DOI: 10.3745/KIPSTB.2009.16-B.1.55

이웃크기를 이용한 사용자기반과 아이템기반 협업여과의 결합예측 기법

최 인 복[†] · 이 재 동[††]

요 약

협업여과는 추천시스템에서 널리 사용되는 기법으로 다른 사용자의 평가를 기반으로 아이템을 추천하는 기법이다. 사용자 데이터베이스를 이용하는 메모리기반 협업여과에는 사용자기반 기법과 아이템기반 기법이 있다. 사용자기반 협업여과는 유사한 선호도를 가지는 이웃사용자들의 선호도를 바탕으로 특정 아이템에 대한 선호도를 예측하는 반면, 아이템기반 협업여과는 아이템들의 유사도를 바탕으로 특정 사용자의 선호도를 예측한다. 본 논문에서는 추천의 성능을 향상시키기 위하여 이웃사용자와 이웃아이템 크기의 비율을 가중치로 하여 사용자기반 예측값과 아이템기반 예측값을 결합함으로써 최종 예측값을 생성하는 결합예측기법을 제안한다. MovieLens 데이터 셋과 BookCrossing 데이터 셋을 이용한 실험을 통해 본 논문에서 제안한 결합예측기법이 영화와 책에 대하여 사용자기반과 아이템기반보다 예측의 정확성을 향상시킴을 보인다.

키워드 : 추천 시스템, 메모리기반 협업여과, 결합예측

A Combined Forecast Scheme of User-Based and Item-based Collaborative Filtering Using Neighborhood Size

In-Bok Choi[†] · Jae-Dong Lee[††]

ABSTRACT

Collaborative filtering is a popular technique that recommends items based on the opinions of other people in recommender systems. Memory-based collaborative filtering which uses user database can be divided in user-based approaches and item-based approaches. User-based collaborative filtering predicts a user's preference of an item using the preferences of similar neighborhood, while item-based collaborative filtering predicts the preference of an item based on the similarity of items. This paper proposes a combined forecast scheme that predicts the preference of a user to an item by combining user-based prediction and item-based prediction using the ratio of the number of similar users and the number of similar items. Experimental results using MovieLens data set and the BookCrossing data set show that the proposed scheme improves the accuracy of prediction for movies and books compared with the user-based scheme and item-based scheme.

Keywords : Recommender Systems, Memory-Based Collaborative Filtering, Combined Forecast

1. 서 론

최근 인터넷의 발달로 다양한 정보의 홍수 속에서 소비자가 원하는 아이템을 찾는 것이 점차 어려워지고 있다. 따라서 소비자가 원하는 아이템을 찾아내고, 찾아낸 아이템을 순서대로 순위화 해주는 추천 시스템의 중요성이 부각되고 있다[9]. 추천시스템은 책, CD, 영화, 뉴스 등에서 활용되고 있으며, 여행, 쇼핑, 음식점 등 다양한 범위의 실생활에서도 필요성이 증대되고 있다[7,12]. 추천시스템에서 사용자에게 불필요한 정보를 제거하여 필요한 정보만을 제공해주는 기술을 정보 여과(information filtering)라고 하며[7], 정보 여과 방법에 따라 추천시스템은 크게 내용기반여과(content-based filtering)와 협업여과(collaborative filtering)로 분류된다. 협업여과기반 추천시스템은 다른 사용자들의 평가를 기반으로 추천을 생성하며[9], Amazon.com, Moviefinder.com, Reel.com 등과 같은 많은 인터넷 웹사이트에서 성공적으로 활용되고 있다[13,14,16]. 협업여과기반 추천시스템은 수학적 모델을 이용하는 모델기반과 사용자 데이터베이스를 활용하는 메모리기반으로 분류되며[11], 메모리기반은 사용자기반 협업여과와 아이템기반 협업여과로 분류된다[18-20].

사용자기반 협업여과 기법은 성향이 유사한 사용자는 동

※ 본 연구는 2007학년도 단국대학교 대학연구비 지원으로 연구되었음.
† 준 회 원 : 단국대학교 컴퓨터과학 및 통계학과 박사과정
†† 정 회 원 : 단국대학교 컴퓨터학부 교수
논문접수 : 2008년 11월 13일
수 정 일 : 1차 2008년 12월 11일
심사완료 : 2008년 12월 15일

일한 아이템에 대하여 유사한 선호도를 가진다는 개념에서 출발한다[8]. 유사한 선호도를 가지는 사용자 그룹을 선정하고 선정된 그룹의 선호도를 바탕으로 해당 사용자의 특정 아이템에 대한 선호도를 예측한다. 아이템기반 협업여과 기법은 사용자는 이미 구매했던 과거 아이템과 비슷한 아이템을 구매할 가능성이 높다는 개념에서 출발한다[10]. 즉, 사용자는 유사한 아이템에 대하여 유사한 선호도를 가질 가능성이 높다. 아이템기반 협업여과 기법에서는 사용자가 기존에 평가한 각각의 상품들과 선호도를 예측하고자 하는 상품의 상관관계를 이용하여 선호도를 예측한다.

그러나 사용자기반과 아이템기반 협업여과는 각각 한계점을 가지고 있다. 사용자기반 협업여과는 두 사용자가 공통적으로 선호도를 표시한 아이템만을 이용하여 유사도를 구하고 이를 기반으로 예측을 수행하기 때문에 두 사용자가 공통적으로 선호도를 표시한 아이템이 적을 경우에는 두 사용자의 성향이 비슷할지라도 정확한 유사도를 구하기 어려우며, 사용자의 평가 자료가 부족하여 예측의 정확성을 떨어뜨리는 희소성문제를 발생시킬 가능성이 높다[10]. 아이템기반 협업여과는 사용자간의 유사도를 고려하지 않고 상품간의 유사도만을 고려하기 때문에 선호도가 비슷하지 않은 사용자들의 평가를 기반으로 예측을 수행할 경우 예측의 성능이 저하될 수 있다[3].

최근에는 사용자기반과 아이템기반 협업여과의 각각의 한계점을 극복하기 위하여 사용자기반과 아이템기반 협업여과를 결합한 추천기법들이 연구되었다. Rong Hu 등은 희소성 데이터에 대하여 예측값들의 가중치 평균(weighted average)을 이용한 스무딩(smoothing) 기법으로 사용자기반과 아이템기반을 결합하는 방법을 제안하였으며[20], Jun Wang 등은 유사도 퓨전(similarity fusion) 기법을 제안하였다[17,18]. 하지만, 위의 기법들은 주로 희소성(sparsity) 문제를 해결하기 위하여 제안되었고 이러한 데이터를 기반으로 실험되었기 때문에 희소성 문제가 크지 않은 데이터에 대한 성능 측정 결과가 미흡하다. 또한, Jun Wang 등이 제안한 기법은 MovieLens 데이터 셋을 기반으로 한 실험을 통해 경험적으로 얻은 가중치 λ=0.7을 고정값으로 이용하기 때문에 데이터 셋이 변할 경우, 실험을 통해 가중치를 재계산해야 하므로 유연성(flexibility)이 부족하다.

따라서 본 논문에서는 이웃사용자와 이웃아이템의 크기의 비율로 생성된 가중치(weight)를 사용자기반 예측값과 아이템기반 예측값에 적용하여 결합함으로써 유연성있게 정확성 높은 예측값을 생성하는 결합예측기법을 제안한다. 또한, MovieLens 데이터 셋과 BookCrossing 데이터 셋에 대하여 제안된 결합예측기법이 사용자기반과 아이템기반보다 예측성능이 우수함을 보이도록 한다.

본 논문은 다음과 같이 구성된다. 2장에서는 사용자기반 협업여과와 아이템기반 협업여과의 차이점에 대하여 알아보고, 협업여과 추천기법의 일반적인 절차와 각 단계에서 활용되는 대표적인 기존 연구들에 대하여 알아본다. 3장에서는 사용자기반과 아이템기반을 이용한 결합예측기법을 제안하고, 4장에서는 본 논문에서 제안한 결합예측기법의 성능을 실험을 통하여 평가한다. 마지막으로 5장에서는 결론 및 향후 연구방향에 대하여 서술한다.

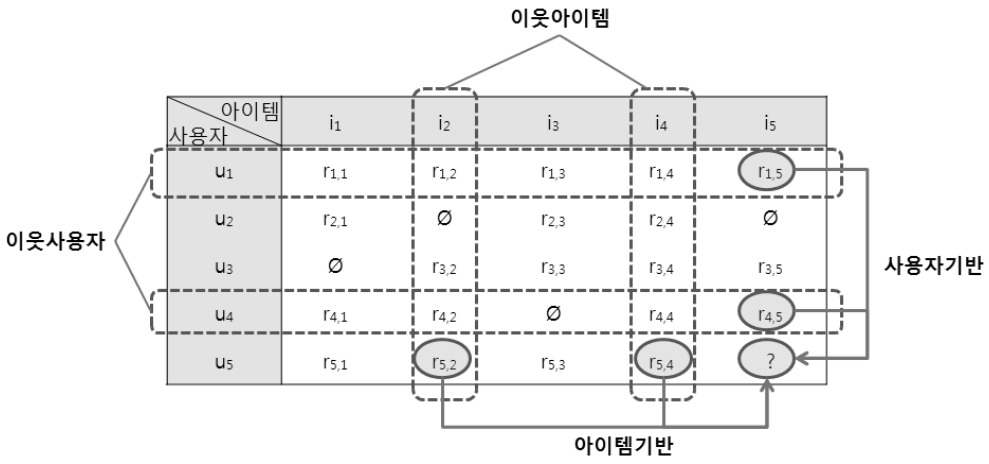
2. 관련 연구

본 장에서는 사용자기반 협업여과와 아이템기반 협업여과의 차이점에 대하여 설명하고, 사용자기반 협업여과를 기준으로 협업여과 추천의 절차와 각 단계별 기존의 연구들에 대하여 알아본다.

2.1 사용자 기반과 아이템기반 협업여과 추천기법

협업여과기법은 일반적으로 사용자, 아이템 그리고 평점 데이터를 활용한다. 사용자, 아이템, 평점 집합은 각각 $U=\{u_x \mid x=1,2, ..., m\}$, $I=\{i_x \mid x=1,2, ..., n\}$, $R=\{r_{u,i} \mid u \in U \text{ and } i \in I\}$으로 정의될 수 있다. 이 집합들은 (그림 1)과 같이 사용자-아이템 매트릭스에 표현되고 예측값 생성 및 추천에 활용된다.

사용자기반 협업여과와 아이템기반 협업여과의 차이를 (그림 1)을 예로 설명하면 다음과 같다. 사용자 u_5의 아이템 i_5에 대한 선호도 $r_{5,5}$를 예측하고자 할 때, 사용자기반 협업



(그림 1) 사용자기반 협업여과와 아이템기반 협업여과

여과에서는 먼저 사용자 u_5와 유사한 사용자 u_1과 u_4를 이웃으로 선정하고, 이들이 평가한 아이템 i_5에 대한 평가값 $r_{1,5}$와 $r_{4,5}$를 활용하여 $r_{5,5}$의 값을 예측한다. 아이템기반 협업여과에서는 아이템 i_5와 유사한 아이템 i_2와 i_4를 이웃으로 선정하고, 해당 아이템에 대한 u_5의 평가값 $r_{5,2}$와 $r_{5,4}$를 이용하여 $r_{5,5}$의 값을 예측한다.

2.2 협업여과 추천기법의 절차

협업여과 추천기법은 일반적으로 유사도 측정, 이웃 선정, 예측값 생성 단계로 이루어진다. 사용자기반과 아이템기반은 앞에서 설명한 것과 같이 각 단계에서의 기준이 사용자인지 아이템인지만 다를 뿐 활용되는 방법들은 동일하다. 따라서 여기에서는 사용자기반 협업여과를 기준으로 각 단계별 주요 기법들에 대하여 설명한다.

2.2.1 유사도 측정

유사도 측정 단계에서는 예측하고자 하는 평점 $r_{u,i}$에 대하여, 사용자 u와 다른 사용자와의 유사도 구한다. 협업여과 추천시스템에서 사용자들 사이의 유사도를 구하는 방법은 매우 다양하다. 대표적인 방법으로는 유클리드거리, 코사인 유사도, 상관관계가 있다[12].

두 사용자간의 유사도는 두 사용자가 아이템들에 대해 평가한 평점을 기반으로 계산된다. 하지만, 두 사용자가 평가한 아이템의 개수나 항목이 다를 수 있다. 따라서 많은 연구에서는 두 사용자가 공통으로 평가한 아이템만을 고려하여 유사도를 측정하였다. 두 사용자 u_a와 u_b가 공통으로 평가한 아이템의 집합을 $RI(a,b)=\{i \mid i \in I \text{ and } r_{a,i} \neq \emptyset \text{ and } r_{b,i} \neq \emptyset\}$라고 할 때, 유클리드거리, 코사인유사도, 피어슨상관계수를 이용한 유사도는 다음 식(1)과 같이 정의된다.

$$\text{(a) } sim(a,b) = \sqrt{\sum_{i \in RI(a,b)} (r_{a,i} - r_{b,i})^2}$$

$$\text{(b) } sim(a,b) = \cos(\vec{a}, \vec{b}) = \frac{\sum_{i \in RI(a,b)} (r_{a,i} \times r_{b,i})}{\sqrt{\sum_{i \in RI(a,b)} r_{a,i}^2} \times \sqrt{\sum_{i \in RI(a,b)} r_{b,i}^2}} \quad (1)$$

$$\text{(c) } sim(a,b) = \frac{\sum_{i \in RI(a,b)} (r_{a,i} - \overline{r_a})(r_{b,i} - \overline{r_b})}{\sqrt{\sum_{i \in RI(a,b)} (r_{a,i} - \overline{r_a})^2 \times \sum_{i \in RI(a,b)} (r_{b,i} - \overline{r_b})^2}}$$

유클리드거리(1a)는 두 사용자의 평점을 벡터로 표시하여 유클리드 거리를 구한다. 코사인유사도(1b)는 두 평점을 벡터로 표시하여 코사인각도를 구하는 방식으로 두 벡터의 상대적인 크기의 차이에 크게 영향을 받지 않는다. 피어슨상관계수(1c)는 두 데이터간의 관련성을 구하는 것으로 -1에서 1사이의 값으로 두 데이터간의 유사도를 파악한다. 상관관계를 이용한 방법에는 피어슨 상관계수 외에도 스피어만 상관계수가 있는데, MovieLens 데이터 셋에서는 두 방법의 성능차이가 미미한 것으로 나타났다[16].

2.2.2 이웃 선정

이웃 선정 단계에서는 사용자와 높은 유사도의 이웃을 선정한다. 일반적으로 선정된 이웃들의 수가 많을수록 예측값의 정확도는 높아지는 경향이 있지만, 계산의 복잡도가 증가하는 단점이 있다. 또한, 이웃의 수가 어느 이상으로 늘어나면 의미없는 이웃으로 인하여 오히려 예측의 정확도가 저하되는 경우도 발생한다[21]. 따라서 이웃 선정에 있어서 적정 크기의 이웃을 선정하는 것이 중요하다[10,16].

K-근접이웃(k-nearest neighbor), 유사도임계값(threshold), 클러스터링(clustering), 베이지안 네트워크(bayesian networks) 등 매우 다양한 이웃 선정 기법들이 제안되었다. 다양한 이웃 선정 기법들 중에서 K-근접이웃과 유사도임계값이 많이 사용되고 있다[3,4]. K-근접이웃은 유사도가 높은 순서대로 앞의 K명을 이웃으로 선정하는 것이고, 유사도임계값은 유사도가 특정 임계값보다 큰 사용자 모두를 이웃으로 선정하는 것이다[4]. K-근접이웃 기법은 K 값이 클 경우 유사도가 낮은 사용자가 이웃으로 선정되어 예측성능을 저하시킬 수 있고, 유사도임계값은 임계값 이상의 사용자가 많을 경우 계산의 복잡도가 증가하는 어려움이 있다. 따라서 이웃 선정 단계에서는 상황에 따라 적합한 기법을 선택하여 사용할 필요가 있다.

본 논문에서는 사용자 u_a의 이웃사용자들의 집합을 $SU(a)$라고 표시한다. 유사도임계값에 의해 선정된 이웃사용자들의 집합을 $SU(a)=\{u_b \mid u_b \in U \text{ and } sim(a,b) \geq \text{threshold}\}$으로 정의하고, K-근접이웃 기법에 의해 선정된 이웃사용자들의 집합을 $SU(a)=\{u_b \mid u_b \in U \text{ and } sim(a,b) \geq \text{similarity of the } k^{th}\text{-nearest neighbor}\}$로 정의한다.

2.2.3 예측값 생성

예측값 생성 단계에서는 선택된 이웃들의 평점을 기반으로 사용자의 평점을 예측한다. 예측값을 생성하는 대표적인 방법에는 단순평균, 가중치합, 조정가중치합이 있으며, 다음의 식 (2)와 같이 계산된다[12,16].

$$\text{(a) } upr(u,i) = \frac{\sum_{u' \in SU(u)} r_{u',i}}{n(SU(u))}$$

$$\text{(b) } upr(u,i) = \frac{\sum_{u' \in SU(u)} sim(u,u') \times r_{u',i}}{\sum_{u' \in SU(u)} |sim(u,u')|} \quad (2)$$

$$\text{(c) } upr(u,i) = \overline{r_u} + \frac{\sum_{u' \in SU(u)} sim(u,u') \times (r_{u',i} - \overline{r_{u'}})}{\sum_{u' \in SU(u)} |sim(u,u')|}$$

단순평균기법(2a)은 이웃사용자들의 해당 아이템에 대한 평가값들의 평균을 예측값으로 생성한다. 가중치합기법(2b)는 이웃사용자의 유사도와 해당 아이템에 대한 이웃사용자의 평점을 곱한 합을 유사도 절대값의 합으로 나누어 예측값을 생성한다. 조정가중치합(2c)은 이웃사용자의 아이템에 대한 평점과 이웃사용자의 평균의 차이에 대하여 유사도를

곱한 합을 유사도 절대값의 합으로 나누고, 이를 해당 사용자의 평균 평점에 합산하여 예측값을 생성한다.

3. 사용자기반과 아이템기반 협업여과를 이용한 결합예측 추천기법

본 장에서는 사용자기반의 예측값과 아이템기반의 예측값을 생성하고 선정된 이웃의 비율에 따른 가중치를 이용하여 최종 예측값을 생성하는 결합예측 추천기법에 대하여 설명한다.

3.1 유사도 측정 알고리즘

메모리기반 협업여과기법의 가장 큰 특징은 사용자-아이템 매트릭스를 기반으로 유사도 매트릭스를 작성한 후, 이를 추천에 활용 한다는 것이다. 일반적으로 유사도 매트릭스는 일정 시간간격을 두고 주기적으로 작성된다. 유사도 매트릭스의 갱신 주기가 빠를수록 더 정확한 예측이 가능한 반면, 갱신에 필요한 리소스 및 프로세싱 시간이 필요하게 된다. 본 논문에서는 사용자유사도와 아이템유사도 두 개의 매트릭스를 사용한다. 사용자기반의 사용자유사도 매트릭스와 아이템기반의 아이템유사도 매트릭스는 수식 (1)을 이용하여 작성된다. 두 개의 유사도 매트릭스를 만드는 MakeSimMatrix 알고리즘은 다음과 같다.

Algorithm MakeSimMatrix

- Input: User-Item Matrix *R*; the number of users *m*; the number of items *n*
- Output: User Similarity Matrix *USim*, Item Similarity Matrix *ISim*

```
1  for a ← 1 to m do
2        for b ← 1 to m do
3              USim(a,b) ← Sim(a,b) using formula(1)
4        end for
5  end for
6  for a ← 1 to n do
7        for b ← 1 to n do
8              ISim(a,b) ← Sim(a,b) using formula(1)
9        end for
10 end for
```

MakeSimMatrix에서는 모든 사용자에 대하여(1-2 줄), 두 사용자간의 유사도를 계산하여 사용자유사도 매트릭스(*USim*)의 각 항목에 저장한다(3줄). 마찬가지로 모든 아이템에 대하여(6-7 줄), 아이템유사도 매트릭스의 각 항목에 저장한다(8줄).

이러한 방식으로 생성되는 사용자유사도 매트릭스와 아이템유사도 매트릭스의 예는 (그림 2)와 같다. (그림 2)는 사용자-아이템 매트릭스(a)를 기반으로 피어슨 상관계수를 이용하여 작성된 사용자유사도 매트릭스(b)와 아이템유사도 매트릭스(c)의 예이다.

3.2 이웃 선정 알고리즘

이웃 선정 알고리즘에서는 예측하고자 하는 사용자와 아이템에 대하여 유사도가 높은 이웃을 선정한다. 이웃을 선정할 때, 유사도만을 고려하여 이웃을 선정하는 방식과 예측하고자 하는 사용자 또는 아이템에 대하여 평가값이 존재하는 이웃만을 선정하는 방식이 있다. 유사도만을 고려하여 이웃을 선정하는 방식은 희소성의 문제를 완화시킬 수 있는 장점이 있는 반면, 예측의 정확도가 저하될 수 있는 단점이 있다. 예측하고자 하는 사용자 또는 아이템에 대한 평가값이 존재하는 이웃만을 선정하는 방식은 선택할 이웃의 범위가 좁아짐으로 인하여 희소성 문제가 발생할 가능성이 높지만 예측의 정확도를 높일 수 있는 장점이 있다. 본 논문에서는 예측의 성능을 높이기 위하여 예측하고자 하는 사용자 또는 아이템에 대한 평점이 존재하는 사용자 또는 아이템을 이웃으로 선정한다. 따라서 이웃사용자 집합과 이웃아이템 집합을 다음과 같이 정의한다.

이웃사용자 집합 $SUR(u,i)=\{u_a \mid u_a \in U$ and $r_{a,i} \neq \varnothing$ and sim(u,a)≧threshold$\}$ 또는,

$SUR(u,i)=\{u_a \mid u_a \in U$ and $r_{a,i} \neq \varnothing$ and sim(u,a)≧similarity of the k^{th}-nearest neighbor$\}$

이웃아이템 집합 $SIR(u,i)=\{i_a \mid i_a \in I$ and $r_{u,a} \neq \varnothing$ and sim(i,a)≧threshold$\}$ 또는,

$SIR(u,i)=\{i_a \mid i_a \in I$ and $r_{u,a} \neq \varnothing$ and sim(i,a)≧similarity of the k^{th}-nearest neighbor$\}$

이렇게 유사도 매트릭스와 사용자-아이템 매트릭스를 이용하여 이웃사용자와 이웃아이템을 선정하여 각각의 배열을 생성하는 MakeNeighbors 알고리즘은 다음과 같다.

(a) User-Item Matrix

User \ Item	i1	i2	i3	i4	i5
u1	1	4	4	3	4
u2	3	Ø	5	1	1
u3	Ø	4	4	3	3
u4	3	5	Ø	4	4
u5	2	5	5	3	?

(b) USim Matrix

User \ User	u1	u2	u3	u4	u5
u1	1	0	0.57	0.86	0.94
u2	0	1	1	-1	0.65
u3	0.57	1	1	1	1
u4	0.86	-1	1	1	0.98
u5	0.94	0.65	1	0.98	1

(c) ISim Matrix

Item \ Item	i1	i2	i3	i4	i5
i1	1	0.86	0.86	-0.20	-0.50
i2	0.86	1	1	0.57	0.50
i3	0.86	1	1	-0.57	-0.94
i4	-0.20	0.57	-0.57	1	0.93
i5	-0.50	0.50	-0.94	0.93	1

(그림 2) 유사도 매트릭스 생성 예

Algorithm MakeNeighbors

- Input: User-Item Matrix *R*; Similarity Matrix *USim* and *ISim*;
  threshold (of kth nearest neighbor) *th*; user *u*; item *i*
- Output: User's Neighbors Array *UserNeighbors*;
  Item's Neighbors Array *ItemNeighbors*

```
1  for a ← 1 to m do
2        if USim(u,a)≧th and R(a,i)≠∅ then
3                Insert a into UserNeighbors
4        end if
5  end for
6  for b ← 1 to n do
7        if ISim(i,b)≧th and R(u,b)≠∅ then
8                Insert b into ItemNeighbors
9        end if
10 end for
```

MakeNeighbors 알고리즘에서는 이웃사용자를 선정하기 위하여 사용자유사도가 임계값(threshold) 이상이고 예측하고자 하는 아이템에 대하여 평점이 존재하는지 판별하여(2번 줄) 두 조건을 모두 만족하는 사용자를 이웃사용자로 선정한다(3번 줄). 이웃아이템을 선정하는 방법도 마찬가지로 아이템유사도가 임계값 이상이고 예측하고자 하는 사용자가 평점을 부여했을 경우(7번 줄), 이웃아이템으로 선정한다(8번 줄).

3.3 사용자기반과 아이템기반 결합예측 알고리즘

결합예측은 예측의 정확도를 높이기 위하여 여러 예측방법을 결합하여 하나의 예측값을 구하는 것이다[2,5,6]. 서로 다른 n개의 개별예측을 p_i(i=1,2,...n)라 할 때, 결합예측값 P는 다음 수식(3)과 같이 구해진다.

$$P = \sum_{i=1}^{n} \omega_i p_i \tag{3}$$

결합예측 연구들의 중점은 여러 예측들을 선형적으로 결합하는 것으로 이때의 가중치(ω_i)를 구하는 것이다[6]. 대표적인 결합예측기법으로는 예측값들의 평균에 의한 단순평균방법, 결합예측오차의 분산최소화 방법(VCM), 회귀분석방법에 의한 가중치 선정방법(RBM), 계열상관을 고려한 가중치 선정방법(SCM), 베이지안 방법 등이 있다[5]. 기존의 예측결합기법들은 이론적인 장점은 있지만, 대부분 시계열 데이터에 대한 예측방법이기 때문에 기존의 협업여과 추천기법에는 적용의 어려움이 있다. 따라서 본 논문에서는 예측값에 참여한 이웃사용자와 이웃아이템의 비율을 가중치로 하는 결합예측기법을 제안한다. 통계학적으로 표본의 수가 많을수록 예측값의 성능이 향상되는 경향이 있다. 따라서 예측값에 참여한 이웃의 수가 많을수록 신뢰성 높은 예측값을 생성할 가능성이 높다고 할 수 있다. 이웃사용자와 이웃아이템의 비율을 가중치로 하여 사용자기반과 아이템기반의 예측값을 결합하는 최종 예측값은 수식 (4)와 같이 구하며, 가중치 ω는 예측값 생성에 참여한 이웃의 비율로써 수식 (5)와 같이 계산된다.

$$cpr(u,i) = \omega \times upr(u,i) + (1-\omega) \times ipr(u,i) \tag{4}$$

$$\omega = \frac{n(SUR(u,i))}{n(SUR(u,i)) + n(SIR(u,i))} \tag{5}$$

수식 (4)에서 *cpr(u,i)*은 최종 예측값이고, *upr(u,i)*은 사용자기반 예측값이며, *ipr(u,i)*은 아이템기반 예측값이다. 수식 (5)에서 *n(SUR(u,i))*는 예측값 생성에 참여한 이웃사용자의 수이며, *n(SIR(u,i))*는 이웃아이템의 수이다. 이렇게 사용자기반과 아이템기반의 예측값을 결합하여 최종 예측값을 생성하는 CombinedForecast알고리즘은 다음과 같다.

Algorithm CombinedForecast

- Input: User-Item Matrix *R*; User Neighborhood Array *UserNeighbors*;
  Item Neighborhood Array *ItemNeighbors*; user *u*; item *i*
- Output: Predicted Rating Value $P_{u,i}$

```
1  un ← size of UserNeighbors
2  in ← size of ItemNeighbors
3  ω ← un/(un+in)
4  for a ← 1 to un do
5        calculate upr(u,i) using formula (2)
6  end for
7  for a ← 1 to in do
8        calculate ipr(u,i) using formula (2)
9  end for
10 cpr(u,i) = ω×upr(u,i) + (1-ω)×ipr(u,i)
```

CombinedForecast 알고리즘에서는 먼저 이웃사용자와 이웃아이템의 비율로 가중치 값을 결정한다(1-3번 줄). 사용자기반의 예측값을 생성하고(4-6번 줄), 아이템기반의 예측값을 생성한 후(7-9번 줄), 가중치를 적용하여 최종 예측값을 생성한다(10번 줄).

4. 성능 평가

본 장에서는 3장에서 제안한 결합예측기법이 사용자기반 및 아이템기반의 예측기법보다 예측 성능이 우수함을 보이기 위하여 MovieLens 데이터 셋과 BookCrossing 데이터 셋을 이용하여 실험을 수행하고, 유사도임계값과 이웃의 크기에 따른 예측의 성능을 비교 평가한다.

4.1 실험 방법

본 논문에서 제안한 결합예측 기법의 성능을 평가하기 위

(그림 3) 10-fold 교차검증 데이터 집합 구성 방법

하여 GroupLens Research Group에서 수집한 MovieLens 데이터 셋과 IIF(Institut fur Informatik Freiburg)에서 수집한 BookCrossing 데이터 셋을 이용한다. MovieLens 데이터 셋은 1997년 9월부터 1998년 4월까지 943명을 대상으로 1,682편의 영화에 대하여 총 100,000개의 평가 레코드로 구성되어 있다[21]. BookCrossing 데이터 셋은 2004년 8월에서 9월까지 4주간 278,858명을 대상으로 271,379편의 책에 대하여 총 1,149,780개의 평가 레코드로 구성되었다[22]. BookCrossing 데이터 셋은 MovieLens 데이터 셋에 비하여 데이터 셋의 크기가 크고 희소성(Sparsity)이 심하기 때문에 일부 데이터를 선별하여 사용하였다. 데이터 선별 방법은 다음과 같다. 먼저, 대상자 중 명시적 평가(explicit rating)가 많은 상위 50명을 선별하고, 이들 50명이 5회 이상 명시적으로 평가한 171개의 책에 대하여 1,000개의 평가 데이터를 선별하였다.

본 연구에서는 추천기법의 성능 평가에 있어서 학습 데이터로부터의 영향을 최소화하고 신뢰성을 확보하기 위하여 10-fold 교차검증(cross validation)을 수행한다. K-fold 교차검증은 전체집합을 K개로 나눈 후, 1개를 실험집합으로 하고 나머지를 학습집합으로 만든 후 K개의 세트로 실험 진행 후 평균값으로 분류 결과를 검증하는 기법이다. 또한, 희소성 문제가 적은 데이터로 구성하기 위하여 한명의 사용자 rating들을 각 fold에 분산하여 배치한다.

본 논문에서 구성한 10-fold 교차검증 데이터 집합은 (그림 3)과 같다. 먼저 rating 테이블을 사용자-아이템 순으로 정렬하고, 모든 레코드가 각 fold에 삽입될 때까지 하나의 레코드씩 1~10번 fold까지 차례대로 삽입한다. 10개의 fold 중 9개의 fold를 훈련(training) 데이터 집합으로 하고 1개의 fold를 평가(test) 데이터 집합으로 한다.

MovieLens 데이터 셋은 아이템에 대하여 1~5의 평가값이 부여되어 있고 BookCrossing 데이터 셋은 아이템에 대하여 1~10의 평가값이 부여되어 있으므로, 본 논문에서는 예측값의 정확도를 평가하기 위하여 MAE(Mean Absolute Error)를 사용한다. MAE는 사용자가 부여한 실제 평가값과 추천 시스템의 예측값의 차이들을 평균하여 추천의 성능을 평가하는 방법으로 수식 (6)과 같다.

$$MAE = \frac{\sum_{i=1}^{n} |p_i - r_i|}{n} \quad (6)$$

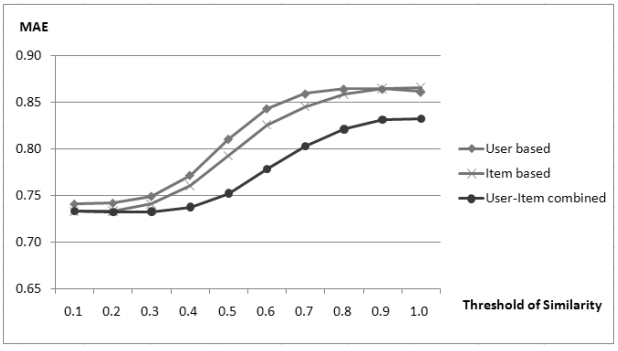
여기에서, n은 예측한 아이템의 개수이고, p_i는 시스템에 의해 생성된 예측값이며, r_i는 사용자가 평가한 평가값이다. 즉, MAE는 n개의 아이템에 대한 예측값과 평가값의 차이(error)의 평균을 의미한다.

본 논문에서는 유사도 측정 방법으로 피어슨상관계수를 이용하며, 예측값 생성 방법으로는 조정가중치합을 사용한다. 이러한 방법을 사용한 이유는 많은 연구에서 피어슨 상관계수와 조정가중치합이 좋은 결과를 나타내었으며, 2장의 관련연구에서 소개한 방법들 중에서 MovieLens 데이터 셋에 적용한 실험결과에서 피어슨상관계수와 조정가중치합의 조합이 가장 예측 성능이 우수했기 때문이다.

4.2 실험 결과

먼저, MovieLens 데이터 셋에 대하여 사용자기반, 아이템기반 예측과 본 논문에서 제안한 결합예측에 대한 성능을 실험하였다. 유사도임계값(threshold)을 0.1에서 1까지 0.1씩 증가시키면서 MAE값을 측정한 결과 (그림 4)와 같은 결과를 얻었다. 공통적으로 유사도임계값이 낮을수록 예측성능이 높아지는 결과를 보인다. 유사도임계값이 낮다는 것은 예측에 참여한 이웃의 수가 많다는 것이며, 이웃의 크기를 증가시킬수록 예측성능을 높일 수 있다고 해석할 수 있다. 유사도임계값이 0.2 이상에서 본 논문에서 제안한 결합예측기법이 사용자기반 예측 및 아이템기반 예측보다 우수한 성능을 보인다. 유사도임계값이 0.6일 때, 가장 큰 예측성능의 격차를 보인다. 유사도임계값이 0.6일 때, 결합예측기법이 사용자기반보다 7.6% 아이템기반보다 5.6% 예측 성능이 향상됨을 보인다.

MovieLens 데이터 셋에 대하여 K-근접이웃(k-nearest neighbor)으로 이웃을 선정한 경우의 예측성능을 평가하기 위하여 총 이웃의 크기를 10에서 50까지 10의 크기로 증가시키면서 사용자기반, 아이템기반과 결합예측기법에 대한 MAE값을 측정하였다. 결합예측에서는 이웃사용자와 이웃아

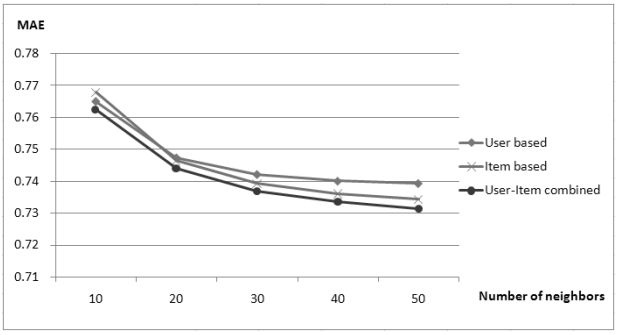


(그림 4) 유사도임계값에 따른 예측성능(MovieLens)

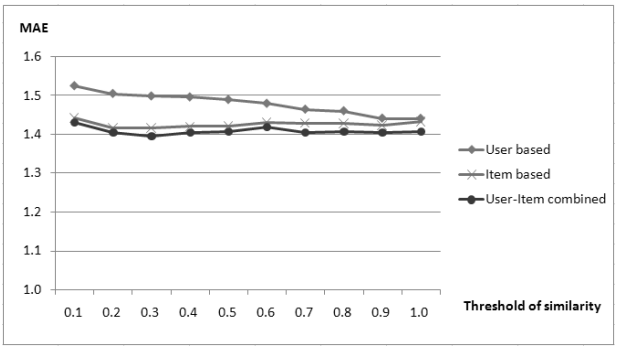
이템의 개수를 절반으로 함으로써 총 이웃의 크기를 사용자기반 및 아이템기반과 동일하도록 하였다. 이렇게 이웃의 크기에 따른 예측성능의 결과는 (그림 5)와 같다. 이 실험에서도 역시 사용자기반 및 아이템기반보다 본 논문에서 제안한 결합예측기법이 가장 우수함을 보인다. 이웃의 수가 50일 때, 결합예측기법이 사용자기반보다 1%, 아이템기반보다 0.4% 예측성능의 향상을 보인다.

다음으로, 선별된 BookCrossing 데이터 셋에 대하여 성능을 실험하였다. 유사도임계값을 0.1에서 1까지 0.1씩 증가시키면서 MAE값을 측정한 결과 (그림 6)와 같은 결과를 얻었다. 결합예측기법이 사용자기반보다 평균 4.9%, 아이템기반보다 평균 1.2% 예측성능의 향상을 보인다.

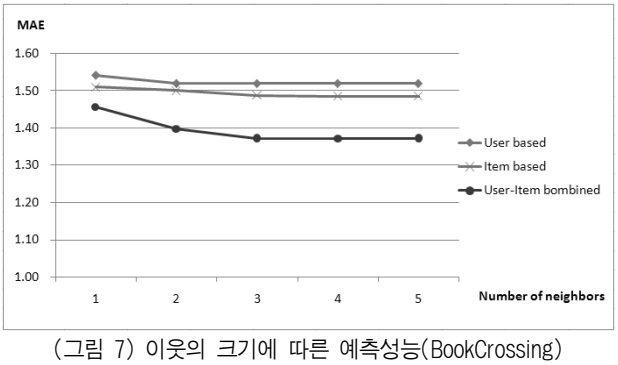
선별된 BookCrossing 데이터 셋에 대하여 K-근접이웃(k-nearest neighbor)으로 이웃을 선정한 경우의 예측성능을 평가하기 위하여 총 이웃의 크기를 1에서 5까지 1의 크기로 증가시키면서 MAE값을 측정한 결과는 (그림 7)과 같다. 결합예측기법이 사용자기반보다 평균 9.1%, 아이템기반보다 평균 7.1% 예측성능의 향상을 보인다.



(그림 5) 이웃의 크기에 따른 예측성능(MovieLens)



(그림 6) 유사도임계값에 따른 예측성능(BookCrossing)



(그림 7) 이웃의 크기에 따른 예측성능(BookCrossing)

5. 결론 및 향후 연구방향

본 논문에서는 추천의 성능을 향상시키기 위하여 사용자기반과 아이템기반 협엽여과를 결합하는 예측기법을 제안하였다. 제안된 결합예측방법은 이웃사용자와 이웃아이템의 크기 비율을 가중치로 하여 사용자기반 예측값과 아이템기반 예측값을 결합함으로써 최종 예측값을 생성하는 것이다. MovieLens 데이터 셋을 이용한 실험 결과 본 논문에서 제안한 결합예측방법이 유사도임계값에 따른 성능에서 사용자기반보다 최대 7.6% 아이템기반보다 최대 5.6% 예측 성능이 향상됨을 보였으며, 이웃의 크기에 따른 성능에서도 사용자기반보다 1%, 아이템기반보다 0.4% 예측 성능이 향상됨을 보였다. 또한, 선별된 BookCrossing 데이터 셋을 이용한 실험에서는 결합예측방법이 유사도임계값에 따른 성능에서 사용자기반보다 평균 4.9%, 아이템기반보다 평균 1.2%의 예측성능 향상을 보였으며, 이웃의 크기에 따른 성능에서도 사용자기반보다 평균 9.1%, 아이템기반보다 평균 7.1% 예측성능의 향상을 보였다. 사용자기반과 아이템기반을 결합한 예측기법은 과거의 연구들에서 희소성 문제에 효과적임을 보였으며, 본 연구를 통하여 희소성 문제가 적은 데이터에 대하여 예측의 성능을 향상시킬 수 있음을 증명하였다. 향후에는 더욱 효과적으로 사용자기반과 아이템기반을 결합할 수 있는 예측방법에 대하여 연구할 것이며, 다양한 데이터 셋을 활용하여 예측성능을 확인할 계획이다.

참 고 문 헌

[1] 고수정, 김진수, 김태용, 최준혁, 이정현, “협력적 여과와 내용기반 여과의 병합을 통한 추천 시스템에서의 사용자 선호도 발견”, 정보과학회논문지 제7권 제6호, Dec. 2001.

[2] 김연형, “결합예측에 관한 연구”, 응용통계연구 제1권 제2호, pp.111-124, 1986.

[3] 박지선, 김택헌, 류영석, 양성봉, “추천 시스템을 위한 2-way 협동적 필터링 방법을 이용한 예측 알고리즘”, 정보과학회논문지:소프트웨어및응용 제29권 제9호, pp.669-675, Oct. 2002.

[4] 이용준, 이세훈, 왕창종, “인구 통계 정보를 이용한 협업 여과 추천의 유사도 개선 기법”, 정보과학회논문지:컴퓨팅의실제 제9권 제5호, pp.521-529, Oct. 2003.

[5] 이우리, “결합예측 방법에 의한 종합주가지수의 예측”, 논문집 제44권 제1호, pp.309-333, 2000.

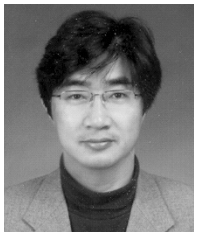
[6] 이태희, 김홍재, “AHP와 ANP의 결합을 통한 합리적 예측모델 구축”, 한국경영과학회 학술대회 논문집 제2호, pp.229-232, 1997.

[7] 이형동, 김형주, “협업 필터링 추천시스템에서의 취향 공간을 이용한 평가 예측 기법”, 정보과학회논문지:데이터베이스 제34권 제5호, Oct. 2007.

[8] 지애띠, 연철, 이승훈, 김홍남, 조근식, “분산 환경에서의 협력적 여과를 위한 멀티 에이전트 프레임워크”, 한국지능정보시스템학회논문지 제13권 제3호, pp.119-140, Sep. 2007.
[9] 최인복, 박태근, 이재동, “소비자의 감성과 소비유형을 이용한 협업여과기반 콘텐츠 추천 기법”, 정보처리학회논문지D, 제15-D권 제3호, June 2008.
[10] Badrul Sarwar, George Karypis, Joseph Konstan, John Riedl, “Item-based Collaborative Filtering Recommendation Algorithms,” Proceedings of the 10th International Conference on World Wide Web, Apr. 2001.
[11] Breese, J., Heckerman, D. and Kadie, C., “Empirical Analysis of Prediction Algorithms for Collaborative Filtering,” Proc. of the 14th Conference on Uncertainly in Artificial Intelligence, pp.43-52, 1998.
[12] Gediminas Adomavicius and Alexander Tuzhilin, “Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions,” IEEE Transactions on Knowledge and Data Engineering, Vol.17, Issue 6, pp.737-749, Apr. 2005.
[13] Greg Linden, Brent Smith and Jeremy York, “Amazon.com recommendations: item-to-item collaborative filtering,” Internet Computing, IEEE, Vol.7, Issue 1, pp.76-80, Jan. 2003.
[14] John Benjamin Schafer, Joseph Konstan, John Riedl, “Recommender systems in e-commerce,” Proceedings of the 1st ACM conference on Electronic commerce, Nov. 1999.
[15] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen and John T. Riedl, “Evaluating collaborative filtering recommender systems,” ACM Transactions on Information Systems (TOIS), Vol.22, Issue 1, pp.5-53, Jan. 2004.
[16] Jonathan L. Herlocker, Joseph A. Konstan, Al Borchers and John Riedl, “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, Aug. 1999.
[17] Jun Wang, Arjen P. Vries and Marcel J.T. Reinders, “Unified Relevance Models for Rating Prediction in Collaborative Filtering,” ACM Transactions on Information Systems, Vol.26, No.3, June 2008.
[18] Jun Wang, Arjen P. de Vries and Marcel J.T. Reinders, “Unifying User-based and Item-based Collaborative Filtering Approaches by Similarity Fusion,” ACM SIGIR ’06, Aug. 2006.
[19] Manos Papagelis and Dimitris Plexousakis, “Qualitative analysis of user-based and item-based prediction algorithms for recommendation agents,” Engineering Applications of Artificial Intelligence 18, pp.781-789, June 2005.
[20] Rong Hu and Yansheng Lu, “A Hybrid User and Item-Based Collaborative Filtering with Smoothing on Sparse Data,” ICAT ’06, pp.184-189, Nov. 2006.
[21] http://www.grouplens.org/system/files/ml-data_0.zip
[22] http://www.informatik.uni-freiburg.de/~cziegler/BX/

최 인 복

e-mail : pluto612@dku.edu
1999년 단국대학교 전자계산학과 (학사)
2002년 단국대학교 대학원 전자계산학과 (석사)
2002년~현 재 단국대학교 대학원 컴퓨터과학및통계학과(박사과정)
관심분야 : 유비쿼터스 컴퓨팅, 콘텐츠 적응화, (모바일)인터넷 기술, 분산/병렬 처리



이 재 동

e-mail : letsdoit@dankook.ac.kr
1985년 인하대학교 전자계산학(학사)
1991년 Cleveland State University(석사)
1996년 Kent State University(박사)
1997.3~현 재 단국대학교 컴퓨터학부 컴퓨터공학전공 교수
2006.4~현 재 국가지정 CT연구소 소장
2005.3~현 재 단국대학교 콘텐츠&컨버전스기술연구소 소장
2004.7~2006.6 단국대학교 정보통신원 원장(C.I.O)
2002.11~현 재 농협중앙회 전산고문
2006.7~2007.12 민관확대 콘텐츠 정책 협의회 위원
2007.2~현 재 Dream economy leader 포럼 위원
2005.1~2006.12 전국대학정보화 협의회 이사
2005.8~2006.8 문화관광부 KOCCA CT포럼/전략기획 운영위원/분과위원장
2004.1~2006.6 (사)이러닝 산업협회 이사
2005.1~2006.6 교통안전공단 자문위원
관심분야 : Ubiquitous Computing, Contents & Entertainment Technologies, (Mobile) Internet Technologies/Applications, Many aspects of parallel/distributed processing