사용자 정보 가중치를 이용한 추천 기법

윤소영* · 윤성대**

A Recommendation Technique using Weight of User Information

So Young Yun* · Sung-Dae Youn**

요 약

협업 필터링은 추천시스템들 중에서 가장 널리 사용되는 기법이다. 그러나 협업 필터링은 추천의 정확성을 떨어뜨리는 희소성과 확장성의 문제를 가지고 있으며 이를 해결하기 위한 다양한 연구가 이루어지고 있다. 본 논문에서는 협업필터링의 희소성과 확장성의 문제를 해결하기 위해 가중치를 사용한 기법을 제안한다. 제안한 기법은 데이터 셋에서 추천의 정확성을 높이기 위해 평가값이 4이상인 데이터들만을 사용하여 아이템을 선호하는 사용자 정보를 분석한다. 아이템의 장르 정보와 분석한 사용자 정보를 유사도 계산 시 가중치로 사용하고 임계값 이상의 유사도를 가진 데이터들만으로 예측값을 계산하여 평가되지 않은 데이터의 평가값으로 사용한다. 제안한 기법은 아이템에 대한 특성을 분석하여 예측값을 계산함으로써 희소성을 줄임과 동시에 정확성을 더 높일 수 있고 새로운 아이템과 사용자가 등록되었을 때 분석된 정보를 바탕으로 빠른 분류가 가능하다. 실험을 통해 제안한 기법이 기존의 아이템 기반, 장르 기반 기법보다 추천의 정확성이 향상되는 것을 확인하였다.

ABSTRACT

A collaborative filtering(CF) is the most widely used technique in recommender system. However, CF has sparsity and scalability problems. These problems reduce the accuracy of recommendation and extensive studies have been made to solve these problems, In this paper, we proposed a method that uses a weight so as to solve these problems. After creating a user-item matrix, the proposed method analyzes information about users who prefer the item only by using data with a rating over 4 for enhancing the accuracy in the recommendation. The proposed method uses information about the genre of the item as well as analyzed user information as a weight during the calculation of similarity, and it calculates prediction by using only data for which the similarity is over a threshold and uses the data as the rating value of unrated data. It is possible simultaneously to reduce sparsity and to improve accuracy by calculating prediction through an analysis of the characteristics of an item. Also, it is possible to conduct a quick classification based on the analyzed information once a new item and a user are registered. The experiment result indicated that the proposed method has been more enhanced the accuracy, compared to item based, genre based methods.

키워드

협업 필터링, 추천 기법, 희소성, 확장성, 유사도, 가중치

Key word

Collaborative Filtering, Recommendation Technique, Sparsity, Scalability, Similarity, Weight

* 정회원 : 부경대학교(ysmallzero@pknu.ac.kr)
** 정회원 : 부경대학교 컴퓨터공학과 교수(교신저자)

접수일자 : 2010. 12. 15
심사완료일자 : 2011. 01. 13

Ⅰ. 서 론

인터넷의 보급과 e-commerce의 도입으로 인해 사용자들은 자신이 원하는 아이템에 대한 정보를 빠르게 얻을 수 있게 되었다. 그러나 다양한 정보기기들의 확산과 정보의 급격한 증가로 인해, 사용자들은 넘쳐나는 정보들 속에서 자신이 원하는 정보를 찾기 위해 이전 보다 많은 노력을 기울여야만 하게 되었다. 기업은 이러한 문제를 해결하고 더 많은 사용자를 확보하기 위해 추천시스템을 사용하고 있다. 추천시스템은 사용자들에게 그들이 관심 있고 좋아할 만한 아이템을 추천해 주어 원하는 아이템을 쉽고 빠르게 찾을 수 있도록 돕는다. 이 시스템은 온라인 뉴스, 영화, 다양한 형태의 web resource들을 추천하며 Amazon, CDNow, DangDang, Sinforyou와 같은 많은 e-commerce 사이트에서 사용되고 있다[1].

추천시스템들 중에서는 협업 필터링(collaborative filtering)이 가장 널리 사용된다. 협업 필터링은 유사한 사용자들은 유사한 성향을 가진다는 것에 기반 한다. 협업 필터링은 사용자들의 평가 정보를 사용하여 데이터베이스를 구축하고 목표 사용자와 유사한 선호도를 가진 사용자들을 데이터베이스로부터 찾아내어 이들의 선호에 기반해 새로운 평가를 예측하고 이를 목표 사용자에게 추천하는 방식이다.

협업 필터링이 e-commerce에서 성공적으로 널리 사용되고 있지만 희소성(sparsity), 확장성(scalability) 등의 문제점을 가진다. 희소성은 매우 활동적인 사용자들조차도 user-item rating 데이터베이스에서 이용 가능한 전체 아이템들 중 소수의 아이템만을 평가하기 때문에 발생한다[2]. 희소성은 추천의 정확성을 떨어뜨리는 가장 큰 요인이다. 확장성은 아이템 수와 사용자 수가 증가함에 따라 목표 사용자의 최근접 이웃을 찾기 위한 연산이 급격히 증가하는데 이에 따른 데이터 처리는 신속히 이루어 지지 못함으로써 발생하는 문제이다[3]. 희소성과 확장성의 문제를 해결하기 위해 다양한 연구들이 진행되고 있다.

본 논문에서는 추천의 정확성에 심각한 문제를 발생시키는 희소성을 줄이고 정확성을 높이기 위해 영화를 대상으로 장르 정보와 사용자들의 정보를 사용한 기법을 제안한다.

제안하는 기법은 item user matrix의 장르와 사용자들의 정보를 사용하여 아이템과 사용자의 특성을 분석하고 이 특성들을 가중치로 사용하여 유사도를 계산한다. 유사도 임계값 이상의 데이터들만으로 예측값을 계산하고 예측값 중에서 상위 N개의 값만을 사용하여 평가되지 않은 아이템에 평가값을 부여함으로써 희소성을 줄이고 추천의 정확성을 높일 수 있다. 또한 분석한 특성을 이용하여 새로운 아이템이나 사용자들에 대한 분류와 추천도 빠르게 할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구인 협업 필터링에 대해 기술하고, 3장에서는 가중치를 활용한 제안 기법에 대해 살펴본다. 4장에서는 실제 데이터를 대상으로 제안하는 기법의 성능 평가를 하고 마지막 5장에서는 결론을 기술한다.

Ⅱ. 관련 연구

2.1. 협업 필터링

협업 필터링은 1992년 Goldburg[4]에 의해 처음으로 제안되었다. 그 후 Resnick[5]은 neighbor- hood-based 알고리즘을 사용하여 평가값이 자동으로 부여된 뉴스를 사용자에게 추천하는 협업 필터링 기법을 제안했고, 지속적으로 많은 연구들이 이루어지고 있다.

협업 필터링은 메모리 기반 알고리즘과 모델 기반 알고리즘으로 분류된다. 메모리 기반 알고리즘은 사용자 간의 유사성에 기초한 방법으로 목표 사용자와 가장 유사한 사용자들을 식별하기 위해 전체 데이터베이스를 계산하여 상위 K의 유사 사용자들을 추출한 후 이들의 아이템에 대한 평가 정보로 목표 사용자에게 아이템을 추천하는 방법이다[6].

그러나 이 알고리즘은 사용자가 아이템에 부여한 평가값의 개수가 부족하여 추천의 성과가 떨어지는 문제점인 희소성과 사용자의 수와 거래데이터가 증가함에 따라 목표 사용자의 최근접 이웃을 찾기 위한 연산이 기하급수적으로 늘어난다는 확장성의 문제가 있다[4]. 메모리 기반 알고리즘의 문제점들을 해결하기 위해 평가값이 비어 있는 칸에 기본 평가값을 부여하는 알고리즘[7], 개별 사용자들 간의 유사도를 계산한 방법 대신 아이템 간의 유사도를 계산하는 알고리즘[8] 등 여러 방법

들이 제안되었다.

모델 기반 알고리즘은 평가 패턴들에 기반하여 데이터베이스의 사용자들을 작은 수의 클래스 그룹으로 묶는 기법이다. 특정 아이템에 대한 목표 사용자의 평가를 예측하기 위해 목표 사용자를 하나 또는 그 이상의 정의된 클래스로 분류시킨 후 해당 클래스의 아이템 평가 값을 사용하여 목표 사용자의 평가를 예측한다[6]. 모델 기반 알고리즘에서는 Aspect 기법[9], Bayesian network 기법[7], Clustering 기법[10], Graph model 기법[11] 등의 다양한 알고리즘이 사용된다. 그러나 모델 기반 알고리즘은 사용자 선호 모델들이 빈번하고 급속하게 업데이트되어야만 하는 환경에는 적절하지 않은 문제점이 있다[6].

2.2. 장르 기반 기법

협업 필터링의 희소성 문제를 해결하기 위해 아이템의 장르를 활용한 알고리즘들이 제안되고 있다. Ye Zhang 등에 의해 제안된 알고리즘[12]은 같은 장르에 속한 아이템이 다른 장르에 속한 아이템들보다 더 유사하다는 점에 초점을 둔 연구이다. Zhang 등의 연구[12]에서는 아이템의 장르에 기반해 이웃 아이템들의 후보를 선정한다. 평가 매트릭스를 통해 목표 아이템들과 이웃 아이템들의 후보 간에 유사도를 계산하고 근접 이웃들의 집합을 추출한다. 또한 후보 아이템 생성 시 목표 아이템을 높게 평가한 사용자들의 그룹은 목표 아이템과 장르가 같은 다른 아이템들도 높게 평가할 것이라는 점을 이용하였다.

다음은 Zhang이 제안한 알고리즘이다.

① 랜덤하게 사용자들을 입력한 후 사용자가 평가하지 않은 아이템 집합 I_{unrate}를 가져온 후 I_{unrate}의 속성이 될 목표 아이템 I_{aim}을 선택한다.

② 데이터베이스에서 목표 아이템을 높게 평가한 사용자들의 그룹을 선택한다. (사용자들의 평가 임계값 r은 4이상이다) I_{other}은 목표 사용자가 평가한 아이템이다.

③ 목표 아이템과 I_{other}의 장르사이에 유사도를 계산하기 위해 모든 I_{other}의 장르 수를 계산한다.

④ 유사도를 계산하는 3가지 방법(correlation, cosine, adjusted cosine)으로 목표 아이템과 I_{other} 간의 유사도 simattri(i,j)를 계산한다.

⑤ 다음의 식으로 복합적 유사도를 계산한다.

siminte(i,j) = (1-α)simattri(i,j) + α simattri(i,j)

α는 0-1사이에 가중계수이다.

⑥ siminte(i,j)와 목표 아이템과 가장 근접한 이웃 아이템들의 집합 NI에 속한 아이템들을 평가한 사용자들에 의한 예측값 $P(user, I_{aim})$을 다음 의 식 (1)로 계산한다.

$$P(user, I_{aim}) = \frac{\sum_{j \in NI} sim(I_{aim}, J) \times R_{user,j}}{\sum_{j \in NI} | sim_{\text{int}\,e}(I_{aim}, J) |} \qquad (1)$$

⑦ ②에서 ⑤까지 과정을 반복하여 사용자가 아이템을 평가하지 않은 모든 예측값을 계산한 후 예측값 $P(user, I_{aim})$를 정렬하고 사용자에게 첫 번째 N 예측값을 추천한다.

Ⅲ. 사용자 정보 가중치를 이용한 추천 기법

본 장에서는 협업 필터링의 희소성 문제와 확장성 문제를 해결하기 위해 유사도 계산 시 장르와 사용자 정보 가중치를 사용하여 예측값을 생성하는 사용자 정보 가중치를 이용한 추천 기법에 대하여 설명한다. 제안하는 추천 기법은 Step 1. Item user matrix 생성 단계, Step 2. 아이템 간 유사도 측정 단계, Step 3. 아이템 예측값 생성 단계로 구성된다. Step 1에서는 데이터베이스를 이용하여 아이템의 특성을 분석하고 Matrix를 생성한다. Step 2에서는 생성된 ma -trix의 아이템 특성을 이용하여 유사도를 계산한다. Step 3에서는 임계값 이상의 유사도 값을 가진 데이터만을 사용하여 예측값을 계산하고 평가되지 않은 아이템들의 평가값으로 사용한다.

3.1. Item user matrix 생성

이 단계에서는 user item matrix를 기반으로 유사도 계산 시 아이템에 대한 사용자 정보를 가중치로 사용하기 위해 아이템의 특성을 분석한다. 아이템 특성 분석을 위해서 Movielens 데이터셋의 아이템, 사용자, 평가값, 성별, 직업, 연령, 장르 정보가 사용된다. 본 논문에서는 아이템 선호에 대한 정확성을 높이기 위해 아이템 평가값이 4이상인 사용자들의 데이터만을 사용한다. 그 이유

는 아이템에 대한 평가값을 4 이상으로 한 사용자는 같은 특성을 가진 아이템을 다시 선택할 확률이 높기 때문이다. 아이템을 평가한 사용자들의 성별, 연령, 직업을 단순한 수치가 아닌 비율로 분석하여 아이템의 사용자 특성으로 지정한다. 식 (2)는 아이템에서 사용자 정보의 비율을 구하는 식이다. ar_{i,gen_m} 은 아이템에 대한 사용자 정보 중 성별의 특성을 추출하기 위해 성별의 비율을 계산하는 식이다.

$$ar_{i,k_l} = \frac{k_l}{\sum_{i=1}^{n} k_l} \qquad (2)$$

$$ar_{i,gen_m} = \frac{gen_{i,m}}{\sum_{i=1}^{n} gen_{i,m}}$$

k_l은 아이템을 선택한 사용자의 정보를 의미하며 성별, 연령, 직업이 그 값이 될 수 있다. $gen_{i,m}$은 아이템 i를 선택한 남자의 수를 의미한다. n은 전체 아이템이다. 연령은 5세 간격으로 15개 그룹으로 나누어 계산한다. 식 (2)를 사용하여 사용자 정보에 대한 비율을 계산하고 각 정보에서 비율 중 최대값을 아이템의 나이, 연령, 직업 특성값으로 사용한다. 아이템의 장르는 추천의 정확성을 높이기 위해 19개로 중첩된 장르를 88개로 좀 더 세분화한다. 장르를 세분화함으로써 새로운 아이템이 추가되었을 때 아이템을 빠르게 분류할 수 있으며 유사 아이템의 특성에 기반해 사용자에게 추천이 가능하다.

```
Algorithm MakeMatrix
Input :
(1) Item_set : {I_i}; User_set : {U_j};
    Ratings of users for items:{R_i,j},
    (i=1,2,…,n; j=1,2,…,m)
(2) Attribute of user : k_l, k_l={gender, age, job};
Output : Matrix;

1 ar_{i,k_l} ← ration of k_l
2 for i = 1 to n {
3     for j = 1 to m {
4         compute ar_{i,k_l} using formula (2)
5         insert max(k_l) into original_ij
6     }
7 }
```

그림 1. 아이템 유저 매트릭스 생성
Fig 1. item user Make matrix

표 1. 아이템 유저 매트릭스
Table 1. item user matrix

	genre	gender	age	job	1	2	3	4	5	6	…	14	15	16	17	18	19
1	G3-1	M	A6	lawyer				4	4		…	1		4	5		3
2	G1-1-9	M	A4	none				3			…						
3	G14	M	A4	none	4						…						
4	G1-3	M	A6	lawyer	3					5	…					4	
5	G5-1-2	F	A6	salesman	3						…						
6	G7	M	A12	healthcare	5						…						
7	G7-5	M	A4	none	4				2	5	…	1	5	4			
8	G3-1-1	M	A7	artist					4	5	…		5		5	5	
					∫												
15	G7	F	A13	technician	5				3		…		5				4
16	G4-6	M	A12	lawyer	5						…						
17	G1-3	M	A4	none	3			4			…						
18	G7	M	A10	technician	4						…	1					
19	G7	F	A9	libraian	5	3			4		…						
20	G7-4	F	A14	libraian	4						…	3					

그림 1은 item user matrix를 생성하는 알고리즘이고 표 1은 20개의 아이템과 사용자로 생성된 matrix의 예이다.

3.2. 아이템 간 유사도 측정

이 단계에서는 생성된 Matrix를 기반으로 유사도를 구한다.

Matrix에서 아이템간의 유사도는 피어슨 상관계수에 각 아이템을 선호하는 성별, 연령, 직업과 아이템의 장르를 가중치로 적용하여 계산한다. 식 (3)은 가중치 계산식이고, 식 (4)는 가중치를 적용한 유사도 계산식이다.

$$
\begin{aligned}
w_i &= 1 + w_{gen} + w_{age} + w_{job} + w_{grn} \\
w_{gen} &= \begin{cases} \alpha, & if\ i_{a,gen} = i_{b,gen} \\ 0, & otherwisw \end{cases} \\
w_{age} &= \begin{cases} \beta, & if\ i_{a,age} = i_{b,age} \\ 0, & otherwisw \end{cases} \\
w_{job} &= \begin{cases} \gamma, & if\ i_{a,job} = i_{b,job} \\ 0, & otherwisw \end{cases} \\
w_{grn} &= \begin{cases} \lambda, & if\ i_{a,mgrn} = i_{b,mgrn} \\ \varepsilon, & elseif\ \ i_{a,Lgrn} = i_{b,Lgrn} \\ 0, & otherwisw \end{cases}
\end{aligned}
\qquad (3)
$$

$$
Sim(i,j) = \frac{\sum_{u \in U_{i,j}} w_i^2 (r_{u,i} - \overline{r_i}) \cdot (r_{u,j} - \overline{r_j})}{\sqrt{\sum_{u \in U_{i,j}} (r_{u,i} - \overline{r_i})^2} \cdot \sqrt{\sum_{u \in U_{i,j}} w_i^2 (r_{u,i} - \overline{r_i})^2}} \qquad (4)
$$

식 (3)에서 w_i는 아이템 가중치로서 각각의 속성의 가중치 합으로 지정한다. w_{gen}, w_{age}, w_{job}, w_{grn}은 두 아이템 사이에서 각 아이템을 선호하는 성별, 연령, 직업, 장르 간의 가중치로서 [0, 0.4]사이의 값을 가진다. 식 (4)에서 $r_{u,i}$는 사용자 u가 아이템 i 를 평가한 평가값이다. $\overline{r_i}$는 i 번째 아이템의 전체 평가값 평균이다.

```
Algorithm ComputeItemSimilarity
Input : item user matrix;
Output : Item Similarity sim_{i,j};

1 for i = 1 to n {
2     compute the similarity of every two items using formula (4)
3 end for
```

그림 2. 아이템 유사도 계산
Fig 2. Compute similarity of items

그림 2는 아이템 특성 가중치를 사용하여 유사도를 구하는 알고리즘이다.

3.3. 아이템 예측값 생성

유사도 계산 후 임계값 이상의 아이템만을 사용하여 예측값을 계산하며 식 (5)는 예측값을 계산하는 식이다.

$$
p_{u,i} = \frac{\sum_{j=1}^{n} Sim_{i,j} \times r_{u,j}}{\sum_{j=1}^{n} Sim_{i,j}} \qquad (5)
$$

계산된 예측값 중에서 상위 N개의 데이터만을 사용하여 평가되지 않은 아이템들에 대한 평가값으로 지정한다.

```
Algorithm CreateItemPrediction
Input : item user matrix; similarity threshold : s-th;
        P_{u,i} threshold : p-th;
Output : Predicted Rating Value P_{u,i}; user_matrix

1 for i = 1 to n {
2    if sim(i,j) ≥ s-th then
3          select neighbors of each items;
4    end if
5    compute the P_{u,i} using formula (5)
6       if P_{u,i} ≥ p-th then
7          insert predicted value of rull rating item into Matrix
8       end if
9 end for
```

그림 3. 아이템 유사도 계산과 예측값 생성
Fig 3 Compute Similarity of item and create prediction

표 2. 아이템 예측값 채우기
Table 2. Fill prediction of item

	1	2	3	4	5	6	…	14	15	16	17	18	19
1	**3.9**			4	4		…	1		4	5		3
2				3			…						
3	4						…						
4	3					5	…					4	
5	3						…						
6	5						…						
7	4				2	5	…	1	5	4	**2.6**		
8	**4.8**				4	5	…		5		5	5	

∫

15	5				3		…		5				4
16	5						…						
17	3			4		**3.1**	…						
18	4						…	1			**1**		
19	5	3			4		…				3		
20	4						…	3			**2.3**		

그림 3은 예측값을 계산하는 알고리즘이고 표 2는 표 1의 예측값을 구한 결과이다.

제안하는 기법은 아이템의 특성을 추출한 후 그 특성을 가중치로 사용하여 유사도를 계산하고 예측값을 생성하는 방법이다. 이 방법은 새로운 아이템이나 사용자 모두를 분류하는 것이 가능하여 희소성의 문제를 줄일 수 있으며 추천의 정확성도 높일 수 있다. 또한 아이템 특성을 가중치로 사용하여 유사도를 계산하기 때문에 아이템 기반이나 장르기반에 비해 최근접이웃의 수가 줄어들어 예측값을 계산할 때 시간이 단축할 수 있어 확장성의 문제도 줄일 수 있다. 최근접 이웃의 수는 좀 줄어들지만 아이템 특성이 더 유사한 사용자들이 이웃으로 지정되므로 정확성이 떨어지지는 않는다.

그러나 사용자의 수가 너무 작을 경우 아이템 특성 분류의 정확성이 많이 떨어지므로 일정 수의 사용자가 확보되지 않은 초기 상태에는 적합하지 않을 수 있다.

Ⅳ. 실험 및 평가

4.1. 실험데이터

본 논문에서 제안하는 기법의 희소성 감소와 예측의 정확성을 실험하기 위해 MovieLens 데이터 셋을 사용하였다. MovieLens 데이터 셋은 미네소타 대학의 Group Lens Research Project에 의해 수집된 자료로서 943명의 사용자가 1,682개의 영화에 대하여 평가한 100,000개의 평가값을 갖는 데이터 셋이다[8].

본 논문에서 제안하는 기법의 성능을 평가하기 위해 데이터 셋을 80%의 training dataset과 20%의 test dataset으로 나누어 실험을 하였다.

예측의 정확성을 평가하기 위해 statistical accuracy metrics와 decision-support metrics 가 사용된다. Statistical accuracy metrics는 사용자가 평가한 값들과 예측된 값들을 비교하여 예측의 정확성을 측정하는 기법이다. Decision-support metrics는 사용자가 양질의 아이템들을 선택하는 것을 얼마나 잘 지원하는가를 측정하는 기법이다[13].

MAE(Mean Absolute Error)는 Statistical accu -racy metrics 기법으로 아이템에 대한 사용자의 실제 평가값과 추천시스템의 예측값의 차이에 대한 절대 평균으로 추천의 성능을 평가한다. 본 논문에서는 예측 성능 평가를 위해 MAE 기법을 사용하고, 사용자가 양질의 아이템을 선택하는 것을 얼마나 잘 지원하는 가를 측정하기 위해 F1 기법을 사용한다. 식 (6)은 MAE의 계산식이고 식 (7)는 F1 계산식이다.

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

n은 예측한 아이템의 수이고, p_i는 예측값이며, q_i는 사용자의 평가값이다. MAE의 값이 작을수록 예측의 정확성이 더 높아 사용자에게 더 좋은 추천을 한다.

$$F_1 = \frac{2R_1P_1}{R_1 + P_1} \qquad (7)$$

R_1은 재현율, P_1은 정확율을 나타낸다. 재현율과 정확율은 상호 상충관계에 있으므로 F_1을 사용한다.

4.2. 실험과 분석

실험은 3가지로 이루어졌다. 각 실험은 아이템 기반, 장르 기반 예측과 본 논문에서 제안하는 사용자 정보 가중치 기반 예측에 대하여 MovieLens 데이터 셋에서 예측의 정확성과 추천의 적합성, 신속성을 비교하였다.

실험 1에서는 유사도 임계값을 0.4에서 0.9까지 0.1씩 변화를 주어 아이템 기반 기법, 장르 기반 기법과 제안하는 기법의 예측의 정확성을 MAE의 측정결과로 비교하였다.

실험 2에서는 이웃의 크기를 10에서 50까지 10씩 변화를 주고 이에 따른 장르 기반 기법과 제안하는 기법의 추천의 적합성을 F1척도로 비교하였다.

실험 3에서는 아이템의 수를 200에서 1000까지 200씩 변화를 주고 이에 따른 아이템 기반 기법, 장르 기반 기법과 제안하는 기법의 예측값 생성시간을 비교하였다. 제안하는 기법의 수행시간에는 사용자 정보 속성을 추출하는 단계도 포함되었다.

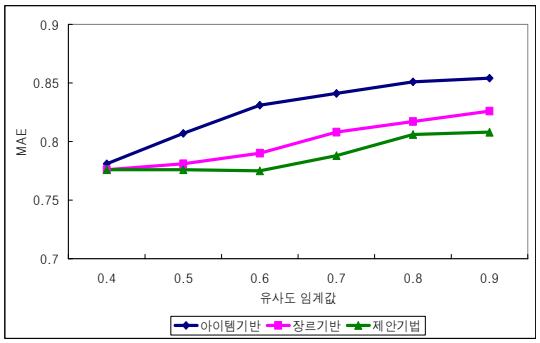

그림 4. 유사도 임계값 변화에 따른 예측성능 비교
Fig 4. Comparison of prediction quality under the different threshold of similarity

그림 4는 유사도 임계값의 변화에 따른 MAE 값을 측정한 결과를 나타내며 제안하는 기법이 아이템 기반 기법, 장르 기반 기법보다 MAE 값이 낮은 것을 볼 수 있다. 이는 제안하는 기법이 비교 기법들보다 예측 성능이 우수함을 나타낸다. 제안하는 기법은 유사도 임계값이 0.6일 때 최저의 MAE 값을 가지므로 이 값으로 임계값을 지정하는 것이 예측의 성능을 가장 많이 향상 시킬 수 있다.

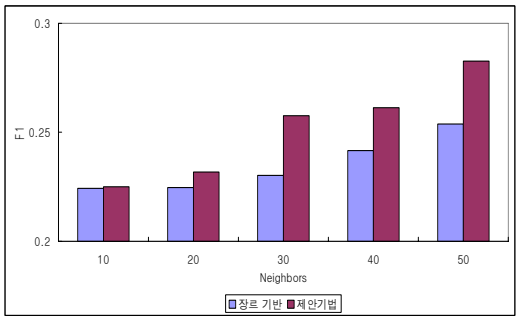

그림 5. 이웃 수 변화에 따른 적합성 비교
Fig 5. Comparison of suitability under the different number of neighbors

그림 5는 이웃의 크기 변화에 따른 장르 기반 기법과 제안하는 기법의 적합성을 측정한 결과를 나타내며 제안하는 기법이 장르 기반 기법보다 추천의 적합성이 더 높음을 알 수 있다.

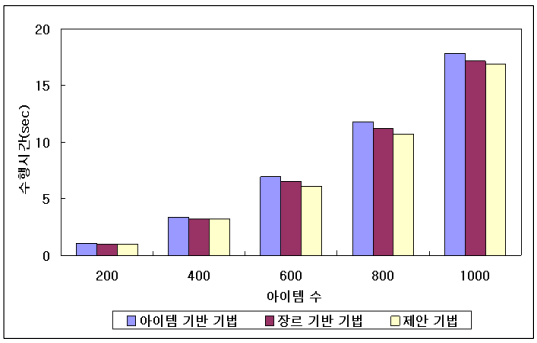

그림 6. 아이템 수 변화에 따른 예측값 생성 시간 비교
Fig 6. Comparison of creating time of prediction under the different number of items

그림 6은 아이템 수의 변화에 따른 예측값 생성시간을 측정한 결과이다. 제안하는 기법은 아이템의 수가 600개 미만일 때는 비교기법들과 예측값 생성시간에 차이가 별로 없지만 600개를 넘어가면서 제안하는 기법과 비교 기법들 사이에 예측값 생성시간에 차이가 커짐을 알 수 있다. 이는 아이템 수가 작을 때는 세 기법 모두 예측값 생성시간이 짧고 제안하는 기법은 사용자 정보속성 추출시간까지 포함하고 있기 때문에 세 기법 간에 시간 차이가 별로 나지 않는다. 그러나 사용자 정보속성 추출에 많은 시간이 소요되지 않으므로 아이템 수가 많아질수록 제안하는 기법과 비교 기법들의 예측값 생성시간에는 차이가 커진다.

V. 결 론

인터넷의 확산과 다양한 정보기기들의 발달로 정보가 급격히 증가하고 있으며 사용자들은 넘쳐나는 정보들 속에서 자신이 원하는 정보를 찾기 위해 많은 노력을 해야 한다. 기업들은 사용자들에게 그들이 원하는 정보를 빠르게 제공하여 더 많은 고객을 확보하기 위해 추천 시스템을 도입하고 있다. 추천시스템은 사용자들이 관심 있고 좋아할 만한 아이템을 추천해 주어 원하는 아이템을 쉽고 빠르게 찾을 수 있도록 돕는 시스템으로 협업 필터링이 가장 널리 사용되고 있다.

협업 필터링이 e-commerce에서 성공적으로 널리 사용되고 있지만 희소성, 확장성 등의 문제점을 가진다. 본 논문에서는 추천의 정확성에 심각한 문제를 발생시키는 희소성을 줄이고 정확성을 높이기 위해 사용자들의 정보를 가중치로 사용한 기법을 제안하였다.

제안하는 기법은 데이터베이스를 이용하여 아이템의 특성을 분석하고 새로운 matrix를 생성한 후 아이템 특성을 이용하여 유사도와 예측값을 계산한다. 계산된 예측값 중에서 임계값 이상인 값으로 평가되지 않은 아이템들을 채운다.

실험 평가를 통해 제안하는 기법의 정확성이 아이템 기반 기법, 장르 기반 기법보다 높음을 알 수 있었다. 제안하는 기법은 유사도가 0.6 일 때 아이템 기반 기법보다 6.7%, 장르 기반 기법보다 4.3% 예측의 정확성이 높았다. 이웃의 크기 변화에 따른 추천의 적합성 비교에서도 제안하는 기법이 장르 기반 기법보다 적합성이 높았다. 또한 제안하는 기법은 사용자 정보를 사용하여 아이템의 특성을 추출하였으므로 새로운 아이템이나 새로운 사용자가 추가되었을 때 추출된 특성을 기준으로 빠른 추천이 가능하며 예측값 생성을 위한 시간도 아이템 기반 기법보다 7.8%, 장르기반 기법보다 1.4% 단축되었다.

제안하는 기법은 사용자 정보를 분석하는 기법이기 때문에 사용자의 수가 너무 작을 경우 아이템 특성 분류의 정확성이 떨어지므로 일정 수의 사용자가 확보되지 않은 초기 상태에는 적합하지 않을 수 있다. 그러나 이 문제는 e- commerce의 이용이 보편화되고 있기 때문에 빠르게 해결될 수 있는 문제이다.

향후 연구과제는 다양한 종류의 핸드 헬드의 보급으로 m-commerce 이용자들이 증가하고 있으므로 모바일 환경에서 아이템을 보다 빠르고 정확하게 추천할 수 있는 방법을 연구하는 것이다.

참고문헌

[1] T. HengSong, Y. HongWu, “A Collaborative Filtering Recommendation Algorithm Based On Item Classification,” *Pacific-Asia Conference on Circuits, Communications and System.,* pp. 694 - 697, May 2009.

[2] Manow Papagelisa, Dimitris Plexoosakis, “Qualotative analysis of user-based and item-based prediction algorithms for recommendation agents,” ACM Transactions on Information Systems, 22, vol 1, pp116-142, 2004.

[3] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, “Analysis of Recommendation Algorithms for E-Commerce,” *In Processing of the 2nd ACMConference on Electronic Commerce*, pp. 158-67, Oct 2000.

[4] D. Goldberg, D. Nichols, B. Oki, D. Terry, ″Using collaborative filtering to weave an information tapestry,″ In Communications of the ACM, Vol. 35, No. 12, pp. 61-70, 1992.

[5] P. Resnick, N. Iacovou, M. Suchak, P. Bergs- trom, J. Riedl, “Grouplens: an open archi- tecture for collaborative filtering of netnews,” *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, pp. 175-186, 1994.

[6] Liang Zhang, Bo Xiao, Jun Guo, Chen Zhu, ″A Scalable Collaborative Filtering Algorithm Based on Localizied Prefernce,″ *Proceedings of the 7th International Conference on machine Learning and Cybernetics, Kunming*, pp. 160-167, July 2008.

[7] J. Breese, D. Heckerman, C. Kadie, ″Empirical Analysis of Predictive Algorithms for Collaborative Filtering,″ *Procedigs of the 14th Annual Conference on Uncertainty in Artificial Intelligence*, pp. 43-52, 1998.

[8] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, “Item based Collaborative Filtering Recommendation Algorithms,″ *Processing of the 10th International World Wode Web Conference*, pp. 285-295, 2001.

[9] T. Hofmann, J. Puzicha, ″Latent Claa Models for Collaborative Filtering,″ *In Proceedings of the 16th International Joint Conference on Artificial Intelligence*, pp. 688-693, 1999

[10] A. Kohrs, B. Merialdo, ″C;ustering for Collaborative Filtering Application,″ In proceedings of CIMCA’99. IOS Press, 1999.

[11] Z. Huang, H. Chen, D. Zeng, ″Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering,″ ACM Trans. Inf. Syst., 22(1)

pp. 116-142, 2004.
[12] Ye Zhang, Wei Song, "A Collaborative Filtering Recommendation Algorithm Base on Item Genre and Rating Similarity," *International Conference on Computational Intelligence and Natural Computing*, pp. 72-74, 2009.
[13] T.Hofmann, "Collaborative Filtering via Gaussian Probabilistic Latent Semantic Analysis," Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.259-266, 2003.

저자소개

윤소영(So Young Yun)

2007. 2 부경대학교 경영대학원 경영학석사
2009. 8 부경대학교 대학원 공학박사 수료

※ 관심분야 : 전자상거래, 데이터마이닝, 추천시스템, M-commerce 등

윤성대(Sung-Dae Youn)

1980. 2 경북대학교 컴퓨터공학과 공학사
1984. 2 영남대학교 대학원 전자계산학과 공학석사
1997. 2 부산대학교 대학원 전자계산학과 이학박사
1981～1986 경남정보대학 전산과 조교수
1991～1992 MIT 방문교수
1989～현재 부경대학교 컴퓨터공학과 교수
※관심분야 : 병렬처리, 멀티캐스팅통신, 데이터 마이닝 등