FolksoViz: Wikipedia 본문을 이용한 상하위 관계 기반 폭소노미 시각화 기법

(FolksoViz: A Subsumption-based Folksonomy Visualization Using the Wikipedia)

이 강 표 † 김 현 우 † 장 충 수 † 김 형 주 ††
(Kangpyo Lee) (Hyunwoo Kim) (Chungsu Jang) (Hyoung-Joo Kim)

요 약 다수의 사용자들의 협력태깅으로 생성되는 폭소노미는 웹 2.0을 이끌고 있는 대표적인 요소이다. 태그는 어떤 웹 문서를 기술하는 웹 메타데이타라고 할 수 있는데, 협력태깅으로 이루어진 태그들 사이의 의미적인 상하위 관계를 밝혀내 이를 시각화한다면, 사용자들이 문서의 메타데이타를 보다 직관적으로 이해하는 데 도움을 줄 수 있다. 이에 본 논문에서는 del.icio.us의 태그들을 대상으로 하여, Wikipedia 텍스트를 이용한 태그들간 상하위 관계 산출 기법을 제안한다. 이를 위해 태그들이 Wikipeida 텍스트상에서 출현하는 빈도수를 기반으로 태그들간 상하위 관계를 산출하는 통계적인 모델링을 제안하였고, 각각의 태그를 그에 상응하는 Wikipedia 텍스트에 매핑시키는 TSD 기법을 제안하였다. 이렇게 산출된 상하위 관계 짝들은 시각화 기법을 통하여 효과적으로 화면에 표현되었다. 실제로 우리가 제안하는 알고리즘이 태그들간의 상하위 관계들을 높은 정확도로 찾아내었음을 실험을 통해 확인하였다.

키워드 : 폭소노미, 협력태깅, 위키피디아, 시각화, 상하위관계

Abstract Folksonomy, which is created through the collaborative tagging from many users, is one of the driving factors of Web 2.0. Tags are said to be the web metadata describing a web document. If we are able to find the semantic subsumption relationships between tags created through the collaborative tagging, it can help users understand the metadata more intuitively. In this paper, targeting del.icio.us tag data, we propose a method named FolksoViz for deriving subsumption relationships between tags by using Wikipedia texts. For this purpose, we propose a statistical model for deriving subsumption relationships based on the frequency of each tag on the Wikipedia texts, and TSD (Tag Sense Disambiguation) method for mapping each tag to a corresponding Wikipedia text. The derived subsumption pairs are visualized effectively on the screen. The experiment shows that our proposed algorithm managed to find the correct subsumption pairs with high accuracy.

Key words : Folksonomy, Collaborative Tagging, Wikipedia, Visualization, Subsumption

1. 서 론

폭소노미(folksonomy)는 현재의 웹 2.0 시대를 이끌고 있는 새로운 형태의 웹 메타데이타(metadata)이다. 이는 folk(대중)와 taxonomy(택소노미)의 결합으로 이루어진 신조어로서, 대중이 만들어낸 택소노미라는 의미이다. 폭소노미는 여러 가지 측면에서 기존의 택소노미와는 차별화된 특징을 보인다. 카테고리나 디렉토리 시스템과 같이 공급자에 해당하는 전문가 집단이 미리 표준 분류 체계를 정해놓은 택소노미와는 달리, 폭소노미는 대중(folk)으로 불리우는 다수의 일반 사용자들이 직접 정보를 분류하여 만든 정보 분류 시스템을 일컫는다. 태깅(tagging)이 바로 이러한 폭소노미를 대표하는 시스

† 학생회원 : 서울대학교 컴퓨터공학부
kplee@idb.snu.ac.kr
hwkim@idb.snu.ac.kr
cschang@idb.snu.ac.kr
†† 종신회원 : 서울대학교 컴퓨터공학부 교수
hjk@snu.ac.kr
논문접수 : 2008년 1월 10일
심사완료 : 2008년 5월 11일

정보과학회논문지: 컴퓨팅의 실제 및 레터 제14권 제4호(2008.6)

템이라고 할 수 있다. 폭소노미는 흔히 협력 태깅(collaborative tagging) 혹은 소셜 태깅(social tagging)으로도 불리워진다. 택소노미와 같이 소수의 전문가 집단에 의해 논리적으로 고안된 분류 체계는 아니지만, 개개의 사용자들이 자유롭게 참여하여 취합된 분류체계가 더욱 동적이고 유연하게 웹 2.0 환경에 대응할 수 있다는 것이다.

이러한 폭소노미의 특징이 가장 잘 반영된 예가 바로 del.icio.us[1]에서 제공하는 소셜 북마킹(social bookmarking) 서비스이다. 소셜 북마킹이란 자신의 즐겨찾기를 저장하고 타인과 이를 공유하는 것으로서, 어느 한 URL에 대해서 다수의 사용자들이 자신의 즐겨찾기로 저장하고 이를 설명하는 태그들을 자유롭게 기록할 수 있다. 그림 1은 del.icio.us에서 협력 태깅이 어떻게 이루어지고 있는지를 보여주는 예이다. 웹 디자인에 대한 유용한 정보를 제공해주는 어느 인기 URL을 현재 3014명의 사용자들이 자신의 즐겨찾기로 등록해 놓았고, 사용자들이 이 URL을 설명하는 태그를 어떻게 기록하였는지가 태그 기록 히스토리(posting history)에 시간 순으로 잘 나타나 있다.

이와 같이 사용자들의 협력 태깅으로 구축된 폭소노미는 웹 문서들에 대한 훌륭한 메타데이타로 활용될 수 있다. 그러나, 현재 이러한 폭소노미를 시각화하여 사용자들에게 제공해줄 수 있는 방법론에 대해서는 연구가 미흡한 상태이다. 현재 가장 널리 이용되고 있는 태그의 시각화 방법은 Flickr[2]에서 처음 도입한 태그 구름(tag cloud)인데, 보통 각 태그들이 출현한 빈도수에 따라 상위 k개의 태그를 추출하여 나열하되, 빈도수에 비례하여 글자 크기를 할당하는 방식[3]이 널리 이용되고 있다. 하지만, 이 태그 구름은 단순히 태그의 빈도수에만 의존하고, 태그들 간의 관계에 대해서는 유용한 정보를 제공해주지 못하고 있다. 따라서, 사용자들에게 태그들간의 의미적인 관계를 시각화하여 제공해주는 것은 매우 의미 있는 작업이라고 할 수 있는데, 다음과 같이 두 가지 측면으로 살펴볼 수 있다. 첫째로, 자신의 태그를 기록하려는 사용자들에게 유용한 참고가 될 수 있다. 사용자들은 태깅에 앞서 주어진 문서에 가장 적합하고 연관성이 높다고 여겨지는 키워드들을 태그로 선택하는 인지적인 과정을 거치게 되는데[4], 이때 사용자마다 생각하는 가장 기본적인 수준에 차이(basic level variation)가 발생한다[5]. 가령, 어떤 문서가 Java의 EJB 프로그래밍에 대한 내용을 다루었다고 한다면, 어떤 사용자는 가장 기본적인 수준의 태그로 'ejb'를, 또 어떤 사용자는 'java'를, 또 다른 사용자는 더 넓은 개념으로 'programming', 혹은 'computer'등을 선택할 수 있을 것이다. 따라서, 다른 사용자들이 이미 기록해 놓은 태그들간의 의미적인 상하위 관계가 시각화되어 제공된다면, 태그를 기록하는 이들은 자신이 생각하는 가장 기본적인 수준의 태그를 결정하는 데 도움을 받을 수 있다. 둘째로, 태그들간의 관계를 시각화하면 사용자들이 문서의 메타데이타를 더 쉽고 직관적으로 이해하는 데 도움을 줄 수 있다. 현재로서는 사용자들이 기록한 수많은 태그들을 일일이 살펴보거나 태그 구름을 확인하는 방법 정도밖에 없지만, 태그들간의 시각화된 관계를 한 눈에 확인할 수 있다면, 주어진 문서의 메타데이타를 더 잘 이해하는 데 도움을 줄 수 있다. 결국, 폭소노미의 시각화는 어떤 문서의 메타데이타에 대한 직관적인 요약을 제공해줌으로써, 태그를 기록하는 사용자나 태그를 읽는 사용자 모두에게 유용하게 활용될 수 있다.

이에 본 논문에서는 태그들간의 의미적인 상하위 관

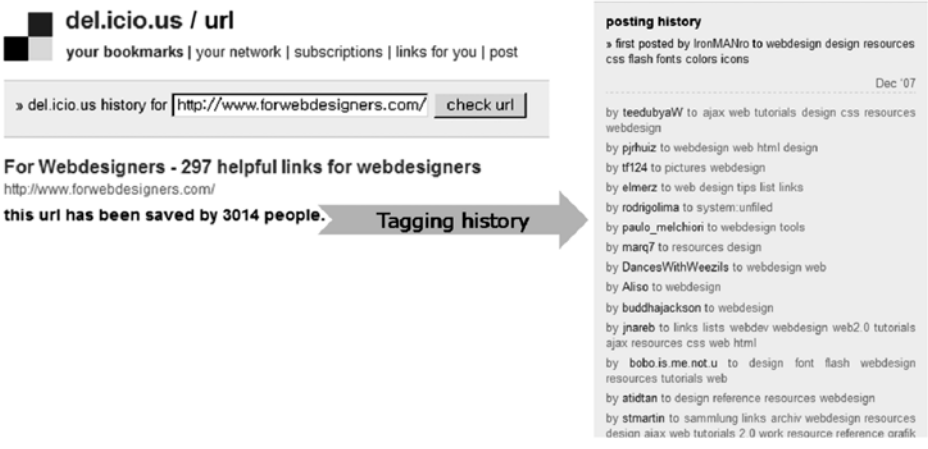

그림 1 del.icio.us에서의 협력태깅

계를 산출한 후 이를 시각화하여 화면에 표현할 수 있는 방법론에 대해 연구하였다. 본 논문의 구성은 다음과 같다. 2장에서는 용어들간의 상하위 관계를 기계적으로 산출해낼 수 있는 기존의 연구방법에 대해 소개한다. 3장에서는 del.icio.us 태그들간의 상하위 관계를 산출해내는 본 논문의 알고리즘에 대해 상세하게 논의하고, 4장에서는 산출된 상하위 관계를 효과적으로 시각화하는 방법에 대해 논의한다. 5장에서는 본 논문에서 제시하는 방법론의 성능에 대해 평가하고, 마지막으로 6장에서는 결론과 향후연구에 대해 논의한다.

2. 관련연구

용어(term)간 상하위 구조(hierarchy)를 기계적으로 산출해내는 것은 정보 검색(information retrieval) 분야에 있어서 매우 의미 있는 작업이지만, 각 용어가 지닌 복잡하고도 다양한 의미(semantics)를 다루어야 하기 때문에 그 해결이 쉽지 않은 문제에 속한다. 그 동안 용어들간의 상하위 관계를 밝히기 위해 다양한 접근 방법들이 시도되었는데, 초기의 연구에 있어서는 WordNet [6]과 같은 정형화된 시소러스(thesaurus)를 이용하거나, 용어들이 출현한 빈도수, 문맥 상의 주변 용어들과의 관계 등을 이용한 방법론들이 소개되었다. 예를 들면, [7]에서는 *Hmod(noun), Hn(noun), Freq(noun)*을 정의하여, 각각 명사의 가장 오른쪽에 위치한 수식어, 대상 명사로부터 n개의 단어 윈도우 안에서 출현하는 단어들의 엔트로피(entropy), 그리고 명사의 빈도수를 계산하여 각각 상하위 관계를 찾아내도록 하였다. 그러나, 이러한 초기의 연구들에 있어서 특히 주목할만한 접근방법은 바로 두 용어들이 함께 출현하는 코어커런스(co-occurrence)의 빈도수에 기반한 접근방법이다. [8]는 이를 상하위 관계 산출에 적용한 최초의 시도 중에 하나라고 볼 수 있는데, 어떤 용어가 다른 용어보다 더 많은 문서들에 출현할수록 그 용어는 더 일반적인 개념을 지닌다는 가정하에 용어들의 일반성(generality)과 특수성(specificity)을 그 용어들의 문서 빈도수(document frequency, DF)에 의해 결정하였다. [9]에서는 이를 다음과 같이 구체화하여 효과적으로 상하위 관계를 산출할 수 있는 모델링을 제시하였다.

$$P(x|y) >= 0.8,\ P(y|x) < 1.$$

즉, 용어 y가 출현하는 문서들의 대부분(80%)이 용어 x가 출현하는 문서들의 부분집합일 때 우리는 x가 y의 상위개념이라고 말할 수 있다는 가정이다. 이러한 가정은 x와 y가 함께 출현하는 문서들에 있어서 상당한 설득력을 지니고 있는데, 직관적으로 x가 y의 상위개념이라면 y를 기술하기 위해 x가 사용되는 빈도수가 x를 기술하기 위해 y가 사용되는 빈도수보다 많을 것이기 때문이다. 그리고 이 논문에서는 실험을 통해 48%의 올바른 상하위 관계를 찾아내는 데 성공하였음을 보여주고 있다. [10]에서는 이미지에 대한 질의 결과를 상하위 구조로 재구성하는 데 있어서 [9]의 모델링을 그대로 채택하였고, [11]에서는 Flickr[2]의 이미지에 기록된 태그들로부터 온톨로지를 추출하는 데 있어서 다음과 같이 [9]의 변형된 모델링을 이용하였다.

$$P(x|y >= t) \text{ and } P(y|x < t),\ Dx >= Dmin,\ Dy >= Dmin,\ Ux >= Umin,\ Uy >= Umin$$

이 수정된 모델링은 Flickr에서 빈번히 발견되는 태그들의 노이즈(noise)를 제거하기 위해 고려된 것이다. 즉, 어떤 문서의 메타데이타로서의 역할을 하지 못하는 태그, 예를 들면, 사용자 본인만 이해할 수 있는 태그, 철자의 오류가 있는 태그, 은어, 약어, 신조어 등의 태그들을 실험 대상에서 제외하였다. 아울러 산출된 상하위 관계 짝들을 그래프로 표현하는 데 있어서 결과의 질을 높이기 위하여, 노드 간 가지치기(tree pruning)와 연결선의 강화(reinforcement) 방법 등을 적용하였다. 한편, [12]에서는 문서 내에서의 키워드들이나 사용자 질의 등과 같이 문맥에 대한 정보를 많이 담고 있지 않은 짧은 텍스트로부터 택소노미를 생성해내는 데 있어 웹의 풍부한 자원을 지식체계로 이용하였고, 이에 계층적 클러스터링(hierarchical clustering) 기법을 적용하여 상하위 관계를 산출하였다.

3. 상하위 관계의 산출

본 장에서는 제 2장에서 소개한 기존 연구들이 제시한 코어커런스를 기반으로 하여, 본 논문에서 제안하는 태그들간 상하위 관계 산출 기법에 대해 논의한다.

3.1 del.icio.us 태그 데이타

del.icio.us 태그들 간의 상하위 관계를 산출하는 데 앞서, del.icio.us에 기록되는 태그들은 어떤 특징이 있는지 간단하게 살펴보겠다. 그림 2는 2007년 10월 현재 del.icio.us에서 가장 사용도가 높은 인기태그 140개를 뽑아 알파벳 순으로 나열한 태그 구름을 캡쳐한 화면이다. 이 태그들을 살펴보면, 가장 큰 특징으로는 대부분의 태그들이 명사라는 사실이다. 6개의 형용사(cool, funny, green, imported, interesting, social)를 제외한 137개의 태그들이 모두 어떤 개념을 가리키는 명사(구)이다. 또 다른 특징으로는 이 명사들 중 상당 수가 고유명사라는 점이다. ajax, apple, .net 등은 모두 어떤 특정 업체나 기술, 상품 등을 가리키는 고유명사들이다. 또한, 단수명사와 복수명사가 혼용되고 있다는 점도 눈에 띈다. article과 articles, blog와 blogs, photo와 photos 등은 실제로 동일한 태그이지만, 태깅 시스템에서는 이들을 서로 다른 독립적인 태그로 간주하기 때문

그림 2 2007년 10월 현재 del.icio.us의 태그 클라우드

이다. bit200f07, diy, fic, seo, sga 등과 같이 일견으로는 그 의미를 파악하기 힘든 신조어나 약어 등도 상당수 포함되어 있다. 이상으로 잠시 살펴본 바와 같이 del.icio.us의 태그들은 시대에 유행하는 어느 특정 개념을 가리키는 명사가 주류를 이룬다고 결론지을 수 있다.

3.2 기본 가정과 정의

본 논문에서는 앞 절에서 살펴본 del.icio.us 태그들의 특징을 반영하여, Wikipedia[13] 텍스트에 기반한 상하위 관계 산출 기법을 제안한다. Wikipedia는 웹 기반의 무료 온라인 백과사전이라 일컬어지는 서비스인데, 전세계 모든 사용자들이 정보의 생산자 혹은 가공자로 참여하여 광범위한 지식 플랫폼을 제공해주고 있다. 웹 2.0을 대표하는 특징인 대중의 지혜(the wisdom of crowds), 혹은 집단지성(the collective intelligence)이 가장 잘 반영된 곳이 바로 Wikipedia라고 할 수 있는데, 이 Wikipedia는 del.icio.us의 다양한 태그들이 지니는 의미를 파악하는 데 있어서 훌륭한 참조자료로 작용할 수 있다. 일반적으로 용어들간의 상하위 관계를 산출하기 위해 통계적인 방법을 이용하는 경우, 신뢰성 높은 결과를 산출해내기 위해 대형의 코퍼스(corpus)를 이용해왔다. 이는 대형의 코퍼스를 이용하는 경우, 옳지 않은 데이타 즉, 잡음(noise)들이 통계적인 방법으로 걸러져 양질의 결과를 산출해내는 효과를 기대하기 위해서이다. Wikipedia의 경우, Wikipedia에서 제공하는 정보는 잡음이 없으며 동시에 어떠한 용어에 대해 가장 정확하고 충분하게 기술하고 있다고 가정한다면, 대용량 코퍼스를 이용하는 것보다 더욱 양질의 결과를 산출해낼 수 있을 것이다. 이에 본 논문에서는 del.icio.us의 각 태그들에 매핑되는 Wikipedia 텍스트를 찾아, 그 텍스트를 근거로 두 태그들 간의 상하위 관계를 산출하였다(그림 3).

이를 위해 다음과 같은 가정들을 정의하였다.

1. del.icio.us의 모든 태그들은 명사로 간주한다. 앞서 밝힌 바와 같이 del.icio.us 태그들의 대부분이 명사일 뿐만 아니라, 우리가 상하위 관계를 밝히려고 하는 대상 또한 어떤 개념을 가리키는 명사이기 때문이다.
2. del.icio.us의 각 태그들은 최소한 하나의 Wikipedia 텍스트에 매핑된다.
3. 각 태그에 매핑된 Wikipedia 텍스트는 그 태그의 개념을 가장 충실하게 기술하고 있다.

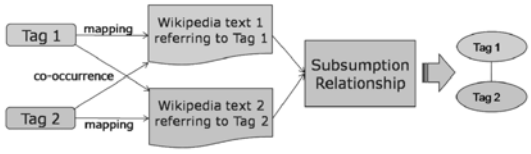

그림 3 Wikipedia를 이용한 상하위 관계 산출

아울러, 두 용어들간의 상하위 관계(subsumption) 또한 정의할 필요가 있다. 이 정의에 있어서는 앞선 연구들에서 공통적으로 제시한 개념과 유사한 개념을 채택하였다. 즉, 텀(term) x가 텀 y보다 더 일반적인 개념을 가리킬 때, x는 y의 상위 텀, y는 x의 하위 텀이라고 말할 수 있다.

3.3 상하위 관계 산출 모델링

본 논문에서는 두 태그간 상하위 관계를 밝히기 위해 [9]에서 제안한 코어커런스를 기반으로 한 모델을 채택하였다. 그러나 del.icio.us의 태그들과 Wikipedia의 특성을 고려하여 다음과 같은 수정된 모델을 제안한다.

정리 1. 두 개의 태그 x, y에 대해 다음의 두 가지 조건이 만족되면, x는 y의 상위 태그라 할 수 있다.

$$TF(y|Wiki(x)) < TF(x|Wiki(y)),\ \mu < TF(x|Wiki(y))$$

여기서, $Wiki(a)$는 태그 a가 매핑된 Wikipedia 텍스트를 가리키고, $TF(b|Wiki(a))$는 태그 b가 $Wiki(a)$에 출현하는 빈도수를 $Wiki(a)$의 전체 토큰의 개수로 나눈 값이고, μ는 실험적으로 정해지는 임계값이다. 위의 정리를 다시 풀어 쓰면, 태그 x가 태그 y의 Wikipedia 텍스트에 출현하는 빈도수가 태그 y가 태그 x의

Wikipedia 텍스트에 출현하는 빈도수보다 크고, 태그 *x*가 태그 *y*의 Wikipedia 문서에 일정 수준 이상으로 많이 출현하면, 태그 *x*가 태그 *y*의 상위개념이라 말할 수 있다는 것이다. 만약 위의 두 조건 중 하나라도 만족되지 않는 경우에는 두 태그는 상하위 관계로 결정되지 못한다. 실제로 *TF(y|Wiki(x))*나 *TF(x|Wiki(y))*의 값이 모두 0으로 나오는 경우가 적지 않게 발생하는데, 이 경우에 두 태그는 위의 첫 번째 조건을 만족하지 못해 상하위 관계로 결정되지 못한다. 실험상에서 실제로 24,500개의 태그 짝들에 대해서 TF값을 계산해본 결과 16,793개의 태그 짝들의 TF값이 0을 기록했다. 즉, 약 68.5% 정도는 TF값이 0으로서 상하위 관계로 산출되지 못하는 것이다. 그리고, 이 중 561개(약 2.29%) 의 태그 짝들이 상하위 관계로 산출되었다. 전체적인 TF의 평균값은 약 0.00122이다. 참고로, 산출된 상하위 개념의 개수와 그들의 질적인 우수성을 고려했을 때 μ = 0.01일 때 최상의 결과를 얻을 수 있었다.

예제 1. 두 개의 태그 *'apple'*과 *'mac'*에 대해서,
TF(apple | Wiki(mac)) = 176 / 7479 = 0.0235326
TF(mac | Wiki(apple)) = 161 / 9814 = 0.0164051
μ = 0.01
이라면, *TF(mac | Wiki(apple)) < TF(apple | Wiki (mac))*이고 *μ< TF(apple | Wiki(mac))*이므로 *'apple'*이 *'mac'*의 상위개념이라고 할 수 있다.

3.4 태그의 의미 결정

위의 정리 1을 적용하기 위해서는 선행되어야 할 작업이 있는데, 바로 어떤 태그 *x*에 매핑될 Wikipedia 텍스트를 찾는 것, 곧 *Wiki(x)*를 구하는 것이다. 하나의 단어는 여러 가지 의미를 지닐 수 있기 때문에, 어떤 태그가 문서 안에서 사용된 의미를 정확히 기술해주는 Wikipedia 텍스트를 찾아서 매핑해주는 작업은 매우 중요하다. 가령, 'apple'이라는 태그는 과일의 한 종류인 사과를 의미할 수도 있으며, IT 기업인 'Apple(사)'를 의미할 수도 있는 것이다. Wikipedia에서는 이 두 가지 의미를 각각 독립적인 토픽(topic)으로 간주하고 각각을 기술하는 텍스트를 따로 제공하고 있다. 이렇게 단어가 가진 여러 가지 의미들 중 문맥 속에서 사용된 의미를 찾는 작업은 단어 의미의 모호성 제거(word sense disambiguation, 이하 WSD)라는 기술에 속한다. 본 논문에서는 이를 태그에 적용하여 태그 의미의 모호성 제거(tag sense disambiguation, 이하 TSD)라 칭하겠다.

[13]에 따르면, 전통적으로 WSD 기술은 깊은 접근방법(deep approaches)와 얕은 접근방법(swallow approaches)으로 나누어 볼 수 있다. 가령, 어떤 단어 'bass'는 물고기의 일종인 농어를 의미할 수도 있고, 음악에서 쓰이는 저음을 의미할 수도 있다고 할 때, 문맥 속에서 'bass'가 둘 중 어떤 의미를 가지는지를 밝히려 한다. 이때 깊은 접근방법은 기존의 지식체계(knowledge base)를 이용하는 것이다. 즉, "물고기를 잡기 위해 낚시를 갈 수는 있지만, 저음을 잡기 위해 낚시를 갈 수는 없다" 등과 같은 우리가 잘 알고 있는 기존의 상식, 혹은 지식 등을 적용하여 의미를 찾는다. 그러나, 이 깊은 접근방법은 실제로 큰 실효를 거두지는 못했다. 얕은 접근방법에서는 그 단어 주변에 위치한 단어들을 고려하는 것이다. 즉, 'bass'라는 단어 주변에 'sea'나 'fishing'등이 단어가 위치해 있으면 그 의미를 농어라고 판단할 수 있다는 것이다. 이 얕은 접근방법은 실제 적용에 있어 깊은 접근방법보다 우수한 결과를 보인다. 그 밖에 WSD 연구분야에 대한 자세한 설명은 [14]에 잘 정리되어 있다.

본 논문에서는 WSD의 얕은 접근방법을 적용하여 TSD를 구현하였다. 기본 개념은 다음과 같다. 어떤 태그의 의미는 그 태그와 이웃하는 태그들의 도움을 받아 결정될 수 있다는 것이다(그림 4.). 이는 사용자들이 태그를 기록할 때 서로 연관성이 깊은 키워드들을 태그로 나열하기 때문에 설득력이 있다. 물론, 서로 연관성이 없는 키워드도 태그로 기록될 수 있지만, 연관성이 있는 단 몇 개의 키워드들만으로도 충분히 TSD에 기여할 수 있다.

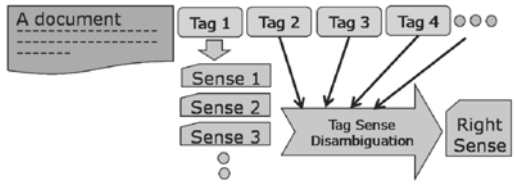

그림 4 이웃 태그들을 활용한 TSD

이와 같은 기본 개념을 근거로, 대상이 되는 태그 tt에 대한 Wikipedia 텍스트 *Wiki(tt)*를 구하는 방법은 다음과 같다.

정리 2. *tt*는 대상이 되는 태그, 집합 *NeighborTags = { ti | ti is the i-th neighbor tag of tt }* (여기서 *$1 \le i \le N$* 이고, *N*은 *tt*의 이웃태그의 개수), 집합 *WikiTopics = { topicj | topicjis the Wikipedia topic which has the j-th sense of tt }* (여기서 *$1 \le j \le M$* 이고, *M*은 *tt*가 지니는 서로 다른 의미들의 개수), 함수 *Wiki(x)*는 태그 *x*에 매핑된 Wikipedia 텍스트, $SumOfTF_j = \sum_{i=1}^{N} TF(t_i \mid Wiki(topic_j))$ 일 때,

Wiki(tt) = Wiki(topicj), where MAX[SumOfTFj]

즉, 대상이 되는 태그 *tt*가 *M*개의 의미를 지니고 있고 *N*개의 이웃태그가 있을 때, 이 이웃태그들이 각각의 의미에 매핑되어 있는 Wikipedia 텍스트에서 가장 많이 출현하는 Wikipedia 텍스트를 선정하여, *Wiki(tt)*로 선정한다는 것이다.

예제 2. 어떤 태그 '*apple*'의 의미를 밝히고자 할 때,

tt = apple

NeighborTags = {mac, leopard, terminal}

WikiTopics = {Apple_(fruit), Apple_Inc., Apple_Bank, Apple_Records}

이라면,

$SumOfTF_1$ = *TF(mac|Wiki(Apple_(fruit))) + TF(leopard|Wiki(Apple_(fruit))) + TF(termina|Wiki(Apple_(fruit))) = 0*

$SumOfTF_2$ = *TF(mac|Wiki(Apple_Inc.)) + TF(leopard|Wiki(Apple_Inc.)) + TF(terminal|Wiki(Apple_Inc.)) = 0.01641*

…

이므로, *Wiki(apple) = Wiki(Apple_Inc.)*

3.5 알고리즘

이상의 과정을 알고리즘으로 정리하면 그림 5와 같다. *FolksoViz* 알고리즘은 입력값으로 del.icio.us 태그 데이타의 집합 *T*와 임계값 μ를 받아들여 출력값으로 상하위 관계 짝의 집합인 S를 산출한다. 2, 3, 4 행에서는 입력값으로 받아들인 모든 태그들에 대해 정리 2에서 정의한 TSD를 적용하여 각각 매핑되는 Wikipedia 텍스트를 찾는다. 5, 6, 7행에서는 각 태그들이 이룰 수 있는 모든 1:1 짝들의 조합에 대해 앞서 언급한 코어커런스 기반 빈도수를 계산한다. 8, 9, 10행에서는 정리 1에서 정의한 모델링을 근거로 상하위 관계를 이루는 짝을 찾아내어 집합 S에 추가시킨다. 그림 6은 TSD를 수행하는 알고리즘이며, 자세한 설명은 생략한다(3.4절 참조).

Algorithm: *FolksoViz(T, μ)*

Input: a set *T* of del.icio.us tag data, a threshold value μ
Output: a set *S* of subsumption pairs

1. Set T as an input in the form of
   T = {ti | ti is the i-th distinct tag of the del.icio.us tag data},
   where 1 ≤ i ≤ N, N is the # of distinct tags
2. Construct W in the form of
   W = {(ti, Wti) | ti is the i-th tag in T, Wti is the Wikipedia text which refers to ti}
3. Set W ← ϕ
4. For each instance ti in T, do loop
   add (ti, Wti) to W, where Wti = TSD(ti, T)
5. Construct C in the form of
   C = {(t1, t2, TF(t1|Wt2)), TF(t2|Wt1))) | t1, t2 are tags, TF(t1|Wt2)) is the term frequency of t1 on the Wikipedia text of t2, TF(t2|Wt1)) is vice versa}
6. Set C ← ϕ, j ← 0, k ← 0
7. for (j = 0; j++; j < N)
   for (k = 0; k++; k < N)
   add (tj, tk, TF(tj|Wtk)), TF(tk|Wtj))) to C
8. Construct S in the form of
   S = {(t1, t2) | t1 is the child term, t2 is the parent term of t1}
9. Set S ← ϕ
10. For each instance in C, do loop
   if ((TF(tj|Wtk)) < TF(tk|Wtj)) && (μ < TF(tk|Wtj)))
   add (tj, tk) to S
11. Return S

그림 5 FolksoViz 알고리즘

Algorithm: *TSD(tt, T)*

Input: a target tag tt, a set *T* of del.icio.us tag data

Output: a Wikipedia text

1. Set T as an input in the form of
   T = {ti | ti is the i-th tag of tt}-[5],
   where $1 \le i \le N$, N is the # of distinct tags
2. Set Topics in the form of
   Topics = {(topicj | topicj is the j-th Wikipedia topic which refers to tt}
3. Set S ← ϕ, imax ← 0, max ← 0
4. For each instance topicj in Topics, do loop

$$SumOfTF_j = \sum_{i=1}^{N} TF(t_i \mid Wiki(topic_j))$$

   if (SumOfTFj > max)
   max ← SumOfTFj
   imax ← j
5. Return Wiki(topicimax)

그림 6 TSD 알고리즘

4. 폭소노미 시각화

본 장에서는 제 3장에서 산출해낸 태그들의 상하위 관계 짝들을 효과적으로 시각화하여 화면에 표현하는 방법론과 그 결과에 대한 분석에 대해 논의한다.

4.1 폭소노미 시각화 원칙

[15]에서는 여러 가지 형태로 요청되는 질의의 결과를 다양하게 표현하는 인터페이스(interface)와 시각화(visualization) 방법에 대해 소개하고 있다. 이를 근거로 FolksoViz 알고리즘을 통해 산출된 상하위 관계 짝들을 시각화하기 위해 필요한 원칙들은 다음과 같다.

1. 모든 태그들의 상하위 관계를 한 화면 내에서 표현하되, 사용자들이 관심을 갖는 태그들을 대상으로 한다. 서론에서 밝힌 바와 같이 폭소노미의 시각화는 사용자들에게 직관적이고 유용한 메타데이타를 제공해야 한다는 목표를 지니고 있다. 따라서 본 논문에서는 지나치게 많은 태그와 그들간의 관계가 효과적인 시각화를 방해하는 것을 방지하기 위해, 각 태그마다 그 태그를 기록한 사용자들의 수를 계산하여 상위 50개 태그에 대해서만 그들 간의 상하위 관계를 구하여 화면에 표현하였다.
2. 전체적인 구조는 트리(tree) 구조 보다는 DAG (directed acyclic graph) 구조가 적합할 것이다. 왜냐하면, 하나의 태그가 그보다 상위 개념으로서 2개 이상의 부모를 가질 수 있기 때문이다. 단, 상하위 관계를 명확하게 보여주기 위해 부모 태그가 자식 태그보다 항상 위쪽에 위치하도록 배치하였다.
3. 태그 구름에서와 같이, 더 많은 사용자들이 기록한 태그는 노드와 글자의 크기를 더 크게 할당한다. 그리고 태그 옆에 얼마나 많은 수의 사용자들이 이 태그를 기록하였는지도 표현하였다.
4. 태그들 간의 상하위 관계에 있어 전이성(transitivity)은 고려하지 않는다. 왜냐하면 개념들간의 상하위 관계에 있어서 항상 전이성이 성립하는 것은 아니기 때문이다. 예를 들면, apple은 mac의 상위개념이고, corporation은 apple의 상위개념이지만, corporation이 mac의 상위개념은 아니다. 이와 같은 비전이성은 대개 상위개념으로 확장될 때 동일한 기준을 적용하지 않는 경우에 발생한다. 따라서, 태그들간의 상하위 관계들이 전이적이든 아니든 간에 관계없이 노드(node)들 사이의 모든 간선(edge)은 그대로 유지하였다.
5. 각각의 노드는 그 노드가 의미하는 태그의 하이퍼링크를 지니고 있어야 한다. 즉, 사용자는 노드를 클릭함으로써 태그 검색을 할 수 있어야 한다.

이와 같은 다섯 가지 원칙에 따라 태그들간 상하위 개념을 시각화하였으며, Java 기반의 JGraph[16]를 이용하여 구현하였다.

4.2 결과 분석

그림 7은 제 1장의 그림 1에서 언급한 del.icio.us 태그들간의 상하위 관계를 구하여 이를 시각화한 결과이다. 그림이 표현하고 있는 상하위 관계들을 살펴보면, 전반적으로 태그들간의 상하위 관계를 잘 찾아서 표현

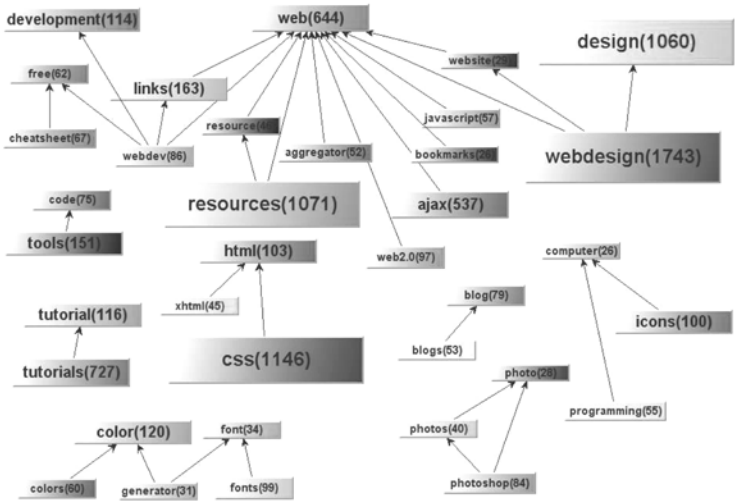

그림 7 태그들간 상하위 관계의 시각화 결과의 예

하고 있다고 볼 수 있다. 가령, web2.0의 상위개념으로서 web이, webdesign의 상위개념으로서 web과 design이, css의 상위개념으로서 html이 산출된 것은 FolksoViz 알고리즘이 좋은 성능을 보이고 있음을 의미한다. 한편, tools의 상위개념으로서 code가, webdev의 상위개념으로서 free가 산출된 것은 다소 어색해 보인다. 이는 서로 상하위 관계를 이루고 있지는 않지만 의미상으로는 밀접한 연관이 있을 때 코어커런스가 높아진다는 사실에 기인한다. 또한, resources의 상위개념으로서 resource가, colors의 상위개념으로 color가, blogs의 상위개념으로 blog가 산출된 것도 엄밀히 옳은 결과는 아니다. 이는 어떤 단어의 단수형이든 복수형이든 태깅 시스템에서는 이들을 서로 독립적인, 별개의 개체로 간주하기 때문에 발생한다.

5. 성능평가

본 장에서는 제 3장에서 제안한 태그들간 상하위 관계 산출을 위한 FolksoViz 알고리즘의 성능을 평가하고, 이에 대한 분석에 대해 논의한다.

5.1 상하위 관계 산출의 성능분석방법

본 성능평가의 목표는 산출된 태그들간상하위 관계가 실제로 얼마나 정확한가를 측정하는 것이다. 그러나, 정량적인 분석이 아닌 정성적인 분석이 이루어져야 하기 때문에 객관적이고 절대적인 평가를 실행하기는 쉽지 않다. 무엇보다도, 태그들간 상하위 관계의 정확성을 판단하기 위한 도메인 전문가(domain expert)나 정답이 되는 기준 데이타셋(dataset)이 존재하지 않기 때문이다. 뿐만 아니라, 어떤 경우에 있어서는 사람에게 있어서도 개념들간의 상하위 관계를 결정하기가 쉽지 않다. 가령, finance와 business의 경우에 있어서, finance를 business에서 관리하는 여러 하위개념 중의 하나로 간주할 수도 있겠으나, 거꾸로 business를 finance의 한 분야에 속하는 하위개념으로 간주할 수도 있다. 이와 같이, 개념들간 상하위 관계가 얼마나 정확한지를 객관적으로 측정하기에는 무리가 있다.

이에 본 논문에서는 컴퓨터 과학을 전공으로 하는 15명의 박사과정 대학원생들을 대상으로 산출된 상하위 관계의 정확성을 평가하도록 하였다. 이 피실험자들은 del.icio.us 태그에 대한 도메인 전문가로 간주되었고, 평가 이전에 이들에게 생소할 수 있는 몇몇 태그들에 대해서는 그 의미를 미리 교육시켜주어, 태그의 의미를 몰라 상하위 관계를 올바르게 평가하지 못하는 경우는 발생하지 않도록 하였다.

평가의 대상이 되는 예제들은 도메인의 다양성을 반영하기 위해 2007년 9월 현재 del.icio.us의 상위 10개의 인기 태그들(mac, webdesign, music, web2.0, software, video, games, shopping, education, business)을 각각의 주제로 삼아, 이 주제들과 관련된 10개의 URL을 무작위로 선정하였다. 표 1은 평가의 대상으로 선정된 10개의 예제 URL들에 대한 정보를 요약한 표이다. 대부분의 URL들이 1000명이 넘는 사용자들과 수천 개에서 수만 개에 이르는 태그들을 보유함으로써, 진정한 의미의 폭소노미의 요건을 충분히 갖추었다고 본다.

실험방식은 다음과 같다. 피실험자는 하나의 URL에 대해서 무작위로 선출된 30개의 상하위 관계의 짝에 대해서 정확성을 평가한다(곧, 총 300개의 짝에 대해 평가). 각각의 짝에 대해서 피실험자는 그 상하위 관계가 옳다고 판단되면 'Correct'를, 상하위 관계가 역으로 되

표 1 평가의 대상으로 선정된 10개의 예제 URL들에 대한 정보

#	URL	Title	Topic	# of taggers	Tot. # of tags
1	http://www.usingmac.com/2007/11/18/leopard-tweaking-terminal-codes	Usingmac.com - Leopard Tweaking - Terminal Codes	mac	758	2973
2	http://www.forwebdesigners.com/	For Webdesigners - 297 helpful links for webdesigners	webdesign	2791	11672
3	http://songza.com/	Songza - The music search engine & internet jukebox. Listen. Now.	music	3779	17709
4	http://www.omnidrive.com/	Omnidrive: The Universal Storage Platform - Home	web2.0	4616	7627
5	http://sourceforge.net/	SourceForge.net: Welcome to SourceForge.net	software	6409	20112
6	http://www.getmiro.com/	Miro - free, open source internet tv and video player	video	4695	20653
7	http://www.gamerankings.com/	Game Rankings - Video Game Reviews, Release Dates, Cheat Codes	games	1379	4208
8	http://www.allposters.com/	AllPosters.com - The World's Largest Poster and Print Store!	shopping	1374	3524
9	http://www.eliteskills.com/free_education/?foo=x	Online Education	education	5514	20477
10	http://www.wesabe.com/	Wesabe: Get to Know Your Money	business	2461	8976

어야 옳다고 판단되면 'Inverted'를, 상하위 관계가 아닌 유의어 관계라고 판단되면 'Synonymous'를, 상하위 관계는 아니지만 연관성은 있다고 판단되면 'Not correct, but related'를, 상하위 관계도 아니고 연관성도 없다고 판단되면 'Neither correct nor related'를, 상하위 관계를 결정하기가 어려운 관계라고 판단되면 'I don't know'를 선택한다.

5.2 실험결과 및 분석

실험결과는 표 2와 같다. 10개의 URL 각각에 대한 평가자들 응답의 통계가 %로 계산이 되었고, 최하단에는 전체적인 평균이 계산되었다. 결과를 분석해보면, 상대적으로 높은 'Correct'응답의 비율(58.4%)과 상대적으로 낮은 'Inverted'와 'Neither correct nor relate'의 비율(각각 1.3%, 7.8%)이 무엇보다도 고무적이다. 이 수치들은 FolksoViz 알고리즘이 상당히 좋은 성능을 보이고 있음을 의미한다. 아울러, 적지 않은 비중을 차지하고 있는 'Synonymous'와 'Not correct, but related'의 비율(각각 4.1%와 14.8%)에 주목할 필요가 있다. 이는 앞서 언급한 바와 같이, 어떤 두 태그가 서로 상하위 관계를 이루고 있지는 않더라도 의미적으로 밀접한 연관이 있으면, 동시에 출현하는 가능성이 증가하기 때문이다. 이는 텀들간의 상하위 관계 산출 시 코어커런스에만 의존하는 본 논문의 모델링이 지니는 한계라고도 볼 수 있다. 한편, 적지 않은 평가자들(13.8%)이 'I don't know'를 선택하였는데, 이들이 응답한 질문을 분석해본 결과, 단수형과 복수형간의 관계(가령, article-articles), 또는 명사와 동사 ing 형태간의 관계(tweak-tweaking), 본어-약어간의 관계(newyork-ny) 등과 같은 수긍할 수 없

표 2 실험결과

#	Topic	Correct	Inverted	Synonymous	Not correct, but related	Neither correct, nor related	Don't know
1	mac	66.6667	6.6667	0.8333	11.6667	5.8333	8.3333
2	webdesign	54.1667	0.8333	0	13.3333	10.8333	20.8333
3	music	63.3333	0	4.1667	16.6667	3.3333	12.5000
4	web2.0	65.8333	0	1.6667	23.3333	0.8333	8.3333
5	software	55.0000	0.8333	0	17.5000	19.1667	7.5000
6	video	52.5000	1.6667	6.6667	19.1667	7.5000	12.5000
7	games	61.6667	0	0.8333	11.6667	12.5000	13.3333
8	shopping	39.1667	0.8333	18.3333	15.8333	5.8333	20.0000
9	education	65.8333	0	5.0000	7.5000	5.0000	16.6667
10	business	60.0000	1.6667	3.3333	10.8333	6.6667	17.5000
Avg.		**58.4**	**1.3**	**4.1**	**14.8**	**7.8**	**13.8**

표 3 기존연구들과의 비교

Model	Correct	Inverted	Synonymous	Not correct, but related	Neither correct, nor related	Don't know
Sanderson and Croft	23%	NA	8%	49%	19%	NA
Clough et al.	15%	NA	0.2%	10%	43%	NA
Patrick Schmitz	51%	NA	5%	21%	23%	NA
FolksoViz	**58.4%**	**1.3%**	**4.1%**	**14.8%**	**7.8%**	**13.8%**

는 상하위 관계에 대한 판단이 대다수를 이루었다. 이는 어떤 두 태그의 동등(equivalence)관계를 파악해내지 못하는 FolksoViz 알고리즘의 한계라고 할 수 있다.

표 3은 기존의 연구방법들과 FolksoViz 알고리즘의 성능을 비교한 것이다. 일단, 이 표는 각 연구방법들이 각기 고유한 도메인에서 각각의 데이타셋을 대상으로 성능을 평가한 후 최종평균 수치만을 정리한 것이기 때문에, 수치상의 단순 비교로 성능의 우월을 가늠하기는 어렵다. 그러나, 기존 연구방법들과 비교하여 전반적으로 높은 'Correct'의 수치와 상대적으로 낮은 'Inverted'와 'Neither correct, nor related'의 수치는 FolksoViz 알고리즘이 좋은 성능을 지니고 있음을 보여준다.

6. 결론 및 향후연구

본 논문에서는 태그들간의 상하위 관계를Wikipedia 텍스트를 이용하여 산출해내는 알고리즘을 제안하였다. 이 FolksoViz 알고리즘은 del.icio.us 태그들을 대상으로 효과적으로 상하위 관계를 산출해냈으며, 이렇게 산출된 상하위 관계들을 시각화하여 화면에 표현함으로써 폭소노미의 시각화를 이루었다. 이러한 과정에 있어서 본 논문에서는 현재 각광받고 있는 웹 2.0의 여러 특징들을 적극적으로 도입하여 적용하였다는 데 큰 의의를 둔다. 본 논문에서는 진정한 의미의 폭소노미라고 할 수 있는 del.icio.us의 협력태깅을 그 대상으로 하였고, 알고리즘 수행을 위한 기반으로서 집단지성의 산물이라고 할 수 있는 Wikipedia 텍스트를 활용하였다.

한편, 앞서 지적한 바와 같이 본 논문에서 제안하는 알고리즘은 몇 가지 측면에 있어서 그 한계를 노출했다. 무엇보다도, 텀들간 코어커런스에 기반한 통계적인 모델링은 의미적으로 유사한 태그들을 상하위 관계로 규정짓는 단점을 지니고 있다. 따라서, 코어커런스에만 의존하지 않는 새로운 방식의 모델링이 필요할 것이다. 그리고, 의미적으로 동등한 관계를 지니는 태그들을 밝혀내는 작업 또한 선행되어야 한다. 앞서 밝힌 바와 같은 단수-복수, 명사-동사ing, 본어-약어 등과 같은 관계는 상하위 관계가 아닌 동등 관계로 취급할 수 있는 알고리즘이 필요하다. 결국, 태그들간의 상하위 관계뿐만 아니라, 의미적으로 가능한 여러 가지 관계를 밝혀낼 수 있다면, 폭소노미의 시각화에 크게 기여할 수 있을 것이다.

참 고 문 헌

[1] del.icio.us, http://del.icio.us.

[2] Flickr, http://www.flickr.com.

[3] K. Bielenberg and M. Zacher, "Groups in Social Software: Utilizing Tagging to Integrate Individual Contexts for Social Navigation," Masters Thesis submitted to the Program of Digital Media, Unisersitat Bremen, 2006.

[4] Rashmi Sinha, "A cognitive analysis of tagging," http://www.rashmisinha.com/archives/05_09/tagging-cognitive.html, 2005.

[5] Scott Golder and Bernardo A. Huberman, "Usage Patterns of Collaborative Tagging Systems," Journal of Information Science, Vol. 32, No. 2, pp. 198-208, 2006.

[6] Miller G. A., "WordNet: A Lexical Database for English," Communications of the ACM, Vol. 38, No. 11, pp. 39-41, 1995.

[7] Sharon A. Caraballo and Eugene Charniak, "Determining the Specificity of Nouns from Text," in Proceedings of the joint SIGDAT conference on empirical methods in natural language processing (EMNLP) and very large corpora (VLC), pp. 63-70, 1999.

[8] Forsyth R. and Rada R., "Adding an edge in Machine Learning: applications in expert systems and information retrieval," Ellis Horwood Ltd., pp. 198-212, 1986.

[9] Mark Sanderson and Bruce Croft, "Deriving Concept Hierarchies from Text," in Proceedings of the 22nd ACM Conference of the Special Interest Group in Information Retrieval, pp. 206-213, 1999.

[10] Paul Clough, Hideo Joho, and Mark Sanderson, "Automatically Organising Images using Concept Hierarchies," in Proceedings of the Workshop on Multimedia Information Retrieval, SIGIR, pp. 2005.

[11] Patrick Schmitz, "Inducing ontology from Flickr tags," in Proceedings of the Collaborative Web Tagging Workshop, WWW, pp. 2006.

[12] Shui-Lung Chuang and Lee-Feng Chien, "Taxonomy generation for text segments: A practical web-based approach," ACM Transactions on Information Systems (TOIS), Vol. 23, No. 4, pp. 363-396, 2005.
[13] Wikipedia, http://wikipedia.org/.
[14] Nancy Ide and Jean Véronis, "Word Sense Disambiguation: The State of the Art," http://www.up.univ-mrs.fr/~veronis/pdf/1998wsd.pdf, 1998.
[15] Marti A. Hearst, "User Interfaces and Visualization," Modern Information Retrieval, New York: ACM Press, pp. 257-323, 1999.
[16] JGraph, http://www.jgraph.com.

이 강 표
정보과학회논문지 : 데이타베이스
제 35 권 제 2 호 참조

김 현 우
2007년 KAIST 전산학과(학사). 2007년~현재 서울대학교 컴퓨터공학부 석박통합과정 재학중. 관심분야는 데이타베이스, 웹 2.0, 시맨틱웹

장 충 수
2007년 건국대학교 컴퓨터공학부(학사) 2007년~현재 서울대학교 전기컴퓨터공학부(석사). 관심분야는 데이타베이스, 웹 2.0, 정보검색

김 형 주
정보과학회논문지 : 데이타베이스
제 35 권 제 2 호 참조