 (2)의 확률값이 크게 차이가 발생할 수 있음을 보여 준다. 식 (2)는 주요 인자인 진입링크 개수에 따라 크게 변동되기 때문에 상위 개념은 항상 하위 개념보다 진입 링크를 많이 포함하고 있어야 한다. 다시 말해서, 두 개념 간 계층적 위상 관계를 고려할 때 실제로 하위 개념일지라도 진입링크의 개수가 상대적으로 크다면 상위 개념으로 평가될 수 있는 것이다.

3. 랭크 기반 개념 계층 관계 생성

본 연구에서는 제2장에서 제시한 진입 링크 기반 개념 계층 관계 산출 기법을 보완하여 보다 향상된 개념 계층 관계를 생성하는 기법을 제안한다. <Figure 2>는 제안 기법의 전체적인 프로세스를 보여준다. 우선 합당한 수준의 개념 집합을 얻기 위해 양질의 개념 문서 데이터를 보유한 위키피디아를 활용한다. 여기서, 후보 개념을 올바르게 선정하기 위해서 여러 유형의 메타 정보를 취득한다. 추출된 메타 정보는 필터링 휴리스틱을 통해 고품질의 후보 개념 집합을 구성한다. 본 논문의 핵심 단계인 계층관계의 산출 과정에서 후보 개념 집합에 대하여 일반성 랭크(rank) 인자를 추가하여 개념 간 계층 관계를 평가한다. 여기서 랭크 인자는 일반성을 가늠하는 순위 수치로서 계층관계를 명확히 구성하는데 기여한다. 최종적으로, 식 (2), 식 (3)의 확률식에 의해 도출된 개념



〈Figure 2〉 The Process of Automated Development of Rank-Based Concept Hierarchical Structures using Wikipedia Links

쌍들은 DAG 구조로 시각화 되며 이는 ODP의 계층구조와 동일한 형태이다.

위키피디아 데이터로부터 후보 집합을 얻고 이들 간의 계층관계를 산출하기 위해서 과거 관련 연구에서 제시했던 가정을 포함하여 다음과 같은 가정을 정리한다.

1. 하나의 개념에 해당하는 위키피디아 문서는 반드시 한 개 존재한다. 문서에 포함된 내용물이 해당 개념을 정의하는데 기여하며, 그것의 명칭은 매핑되는 위키피디아 문서의 제목이다.
2. 위키피디아 문서의 제목이 고유명사로 지정된 것은 후보 개념집합에 포함시키지 않는다.
3. 각 개념에 매핑된 위키피디아 문서에 존재하는 앵커텍스트는 해당 문서의 개념을 정의하는데 기여해야 한다.
4. 두 개념 c_i, c_j가 존재하고 c_i가 c_j보다 더 일반적인 내용을 설명하는 개념일 때, c_i는 c_j의 상위 개념이고 c_j는 c_i의 하위 개념이다.

3.1 위키피디아 메타데이터

제안 기법은 위키피디아를 통해 적절한 개념을 얻고 이들 간의 계층구조를 구성하기 위해서, 위키피디아 개념 문서 및 그 개념과 관련한 메타 데이터를 적극 활용한다.

<Table 1>은 데이터베이스에 저장되는 위키피디아 메타데이터 속성을 설명한 것이다. 이는 주어진 개념에 대한 식별자(concept_id)와 개념 명칭(concept_name) 속성을 주축으로 하여 인포박스 정보(infobox), 링크 정보(in, out, in_set, out_set)와 통계적 정보(size, edit, editor, view) 등을 포함한다. concept_id는 위키피디아 문서에 부여하는 식별자로서 DB 테이블 생성시 주키(primary key)로 사용된다. concept_name은 위키피디아 문서의 제목인 개념 명칭을 의미한다. infobox는 인포박스 의 템플릿 유형에 대한 속성이며, 예를 들어, 인포박스 'film'은 name, director, producer, writer, story, runtime, language 등의 속성을 제공한다. 즉, 이러한 속성들로 설명이 가능한 개념들은 인포박스 'film'을 사용할 수 있다. 'film'을 사용한 개념의 예는 'Begin Again(film)', 'The Avengers (2012 film)' 등이 있다. 한편, size, edit, editor, view는 개념에 대한 수치적 메타 데이터 정보로서 각각 문서 크기, 편집 횟수, 참여 편집자 수, 문서 열람 횟수에 대한 속성이다. 아울러, in, out, in_set, out_set은 링크 정보와 관련한 속성들로서, 주어진 개념에 대한 진입링크와 진출

〈Table 1〉 Attributes and Descriptions of Wikipedia Meta Information

Attributes	Descriptions
concept_id	identifier of a given concept
concept_name	name of a given concept
infobox	template name of the infobox in a given concept
in	the number of incoming links
out	the number of outgoing links
in_set	a set of incoming links
out_set	a set of outgoing links
size	size of Wikipedia page
edit	the number of edits
editor	the number of editors
view	the number of views for a given Wikipedia article

링크의 개수와 기준 개념과 링크로 맺어진 다른 개념들의 식별자(concept_id) 집합을 저장한다. 이러한 메타 정보는 제3.2절에서 후보 개념 집합을 선정할 때 휴리스틱 필터링 인자로 활용된다. 링크 관련 속성 정보는 후보 개념 집합 선정뿐만 아니라 개념 간 계층 관계를 확률적으로 산출할 때도 이용한다.

3.2 후보 개념 집합 선정

개념 계층구조에서 사용되는 개념은 일반적 수준에 해당하는 개념이면서 의미적 품질이 높아야 한다. 이에 본 연구에서는 개념 계층구조를 구축에 필요한 후보 개념 집합을 선정하기 위하여 개념으로 적절하지 않은 문서와 품질이 낮은 문서를 제거하기 위한 6가지 필터링 휴리스틱(filtering heuristic)을 제시한다. 필터링 휴리스틱에 포함되는 위키피디아 문서는 후보 개념 집합에 포함되지 않는다.

1. 인포박스가 존재하는 문서
2. 제목이 숫자 또는 특수 문자로 시작하는 문서
3. 제목이 전치사를 동반하여 3단어 이상으로 구성된 문서
4. $\frac{count(outgoing\,link)}{count(incoming\,link)} \geq 0.3$인 문서
5. $count(backlink) \leq 500$인 문서
6. 개념에 대한 속성 size, edit, editor, view를 정규화한 값 중 하나가 임계값 미만인 문서

첫 번째 휴리스틱과 관련하여, 인포박스(infobox)는 개념에 대한 부가적인 정보를 테이블 형식으로 제시하기 때문에 고유명사 수준의 개념을 설명하는 위키피디아 문서에 사용되는 것이 일반적이다. 예를 들어 <Figure 3>을 보면, 개념 'Larry Page'의 인포박스에서 제공하는 내용으로서 이름, 생년월일, 출생지, 학력, 소속, 직책 등이 있다.

그리고 개념 '1991', '300(film)', '@midnight'처럼 위키피디아 문서 제목이 숫자 또는 특수 문자로 시작하는 문서는 일반적 개념으로 보기 어렵기 때문에 이는 후보 개념에서 제외한다. 또한 개념 'List of United Kingdom locations', 'Romeo and Juliet'처럼 전치사를 동반하여 3단어 이상으로 구성된 문서는 대부분 고유명사 수준에 해당한다.

〈Figure 3〉 Example of Infobox for the Concept 'Larry Page'

한편 한 문서에서 가질 수 있는 링크로 진입 링크와 진출 링크가 있는데, 이들 링크는 상호 간 개수의 차이에 따라 개념이 가지는 의미의 크기가 달라진다. 즉, 기준 문서가 가질 수 있는 링크 집합 중에서 진입 링크보다 진출 링크가 많다면 기준 문서에 대한 개념을 설명하기 위해 다른 문서(개념)을 차용한 것을 의미한다.

또한 절대적인 진입 링크 집합의 규모가 적기 때문에, 이는 다른 문서들에게 제공되어야 할 개념에 대한 정보가 적다는 것을 의미한다. 본 연구에서는 이러한 고찰에 따라 진출 링크 개수에 대한 진입 링크 개수의 비율이 임계비율 이상인 문서와 진입 링크 개수가 임계값 이하인 문서를 후보 개념에서 제외시킨다. 위키피디아 데이터에 대한 면밀한 실험 결과 적정한 임계비율과 임계값은 각각 0.3과 500인 것으로 결정되었다.

마지막으로 각 위키피디아 문서에 대한 통계적 메타 정보는 해당 문서를 수치적으로 표현된 데이터로 문서의 성질을 직관적으로 알 수 있는 속성들이다. 이에 보다 문서들 간 상대적인 척도를 측정하기 위해 문서 크기(size), 편집횟수(edit), 참여편집자수(editor), 문서열람횟수(view)를 0에서 1사이의 값으로 정규화 한다. 이때 정규화 값들 중 하나라도 임계비율보다 작으면 후보 개념에서 제외시킨다. 구체적으로, 식 (4)와 같이 문서 크기인 $size$에 대해서 정규화식 $Nsize$를 정의할 수 있다[3].

$$Nsize = \frac{\log(size) - \log(\min(size))}{\log(\max(size)) - \log(\min(size))} \quad (4)$$

마찬가지로 정규편집횟수($Nedit$), 정규참여편집자수($Neditor$), 정규문서열람횟수($Nview$)의 경우에도 식 (4)와 동일한 형식으로 정규화 한다. 결과적으로, 한 개의 위키피디아 문서에 대하여 4개의 정규화 값을 얻게 된다. 최종적으로 후보 개념을 얻기 위해 설정한 임계비율은 속성별 정규화 값 집합들의 평균값으로 정의한다. 즉 앞서 진행한 필터링 휴리스틱에 따라 남겨진 개념들에 대해서 속성별 정규화 값들의 평균을 임계비율로 정의한 것이다. 구체적인 절차를 정리하면, 우선 후보 개념을 선별하기 이전에 전체 개념들의 속성별 정규화 값을 정의하고, 이어서 정의된 개념들은 6가지 유형의 필터링 휴리스틱을 적용하여 후보 개념 집합을 얻는다. 이 후보 개념들은 속성별 평균 정규화 값을 계산하여 그 임계비율에 따라 최종적으로 2차 후보 개념 집합을 결정하게 된다.

3.3 랭크 기반 개념 계층 관계 생성 기법

3.3.1 개념 집합에 대한 랭크 정의

제2.2절에서 지적한 바와 같이, 진입링크 기반 개념 계층관계 생성 기법은 링크 개수에 대하여 매우 민감하게 반응하는 단점을 지닌다. 이 문제를 해결하기 위하여 본 논문에서는 사전에 개념 집합에 대한 랭크(rank)를 정의하여 링크 개수에 강인한(robust) 개념 간 계층관계를 생성하는 기법을 제안한다. 여기서 '랭크'는 개념 간의 계층적 위상관계를 결정하는 데 있어서 일반성 정도에 대한 상대적인 순위의 의미를 갖는다. 주어진 개념 c에 대한 랭크를 정의하는 랭킹 함수를 식 (5)와 같이 제안한다.

$$w(c) = \frac{\log_{10} c_{in} - \log_{10} c_{out}}{\max_{c_{\text{in}}, c_{\text{out}} \in T} (\log_{10} c'_{in} - \log_{10} c'_{out})} \quad (5)$$

여기서 c_{in}은 개념 c의 진입 링크 개수, c_{out}은 개념 c의 진출 링크 개수를 의미한다. 그리고 T는 후보 개념 집합을 나타내며, 결과적으로 $w(c)$는 개념 c의 진입링크와 진출링크의 상용로그 값의 차에 대한 순위 값을 0과 1사이의 값으로 정규화 시킨 값이다. 즉, 개념 c_i의 진입링크와 진출링크의 차가 후보개념 집합 T의

링크 차보다 값이 크다면 순위가 상위에 위치하여 상대적으로 높은 $w(c)$값을 얻게 된다. <Table 2>는 무작위로 추출한 50개의 개념에 대해서 실험한 결과, 상위 10%와 하위 10%에 해당하는 개념들의 예를 보여준다.

〈Table 2〉 Concepts Isolated with $w(c)$

top 10%	bottom 10%
Security	Six sigma
Multimedia	Virtual screening
Software	Bayesian network
Operating system	Storage (memory)
Internet	Computer crime

<Table 2>에서 보는 바와 같이, 개념들이 식 (5)에 따라 랭크가 정의되어 상위 10%의 일반적 의미를 가지는 개념들과 하위 10%의 구체적 의미를 가지는 개념들로 확연히 분별되는 것을 확인할 수 있다. <Table 2>의 결과에 따라, 본 논문에서 제안한 랭킹 함수는 개념들에 대해 합당한 랭크를 정의할 수 있고, 최종적 계층관계를 도출하는데 크게 기여할 수 있음을 실험적으로 확인하였다. 이런 맥락에서는 식 (5)는 개념의 일반성을 가늠하는 일반성 랭킹 함수라 할 수 있다. 결과적으로 이 일반성 랭킹 함수를 통해 보다 안정성 있는 개념 계층구조를 구축할 수 있게 된다.

3.3.2 개념 계층 관계 산출

식 (5)에 의한 랭크값은 각 개념의 독립적인 계층적 위상을 설명할 수는 있지만, 개념 간 계층적 위상 관계까지는 설명하지 못한다. 가령, <Table 2>의 예를 다시 보면, 'Operating system'이 상위에 랭크되어 있고, 'Bayesian network'가 하위에 랭크되어 있지만 이들 간에 계층적 관계나 의미적 연관성이 존재하는지는 알 수 없다. 식 (5)의 랭킹 함수를 개념 계층관계의 형성에 활용하기 위해서는 개념 간 포함관계 정도를 확률적으로 산출하는 작업이 선행되어야 한다. 본 논문에서는 이전 연구로서 제안된 진입 링크 기반 개념 계층구조 생성 기법에 식 (5)의 일반성 랭킹 함수를 결합시켜 보다 안정된 개념 계층 관계를 도출하는 기법을 제안한다. 기본 아이디어는 개념 c_i와 c_j가 존재할 때, 이들 간의 개념적 포함관계와 일반성 랭크를 산출하여 연관성과 계층성을 동시에 고려한 개념 계층구조를 구축하는 것이다. 이를 위한 구체적인 확률식이 식 (6)~식 (7)에 주어진다.

$$Hr(c_i, c_j) = \Pr(c_i|c_j) \bullet w(c_i) \tag{6}$$

$$|Hr(c_i, c_j) - Hr(c_j, c_i)| \geq \delta, \quad Hr(c_i, c_j) > Hr(c_j, c_i) \tag{7}$$

여기서 $Hr(c_i, c_j)$는 확률 $\Pr(c_i|c_j)$과 식 (5)의 $w(c_i)$의 곱으로 표현된다. 이는 개념 c_i, c_j가 서로 연관성이 있는 확률을 고려하고 추가적으로 랭킹함수에 의해 도출된 값을 가중치로 반영한다. 즉, $Hr(c_i, c_j)$는 '개념 c_i가 c_j의 상위 개념일 확률'을 나타낸다. $Hr(c_i, c_j)$의 상호 확률 $Hr(c_j, c_i)$는 반대로 '개념 c_j가 c_i의 상위 개념일 확률'을 나타낸다. 이러한 개념 c_i, c_j에 대한 확률값인 $Hr(c_i, c_j)$, $Hr(c_j, c_i)$을 식 (7)에 따라 상호 비교하여 최종적으로 두 개념 간 계층 관계를 평가한다.

4. 계층 관계 시각화 및 성능 평가

본 장에서는 제3장에서 제시한 기법에 따라 생성한 개념 쌍들을 시각화하는 방안과 이에 대한 성능평가에 대해 기술한다.

시각화 및 성능평가를 위해 사용한 데이터는 2015년 7월 현재 약 490여만 개의 위키피디아 문서 집합이다. 이 데이터로부터 개념 계층구조의 구축에 필요한 속성을 추출, 데이터베이스에 저장한다. 제3.1절의 필터링 휴리스틱에 따라 후보 개념을 얻는 과정에서, 식 (4)에 따라 도출된 정규화 값인 정규 문서 크기($Nsize$), 정규 편집 횟수($Nedit$), 정규 참여 편집자 수($Neditor$), 정규 문서 열람 횟수($Nview$)의 임계값은 모두 0.6으로 설정한다. 이 임계값은 식 (4)를 이용하지 않은 필터링 휴리스틱에 따라 걸러진 11,000여 개의 개념들에 대하여 식 (4)의 의한 정규화 값들의 평균으로 결정된 것이다. 식 (4)를 통해 최종적으로 선정된 3,070개의 후보 개념들이 선정되었다. 그리고 생성된 개념 계층구조의 정확도를 산출하기 위해 편의상 ODP[16]에서 제공하는 카테고리(주제) 중에서 'Computer' 카테고리에 속하는 후보 개념들만을 선택하여 성능평가를 수행하였다.

<Table 3>은 ODP 내의 Computer 카테고리에 한정한 계층구조의 예를 보여준다. 이때 ODP 내의 카테고리 명칭은 필요에 따라 중복 사용하고 있다. 예를 들어, 'Robotics'의 하위 수준에 위치하고 있는 'Building'은 Computer 카테고리가 아닌 Arts 카테고리의 하위에도 존재한다. 이는 'Robotics'의 하위 수준에 위치하는 개념들 중에서 'Building'에 해당하는 개념들을 분류함으로써 웹문서를 보다 세밀하게 분류할 수는 있기 때문이다.

〈Table 3〉 Example of the ODP Hierarchical Structure

level 1	level 2	level 3
Computer	Robotics	Buildings Arts Robots Software
	System	Handhelds Tablet PC
	Computer science	Districuted computing Computer graphics Database theory Theoretical
	Internet	Searching Broadcasting Cloud computing Email Policy Telephony
	Computer network	FidoNet Hotline
	Artificial intelligence	Support vector machine Machine learning Artificial neural network Belief network
	Computer programming	Database Disassemblers Internet
	Software	Software engineering Database Data mining Online analytical processing
	Computer hardware	Bus(computing)

4.1 시각화 결과 분석

본 논문의 제안 기법에 따라 도출된 개념 계층구조는 구체적으로 DAG(Directed Acyclic

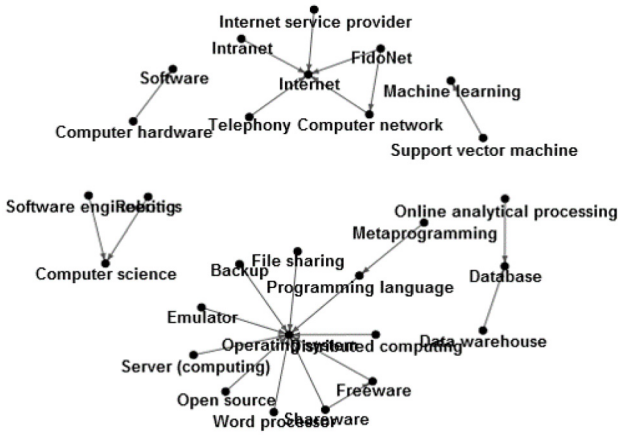
Graph)[6, 7] 구조를 가진다. 이는 하나의 개념이 1개 이상의 상위 개념을 가질 수 있기 때문이다. 본 실험에서 DAG를 기반으로 한 개념 간 계층 관계는 '상위개념 ← 하위개념'의 형태로 표현한다. <Figure 4>는 Computer 관련 카테고리와 대응되는 개념들에 대하여 제안 기법을 적용한 결과의 일부로서 상위 25개의 개념 쌍에 대한 DAG를 보여준다.

<Figure 4>의 계층구조에서 ODP의 계층구조와 상이한 부분 중의 하나는 'Online analytical processing'이 'Database'의 관계이다. ODP에서는 'Online analytical processing'과 'Database' 개념이 'Software'의 하위 개념으로서 존재하지만, 제안 기법에서는 'Online analytical processing' 개념이 'Database'의 하위 개념으로 존재한다. 최근 Database 및 Big data 분야의 발전상을 놓고 볼 때 제안 기법이 도출한 계층 관계가 보다 정확한 것으로 평가한다[21].

<Table 4>는 <Figure 4>의 DAG의 포함된 일부 개념들이 구체적으로 가지는 확률값들을 보여준다. 여기서 Hypernym, Hyponym는 상위 및 하위 개념을 의미하며, Hr 확률값은 식 (6)의 계산 결과이다. 또한 Δ_{Hr} 컬럼은 $|Hr(c_i, c_j) - Hr(c_j, c_i)|$를 의미하며, 이는 두 개의 개념 c_i와 c_j 간에 얼마나 계층 관계적인 차이가 있는지 보여주는 지표로 사용한다. 즉, Δ_{Hr}값이 클수록 계층관계가 뚜렷함을 의미한다. 본 실험에서는 그 값이 0.1 이상이면 계층관계가 큰 것으로 평가하였다. <Table 4>에서 개념 'Database', 'Online analytical processing'의 Hr값은 0.5437이고, Δ_{Hr}값이 0.5422로서 상호간 Hr값의 차이가 매우 크기 때문에, 이 개념들은 강한 계층적 관계가 있음으로 평가한

〈Table 4〉 Hierarchical Relationships by the Proposed Method

Hypernym	Hyponym	$Hr(c_i, c_j)$	Δ_{Hr}
Internet	FidoNet	0.8383	0.8018
Computer network	FidoNet	0.6518	0.5059
Database	Online analytical processing	0.5437	0.5422
Internet	Computer network	0.4663	0.3755
Operating system	Emulator	0.4213	0.4104
Machine learning	Support vector machine	0.3887	0.3862
Internet	Intranet	0.3288	0.3242
Internet	Internet service provider	0.3070	0.2871
Database	Data warehouse	0.3033	0.2970
Computer science	Robotics	0.2874	0.2271
Internet	Telephony	0.2502	0.2494
Internet	Email	0.2204	0.2037
Computing platform	Shareware	0.2054	0.1979
Computing platform	Freeware	0.2045	0.1841
Internet	File sharing	0.1975	0.1878
MIDI	Bus (computing)	0.1940	0.1806
Machine learning	Data mining	0.1830	0.1373
Database	Data mining	0.1816	0.1684
Computer science	Artificial intelligence	0.1790	0.1203
Machine learning	Bayesian network	0.1786	0.1780
Computer science	Parallel computing	0.1740	0.1640



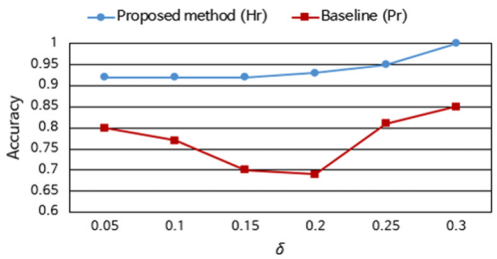
〈Figure 4〉 DAG of the Concepts Related to the 'Computer' Category

다. 또 하나의 예로서, 개념 'Machine learning', 'Bayesian network'의 Hr값은 0.1786으로 매우 낮은 값으로 도출되었지만, 이들에 대한 Δ_{Hr}는 값이 0.1보다 크므로 그 사이에 계층적 관계가 성립하는 것으로 결정한다.

4.2 성능 평가

제안 기법에 의해 도출된 개념 계층구조의 정확도를 측정하기 위해서는 결국 계층구조에 존재하는 모든 개념 간 계층 관계가 실제로 정확한지를 평가해야 한다. 하지만 아직 개념 계층관계를 객관적으로 평가하기 위한 공인된 실험 데이터 셋이 존재하지 않으므로, 제안 기법의 성능평가는 주관적 평가에 의존할 수밖에 없다. 그리고 제안 기법과 비교되는 기준 기법으로서 Lee and Kim[10]의 기법을 사용하였다. 성능평가 수치인 계층구조의 정확도는 도출된 계층구조에 존재하는 모든 개념 쌍 중에서 ODP에 존재하는 개념 쌍의 비율로 계산된다. 그리고 ODP에서 직접적으로 $C_i \rightarrow C_j$의 형태로 표현된 것을 전이적 형태인 $C_i \rightarrow X \rightarrow C_j$로 발견한 것 또한 올바른 것으로 판별하며, 그 반대의 경우도 마찬가지이다. 제안 기법과 기준 기법에 의해 생성된 개념 쌍의 수는 약 1,000개에 이르며 이에 대한 정확도를 평가하게 된다.

앞서 설명한 바와 같이, Lee and Kim[10]의 기준 기법은 개념 간 계층관계가 진입링크 개수에 따라 변동성이 크게 나타난다. 또한 특정 개념의 일반성 정도에 비해 상대적으로 적은 진입링크 개수를 가질 때에는 이와 계층 관계를 맺어야 하는 개념들을 포착하지 못한다.



〈Figure 5〉 Changes in the Accuracy of Hierarchical Relationships from Varying the Threshold Value of Delta

〈Table 5〉 Examples of Concept Pairs

Proposed Method		Baseline Method	
Hypernym	Hyponym	Hypernym	Hyponym
Computer hardware	Bus (computing)	Artificial intelligence	Computer graphics
Cloud computing	Supercomputer	Artificial intelligence	Distributed computing
Computer hardware	Distributed computing	Artificial intelligence	Parallel computing
Computer hardware	Supercomputer	Artificial intelligence	Storage (memory)
Computing platform	Email	Artificial intelligence	Support vector machine
Freeware	Demoscene	Artificial intelligence	System
Multimedia	Digital video	Artificial intelligence	Virtual reality
Distributed computing	Parallel computing	Bluetooth	Bus (computing)
Freeware	Email	Bluetooth	Calculator
Software	Open source	Bluetooth	Mobile computing

<Figure 5>는 식 (7)의 임계값 δ의 변화에 따른 계층 관계의 정확도의 변화를 보여준다. 기본적으로 제안 기법의 정확도는 90%를 초과하고 있으며, 임계값 δ의 변화에 민감하게 반응하지 않는다. 임계값 δ이 0.15와 0.2인 경우 기준 기법의 의한 계층 관계의 정확도가 떨어지는 양상을 보이지만, 제안 기법의 경우는 임계값 δ이 증가함에 따라 계층관계 정확도가 증가하는 형태를 보인다.

제안 기법에 의해 도출된 개념 쌍들 중 약 84%가 기준 기법과 동일한 개념 쌍을 도출하였다. <Table 5>는 각 기법에서 상이하게 도출한 개념 쌍의 일부를 보여준다. 기준 기법을 적용한 경우, 위키피디아 개념 문서가 일반성 강도에 비해 진입링크의 개수가 많은 경우에는 잘못된 계층관계를 도출하게 된다. 예를 들어, 위키피디아 개념 'Artificial intelligence', 'Bluetooth'은 다른 개념들보다 일반성 강도에 비해 상대적으로 진입링크 개수가 과다하게 존재하여 거의 대부분 상위 개념으로 결정되었다. 제안 기법은 진입링크 개수뿐만 아니라 일반성 강도 인자를 포함시킴으로써 기준 기법의 단점을 극복하여 안정된 개념 간 계층관계를 도출할 수 있다.

5. 결론 및 향후 연구

본 논문은 위키피디아 데이터로부터 개념 수준의 문서를 추출하여 안정된 개념 계층구조를 생성하는 기법을 제안하였다. 개념으로 정의할 수 있는 수준의 위키피디아 문서들을 선정하기 위하여 하이퍼링크 속성 정보를 포함한 다양한 유형의 위키피디아 메타 데이터를 적극 활용하였다. 선정된 후보 개념들 간의 계층 관계를 정확히 판별하기 위하여 상대적 확률 포함관계뿐만 아니라 각 개념들의 절대적 일반성 강도를 산출하여 높은 정확도의 개념 계층 구조를 안정적으로 생성하였다.

본 연구가 지향하는 개념 계층구조는 문서 인덱싱을 위한 것이므로, 궁극적으로 현재 유입되는 문서의 내용을 인식하는 개념 계층구조를 생성해야 한다. 이런 맥락에서 향후 현재 유입되는 문서의 키워드를 포함한 양질의 위키피디아

문서를 골라내어 점진적으로 개념 계층트리를 진화시켜나가는 기법을 개발할 것이다.

References

[1] Agrawal, D., Das, S., and El Abbadi, A., "Big data and cloud computing: new wine or just new bottles?," Proceedings of VLDB Endowment, Vol. 3, No. 1-2, pp. 1647-1648, 2010.

[2] Allan, J., "Automatic hypertext link typing," Proceedings of the 7th ACM Conference on Hypertext, pp. 42-52, 1996.

[3] Amiri, H., Ahmad, A., Rahgozar, M., and Oroumchian, F., "Query Expansion Using Wikipedia Concept Graph," University of Wollongong in Dubai, 2008.

[4] Conklin, J., "Hypertext: An Introduction and Survey," IEEE Computer, Vol. 20, No. 9, pp. 17-41, 1987.

[5] De Melo, G. and Weikum, G., "MENTA: Inducing multilingual taxonomies from Wikipedia," Proceedings of the 19th ACM International Conference on Information and Knowledge Management, pp. 1099-1108, 2010.

[6] Dubitzky, W., Wolkenhauer, O., Yokota, H., and Cho, K. H., "Encyclopedia of systems biology," Springer Publishing Company, 2013.

[7] Jensen, F. V., "An introduction to Bayesian Networks," UCL press, London, Vol. 210, 1996.

[8] Kim, H. and Chang, J., "A Semantic Text Model with Wikipedia-based Concept Space," The Journal of Society for e-Business Studies, Vol. 19, No. 3, pp. 107-123, 2014.

[9] Kim, H. and Hong, K., "Building Semantic Concept Networks by Wikipedia-Based Formal Concept Analysis," Advanced Science Letters, Vol. 21, No. 3, pp. 435-438, 2015.

[10] Lee, G. and Kim H., "Automated Development of Concept Hierarchy Tree using Backlink Information of Wikipedia," Database Research, Vol. 31, No. 1, pp. 40-49, 2015.

[11] Lohr, S., "The age of big data," New York Times, Vol. 11, 2012.

[12] Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., and Byers, A. H., "Big data: The next frontier for innovation, competition, and productivity," The McKinsey Global Institute, 2011.

[13] McAfee, A., Brynjolfsson, E., Daven port, T. H., Patil, D. J., and Barton, D., "Big data," The Management Revolution Harvard Bus Review, Vol. 90, No. 10, pp. 61-67, 2012.

[14] Miller, G. A., "WordNet: a lexical database for English," Communications of the ACM, Vol. 38, No. 11, pp. 39-41, ACM, 1995.

[15] Nastase, V., Strube, M., Börschinger, B., Zirn, C., and Elghafari, A., "WikiNet: A Very Large Scale Multi-Lingual Concept

Network," LREC, 2010.

[16] Open directory project, http://dmoz.org

[17] Perugini, S., "Supporting mutiple paths to objects in information hierarchies: Faceted classification, facet search, and symbolic links," Information Processing and Management, Vol. 46, No. 1, pp. 22-43, 2010.

[18] Sanderson, M. and Croft, B., "Deriving concept hierarchies from text," Proceedings of the 22nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 206-213, 1999.

[19] STAMFORD, Conn, "Gartner Says Solving Big Data Challenge Involves More Than Just Managing Volumes of Data," http://www.gartner.com/newsroom/id/1731916, 2011.

[20] Strube, M. and Ponzetto, S. P., "WikiRelate! Computing semantic relatedness using Wikipedia," AAAI, Vol. 6, pp. 1419-1424, 2006.

[21] Vassiliadis, P. and Sellis, T., "A survey of logical models for OLAP databases," ACM SIGMOD Record, Vol. 28, No. 4, pp. 64-69, 1999.

[22] Wikipedia, http://en.wikipedia.org.

[23] Xu, M., Wang, Z., Bie, R., Li, J., Zheng, C., Ke, W., and Zhou, M., "Discovering missing semantic relations between entities in Wikipedia," The Semantic Web-ISWC 2013, pp. 673-686, 2013.

저 자 소 개

이가희 (E-mail: larah1007@uos.ac.kr)
2014년 을지대학교 의료IT마케팅학과 (이학사)
2014년～현재 서울시립대학교 전자전기컴퓨터공학부 석사과정
관심분야 텍스트마이닝, 온톨로지, 기계학습, 데이터베이스, 빅데이터 분석

김한준 (E-mail: khj@uos.ac.kr)
1994년 서울대학교 계산통계학과 (이학사)
1996년 서울대학교 전산과학과 (이학석사)
2002년 서울대학교 컴퓨터공학부 (공학박사)
2002년～현재 서울시립대학교 전자전기컴퓨터공학부 교수
관심분야 텍스트마이닝, 데이터베이스, 기계학습, 정보검색, 빅데이터 분석