협력적 태그를 이용한 추천 시스템

연 철[a], 김흥남[a], 지애띠[a], 조근식[b]

[a] *인하대학교 정보공학과*
인천 남구 용현동 253, 402-751
Tel: +82-32-875-5863, Fax: +82-32-875-5863, E-mail: {entireboy, nami, aerry13}@eslab.inha.ac.kr

[b] *인하대학교 컴퓨터 정보공학부*
인천 남구 용현동 253, 402-751
Tel: +82-32-860-7440, E-mail: gsjo@inha.ac.kr

요약

디지털 기기가 보편화 되면서 많은 디지털 컨텐츠가 생성되고 있다. 또한, 인터넷 서비스의 발전으로 이들 컨텐츠를 과거에 비해 손쉽게 웹 상에 개제할 수 있게 되었다. 따라서, 많은 컨텐츠를 추천해 주기 위해 추천 시스템에 관한 연구가 활발히 진행되고 있다.

이들 컨텐츠가 기존의 텍스트 기반에서 사진이나 동영상, 사운드 등 컴퓨터가 자동으로 내용을 파악하기 힘든 컨텐츠로 변화하면서, 내용의 파악이 필요 없는 협력적 여과(Collaborative Filtering)가 추천 시스템에서 유용하게 이용될 수 있다.

또한, web 2.0의 영향으로 컨텐츠를 분류하고 재검색을 용이하게 하기 위해 태깅(tagging)을 제공하는 서비스가 많아지고 있다.

본 논문에서는 내용 파악이 힘든 컨텐츠의 효과적인 추천을 위해 협력적 여과(Collaborative Filtering)와 협력적 태깅(Collaborative Tagging)을 접목시킨 방법을 제안하고, 전통적인 협력적 여과 방법과 제안한 방법의 비교 실험을 통하여 협력적 여과 방법에서의 태깅의 효과에 대해 논한다.

키워드:

추천 시스템; 태그; 협력적 태깅; 협력적 여과;

1. 서론

인터넷 기술과 서비스의 발전으로 전문적인 지식이 없이도 손쉽게 웹 상에 컨텐츠를 개제할 수 있게 되어 하루에도 수많은 컨텐츠가 생성된다. 또한 디지털 카메라나 핸드폰, 캠코더, MP3P 등 디지털 기기의 발전과 보편화로 사진이나 동영상, 사운드와 같은 디지털 컨텐츠가 많이 생성된다. 하지만, 이러한 다량의 컨텐츠들을 사용자의 선호에 맞게 선별해서 추천해 준다는 것이 쉽지만은 않다. 더욱이 기존의 웹 컨텐츠들은 대부분이 텍스트 위주였으나 서비스의 발전으로 사진이나 동영상, 사운드와 같은 컨텐츠의 비중이 비약적으로 증가하고 있다. 그리고 이들 컨텐츠는 컴퓨터가 자동적으로 어떠한 내용을 포함하고 있는지 어떤 의미를 내포하는지 정확하게 파악하기 힘들며, 개략적인 정보만을 알아내고자 해도 상당한 컴퓨팅 파워를 요한다.

이렇게 자동으로 내용을 파악하기 어려운 컨텐츠는 내용기반 여과(Content-based Filtering)로 컨텐츠를 선별하기 힘들다. 이 경우 협력적 여과(Collaborative Filtering)가 상당히 유용하게 사용될 수 있다[3, 5, 11]. 협력적 여과는 다른 많은 사용자들의 컨텐츠 선호 이력(preference history)을 바탕으로 추천 대상 사용자(target user)의 선호를 예측해 낼 수 있다[5, 11, 13]. 즉, 단지 사용자가 과거에 컨텐츠를 선호했는지 안 했는지 혹은 얼마나 선호했었는가에 대한 이력만을 바탕으로 추천해 주므로 컨텐츠가 어떤 것인지는 중요하지 않다. 그러므로 컨텐츠의 내용을 파악하기 힘든 경우에 컨텐츠를 여과해서 추천해 주기 용이하다.

또한, web 2.0의 영향으로 컨텐츠의 분류와 재검색의 용이성을 위해 태깅(tagging)을 제공하는 인터넷 서비스들이 많아졌다. 태깅은 기존의 키워드(keyword)와 동일한 개념인 태그(tag)를 컨텐츠에 붙이는 행위 자체를 가리키는 말이다. 사용자가 태깅한 태그는 사용자의 선호를 간접적으로 나타낼 수 있으며 컨텐츠에 달린 태그는 컨텐츠를 보다 정확하게 분류할 수 있다.

본 연구에서는 이러한 컨텐츠와 서비스의 변화에 맞추어 협력적 여과를 보다 효과적으로 추천에 이용하기 위해 협력적 태깅(Collaborative Tagging)을

이용한다. 사용자가 태깅한 정보를 바탕으로 사용자의 선호 경향을 파악하고, 그 경향에 맞는 컨텐츠를 여과하여 추천해 주는 방법을 제안한다.

본 논문은 2장에서 협력적 여과와 태깅에 대한 관련연구를 개략적으로 살펴보고, 3장에서 제안하고자 하는 협력적 태깅을 이용한 추천 시스템에 대해 기술한다. 4장에서는 다른 추천 시스템과의 비교 실험을 통해 제안한 방법의 효율성을 보이고, 5장에서 결론과 향후 연구에 대하여 언급한다.

2. 관련연구

2.1. 협력적 여과

협력적 여과(Collaborative Filtering)는 다른 여러 사용자의 아이템에 대한 과거 선호 이력을 바탕으로 추천 대상 사용자에게 추천해 줄 아이템(컨텐츠)을 자동적으로 여과해 주는 예측 방법이다[5, 10, 13]. 협력적 여과는 사용자의 선호 이력이 미래에도 비슷할 것이라는 가정을 바탕에 두고 있다. 예를 들어, 영화를 보는데 어떤 사용자가 과거에 공포 영화를 많이 봤 었다면 미 래에도 공포 영화를 많 이 볼 것이라는 것이다.

협력적 여과의 접근 방법에는 크게 2가지 방법이 있다[10]. 하나는 사용자 기반의 접근 방법이고, 다른 하 나는 사 용자가 선 호했던 아이 템 기반 의 접 근 방법이다. 사용자 기반의 접근 방법은 추천 대상 사용자와 유사한 선호 경 향을 가 지는 사 용자를 찾 아 그 사용자들의 과거 선호 아이템들을 추천해 주는 방법이다. 추천 대상 사용자와 다른 사용자들의 과거 선호 아이템이 얼마나 유사한가에 따라 사용자간 유사도를 구 하고 추천 대 상 사용자와 선 호 경향이 가장 유사한 (즉, 사용자간 유사도가 가장 높은) k명의 이 웃을 선 택한다. 이 k명의 가장 유사한 이웃을 이웃 집단(k nearest neighbors, *KNN*)이라고 한다[11]. 추천 대상 사용자와 각 이웃들간의 유사도를 가중치로 이웃 집단 내의 이웃들의 선호 이력을 종합하여 추천 대상 사용자의 선호를 예측하는 방법이 사용자 기반의 협력적 여과 방법이다[11].

이와 비슷하게 아이템 기반의 접근 방법은 추천 대상 사 용자가 선호 이 력을 가지고 있는 아이 템들과 가지고 있지 않은 아이템들간의 유사도를 통해 추천해 주는 방법이다. 우선, 추천 대상 사용자가 선호 이력을 가지고 있지 않은 아이템들 각각의 이웃 집단을 구한다. 이웃 집단은 사용자 기반의 방법과 동일하게 각 아이템과 다른 아이템을 선호했었던 사용자가 얼마나 유사한가에 따라 아이템간의 유사도를 통해 구한다. 추천 대상 사용자가 선호 이력을 가지고 있지 않은 아이템을 대상으로 각 아이템 별 k개의 유사한 아이템을 선택한다. 이 유사 아이템에 대한 추천 대상 사용자의 선호 이력을 아이템간의 유사도를 가중치로 하여 종합한 결과를 추천 대상 사용자의 아이템 선호로 예측하는 방법이 아이템 기반의 협력적 여과 방법이다[7].

2.2. 태깅

태그는 컨텐츠에 붙이는 키워드와 동일한 개념이고, 태깅은 컨텐츠에 사용자가 원하는 태그를 붙이는 행위 자체를 가리킨다. 태그는 한 단어든 여러 단어의 조합이든 서술형이든 사용자가 원하는 어떠한 문장으로도 사용 가능하다. 또, 하나의 컨텐츠에 여러 개의 태그를 붙일 수도 있고, 컨텐츠의 생성자 뿐만 아 니라 여 러 사용자가 태 그를 다는 것 이 가능하기도 하다.

태깅 기능은 예전부터 존재해 왔었으나 최종 사용자(end-user)에게 직접적인 이득을 가져다 줄 서비스가 존재하지 않아 그다지 많이 사용되지는 않았다. 하지만, 최근 컨 텐츠 수가 방 대해 짐에 따라 컨텐츠의 분류와 재검색의 필요성이 대두되기 시작하여 Gmail[1]이나 Flickr[2], del.icio.us[3], Technorati[4], 올블로그(allblog)[5] 등 많은 서비스에서 태깅 기능을 제공하기 시작했다. 이들 서비스에서 태깅은 카테고리(category)와 함께 컨텐츠의 분류와 재검색을 용이하게 하 기 위한 도 구로 많 이 사용된다. 태깅은 카테고리와 같은 디렉토리(directory) 형식의 수직(계층)적인 분류를 보완하기 위한 보완재로 수평적인 분류를 지원한다. 수직적인 분류를 사용할 때는 상위나 하위 분류의 포함에 따른 의미 관계 문제, 컨텐츠가 여러 분류에 포함되지 못 하는 배타적(exclusive) 포함으로 발생하는 문제, 구축의 어려움 등의 문제점이 있다[1].

2.3. 폭소노미(folksonomy)

태깅은 그 특성에 따라 몇 가지 형태를 띈다[2]. 우선, 컨텐츠에 태깅을 할 수 있는 권한에 따라서 self-tagging과 permission-based, free-for-all로 구분할 수 있다. self-tagging 방식은 올블로그(allblog)나 Technorati, YouTube[6] 등 과 같은 서 비스에서 사용하는 태깅 방식으로 컨텐츠 생성자만이 태그를 붙일 수 있고, permission-based 방식은 컨텐츠 생성자에게서 권한을 부여받은 사용자만이 컨텐츠에 태깅할 수 있는 방식으로 Flickr와 같은 서비스에서 사용하고 있다. free-for-all 방식 은 de l.icio.us나 Yahoo! MyWeb[7],

---

1 http://www.gmail.com/
2 http://www.flickr.com/
3 http://del.icio.us/
4 http://www.technorati.com/
5 http://www.allblog.net/
6 http://www.youtube.com/
7 http://myweb.yahoo.com/

Last.fm[8]처럼 모든 사용자가 컨텐츠에 태그를 붙일 수 있는 권한을 가진 방식이다.

또한, 태깅의 모음 방식에 따라 bag 형태와 set 형태로 나눌 수 있다. 한 컨텐츠에 여러 사용자들이 달아 놓은 태그의 중복을 허용하는 것이 bag 형태이고, 중복을 허용하지 않는 것이 set 형태이다. 중복을 허용하는 bag 형태는 del.icio.us나 Yahoo! MyWeb과 같은 서비스에서 사용하며, 컨텐츠에 태깅된 각 태그 별로 빈도수를 파악할 수 있다. 하지만 set 형태는 중복을 허용하지 않아 태깅되어 있는 태그의 종류만을 알 수 있으며, Flickr나 YouTube, Technorati 등의 서비스에서 사용하고 있다.

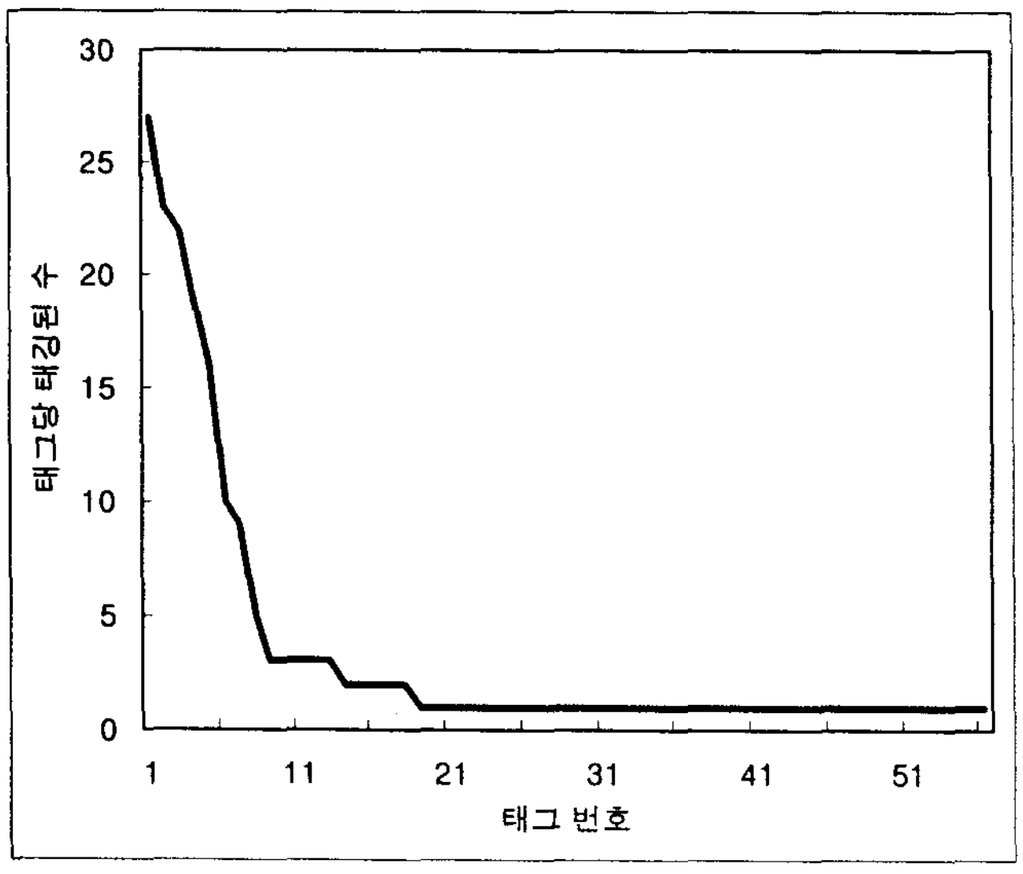

그림 1. 한 컨텐츠에 태깅되어 있는 태그 별 빈도수

bag 형태의 태깅을 이용하면 컨텐츠에 태깅되어 있는 태그의 수를 통한 통계적인 접근이 가능하다. 그림 1은 본 논문의 실험을 위한 데이터 집합의 컨텐츠(아이템) 중 한 컨텐츠에 태깅되어 있는 각 태그 별 빈도수를 나타낸 것이다. 일반적으로 한 컨텐츠에 태깅된 태그의 빈도수는 그림 1과 같은 롱테일 (long-tail) 형태의 곡선(power law curve 또는 power curve)을 그리게 된다. 한 컨텐츠에 태깅되어 있는 태그들의 대다수를 차지하는 태그는 인기 있는(popular, common) 태그(그림 1의 태그 번호 1 ~ 5)이다. 이 컨텐츠 전체를 이 태그가 표현할 수는 없지만, 최소한 이 컨텐츠는 사용 빈도가 높은 이 태그들로 대표할 수 있다. 이러한 롱테일 현상은 한 컨텐츠에 태깅할 수 있는 사용자가 많은 브로드 폭소노미(broad folksonomy)에서 두드러진다[14].

브로드 폭소노미는 del.icio.us와 같은 서비스에서 사용하는 태깅 방법(free-for-all)처럼 같은 컨텐츠에 태깅할 수 있는 사용자가 많은 태깅 방법을 뜻한다. 이에 비해 네로우 폭소노미(narrow folksonomy)는 Flickr에서 사용하는 태깅 방법(permission-based)처럼 제한된 사용자만이 태깅할 수 있는 방법을 의미한다[14].

3. 협력적 태그 기반의 추천 시스템

그림 2와 같이 본 연구에서 제안하는 추천 방법은 크게 2부분으로 나뉘어진다. 추천 대상 사용자의 선호 경향을 파악해 내는 부분과 파악한 선호 경향에 맞는 아이템을 찾아내 추천해 주는 부분으로 나뉘어진다. 추천 대상 사용자의 선호 경향은 사용자 기반의 협력적 여과 방법을 이용하여 사용자가 과거에 사용했던 태그들을 바탕으로 파악하고, 파악된 사용자의 선호 경향은 후보 태그

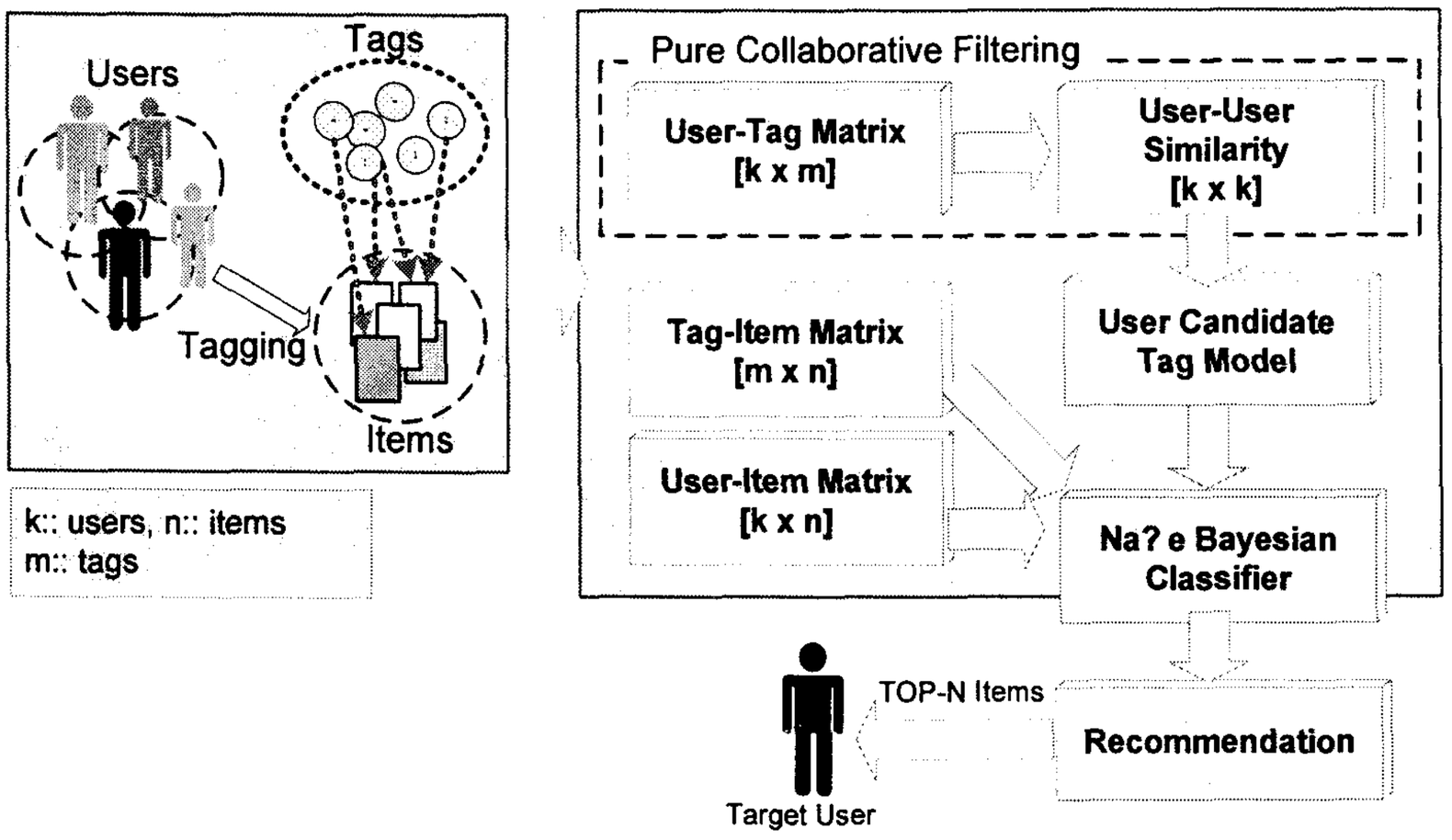

그림 2. 협력적 태그 기반의 추천 시스템 구조도

[8] http://www.last.fm/

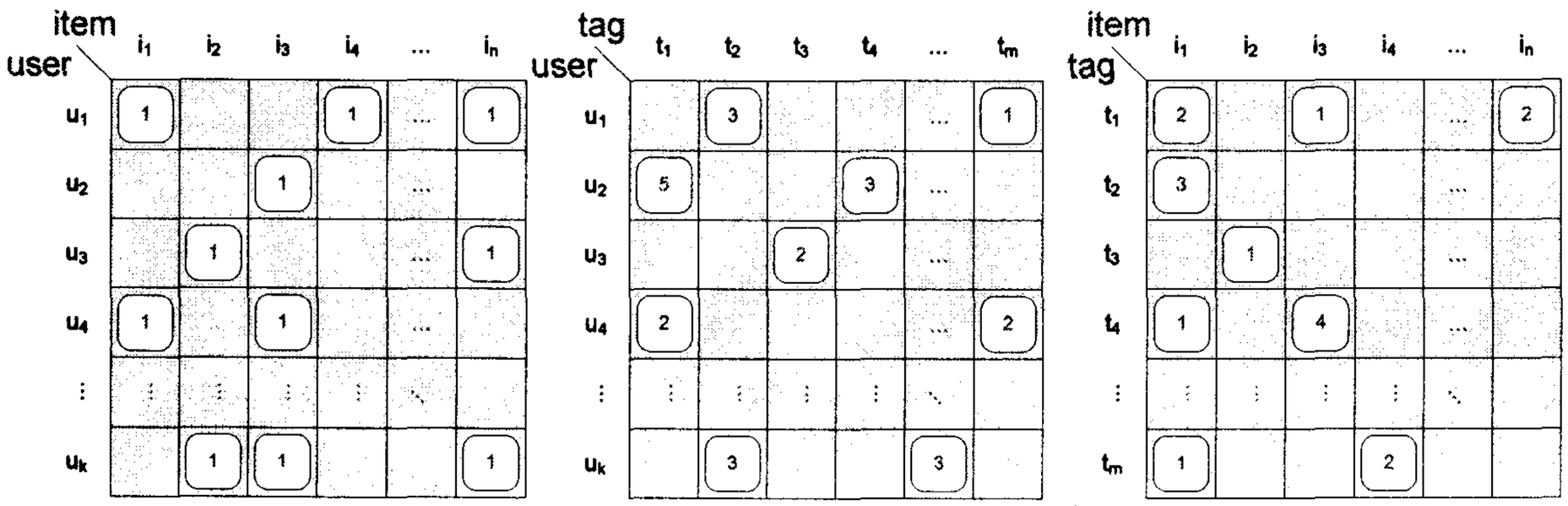

그림 3. 서로 다른 3개의 행렬

집합(Candidate Tag Set, *CTS*)으로 표현된다. 후보 태그 집합으로 표현된 사용자의 선호 경향을 나이브 베이즈 분류자(Naive Bayes Classifier)를 이용하여 후보 태그 집합의 분류(태그)에 해당하는 아이템을 추천해 주게 된다.

협력적 여과에서 사용자의 선호 이력은 사용자와 선호 대상간의 행렬로 표현할 수 있다. 본 연구에서 사용하게 되는 행렬(matrix)은 다음과 같다.

- 사용자-아이템 이진 행렬(user-item binary matrix) R : k명의 사용자 집합 $U=\{u_1, u_2, ..., u_k\}$의 n개의 아이템 집합 $I=\{i_1, i_2, ..., i_n\}$에 대한 선호 이력이 있다면, 이것은 일반적으로 그림 3의 왼쪽처럼 k × n 사용자-아이템 행렬로 표현되어 질 수 있다. 이 행렬의 행은 사용자를, 열은 아이템을 나타내며, $R_{u,i}$는 사용자 u의 아이템 i에 대한 선호 이력을 의미한다. 만약 사용자 u가 아이템 i를 선택했었다면 사용자-아이템 행렬의 u행 i열은 1의 값을 가지며, 그렇지 않다면 0의 값을 가진다. $R_{u,i} \in \{0,1\}$.
- 사용자-태그 행렬(user-tag matrix) A : 그림 3의 가운데와 같이, m개의 태그들의 집합 $T=\{t_1, t_2, ..., t_m\}$에 대하여 사용자들이 사용한 태그들의 이력을 k × m 사용자-태그 행렬로 표현될 수 있다. 이 행렬의 행은 사용자를, 열은 태그를 나타내며, $A_{u,t}$는 사용자 u의 태그 t에 대한 사용 빈도수를 의미한다.
- 태그-아이템 행렬(tag-item matrix) Q : 사용자들이 아이템들에 태깅한 태그들의 정보는 m × n 태그-아이템 행렬로 나타낼 수 있다. 이 행렬의 행은 태그를, 열은 아이템을 나타내며, $Q_{t,i}$는 사용자들에 의해 아이템 i에 태깅된 태그 t의 태깅 빈도수를 의미하며, 그림 3 의 오른쪽과 같이 나타낼 수 있다.

3.1. 후보 태그 집합 생성

협력적 여과를 이용하여 사용자-아이템 행렬 R에서 사용자의 아이템에 대한 선호 경향을 파악하여 미래에 어떤 아이템을 선호할지 예측할 수 있듯이, 본 연구에서는 사용자-태그 행렬 A에서 사용자의

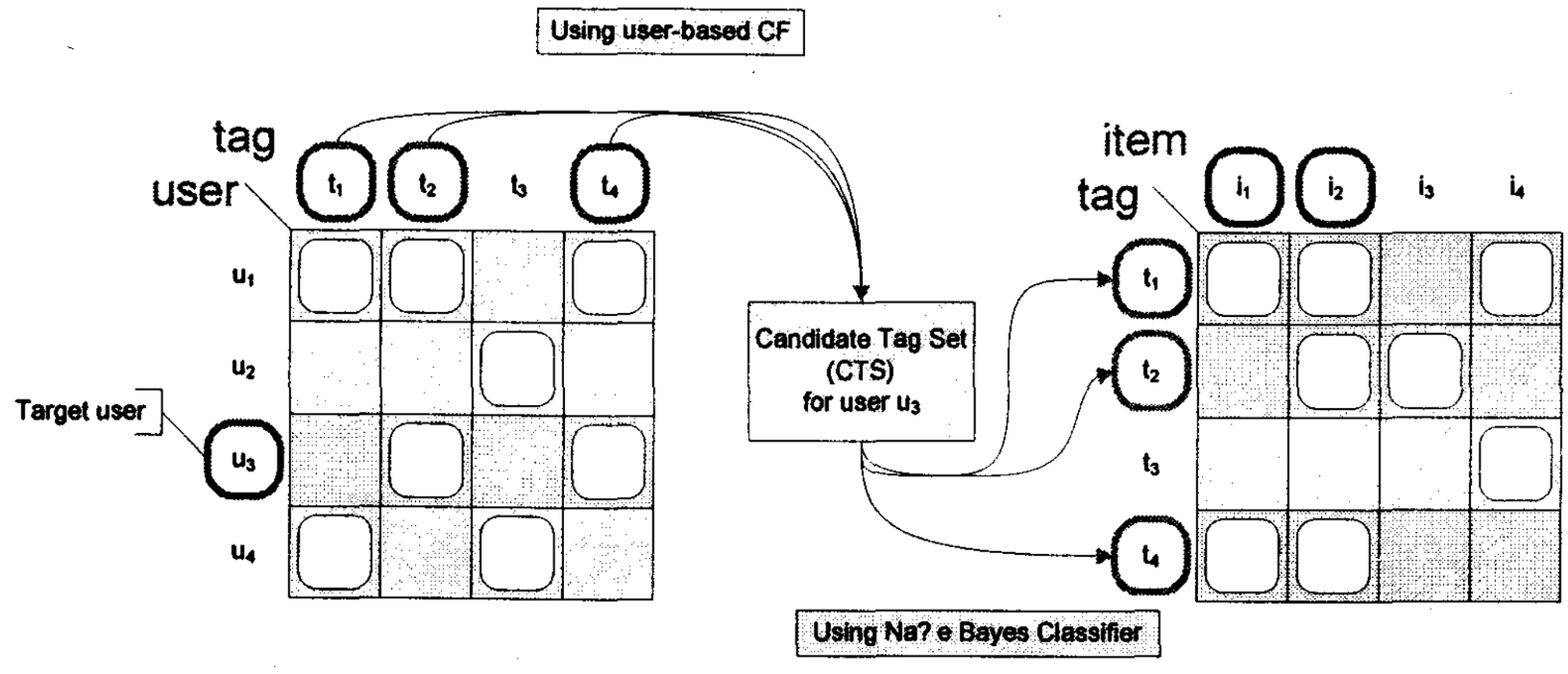

그림 4. 행렬로 나타낸 전체 흐름도

태그에 대한 선호 경향을 파악한다. 이 파악된 사용자의 경향을 후보 태그 집합(Candidate Tag Set, *CTS*)으로 정의한다. 후보 태그 집합은 사용자의 선호 경향을 나타내는 태그들의 집합으로, 사용자가 어떠한 분류(태그)에 해당하는 아이템들을 선호하는지를 알 수 있다.

- **후보 태그 집합 (Candidate Tag Set, *CTS*)** 후보 태그 집합은 사용자의 선호 경향을 나타내는 태그의 집합으로, 사용자가 어떤 분류(태그)에 해당하는 아이템들을 선호하는지를 표현한다. 사용자 u에 대한 후보 태그 집합을 $CTS_w(u)$로 표시한다.

  $CTS_w(u) = \{t_x | x=1,2,\ldots,w, t_x \in T\}$

  여기서 w는 후보 태그의 개수고 T는 전체 태그의 집합을 의미한다.

추천 대상 사용자에게 선호 경향에 맞는 아이템의 분류를 파악하고 그 분류에 맞는 아이템을 추천해 주는 것은 아이템을 직접 추천해 주는 것 보다 더 많은 아이템을 추천 대상으로 선정할 수가 있다.

예를 들어, 철수는 "지하철 노선도"와 "버스 노선도"를 북마크하였고 영희는 "지하철 노선도"와 "버스 노선도", "대중 교통 요금"을 북마크하였고 창식이는 "대중 교통 요금"을 북마크하였을 경우, 아이템을 직접 추천해주면 철수는 영희와 같은 아이템을 북마크한 이력이 있어 영희로부터 "대중 교통 요금"을 추천받을 수 있지만 창식이와는 같은 아이템을 북마크하지 않았으므로 창식이로부터는 어떠한 아이템도 추천을 받을 수 없다. 하지만 철수가 "지하철 노선도"와 "버스 노선도"에 공통적으로 "route map", "public transportation", "traffic"이란 태그를 이용하여 북마크하였고 창식이는 "대중 교통 요금"에 "public transportation", "fare"라는 태그를 이용하여 북마크하였다면, 철수는 창식이와 공통 태깅된 "public transportation"이라는 태그를 통해 창식이에게서 "public transportation"외에 "fare"라는 분류를 추천 받을 수 있고, 이 분류에 해당하는 창식이의 "대중 교통 요금"을 혹은 다른 사용자의 "public transportation"이나 "fare" 분류에 해당하는 아이템을 추천 받을 수 있다.

사용자의 선호 경향을 나타낼 때 태그를 사용하면, 카테고리를 이용한 추천보다 사용자의 선호 경향을 더 자세하고 효율적으로 나타낼 수 있다. 그리고 시간의 흐름에 따라 분류가 더 세분화되거나 다른 형태로 변형되는 경우, 카테고리 형식의 분류는 카테고리를 재구성 해야 하는 등 이에 대처하기가 힘들지만 태그를 이용한 분류는 더 자세히 태깅을 하거나 기존의 태깅된 태그의 문장을 변경하여 손쉽게 대처할 수 있다.

3.1.1. 사용자 유사도 측정

후보 태그 집합은 사용자 기반의 협력적 여과를 이용하여 추천 대상 사용자가 선호했던 태그 이력들을 바탕으로 구한다. 우선, 사용자와 비슷한 태그 선호 성향을 보이는 이웃인 이웃 집단(k nearest neighbor, *KNN*)을 구한다. 이를 위해 추천 대상 사용자와 각 사용자간의 유사도를 구해야 하는데, 본 연구에서는 코사인 유사도를 이용한다[10]. 사용자 u와 v의 유사도 $sim(u,v)$는 식 (1)과 같은 식으로 정의할 수 있다.

$$sim(u,v) = \cos(\vec{u},\vec{v}) = \frac{\sum_{t\in T} A_{u,t} \cdot A_{v,t}}{\sqrt{\sum_{t\in T}(A_{u,t})^2}\sqrt{\sum_{t\in T}(A_{v,t})^2}} \quad (1)$$

T는 전체 태그 집합을 나타내고, $A_{u,t}$와 $A_{v,t}$는 사용자-태그 행렬의 각각 사용자 u와 v가 태그 t를 이용하여 태깅한 빈도수를 나타낸다. 두 사용자가 동시에 선호한 태그의 유사도는 0 이상, 1 이하의 실수값으로 나타나며, 결과값이 클수록 두 사용자의 전체 태그에 대한 선호 경향이 비슷하다고 할 수 있다.

이와 같은 방법으로 추천 대상 사용자와 다른 사용자간의 유사도를 구하고, 유사도가 비슷한 k명의 사용자인 이웃 집단을 구한다. 이웃 집단의 크기가 너무 작으면 올바른 예측이 어려우며, 크기가 커질수록 보다 정확한 예측이 가능하지만 계산량이 늘어나므로 적당한 이웃 집단의 크기를 결정해야 한다[3, 4].

3.1.2. 태그 선호도 예측

사용자가 태그를 얼마나 선호하는가를 나타내는 선호도를 예측하는 식은 다음과 같이 정의할 수 있다[5].

$$S_{u,t} = \sum_{o\in KNN(u)} (A_{o,t}) \cdot sim(u,o) \quad (2)$$

식 (2)는 사용자 u의 태그 t에 대한 선호도 예측값 $S_{u,t}$를 구하는 식이다. $KNN(u)$는 사용자 u의 k명의 이웃인 이웃 집단이다. 이웃 집단 내의 각 이웃들의 태깅 이력 $A_{u,t}$를 추천 대상 사용자 u와 각 이웃 o간의 유사도 $sim(u,o)$를 가중치로 하여 합산한다. 추천 대상 사용자와 태그 선호 성향 유사도가 높은 이웃의 태그가 더 높은 선호도 예측값을 가지게 되는 것이다. 이렇게 계산된 추천 대상 사용자의 태그에 대한 선호도 예측값 $S_{u,t}$이 높은 태그 w개를 추천 대상 사용자의 후보 태그 집합으로 선택한다.

알고리즘 1은 전체 사용자의 각 태그에 대한 선호도를 예측하는 알고리즘이다. 사용자의 각 태그에 대한 선호도는 각 사용자와 다른 사용자간의 유사도를 가진 사용자간 유사도 행렬 D에서 가장 유사도가 높은 k명의 이웃 집단을 선택한다. 그리고 각 태그에 대한 그 이웃 집단 내의 이웃의 사용빈도와 이웃간의 유사도를 곱한 값을 합산하여

구한다.

알고리즘 1 태그 선호도 예측 알고리즘

ComputeUserTagPreferenceMatrix(*U*, *k*, *A*, *D*, *S*)

input

U : total user list
k : size of KNN
A : user-tag matrix
D : user-user similarity matrix
S : (empty) user-tag preference matrix

```
01 set all elements in matrix S with 0
02 for each u ∈ U
03     // get KNN of each user
04     for i ← 1 to r // r is row count of matrix U
05         add D_{u,i} to itemset KNN
06     for each x ∈ KNN
07         if x ≠ among the k largest values in KNN then
08             remove x from KNN
09     // compute user-tag preference matrix S
10     for each t ∈ T
11         for each x ∈ KNN
12             S_{u,t} ← S_{u,t} + (A_{x,t} × D_{u,x})
```

3.2. 나이브 베이즈 기반의 아이템 추천

본 연구에서는 추천 대상 사용자에게 확률적인 방법을 통하여 상위 N 개의 아이템을 추천하며, 확률적인 방법으로 나이브 베이즈 분류자(Naive Bayes Classifier)를 이용한다[9].

- **상위-N 추천 (Top-N Recommendation)** 모든 아이템 집합 I에 대해서 I_u는 사용자 u가 과거에 이미 구매한 아이템(또는 이미 선택한) 집합($\{R_{u,i} | R_{u,i}=1 \wedge R_{u,i} \in R\}$)이라 하고, L_u는 사용자 u가 아직 구매하지 않은 (또는 선택하지 않은) 아이템 집합($\{R_{u,i} | R_{u,i}=0 \wedge R_{u,i} \in R\}$)이라고 하자. L_u는 다음과 같은 성질을 가지고 있다. $L_u = I - I_u$, $I_u \cap L_u = \varnothing$
  사용자 u에 대한 상위-N 추천은 다음 조건$|TopN_u| \leq N$, $TopN_u \cap I_u = \varnothing$과 $TopN_u \subseteq L_u$을 만족하는 순서화 된 아이템 집합 $TopN_u$를 제공하는 것이다.

본 연구에서는 클래스 집합으로 아이템 집합 $I=\{i_1,i_2,\ldots,i_n\}$을 이용하며, 사용자 후보 태그 집합에 있는 태그들을 특징(feature)으로 사용한다. 추천 대상 사용자 u의 아이템 y에 대한 나이브 베이즈 분류자를 그림 5와 같이 베이지안 네트워크로 표현할 수 있다.

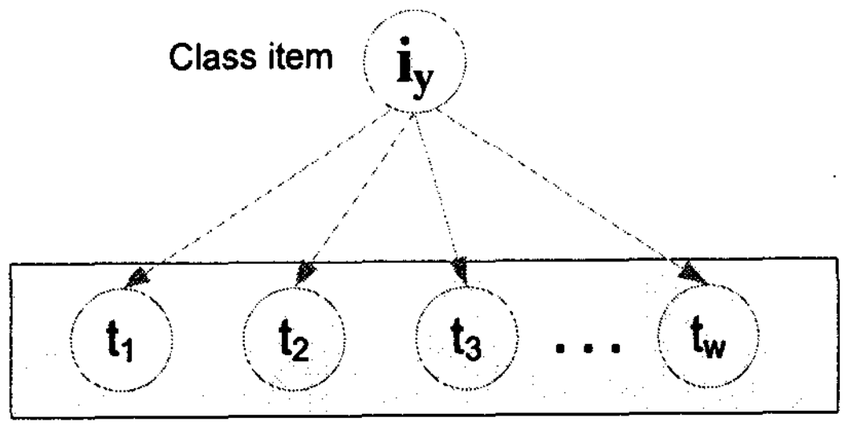

그림 5. 아이템 선호도 예측을 위한 나이브 베이즈 분류자

나이브 베이즈 분류자를 이용하여 학습을 위해서는 사전 확률(priori probability) $P(I=i_y)$와 아이템 클래스 i_y일 때의 후보 특징 j번째 태그에 대한 확률 $P(t_j|I=i_y)$값이 필요하며, 이것은 식 (3)과 식 (4)로 계산할 수 있다.

$$P(I = i_y) = \frac{\sum_{u=1}^{k} R_{u,y}}{\sum_{y=1}^{n}\sum_{u=1}^{k} R_{u,y}} \quad (3)$$

$$P(t_j \mid I = i_y) = \frac{1 + Q_{j,y}}{m + \sum_{t=1}^{m} Q_{t,y}} \quad (4)$$

여기서 $R_{u,y}$는 사용자-아이템 행렬 R에서 u번째 사용자의 y번째 아이템 값을 의미하고, $Q_{j,y}$는 태그-아이템 행렬 Q에서 j번째 태그의 y번째 아이템 값을 의미한다. 그리고 식 (4)에서는 y번째 아이템에 j번째 태그가 한번도 발생하지 않았을 때 0이 되는 것을 방지하기 위해 Laplace correction을 사용하였다[8].

각 특징 태그들이 상호 독립적(conditionally independence)이라 가정한다면, $CTS_w(u)=\{t_1,t_2,\ldots,t_w\}$의 특징 태그 집합을 가진 추천 대상 사용자 u의 아이템 y에 대한 선호도 확률 $P_{u,y}$는 다음과 같이 구할 수 있다.

$$P_{u,y} = P(I = i_y)\prod_{j=1}^{w} P(t_j|I = i_y) \quad (5)$$

최종적으로, 추천 대상 사용자 u가 아직 선호하지 않은 아이템들에 대한 확률을 모두 계산한 후 그 값이 높은 상위 N개의 아이템을 추천한다[7].

알고리즘 2는 제안하는 추천 시스템의 전체 알고리즘을 나타낸다. 전체 사용자 각각에 대한 다른 사용자간의 유사도를 계산(식 (1))하여 사용자간 유사도 행렬 D를 채운다. 각 사용자의 태그 선호도를 계산하기 위해 알고리즘 1을 수행하여 사용자의 태그별 선호 행렬 S를 계산하여

추천에 이용한다.

알고리즘 2 추천 시스템 전체 알고리즘

```
RecommenderSystemUsingCollaborativTagging
01 // generate user-user similarity matrix D
02 for each u ∈ U
03       for each v ∈ U
04             if v ≠ u
05                   D_{u,v} ← sim(u, v)
06
07 // generate user-tag preference matrix S
08 ComputeUserTagPreferenceMatrix(U, k, A, D, S)
09
10 // recommend items to each user
11 for each u ∈ U
12       Recommend(u, w, N, L_u, S)
```

알고리즘 3은 후보 태그 집합을 이용하여 상위 N개의 아이템을 추천하는 알고리즘이다. 계산된 사용자의 태그 별 선호 행렬 S에서 사용자 u의 가장 높은 선호도를 보인 w개의 태그를 선택하여, 사용자 u의 후보 태그 집합 $CTS_w(u)$를 구성한다. $CTS_w(u)$의 각 태그들을 특징으로 나이브 베이즈 분류자를 이용하여 사용자가 아직 선택하지 않은 아이템 L_u의 사전 확률을 계산하고 사전 확률이 높은 N개의 아이템을 추천해 준다.

알고리즘 3 후보 태그 집합을 이용한 추천 알고리즘

```
Recommend(u, w, N, L_u, S)

input
    u : target user
    w : size of CTS
    N : size of Top-N
    L_u : items not rated by user u
    S : user-tag preference matrix

output
    TopN_u : itemset which will be recommended to user u

01 // get CTS of user u from user-tag preference matrix S
02 for i ← 1 to m // m is column count of matrix S; same with one of matrix A
03       add S_{u,i} to itemset CTS_w(u)
04 for each x ∈ CTS_w(u)
05       if x ≠ among the w largest values in CTS_w(u) then
06             remove x from CTS_w(u)
07
08 for each i_y ∈ L_u
09       add NaiveBayesClassifier(u, CTS_w(u), i_y, Q) to itemset TopN_u // calculate by equation(5)
10
11 // recommend Top-N items to user u
12 for each z ∈ TopN_u
13       if P_{u,z} = 0 ∨ P_{u,z} ≠ among the N largest values in TopN_u then
14             remove z from TopN_u
15
16 return TopN_u
```

4. 실험 및 평가

이번 장에서는 본문에서 제안한 사용자들의 협력적 태그를 추천 시스템에 적용했을 때의 성능에 대한 실험 결과를 보이고 분석한다. 그리고, 사용자 기반의 협력적 여과 방법[11]과 아이템 기반의 협력적 여과 방법[7]을 이용하여 상위-N개의 아이템 추천에 따른 재현률(recall)을 비교 평가한다. 제안하는 시스템의 실험은 JDK 5.0과 MySQL 5.0을 이용하였으며 실험 환경은 Dual Xeon 3.0 GHz, 2.5GB RAM의 시스템 2대를 사용하였다.

4.1. 실험 데이터 및 평가 기준

본 논문의 실험에서 사용된 데이터 집합(dataset)은 소셜 북마킹(social book marking) 서비스인 del.icio.us 사이트에서 크롤링(crawling)을 통해 일부 수집하였다. del.icio.us는 사용자에게 북마크 서비스를 제공하는 사이트로, 웹 페이지를 북마크할 때 1개 이상의 태그를 태깅하도록 되어 있다. 또한, 하나의 북마크에 여러 다른 사용자들이 달아놓은 태그의 중복을 허용하는 bag 형태의 태그를 지원한다[2]. 수집된 데이터 집합은 1,544명의 사용자로부터 17,390개의 북마크 웹 사이트와 10,077개의 태그, 그리고 사용자와 웹 사이트간 27,066개의 북마킹 정보, 사용자와 태그간의 44,681개의 태깅 정보를 포함한다. (표 1)

표 1. 실험에 사용된 데이터 집합

users	items	tags	book marking	tagging
1,544	17,390	10,077	27,066	44,681

- 1544 × 17,390 사용자-아이템 이진 행렬, R
- 1544 × 10,077 사용자-태그 행렬, A
- 10,077 × 17,390 태그-아이템 행렬, Q

각 행렬 내에 얼마만큼의 선호 이력이 있는가를 판단할 수 있는 희박성 수준(Sparsity Level)은 1-(nonzero elements / total elements)로 계산할 수 있으며[3], 사용자-아이템 행렬의 희박성 수준은 0.9989, 사용자-태그 행렬의 희박성 수준은 0.9971이다.

추천의 성능평가를 위해 총 사용자가 북마크한 데이터를 80% 트레이닝 부분 (21,653개의 북마크)과 20% 테스트 부분 (5,413개의 북마크)으로 나누어 사용하였다. 성능 평가 방법으로는 알고리즘에 의해 사용자에게 추천된 아이템이 얼마나 실제로 그 사용자의 과거 선호 이력에 포함되어 있는지의 재현률(recall) 측정식을 이용하였다[7, 11]. 추천 대상 사용자에 대한 hit율은 다음과 같이 정의 될 수 있다.

$$hit - ratio(u) = \frac{|Test_u \cap TopN_u|}{|Test_u|} \quad (6)$$

여기서 $Test_u$는 테스트 집합 안에 있는 사용자 u의 아이템 집합이고 $TopN_u$는 알고리즘에 의해 사용자 u에게 추천된 상위 N개의 아이템 집합을 의미한다. 최종적으로 전체 사용자의 재현률은 식 (7)로 측정된다.

$$recall = \frac{\sum_{u=1}^{k} hit - ratio(u)}{k} \times 100 \quad (7)$$

4.2. 실험 결과 및 분석

4.2.1. 기존 알고리즘(benchmark algorithms) 실험

이웃 집단의 크기는 추천 성능에 큰 영향을 미치는 중요한 요인이다[4]. 적당한 이웃 집단의 크기를 찾아 보다 정확한 실험을 하기 위해 전통적인 협력적 여과 방법인 사용자 기반의 협력적 여과 방법[11] 유사한 사용자 집단 크기와 아이템 기반의 협력적 여과 방법[7] 유사한 아이템 집단의 크기에 따른 실험을 수행하였다. 사용자에게 추천한 아이템의 개수 N을 10개로 고정한 후 유사한 사용자 집단 크기 k를 10, 30, 50, 70, 그리고 100으로 변경하면서 추천 성능을 측정하였다.

그림 6은 이웃 집단 크기의 변화에 따른 재현률의 변화를 그래프로 나타낸 것이다. 실험 결과 대체적으로 이웃 집단의 크기가 커짐에 따라 성능은 좋아졌고, 사용자 기반의 협력적 여과는 이웃 집단의 크기가 50 부근에서 성능 증가율이 서서히 감소했다.

전반적으로 아이템 기반의 협력적 여과 방법이 사용자 기반의 협력적 여과 방법 보다 높은 정확도를 보였다. 이는 데이터 집합의 희박성 수준이 너무 높기 때문으로 분석된다[13]. 아이템 기반의 협력적 여과 방법은 사용자 기반의 협력적 여과 방법 보다 작은 모델 사이즈를 가지고도 보다 정확한 추천 성능을 보이지만 아이템의 개수가 사용자의 수 보다 너무 많아서 아이템 기반의 협력적 여과 방법은 아이템간의 유사도 계산이 상당히 오래 걸렸다.

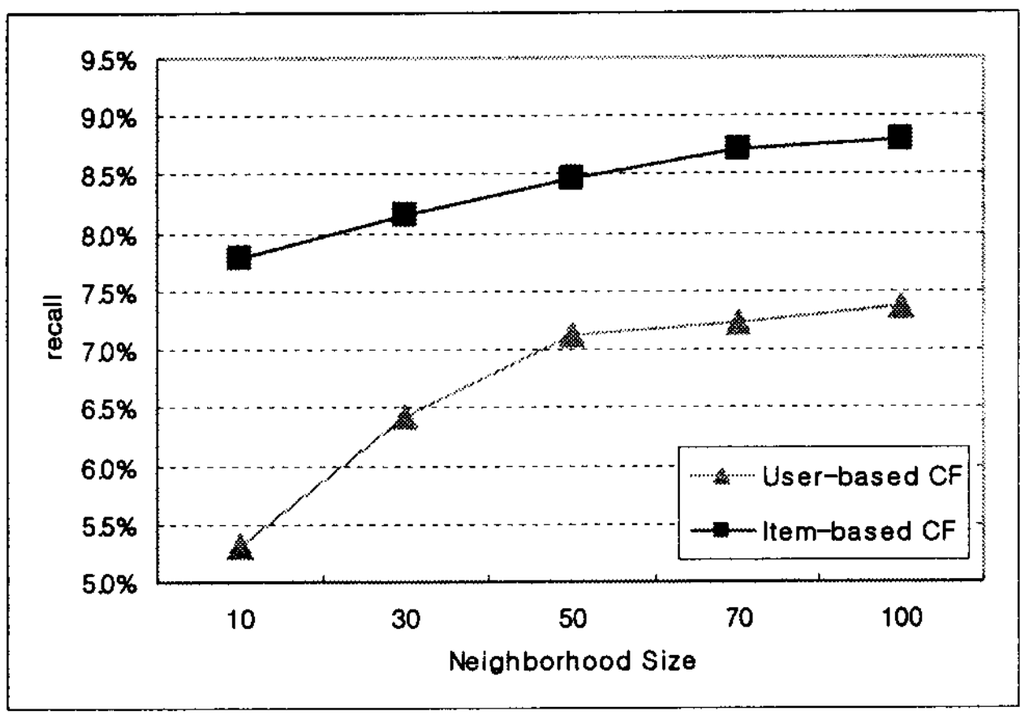

그림 6. 이웃 집단 크기에 따른 재현률

4.2.2. 후보 태그 집합의 크기에 따른 실험

전통적인 협력적 여과 방법에서 이웃 집단의 크기에 따라 성능의 차이가 있듯이, 본 연구에서 제안한 방법의 후보 태그 집단의 크기 w가 추천 성능에 영향을 미친다. 따라서 후보 태그 집단의 크기 에 따라 성능 실험을 수행하였다.

성능 측정 방법은 전통적인 협력적 여과 방법의 이웃 집단 크기에 따른 실험과 동일하게 테스트 데이터 집합의 재현률을 후보 태그 집합의 크기에 대하여 평가하였다.

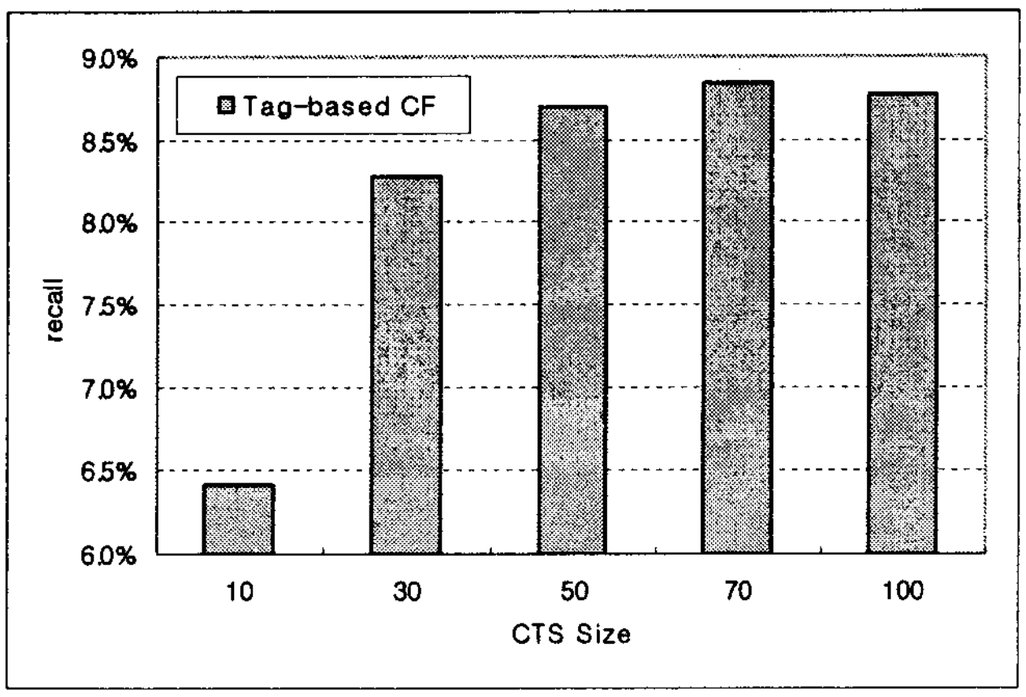

그림 7. 후보 태그 집단의 크기에 따른 재현률

후보 태그 집단의 크기를 구하기 위해 사용된 사용자-태그간의 사용자 기반의 협력적 여과에서의 이웃 집단의 크기는 50으로 하였으며, 사용 자에게 추천한 아이템의 개수 N은 10개로 고정하였다 (N=10).

결과는 그림 7와 같이 나타났으며, 일반적으로 후보 태그 집단의 크기가 증가함에 따라 성능이 향상됨을 알 수 있다. 그러나 후보 태그 집단의 크기가 70일 때 8.839%로 가장 높은 재현률을 보인 반면, 후보 태그 집단의 크기가 100인 경우 재현률 8.772%로 오히려 그 성능이 저하되었다. 이는 사용자의 선호도를 정확히 반영하지 않는 불필요한 태그가 후보 태그 집단에 포함되었기 때문으로 분석된다. 즉, 너무 많은 수의 후보 태그를 선정하는 것은 오히려 사용자의 정확한 취향을 반영하는데 좋지 않은 영향을 미칠 수 있을 뿐만 아니라 불필요한 계산량을 증가시키기 때문에, 후보 태그 집단의 적당한 크기를 결정해야 한다.

4.2.3. 성능 비교 평가

전통적인 협력적 여과 방법들과 본 연구에서 제안한 방법의 성능을 비교하기 위해서 상위-N 추천 방법의 추천하는 아이템 개수인 N을 증가시키며 성능을 비교하였다.

비교 대상인 전통적인 방법들의 이웃 집단의 크기 k는 사용자 기반의 협력적 여과 방법의 성능 상승이 감소하기 시작하는 50으로 하였으며, 제안한 방법의 후보 태그 집합의 크기 w는 이전 실험에서 가장 좋은 성능을 보인 70으로 하였다.

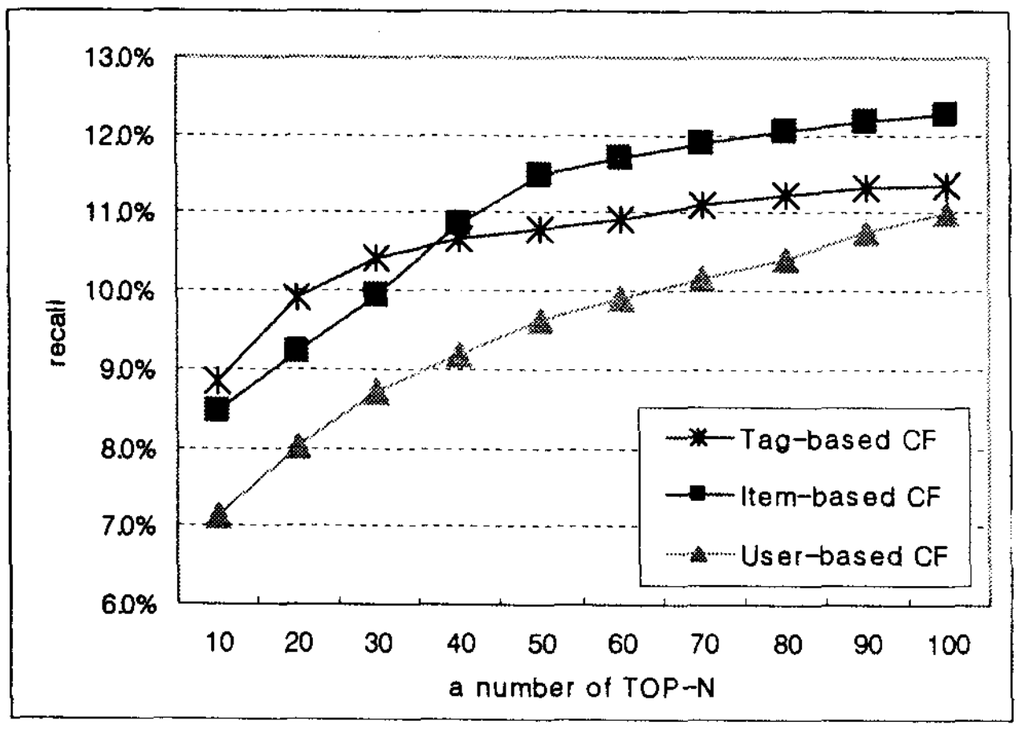

그림 8. 상위-N의 크기에 따른 재현률 비교

일반적으로 추천 아이템의 개수 N이 증가함에 재현률이 향상된다. 하지만, del.icio.us에서 수집한 데이터 집합은 모든 행렬의 밀도가 상당히 낮았고, 사용자의 수 (1,544)에 비해 아이템의 수(17,390)가 상대적으로 너무 많아 3가지 방법 모두 낮은 추천 성능을 보였다.

그림 8에서 보는 바와 같이, 본 연구에서 제안하는 방법이 사용자 기반의 협력적 여과 방법 보다는 전반적으로 좋은 성능을 보였다. 또한, 아이템 기반의 협력적 여과 방법과 비교하여 N이 커짐에 따라 성능이 떨어지는 것이 보였지만, N이 작을 때 높은 성능을 보였다. 예를 들어, N=10일 때는 4%, N=20일 때는 7%, N=30일 때는 5%의 성능 향상을 보였다. 추천 아이템의 개수가 적을 때 좋은 성능을 보인다는 것은 추천 받은 아이템 목록의 상위에 추천 대상 사용자가 선호하는 아이템이 포함될 확률이 높다는 것을 의미한다. 따라서, 협력적 태그가 아이템 추천의 성능 향상에 효과적이라는 것을 알 수 있었다.

5. 결론 및 향후 연구

web 2.0의 영향으로 다변화하는 컨텐츠들 속에 컨텐츠의 분류와 재검색의 용이성을 위한 태깅을 제공하는 서비스들이 많아졌다.

이에 따라, 본 연구에서는 협력적 태깅을 이용한 협력적 여과 방법의 추천 시스템을 제안하여 보다 더 효과적인 추천이 가능하게 하고자 하였다. 또한 실험을 통하여 추천 시스템에서 협력적 태그의 효과를 살펴 보았다.

제안한 방법과 전통적인 협력적 여과 방법을 상위-N 추천 방법으로 비교하였으며, 사용자 기반의 협력적 여과 방법 보다는 전반적으로 향상된 성능을 보였고, 아이템 기반의 협력적 여과 방법과 비교하여, 추천하는 아이템 개수 N이 큰 경우는 성능이 좋지 않았지만 N이 작은 경우 보다 좋은 성능을 보였다.

하지만, 사용자의 선호 경향을 파악하는데 있어서, "bad"나 "my work", "to read" 등 개인적이거나 감정적인 불필요한 태그로 인한 잡음(noise)이나 태그의 동음이의어, 이음동의어와 같은 문제는 추후 보완해야 할 문제이다.

향후에는 동음이의어와 같은 문제를 해결하기 위해 태그의 의미를 파악하는 의미 기반의 태그(semantic tagging)에 대한 연구와 이를 추천 시스템에 적용하는 것에 대한 연구가 요구된다.

참고문헌

[1] Scott A. Golder and Bernardo A. Huberman (2005). "The Structure of Collaborative Tagging Systems", http://arxiv.org/abs/cs/0508082

[2] Cameron Marlow, Mor Naaman, Danah Boyd, and Marc Davis (2006). "HT06, tagging paper, taxonomy, Flickr, academic article, to read", *Proceedings of the seventeenth conternce on Hypertext and hypermedia*, pp. 31-40.

[3] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl (2001). "Item-based Collaborative Filtering Recommendation Algorithms", *Proceedings of the 10th international conference on World Wide Web*, pp.285-295.

[4] Jonathan L. Herlocker, Joseph A. Konstan, Al Borchers, and John Riedl (1999). "An algorithmic framework for performing collaborative filtering", *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 230-237.

[5] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, John Riedl (1994). "GroupLens: An Open Architecture for Collaborative Filtering of Netnews", *Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work*, pp. 175-186.

[6] Scott A. Golder andBernardo A. Huberman (2006). "Usage patterns of collaborative tagging systems", *Journal of Information Science*, Volume 32, Issue 2, pp. 198-208.

[7] Mukund Deshpande and George Karypis (2004). "Item-based top-N recommendation algorithms", *ACM Transactions on Information Systems (TOIS)*, Volume 22, Issue 1, pp. 143 – 177.

[8] Jiawei Han and Micheline Kamber (2006). *Data Mining Concepts and Techniques (2nd Edition)*. Morgan Kaufmann Publishers.

[9] Mehran Sahami, Susan Dumais, David Heckerman, and Eric Horvitz (1998). "A Bayesian Approach to Filtering Junk E-Mail", Learning for Text Categorization: Papers from the 1998 Workshop.

[10] John S. Breese, David Heckerman, and Carl Kadie (1998). "Empirical Analysis of Predictive Algorithms for Collaborative Filtering", Technical Report, MSR-TR-98-12, Microsoft Research, Microsoft Corporation

[11] Badrul Sarwar. George Karypis, Joseph Konstan, and John Riedl (2000). "Analysis of Recommendation Algorithms for E-Commerce", *Proceedings of the 2nd ACM conference on Electronic commerce*, pp. 158-167.

[12] Jakob Voss (2006). "Collaborative thesaurus tagging the Wikipedia way", http://arxiv.org/abs/cs/0604036

[13] Bradley N. Miller, Joseph A. Konstan, and John Riedl (2004). "PocketLens: Toward a personal recommender system", ACM Transactions on Information Systems (TOIS), Volume 22, Issue 3, pp. 437 – 476.

[14] T. Vander Wal (2005). "Explaining and Showing Broad and Narrow Folksonomies", http://www.personalinfocloud.com/2005/02/explaining_and_.html