항목 내용물의 클러스터 정보를 고려한 협력필터링 방법의 확률적 재해석

(Probabilistic Reinterpretation of Collaborative Filtering Approaches Considering Cluster Information of Item Contents)

김 병 만 † 이 경 †† 오 상 엽 †††
(Byeong Man Kim) (Qing Li) (Sangyeop Oh)

요 약 인터넷의 상업적 이용이 증가하고 인터넷에서 쉽게 얻을 수 있는 정보의 양이 풍성해지면서 정보 필터링 (information filtering) 기법은 대량의 정보 공간에서 사용자의 요구와 기호에 맞는 항목을 찾는 과정에 널리 사용되고 있다. 많은 협력필터링 (collaborative filtering) 시스템이 사용자 평가를 기반으로 사용자나 항목들 사이의 유사성을 찾아내고 이를 바탕으로 추천을 해왔지만 사용자 편향 (user bias), 비전이 연관 (non-transitive association), cold start 문제와 같이 성능을 높이기 위해 해결해야 할 문제들이 남아있다. 이 세 가지 문제는 사용자나 항목들 사이에 더 정확한 유사도를 찾아내는 과정에 장애가 된다. 본 논문에서는 이러한 문제들을 해결하기 위해 제안된 UCHM 및 ICHM 방법을 확률적으로 재해석하였다. 이 확률적 모델은 객체 (사용자 또는 품목)들을 그룹들로 구분하고 각 그룹 내에서 사용자 평가가 가우시안 분포를 따른다는 가정 하에 사용자들이 무엇을 선호할 것인지 예측한다. 실세계 자료에 대한 실험 결과, 제안된 방식이 다른 방식들과 비교할 만한 성능을 보인다는 것을 확인할 수 있었다.

키워드 : 정보 필터링, 협력필터링, 클러스터링, 확률모델, 혼합 추천 시스템

Abstract With the development of e-commerce and the proliferation of easily accessible information, information filtering has become a popular technique to prune large information spaces so that users are directed toward those items that best meet their needs and preferences. While many collaborative filtering systems have succeeded in capturing the similarities among users or items based on ratings to provide good recommendations, there are still some challenges for them to be more efficient, especially the user bias problem, non-transitive association problem and cold start problem. Those three problems impede us to capture more accurate similarities among users or items. In this paper, we provide probabilistic model approaches for UCHM and ICHM which are suggested to solve the addressed problems in hopes of achieving better performance. In this probabilistic model, objects (users or items) are classified into groups and predictions are made for users considering the Gaussian distribution of user ratings. Experiments on a real-word data set illustrate that our proposed approach is comparable with others.

Key words : Information Filtering, Collaborative Filtering, Clustering, Probabilistic Model, Hybrid Recommender System

1. 서 론

필터링 기술을 이용한 추천시스템(recommender system)은 광범위한 항목(item) 중에서 사용자가 흥미로워 하거나 사용자에게 유용한 것을 추천하거나 그런 것을 찾을 수 있도록 안내하는 역할을 한다. 추천시스템의 적용 분야로는 책, 음악, 영화 등을 들 수 있다. 이러한 시스템은 온라인 정보의 양이 한 개인이 조사하기에는 너무 방대한 경우에 특히 유용하다. 최근에는 Amazon.com, CDNow.com, Levis.com, 등등 점점 더 많은

· 이 논문은 2004년도 한국학술진흥재단의 지원에 의하여 연구되었음 (R05-2004-000-10190-0)
† 종신회원 : 금오공과대학교 컴퓨터공학부 교수
bmkim@kumoh.ac.kr
†† 정 회 원 : 금오공과대학교 컴퓨터공학부
liqing@se.kumoh.ac.kr
††† 정 회 원 : 금오공과대학교 컴퓨터공학부 교수
syoh@kumoh.ac.kr
논문접수 : 2005년 1월 31일
심사완료 : 2005년 7월 27일

회사들이 이러한 추천시스템을 채용하고 있다.

검색엔진은 '일치(match)'라는 측면에 초점을 맞추는데, 이와 달리 추천시스템은 '개인화된(individualized)' 결과를 얻는 것에 초점을 맞춘다. 검색엔진은 질의(query)에 부합하는 모든 항목들을 일치된 정도에 따라 사용자에게 결과로서 보여준다. 초기에는 많은 추천시스템이 상당히 단순한 질의 기반의 정보검색 시스템이었는데 이들은 내용 기반(content-based) 필터링 기법을 사용했다. 그 이후 협력필터링에 기초한 GroupLens[1]와 Ringo[2]가 각각 독립적으로 개발됐다. 협력(collaborative) 추천시스템은 많은 사용자로부터 항목들에 대한 평가(rating)를 모은 다음, 이를 기반으로 특정 사용자에게 항목을 추천한다.

초창기에는, 협력필터링(CF)이란 다른 동료 사용자들의 의견을 이용하여 대상이 되는 사용자의 관심 사항을 예측하는 기법을 의미했다. 이 기법에서는 대상 사용자가 주어지면 먼저, 데이타베이스에서 그에 부합하는 이웃 사용자들을 찾고, 이웃 사용자들이 선호하는 항목을 대상 사용자에게 추천하는 것이다. 이웃 사용자는 현재까지 비슷한 종류의 관심을 가졌던 사람들로 데이타베이스에 저장되어 있는 과거의 히스토리를 바탕으로 구해진다. 그 이후, Sarwar[3]는 이 개념을 항목으로 확장하여 대상 항목이 주어지면 데이타베이스에서 그에 부합하는 이웃 항목들을 찾는 방법을 제안했다. 이웃 항목이란 사용자들에게 비슷한 평가를 받는 것들을 말한다.

협력필터링 기법은 보통 메모리 기반 방식과 모델 기반 방식의 두 종류로 구분한다. 메모리 기반 방식은 개념적으로 단순하고 직관적일 뿐 아니라 실세계에서 응용될 때 충분히 정확한 결과를 보여주기 때문에 많은 상업적 웹 사이트에서 광범위하게 이용되고 있다. 메모리 기반 방식은 사용자의 과거 평가를 데이타베이스에 기록하고 대상 사용자와 비슷한 선호 경향을 갖는 사용자들을 찾아낸다. 그리고, 이러한 비슷한 사용자들의 평가를 기반으로 대상 사용자의 평가를 예측한다. 이와는 달리, 모델 기반 알고리즘은 현재까지의 사용자들의 평가를 잘 설명하는 모델을 구축하고 이 모델을 이용하여 대상 사용자의 평가를 예측한다. 두 종류의 방식 모두 협력필터링에서 좋은 성능을 보여왔다.

일반적으로 모든 협력필터링 방법들은 유사한 '취향(taste)'을 갖는 사용자들이 항목을 비슷하게 평가한다고 가정하고 클러스터링(clustering)의 아이디어를 명시적이거나 비명시적으로 사용한다. 메모리 기반 방식에 비해, 모델 기반 방식은 더 원칙에 의거한 클러스터링을 한다. 지금까지 여러 가지 접근 방법들[4-8]이 제안되고 연구되어 왔는데, 이러한 모델들은 확률적 클러스터링(probabilistic clustering)을 통해 사용자/항목 유사성을 성공적으로 포착해 왔다. 하지만 대부분의 방법들은 순수 협력필터링에 기초하기 때문에 여전히 항목 또는 사용자 유사성을 얻기 위해 사용자 평가를 사용할 때 나타나는 세 가지 문제점 - 사용자 편향(user bias), 비전이 연관(non-transitive association), cold start 문제 - 들을 갖고 있다.

이러한 문제를 해결하기 위해 내용기반 필터링과 협력필터링을 결합하는 연구들[7,9-11]이 진행되어 왔고 본 연구진도 사용자 프로파일과 항목속성을 추가로 이용하는 메모리기반 협력필터링 방법인 UCHM(User-based Clustering Hybrid Method)[12,13]과 ICHM(Item-based Clustering Hybrid Method)[14,15] 방법을 제안하고 그 유용성을 보였다. 두 방법 모두 협력필터링의 문제점을 보완하기 위해 내용기반에 의해 형성된 항목 (또는 사용자) 그룹 정보를 협력필터링 틀 안에서 사용하는 방법으로 UCHM에서는 사용자 그룹을 항목처럼 취급하였고 반면에 ICHM에서는 항목 그룹을 마치 사용자처럼 취급하였다. 그리고, UCHM에서는 사용자기반 협력 필터링 방법을 사용한 반면에 ICHM에서는 항목기반 협력 필터링 방법을 사용하였다. 본 논문에서는, 이러한 형태의 필터링에 대한 확률적 해석을 제공하기 위한 확률 모델을 제안하고 이의 유용성을 파악하고자 한다.

2장에서는 본 연구와 관련된 연구 내용을 소개하고 3장에서는 본 논문에서 제안한 순수 협력필터링에 대한 새로운 확률모델에 대해서 소개한다. 4장에서는 내용정보를 고려한 협력필터링 방법인 UCHM 및 ICHM에 대해서 3장에서 제안한 확률 모델의 적용 방안을 살펴본다. 5장에서는 제안 확률모델의 유용성을 보이기 위한 다양한 실험 결과에 대해서 살펴본다.

2. 관련 연구

메모리 기반 방식과 모델 기반 방식 모두에서 유사한 객체를 클러스터링하는 아이디어를 사용하고 있다. 가장 광범위하게 사용되는 피어슨 상관계수(Pearson correlation coefficient) 방법과 최근에 제안된 유망한 방법들을 포함하여 대부분의 협력필터링 기법들의 궁극적인 목표는 사용자나 항목들을 유사한 취향이나 특성을 갖는 그룹으로 구분하고 이를 기반으로 적절한 항목을 사용자에게 추천하는 것이다. 따라서, 클러스터링 아이디어는 대부분의 협력필터링 기법에서, 명시적으로나 비명시적으로, 다양한 방법으로 구현된다.

모델 기반 방식은 일반적으로 클러스터링 아이디어를 명시적으로 이용하는데, 먼저 사용자 경향성을 설명하는 모델을 구축하고 이것으로부터 대상 사용자에 대한 예

측을 이끌어 낸다. 이런 모델로는 Bayesian Clustering (BC) 모델[16], Bayesian Network(BN) 모델[16], Aspect 모델[4] 등이 있다. BC 모델의 기본 아이디어는 사용자 평가가 변수들의 다항 혼합 모델의 관찰치라고 가정하는 것인데, EM[20]을 이용해 모델을 추정한다. BN 모델은 품목 사이의 종속관계를 찾아내는 것을 목표로 하는데 여기에서 각 항목은 노드(node)로 표현되고 그 사이의 종속관계는 사용자들의 행동을 관찰하여 얻는다. 한편, Aspect 모델은 숨겨진 인과(latent cause) 모델을 기반으로 하는데 이것은 사용자 집단이나 항목 그룹의 개념을 도입한다. Aspect 모델이 성공적으로 적용되면서 이것을 개선하고자 하는 여러 확률모델들이 개발돼 왔다. Flexible mixture 모델 [8]은 협동적 필터링을 위한 클러스터링 알고리즘을 확장하는데 여기에서는 각 사용자와 항목이 각각의 클러스터에 속한다는 가정을 배제하고 사용자와 항목에 대한 클러스터링을 동시에 실행한다. Preference-based graphic 모델[5]은 사용자 사이의 차이에 주목하는데 여기에서는 항목에 대해 유사한 취향을 가진 사용자들이 아주 다른 평가 패턴을 가질 수 있다는 점에 초점을 맞추었다. 즉, 어떤 사용자들은 모든 항목에 대해 다른 사용자들보다 더 높은 평가를 할 수 있다는 점에 착안하여 이에 대한 해결책을 제시하고자 했다. Popescul[7]은 추천의 질을 높이고 희소성(sparsity) 문제를 해결하기 위해 내용 정보를 함께 사용하는 방식으로 Aspect 모델을 확장했다. Hofmann[4]은 Aspect 모델을 pLSA로 확장하는데, 이것은 카테고리 분류 대신에 수적인(numerical) 평가를 지원한다.

이상의 방법들은 클러스터링의 아이디어를 확률모델이나 메모리기반 기법에 통합해서 사용하는데, 협력필터링에서 자료 클러스터링 알고리즘을 자료를 평가하기 위한 별개의 단계로서 이용하는 연구들도 있다. O'Conner[6]는 먼저 사용자 평가 자료를 기반으로 항목들을 분류하기 위해 기존의 클러스터링 알고리즘을 사용하고, 그 다음에 각 분할된 영역에서 독립적으로 예측 결과를 계산한다. SWAMI[17]는 메타-사용자 메커니즘을 추천에 적용하는데, 먼저 사용자들을 분류하여 각 클러스터의 프로필을 메타-사용자(meta-user)로서 만들고, 그 다음에 이 메타-사용자들만을 잠재적인 이웃으로 고려하면서 피어슨 상관계수 방법을 사용하여 예측 결과를 계산한다. Li와 Kim[12-15]은 협력필터링에 내용 정보를 함께 고려하기 위해 클러스터링을 적용하는 방법을 제안했다.

메모리 기반 방식이든 모델 기반 방식이든 사용자나 항목의 유사도를 찾아내기 위해서는 다음의 문제들을 해결해야 한다. 첫 번째는 비전이 연관 문제다[18]. 사용자 기반의 CF에서, 두 사용자가 비슷한 항목들을 선호하지만 유사도 계산 방법으로 인해 이 항목들이 유사한 항목으로 취급되지 않아 이들 사이의 관계를 이용할 수 없다. 이런 문제를 사용자 기반의 비전이 연관 문제라고 한다. 사용자 유사도 대신에 항목 유사도를 사용하면 이러한 사용자 기반의 비전이 문제를 피할 수 있다. 하지만 이것은 항목 기반 비전이 문제를 야기시킨다. 즉, 동일한 사용자가 두 개의 비슷한 항목을 선호하더라도 이러한 관계를 이용할 수 없어 순수한 항목기반 CF에서는 이 두 항목이 동일한 집단으로 분류될 수 없다. 비슷한 항목이 다른 집단에 들어있으면 당연히 추천의 질에 악영향을 미칠 것이다.

두 번째는 과거의 평가 기록에 대한 사용자 편향이다. 예를 들어, 사용자들로부터 동일한 평가를 받은 기록은 가지고 있지만 내용 특성이 다른 두 항목을 항목기반 협력필터링에서는 구별해 내지 못한다. 표 1이 보여주듯이, 영희와 만득이의 평가에 따르면 음악 3, 4는 음악 1, 2와 동일한 평가 기록을 가지고 있어, 항목 기반 CF에 따르면, 음악 3과 4는 철수에게 추천될 확률이 같아지게 된다. 그런데 음악 1, 2, 3이 락 음악이고 음악 4는 컨트리 음악이라면 음악 3이 음악 4에 비해 더 선호돼야 할 것이다. 왜냐하면 평가 기록에 따르면 철수가 록 음악을 선호한다는 것을 유추할 수 있기 때문이다. 이러한 문제는 또한 사용자 유사도를 찾는 과정에서도 발생한다.

표 1 사용자 편향 문제

음악 ID	철수	영희	만득	Rock	Contry
음악 1	5	4	3	98%	2%
음악 2	4	4	3	90%	10%
음악 3		4	3	98%	2%
음악 4		4	3	2%	98%
음악 5				98%	2%

세 번째는 cold start 문제다. 순수한 CF는 새로운 항목을 추천하기 어려운데 이것은 새로운 항목에 대해서는 사용자 평가 기록이 아직 없어서 이들이 어느 집단에 속하는지 결정하기 어렵기 때문이다. 이러한 문제는 새로운 사용자에 대해서도 발생하는데 Jester 추천시스템 [19]에서는 측정 집합(gauge set)을 이용해 이에 대한 부분적인 해결책을 제시하기도 했다.

3. 별개의(isolated) 클러스터링 기법을 이용한 확률 모델

필터링 대상에 따라 협력필터링은 사용자기반 방법과 항목기반 방법으로 나눌 수 있다. 따라서, 먼저 이 각각의 방법에 대한 확률모델을 제시하고 다음 장에서 이

모델들을 기초로 하여 내용정보를 고려한 메모리기반의 협력 필터링 방법인 UCHM과 ICHM을 수용하는 방법에 대해서 살펴본다.

3.1 사용자기반 협력필터링에 대한 확률모델

본 논문에서 제안하는 확률 모델은 Hoffman의 일반화된 pLSA(probabilisitic Latent Semantic Analysis) 방법 [4]을 변형한 형태이다. 일반적으로 사용자의 특정 항목에 대한 평가치는 조건부 확률 $p(v|u,y)$를 이용하여 표시할 수 있다. 그리고 예측 함수는 $g(u,y)=\int_v vp(v|u,y)dv$로 표시할 수 있다. 여기서, u는 사용자를, y는 항목을, v는 평가치를 나타내는 변수이다. 그림 1과 같은 종속구조를 바탕으로 $p(v|u,y)$를 아래와 같이 숨겨진 변수(hidden variable) z를 이용하여 이들에 대한 mixture로 정의할 수 있을 것이다.

$$p(v|u,y)=\sum_z p(v|y,z)p(z|u) \tag{1}$$

여기서, z는 사용자와 항목 사이의 숨겨진 인과관계(latent cause)를 모델링하기 위해 도입한 변수로 보면 직관적으로 이해하기가 쉬울 것이다. 즉, $p(v|y,z)p(z|u)$를 사용자 u가 z 때문에 y를 v 만큼 좋아한다로 해석할 수 있을 것이다. 그리고 Hofmann과 마찬가지로 $p(v|y,z)$가 Gaussian 모델을 따른다고 가정한다면, 즉,

$$p(v|u,y)=\sum_z p(z|u)p(v;\mu_{y,z},\sigma_{y,z}) \tag{2}$$

$$p(v;\mu,\sigma)=\frac{1}{\sqrt{2\pi\sigma}}exp\left[-\frac{(v-\mu)^2}{2\sigma^2}\right] \tag{3}$$

라 하면, 예측함수는 아래와 같이 간단히 계산할 수 있을 것이다.

$$g(u,y)=\int_v vp(v|u,y)dv=\sum_z p(z|u)\mu_{y,z} \tag{4}$$

여기서, $\mu_{y,z}$는 그룹 z에서의 항목 y에 대한 평균 평가치를, $\sigma_{y,z}$는 그룹 z에서의 항목 y에 대한 평가치의 표준편차를 나타낸다.

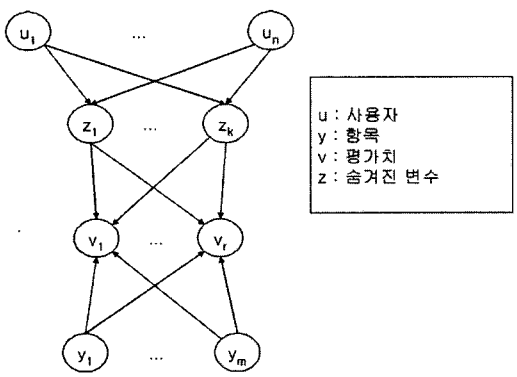

그림 1 변수들 간의 종속관계

문제는 $p(z|u)$와 $\mu_{y,z}$를 어떻게 학습 데이타를 이용하여 적절히 평가하느냐이다. Hofmann은 이를 EM 알고리즘[20]을 사용하여 아래와 같이 구하였다. 즉; E-step과 M-step 과정을 더 이상 변화가 없을 때까지 반복, 적용하여 원하는 확률들을 얻어내는 방법을 사용하였다.

E-step:

$$p(z|u,v,y)=\frac{p(v|y,z)p(z|u)}{\sum_{z'}p(v|y,z')p(z'|u)} \tag{5}$$

M-step :

$$p(z|u)=\frac{\sum_{<u',v,y>:u'=u}p(z|u,v,y)}{\sum_{z'}\sum_{<u',v,y>:u'=u}p(z'|u,v,y)} \tag{6}$$

$$\mu_{y,z}=\frac{\sum_{<u,v,y'>:y'=y}vp(z|u,v,y)}{\sum_{<u,v,y'>:y'=y}p(z|u,v,y)} \tag{7}$$

$$\sigma^2_{y,z}=\frac{\sum_{<u,v,y'>:y'=y}(v-\mu_{y,z})^2p(z|u,v,y)}{\sum_{<u,v,y'>:y'=y}p(z|u,v,y)} \tag{8}$$

본 연구에서는 EM 알고리즘 대신에 보다 직관적이고 구현하기 용이한 방법으로 $p(z|u)$와 $\mu_{y,z}$를 평가하는 방법을 사용하였다. 즉, 사용자 평가 데이타를 클러스터링 알고리즘을 사용하여 k 개의 그룹으로 나누고 각 그룹을 숨겨진 변수에 대응시키는 것이다. 이렇게 되면, $p(z|u)$는 사용자 u가 클러스터 z에 소속될 정도로 해석할 수 있을 것이며 아래와 같은 간단한 식으로 계산할 수 있을 것이다.

$$p(z|u)=\frac{ED'(V_u,V_z)^{-1}\frac{|C_z|}{|C|}}{\sum_{z'=1}^{k}ED'(V_u,V_{z'})^{-1}\frac{|C_{z'}|}{|C|}}=\frac{ED'(V_u,V_z)^{-1}|C_z|}{\sum_{z'=1}^{k}ED'(V_u,V_{z'})^{-1}|C_{z'}|} \tag{9}$$

여기서, V_u는 사용자 u의 평가 벡터를, V_z은 클러스터 z의 중심벡터를, $|C_z|$는 클러스터 z의 크기를, $|C|$는 전체 사용자의 수를 의미한다. 그리고 ED'은 아래와 같은 변형된 유클리디안 거리를 의미한다.

$$ED'(V_u,V_z)=\frac{\max(|V_u\cap V_z|,\beta)}{\beta}ED(V_u,V_z) \tag{10}$$

여기서, $ED(V_u,V_z)$는 두 벡터 사이의 유클리디안 거리를, $|V_u\cap V_z|$는 V_u와 V_z의 원소 중 두 벡터 모두가 0이 아닌 원소의 수를 나타낸다. 클러스터 z의 중심벡터는 의미적으로 클러스터 z를 대표하는 사용자의 평가 벡터로 해석할 수 있기 때문에 $|V_u\cap V_z|$는 결국사용자 u와 z가 공통으로 평가한 항목의 수를 나타낸다. β

는 임계값으로 공통으로 평가한 항목의 수가 β보다 적을 경우는 이의 가중치를 보상하기 위한 것이다.

클러스터 z에서의 항목 y에 대한 평균 평가치인 $\mu_{y,z}$는 클러스터 z에 속하는 사용자 벡터들의 가중치 평균을 통하여 얻을 수 있을 것이다.

$$\mu_{y,z} = \frac{\sum_{u \in U_z} v_{u,y} p(z|u)}{\sum_{u \in U_z} p(z|u)} \quad (11)$$

여기서, $v_{u,y}$는 사용자 u의 항목 y에 대한 평가치를, U_z는 클러스터 z에 속한 사용자의 집합으로 모든 사용자가 z에 속할 수 있기 때문에 사용자 전체집합으로 해석하면 된다.

3.2 항목기반 협력필터링에 대한 확률모델

사용자기반 협력필터링은 사용자의 평가정보를 바탕으로 유사한 사용자들을 찾고 이를 바탕으로 예측을 하는 방법을 사용한다. 반면에 항목기반 협력필터링은 항목에 대한 평가정보를 기반으로 유사한 항목들을 찾고 이를 기반으로 예측을 한다. 따라서, 항목기반 협력필터링에 대한 확률모델에서는 사용자 그룹 z를 항목 그룹으로 대치하면 된다. 즉, 다음과 같은 예측함수를 사용하여 사용자가 특정 항목을 좋아할 정도를 예측하면 된다.

$$p(v|u,y) = \sum_z p(v|u,z)p(z|y) \quad (12)$$

$$g(u,y) = \int_v vp(v|u,y)dv = \sum_z p(z|y)\mu_{u,z} \quad (13)$$

여기서, $p(z|y)$는 항목 y가 그룹 z에 포함될 확률이며, $\mu_{u,z}$는 항목 그룹 z에 속하는 항목들에 대한 사용자 u의 평균 평가치이다. $p(z|y)$는 다음과 같은 식에 의해 계산되어진다.

$$p(z|y) = \frac{ED'(V_y, V_z)^{-1}|C_z|}{\sum_{z'=1}^{k} ED'(V_y, V_{z'})^{-1}|C_{z'}|} \quad (14)$$

$$ED'(V_y, V_z) = \frac{\max(|V_y \cap V_z|, \beta)}{\beta} ED(V_y, V_z) \quad (15)$$

$\mu_{u,z}$는 그룹 z 내의 항목에 대한 사용자 평가치가 가우시안 분포를 따른다고 가정하면 아래와 같이 계산된다.

$$\mu_{u,z} = \frac{\sum_{y \in U_z} v_{u,y} p(z|y)}{\sum_{y \in U_z} p(z|y)} \quad (16)$$

4. 내용 정보를 고려한 협력필터링으로의 확장

본 장에서는 2장에서 언급한 문제들을 해결하기 위해 본 연구진들이 제안했던 UCHM 및 ICHM 필터링 방법을 3장에 제시한 확률모델을 기반으로 재해석하고자 한다.

4.1 UCHM 및 ICHM에 대한 확률 모델

사용자들은 그들의 관심사나 브라우징 또는 구매 기록 등을 나타내는 프로파일들을 갖고 있다. 이러한 프로파일은 항목에 대한 평가정보에서는 제공하지 못하는 사용자에 대한 유용한 정보를 제공한다. UCHM에서는 이러한 내용정보를 협력필터링 틀 안에서 수용하기 위해 사용자들을 클러스터링 알고리즘을 사용하여 유사한 그룹으로 묶고 각 사용자가 각 그룹에 속할 정도를 구하여 그 정보를 마치 평가정보처럼 활용한다.

대표적인 K-평균 클러스터링 알고리즘은 간단하면서도 빠른 방법으로 널리 사용되고 있다[21]. 하지만 K-평균 클러스터링 알고리즘에서는 보통 객체가 속할 그룹 하나가 정해진다. 따라서, UCHM에서는 사용자가 각 그룹에 속할 정도를 계산하기 위해서 K-평균 클러스터링 알고리즘을 목적에 맞게 변형하여 사용하였다. 이 보완 K-평균 알고리즘은 기존 K-평균 알고리즘과 거의 유사한데, 단지, 최종적으로 그룹을 형성한 후 아래와 같은 수식을 이용하여 각 그룹에 속할 정도를 계산하는 점만 틀리다. 즉, 객체가 속할 그룹 정보가 퍼지집합 형태로 표현된다.

$$\mathrm{Pro}(j,k) = 1 - \frac{CS(j,k)}{Max_i\ CS(i,k)} \quad (17)$$

여기서, $Pro(j,k)$는 객체 j가 클러스터 k에 속할 정도를 나타내며 $CS(j,k)$는 객체 j와 클러스터 k와의 역유사도(counter-similarity)로 유클리디안 거리를 사용하여 계산한다.

결과적으로, UCHM에서는 표 2에서 보는 바와 같이 사용자 프로파일 그룹을 마치 항목처럼 취급하여 원래의 항목 평가 행렬을 확장하는 것으로 볼 수 있다. 그리고 새로 추가된 가상 항목에 대한 평가 정보를 해당 사용자의 프로파일이 그 그룹에 속할 정도로 해석하고 있다. 따라서, UCHM을 3.1절에서 제시한 사용자기반 확률모델로 모델링할 수 있다. 단, 원래의 항목 평가 정보 대신에 확장된 평가 정보를 사용한다는 점만 틀리다.

표 2 사용자기반 확장 평가 행렬

	항목 평가			그룹 평가	
	항목1	항목2	항목3	그룹1	그룹2
철수	5		1	0.98	0.02
영희		4	3	0.90	0.10
만득			3	0.98	0.02

마찬가지로, 각 항목은 자신들의 속성(attribute), 예를 들어, 영화인 경우는 배우, 감독, 장르, 줄거리 등을 갖고 있다. ICHM에서는 이러한 속성정보를 항목기반 협력필터링에서 사용하기 위해, 항목들을 속성에 따라 클러스터링을 한 후 항목이 각 그룹에 속할 정도를 구하

고 이를 평가정보처럼 활용하고 있다. UCHM에서는 그룹을 마치 항목처럼 취급하지만 ICHM에서는 그룹을 사용자처럼 취급한다. 표 3에서 보는 바와 같이 각 그룹을 가상의 사용자로 보고 항목의 그룹에 속할 정도를 평가치로 해석하고 있다. 따라서, ICHM도 UCHM과 유사하게 확률 모델로 해석할 수 있다. UCHM과 다른 점은 3.2절의 항목기반 협력필터링에 대한 확률모델을 사용하고 사용하는 평가정보가 표 3의 형태의 평가 정보라는 점이다.

표 3 항목기반 확장 평가 행렬

	항목 평가			그룹 평가	
	철수	영희	만득	그룹1	그룹2
항목1	5			0.98	0.02
항목2		4		0.90	0.10
항목3	1	3	3	0.98	0.02

4.2 데이타 클러스터링

UCHM 및 ICHM에 3장의 확률모델을 적용시키기 위해서는 클러스터 또는 그룹 z를 구축해야 하는데 이는 확장된 평가 행렬에 대해서 클러스터링 알고리즘을 적용시키면 된다. 본 논문에서는 클러스터링 방법으로 k-Medoids 클러스터링 알고리즘[21]을 사용한다. 구체적인 알고리즘은 그림 2와 같다. 이 알고리즘은 초기 클러스터 선택에 따라 성능이 달라 질 수 있는데 이에 대한 해결책은 Bradly[22]가 제안한 알고리즘을 참조하기 바란다.

4.3 사용자 프로파일과 항목 속성 정보의 유용성

항목기반 필터링은 항목간의 관련성을 파악하고 이를 바탕으로 사용자가 항목을 좋아할 정도를 예측하게 된다. 이러한 접근 방법은 사용자가 전에 좋아했던 항목과 유사한 항목은 좋아하고 이전에 좋아하지 않았던 항목과 유사한 항목에 대해서는 꺼려할 것이라는 직관이 그 밑바탕에 깔려있다. 이론적 관점에서 보면, 항목기반 방법이 사용자기반 방법에 비해 더 많은 장점을 갖고 있다. 첫째, 사람은 감성적 동물이기 때문에 환경변화나 시간의 흐름에 따라 관심사가 변하기 쉬운 반면 항목은 그렇지 않다. 따라서, 항목간의 관계가 통상적으로 사용자간의 관계보다 견실하다고 할 수 있다. 둘째, 사용자 프로파일 구성은 사용자에게 부담을 준다. 비록, 사용자의 행동 패턴을 분석하여 프로파일 구성을 자동으로 할 수는 있지만 기술적인 어려움과 이론적 한계가 있다. 셋째, 사람들은 종종 자신의 관심사를 외부에 노출시키기를 꺼려한다. 최근처럼 개인들이 프라이버시에 관심을 갖는 상황에서는 더욱 그렇다.

앞에서 언급했듯이 사용자기반 협력필터링과 항목기반 협력필터링에서 사용자간 또는 항목간 유사도를 구할 시 해결해야 할 세 가지 문제가 있었다. 사용자 프로파일을 사용자기반 방법에서 활용함으로써 해결할 수

입력 : 확장 평가 행렬과 클러스터 수 m

출력 : 클러스터의 대표 벡터

i) 초기 클러스터로 랜덤하게 m 개의 객체 (UCHM인 경우는 사용자를, ICHM인 경우는 항목을)를 선택한다.

ii) 모든 객체에 대해서 최적의 클러스터에 할당한다. 최적의 클러스터는 주어진 객체와 상호연관성 (correlation)이 가장 큰 클러스터를 의미하며 다음과 같은 수식에 의해 구해진다.

$$COR_z(k) = \sum_{l \in z \wedge k \neq l} \mathrm{sim}'(k,l) = \frac{Max(|V_k \cap V_l|, \beta)}{\beta} \mathrm{sim}(k,l) \quad \text{(식 18)}$$

여기서, $COR_z(k)$는 객체 k와 클러스터 z와의 상호연관성을, V_k와 V_l은 두 객체 k와 l의 평가 벡터를 의미한다. UCHM인 경우, 객체는 사용자를 의미하기 때문에 표 2의 예를 사용하면 $V_{영희}$ = (0, 4, 3, 0.9, 0.1)이고 $V_{만득}$= (0, 0, 3, 0.98, 0.02)이다. ICHM인 경우는, 객체가 항목을 의미하기 때문에 표 3의 예를 들면 $V_{항목1}$ = (5, 0, 0, 0.98, 0.02)이고 $V_{항목2}$ = (0, 4, 0, 0.9, 0.1)이다. 그리고 $sim(k,l)$은 객체 k와 l 사이의 유사도로 [13, 15]를 참조하기 바란다.

iii) 객체의 클러스터에 변화가 없을 때까지 단계 2의 과정을 반복한다.

iv) 각 클러스터의 대표 벡터를 구성한다. 대표벡터는 클러스터를 구성하고 있는 벡터들의 평균벡터이다.

그림 2 클러스터링 알고리즘

있는 문제와 항목 속성을 항목기반 방법에 이용함으로써 해결할 수 있는 문제가 비슷하기 때문에 여기서는 항목 속성을 이용할 경우만 살펴본다.

• 항목기반 비전이 연관 문제

표 3에서 원래의 사용자 평가 정보만 사용한다면 평가 자료의 부족으로 항목 1, 2, 3간의 연관성 (또는 유사성)을 찾을 수 없다. 하지만 그룹 평가 정보를 같이 고려하면 이들간의 숨겨진 유사성을 찾을 수 있고 이로 인해 그룹 z를 형성하는 과정 중에 유용하게 사용할 수 있다.

• 사용자 편향 문제

표 1에서 만약 항목 1과 2가 같은 그룹으로 분류된다면 항목 3이나 4가 이 그룹과 관련될 정도는 같아지게 된다. 하지만, 자세히 살펴보면 항목 3이 더 관련성이 많다. 왜냐하면, 항목1과 2 모두가 락 계열의 음악이기 때문이다. 원래의 사용자 정보만 이용한다면 이러한 관련성을 찾기가 힘들다. 하지만 항목의 속성정보 (여기서는 락 계열이냐 컨트리 계열이냐)를 추가로 이용한다면 이러한 숨겨진 관계를 찾을 수 있다.

• cold start 문제

새 항목 5가 표 1처럼 데이타베이스에 추가되면 이 항목에 대한 사용자 평가 정보가 없기 때문에 이 항목에 대한 예측을 할 수가 없다. 하지만, 그룹 평가정보를 이용한다면 다른 항목들과의 관련성을 구할 수 있고 이를 바탕으로 사용자들이 이 새로운 항목을 좋아할 정도를 예측할 수 있다.

5. 실험 및 분석

위에서 살펴본 것처럼 항목 속성을 이용함으로써 앞에서 언급한 문제점들을 잘 해결할 수 있음을 알 수 있다. 하지만, 이러한 해결책은 항목 그룹 정보가 정확하다는 가정에 바탕을 두고 있다. 이의 가정의 유효성을 검증하기 위해, 즉 유사 항목 그룹이 유용한 정보를 제공하며 이러한 정보가 본 논문에서 제안한 확률모델 기반 추천시스템에 어떻게 영향을 미치는 지를 알아보기 위해 다양한 실험을 하였다.

5.1 평가 데이타 및 척도

평가 데이타는 DERC(Digital Equipment Research Center)에서 수집한 Each-Movie 데이타를 이용하였다. Each-Movie 데이타에는 1,623 개의 영화 항목이 있으며 61,265 명의 사용자가 평가한 2,800,000 개 이상의 평가 데이타가 있다. 평가는 사용자가 직접 입력한 것이며 정수형이다. 이 데이타는 본 연구진이 알고 있는 바로는 협력필터링을 위한 최대 규모의 공개 테스팅 데이타로 알고 있다.

평가는 Allbut1 프로토콜[16]에 따라 수행하였다. 즉, 최소한 두 개의 평가치를 갖고 있는 모든 사용자에 대해서 평가치중 하나를 제거시켜 이를 테스트 데이타로 사용하였다. 그리고 평가 척도는 협력필터링의 유효성 평가를 위해 널리 사용되는 두 가지 척도를 사용하였다. 이 중 하나인 MAE(Mean Absolute Error)는 실제 사용자 평가에 대하여 예측치를 비교함으로써 추천 시스템의 정확도를 평가하는 방법으로 현재 널리 사용되고 있다. MAE는 모든 테스트 대상에 대해서 평가치와 예측치 간의 오류를 구하고 이 오류의 절대값을 합한 후 테스트 대상의 수로 나누어 줌으로써 얻을 수 있다. MAE가 낮을수록 예측의 정확도는 좋아지게 된다.

$$MAE = \frac{\sum_{i=1}^{n} |p_i - q_i|}{n} \tag{19}$$

여기서, n은 평가 대상의 수를, p_i는 대상 i에 대한 예측치를, q_i는 대상 i에 대한 실 평가치를 나타낸다.

두 번째로 사용한 척도는 Breese와 그 일행[16]이 제안한 추천항목 순위리스트(ranked list)의 유용성 기대치(expected utility)이다. 이는 추천항목의 순위와 예측치 모두를 고려한 방법으로 아래와 같이 정의된다.

$$R_u = \sum_j \frac{\max(v - \overline{v}, 0)}{2^{(j-1)/(\alpha-1)}} \tag{20}$$

여기서, u는 사용자를, j는 추천시스템이 제안한 전체 항목 중에서 관심 항목의 순위를, v는 관심 항목의 평가 예측치를, $\overline{v}$는 추천 항목들의 평가 예측치에 대한 평균을, α는 사용자가 볼 가능성이 50% 정도가 되는 순위를 나타낸다. 본 논문에서는 α를 5로 하였다. 그리고 실험결과, 성능이 이 값에 민감하지 않음을 알 수 있었다. 테스트 데이타에 있는 모든 사용자의 유용성 기대치를 반영한 최종값은 아래와 같이 구해진다.

$$R = 100 \frac{\sum_u R_u}{\sum_u R_u^{\max}} \tag{21}$$

여기서, $R_u^{\max}$는 사용자 u의 관심항목들이 예측치 순으로 상위에 위치한다고 가정했을 경우의 유용성 기대치이다. Allbut1 프로토콜인 경우는 관심항목이 평가치가 제거된 항목 하나 뿐이므로 이 항목을 1위로 보았을 경우의 유용성 기대치를 계산하면 된다. 이 수식은 테스트 데이타의 크기와 예측 항목 수에 무관한 척도를 제공하며 값이 클수록 정확도가 높다.

5.2 기본 확률모델의 성능

3장의 기본 확률모델에서는 객체 그룹 z가 클러스터링 알고리즘에 의해 구해진다. 클러스터링 시 평가치가 없는 부분을 어떻게 처리하느냐에 따라 성능이 좌우된다. 참고로, Each-movie인 경우 단지 2.82%만 평가치가 주어져 있고 나머지는 없다. 따라서, 본 논문에서는

먼저 이에 대한 효과를 측정하기 위해 아래와 같은 방법에 대해 성능을 평가하였다.

- 평가치가 없는 부분을 무시. 대신에 두 객체(사용자 또는 항목) 사이의 유사도 계산 시 피어슨 상관관계 공식에 따라 두 객체가 공통으로 평가한 평가 정보만 이용.
- 평가치가 없는 부분을 0으로 채움.
- 평가치가 없는 부분을 사용자기반 협력필터링인 경우는 사용자의 평균 평가치로, 항목기반 협력필터링인 경우는 항목의 평균 평가치로 채움.

그림 3에서 보는 바와 같이 평가 부재 데이타에 대해서 무시하는 방법이 평가치를 채워 넣는 방법에 비해 성능이 좋음을 알 수 있다. 그림 3에서 PME(Probabilistic Model Estimation)는 3장에서 제안한 확률모델을 의미하며 "UPME"는 3.1절의 확률모델을, "IPME"는 3.2절의 확률모델을 의미한다. 평가 부재 데이타의 처리 방법에 따라 성능이 달라질 뿐만 아니라 클러스터의 수에 따라서도 성능이 달라진다. 그림 4에서 보는 바와 같이 사용자기반 방법에서는 k=90인 경우가 성능이 좋았으며 항목기반 방법인 경우는 k=70인 경우가 성능이 좋았다.

타 방법과의 상대적 성능평가를 위하여 기준(baseline) 방법과 메모리기반 협력필터링의 대표적 방법인 단순 피어슨(Simple Peason) 방법[23]을 구현하여 평가하였다. 기준방법은 3장의 사용자기반 확률모델과 유사하다. 단, 클러스터링 알고리즘을 사용하지 않고 랜덤하

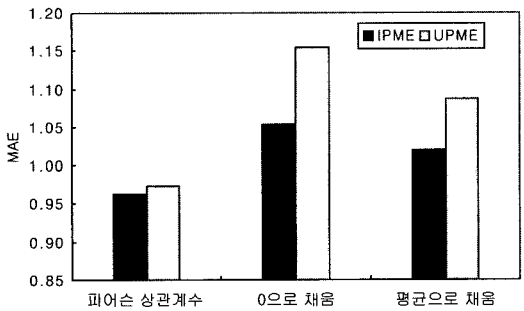

그림 3 평가 부재 데이타 처리에 따른 성능

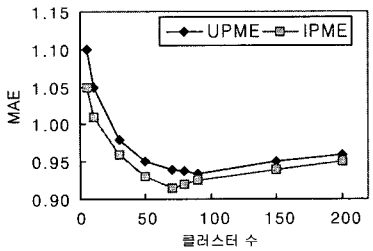

그림 4 클러스터 수에 따른 성능

표 4 성능 비교

방법	Rank Scoring / 상대적 개선도	MAE / 상대적 개선도
기준	13.46 / 0%	1.472 / 0%
사용자기반 피어슨	20.46 / 52%	1.03 / 44.3%
항목기반 피어슨	21.50 / 59.7%	0.984 / 49.5%
UPME	23.16 / 72.1%	0.952 / 54.6%
IPME	23.56 / 75%	0.946 / 55.6%

게 사용자들을 클러스터링하였다. 표 4에서 보는 바와 같이 본 제안 방법 - UPME, IPME - 의 성능이 제일 좋음을 알 수 있다.

5.3 항목 속성을 이용한 확률모델의 성능

앞에서 언급했듯이 사용자 프로파일이나 항목에서 추출한 정보가 사용자간 또는 항목간 숨겨진 관련성을 찾을 수 있음을 보았다. 하지만 Each-movie 데이타는 사용자에 대한 간단한 정보만 제공하기 때문에 이로부터 효과적인 사용자 프로파일을 구성하기가 어렵다. 따라서, 본 논문에서는 항목속성의 유용성에 대해서만 실험을 하였다.

알다시피 항목 속성을 추가로 고려할 경우는 확장된 평가행렬을 이용하여 항목의 그룹을 형성하고 이를 바탕으로 3.2절의 방법을 사용하여 사용자가 항목을 좋아할 정도를 구하게 된다. 확장된 평가행렬은 두 부분으로, 즉 원래 주어진 평가행렬과 항목 속성 정보를 그룹핑하여 얻은 그룹 평가행렬로 구성된다. 이 확장 평가행렬을 이용하여 항목간 유사도를 구하는 방법은 여러 가지가 있는데 본 실험에서는 간단히 각 부행렬을 이용한 유사도를 구하고 이를 평균하는 방법을 사용하였다. 다른 방법에 대한 내용은 [12,13,15]를 참조하기 바란다. 확장 평가행렬을 이용하여 항목 간 유사도를 구할 때 성능에 영향을 미칠 수 있는 다른 요소는 그룹 평가행렬의 그룹 수이다. 실험 결과, 그림 5에서 보는 바와 같이 그룹의 수가 40에서 성능이 좋았다. 이후 실험에서는 그룹 평가행렬의 그룹의 수를 40으로 고정하였다.

항목 속성의 유용성을 평가하기 위해 먼저 속성 정보

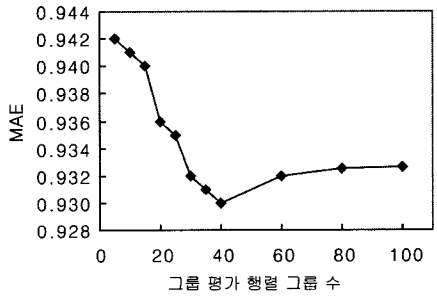

그림 5 그룹 평가 행렬의 그룹 수에 따른 성능

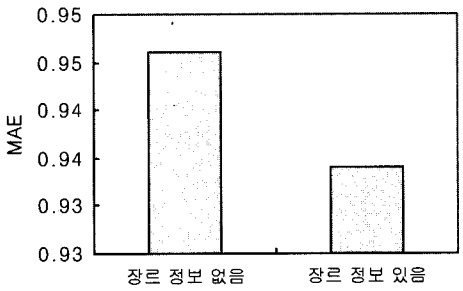

그림 6 항목 속성을 이용할 경우의 성능

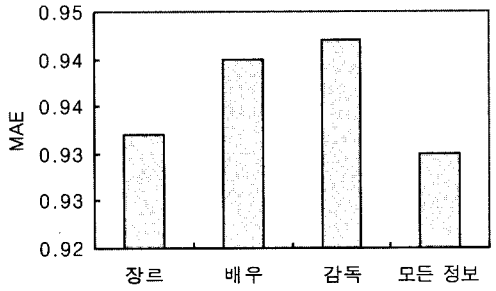

그림 7 항목 속성에 따른 성능

를 이용하지 않는 경우와 Each-movie 데이타에서 제공하는 장르 정보를 이용하여 그룹 평가행렬을 구성한 후 이를 추가적으로 고려한 경우의 성능 평가를 하였다. 그림 6에서 보는 바와 같이 항목 속성을 이용한 경우가 상당히 성능 향상에 기여함을 알 수 있다.

Each-movie 데이타에는 영화와 관련된 여러 속성 정보를 제공한다. 이런 속성정보의 상대적 중요성과 여러 속성정보를 동시에 사용할 경우의 효과를 평가하기 위해 추가적으로 장르, 배우, 연출자, 그리고 이 모두를 사용한 경우의 성능을 평가하였다. 그림 7에서 보는 바와 같이 단일 속성으로서는 장르정보가 가장 유용함을 알 수 있고 여러 속성을 동시에 고려할 경우가 좀 더 나은 성능을 보임을 알 수 있다.

위의 실험을 통하여 항목 속성정보를 이용할 경우가 그렇지 않을 경우보다 성능이 향상됨을 확인할 수 있었다. 추가로, 본 논문에서는 항목 속성정보가 2장에서 언급한 문제점들을 해결하는데 얼마나 유용한지를 살펴보기위해 두 가지 실험을 추가로 수행하였다. 먼저, 사용자 편향 문제에 대한 효과를 검증하기 위해 비전이 연관 문제와 cold-start 문제를 야기하지 않는 항목 100개를 선택한 후 본 논문에서 제안한 확률모델을 적용하여 보았다. 그림 8에서 보는 바와 같이 항목 내용을 이용하였을 경우가 그렇지 않는 경우보다 성능이 많이 향상됨을 알 수 있다.

새로운 항목은 비전이 연관 문제와 cold start 문제를 모두 갖고 있어 항목 속성이 이 두 가지 문제에 미치는 영향을 파악하기에 적당하다. 그래서 본 논문에서는, 둘째로, Each-movie 데이타에서 영화 10, 40, 60, 80, 100개를 무작위로 선택한 후 이들에 대한 평가정보를 삭제하여 마치 새로운 항목인 것처럼 만들어 실험을 하였다. 그림 9에서 "새 영화를 제외한 모든 영화"는 새 영화를 제외한 나머지 영화들만 고려한 성능이며 "새 영화"는 새 영화도 같이 고려한 경우의 성능이다. 그림 9에서 보는 바와 같이 새로운 항목을 고려해도 그렇지 않은 경우에 비해 크게 성능이 떨어지지 않음을 알 수 있다.

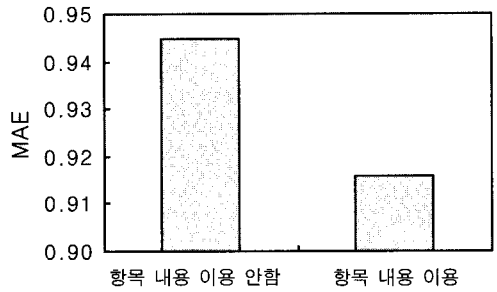

그림 8 사용자 편향 문제에 대한 성능

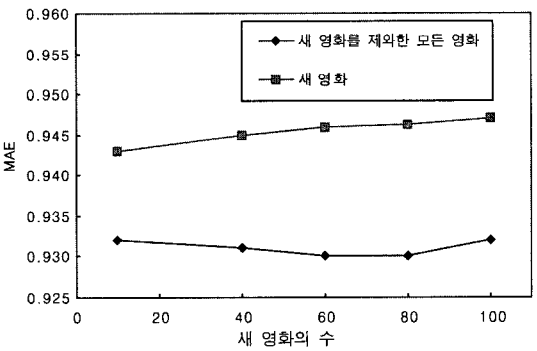

그림 9 새 항목에 대한 성능

6. 결 론

본 논문에서는 객체(사용자 또는 항목)들을 그룹으로 나누고 각 그룹에 속하는 객체의 평가치가 가우시안 분포를 따른다는 가정 하에 협력필터링에 대한 확률모델을 제시하고 이 확률모델을 기초로 하여 혼합필터링의 한 방법인 UCHM과 ICHM 방법을 재해석하였다. 또한, 협력 필터링 시 고려해야 할 세 가지 문제 - 사용자 편향 문제, 비전이 연관 문제, cold start 문제 - 가 본 확률모델에서 어떻게 해결되는 지를 보였다. 실험을 통해 본 제안 방법이 좋은 성능을 보이며 항목의 속성을 제대로 적용했을 경우 추천 질을 향상시킬 수 있음을 확인 할 수 있었다.

본 제안방법은 정수형 평가치를 이용하는 경우에 초점을 맞추어 제안되었다. 하지만, 최근, 이진 평가치가 인터넷 상에서 널리 사용되고 있고 여러 연구자[24,25]들이 이에 대한 연구들을 진행하고 있다. 본 논문에서 가정한 가우시안 분포는 이러한 이진 평가치에는 적당하지 않다. 베르누이(Bernoulli) 분포가 더 이진 데이타에 정확하다. 앞으로, 이진 평가치도 효과적으로 다룰 수 있게 베르누이 분포에 기초하여 본 제안방법을 확장하고자 한다.

참 고 문 헌

[1] Resnick, P., Iacovou, N., Suchak, M., Bergstorm, P. and Riedl, J., "GroupLens: An open architecture for collaborative filtering of Netnews," *Proc. of the ACM CSCW-94*, pp.175-186, 1994.

[2] Upendra, S. and Patti, M., "Social Information Filtering: Algorithms for Automating Word of Mouth," *Proc. of the ACM CHI'95 Conf. on Human Factors in Computing Systems*, pp.210-217, 1995.

[3] Sarwar, B. M., Karypis, G., Konstan, J. A. and Riedl, J., "Item-based Collaborative Filtering Recommendation Algorithms," *Proc. of the Tenth Int. WWW Conf. 2001*, pp.285-295, 2001.

[4] Hofmann, T., "Collaborative Filtering via Gaussian Probabilistic Latent Semantic Analysis," *Proc.of the SIGIR'03*, pp.259-266, 2003.

[5] Jin, R., Luo Si, Chengxiang Zhai, James P. Callan, "Collaborative filtering with decoupled models for preferences and rating," *Proc. of the CIKM 2003*, pp.309-316, 2003.

[6] O'Conner, M. and Herlocker, J., "Clustering items for collaborative filtering," *Proc.of the ACM-SIGIR Workshop on Recommender Systems*, 1999.

[7] Popescul, A., Lyle H. Ungar, David M. Pennock, and Steve Lawrence, "Probabilistic Models for Unified Collaborative and Content-Based Recommendation in Sparse-Data Environments," *Proc. of the 17'th Conference on UAI*, 2001.

[8] Si, Luo and Jin, Rong, "Flexible Mixture Model for Collaborative Filtering," *Proc. of the ICML 2003*, pp.704-711, 2003.

[9] M. Balabanovic and Y. Shoham, "Fab : Content-based collaborative recommendation," CACM, Vol. 40, No.3, 1997.

[10] C. Basu, H. Hirsh, and W. Cohen, "Recommendation as Classification : Using Social and Content-Based Information in recommendation," *Proc. of AAAI*, 1998.

[11] N. Good, J.B. Schafer, J.A. Konstan, A. Borchers, B. Sarwar, J. Herlocker and J. Riedl, "Combininig Collaborative Filtering with Personal Agents for Better Recommendations," *Proc. of the AAAI-99*, 1999.

[12] Q. Li and B. M. Kim, "Constructing User Profiles for Collaborative Recommender System," *Advanced Web Technologies and Applications: 6th Asia-Pacific Web Conference*, J. X. Yu, X. Lin, H. Lu and Y. Zhang, eds., LNCS 3007, Springer-Verlag, pp.100-110, 2004.

[13] 김병만, 이경, 김시관, 임은기, 김주연, "추천시스템을 위한 내용기반 필터링과 협력필터링의 새로운 결합 기법", *한국정보과학회논문지: 소프트웨어및응용*, 31권 3호, pp.332-342, 2004.

[14] Byeong Man Kim, Qing Li, Jong-Wan Kim and Jinsoo Kim, "A New Collaborative Recommender System Addressing three Problems," *PRICAI 2004 : Trends in Artificial Intelligence*, C. Zhang, H. W. Guesgen and W. K. Yeap, Eds., LNAI 3157, Springer-Verlag, pp.495-504, 2004.

[15] 김병만, 이경, "항목 속성과 평가정보를 이용한 혼합 추천 방법", *한국정보과학회논문지 : 소프트웨어 및 응용*, 31권 12호, pp.1672-1683, 2004.

[16] Breese, J. S., Heckerman, D. and Kardie, C., "Empirical Analysis of Predictive Algorithms for Collaborative Filtering," *Proc. Of the 14th UAI*, pp.43-52, 1998.

[17] Fisher,D., Kris Hildrum, Jason Hong, Mark Newman, Megan Thomas, Rich Vuduc, "SWAMI: a framework for collaborative filtering algorithm development and evaluation," *Proc. of the SIGIR 2000*, pp.366-368, 2000.

[18] Huang, Z., Hsinchun Chen and Daniel Zeng, "Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering," *ACM TOIS*, Vol. 22, No. 1, 2004.

[19] Gupta, D,, Digiovanni, M., H. Narita, K.. Goldberg, "Jester 2.0: Evaluation of an New Linear Time Collaborative Filtering Algorithm," *Proc. of the SIGIR-99*, pp.291-292, 1999.

[20] T. M. Mitchell, *Machine Learning*, McGRAW-HILL, 1997.

[21] Han, J., and Kamber, M., *Data mining: Concepts and Techniques*. New York: Morgan-Kaufman, 2000.

[22] Bradley, P. S. and Fayyad, U.M., "Refining Initial Points for K-Means Clustering," *Proc. of the ICML '98*, pp.91-99, 1998.

[23] Herlocker, J., Konstan, J., Borchers A., and Riedl, J., "An algorithmic framework for performing collaborative Filtering," *Proc. of the SIGIR-99*, pp. 230-237, 1999.

[24] Nasraoui,O. and M. Pavuluri, "Accurate Web Recommendations Based on Profile-Specific URL-Predictor Neural Networks," *Proc. of the WWW04*, 2004.

[25] Nasraoui,O. and M. Pavuluri, "A Context Ultra-Sensitive Approach to High Quality Web Recommendations based on Web Usage Mining

and Neural Network Committees," *Proc. of the International Web Dynamics Workshop of WWW04*, 2004.

김 병 만

1987년 서울대학교 컴퓨터공학과(학사) 1989년 한국과학기술원 전산학과(석사) 1992년 한국과학기술원 전산학과(박사) 1992년~현재 금오공과대학교 교수. 1998년~1999년 미국 UC, Irvine 대학 방문교수. 2005년~현재 미국 콜로라도 주립대학 방문교수. 관심분야는 인공지능, 정보검색, 프로그램 테스팅 및 검증

이 경

1999년 하얼빈 공정대학교 기계공학과(공학사). 2001년 하얼빈 공정대학교 기계공학과(공학석사). 2005년 금오공과대학교 컴퓨터공학과(공학박사). 2005년~현재 한국정보통신대학교(ICU) 박사학위 후 연수. 관심분야는 정보검색, 정보필터링, 추천시스템, 인공지능

오 상 엽

1992년 한국과학기술원 물리학 학사 1994년 한국과학기술원 전산학 석사 2001년 한국과학기술원 전자전산학 전산학 전공 박사. 2001년~2002년 (주) 서치솔루션 선임연구원. 2002년~2003년 University of Michigan 방문 연구원 2003년~2004년 한국과학기술원 초빙교수. 2004년~현재 금오공과대학교 컴퓨터공학부 전임강사. 관심분야는 정보검색, 인공지능