

추천시스템을 위한 내용기반 필터링과 협력필터링의 새로운 결합 기법

(A New Approach Combining Content-based Filtering and Collaborative Filtering for Recommender Systems)

김병만[†] 이경^{**} 김시관^{***} 임은기^{****} 김주연^{*****}
(Byeong Man Kim) (Qing Li) (Si-Gwan Kim) (En Ki Lim) (Ju-Yeon Kim)

요약 엄청난 속도로 증가하고 있는 정보의 홍수 시대에서는 정보들을 선별하기 위하여 정보 필터링 기법이 필요하다. 정보 필터링은 내용 기반 방법과 협력에 의한 방법으로 분류할 수 있다. 내용 기반 기법에서는 내용에 기반을 두어 정보를 추출하는 반면 협력 기법은 다른 사람들의 의견을 이용하게 된다. 본 논문에서는 기존 협력 필터링 방법의 문제점을 해결하기 위한 방법의 일환으로 내용 기반 기법과 협력 기법을 보다 유기적으로 결합시키는 연구를 수행하였다. 이를 위해 협력 필터링 틀을 그대로 유지하면서 사용자 프로파일을 효과적으로 이용하는 방법을 제안하였다. 또한, 본 논문에서 제시한 기법을 실험적으로 분석하고 기존의 필터링 기법과 비교하였다. 실험 결과, 본 방법이 예측 질 면에서 상당한 성능 향상이 있었고 새로운 사용자에게도 보다 나은 추천을 할 수 있음을 알 수 있었다.

키워드 : 정보 필터링, 추천 시스템, 협력 필터링, 내용기반 필터링, 클러스터링

Abstract With the explosive growth of information in our real life, information filtering is quickly becoming a popular technique for reducing information overload. Information filtering technique is divided into two categories: content-based filtering and collaborative filtering (or social filtering). Content-based filtering selects the information based on contents; while collaborative filtering combines the opinions of other persons to make a prediction for the target user. In this paper, we describe a new filtering approach that seamlessly combines content-based filtering and collaborative filtering to take advantages from both of them, where a technique using user profiles efficiently on the collaborative filtering framework is introduced to predict a user's preference. The proposed approach is experimentally evaluated and compared to conventional filtering. Our experiments showed that the proposed approach not only achieved significant improvement in prediction quality, but also dealt with new users well.

Key words : Information Filtering, Recommender System, Collaborative Filtering, Content-based Filtering, Clustering

1. 서론

최근에 인터넷과 네트워크를 통하여 제공되는 정보들이 폭발적으로 증가하면서 대량의 정보들을 효과적으로 탐색할 수 있는 자동화된 도구에 관한 연구가 증가하고 있으며, 이러한 연구 결과로서 제안된 기술이 정보 검색과 정보 필터링이다[1]. 정보 검색은 사용자의 정보 요구가 가변적인 상황에서 비교적 안정적인 문서들을 대상으로 사용자의 요구를 만족하는 문서들을 추천하는 방법이며, 정보필터링은 사용자의 정보 요구가 비교적 안정적인 상황에서 가변적인 문서들을 대상으로 사용자의 요구를 만족하는 문서들을 추천하는 방법이다. 정보 필터링은 사용자의 요구 정보를 기반으로 문서를 추천하고 있으며, 추천 방법은 내용기반 필터링(CBF :

· 이 논문은 2002년도 한국학술진흥재단의 지원에 의하여 연구되었음 (KRF-2002-041-D00459)

† 종신회원 : 금오공과대학교 컴퓨터공학과 교수
bmkim@se.kumoh.ac.kr
** 학생회원 : 금오공과대학교 컴퓨터공학과
liqing@se.kumoh.ac.kr
*** 정 회원 : 금오공과대학교 컴퓨터공학과 교수
sgkim@se.kumoh.ac.kr
**** 종신회원 : 금오공과대학교 컴퓨터공학부 교수
eklim@kumoh.ac.kr
***** 비 회원 : 부천대학 전산정보처리과 교수
jykim@bc.ac.kr
논문접수 : 2003년 1월 11일
심사완료 : 2003년 12월 5일

Content-Based Filtering][2-6]과 사회기반 필터링(SF : Social Filtering)[7-14]으로 분류할 수 있다.

내용 기반 필터링은 정보의 내용과 사용자 프로파일의 내용을 비교하여 사용자에게 적절한 정보를 선택해 준다. 내용 기반 정보 필터링은 정보검색의 주요 모델인 불리안 모델, 벡터공간 모델, 확률모델, 인공지능경망 모델, 퍼지집합 모델과 같은 기법을 사용하여 주제에 적절한 텍스트 정보를 찾아 주는데 아주 효과적인 것이 증명되었다. 그러나, 내용 기반 기법은 항목(찾고자 하는 정보)은 반드시 기계가 분류할 수 있는 형태(예: 텍스트)로 되어 있어야 하고, 프로파일에 존재하는 단어와 항목의 내용에 나타나는 단어가 정확하게 일치하지 않을 경우 항목을 선택할 수 없는 용어 불일치 문제를 가지고 있다.

사회기반 필터링은 협력 필터링(Collaborative Filtering)이라고도 부르며 다른 사람들의 관심 사항을 예측하기 위해 동료의 의견을 사용하는 기법으로, 제록스 팔로알토 연구소(PARC)의 Nichols등에 의해서 개발된 Tapestry가 협력 필터링을 적용한 최초의 시스템이다[15, 16]. 그리고 미네소타 대학의 GroupLens는 현재로서 가장 유명한 협력 필터링 시스템 형태이다[17]. 협력 필터링 시스템은 같은 취향이나 취미를 가진 사람들의 정보를 이용해 추천을 할 때 도움을 줄 수 있는 가장 널리 사용되는 시스템이다. 랭고 시스템[14]은 음악 앨범 추천용으로, 무비렌즈 시스템[17]은 영화 추천용으로, 제터 시스템[18]은 조크 추천용으로, 그리고 플라이캐스팅[19]은 온라인 라디오 추천용으로 개발된 협력 필터링 시스템들이다. 최근에는 인터넷 서점인 Amazon이나 인터넷 CD 상점인 CDNow, 그리고 인터넷 영화 추천 사이트인 MovieFinder 등에서 협력 필터링 방법을 적용시켜 성공을 거두고 있다. 이처럼 협력 필터링은 정보 필터링 응용분야나 상거래 응용분야에서 연구나 실용적 측면 모두에서 성공을 거두고 있으나 다음과 같은 근본적으로 해결해야 할 몇 가지 문제점을 갖고 있다[13,20].

- 초기 평가 문제(Early rate problem) : 순수한 협동여과는 새로운 항목이 등장하여 예측에 기반이 되는 사용자들의 평가들이 없을 경우, 이 항목에 대한 예측을 제공할 수 없다. 더군다나, 특정 항목에 대한 초기 예측들은 종종 부정확하다. 왜냐하면 예측에 사용되는 평가들이 매우 적기 때문이다[21]. 이와 유사하게, 심지어 이미 설치되어 운영되는 시스템조차도 시스템에 가입하는 모든 새로운 사용자들에게 불충분한 예측들을 제공할 것이다. 초기 평가 문제의 극단적인 예로서, 협력 필터링 시스템이 처음 시작될 때 모든 사용자는 모든 항목에 대해서 초기 평가 문제로부터 영향을 받는다.

- 희소성 문제(Sparsity problem) : 많은 정보 도메인에서는, 항목들의 개수는 개별 사용자들이 소화할 수 있는 개수를 훨씬 초과한다. 따라서, 모든 사용자들에 대한 모든 항목들의 평가들을 포함하고 있는 행렬들은 매우 드문드문한 분포성을 띤다. 이는 사용자간 유사도 계산 시 부정확성을 초래하며 결과적으로 협력 예측의 질을 떨어뜨리게 된다.

- 확장성(Scalability) : 협력 필터링 분야에서 주로 사용하는 최근접 이웃 알고리즘(Nearest Neighbor Algorithm)은 사용자와 항목 수에 비례해서 계산 시간이 비례한다. 따라서, 사용자 수와 항목 수가 수백만이나 되는 환경 하에서는 이러한 계산 시간이 치명적일 수 있다.

추천 질을 향상시키기 위해 최근접 이웃 알고리즘에 기반한 방법[10,13]들과 Bayesian networks[8], clustering[8], 그리고 Horting[7] 등의 기법을 이용한 방법 등 다양한 방법들이 제안되었다. 그러나 이러한 대부분의 방법들은 순수 협력 필터링 방법, 즉 사용자-항목 평가 정보만을 이용하는 방법으로 위에서 언급한 문제 중 초기 평가 문제와 희소성 문제를 해결하는 데는 한계가 있다. 반면에 내용기반 필터링은 이러한 문제가 발생하지 않는다. 따라서, 좀 더 나은 성능을 얻기 위해서 내용기반 방법과 협력 필터링 방법을 결합하여 각각의 장점을 취하는 접근방법은 아주 자연스러운 방법이라 할 수 있다. 이러한 측면에서 사용자 프로파일 정보를 협력 필터링 방법에서 사용하는 몇몇 연구들[20,22-25]이 있어 왔다. 하지만, 이러한 접근 방법들은 사용자 프로파일 정보를 효과적으로 이용하는 데는 한계가 있다. 따라서, 본 논문에서는 협력필터링 틀 안에서 사용자 프로파일 정보를 효과적으로 이용하는 방법을 제안하였다. 그리고, 본 제안 방법에 대해 여러 가지 요소들을 변화시키면서 특성을 분석하였고 실험을 통해 기존의 협력 필터링 방법과의 비교, 분석도 하였다.

2장에서는 사용자 프로파일 정보를 고려한 새로운 형태의 협력필터링 방법에 대해서 기술하고, 3장에서는 제안 방법의 유용성을 보이기 위한 실험결과를 기술한다. 4장에서는 기존 관련 연구와의 비교를 그리고 5장에서는 결론과 향후 연구에 대해서 언급한다.

2. 새로운 협력필터링 방법

본 논문에서 제안한 방법은 사용자간 유사도를 계산하기 위해 사용자 프로파일 정보와 항목 평가 정보를 같이 이용한다(그림 1 참조). 제안 방법은 다음과 같다.

- ① 사용자 프로파일을 클러스터링하고 이를 이용하여 그룹 평가 행렬을 생성한다.

- ② 사용자간 유사도를 계산한다. 기존 협력 필터링에

서 사용하던 사용자-항목 평가 행렬과 1단계에서 생성된 사용자-그룹 평가 행렬을 이용하여 사용자간 유사도를 구한다.

③ 이웃 사용자의 평균으로부터 가중치 편차 평균을 계산하여 항목에 대한 예측을 한다.

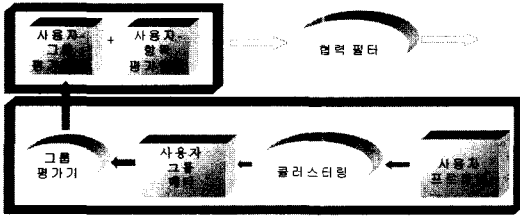


그림 1 사용자 프로파일을 고려한 협력 필터링 방법의 개념도

위 개념도에서 보듯이 사용자 관심사 정보에 클러스터링을 적용하여 그룹 벡터들을 생성하고, 이 그룹 벡터들과 사용자 프로파일 벡터를 입력으로 하여 그룹 평가기가 사용자-그룹 평가 정보를 생성한다. 그리고 이렇게 생성된 정보와 기존 협력필터링에서 사용하던 사용자-항목 평가정보가 협력 예측을 위한 자료로 쓰인다. 즉, 기존 협력 필터링 틀을 그대로 유지하면서 내용기반 필터링에서 사용하던 사용자 프로파일 정보를 수용하고 있다. 그림 2에서 보는 바와 같이 프로파일 정보에 따른 사용자 그룹을 일종의 항목으로 취급함으로써 사용자 유사도 계산 시 프로파일 정보가 일부 반영되어 추천 질을 향상시킬 수 있으며, 특히, 새로운 사용자인 경우 사용자-항목 정보가 없더라도 사용자 프로파일에 기초한 그룹 평가 정보를 바탕으로 협력 예측을 할 수 있어 새로운 사용자 문제를 자연스럽게 해결할 수 있다.

	영화1	...	영화m	사용자그룹1	...	사용자그룹k
사용자1
...
사용자n

사용자-항목 행렬 사용자-그룹 행렬
그림 2 확장된 평가 행렬

2.1 그룹 평가(Group Rating)

사용자 프로파일에 클러스터링 기법을 적용하는 주목적은 사용자들을 비슷한 취향의 그룹으로 묶어 이를 협력 예측 시 사용자간 유사도 계산 시에 반영하기 위해서이다. 이러한 그룹 정보는 사용자-항목 정보만을 기반으로 사용자 간 유사도를 판단하던 기존 협력필터링에

프로파일 정보를 추가로 고려하는 기회를 제공한다. 사용자 프로파일은 사용자가 직접 입력할 수도 있고 아니면 시스템이 사용자 행위 분석 등의 방법을 사용하여 간접적으로 구성할 수도 있다. 본 논문에서는 한 사용자에게 대한 프로파일을 여러 개의 관심집합으로 구성하며 하나의 관심집합은 속성들과 속성에 대한 중요도로 표현한다.

$$\{Aspect_1 : \{a_{1,1}/w_{1,1}, \dots, a_{1,n_1}/w_{1,n_1}\}, \dots,$$

$$Aspect_k : \{a_{k,1}/w_{k,1}, \dots, a_{k,n_k}/w_{k,n_k}\}$$

여기서, Aspect_i는 i번째 관심사항을, a_{i,j}와 w_{i,j}는 i번째 관심사항의 속성과 그의 가중치를 나타낸다. 예를 들어, 영화 추천 시스템인 경우, 사용자는 영화를 여배우나 남배우에 따라 추천을 받고 싶어하는 경우도 있고, 장르에 따라 혹은 줄거리의 내용에 따라 추천 받고 싶어하는 경우도 있다. 이러한 경우에는 여배우, 남배우, 장르, 줄거리 각각이 관심집합으로 표시된다. 그리고, 예를 들어 시스템이 3개의 장르(사랑, 코메디, 액션)와 2명의 여배우(장미희, 홍금보)만 제공한다면 어느 한 사용자의 프로파일은 아래처럼 표현된다.

$$\{\text{장르} : \{\text{사랑}/0.7, \text{코메디}/0.9, \text{액션}/0.3\},$$

$$\text{여배우} : \{\text{장미희}/0.5, \text{홍금보}/1.0\}$$

본 제안 방법의 유용성을 보이기 위해 사용한 MovieLens 데이터에는 사용자 프로파일 정보가 명시적으로 주어지지 않는다. 단지, 영화의 장르와 그 영화에 대한 사용자의 평가치만 주어진다. 따라서, 본 실험에서는 이 정보만을 이용하여 사용자 프로파일을 구성하였다. 즉, 사용자가 평가한 영화의 장르와 그의 가중치로 사용자의 프로파일을 구성하였다. 특정 사용자가 해당 장르를 좋아할 가중치를 계산하기 위해 아래와 같은 간단한 방법을 사용하였다.

$$w_m = \frac{Num_{\text{item} \in \text{attribute}_m | \text{item} > \text{threshold}}}{Num_{\text{item} > \text{threshold}}}$$

여기서, w_m은 m번째 장르에 대한 가중치를, Num_{item > threshold}는 해당 사용자가 평가한 영화 중에서 평가치가 threshold 보다 큰 영화의 개수를, Num_{item ∈ attribute_m | item > threshold}는 해당 사용자가 평가한 영화 중에서 평가치가 threshold 보다 크고 장르가 m번째 장르인 영화의 개수를 의미한다. MovieLens 데이터의 평가치는 1과 5사이의 정수값으로 1과 2는 부정적 평가를 나타내기 때문에 본 논문에서는 threshold를 3으로 하였다. 예를 들어, 길동이가 “뽕”, “친구”, “보디가드”에 대해 평가를 하였고 그 평가치가 모두 3보다 크거나 같다고 하자. 그리고, 각 영화의 장르가 사랑, 액션, 액션이라면 길동이의 프로파일은 {장르 : {사랑

입력 : 클러스터 수 k 와 사용자 프로파일 집합

출력 : k 개의 클러스터와 각 사용자가 해당 클러스터에 속할 정도

- (1) 각 사용자가 속할 클러스터를 무작위로 할당
- (2) 클러스터에 더 이상 변화가 없을 때까지 다음의 과정을 반복
 - (a) 각 클러스터의 중심을 구함.
 - (b) 각 클러스터의 중심과 사용자 프로파일 간의 거리를 계산해서 가장 가까운 클러스터에 사용자 재 할당
- (3) 아래의 식을 이용하여 각 사용자가 각 클러스터에 속할 정도 계산.

$$P = 1 - \frac{E(j, m)}{\text{Max}_i E(i, m)}$$

여기서, j 는 j 번째 사용자 프로파일 벡터를, m 은 (2) 단계에서 구해진 m 번째 클러스터의 중심벡터를, $E(j, m)$ 은 j 와 m 간의 유클리디안 거리를, $\text{Max}_i E(i, m)$ 은 모든 사용자의 프로파일 벡터와 m 번째 클러스터와의 거리 중 최대 값을 의미한다

그림 3 그룹 평가 알고리즘

/0.333, 액션/0.666))로 표현된다.

본 논문에서는 사용자 프로파일의 내용을 그림3과 같은 변형된 K-means 알고리즘을 적용하여 사용자 그룹을 형성하는 방법을 선택하였다. 단계 1과 2는 K-means 알고리즘을 적용하여 클러스터를 생성하는 단계이며 단계 3은 구해진 클러스터의 중심들과 각 사용자의 프로파일 벡터 간의 유사도를 구하는 과정이다. 보통 클러스터링의 결과는 해당 객체가 속할 그룹만 결정되지만, 단계 3에서는 해당 객체가 각 그룹에 속할 가능성이 계산되어 지고 이를 행렬 형태로 표현할 수 있다. 이 행렬을 사용자-그룹 평가 행렬이라 한다.

2.2 유사도 계산

협력 필터링 방법을 사용하여 특정 항목에 대한 특정 사용자의 평가 정도를 예측하기 위해서는 위에서 얻어진 행렬과 원래 주어진 사용자-항목 평가 행렬을 사용하여 사용자간의 유사도를 계산할 필요가 있다. 여기서는, 사용자-항목 평가 행렬을 이용하여 사용자간 유사도를 구하는 기존의 대표적인 방법 2가지를 간단히 소개하고, 본 논문에서 사용자-그룹 평가 행렬을 추가로 고려한 사용자간 평가방법을 소개한다.

2.2.1 피어슨 상관관계 공식을 이용한 유사도[17]

가장 많이 사용되는 가중치 측정은 피어슨 상관관계(Pearson correlation) 방법이다. 피어슨 상관관계 방법은 두 변수간에 존재하는 선형 관계의 정도를 측정한다. 피어슨 상관계수는 선형 리그레션 모델에서 유도되며 상관관계는 선형적이며 에러는 서로 독립적이고 평균 0인 확률 분포와 독립 변수의 모든 설정에 대한 상수 편차를 가진다는 가정에 기반을 둔다[26].

$$\text{sim}(k, l) = \frac{\text{cov}(k, l)}{\sigma_k \cdot \sigma_l} = \frac{\sum_{i=1}^m (k_i - \bar{k})(l_i - \bar{l})}{\sqrt{\sum_{i=1}^m (k_i - \bar{k})^2} \sqrt{\sum_{i=1}^m (l_i - \bar{l})^2}} \quad (1)$$

여기서, m 은 항목 수를, k_i 는 사용자 k 의 i 번째 항목에 대한 평가치를, l_i 는 사용자 l 의 i 번째 항목에 대한 평가치를, \bar{k} 와 \bar{l} 은 사용자 k 와 l 의 m 개의 항목에 대한 평가치의 평균을 나타낸다.

2.2.2 보완 코사인 유사도(Adjusted Cosine Similarity)[13]

코사인 유사도는 유사도를 계산하기 위해 한 때 가장 많이 사용된 기법이지만 단점을 가지고 있다. 서로 다른 사용자들 사이에서 평가 스케일(척도)의 차이는 아주 많이 다른 유사도를 초래한다는 것이다. 예를 들어, 길동이 가장 선호하는 영화를 4라고 평가를 생각했다면 5라는 평가는 하지 않을 것이다. 나쁜 영화에 대한 평균 평가가 2인데도 1이라고 평가할 수 있다. 그러나, 순이는 가장 좋은 영화는 5, 나쁜 영화는 2라고 평가한다. 기존의 코사인 유사도를 사용한다면 2명에 대한 유사도는 아주 다를 것이다. 보완된 코사인 유사도는 이러한 단점을 보완할 수 있다.

$$\text{sim}(k, l) = \frac{\sum_{i=1}^m (k_i - \bar{r}_i)(l_i - \bar{r}_i)}{\sqrt{\sum_{i=1}^m (k_i - \bar{r}_i)^2} \sqrt{\sum_{i=1}^m (l_i - \bar{r}_i)^2}} \quad (2)$$

여기서, m 은 항목 수를, k_i 는 사용자 k 의 i 번째 항목에 대한 평가치를, l_i 는 사용자 l 의 i 번째 항목에 대한 평가치를, \bar{r}_i 는 항목 i 에 대한 모든 사용자의 평가치 평균을 나타낸다.

2.2.3 사용자-그룹 평가 행렬을 고려한 사용자간 유사

도 계산 방법

보통, 사용자-항목 평가 행렬 정보는 이산적인 값 (예, MovieLens 데이터인 경우 평가치는 1과 5 사이의 정수)을 가지며 사용자-그룹 평가 행렬은 0과 1 사이의 연속적인 값을 갖는다. 따라서, 한쪽의 값을 다른쪽의 값으로 확대 또는 축소하여 동일한 유사도 척도를 적용하는 방법도 고려할 수 있으며 각각의 행렬에 대해 다른 유사도 척도를 적용시키고 이들의 결과를 조합하여 사용하는 방법도 고려해 볼 수 있다. 본 논문에서는 아래의 5가지 방법에 대해서 그 성능 평가를 수행하였다.

- Non-enlarged Pearson : 사용자-항목 평가 행렬과 사용자-그룹 평가 행렬을 하나의 행렬로 취급한 후 기존의 Pearson 상관 공식을 적용
- Enlarged Pearson : 사용자-그룹 평가 행렬을 사용자-항목 평가 값의 범위로 확대하고 두 개의 행렬을 하나의 행렬로 취급한 후 기존의 피어슨 상관관계 공식을 적용
- Non-enlarged Adjusted Cosine : 사용자-항목 평가 행렬과 사용자-그룹 평가 행렬을 하나의 행렬로 취급한 후 기존의 보완 코사인 공식을 적용
- Enlarged Adjusted Cosine : 사용자-그룹 평가를 사용자-항목 평가 값의 범위로 확대하고 두 개의 행렬을 하나의 행렬로 취급한 후 기존의 보완 코사인 공식을 적용
- Combination Approach : 사용자-항목 평가 행렬에는 피어슨 상관관계 공식을, 사용자-그룹 평가 행렬에는 보완 코사인 유사도를 적용시킨 후 아래의 수식에 따라 결합

$$\text{sim}(k, u) = \text{sim}(k, u)_{\text{item}} \times (1 - c) + \text{sim}(k, u)_{\text{group}} \times c$$

여기서, $\text{sim}(k, u)_{\text{item}}$ 은 사용자-항목 평가 행렬에 피어슨 상관관계 공식을 적용시켜 얻은 사용자 k와 이웃 u 사이의 유사도를, $\text{sim}(k, u)_{\text{group}}$ 은 사용자-그룹 평가 행렬에 보완 코사인 공식을 적용시켜 얻은 사용자 k와 이웃 u 사이의 유사도를, c는 이들 사이의 결합 계수를 의미한다.

2.3 협력 예측

사용자 K의 항목 i에 대한 예측을 구하기 위해 GroupLens에서 제안한 식 [17]을 사용하였다. 여기서는, 항목에 대한 예측은 이웃의 평균값으로부터 편차의 가중치 평균을 수행함으로써 계산된다. 그리고 사용자의 유사성에 기반한 가장 인접한 N 개의 이웃을 선택하기 위하여 top N 규칙을 사용한다.

$$P_{k,i} = \bar{k} + \frac{\sum_{u=1}^N (k_i - \bar{u}) \times \text{sim}(k, u)}{\sum_{u=1}^N |\text{sim}(k, u)|} \quad (3)$$

여기서, $P_{k,i}$ 는 항목 i에 대한 사용자 k의 예측을 표시한다. n은 사용자 k의 최근접 이웃의 수, k_i 는 항목 i에 대한 k의 평가, \bar{k} 는 항목에 대한 사용자 k의 평균 평가, sim은 사용자 k와 이웃 u 사이의 유사도, \bar{u} 는 항목에 대한 사용자 u의 평균 평가를 의미한다.

전통적인 협력 필터링 방법은 새로운 사용자에 대해서는 평가 정보가 없기 때문에 추천하기가 곤란하다. 반면에, 본 방법에서는 사용자 프로파일에 기초한 그룹 정보가 있기 때문에 추천이 가능하다. 하지만, 항목에 대한 아무런 평가 정보가 없어 \bar{k} 가 0이 되기 때문에 위의 수식을 그대로 사용하기는 곤란하다. 따라서, 본 방법에서는 새로운 사용자인 경우는 \bar{k} 대신에 $\overline{k_{\text{neighbor}}}$ 를 사용한다. $\overline{k_{\text{neighbor}}}$ 는 새로운 사용자와 최근접한 이웃들의 항목에 대한 평균 평가치를 의미한다.

2.4 추천 예

이해를 돕기 위해서 다음과 같은 예를 사용하기로 하자.

[사용자]

사용자의 수: 5

사용자의 이름: 길동, 호돌, 호순, 넘치, 박이

[항목]

항목 분야: 영화

항목의 수: 3

항목의 제목: 친구, 별들의 전쟁, 박하사탕

[평가]

평가정보는 표 1과 같다고 가정. 여기서, 평가치는 1~5 사이의 정수이며 그 의미는 아래와 같음.

- 1: 매우 나쁨 2: 나쁨 3: 보통
- 4: 좋음 5: 매우 좋음

표 1 사용자-항목 평가표

항목 \ 사용자	길동	호돌	호순	넘치	박이
친구	5	5	2	4	
별들의 전쟁	3	2	5	2	
박하사탕		4			

Step 1.

사용자들을 그룹핑하기 위하여 사용자 프로파일을 기반으로 하여 클러스터링 알고리즘을 적용하여야 하나 여기서는 단순히 클러스터링 결과가 표 2와 같다고 가정한다. 그룹1: {길동 (98%), 호돌 (100%), 호순 (1.0%), 넘치(95%), 박이 (0.12%)}, 그룹2: {길동 (0.13%), 호돌 (0.02%), 호순 (95%), 넘치 (1.2%), 박이 (98%)}. 사용자 이름 옆 괄호 안의 숫자는 사용자가 클

러스터에 속할 정도를 뜻한다.

표 2 사용자-그룹 평가표

사용자 그룹	길동	호돌	호순	넙치	박이
그룹1	98%	100%	1.0%	95%	0.12%
그룹2	0.13%	0.02%	95%	1.2%	98%

Step 2.

그룹 평가 행렬과 항목 평가 행렬을 결합하여 표 3와 같은 새로운 평가 행렬을 만든다.

표 3 최종 평가표

사용자 항목	길동	호돌	호순	넙치	박이
친구	5	5	2	4	
별들의 전쟁	3	2	5	2	
박하사탕		4			
그룹1	0.98	1.0	0.01	0.95	0.0012
그룹2	0.0013	0.0002	0.95	0.012	0.98

Step 3.

이 새로운 통합된 평가 데이터 행렬을 기반으로 2.2.3에 기술된 방법중의 하나를 사용하여 사용자들간의 유사도를 계산할 수 있다. 여기서는, 길동이와 호돌이 간의 유사도를 다섯 번째 방법, 즉 선형결합 방법을 사용하여 구하는 예를 보인다. 먼저 피어슨 상관관계 기반 알고리즘을 사용하여 사용자들간의 유사도 ($\text{sim}(G, H)_{\text{item}}$)를 계산한다. 그 다음, 보완된 코사인 알고리즘을 사용하여 사용자 클러스터링 결과를 토대로 사용자간의 유사도 ($\text{sim}(G, H)_{\text{group}}$)를 계산한다. 여기서, G는 길동이를, H는 호돌이를 의미한다.

$$\text{sim}(G, H)_{\text{item}} = \frac{(5-4) \times (5-3.5) + (3-4) \times (2-3.5)}{\sqrt{(5-4)^2 + (3-4)^2} \times \sqrt{(5-3.5)^2 + (3.5-2)^2}} = 1$$

$$\text{sim}(G, H)_{\text{group}} = \frac{(0.98-0.59) \times (1-0.59) + (0.0013-0.39) \times (0.0012-0.39)}{\sqrt{(0.98-0.59)^2 + (0.0013-0.39)^2} \times \sqrt{(1-0.59)^2 + (0.0012-0.39)^2}} = 0.999722$$

최종 사용자간의 유사도는 이 두 결과의 합을 가중치 평균한 것이다. 이 때, 결합 계수는 0.4를 사용한다고 하자.

$$\text{sim}(G, H) = 1 \times (1-0.4) + 0.999722 \times 0.4 = 0.999889$$

표 4는 각 사용자 간 최종 유사도를 나타낸다.

표 4 사용자간 최종 유사도

	길동	호돌	호순	넙치	박이
길동	1.0	0.999889	-0.999889	0.999864	-0.999996
호돌	0.999889	1.0	-1.0	0.999507	-0.999655
호순	0.999889	-1.0	1.0	-0.999648	0.999828
넙치	0.999864	0.999507	-0.999648	1.0	-0.999727
박이	-0.999996	-0.999655	0.999828	-0.999727	1.0

Step 4.

2.3절의 공식을 사용하여 각 사용자의 항목에 대한 예측치를 계산한다. 표 5는 그 결과를 나타낸다.

표 5 선호도 예측치

사용자 항목	길동	호돌	호순	넙치	박이
친구	5	5	2	4	0.5
별들의 전쟁	3	2	5	2	3.5
박하사탕	4.5	4	3	3.5	2.5

3. 협력필터링 평가

3.1 평가 환경

[평가 데이터 집합]

현재 웹에 기반을 둔 추천 시스템인 MovieLens [27]에서 수집된 영화 평가에 대한 데이터를 이용하였다. 데이터 집합은 943명의 사용자, 1682개의 영화를 각 사용자가 적어도 20개의 항목에 대해 평가를 한 100,000개의 평가를 포함하고 있다. MovieLens의 평가는 사용자들이 1에서 5사이의 정수값으로 직접 평가 자료를 입력하였다. 이 데이터 집합은 훈련 집합과 테스트 데이터 집합으로 구성되어 있다.

[평가 척도]

MAE (Mean Absolute Error)는 테스트 자료에서 실제 사용자 평가에 대하여 예측치를 비교함으로써 추천 시스템의 정확도를 평가하는데 가장 많이 사용되고 있다. 본 논문에서도 이 척도를 사용하였다. 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.

$$MAE = \frac{\sum_{i=1}^n |p_i - q_i|}{n} \quad (4)$$

여기서, n은 평가 대상의 수를, p_i 는 대상 i에 대한 예측치를, q_i 는 대상 i에 대한 실 평가치를 나타낸다.

3.2 성능 비교

3.2.1 패러미터에 따른 자체 성능 비교

[결합 계수에 따른 성능 비교]

앞에서 설명한 사용자간 유사도 계산 방법 중 다섯 번째 방법, 즉 결합 방법은 결합 계수에 따라 성능이 달라진다. 그림 4는 그 결과를 보여준다. 그림에서 보는 바와 같이 결합 계수가 0.4인 경우가 가장 우수함을 알 수 있다. 앞으로의 실험에서는 이 계수를 사용하였다.

[클러스터 수에 따른 성능 비교]

그림 5에서 보는 바와 같이 클러스터 수가 20일 때가

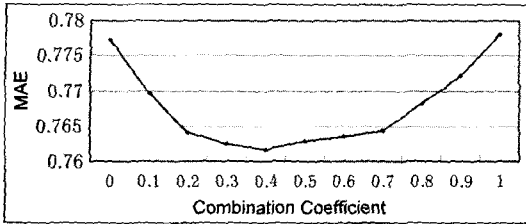


그림 4 결합 계수에 따른 성능

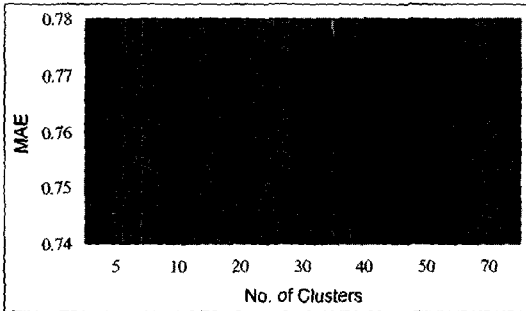


그림 5 클러스터 수에 따른 성능

지는 클러스터 수가 증가하면 성능도 같이 좋아짐을 알 수 있다. 클러스터 수가 20이상인 경우는 거의 변화가 없음을 알 수 있다. 앞으로의 실험에서는 클러스터의 수를 20으로 고정하여 실험하였다.

[사용자 프로파일 그룹핑 방법에 따른 성능 비교]

사용자 프로파일 그룹핑 시 클러스터 중심과 사용자 프로파일 벡터 사이의 유사도 계산이 필요한데, 본 실험에서는 두가지 방법 - 유클리디안 거리와 코사인 유사도 -의 성능을 비교해 보았다. 그림 6에서 보는 바와 같이 코사인 유사도가 성능이 우수함을 알 수 있다. 하지만, 그 차이가 미미하다.

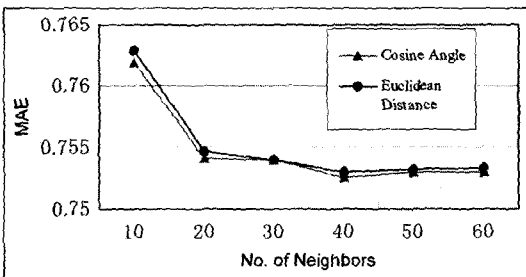


그림 6 프로파일 그룹핑 방법에 따른 성능

[사용자간 유사도 계산 방법에 따른 성능 비교]

본 논문에서는 2.2.3 절에서 제안된 5가지 경우를 구현하여 기존의 순수 협력 필터링 방법들, 즉, 사용자-항목 평가 행렬만을 이용하고 사용자간 유사도는 피어슨

상관관계 공식을 이용하는 방법(Classic Pearson)과 보완 코사인 공식을 이용한 방법(Classic Adjusted Cosine)과 비교하였다. 피어슨 상관관계 공식을 사용한 경우와 보완 코사인 공식을 사용한 경우의 결과가 비슷하여 그 결과를 그림 7, 8로 나누어 표시하였다.

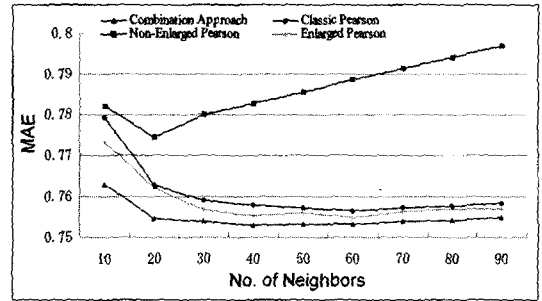


그림 7 사용자간 유사도 계산 방법에 따른 성능 (피어슨 상관관계 공식)

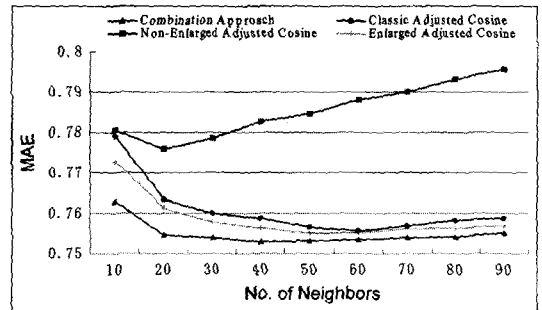


그림 8 사용자간 유사도 계산 방법에 따른 성능 (보완 코사인 공식)

그림 7, 8에서 보는 바와 같이 본 제안방법이 기존의 순수 협력 필터링 방법보다 좋음을 알 수 있다. 이는 비록 실험에서 사용자 프로파일을 사용자-항목 정보를 바탕으로 구성하였지만 장르라는 내용 정보가 추가로 고려되어 보다 정확하게 사용자 간 유사도가 계산되었기 때문이라 생각된다. 또한, 사용자-그룹 평가의 값을 확대시킬 경우가 그렇지 않은 경우보다 성능이 월등함을 알 수 있다. 더욱이, 사용자-그룹 평가의 값을 확대시키지 않을 경우는 사용자-그룹 평가 정보를 사용하지 않는 경우, 즉 순수 협력 필터링 보다 성능이 나쁨을 알 수 있다. 이는 그룹 평가정보가 제대로 반영이 되지 않고 오히려 부정적으로 작용하는 것으로 보인다. 또한, 두 개의 행렬에 다른 유사도 척도를 적용한 경우, 즉 결합 방법이 가장 성능이 우수함을 알 수 있다. 이는 두 개의 행렬이 이질적인 데 기인한 것으로 보인다.

[근접 이웃 수에 따른 성능 비교]

그림 7, 8에서 보는 바와 같이 이웃 수에 따라 성능이 많이 좌우됨을 알 수 있다. 사용기간 유사도 계산 시 다섯 번째 방법을 사용하였을 경우, 이웃 수가 10~30 개까지는 이웃 수가 증가하면 증가할수록 성능이 향상됨을 볼 수 있고 40 이상인 경우는 성능 향상이 거의 없거나 오히려 미미하게 떨어지는 것을 알 수 있다. 다른 유사도 계산 방법을 사용하는 경우에도 이웃 수가 일정 임계치를 넘어서면 성능 향상이 없거나 떨어짐을 관측할 수 있다. 이는 특정 임계치 이상의 사용자들의 성향과 해당 사용자의 성향이 그리 유사하지 않는데도 그들의 의견이 반영되어서 부정적 효과를 불러온 것으로 보인다.

3.2.2 새로운 사용자에 대한 성능

MovieLens 데이터 집합은 943명의 사용자가 평가한 정보로 구성되어 있으며 학습 데이터와 테스트용 데이터로 구성되어 있다. 본 실험에서는 새로운 사용자에 대한 성능을 평가하기 위하여 학습용 데이터에 있는 몇몇 사용자의 평가 정보를 모두 제거시키고 이들을 새로운 사용자로 취급하는 방법을 택했다. 새로운 사용자에 대해 항목을 추천하는 방법으로 2.3절에 기술한 방법 외에 아래와 같은 Sarwar가 제안한 방법 [13]을 추가로 실험하였다. 앞으로 2.3절의 방법을 average method라 하고 Sarwar의 방법을 weighted sum method라 한다.

$$P_{k,i} = \frac{\sum_{u=1}^n r_{u,i} \times \text{sim}(k, u)}{\sum_{u=1}^n |\text{sim}(k, u)|} \quad (5)$$

여기서, $P_{k,i}$ 는 항목 i 에 대한 사용자 k 의 평가 예측치를, n 은 사용자 k 의 근접 이웃들의 수를, $r_{u,i}$ 는 사용자 u 의 항목 i 에 대한 평가치를, $\text{sim}(k, u)$ 는 사용자 k 와 u 사이의 유사도를 의미한다.

먼저, 무작위로 선택된 한 사용자(No. 73)를 새로운 사용자로 취급하여 그 예측 결과를 살펴 보았다. 학습 데이터에는 No. 73 사용자가 평가한 항목이 34개가 있고 테스트 데이터에는 32개의 평가 정보가 있다. 학습 데이터에 있는 평가 정보를 제거시킨 후 테스트 데이터에 있는 항목들에 대해 두가지 방법을 사용하여 예측하여 보았다. 그 결과는 그림 9와 같다. 여기서, Real Rating은 테스트 데이터에 있는 사용자 No. 73의 평가치를 나타내며, Real Predict는 학습 데이터에 있던 No. 73의 평가 정보를 제거시키지 않고 기존 사용자 평가 예측 방법을 사용할 경우의 예측치를 나타낸다. 분석 결과 새로운 사용자에 대한 평가 예측치가 실제 예측치와 패턴이 유사함을 알 수 있었다. 일례로, 항목 a, b에 대한 실 평가치가 1과 5라면, 그에 대한 예측치는 2와 4가 된다. 즉, 절대적 예측치는 다르지만 상대적인 예측치는 거의 유사하였다.

average method와 weighted sum method의 비교를 위해서 1에서부터 50 사용자까지 10 단위로 변화시키면서 추가의 실험을 하였다. 즉, 1, 10, 20, 30, 40, 50 명을 새로운 사용자로 취급해서 그 때의 성능을 각각 평가하여 보았다. 표 6에서 보는 바와 같이 average method가 weighted sum method 보다 성능이 우수함을 알 수 있었다.

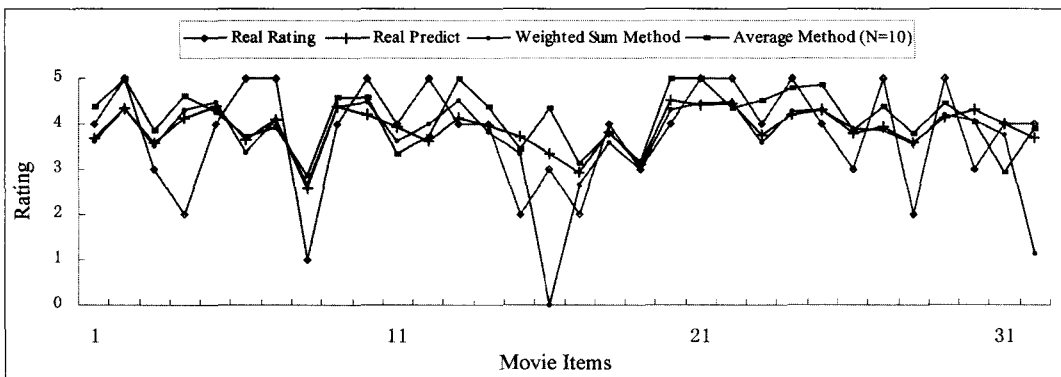


그림 9 예측 방법에 따른 한 새로운 사용자에 대한 예측 성능

표 6 예측 방법에 따른 여러 새로운 사용자에 대한 예측 성능

	1	10	20	30	40	50
MAE of average method	0.819	0.677	0.756	0.786	0.745	0.695
MAE of weighted sum method	0.878	0.716	0.812	0.855	0.824	0.744

4. 관련연구와의 비교

국내외에서는 협력 필터링과 관련된 다양한 연구들 [7,8,10-12,21,28,29]이 진행되어 왔다. 이와 관련된 연구들이 너무 많기 때문에 여기서는 본 연구와 직접 관련된 내용기반 필터링과 협력필터링을 결합하는 연구에 대해서만 언급한다.

GroupLens는 Filterbot이라는 가상의 사용자들을 이용하여 내용기반 필터링을 지원하는 유즈넷 뉴스를 위한 협력 필터링 시스템을 구현하였다[24]. 이들 Filterbot들은 새로운 기사가 나오자마자 평가를 하고 이들 문서들에 대한 평가 점수를 부여한다. 협력 필터링 시스템은 Filterbot을 많은 평가들을 입력하는 또 다른 평범한 사용자로 여긴다. Filterbot 작성자는 새로운 기사가 도착할 때마다 평가 점수를 시스템에 돌려주는 내용기반 에이전트처럼 Filterbot을 작성한다. GroupLens의 Filterbot은 희소성 문제를 해결하는데 도움을 준다. 왜냐하면 많은 수의 기사들을 빨리 평가할 수 있기 때문이다. 그러나 GroupLens 예측들은 아직까지 순수한 협력 필터링을 사용하고 있기 때문에 초기 평가 문제로부터 악영향을 받는다. 모순되게도, 초기 구동시, Filterbot과 그 작성자 사이의 상관관계가 좋을 것 같음에도 불구하고 이들 사이에 좋은 상관관계를 가지지 않는다. 이 방법과 본 제안 방법의 차이점을 살펴보면, GroupLens 방법은 Filterbot이라는 가상의 사용자들 통하여 항목의 내용을 반영하고 있으나, 기존의 순수 협력 필터링 방법을 사용하여 사용자-항목 간 평가를 예측하기 때문에, 사용자들간의 유사도 계산 시 사용자 프로파일 기반의 사용자 관심도가 고려되지 않는다. 반면에 본 제안 방법에서는 사용자간 유사도 계산 시 사용자-항목 간 평가 정보와 프로파일 기반 사용자간 유사도가 같이 고려된다. 이러한 특성 때문에 Filterbot 방법보다 추천 질이 좋아질 것으로 예상된다.

Fab[22]는 웹 페이지를 추천하는 이종의 내용기반 협력 시스템을 구현하였다. Fab에서는, 사용자가 선호하는 페이지들에 기반한 사용자 프로파일들이 내용기반 기법들을 통하여 유지되어지고, 이러한 프로파일들은 협력 필터링에서 예측을 위한 목적으로 사용자들간의 유사도를 결정하기 위해서 직접적으로 비교되어진다. 효과성을 높이기 위해서, Fab 접근 방법은 내용기반 기법들이 매우 정확한 사용자 프로파일을 구축하도록 하였다. Fab 방법은 본 방법과 유사한 측면이 많다. 이 방법에서는 사용자 간 유사도 계산 시 본 논문에서 사용하는 프로파일 기반 사용자간 유사도를 고려하고 있다. 대신에 기존 협력 필터링 방법에서 사용하던 사용자-항목 간 평가 정보는 고려되고 있지 않다. 이로 인해 프로파

일 구성이 정확하지 않을 경우 추천 질이 많이 떨어지게 된다. 하지만 본 방법에서는 두 가지 정보를 모두 이용하기 때문에 프로파일 정보가 부정확하더라도 사용자 항목 간 평가 정보에 의해 보완되는 효과가 기대된다.

Basu, Hirsh 그리고 Cohen[23]은 사용자 영화 선호도들을 예측하고자 평가들과 인위적인 정보들 이용하는 귀납적 학습 접근 방법을 적용했다. Basu등이 제안한 방법은 기계학습 기법을 응용한다는 장점이 있으나, 기존 협력 필터링 방법보다 좋은 성능을 보이기 위해서는 추가의 특징을 만들어야 한다. 하지만, 이러한 특징을 만들어 내기가 쉽지 않다. 그러나, 본 방법은 기존 협력 필터링 방법의 뼈대를 그대로 유지하면서 추가의 정보를 이용하기 때문에 이러한 부담이 없으며 성능면에서도 기존 협력 필터링 방법보다 성능이 좋을 것으로 판단된다.

ProfBuilder[25]는 내용기반과 협력 필터링 둘 모두를 사용하여 웹 페이지를 추천하였다. 사용자들은 추천된 웹 사이트들에 대한 두개의 목록들에 대해서 단일한 인터페이스를 제공받는다. 한 목록은 협력 필터링에 의해 생성된 것이고 다른 목록은 내용기반 필터링에 의해 생성된 것이다. 어쨌든, 두 목록들은 결합된 예측들로 이루어진 하나의 목록으로 합쳐지지 않는다. 또한 사용자들이 스스로 양쪽 목록들로부터 가장 좋은 사이트들을 선택할 수 있도록 가능하게 하는, 각각의 예측에 대한 상대적인 중요성들도 제공되지 않는다. Claypool[20]은 ProfBuilder와 마찬가지로 내용기반 필터링과 협력 필터링 모두를 사용하여 온라인 신문을 필터링하는 방법을 제안하였다. 이 방법에서는 ProfBuilder와는 달리 각 필터링에서 독립적으로 얻은 결과를 하나의 결과로 통합해서 제공한다. ProfBuilder나 Claypool 방법은 내용기반 필터링 방법과 협력 필터링 방법을 모두 사용하기 때문에 각각의 장점을 살릴 수 있는 특징이 있다. 하지만 이 두 방법이 사용하는 정보가 각각 배타적이기 때문에 두가지 정보를 유기적으로 사용하는 본 방법과는 차이가 있다. 즉, 이 방법들은 두 협력 필터링 방법에서 내용정보를 이용하지 않고 내용기반 필터링에서는 사용자-항목 간 정보를 이용하지 않는다. 따라서, 이로 인해 좀 더 나은 예측 결과를 얻을 수 있는 기회를 상실하게 된다.

5. 결론

본 논문에서는 현재 추천 시스템에서 성공적으로 사용되는 협력필터링 방법의 문제점에 대한 부분적인 해결책을 제시하고자 하였다. 그 결과로 협력필터링 방법의 틀을 유지하면서 내용기반 필터링에서 사용자 관심사 정보를 유지하는 데 사용하는 사용자 프로파일을 효

과적으로 이용하는 방법을 제안하였다. 사용자 프로파일을 근거로 유사 사용자끼리 클러스터를 구축한 후 각 사용자가 각 클러스터에 속할 정도를 구하고 이 정보를 협력 필터링 방법에서 사용하는 방법을 선택하였다. 즉, 마치 사용자 그룹에 대한 평가 정보를 항목에 대한 평가 정보와 동일하게 다루는 방법을 택하였다. 실험을 통해 본 방법이 순수 협력 필터링 방법보다 좋은 성능을 보임을 객관적 데이터를 이용하여 보였다. 실험 데이터가 장르에 대한 정보만 제공하기 때문에 이의 정보만을 이용하여 그룹 평가 행렬을 구성하였다. 만약, 기타 다른 정보, 특히, 텍스트 형태의 내용 정보가 주어진다면 보다 나은 성능 향상을 기대할 수 있을 것이다.

기존 순수 협력 필터링 방법에서는 새로운 사용자에게서는 타 사용자가 평가한 평균치를 이용할 수 밖에 없다. 따라서, 모든 새로운 사용자는 본인의 관심사에 무관하게 비슷한 항목들을 추천 받게 된다. 하지만, 본 제안 방법을 사용할 경우는 사용자 프로파일에 대한 정보를 이용함으로써 새로운 사용자인 경우도 사용자의 관심사에 맞춰 추천 받을 수 있게 되어 초기부터 원하는 항목들을 추천 받을 가능성이 높아지게 된다. 이러한 장점에 비해, 본 방법은 추가로 사용자 프로파일을 유지해야 하는 부담이 있다. 즉, 사용자의 관심사가 정적이지만은 않기 때문에 사용자 관심사 변화에 맞추어 사용자 프로파일을 갱신할 필요가 있다. 이를 위해서는, 향후, 관심사 변경 시점을 자동으로 파악하는 방법과 그동안 축적된 정보를 이용하여 사용자 프로파일을 보다 정확하게 변경하는 방법에 대한 연구가 필요하다.

참 고 문 헌

- [1] Oard, D.W. and Marchionini, G., "A conceptual framework for text filtering," Technical Report EE-TR-96-25, CAR-TR-830, CLIS-TR-96-02, CS-TR3643, University of Maryland, College Park, 1996.
- [2] J. Alspector, A. Kolcz, and N. Karunanithi, "Comparing Feature-Based and Clique-Based User Models for Movie Selection," *Proc. of the 3rd ACM Conference on Digital Libraries*, 1998.
- [3] Pazzani, M., Billsus, D., "Learning and revising user profiles: the identification of interesting Web sites," *Machine Learning*, 1997.
- [4] Seo, Y., Zhang, B., "Personalized Web Document Filtering Using Reinforcement Learning," *Applied Artificial Intelligence*, 2001.
- [5] Curt Stevens, "Automating the creation of information filters," *CACM*, Vol.35, No.12, 1992.
- [6] Tak W.Yan and Hector Garcia-Molina, "Distributed selective dissemination of information," *Proc. of the Third International Conference on Parallel and Distributed Information Systems*, pp. 89-98, 1994.
- [7] C. C. Aggarwal, J. L. Wolf, K. Wu, and P. S. Yu, "Hortling Hatches an Egg : A New Graph-theoretic Approach to Collaborative Filtering," *Proc. of the ACM KDD'99*, 1999.
- [8] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," *Poc. of the 14th UAI*, 1998.
- [9] D. Goldberg, D. Nichols, B. Oki, and D. Terry, "Using Collaborative Filtering to Weave an Informtion Tapestry," *CACM*, Vol.35, No.12, pp. 61-70, 1992.
- [10] Herlocker, J., Konstan, J., Borchers, A., Riedl, J., "An Algorithmic Framework for Performing Collaborative Filtering," *Proc. of the 1999 Conference on Research and Development in Information Retrieval*, 1999.
- [11] W. Hill, L. Stead, M. Rosenstein, and G. Furnas, "Recommending and Evaluating Choices in a Virtual Community of Use," *Proc. of ACM CHI '95*, 1995.
- [12] O'Conner, M. and Herlocker, J., "Clustering items for collaborative filtering," *Proc. ACM-SIGIR Workshop on Recommender Systems*, 1999.
- [13] B. Sarwar, G. Karypis., J. Konstan, and J. Riedl, "Item-based Collaborative Filtering Recommendation Algorithms," *Proc. of WWW10*, 2001.
- [14] Upendra S. and Patti M., "Social Information Filtering: Algorithms for Automating "Word of Mouth", *Proc. of ACM CHI'95 Conference on Human Factors in Computing Systems*, pp. 210-217, 1995.
- [15] Donna Harman., "Overview of the third Text Retrieval Conference(TREC-3)," D. K. Harman, editor, *Overview of the Third Text Retrieval Conference (TREC-3)*, pp. 1-19, 1994.
- [16] Douglas B. Terry, "A tour through tapestry," *Proc. of the ACM Conference on Organizational Computing Systems(COOCs)*, pp. 21-30, 1993.
- [17] Resnick, P., Iacovou, N., Suchak, M., Bergstorm, P. and Riedl, J., "GroupLens: An open architecture for collaborative filtering of Netnews," *Proc. of ACM Conf. on Computer-Supported Cooperative Work*, pp.175-186, 1994.
- [18] D. Gupta, M. Digiovanni, H. Narita, and K. Goldberg, "Jester 2.0: A New Linear-Time Collaborative Filtering Algorithm Applied to Jokes," *Proc. of Workshop on Recommender Systems: Algorithms and Evaluation*, Aug. 1999.
- [19] Hauer, D. B. and French, J. C., "Flycasting: Using Collaborative Filtering to Generate a Playlist for Online Radio," *Proc. of Int. Conf. on Web Delivery of Music*, 2001.
- [20] M. Claypool, A. Gokhale, T. Mirana, P. Murnikov, D. Netes and M. Sartin, "Combing Content-Based

and Collaborative Filters in an Online Newspaper," *Proc. of Workshop on Recommender Systems - Implementation and Evaluation*, 1999.

- [21] A. Gokhale and M. Claypool, "Thresholds for More Accurate Collaborative Filtering," *Proc. of the IASTED International Conference on Artificial Intelligence and Soft Computing*, 1999.
- [22] M. Balabanovic and Y. Shoham, "Content-based collaborative recommendation," *CACM*, Vol.40, No.3, 1997.
- [23] C. Basu, H. Hirsh, and W. Cohen, "Recommendation as Classification : Using Social and Content-Based Information in recommendation," *Proc. of AAAI*, 1998.
- [24] B. Sarwar, J. Konstan, A. Borchers, J. Herlocker, B. Miller, and J. Riedl, "Using Filtering Agents to Improve Predictive Quality in the GroupLens Research Collaborative Filtering System," *Proc. of the ACM CSCW*, 1998.
- [25] Wasfi, A. M. A., "Collecting User Access Patterns for Building user Profiles and Collaborative Filtering," *Proc. of Int. Conf. on Intelligent User Interfaces*, pp. 57-64, 1999.
- [26] McClave, J. T. and Dietrich, F. H., *Statistics*, San Francisco:Dellen Publishing Company.
- [27] <http://www.cs.umn.edu/research/GroupLens/>.
- [28] Delgado, J., Ishii, N. and Ura, T., "Content-based Collaborative Information Filtering: Actively Learning to Classify and Recommend Documents," *Proc. of Second Int. Workshop, CIA'98*, pp. 206-215, 1998.
- [29] Popescul, A., Ungar, L. H., Pennock, D. M. and Lawrence, S., "Probabilistic Models for Unified Collaborative and Content-Based Recommendation in Sparse-Data Environments," *Proc. of Conf. on UAI*, 2001.



동컴퓨팅등

김 시 관

1982년 경북대학교 전자공학과 졸업
 1984년 한국과학기술원 전산학과 석사
 1984년~1995년 삼성전자, LG정보통신
 2000년 한국과학기술원 전산학과 박사
 2002년~현재 금오공대 컴퓨터공학부 교수. 관심분야는 컴퓨터구조, 병렬처리, 이



임 은 기

1977년 서울대학교 수학과(학사). 1988년 한국과학기술원 전산학과(석사). 1993년 한국과학기술원 전산학과 박사과정 수료
 1989년~현재 금오공과대학교 컴퓨터공학부 교수. 관심분야는 데이터베이스 설계, 소프트웨어 개발 프로세스, 컴퓨터교육



김 주 연

1994년 금오공과대학교 전자계산학과(학사). 1997년 금오공과대학교 전자과(석사). 2001년 금오공과대학교 전자과(박사). 2001년~2003년 부천대학 전산정보처리과 전임강사. 2003년~현재 부천대학 전산정보처리과 조교수. 관심분야는 정보

검색, 정보보호



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 방문 교수. 관심분야는 인공지능, 정보검색, 프

로그랩 테스트 및 검증



이 경

1999년 하얼빈 공정대학교 기계공학과(공학사). 2001년 하얼빈 공정대학교 기계공학과(공학석사). 2001년~현재 금오공과대학교 컴퓨터공학과 박사과정. 관심분야는 정보검색, 정보필터링, 추천시스템, 인공지능