항목 속성과 평가 정보를 이용한 혼합 추천 방법
(A Hybrid Recommendation Method based on Attributes of Items and Ratings)

김 병 만 † 이 경 ††
(Byeong Man Kim) (Qing Li)

요 약 추천 시스템은 일상의 정보를 필터링 해주는 웹 지능화 기술 중의 하나이다. 현재까지 협력기반 (사회기반) 추천 시스템, 내용기반 추천시스템과 이들의 장점을 혼합한 추천시스템들이 개발되어 왔다. 본 논문에서는 클러스터링 기법을 항목기반 협력필터링 틀에 적용한 일명 ICHM이라 불리는 새로운 형태의 혼합 추천 시스템을 소개한다. 이 방법은 항목의 내용 정보를 협력필터링 틀 안에 통합시킴으로써 평가 데이타의 희박성을 줄일 수 있을 뿐만 아니라 새로운 항목 추천 시 발생하는 문제점을 해결할 수 있다. ICHM 방법의 특성 및 성능을 평가하기 위하여 MovieLense 데이타를 이용한 다양한 실험을 하였다. 실험 결과, ICHM 방법이 항목기반 협력 필터링의 예측 질을 향상시킬 뿐만 아니라 새로운 항목 추천 시에도 아주 유용함을 확인할 수 있었다.

키워드 : 혼합 추천 시스템, 항목기반 협력필터링, 내용기반 필터링, 클러스터링

Abstract Recommender system is a kind of web intelligence techniques to make a daily information filtering for people. Researchers have developed collaborative recommenders (social recommenders), content-based recommenders, and some hybrid systems. In this paper, we introduce a new hybrid recommender method - ICHM where clustering techniques have been applied to the item-based collaborative filtering framework. It provides a way to integrate the content information into the collaborative filtering, which contributes to not only reducing the sparsity of data set but also solving the cold start problem. Extensive experiments have been conducted on MovieLense data to analyze the characteristics of our technique. The results show that our approach contributes to the improvement of prediction quality of the item-based collaborative filtering, especially for the cold start problem.

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

1. 서 론

정보필터링 방법 혹은 추천 방법에는 내용기반(content-based) 방법과 협력(collaborative) 방법이 있다. 내용기반 방법은 사용자의 관심사를 표현한 프로파일의 내용과 필터링 대상 항목의 내용을 비교하여 사용자에게 흥미로운 또는 유익한 항목들을 선택하는 방법이다. 이 방법은 텍스트 형태의 항목을 다루는 데 아주 유용한 것으로 알려져 있으며 불리안 모델[1-3], 벡터공간 모델[4], 확률 모델[5] 뉴럴 네트워크 모델[6] , 퍼지집합 모델[7] 등에 기초한 방법들이 있다. 그러나 내용기반 필터링은 내용기반이기 때문에 영화나 음악 등 텍스트 형태가 아닌 항목에 대해서는 적용하기가 어렵다. 이러한 유형의 항목들에 대해서는 자동으로 내용을 분석하기가 상당히 어렵다. 또한, 사용자, 특히, 초보자는 자신의 원하는 정보를 정확히 표현하기가 어렵다. 즉, 사용자 프로파일 구성에 어려움이 있다.

협력 필터링은 타 사용자의 관심사를 예측하는데 동일한 생각을 갖는 사람들의 의견을 이용하는 방법이다. 이 방법은 이력(history) 데이타베이스를 조사하여 대상 사용자와 유사한 관심사를 갖는 사용자들을 찾고 이들이 대상 항목에 대해 어떻게 평가했는지에 대한 정보를 이용하여 대상 항목을 필터링하는 방법이다. 제록스 팔

· 이 논문은 2002년도 한국학술진흥재단의 지원에 의하여 연구되었음 (KRF-2002-041-D00459)
† 종신회원 : 금오공과대학교 컴퓨터공학과 교수
bmkim@se.kumoh.ac.kr
†† 비 회 원 : 금오공과대학교 컴퓨터공학과
liqing@se.kumoh.ac.kr
논문접수 : 2003년 12월 26일
심사완료 : 2004년 9월 22일

로 알토 연구소에서 개발된 Tapestry 텍스트 필터링 시스템[8,9]과 미네소타 대학의 Group Lens 시스템[10]들의 대표적 협력 시스템이다. 협력 필터링 방법은 다양한 분양에서 활용되고 있는데 Ringo 시스템[11]에서는 음악 앨범을 추천하는데 사용하고 있으며, MovieLens 시스템[12]에서는 영화를, Jeter 시스템[13]에서는 유머를, Flycasting[14]에서는 온라인 라디오를 추천하는데 사용하고 있다.

협력 필터링은 내용기반 필터링의 문제점을 해결할 수 있다. 즉, 협력 필터링은 내용을 기반으로 하지 않고 단지 항목에 대한 사용자들의 평가에 기반으로 하기 때문에 음악이나 영화 등과 같이 자동으로 내용을 파악하기 힘든 항목에 대해서도 잘 동작하며 프로파일 구성에 신경을 쓸 필요 없이 항목에 대해서 단지 자신의 평가치만 기술해주면 되기 때문에 초보자인 경우에도 별 문제없이 사용할 수 있다. 이러한 특성으로 인해 협력 필터링 방법은 학문적으로나 상업적으로 널리 사용되고 있다. 하지만, 보다 나은 필터링을 제공하기 위해서는 여전히 해결해야 할 몇 가지 문제점들을 갖고 있다.

- 초기 평가 문제(Early rate problem or Cold start problem) : 새로운 사용자는 아무런 평가 정보도 없기 때문에 협력 필터링 방법을 적용할 수 없다. 또한, 새 항목에 대해서도 아무런 평가 정보도 없기 때문에 협력 필터링 방법을 적용할 수 없다.
- 희박성 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.

반면에 내용기반 필터링은 이러한 문제가 발생하지 않는다. 뿐만 아니라, 오늘날 웹을 통하여 영화나 음악 등 기존에 텍스트 형태의 정보를 얻기 어려웠던 항목에 대해서도 다양한 정보들이 텍스트 형태로 제공되고 있고 영화나 음악으로부터 자동으로 내용을 추출하는 연구들이 다양하게 이루어지고 있다. 따라서, 협력 필터링의 문제점들을 해결하여 궁극적으로 좀 더 나은 성능을 얻기 위해서는 내용기반 방법과 협력 필터링 방법을 결합하는 시도가 필요하다. 여러 결합 방법[15-21]이 제안되어 왔는데 최근[22,23]에서는 협력 필터링 방법 내에서 내용기반 방법의 장점을 살릴 수 있는 새로운 방법을 제안하였다. 이 방법에서는 사용자 프로파일들을 클러스터링 기법을 사용하여 몇 개의 그룹으로 나누고 각 사용자가 그룹에 속할 정도를 퍼지집합 형태로 표시하였다. 바로 이 그룹에 대한 정보를 하나의 항목에 대한 평가 정보로 해석하여 협력 필터링 방법을 적용하는 방법(일명 UCHM : User-based Clustering Hybrid Method)을 사용하였다. 즉, 이 방법에서는 두 종류의 평가 정보를 사용한다. 하나는 순수 협력 필터링 방법에서 사용하던 사용자-항목 평가정보이고 다른 하나는 사용자 프로파일들을 그룹핑하여 얻어진 사용자-그룹 평가정보이다.

UCHM에서는 항목의 내용을 협력 필터링 방법 내에서 매끄럽게 통합함으로써 보다 나은 성능을 보일 수 있고 또한 새로운 사용자에 대해서도 사용자-그룹 평가 정보를 이용하여 협력 필터링 방법을 적용할 수 있게 해준다. 하지만, 이 방법은 여전히 새로운 항목에 대해서는 적절히 처리를 하지 못하며 사용자기반 협력 필터링[24]을 사용하기 때문에 여전히 성능 개선의 여지가 남아 있다. 최근, 저자는 사용자를 그룹핑하는 대신에 항목들을 그 내용에 따라 그룹핑하여 그룹-항목 평가 정보를 얻어내고 이 평가 정보와 기존의 사용자-항목 평가정보에 항목기반 협력 필터링 방법[24]을 적용시키는 방법(일명 ICHM : Item-based Clustering Hybrid Method)[25,26]을 제안하였다. 본 논문에서는 ICHM 방법에 대해서 소개하고 이의 특성을 다양한 실험을 통하여 분석하고자 한다. 그리고, UCHM과의 비교를 통하여 ICHM 방법의 유용성을 보이고자 한다. 먼저, 다음 장에서는 내용기반 필터링과 협력 필터링을 결합시키는 기존 다른 접근 방법에 대해서 살펴보고 3장에서는 ICHM 방법에 대한 내용과 UCHM과의 유사성 및 차이점 그리고 이해를 돕기 위한 예제를, 4장에서는 다양한 실험결과에 대해서 소개하도록 하겠다.

2. 관련 연구

내용기반 방법과 협력 방법을 결합하는 방법은 크게 세 부류로 나누어 볼 수 있다. 첫 번째는 그림 1에서 보는 바와 같이 협력 필터링 방법과 내용기반 방법의 결과를 선형적으로 결합시키는 방법으로 Claypool 이 제안한 시스템[15]과 Wasfi가 제안한 시스템[16]이 이러한 유형에 속한다. Wasifi에 제안된 ProfBuilder는 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 목

록은 내용기반 필터링에 의해서 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. Claypool은 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 두 가지 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.

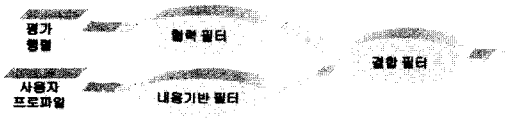

그림 1 선형 결합 방법

두 번째 유형은 그림 2처럼 내용기반 필터링과 협력 필터링을 순차적으로 결합시키는 방법이다. 이러한 유형에서는 먼저 유사한 취미를 가지는 사용자를 찾기 위해 내용기반 필터링 기법을 적용하고 그 다음 그 결과에 협력 필터링 방법을 적용하는 방법으로 RAPP 시스템[17]과 Fab 시스템[18] 등이 있다. RAPP는 WWW 상에서 발견한 특정 영역의 정보를 분류하는데 도움을 주며 이러한 URL들을 유사한 관심사를 갖는 사용자에게 추천하여 준다. 이 시스템에서는 유사한 사용자를 찾기 위해 웹 페이지의 카테고리를 기반으로 확장가능한 피어슨 상관 알고리즘(scalable Pearson correlation algorithm)을 사용한다. Fab 시스템에서는 사용자의 항목 평가(rating) 정보를 이용하지 않고 대신에 내용기반의 사용자 프로파일을 생성하여 사용한다. 따라서, 예측의 질은 상당 부분 내용기반 방법에 의존하게 된다. 즉,

그림 2 순차 결합 방법

그림 3 하이브리드 결합 방법

부정확한 프로파일은 부정확한 사용자간 상관관계를 초래하며 이로 인해 결과적으로 좋지 않은 예측 결과를 낳게 된다.

세 번째 유형은 그림 3처럼 항목의 내용과 평가 정보를 혼합시키는 방법으로 Popescul의 확률모델[20], Basu의 Ripper 시스템[21], Good의 에이전트기반 방법[19], 본 연구자들의 이전에 제안한 UCHM[22,23] 등이 있다. Basu와 그 일행은 좀 더 나은 추천을 위해 Ripper 시스템을 사용하여 내용정보와 사용자 평가 정보 모두를 학습시키는 방법을 사용했으며, Good와 그 일행은 개인 에이전트와 사용자 평가정보를 협력필터링 프레임워크 안에서 결합시키는 방법을 통하여 내용기반 필터링과 협력 필터링을 결합시키려고 하였다. Kim과 그 일행은 사용자 프로파일을 그룹핑하여 얻어낸 정보를 일종의 항목으로 취급하고 이 확장된 평가 정보를 바탕으로 협력필터링을 실행하는 방법을 제안하였다. Popesscul과 그 일행은 기존의 협력 필터링에 대한 확률모델을 확장하여 항목의 내용도 다룰 수 있도록 하였다.

3. ICHM

본 논문에서는 UCHM 방법의 문제점과 성능개선을 위하여 항목의 내용을 바탕으로 자동으로 얻어낸 그룹-항목 정보(즉, 항목이 각 그룹에 속할 정도)와 사용자-항목 평가정보에 항목기반 협력필터링 방법을 적용하는 방법(ICHM : Item-based Clustering Hybrid Method)을 소개하고자 한다. 본 방법은 UCHM과 방법론적으로 상당히 유사하다. 주요한 차이점은 그림 4에서 보는 바와 같이 UCHM에서는 사용자 그룹을 항목처럼 취급하였고 반면에 ICHM에서는 항목 그룹을 마치 사용자처럼 취급하였다. 그리고, UCHM에서는 사용자기반 협력필터링 방법을 사용한 반면에 ICHM에서는 항목기반 협력 필터링 방법을 사용하였다.

그림 5에서 보는 바와 같이 본 방법은 먼저 항목의 내용을 클러스터링 알고리즘을 사용하여 몇 개의 그룹으로 나누고 그 결과로부터 항목이 각 그룹에 속할 정도를 파악하여 그룹-항목 평가정보를 구축한다. 이렇게 구해진 그룹-항목 평가 정보와 원래 주어진 사용자-항목 평가정보를 결합하여 최종 항목간 유사도를 구한 후

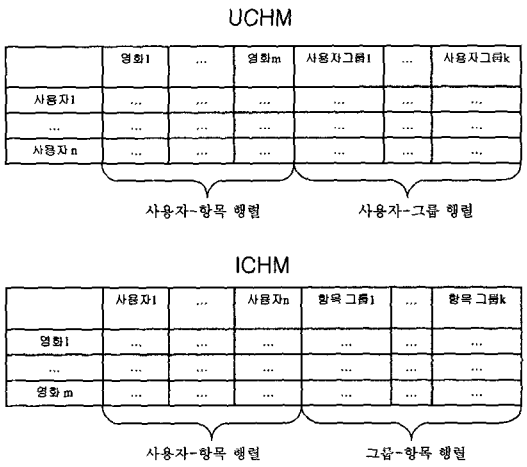

그림 4 UCHM 및 ICHM에서의 평가 데이터

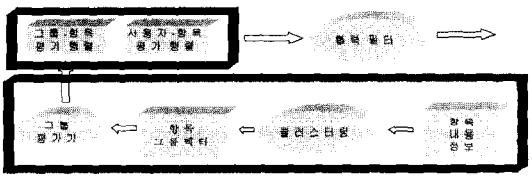

그림 5 ICHM 구성도

이웃 항목들의 평균으로부터 가중치 편차 평균을 계산하여 항목에 대한 예측을 한다.

3.1 그룹 평가

그룹 평가는 항목들을 몇 개의 유사 그룹으로 묶어 이 정보를 협력 필터링 시 유사도 계산에 반영시킬 수 있도록 하는데 있다. 그룹은 항목의 내용에 의해 결정되기 때문에 결과적으로 항목의 내용이 유사도 계산 시에 반영이 된다. 각 항목은 자신들의 속성(attribute), 예를 들어, 영화인 경우는 남자배우, 여배우, 감독, 장르, 줄거리 등을 갖고 있는데 이 속성의 값을 이용하여 그룹핑 할 수 있다.

K-평균 클러스터링 알고리즘은 간단하면서도 빠른 방법으로 널리 사용되고 있다[27]. K-평균 클러스터링 알고리즘에서는 보통 객체가 속할 그룹 하나가 정해진다. 하지만, 본 논문에서는 객체가 각 그룹에 속할 정도를 계산하여야 한다. 따라서, 본 논문에서 이 알고리즘을 목적에 맞게 변형하여 사용하였다. 보완 k-평균 알고리즘은 그림 6에서 보는 바와 같이 기존 K-평균 알고리즘과 거의 유사하다. 단지, 최종적으로 그룹을 형성한 후 아래와 같은 수식을 이용하여 각 그룹에 속할 정도를 계산하는 점만 틀리다. 즉, 객체가 속할 그룹 정보가 퍼지집합 형태로 표현된다.

$$\Pr o(j,k) = 1 - \frac{CS(j,k)}{Max_{i\ CS(i,k)}} \quad (1)$$

여기서, Pro(j,k)는 객체 j가 클러스터 k에 속할 정도를 나타내며 CS(j,k)는 객체 j와 클러스터 k와의 역유사도(counter-similarity)로 Cosine 방법을 사용하여 계산한다. $Max_i CS(i,k)$는 클러스터 k와 가장 유사하지 않는 객체와의 역유사도 값이다.

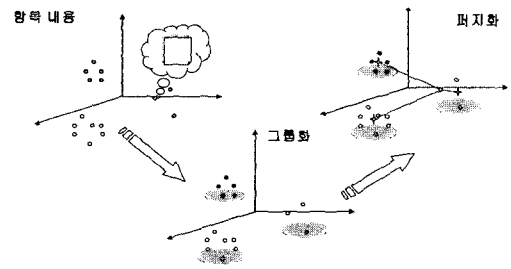

그림 6 보완 k-평균 클러스터링 알고리즘

위의 방법 외에 K-평균 알고리즘에 퍼지집합 개념을 도입하여 객체가 각 그룹에 속할 정도를 파악할 수 있는 퍼지 k-평균 알고리즘[28]이 있다. 이 방법에서는 그림 7에서 보는 바와 같이 그룹 형성 과정 중에 이미 퍼지집합이 사용되어 지고 자연스럽게 최종 결과도 퍼지

퍼지 k-menas 클러스터링 알고리즘
입력 : 클러스터 수 k, 항목 내용벡터들

(1) 클러스터와 항목 간 소속정도를 무작위로 초기화 한다.
(2) 전체비용함수가 거의 변화가 없을 때까지 (a)와 (b)의 과정을 반복한다.
  (a) 각 클러스터의 중심 벡터를 재계산한다.
  (b) 각 항목이 클러스터에 속할 정도를 재계산한다.
(3) 각 항목의 퍼지 소속정도를 반환한다.

그림 7 퍼지 k-평균 알고리즘

집합으로 표현이 된다. 본 논문에서는 클러스터링 방법에 따른 효과를 파악하기 위해 이 방법도 사용하였으며, 이때 다음과 같은 비용함수(cost function)와 소속함수(membership function)을 사용하였다.

$$GCF_{fuz} = \sum_{i=1}^{c}\left(\sum_{j=1}^{n}\left((\mathrm{Pro}_{i,j})^{b} \times Dis_{i,j}\right)\right) \quad (2)$$

$$Mean_i = \frac{\sum_{j=1}^{n}(\mathrm{Pro}_{i,j})^{b} X_j}{\sum_{j=1}^{n}(\mathrm{Pro}_{i,j})^{b}} \quad (3)$$

$$\mathrm{Pro}_{i,j} = \frac{\left(\frac{1}{Dis_{i,j}}\right)^{\frac{2}{b-1}}}{\sum_{r=1}^{c}\left(\frac{1}{Dis_{r,j}}\right)^{\frac{2}{b-1}}} \quad (4)$$

여기서, GCF_{fuz}는 퍼지 전체비용함수(global cost function)이며 c는 크러스터의 개수, b는 클러스터 간의 비중을 조절하는 값이며 $Dis_{i,j}$는 클러스터의 중심 벡터와 객체 j 벡터 X_j와의 유클리디안 거리를 의미한다. $\mathrm{Pro}_{i,j}^{b}$는 객체 j가 클러스터 i에 속할 정도를 의미한다.

3.2 항목간 유사도

앞절의 방법을 통하여 항목들을 그룹핑하면 그 결과로 새로운 평가 행렬, 즉 그룹-항목 행렬을 얻을 수 있다. 이 행렬과 원래 주어진 사용자-항목 평가행렬에 항목기반 필터링 방법을 적용시켜 특정 사용자가 특정 항목을 선호할 정도를 예측할 수 있다. 이를 위해서는 항목간 유사도를 계산하여야 하는데 본 논문에서는 두 종류의 행렬을 사용하기 때문에 이들을 결합하여 유사도를 계산하는 몇 가지 방법을 제안하였고 이들의 성능을 다음 장에서 비교하였다.

• 피어슨 상관관계 공식을 이용한 유사도

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. 본 논문에서 사용한 아래의 공식은 [24]에서 사용한 공식으로 UCHM에서 사용한 공식과 거의 유사하다. 단, UCHM에서는 사용자 기반이기 때문에 항목 대신에 사용자를 사용했다는 점이 틀리다.

$$\mathrm{sim}(k,l) = \frac{\sum_{u=1}^{m}(R_{u,k}-\overline{R_k})(R_{u,l}-\overline{R_l})}{\sqrt{\sum_{u=1}^{m}(R_{u,k}-\overline{R_k})^2}\sqrt{\sum_{u=1}^{m}(R_{u,l}-\overline{R_l})^2}} \quad (5)$$

여기서, sim(k, l)은 항목 k와 l 사이의 유사도를 나타내며, m은 항목 k와 l 모두에 평가를 한 사용자 수를, $R_{u,k}$ 와 $R_{u,l}$은 각각 사용자 u가 항목 k와 l에 내린 평가치를, $\overline{R_k}$와 $\overline{R_l}$는 항목 k와 l의 평균 평가치를 의미한다.

• 보완 코사인 유사도(Adjusted Cosine Similarity)

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, Bob이 가장 선호하는 영화를 4라고 평가를 생각했다면 5라는 평가는 하지 않을 것이다. 나쁜 영화에 대한 평균 평가가 2인데도 1이라고 평가할 수 있다. 그러나, Oliver는 가장 좋은 영화는 5, 나쁜 영화는 2라고 평가한다. 기존의 코사인 유사도를 사용한다면 2명에 대한 유사도는 아주 다를 것이다. 보완된 코사인 유사도 [24]는 이러한 단점을 보완할 수 있다. 피어슨 상관관계 공식과 마찬가지로 본 논문에서 사용한 아래의 공식은 UCHM에서 사용한 공식과 거의 유사하다. 단, UCHM에서는 사용자 기반이기 때문에 항목 대신에 사용자를 사용했다는 점이 틀리다.

$$\mathrm{sim}(k,l) = \frac{\sum_{u=1}^{m}(R_{u,k}-\overline{R_u})(R_{u,l}-\overline{R_u})}{\sqrt{\sum_{u=1}^{m}(R_{u,k}-\overline{R_u})^2}\sqrt{\sum_{u=1}^{m}(R_{u,l}-\overline{R_u})^2}} \quad (6)$$

여기서, sim(k, l)은 항목 k와 l 사이의 유사도를 나타내며, m은 항목 k와 l 모두에 평가를 한 사용자 수를, $R_{u,k}$ 와 $R_{u,l}$은 사용자 u가 항목 k와 l에 내린 평가치를, $\overline{R_u}$는 사용자 u의 평균 평가치를 의미한다.

• 선형 결합

보통, 사용자-항목 평가 행렬 정보는 이산 값(예, MovieLens 데이타인 경우 평가치는 1과 5 사이의 정수)을 갖으며 그룹-항목 평가 행렬은 0과 1 사이의 연속적인 값을 갖는다. 따라서, 한쪽의 값을 다른 쪽의 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 두 가지 방법에 대해서 그 성능 평가를 수행하였다.

- Enlarged Pearson : 그룹-항목 평가 행렬을 사용자-항목 평가 값의 범위로 확대하고 두 개의 행렬을 하나의 행렬로 취급한 후 피어슨 상관관계 공식을 적용
- Combination Approach : 피어슨 상관관계 공식과 보완 코사인 유사도를 적용시킨 후 아래의 수식에 따라 결합

$$\mathrm{sim}(k,u) = \mathrm{sim}(k,u)_{\mathrm{item}} \times (1-c) + \mathrm{sim}(k,u)_{group} \times c \quad (7)$$

여기서, $\mathrm{sim}(k,u)_{\mathrm{item}}$은 사용자-항목 평가 행렬에 피어슨 상관관계 공식을 적용시켜 얻은 항목 k와 u 사이의 유사도를, $\mathrm{sim}(k,u)_{group}$은 그룹-항목 평가 행렬에 보완 코사인 공식을 적용시켜 얻은 항목 k와 u 사이의

유사도를, c는 이들 사이의 결합 계수를 의미한다.

3.3 협력 예측

사용자 u의 항목 k에 대한 예측을 구하기 위해 GroupLens에서 제안한 식을 사용하였다. 여기서는, 항목에 대한 예측은 해당 항목에 대한 사용자들의 평균 평가치에 이웃 항목들의 평균값에 대한 편차의 가중치 평균을 더함으로써 계산된다. 그리고 항목의 유사성에 기반한 가장 인접한 N 개의 이웃을 선택하기 위하여 top N 규칙을 사용한다.

$$P_{u,k}=\overline{R_k}+\frac{\sum_{i=1}^{n}(R_{u,i}-\overline{R_i})\times \mathrm{sim}(k,i)}{\sum_{i=1}^{n}|\mathrm{sim}(k,i)|} \quad (8)$$

여기서, $P_{u,k}$는 항목 k에 대한 사용자 u에 대한 예측을 표시한다. n은 항목 k의 최인접 이웃의 수, $R_{u,i}$는 항목 i에 대한 u의 평가, $\overline{R_k}$는 항목 k에 대한 평균 평가, $\overline{R_i}$는 항목 i에 대한 평균 평가, sim은 항목 k와 이웃 항목 i사이의 유사도를 의미한다.

3.4 새로운 항목 문제

전통적인 협력 필터링 방법은 새로운 사용자에 대해서는 평가 정보가 없기 때문에 추천하기가 곤란하다. UCHM 방법[22]에서는 이러한 문제를 협력 필터링 프레임워크 안에서 해결할 수 있음을 보였다. 즉, 사용자 프로파일에 기초한 그룹 정보에 기존 협력 필터링 방법을 적용시키는 방법을 제안하였다. 하지만, 이 UCHM 방법도 여전히 새로운 항목에 대해 어떻게 추천해야 할지를 제시하지 못하고 있다. ICHM 방법에서는 항목 그룹에 대한 정보를 이용하기 때문에 이러한 문제를 해결할 수 있다. 단, 수식 (8)을 그대로 사용하기는 곤란하다. 왜냐하면, 항목에 대한 아무런 평가 정보가 없기 때문에 $\overline{R_k}$가 0이 된다. 이에 대한 첫 번째 대안은 평균 방법 (average method), 즉, $\overline{R_k}$ 대신에 $\overline{R_{\mathrm{neighbor}}}$를 사용한다. $\overline{R_{\mathrm{neighbor}}}$는 새로운 항목 k와 최근접한 이웃항목들의 평균 평가치를 의미한다. 새로운 항목 k의 이웃은 사용자 항목 평가 정보 없이도 구할 수 있다. 즉, 항목 내용을 기반으로 자동으로 생성되는 그룹 항목 평가 정보를 이용하면 이웃 항목들을 구할 수 있다. 또, 다른 대안은 [24]에서 제안한 아래와 같은 가중치 합 (weighted sum) 방법을 사용하는 것이다.

$$P_{u,k}=\frac{\sum_{i=1}^{n}R_{u,i}\times \mathrm{sim}(k,i)}{\sum_{i=1}^{n}|\mathrm{sim}(k,i)|} \quad (9)$$

여기서, $P_{u,k}$는 항목 k에 대한 사용자 u의 평가 예측치를, n은 항목 k의 근접 이웃들의 수를, $R_{u,i}$는 사용자 u의 항목 i에 대한 평가치를, sim(k, i)는 항목 k와 i 사이의 유사도를 의미한다. 이 방법은 보는 바와 같이 항목 k에 대한 평균 평가치를 구할 필요가 없다. 이러한 점이 새로운 항목에 대해 평가정보 없이 내용기반 정보만으로도 예측 가능케 한다. 본 논문에서는 위의 두 방법을 모두 구현하여 그 성능을 비교하여 보았다.

3.5 제안방법 적용 예

표 1과 같은 사용자-항목 평가 행렬을 이용하여 본 제안방법에 대한 이해를 돕도록 하겠다. 단, 평가치는 1과 5 사이의 값으로 다음과 같은 의미를 갖는다.

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

표 1 사용자-항목 평가 행렬

	철수	옥경	상원
국화꽃향기 (G)	5	3	
클래식 (K)	5	2	4
색즉시공 (S)	2	5	4
친구 (F)	4	2	
선생김봉두 (T)			

(1) 먼저, 영화의 내용 (예를 들어, 장르, 감독, 연기자 등)을 바탕으로 영화를 그룹핑한다. 여기서는 지면 제약 상 단순히 그 결과가 아래와 같다고 가정한다. 영화 제목 옆의 비율은 해당 영화가 그룹에 속할 정도를 백분율로 표현한 것이다.

그룹1 : 국화꽃향기(98%), 클래식(90%), 색즉시공(10%), 친구(98%), 선생김봉두(10%)

그룹2 : 국화꽃향기(2%), 클래식(10%), 색즉시공(90%), 친구(2%), 선생김봉두(90%)

(2) 1의 결과로부터 표 2와 같은 그룹-항목 평가 행렬을 구축한다.

표 2 그룹-항목 평가 행렬

	그룹1	그룹2
국화꽃 향기 (G)	98%	2%
클래식 (K)	90%	10%
색즉시공 (S)	10%	90%
친구 (F)	98%	2%
선생 김봉두 (T)	10%	90%

(3) 표 1과 표 2를 이용하여 특정 사용자가 특정 항목을 좋아할 정도를 계산하여야 한다. 이를 위해서는 항목간의 유사도를 계산하여야 하는데 여기서는 "국화꽃향기" (G로표기)와 "클래식" (K로 표기) 간의 유사도를 선형 결합 방법을 통하여 구하는 예를 보인다. 먼저, 표 1를 이용하여 두 영화간의 유사도를 피어슨 상관관계 공식을 이용하여 아래와 같이 구한다.

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

다음, 표 2의 정보를 바탕으로 보완 코사인 공식을 이용하여 아래와 같이 두 영화간의 유사도를 구한다.

$$\mathrm{sim}(G,K)_{\mathrm{group}}=\frac{(0.9-0.5)\times(0.98-0.5)+(0.1-0.5)\times(0.02-0.5)}{\sqrt{(0.9-0.5)^2+(0.1-0.5)^2}\times\sqrt{(0.98-0.5)^2+(0.02-0.5)^2}}=1$$

최종적으로 위의 결과를 아래와 같이 선형 결합한다. 여기서는, 결합 계수를 0.4로 사용하였다.

$$\mathrm{sim}(G,K)=\mathrm{sim}(G,K)_{\mathrm{item}}\times0.6+\mathrm{sim}(G,K)_{\mathrm{group}}\times0.4=1.0\times0.6+1.0\times0.4=1$$

비슷한 방법으로 "국화꽃향기"와 "색즉시공" 간의 유사도도 아래와 같이 구할 수 있다.

$$\mathrm{sim}(G,S)=\mathrm{sim}(G,S)_{\mathrm{item}}\times0.6+\mathrm{sim}(G,S)_{\mathrm{group}}\times0.4=(-1.0)\times0.6+(-1.0)\times0.4=-1$$

(4) 3의 방법을 통해 구한 항목간 유사도를 바탕으로 식 (8)을 적용하여 최종적인 예측값들을 계산한다. 표 3은 그 결과를 보여주고 있다.

표 3 예측 결과

	철수	옥경	상원
국화꽃향기 (G)	5	3	4
클래식 (K)	5	2	4
색즉시공 (S)	2	5	4
친구 (F)	4	2	3
선생김봉두 (T)	1.6	5.8	3.7

상원이가 "국화꽃향기"를 좋아할 정도는 아래의 수식을 통하여 구하였으며 상원이가 "친구"를 좋아할 정도도 비슷한 수식을 통하여 구하였다.

$$P_{\text{상원},G}=\overline{R_G}+\frac{(R_{\text{상원},K}-\overline{R_K})\times\mathrm{sim}(G,K)+(R_{\text{상원},S}-\overline{R_S})\times\mathrm{sim}(G,S)}{|\mathrm{sim}(G,K)|+|\mathrm{sim}(G,S)|}=4+\frac{(4-3.5)\times1+(4-3.5)\times(-1)}{1+1}=4$$

그리고, 새로운 항목 "선생김봉두"를 각 사용자가 좋아할 정도는 3.4절에서 언급한 평균 방법을 사용하여 구하였다. 이때, $\overline{R_{\mathrm{neighbor}}}$는 "선생김봉두"와 가장 유사한 "색즉시공"의 평균값 ((2+5+4)/3=3.7)을 사용하였다.

$$P_{\text{철수},T}=3.7+\frac{(5-3)\times(-1)+(5-3)\times(-1)+(2-4.5)\times1+(4-2)\times(-1)}{4}=1.6$$

$$P_{\text{옥경},T}=3.7+\frac{(3-5)\times(-1)+(2-4.5)\times(-1)+(5-3)\times1+(2-4)\times(-1)}{4}=5.8$$

$$P_{\text{상원},T}=3.7+\frac{(4-3.5)\times(-1)+(4-3.5)\times1}{2}=3.7$$

여기서, 사용된 "선생김봉두"와 다른 항목 간의 유사도는 아래와 같다.

$$\mathrm{sim}(G,T)_{group}=\frac{(01-0.5)\times(0.98-0.5)+(0.9-0.5)\times(0.02-0.5)}{\sqrt{(01-0.5)^2+(0.9-0.5)^2}\times\sqrt{(0.98-0.5)^2+(0.02-0.5)^2}}=-1$$

$$\mathrm{sim}(K,T)_{group}=-1,\quad \mathrm{sim}(S,T)_{group}=1,\quad \mathrm{sim}(F,T)_{group}=-1$$

표 3에서 보는 바와 같이 "선생 김봉두"는 원래 아무런 평가 정보도 없었다. 즉, 새로운 항목이다. 기존의 항목기반 협력 필터링 방법은 표1과 같은 사용자-항목 평가정보만을 이용하기 때문에 "선생 김봉두"와 같은 새로운 항목에 대해서는 예측을 하기가 불가능하다. 하지만, 본 제안 방법은 그룹-항목 평가정보로부터 항목의 내용을 이용할 수 있어 새로운 항목에 대해서도 예측을 할 수 있으며 또한 기존 항목에 대해서도 단순히 평가 정보뿐 아니라 내용 정보도 같이 고려하여 예측할 수 있어 좀 더 나은 예측을 할 수 있다.

4. 실험 및 평가

4.1 평가 데이타 및 평가 함수

현재 웹에 기반을 둔 추천 시스템인 MovieLens[12]에서 수집된 영화 평가에 대한 데이타를 이용하였다. 데이타 집합은 943명의 사용자, 1682개의 영화를 각 사용자가 적어도 20개의 항목에 대해 평가를 한 100,000개의 평가를 포함하고 있다. MovieLens의 평가는 사용자들이 1에서 5사이의 정수 값으로 직접 평가 자료를 입력하였다. 이 데이타 집합은 훈련 집합과 테스트 데이타 집합으로 구성되어 있다.

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 예측치간의 오류를 구하고 이 오류의 절대값을 합한 후 테스트 대상의 수로 나누어 줌으로써 얻을 수 있다. MAE가 낮을수록 예측의 정확도는 좋아지게 된다.

$$MAE=\frac{\sum_{i=1}^{n}|p_i-q_i|}{n} \tag{10}$$

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

4.2 성능 평가

ICHM 방법의 항목 그룹핑 방법에 따른 영향을 파악

하기 위해 3.1 절에 기술한 두 가지의 그룹핑 방법들을 구현하고 이들의 그룹 수에 따른 성능을 평가하였다. 그림 8에서 보는 바와 같이 그룹의 수가 성능에 영향을 미침을 알 수 있다. 이론적으로 퍼지 k-평균 알고리즘이 보다 보완 k-평균 알고리즘 보다 애매함을 잘 표현하지만 본 실험에서는 뚜렷한 장점을 보이지 않았다. 퍼지 k-평균 알고리즘이 계산 복잡도가 보완 k-평균 알고리즘보다 좀 더 복잡하기 때문에 이 후 실험에서는 보완 k-평균 알고리즘을 사용하였다.

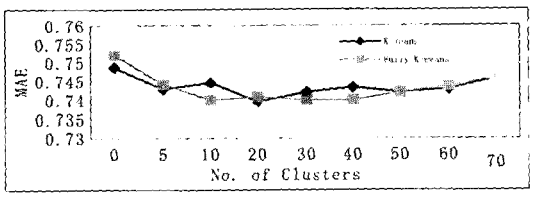

그림 8 그룹핑 방법에 따른 성능

본 논문에서는 항목간 유사도를 구하는 방법으로 두 가지를 사용하여 그 성능을 평가하고자 하였다. 이 중 선형결합 방법은 결합계수에 따라 성능이 달라진다. 따라서, 먼저 이 결합 계수를 0.1 단위로 변경시키면서 이 방법의 성능을 평가하였다. 그림 9에서 보는 바와 같이 0.4 부근에서 성능이 좋음을 알 수 있다.

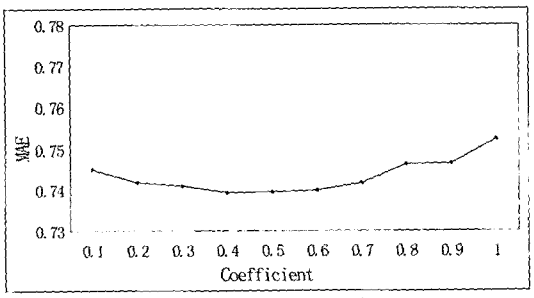

그림 9 결합 계수에 따른 성능 평가

그림 10은 유사도를 구하는 두 가지 방법의 성능을 기존 항목기반 협력 필터링 방법과 비교하였다. 선형결합 방법을 사용한 것이 "combination ICHM"이고, 확대 방법을 사용한 것을 "Enlarged ICHM"이다. 두 방법 모두 기존 항목기반 협력 필터링 방법 보다 좋음을 알 수 있다. 두 방법 중에서는 선형 결합방법이 확대 방법보다 더 좋음을 알 수 있다. 보통 협력 필터링 방법은 고려되는 이웃의 수에 따라 성능이 달라진다. 본 논문에서도 이웃의 수를 변화시키면서 그 성능을 비교하여 보았다. 그림 10에서 보는 바와 같이 이웃의 수를 30으로 했을 때가 보다 성능이 좋음을 알 수 있다.

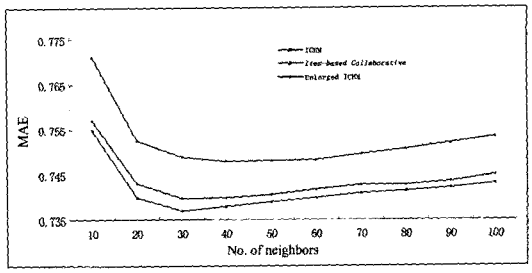

그림 10 유사도 계산 방법에 따른 성능 비교

앞에서의 실험은 항목 그룹 형성 시 영화의 장르만을 고려하였다. 즉, MovieLens 데이타는 19개의 장르를 사용하기 때문에 각 영화를 19차원의 이진 벡터로 보고 이들을 유사한 벡터들로 그룹핑하여 사용하였다. 장르는 영화의 내용을 나타내는 아주 간단한 방법이다. 좀 더 정확한 방법으로 영화의 내용을 표현할 수 있다면 좀 더 나은 성능을 기대할 수 있을 것이다. 이를 검증하기 위해 Internet Movie Database(http://www.imdb.com)에서 여배우, 남자배우, 감독 및 영화에 대한 줄거리(synopsis)를 수집하고 이를 실험에 사용하였다. 여배우, 남자배우 및 감독을 사용하여 항목의 내용을 표현할 경우는 장르와 마찬가지로 이진 벡터 형태로 표현하였으며 줄거리를 사용할 경우는 정보검색 모델 중의 하나인 벡터공간 모델에 기초하여 표현하였다. 즉, 문서를 그 내부 단어들의 가중치 벡터로 표현하였다. 가중치는 아래와 같이 산정하였다.

$$w_{i,j} = f_{i,j} \times idf_i, \quad f_{i,j} = \frac{freq_{i,j}}{Max_k freq_{k,j}}, \quad idf_i = \log \frac{N}{n_i} \quad (11)$$

여기서, $w_{i,j}$는 단어 i의 문서 j에서의 가중치를, $f_{i,j}$는 단어 i의 문서 j에서의 정규화된 출현빈도수를, $freq_{i,j}$는 단어i의 문서 j에서의 출현빈도수를, idf_i는 단어 i의 역문헌 빈도수를, N은 전체 문헌수를, n_i는 단어 i를 갖는 문헌의 수를 의미한다. 실험 결과, 그림 11에서 보는 바와 같이 줄거리를 사용하였을 경우가 배우나 감독, 장르 정보보다 더 유용함을 알 수 있다. "모두"는 감독, 장르, 배우, 줄거리 정보 모두를 사용한 경우로 줄거리 정보만을 사용한 경우 보다 성능 향상이 그리 크지 않았다.

새로운 항목에 대한 본 방법의 특성을 분석하기 위해, 먼저, MovieLens 학습 데이타에서 무작위로 하나를 선택하여 그것에 대한 평가 정보를 모두 삭제하여 새로운 항목으로 만들어 실험을 하였다. 실제 실험에서는 946번 항목이 선택되었다. 학습 데이타에서 이 항목에 대한 평가 정보를 모두 제거시킨 후 테스트 데이타에 있는 11 사용자의 평가 정보를 평가용으로 사용하였다. 그림 12에서 "Real"이라고 표현된 부분이 실제 평가 정보이며

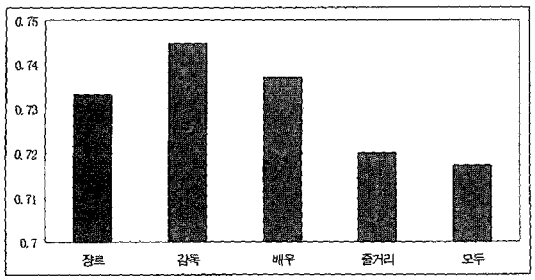

그림 11 항목 내용에 따른 성능 평가

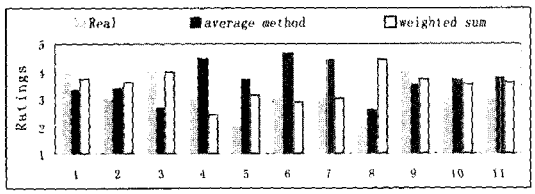

그림 12 한 새로운 항목에 대한 예측 결과

"average method"라고 표기된 부분이 3.4절의 첫 번째 방법을, "weighted sum"은 두 번째 방법을 나타낸다. 보는 바와 같이 새로운 항목에 대해서도 어느정도 사용자의 관심사를 예측할 수 있음을 알 수 있다.

위의 결과는 단지 한 항목에 대한 결과이다. 이를 객관화하기 위해 새로운 항목 10 개에서부터 50개까지 10 단위로 변경시키면서 그 성능을 측정하였다. 표 4에서 보는 바와 같이 "Averge method"가 새로운 항목에 대해서는 더 나은 성능을 보임을 알 수 있다.

표 4 새로운 항목에 대한 성능 평가(평가척도: MAE)

	10	20	30	50
Weighted sum	0.756	0.765	0.885	0.956
Average method	0.623	0.749	0.825	0.876

표 4는 새로운 항목만을 고려한 성능 평가이다. 그러나, 실제 환경 하에서는 새로운 항목도 있고 그렇지 않은 항목들도 있다. 따라서, 새로운 항목뿐만 아니라 기존 항목 모두에 대한 성능을 평가해 볼 필요가 있다. 이를 위해 100개의 항목을 새로운 항목으로 취급했을 경우와 그렇지 않았을 경우를 비교 실험하였다. 즉, 전체 학습데이타 중 100개를 새로운 항목으로 취급하고 나머지는 그대로 사용한 경우와 전체 데이타를 그대로 사용한 경우를 비교 실험하였다. 그림 13은 그 결과를 보여준다. "Cold Start"라고 표시된 것이 100개 항목을 새로운 항목으로 취급한 경우의 결과이다. 그림에서 보는 바와 같이 당연히 새로운 항목으로 취급했을 경우가 그렇지 않는 경우보다 성능이 떨어진다. 하지만, 그 성능 차

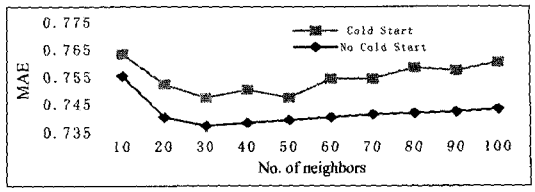

그림 13 새로운 항목과 기존 항목 모두 고려한 성능 평가

이가 그리 크지 않음을 알 수 있다. 이는 새로운 항목에 대해서 본 제안 방법을 사용하면 내용정보만으로도 충분히 유용한 예측 결과를 얻을 수 있음을 의미한다.

4.3 타 연구와의 비교

협력 필터링에 클러스터링 기법을 도입한 몇몇 연구[29,30]가 있었다. [29]는 항목공간의 축소를 통하여 평가정보의 희박성(sparsity) 문제를 줄이려고 항목들을 클러스터링하는 방법을 사용하였다. 하지만, 여러 클러스터링 방법에 따른 실험에도 불구하고 예측 성능이 클러스터링을 적용하지 않는 경우보다 좋지 않았다. 단, 계산시간 측면에서 이득은 있었다. [30]에서는 유사한 성향의 사용자들을 그룹핑하기 위하여 평가 정보를 바탕으로 사용자들을 RecTree 알고리즘에 기초하여 클러스터링하는 방법을 제안하였다. 이 방법은 협력 필터링의 확장성(scalability) 문제에 주로 초점이 맞추어졌으나 특정 업무에 대해서는 예측 질 면에서도 클러스터링을 적용하지 않는 방법보다 성능이 좋은 경우가 있었다. 어쨌든, 이러한 기존의 방법들은 협력 필터링의 문제점 중의 하나를 해결하기 위해 클러스터링 기법을 적용했다는 측면에서 본 방법과 유사하나 추구하는 목적과 구체적인 방법에서 차이가 있다. 즉, 본 방법은 항목의 내용을 협력필터링 방법에 효과적으로 도입하기 위하여 클러스터링 기법을 사용하였으며 이를 통해 협력필터링의 희박성 문제를 줄이고 예측 질도 향상시키고자 하였다.

2장의 관련연구에서 보았듯이 내용기반 필터링과 협력 필터링을 결합시키는 다양한 연구가 이루어져 왔다. 이 중 어떤 시스템[17]은 사용자의 선호도를 표현하기 위하여 불리안 값을 사용하며, 또한 어떤 시스템[15]는 시간에 따라 예측 질이 달라지는 경우가 있다. [15]에서는 시스템 패러미터가 사용자 피드백에 따라 변하는데 이에 대한 명확한 기술이 없다. 이러한 이유들로 인하여 이들 간의 객관적 비교가 힘들다. 하지만, 이들 간의 개념적인 비교를 통하여 본 방법의 효율성을 판단할 수 있다. 이러한 비교는 [22]에 자세히 기술되어 있어 지면상 생략하고 여기서는 본 방법과 유사한 UCHM 방법[22,23]과 ICHM 방법과의 비교를 통하여 제안방법의 유용성을 보이고자 한다.

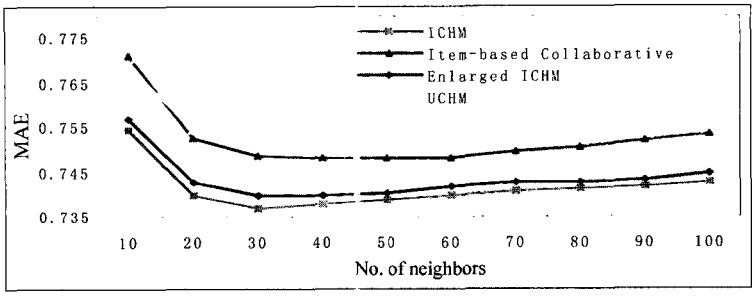

그림 14 UCHM과 ICHM의 성능 비교

MovieLens 데이타에는 사용자 프로파일 정보가 명시적으로 주어지지 않는다. 단지, 영화의 장르와 그 영화에 대한 사용자의 평가치만 주어진다. 따라서, [22]에서는 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.

$$w_m = \frac{Num_{\text{item} \subset attribute_m \mid \text{item} > threshold}}{Num_{\text{item} > threshold}}$$

여기서, w_m은 m번째 장르에 대한 가중치를, $Num_{\text{item} > threshold}$는 해당 사용자가 평가한 영화 중에서 평가치가 threshold 보다 큰 영화의 개수를, $Num_{\text{item} \subset attribute_m \mid \text{item} > threshold}$는 해당 사용자가 평가한 영화 중에서 평가치가 threshold 보다 크고 장르가 m번째 장르인 영화의 개수를 의미한다. MovieLens 데이타의 평가치는 1과 5사이의 정수값으로 1과 2는 부정적 평가를 나타내기 때문에 [22]의 실험에서는 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/0.333, 액션/0.666}}로 표현된다.

본 논문에서는 사용자 프로파일을 위와 같이 구성한 후 UCHM 방법을 적용시킨 경우와 장르 정보만을 이용한 ICHM 방법의 성능을 비교하여 보았다. 실험 결과, ICHM 방법이 UCHM 방법보다 좋음을 알 수 있다. 또한, 내용정보를 이용하지 않는 기존 항목기반 협력 필터링 방법이 UCHM 방법보다 좋음을 알 수 있다. 이는 [24]에서 보인 바와 같이 항목기반 방법이 사용자기반 방법 보다 우수하다는 결과와 일맥 상통한다. 하지만, 이런 결과는 사용자 프로파일을 간단한 방법을 사용하여 자동으로 구성한 경우의 결과다. 사용자가 직접 작성한 경우에는 결과가 달라 질 수 있다.

5. 결 론

본 논문에서는 현재 추천 시스템에서 성공적으로 사용되는 협력필터링 방법의 문제점에 대한 부분적인 해결책으로 제시된 ICHM 방법의 특성과 성능을 분석하여 보았다. ICHM에서는 항목의 내용을 협력필터링 방법에 반영하기 위해 항목의 내용을 근거로 유사 항목끼리 클러스터를 구축한 후 각 항목이 각 클러스터에 속할 정도를 구하고 이 정보를 협력 필터링 방법에서 사용하였다. 다양한 실험을 통하여 이러한 접근 방법이 내용 정보를 보다 효과적으로 사용함을 확인할 수 있었다. 특히, 새로운 항목에 대해서 ICHM 방법이 유용함을 확인할 수 있었다. 기존 순수 협력 필터링 방법에서는 새로운 항목에 대해서는 평가정보가 존재하지 않기 때문에 협력 예측 방법을 적용시키기가 곤란하다. 하지만, ICHM 방법을 사용할 경우는 항목 내용에 대한 정보를 이용함으로써 새로운 항목인 경우도 사용자의 취향에 맞춰 추천 받을 수 있게 되어 초기부터 원하는 항목들을 추천 받을 가능성이 높아지게 된다.

ICHM 방법이 유용하기 위해서는 영화, 음악 등 비문서 항목에 대해서 내용에 대한 충분한 텍스트 형태의 정보가 주어져야 한다. 최근에는, 인터넷의 도움으로 이러한 정보들이 상업적 목적이든 개인 취미든 다양하게 제공되고 있어 그 적용 가능성을 높여 주고 있다. 앞으로는 다양한 소스로부터 제공되는 정보들을 이용하여 항목의 내용을 보다 정확하게 표현하는 방법이 필요하리라 본다. ICHM 방법은 내용정보를 추가로 이용함으로써 기존 순수협력 필터링 방법의 문제점인 초기 평가 문제와 평가 데이타의 희박성 문제를 어느 정도 완화시킬 수 있다. 하지만, 여전히 확장성 문제로부터 자유로울 수가 없다. 따라서, 기존의 확장성 해결 방안들을 ICHM 방법에 적용하는 추가의 연구가 필요하다.

참 고 문 헌

[1] P.G. Anick, J.D. Brennan, R.A. Flynn, D.R. Hanssen, B. Alvey and J.M. Robbins, "A Direct

Manipulation Interface for Boolean Information Retrieval via Natural Language Query," *Proc. of ACM-SIGIR Conf.*, pp.135-150, 1990.

[2] J.H. Lee, M.H Kim. and Y.H. Lee, "Ranking documents in thesaurus-based Boolean retrieval systems," *Information Processing and Management*, Vol.30, No.1, pp.79-91, 1993.

[3] J. Verhoeff, W. Goffman and J. Belzer, "Inefficiency of the use of the boolean functions for information retrieval systems," *Communications of the ACM*, Vol.4, pp.557-558, 1961.

[4] G. Salton and C. Buckley, "Term-weight approaches in automatic retrieval," Information Processing and Management, Vol.24, No.5, pp.513-523, 1988.

[5] S.E. Robertson and K. Sparck Jones, "Relevance weighting of search terms," Journal of the American Society for Information Science, Vol.27, No.3, pp.129-146, 1976.

[6] M. Kim and V.V. Raghavan, "Adaptive concept-based retrieval using a neural network," *Proc. Of ACM-SIGIR Workshop on Mathematical/Formal Methods in IR*, 2000.

[7] Y. Ogawa, T. Morita and K. Kobayashi, "A fuzzy document retrieval system using the keyword connection matrix and a learning method," *Fuzzy sets and Systems*, Vol.39, pp.163-179, 1991.

[8] Douglas B. Terry, "A tour through tapestry," *Proc. of the ACM Conference on Organizational Computing Systems (COOCS)*, pp. 21-30, 1993.

[9] 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.

[10] 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.

[11] 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.

[12] http://www.cs.umn.edu/research/GroupLens/.

[13] 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.

[14] Hauver, D. B. and French,J. C, "Flycasting: Using Collaborative Filtering to Generate a Play list for Online Radio," *Proc. of Int. Conf. on Web Delivery of Music*, 2001.

[15] M. Claypool, A. Gokhale, T. Mirana, P. Murnikv, 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.

[16] 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.

[17] 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.

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

[19] 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.

[20] 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.

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

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

[23] 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, April 2004.

[24] B. Sarwar, G. Karypis,, J. Konstan, and J. Riedl, "Item-based Collaborative Filtering Recommendation Algorithms," *Proc. of WWW10*, 2001.

[25] Q. Li and B. M. Kim, "An Approach for Combining Content-based and Collaborative Filters," *Proc. of the Sixth International Workshop on Information Retrieval with Asian Languages*, pp. 17-24, 2003.

[26] Q. Li and B. M. Kim, "Clustering Approach for Hybrid Recommender System," *Proc. of the 2003 IEEE/WIC International Conference on Web Intelligence (WI 2003)*, pp. 33-38, 2003.

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

[28] R.O. Duda, P.E. Hart and D.G. Stork, *Pattern Classfication*, Wiley-Interscience Publication, New

York, pp.528-530, 2000.
[29] M. O'Connor and J. Herlocker, "Clustering items for collaborative filtering," tech. rep., University of Minnesota, department of Computer Science, Minneapolis, USA, 2000.
[30] Sonny Han Seng Chee, Jiawei Han and Ke Wang, "RecTree: An Efficient Collaborative Filtering Method," *Lecture Notes in Computer Science*, Vol. 2114, pp.141-151, 2001.

김 병 만
정보과학회논문지 : 소프트웨어 및 응용
제 31 권 제 3 호 참조

이 경
정보과학회논문지 : 소프트웨어 및 응용
제 31 권 제 3 호 참조