범주별 태그 안정성을 이용한 태그 부착 자원의 SVM 기반 분류 기법

(A SVM-based Method for Classifying Tagged Web Resources using Tag Stability of Folksonomy in Categories)

고 병 걸 † 이 강 표 † 김 형 주 ††
(Byunggul Koh) (Kangpyo Lee) (Hyoung-Joo Kim)

요 약 폭소노미(Folksonomy)는 자유롭게 선택된 키워드의 집합인 태그를 이용하여 이루어지는 협업적 분류로서 웹 2.0의 대표 요소이다. 폭소노미는 기존 분류 방법인 택소노미(Taxonomy)에 비해 적은 비용으로 구축할 수 있다는 장점이 있으나 택소노미에 비해 계층적, 체계적 구조가 부족하다는 단점을 가지고 있다. 이에 폭소노미에 존재하는 집단 지성을 학습하여 웹 자원을 분류할 수 있는 분류기를 구축할 수 있다면 기존 방법인 택소노미를 적은 비용으로 구축할 수 있을 것이다. 본 논문에서는 Slashdot.org에 구축되어 있는 폭소노미를 대상으로 일반적 모델을 정의하고 이 안에서 안정성이 존재함을 보임으로써 분류기를 생성할 수 있는 집단 지성이 폭소노미에 실제로 존재함을 보인다. 그리고 이 집단 지성으로부터 형성되는 범주 별 태그의 특징인 안정성 값을 이용하여 SVM으로 분류기를 구축하는 방법을 제안한다. 실제로 우리가 제안하는 방법으로 폭소노미로부터 높은 정확도로 택소노미를 구축하였음을 실험을 통해 확인하였다.

키워드 : 폭소노미, 태그, 집단 지성, 택소노미, 분류, SVM

Abstract Folksonomy, which is collaborative classification created by freely selected keywords, is one of the driving factors of the web 2.0. Folksonomy has advantage of being built at low cost while its weakness is lack of hierarchical or systematic structure in comparison with taxonomy. If we can build classifier that is able to classify web resources from collective intelligence in taxonomy, we can build taxonomy at low cost. In this paper, targeting folksonomy in Slashdot.org, we define a general model and show that collective intelligence, which can build classifier, really exists in folksonomy using a stability value. We suggest method that builds SVM classifier using stability that is result from this collective intelligence. The experiment shows that our proposed method managed to build taxonomy from folksonomy with high accuracy.

Key words : Folksonomy, Tag, Collective intelligence, Taxonomy, Classification, SVM

· This research was supported by the Brain Korea 21 Project, the Ministry of Knowledge Economy, Korea, under the Information Technology Research Center sup-port program supervised by the Institute of Information Technology Advancement (grant number IITA-2008-C1090-0801-0031), and a grant (07High Tech A01) from High tech Urban Development Program funded by Ministry of Land, Transportation and Maritime Affairs of Korean government
· 본 연구는 BK-21 정보기술 사업단, 지식 경제부 및 정보통신연구진흥원의 대학 IT연구센터 육성지원사업(IITA-2008-C1090-0801-0031), 국토해양부 첨단도시개발사업의 연구비지원(07첨단도시 A01)의 연구 결과로 수행되었음

† 비 회 원 : 서울대학교 컴퓨터공학부
byunggulkoh@gmail.com
(Corresponding author임)
kplee@idb.snu.ac.kr
†† 종신회원 : 서울대학교 컴퓨터공학부 교수
hjk@snu.ac.kr

논문접수 : 2008년 10월 21일
심사완료 : 2009년 5월 11일

정보과학회논문지: 컴퓨팅의 실제 및 레터 제15권 제6호(2009.6)

1. 서 론

태그(Tag)는 기사, 사진, 동영상 등의 자원(Resource)을 기술하기 위하여 사용자가 자원에 붙인 키워드들의 집합을 지칭한다. 이 태그들은 자원을 기술하는 기술적 메타 데이터(Descriptive metadata)로 볼 수 있다. 전통적으로 메타데이터는 소수의 전문가들에 의하여 생성 및 관리되어 왔다. 이러한 전문가 집단에 의하여 만들어진 메타 데이터는 양질의 데이터로 간주할 수 있으나 이를 생성 및 관리하기 위한 비용이 크다[1]. 이로 인하여 기존의 메타 데이터 생성 방법은 인터넷 데이터와 같은 방대한 데이터에 적용하기에는 어려움이 있다. 이에 현재의 웹2.0 사이트들은 다수의 사용자 참여를 통하여 생성된 태그 데이터를 이용하려는 시도를 하고 있다. 태깅 시스템에서 사용자들은 생각나는 키워드들을 형식에 제한 없이 자원에 붙일 수 있다. 이렇게 사용이 쉽다는 장점 하에 태그 기술은 많은 사용자의 참여를 유도하였으며 웹2.0의 대표 기술로 자리매김하였다.

다수의 사용자에 의해 생성된 태그는 미리 정의된 범주(Category)가 없는 새로운 분류체계인 폭소노미(folksonomy)를 만들게 된다. 폭소노미란 folk(대중)와 taxonomy(택소노미)의 결합으로 이루어진 신조어로서, 대중이 만들어낸 택소노미라는 의미이다. 이는 Thomas Vander Wall의 Information architecture mailing list [2]에서 처음 사용되게 되었다. 폭소노미는 기존의 택소노미와 달리 비통제어휘(Uncontrolled vocabulary)가 가지고 있는 본질적인 특징을 공유한다. 미리 정의된 범주가 없기 때문에 사용자들은 원하는 방식으로 유연하게 자원의 분류가 가능하며 느끼는 어려움도 택소노미에 비하여 적다는 특징[3]이 있다. 또한 다수의 사용자들에 의하여 협업적으로 만들어지기 때문에 인터넷 자원 등의 대용량 자료에도 적용 가능하다는 장점이 있다. 현재 폭소노미를 컨텐츠 관리 체계로 사용하고 있는 대표적 사이트에는 del.icio.us가 있다. del.icio.us에서는 기존의 기계적 분류 방법으로는 처리하기 힘든 사진 데이터들을 폭소노미를 이용하여 분류하고 있다. 사진 등의 멀티미디어 데이터들을 분류할 수 있는 기계적 방법이 기대할만한 성능을 이끌고 있지 못한 상황에서 태그로부터 형성된 폭소노미를 효과적으로 사용하고 있는 것이다.

폭소노미는 그 안에서 태그의 검색과 브라우징을 통하여 그 기능을 발휘하게 된다. 검색 시 사용자가 제출한 질의의 키워드들과 자원에 달려있는 태그들과의 매칭을 통하여 매칭된 자원을 반환하는 방법을 사용한다. 브라우징을 위해서는 태그 클라우드라는 방법이 사용되는데 이는 일정한 공간 안에 현재까지 사용된 태그들의 목록을 나열한다. 이를 통하여 사용자는 이 사이트에 어떠한 자료들이 있는지 파악할 수 있게 된다. 이로 인한 폭소노미의 중요한 특징은 그것이 평평한 이름공간(flat namespace)으로 이루어져 있다는 것이다. 이는 그것에 계층적, 구조적 체계가 부족하다는 것을 의미한다[1]. 이로 인하여 많은 사이트들은 서로의 단점을 보완하기 위하여 폭소노미 시스템과 미리 정의된 범주로의 분류인 택소노미 시스템을 동시에 적용하고 있다(그림 1).

그러나 이와 같은 혼합(Hybrid) 접근 방법은 사용자에게 추가적인 부담을 주며 블로그 포탈(Blog portal)의 경우와 같이 태그 정보는 얻을 수 있으나 블로그 포탈에 의해 미리 정의된 범주에 대한 분류 정보는 얻을 수 없는 경우 사용이 불가능하다. 이에 구조적 체계가 부족하나 쉽게 구축이 가능한 폭소노미로부터 상대적으로 그 구현이 어려운 구조적 택소노미를 구성할 수 있다면 이는 다음과 같은 측면에서 유용한 작업이 될 것이다. 첫째, 이는 구현 비용이 높은 택소노미를 낮은 비용으로 구축 가능하게 하며, 기존에 기계적 방법으로 분류가 어려웠던 멀티미디어 데이터를 태그를 사용함으로써 분류의 성능을 개선시킬 것이다. 둘째, 이렇게 구축된 택소노미는 택소노미의 구조적 특징과 폭소노미의 동적인 특징을 가질 수 있다. 이는 사용자들은 쉽게 사용이 가

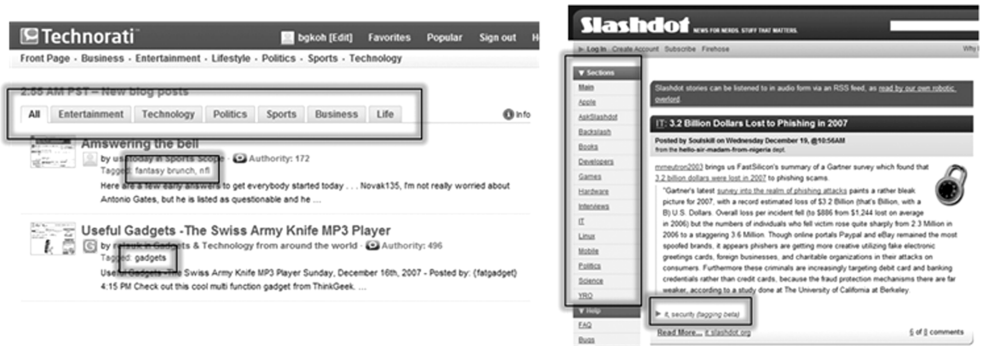


그림 1 택소노미 시스템과 폭소노미 시스템의 혼합 사용

능한 태깅 작업을 하기만하면 시스템이 이 정보로부터 택소노미를 구축함으로써 가능해진다. 결과적으로 택소노미 시스템의 장점과 폭소노미 시스템의 장점만을 취할 수 있는 효과적인 분류 시스템의 구축이 가능할 것이다.

이에 본 연구에서는 폭소노미로부터 택소노미를 구축할 수 있는 방법에 대하여 연구하였다. 본 연구의 구성은 다음과 같다. 2장에서는 태그와 폭소노미의 성질에 관한 기존의 연구결과를 소개한다. 3장에서는 Slashdot[4]와 같은 모델에서 실제로 학습 가능한 집단 지성이 존재함을 보인다. 4장에서는 폭소노미에 존재하는 집단 지성으로부터 택소노미를 만들기 위한 시스템에 대하여 소개한다. 5장에서는 연구에서 소개한 시스템 평가를 제시하고, 마지막 6장에서는 결론과 향후 연구에 관하여 논의한다.

2. 관련연구

태깅 기술은 del.icio.us, flickr, Slashdot[4]과 같은 소셜 네트워크(Social network) 서비스 및 블로그 시스템(Blog system) 등에서 널리 사용되고 있다. 그러나 태그의 성질 및 이를 이용한 활용 방안에 대한 연구는 아직까지 많이 이루어지지 않은 상황이다. 이로 인하여 실제 태깅 시스템에서 컨텐츠를 관리할 수 있는 일관적인 체계(Coherent scheme)가 존재하는지, 어떻게 이를 이용할 수 있을지에 대한 연구가 진행 중이다. 우리의 연구에서는 폭소노미 속 태그의 안정성 값을 정의하고 이를 바탕으로 태깅 시스템에 컨텐츠를 관리할 수 있는 일관적인 체계가 존재함을 보인다. 그리고 이를 이용한 분류 시스템을 제안한다. Brooks[5]의 연구는 사용자가 생성한 태그의 유용성을 분석한 초기 논문이다. 그는 같은 태그를 공유하는 블로그 기사의 클러스터를 찾고 이들의 유사성을 분석하였다. 그 결과 태그로 만들어진 클러스터는 임의적으로 만들어진 클러스터 보다 약간 높은 유사성만을 보였으며 $tf \cdot idf$ 방법으로 만들어진 클러스터가 이보다 효과적임을 보였다. 이를 근거로 사용자가 만들어 낸 태그들은 기사의 특정 내용을 제시하는 용도로는 유용하지 않다고 주장하였다. 그러나 이들의 연구는 한 개 태그의 동시 출현(Co-occurrence)만을 고려하고 다수 태그의 동시 출현은 고려하지 못했다는 한계를 갖는다. 이 연구를 기반으로 Chirita[6]는 태그를 자동적으로 생성하는 방법에 관하여 연구하였다. 그는 사용자의 PC에서 태그가 붙여질 자원과 비슷한 문서를 검색한 후 그 문서로부터 키워드들을 추출하여 태그를 생성함으로써 개인적 성향이 반영된 태그를 생성하는 방법을 제안하였다. 우리의 연구에서는 Chirita[6]의 기법을 개선하여 다른 사용자의 집단 지성을 이용하는 방법으로 기존의 태그를 확장한다. 그리고 이를 분류의 정확성 향상을 위해 사용하였다.

Golder[7]와 Halpin[8]는 협업 태깅 모델(Collaborative Tagging model)을 가지고 있는 del.icio.us에서의 태그의 안정성(Stability)에 대하여 연구하였다. 여기서 안정성의 의미는 del.icio.us에 즐겨 찾기된 어떤 URL에 대하여 다수의 사용자들이 그 URL을 최적으로 묘사할 수 있는 태그에 관해 공통적인 합의가 있는가에 대한 것이다. 이 안정성의 존재는 컨텐츠를 관리할 수 있는 일관적인 체계가 존재함을 의미한다. Golder[7]는 del.icio.us의 즐겨 찾기URL이 가진 동적인 태그들의 분포가 안정화 된다는 사실을 발견하였다. 그는 그 이유로 사용자간의 모방효과를 들었다. Golder[7]의 연구를 발전시킨 Halpin[8]의 연구에서는 안정성을 보이는 태그의 분포를 거듭 제곱 법칙(Power law)을 사용하여 증명하였다. 태그의 동시 출현 횟수에 기반하여 태그의 상하관계를 유추하는 Schmitz[9]의 연구와 태그로부터 관심사가 비슷한 소셜 네트워크를 찾는 Li[10]의 연구는 모두 이 안정성에 기반한 결과라고 볼 수 있다. 우리의 연구에서는 이 안정성 값에 착안하여 이를 변형한 새로운 안정성 값을 정의한다. 우리의 새로운 안정성 값은 자원을 묘사하는 태그의 안정성 보다는 각 범주에 존재하는 태그들의 시간에 따른 안정성에 초점을 맞춘다.

기계적 방법을 이용한 텍스트 분류(Text classification)는 많은 연구[11]가 이루어진 분야이다. 이는 문서를 특징벡터로 표현하고 이를 분류기(Classifier)로 학습하는 작업으로 이루어진다. 문서를 특징벡터로 표현하기 위해서는 텍스트로부터 용어(Term)들을 선택하는 작업이 필요하다. 이는 전통적으로 Mutual Information, Information gain, χ^2, Latent Semantic Indexing 등 다양한 방법[11,12]이 존재한다. 그러나 이들 연구는 텍스트로부터 특징벡터를 생성하는 방법에 대한 것이며 태그 데이터만을 사용하여 특징벡터를 생성하는 방법에 대한 연구는 아직 존재하는 않는다. 이에 우리는 태그 데이터로 특징벡터를 생성하는 방법을 제안하며 이 방법이 높은 정확도를 보임을 보인다.

3. 범주 별 태그의 안정성

3장에서는 Slashdot 등의 소셜 커뮤니티 사이트와 블로그 포탈 등에서 사용하는 모델을 설명한 후 실제 폭소노미로부터 택소노미를 만들 수 있는 안정성을 정의한다. 마지막으로 우리의 모델에서 실제 안정성이 존재함을 보인다.

3.1 데이터 모델

시작에 앞서 태그의 안정성을 보이고자 하는 모델에 대하여 설명한다. 우리의 모델은 Slashdot[4]의 모델을 기

반으로 하나 아래와 같은 모델을 갖춘 Technorati 등의 블로그 포털이나 소셜 네트워크 커뮤니티에도 적용 가능하다. 우리의 모델은 다음과 같은 요소로 이루어진다.

1. 사용자 집합
2. 태그 된 자원(Resource)의 집합
3. 미리 정의된 범주 집합
4. 태그의 집합

모델에서 사용자들은 공유하고 싶은 블로그 기사(Article)나 웹사이트 등의 자원에 원하는 태그를 붙인다. 이후 커뮤니티에서 미리 정의해 놓은 범주를 선택함으로써 그 행위를 마치게 된다. 다른 사용자들은 범주별로 분류된 자원을 탐색하면서 다른 사용자들이 제출한 그 범주에 속한 자원과 자원에 붙은 태그들을 열람할 수 있다. 이 열람의 과정에서 사용자들은 자신의 태그를 추가할 수 있으며 이렇게 추가된 태그는 그 사용자 공간에 저장되어 추후 검색이나 스크랩 등에 사용될 수 있다.

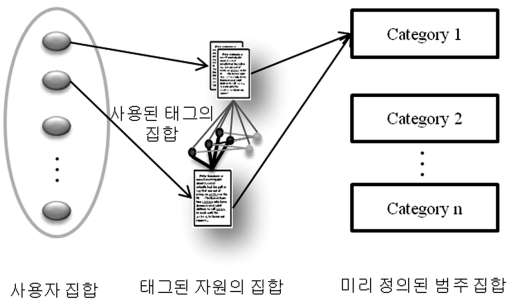


그림 2 혼합(Hybrid) 모델에서의 데이터 모델

위의 그림 2에서 볼 수 있듯이 각 자원은 하나의 범주에 속하며 이 자원으로부터 그 범주에 속하는 태그 집합을 구할 수 있다. 이 태그 집합은 위와 같은 피드백(Feedback) 과정을 거치며 범주에 존재하는 기존의 태그를 강화하는 선택이나 새로운 태그를 추가하는 선택의 과정을 반복한다. 우리의 연구에서는 이 반복의 과정에서 태그를 사용하여 하나의 범주를 선택할 수 있게끔 하는 안정성 즉, 집단 지성(Collective intelligence)이 존재함을 보이고 이를 이용한 분류 시스템을 제안한다.

3.2 범주 별 태그의 안정성 정의

이번 장에서는 3.1장에서 정의한 모델에서 미리 정해진 범주로 분류 작업을 할 시에 필요로 하는 안정성에 대하여 정의한다.

태그를 분류 작업에 사용하기 위해서는 다음을 만족해야 한다. 만일 집단 지성이 우리의 모델에 존재한다면, 사용자들이 기존의 범주에 있는 태그들의 집합을 강화하거나 추가하는 과정에서 일반적으로 따르는 합의(Consensus)가 존재할 것이다. 시스템의 안정성은 이 일반적 합의로부터 생성될 것이다. 우리는 이 합의를 따르는 유저들에 의해 생성된 태그들은 다음과 같은 성질을 가질 것으로 가정한다.

성질 1. 기존에 사용된 태그의 분포는 현재 사용되는 태그의 분포와 유사해야 한다.

성질 2. 과거 태그, 태그들의 집합이 하나의 범주를 대표하는 정도의 값은 현재의 그 값과 유사해야 한다.

성질 1은 학습을 위한 데이터와 테스트를 위한 데이터와의 관계에서 나온다. 테스트를 위한 자료가 기존에 학습했던 자료와 전혀 다르다면 기존의 학습 데이터는 전혀 쓸모 없는 데이터가 될 것이다. 예를 들어 'A', 'B', 'C'라는 태그의 결과를 학습한 시스템이 'ㄱ', 'ㄴ', 'ㄷ'이라는 태그를 테스트 데이터로 받게 된다면 어떠한 결과 예측도 할 수 없다.

성질 2는 지식의 일관성에 대한 언급이다. 이는 테스트를 위한 데이터의 결과가 기존에 학습했던 결과와 같아야 함을 의미한다. 즉, 학습 데이터에서 'A'라는 태그가 범주 1을 대표하기 위해 사용되었다가 테스트 시에서는 'A'라는 태그가 범주 2를 대표하기 위해 사용되었다면 기존의 학습은 더 이상 유효하지 않게 된다. 이는 기존의 학습결과를 버리고 새로운 학습을 시작해야 함을 의미한다.

위의 두 성질을 바탕으로 우리는 태그 안정성의 정도를 정의한다. 먼저 성질 1의 정도를 측정하기 위하여 범주 c의 태그 x의 분포 중요 값을 다음과 같이 정의한다.

정의 1. 범주 c에서 태그 x의 분포 일관성 값

$$I(x, c) = \frac{R(x, c)}{\sum R(j, c)} \quad (1)$$

여기서 R(x,c)는 태그 x가 범주 c에서 사용된 총 횟수를 나타내며, ΣR(j,c)는 범주 c에서 사용된 모든 태그의 사용 횟수의 합을 나타낸다. 식 (1)을 성질 1을 만족하는 정도의 척도로 사용한다. 성질 2의 정도는 다음과 같이 측정한다.

정의 2. 태그 x가 범주 c를 나타내는 대표도의 일관성 값

$$C(x, c) = \frac{N(x, c)}{\sum_{j \in \{All\ categories\}} N(x, j)} \quad (2)$$

여기서 N(x,c)는 범주 c안에 포함된 태그 x의 총 횟수 이며 $\sum_{j \in \{All\ categories\}} N(x, j)$는 모든 범주에서 태그 x가 발생된 총 횟수를 나타낸다. 식 (2) 역시 성질 2의 정도를 나타내는 확률 값으로써 성질 2를 만족하는 정도의 척도로 사용한다. 이 식 (1)과 (2)를 기반으로 태그 x의 안정성 정도를 나타내는 확률 값 P(x,c)를 다음과 같이 정의한다.

정의 3. 범주 c에서 태그 x의 안정성 값

$$P(x,c)=\lambda \cdot C(x,c)+(1-\lambda)\cdot I(x,c) \quad (3)$$

λ값은 식 (1)과 (2)의 가중치 값으로 시스템에 따라 적용되는 값이다. 다음 3.3장에서는 이 식 (3)의 값을 기반으로 우리의 가정을 확인한다.

3.3 Slashdot에서의 안정성 분석

이번 장에서는 3.2장에서 정의한 범주 별 태그의 안정성이 Slashdot에 존재하는 폭소노미에 실제 존재함을 보인다. 분석에 쓰인 데이터는 Slashdot으로부터 크롤하였으며 데이터의 자세한 사항은 5장에 언급되어 있다.

하나의 범주에서 사용되는 다른(distinct) 태그의 개수는 그 범주에 속하는 기사(Article)의 개수가 늘어남에 따라 같이 증가하게 된다. 우리 연구의 데이터에서 만약 태그의 중복이 전혀 없다면 기사 당 평균 4.3개의 태그가 존재함으로 기사 개수×4.3 개의 태그가 존재해야 할 것이다. 그러나 성질 1을 만족하기 위해서는 한 범주에서 과거 사용되었던 태그가 다시 사용되어야 한다. 이를 확인하기 위하여 기사의 증가에 따른 태그의 증가를 살펴본다.

그림 3에서 x축인 기사의 개수가 증가하나 태그의 개수는 선형적으로 증가하지 않음을 볼 수 있다. 이는 과거 사용되었던 태그가 어떤 형식으로든 다시 사용됨을 의미한다. 사용자들이 기사에 태그를 붙일 때 기존에 다른 사용자들이 만들었던 태그를 모방하거나 다수의 사용자들에 의하여 형성된 집단 지성을 따름으로써 이러한 결과를 만들었을 거라고 추측할 수 있다. 그러나 이 결과만을 보고 slashdot의 폭소노미에 안정성이 존재한다고는 볼 수 없다. 그 이유는 시간의 흐름에 따른 태그의 분포가 일정하다고 볼 수 없기 때문이다. 즉, 그림 3은 임의의 시간 T1과 T2에서의 태그 분포가 비슷함을 나태내지는 못한다. 우리의 연구에서는 이를 측정하기 위하여 Kullback-leibler divergence 값을 사용한다.

Kullback-leibler divergence는 두 확률 분포 값의 차이를 측정하기 위한 척도이다. 정보이론 등의 분야에서 사용되는 이 값은 두 확률 분포 P, Q가 주어졌을 때 Q를 통한 모델링이 실제 사건 P의 불확실성을 얼마나 증가시켰는지를 나타낸다. 우리의 연구에서는 P, Q는 이산 확률 분포를 나타내는 두 개의 벡터(Vector) 값이다. 이산 확률 P, Q에 적용하기 위하여 이 값은 형식적으로 다음과 같이 정의된다.

$$D_{KL}(P \parallel Q)=\sum_i P(i)\log\frac{P(i)}{Q(i)}$$

이 값은 두 확률분포가 비슷할수록 작은 값을 가지며 완전히 일치할 경우 0의 값을 가진다. Kullback-leibler divergence는 Halpin[8]의 연구에서 하나의 자원을 묘사하는 태그들의 비율이 일정해진다는 것을 보이기 위해 측도로 사용하였다. 우리는 개개의 자원들보다는 범주 전체에 존재하는 안정성에 초점을 맞춘다. 이를 위해 3.2장에서 정의한 범주 별 태그의 안정성 값을 Kullback-leibler divergence을 이용하여 측정한다. 2년 동안의 연속된 시간 T1, T2로 P(x,c)을 각각 측정하여 그 두 값을 Kullback-leibler divergence을 사용하여 관찰한다. 만약 이 값이 0으로 수렴한다면 가정하였던 조건들을 만족한 것이다. 그림 4에서 하나의 시간 점은 10일의 시간 경과를 의미한다.

위의 그림에서 볼 수 있듯이 범주에 충분한 양의 태그가 존재하지 않고 태그의 안정성이 형성되기 전인 결과의 처음 부분은 높은 Kullback-leibler divergence 값을 가짐을 볼 수 있다. 이 시기에 형성된 폭소노미는 아직 택소노미를 구성하기 위한 범주 별 태그의 안정성이 부족한 시기라고 볼 수 있다. 그러나 기대하였던 것과 같이 시간이 지나서 측정된 Kullback-leibler divergence 값은 0에 가까운 값을 보이고 있다. 이것은 우리가 정의한 안정성의 확률 값(3)이 더 이상 변화하지 않음을 의미한다. 이로부터 이 시기의 태그들은 우리가 3.2장에서 가정한 성질 1, 2를 모두 만족함을 알 수 있다. 그러므로 이 시기의 폭소노미를 이용하면 우리가 원하는 택소노미를 구축할 수 있을 것이다. 또한 그림 4는 우리에게 얼마만큼의 학습데이터가 필요한지를 가르쳐준다. Kullback-leibler divergence가 0에 근접하는 폭소노미 시스템을 구축하였으면 그 동안의 데이터를 학습 데이터로 사용하여 택소노미를 구축할 수 있는 시스템을 만들 수 있게 된다.

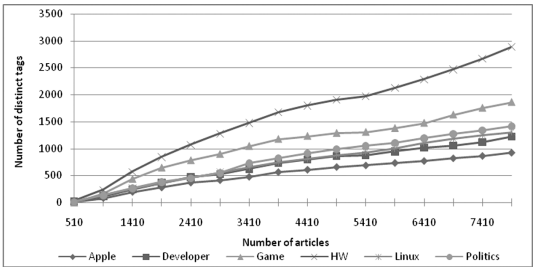


그림 3 기사의 증가에 따른 태그 개수의 증가

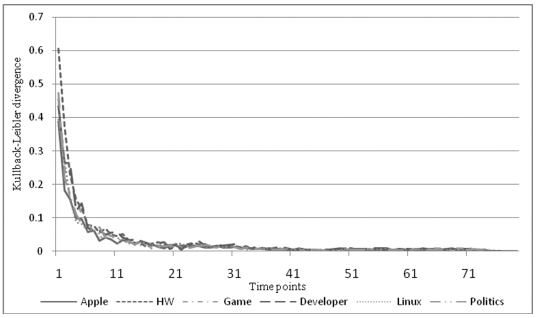


그림 4 연속된 시간에서의 P(x)의 KL divergence

4. 태그된 자원(Tagged Resource)의 분류 기법

이번 장에서는 제 3장에서 보인 범주 별 태그의 안정성을 이용하여 폭소노미로부터 택소노미를 구축하는 방법에 대하여 논의한다.

4.1 시스템 개요

우리의 시스템은 3장에서 보인 집단 지성으로 형성된 폭소노미의 안정성을 기계 학습 방법을 사용하여 학습한다. 이를 위하여 태그가 붙은 기사를 3.2장에서 정의한 태그 안정성 정보(3)를 기반으로 벡터 공간에 표시한다. 이 벡터 표현은 공간의 분리도를 높이기 위해서 아래 소개한 특징 확장 기법으로 확장된다. 이렇게 완성된 벡터 표현을 Support Vector Machine(SVM)이라는 기계학습 방법으로 학습한다. 이렇게 학습을 마친 SVM은 태그가 붙어있는 기사를 미리 정의된 범주로 분류시키는데 사용된다.

4.2 Support Vector Machine

Boser[13]에 의해 처음 제안된 Support Vector Machine(SVM)은 이진분류 문제를 풀기 위한 알고리즘이다. SVM은 Joachims[14]가 처음 텍스트 분류에 사용한 이래 텍스트 분류를 위한 효과적인 방법으로 널리 사용되어 왔다. 또한 텍스트 분류 이외에도 감성 분류, E-mail 스팸 분류 등 다양한 응용분야에서 높은 성능을 보여왔다. SVM은 오류데이터에 대한 처리능력 및 수식적으로 잘 정의될 수 있다는 특징이 있다. 이에 우리의 태그 데이터는 사용자가 직접 생성한 데이터로 몇몇의 오류 데이터를 포함한다는 점과 이를 수식적으로 분석할 수 있는 SVM을 분류기로서 선택하였다.

SVM은 다음과 같이 주어진 학습 데이터에 대해서,

$$(x_i, y_i), i = 1, ..., l \quad x_i \in R^n, \quad y_i \in (1, -1)$$

SVM은 다음의 최적화 문제에 대한 답을 찾는다.

$$\min_{w,b,\xi} \frac{1}{2} w^T \phi(x_i) + C \sum_{i=1}^{1} \xi_i$$

다음의 조건 하에 $y_i(w^T \phi(x_i)+b) \geq 1-\xi$,

이 식에서 C는 오류텀의 패널티 값이며 학습벡터 X_i는 함수 ϕ에 의하여 높은 차원의 공간으로 사상(Mapping) 된다. 이러한 방법으로 SVM은 원래의 공간보다 높은 차원의 공간에서 Margin을 최고로 하는 선형 의사결정 경계를 찾는다. 여기서 일어나는 높은 차원으로의 공간 변화는 많은 계산량을 필요로 하고 차원의 저주 문제가 발생하기 때문에 커널 트릭(Kernel trick)이라는 방법이 도입된다. 이는 원래의 속성 집합을 사용하여 변환된 공간에서의 유사성을 계산할 수 있기 때문이다. 이로 인하여 변환된 공간에서 두 사례 x_i와 x_j 사이의 유사성은 원 속성 공간에서 계산 가능하며 이는 커널 함수 $K(x_i, x_j)=\phi(x_i) \cdot \phi(x_j)$로 나타낸다.

우리의 연구에서는 일반적인 학습에서 우수한 성능을 보이는 것으로 알려진 RBF 커널을 사용하여 분류 작업을 실시한다. 한편 SVM은 이진 클래스 문제에 적용되므로 우리의 연구에서 다루는 다수의 범주를 판별하기 위하여 바로 사용할 수 없다. 우리의 연구에선 오류 교정 출력 코딩(Error-Correcting Output Coding) 방법을 사용하여 SVM으로 멀티 클래스 문제를 해결하였다.

4.3 특징 선택 / 표현(Feature Selection / Representation)

대부분의 기계학습 방법은 분류 작업을 위하여 문서의 벡터 표현을 사용한다. 이 벡터는 특징 벡터(Feature Vector)라 불리며 자원이 가진 특징들의 집합인 $\{f_1, f_2, \cdots, f_m\}$으로 표현된다. 또한 자원들은 그 자원을 기술하는 특징들의 가중치를 가진다. 이 때문에 일반적으로 $\{w(f_1), w(f_2), \cdots, w(f_m)\}$으로 표현한다. 전통적인 웹 문서를 위한 특징의 타입은 단어의 집합(Bag-of-Words)을 사용한다. 이러한 텍스트 자료에 대한 특징 선택에 대해서는 기존에 많은 방법[12]이 제안되었다. 그러나 태그를 이용하여 특징 벡터를 만드는 방법에 대한 연구는 거의 존재하지 않는다.

이에 우리는 3.2장에서 정의한 태그의 안정성 값(3)을 이용하여 태그로부터 자원의 특징벡터를 만든다. 이 안

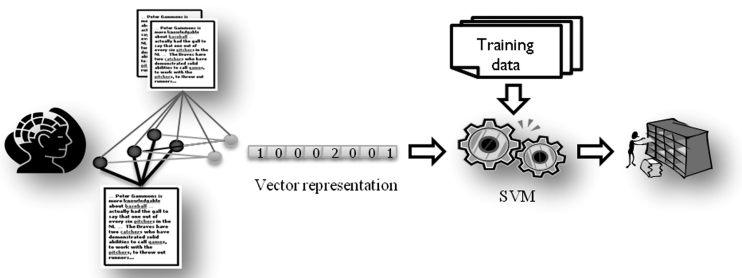


그림 5 시스템 개요

정성 값은(3) 자원이 속하는 범주의 정보를 필요로 한다. 그러나 테스트 데이터(예측을 위한 데이터)에는 이 범주 정보가 존재하지 않는다. 그러므로 특징벡터를 생성하기 위한 방법은 학습 상태를 위한 방법과 예측을 위한 방법으로 나누어진다. 먼저 학습 및 예측을 위한 특징 벡터의 생성 방법은 다음을 따른다.

정의 4. 자원 A 기술하기 위한 특징 집합

$$Futures = \{T | T\text{는 } 기사가\ 포함하고\ 있는\ 태그\} \quad (4)$$

정의 5. 특징 T_i 대한 가중치

$$W(T_i) = P(T_i, c)\ \ 단\ 여기서\ c는\ 자원\ A가\ 속한\ 범주 \quad (5)$$

위의 식 (5)의 범주 정보 c는 테스트 상태에서는 알 수 없는 값이다. 그러므로 예측 상태를 위한 c값은 다음과 같이 구한다.

Algorithm: PredictCval(SetofTags, SetofAllCategories)

Input: 1. *SetofTags*: All tags that the article 'A' has.
2. *SetofAllCategories*: All categories that the model has.
Output: Predicted category that 'A' may belongs to.

```
/*  cVal is the degree of certainty
    P(t,s) is defined in (3) /*
For each categories s ∈ SetofAllcategories do
    For all tags t ∈ SetofTags do
        cVal ← ∑t P(t, s)
    End for
    if cVal is greater than maxC
        then  maxC ← cVal
              PredictedCategory = s
End For
Return PredictedCategory
```

그림 6 예측을 위한 테스트 데이터의 범주 정보를 구하는 기법

위 방법은 각 범주 별로 태그의 안정성 값의 합을 비교하여 가장 큰 값의 범주를 반환한다. 이 정보를 가지고 식 (5)을 이용할 수 있다. 이렇게 1차적으로 구해진 특징벡터는 태그의 동시 출현 정보를 이용하여 확장된다.

동시 출현(Co-occurrence) 정보란 단어들이 한 자원을 기술하기 위해 동시에 사용되는 횟수에 관한 척도이다. 이는 동시 출현 횟수가 높을수록 그 단어들 간의 연관성이 높다고 보는 가정을 기반으로 한다. 동시 출현 횟수는 검색엔진의 성능향상이나 자연어 처리에서 기계적 방법으로 단어의 의미를 해석하기 위해서 사용된다. 우리는 이를 태그에 적용한다. 이 태그의 동시 출현 정보를 사용하여 연관성이 높은 태그를 특징벡터에 추가하는 방법을 제안한다. 이렇게 추가된 특징은 SVM이 분리시킬 의사결정 공간의 응집도를 높임으로써 훈련 오류를 줄이게 된다. 우리의 모델에서 태그 T_i와 T_j의 동시출현 정보는 cosine 거리 측정방법을 사용하여 다음과 같이 측정할 수 있다.

$$Link(T_i, T_j) = \frac{N(T_i, T_j)}{\sqrt{N(T_i) \cdot N(T_j)}} \quad (6)$$

위 식 (6)에서 $N(T_i)$는 태그 T_i가 전체 태그 공간에서 사용된 횟수이며 $N(T_i,T_j)$는 태그 T_i와 T_j가 한 자원을 기술하기 위해 동시에 사용된 횟수이다.

이를 이용하여 자원 'A'의 특징벡터를 확장하기 위해서는 자원'A'에 붙어있는 태그와 전체 태그 스페이스에 존재하는 모든 태그를 비교하고 Link 값이 높은 상위 K개의 태그를 추가하게 된다. 만약 자원 A에 m개의 태그가 있고, 전체 태그 공간에 n개의 태그가 있다면 이는 $n \times m$번은 비교 연산을 필요로 한다. 이는 상당한 계산량이므로 다음의 방법으로 그 연산 횟수를 줄인다.

Algorithm: ExtendTag(SetofTags, topK)

Input: 1. *SetofTags*: All tags that the article 'A' has.
2. *topK*: Number of tags that will be extended
Output: Extended Feature Vector

```
/*  candiateTags starts with empty set.
    Link(T1, T2) is defined in (6)  */
For each tag T ∈ SetofTags do
    Find all articles CA that share the same tag T
    For each article α in CA do
        canditateTags ← canditateTags ∪ tag that α has
    End for
End for  /* candidate tags set is made. */
For each tag T1 ∈ SetofTags do
    For each tag T2 ∈ canditateTags do
        Calculate Link(T1, T2)
    End for
End for  /* all link values are calculated. */
extendedFeatures ← Select Top K tags in candidate
                   Tags set with link value(6)
Return extendedFeatures
```

그림 7 특징벡터의 확장을 위한 기법

그림 7의 방법은 자원 'A'와 같은 태그를 공유하는 자원의 태그 집합이 그 유사도가 높을 것이라는 가정을 바탕으로 한다. 특징벡터를 확장할 자원과 태그를 공유하는 자원들을 검색한 후, 이 검색된 자원들이 가지는

태그 집합과 Link 값을 측정함으로써 그 계산량을 줄일 수 있다. 이후 상위 K개의 태그를 선택한 후 식 (5)를 사용하여 가중치를 계산하고 기존 특징벡터에 추가한다. 실제 실험에서는 이 기법을 사용하여 상당량의 정확도 향상이 있었음을 확인할 수 있었다.

5. 성능평가

5.1 실험 환경 및 데이터

실험에 쓰인 데이터는 크롤러를 구현하여 Slashdot의 2005년 12월(태그 시스템이 도입됨)부터 2008년 4월까지의 기사를 수집하였다. 크롤한 데이터는 기사의 제목, 날짜, 본문에 대한 요약, 태그 데이터, 범주 정보(Apple, Developers, Games, Hardware, Linux, Politics)를 포함한다. 범주 별 실험 데이터의 구성은 다음 표 1과 같다.

표 1 실험 데이터

	기사 수	태그 수	기사 당 평균 태그 수
Apple	8,754	35,322	4.3550
Developers	8,815	30,202	3.8026
Games	8,925	23,022	3.0164
Hardware	8,860	30,943	3.8352
Linux	8,775	32,432	4.0202
Politics	8,795	34,122	4.2011
Sum	52,924	186,043	

실험방식은 다음과 같다. 사용자의 범주 선택 값을 정답 셋으로 보고 SVM이 정답 셋을 얼마나 유추할 수 있는지를 평가한다. 이를 위하여 K-fold cross validation 방법을 사용한다. Cross validation 혹은 rotation estimation 이라 불리는 이 방법은 모델을 평가하기 위하여 널리 사용되는 평가방법이다. 이 방법은 모델이 아직까지 보지 못한 데이터에 대하여 이를 얼마나 일반화시켜 예측할 수 있는지를 효과적으로 측정할 수 있다는 장점이 있다[15]. K-fold cross validation은 전체 데이터를 k개의 부분집합(Subset)으로 나눈다. 이렇게 나누어진 k개의 부분집합 중 k-1개를 학습 데이터로 1개를 validation 데이터로 이용하는 방법이다. 우리의 실험에서는 그림 4에서 측정된 Kullback-leibler값을 기반으로 k는 5의 값을 설정하였다. 한편 본문 단어의 $tf \cdot idf$를 특징 벡터로 사용하는 기존 방법과의 비교를 위해서 다음의 실험을 설계한다. Slashdot에 존재하는 본문 요약 자료를 Bow 툴킷[1]으로 처리한 후 이로부터 특징 벡터를 만들어 SVM으로 학습한다. 그 성능은 Cross validation 방법으로 측정하였다. Slashdot의 본문은 평균적으로 105.7 단어 정도로 상대적으로 짧은 글로 이루어져 있고 멀티 미디어로(동영상, 사진, 음악)만 이루어진 자료에 대해선 본문 단어가 존재하지 않는 전형적인 웹 기사의 특징은 보인다. 우리의 실험에서는 본문 데이터가 존재하지 않는 기사에 대해서는 실패처리 하였다. 이는 기존 $tf \cdot idf$ 방법이 가지는 본질적 한계라고 볼 수 있다. 실험에 사용된 SVM의 RBF 커널함수 파라미터 값은 c는 8.0, g는 0.5를 사용하였으며 안정성 값 λ는 0.5를 설정하였다.

5.2 실험결과 및 분석

우리의 실험에서는 Slashdot의 사용자가 선택한 범주 값을 정답 셋으로 가정한다. 이 후 4장에서 언급된 특징 벡터의 생성 기법으로 태그가 부착된 리소스를 기술하고 이를 SVM을 사용하여 학습하였다. 사용자 선택의 정확도를 100%라고 보고 각 특징 벡터를 사용한 SVM이 얼마나 사용자 선택과 유사하게 예측하는지를 평가하였다. 실험결과는 표 2와 같다.

표 2 실험결과(F1-Measure)

	SVM using $tf \cdot idf$	SVM using P(x)	SVM using P(x) + Extended Features
Apple	82.216%	88.334%	94.572%
Developer	90.325%	87.261%	91.461%
Games	79.824%	86.107%	89.352%
Hardware	81.333%	84.863%	88.346%
Linux	83.450%	86.365%	91.753%
Politics	90.846%	87.833%	92.375%

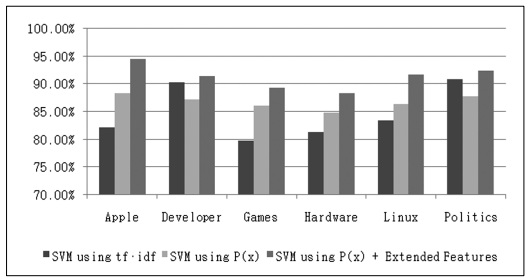


그림 8 실험결과 그래프(F1-Measure)

결과에서 볼 수 있듯이 우리가 제안한 태그 안정성 값을 이용한 특징벡터로 학습한 SVM은 우수한 성능을 보인다. 기존 $tf \cdot idf$ 방법과의 비교에서도 Developer와 Politics 범주를 제외한 나머지 모든 항목에서 우수한 결과를 보여준다. Developer와 Politics 두 항목에서 $tf \cdot idf$를 사용한 방법이 안정성 값만을 사용한 방법에 비하여 좀 더 좋은 성능을 보이는 것은 다음의 두 가지 원인에 기인하다. 첫째, 기존에 알려지지 않은 새롭게

1) Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering

추가된 태그의 비중이 본문의 단어 집단(Bag of words)의 그것보다 높았기 때문이다. 이는 이 두 범주의 자료를 묘사하는데 사용자들이 기존에 알려진 태그를 사용하지 않고 새로운 태그만을 사용하는 경향이 높았다는 것이며 이는 실험 결과에서도 보이듯이 집단 지성을 이용한 태그확장 기법으로 보완 가능하다. 두 번째 이유는 이 두 범주에서 멀티미디어 데이터 등의 비중이 상대적으로 낮았기 때문이기도 하다. 즉, 이 두 범주에 대해서는 우리가 가정한 '현대 데이터의 특징인 단순히 단어로만 이루어진 데이터보다 멀티미디어 자료를 포함한 데이터가 많다'라는 예측이 맞지 않았으며 이로 인하여 기존의 방법이 좋은 성능을 보인 것이다. 한편 이 두 범주 외의 다른 범주에는 본문에 존재하지 않는 단어가 태그로 나오는 경우가 많았는데 태그를 작성할 시 사용자의 주관적 인지 과정이 관여함을 의미한다. 이는 사용자가 생성한 태그가 분류 작업을 할 수 있을 만큼의 충분한 의미를 가지고 있다는 것이다. 이를 기반으로 태그의 동시 출현을 이용한 태그확장 기법은 추가적인 성능 향상을 가져왔음을 확인 하였다. 이로부터 태그 데이터로부터 분류 작업을 위한 특징벡터 생성시 필요 없는 태그를 버리는 작업보다는 모든 태그를 사용한 후 그와 유사한 태그들을 확장시키는 방법이 더 효과적임을 볼 수 있다. 이는 태그가 1~2개 정도로 분류 정보가 부족한 학습 데이터를 다른 사용자의 집단 지성을 이용한 동시 출현 기법(6)을 사용하여 확장시킴으로써 분류를 위한 양질의 학습 데이터를 만들었음을 의미한다. 본문의 단어를 동시 출현 관계를 이용하여 확장하는 방법은 다른 연구 결과[14]에서도 보여 주듯이 벡터 공간의 차원을 더욱 증가시켜 계산의 시간을 크게 증가시키며 벡터 공간의 분리도를 낮추어 정확도를 감소시키는 결과를 가져온다. 반면 우리가 제안한 태그 확장 기법은 태그의 총 개수가 본문의 단어의 개수보다 훨씬 적기 때문에 빠른 시간에 적용이 가능하며 분류의 정확도 향상에도 전체적으로 도움을 주기 때문에 그 효용성을 확인할 수 있다. 한편 상대적으로 HW의 예측 정확도가 떨어지는데, 이는 테스트 시 시스템에 알려지지 않은 새로운 어휘의 비율이 높았기 때문이며 그림 4에서도 확인할 수 있듯이 이는 HW의 안정성 값이 다른 범주에 비하여 떨어지는 결과를 만들었다. 이는 학습데이터의 양을 늘리고 예측된 자료의 재학습을 통하여 해결 가능하다.

6. 결론 및 향후 연구

본 논문에서는 택소노미와 폭소노미의 보완적 특징의 관찰로부터 범주 별 태그의 안정성을 발견하고 이를 이용하여 폭소노미로부터 고정된 범주로의 분류 방법을 제안하였다. 사용자의 집단 지성으로 인하여 형성되는 이 안정성은 실제로 높은 비율로 택소노미를 재현함을 확인하였다. 이는 사용자가 만든 태그 데이터가 범주를 분류할 만큼의 충분한 의미 정보를 가지고 있다는 것을 의미한다. 이러한 과정에서 본 연구는 사용자의 참여로 형성되는 태그 데이터가 가지는 새로운 의미 정보를 제시하고 이를 활용할 수 있는 방안을 고안하였다는 데에 그 의의를 둔다. 한편 우리의 시스템은 신조어 등과 같은 시스템에 처음 알려진 태그들을 처리하는 구조가 부족하다. 이는 예측된 자료의 재학습을 통하여 개선할 수 있을 것이다. 또한 자원에 태그 이외에 제목 및 본문 내용을 추가적으로 알 수 있다면 이를 태그와 결합하여 자원을 더 효과적으로 기술하는 특징벡터를 생성할 수 있을 것이다. 추후 재학습 및 본문 내용을 활용한 성능 개선에 대한 연구가 필요할 것으로 판단된다.

태그 데이터는 사용자가 직접 작성한 기술적 메타데이터로 기존의 기계적 방법으로 추론하는데 한계가 있었던 자원에 대한 정보 및 사용자에 관한 많은 정보가 포함되어 있다. 태그 데이터에서 이러한 유용한 의미 정보를 찾는다면 기존 시스템을 개선시키는 데 도움을 줄 수 있을 것으로 생각된다.

참 고 문 헌

[1] Mathes, A., *Folksonomies-Cooperative Classification and Communication Through Shared Metadata*. Computer Mediated Communication, LIS590 CMC (Doctoral Seminar), Graduate School of Library and Information Science, University of Illinois Urbana-Champaign, December, 2004.

[2] Smith, G., *Atomiq: Folksonomy: social classification*. Information Architecture, 2004. 3.

[3] Sinha, R., *A cognitive analysis of tagging*. Rashmi Sinha's weblog, available at: www. rashmisinha. com/archives/05_09/tagging-cognitive.html, 2005.

[4] *Slashdot*. http://slashdot.org.

[5] Brooks, C.H. and N. Montanez, *Improved annotation of the blogosphere via autotagging and hierarchical clustering*. Proceedings of the 15th international conference on World Wide Web, pp. 625-632, 2006.

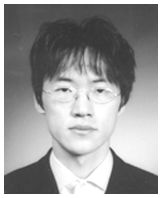
[6] Chirita, P.A., et al., *P-TAG: large scale automatic generation of personalized annotation tags for the web*. Proceedings of the 16th international conference on World Wide Web, pp. 845-854, 2007.

[7] Golder, S. and B.A. Huberman, *Usage Patterns of Collaborative Tagging Systems*. Journal of Information Science, 32(2), pp. 198-208, 2006.

[8] Halpin, H., V. Robu, and H. Shepherd, *The complex dynamics of collaborative tagging*. Proceedings of the 16th international conference on World Wide Web, pp. 211-220, 2007.

[9] Schmitz, P. *Inducing ontology from Flickr tags*. in *Proceedings of the Collaborative Web Tagging Workshop*, WWW. 2006.
[10] Li, X., L. Guo, and Y. Zhao. *Tag-based Social Interest Discovery. in Proceedings of the 17th International World Wide Web Conference*. 2008.
[11] Sebastiani, F., *Machine learning in automated text categorization*. ACM Computing Surveys (CSUR), 34(1), pp. 1-47, 2002.
[12] Yang, Y. and J.O. Pedersen, *A Comparative Study on Feature Selection in Text Categorization*. Proceedings of the Fourteenth International Conference on Machine Learning table of contents, pp. 412-420, 1997.
[13] Boser, B.E., I.M. Guyon, and V.N. Vapnik, *A training algorithm for optimal margin classifiers*. Proceedings of the fifth annual workshop on Computational learning theory, pp. 144-152, 1992.
[14] Joachims, T., *Text categorization with support vector machines: Learning with many relevant features*. Proceedings of ECML-98, 10th European Conference on Machine Learning, 1398, pp. 137-142, 1998.
[15] Kohavi, R., *A study of cross-validation and bootstrap for accuracy estimation and model selection*. Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence, 2(12), pp. 1137-1143, 1995.

고 병 걸
2006년 인하대학교 컴퓨터공학과(학사) 2008년 서울대학교 컴퓨터공학부(석사) 2008년 8월~현재 TmaxSoft R&D Center Core본부 선임 연구원으로 근무중. 관심분야는 데이타베이스, 시멘틱 웹, Web2.0.



이 강 표
2004년 연세대학교 컴퓨터과학과(학사) 2006년 서울대학교 컴퓨터공학부(석사) 2006년~현재 서울대학교 컴퓨터공학부 박사과정 재학중. 관심분야는 데이타베이스, 웹 2.0, 시맨틱웹

김 형 주
1982년 서울대학교 전산학과(학사). 1985년 Univ. of Texas at Austin(석사) 1988년 Univ. of Texas at Austin(박사). 1988년~1988년 Univ. of Texas at Austin(Post-Doc). 1988~1990년 Georgia Institute of Technology(부교수). 1991년~현재 서울대학교 컴퓨터공학부 교수. 관심분야는 데이타베이스, XML, 생물정보학, 시멘틱웹, 웹 2.0