# 연관 아이템 트리를 이용한 추천 에이전트
## (A Recommender Agent using Association Item Trees)

고 수 정 †
(Su-Jeong Ko)

**요 약** 협력적 여과 시스템은 내용 기반 여과 시스템과는 대조적으로 아이템에 대한 정보를 반영하지 않으며, 또한 사용자가 자신의 흥미에 대한 정보를 제공하지 않았을 경우 추천을 할 수 없다는 단점을 갖는다. 본 논문에서는 협력적 여과 시스템의 단점을 해결하기 위하여 연관 아이템 트리를 이용한 추천 에이전트를 제안한다. 제안된 방법은 벡터 공간 모델과 K-means 알고리즘을 이용하여 사용자를 군집시킨 후 그룹의 대표 평가값을 추출한다. 다음으로, 군집된 그룹별로 아이템간의 상호정보량을 계산하여 아이템간의 연관도를 파악하며, 이를 기반으로 연관 아이템 트리를 생성한다. 이와 같이 생성한 각 그룹의 연관 아이템 트리와 그룹의 대표 평가값을 이용하여 새로운 사용자에게 아이템을 추천한다. 제안된 추천 에이전트는 사용자 정보와 아이템 정보를 병합하여 새로운 사용자에게 아이템을 추천하며, 아이템간의 유사도를 계산하기 위하여 상호정보량을 사용하고 이를 기반으로 연관 아이템 트리를 생성함으로써 초기에 아이템에 대하여 평가한 정보가 부족한 사용자에게 정확도가 높은 아이템을 추천할 수 있다는 장점을 갖는다. 제안된 방법은 MovieLens 추천 시스템의 데이터 집합을 사용하여 기존의 방법과 비교하였다.

**키워드** : 연관 아이템 트리, 추천 에이전트, 협력적 여과 시스템

***Abstract*** In contrast to content_based filtering systems, collaborative filtering systems not only don't contain information of items, they can not recommend items when users don't provide the information of their interests. In this paper, we propose the recommender agent using association item tree to solve the shortcomings of collaborative filtering systems. Firstly, the proposed method clusters users into groups using vector space model and K-means algorithm and selects group typical rating values. Secondly, the degree of associations between items is extracted from computing mutual information between items and an associative item tree is generated by group. Finally, the method recommends items to an active user by using a group typical rating value and an association item tree. The recommender agent recommends items by combining user information with item information. In addition, it can accurately recommend items to an active user, whose information is insufficient at first rate, by using an association item tree based on mutual information for the similarity between items. The proposed method is compared with previous methods on the data set of MovieLens recommender system.

**Key words** : Association item tree, recommender agent, collaborative filtering system

## 1. 서 론

추천 시스템은 대용량의 정보로 인하여 발생하는 문제점을 해결하기 위한 방법으로 개발되어 왔으며, 불충분한 사용자의 정보를 보완하여 사용자가 제공하지 않았던 정보를 제시함으로써 사용자에게 보다 많은 선택의 기회를 부여한다[1]. 이러한 추천 시스템은 크게 두 종류, 즉 내용 기반 여과와 협력적 여과로 나눌 수 있다. 내용 기반 여과는 아이템에 대한 문서의 내용과 같은 객관적인 속성을 추출하여 다른 아이템과의 유사도를 분석함으로써 추천을 하며, 협력적 여과는 아이템에 대한 사용자의 의견과 같은 주관적인 속성을 분석하여

· 이 연구는 인덕대학 학술연구비 일부 지원에 의하여 수행되었음

† 종신회원 : 인덕대학 컴퓨터소프트웨어과 교수
sjko@induk.ac.kr

논문접수 : 2008년 1월 25일
심사완료 : 2008년 9월 25일

다른 사용자와 비슷한 흥미를 갖는 사용자를 찾고, 이 사용자의 흥미를 기반으로 새로운 아이템을 추천한다. 그러나 내용 기반 여과는 오디오나 비디오와 같은 아이템을 추천하기 어려우며, 사용자가 접근 하였던 아이템을 기반으로 추천하기 때문에 예상외의 아이템을 추천하는 경우가 드물다. 협력적 여과 시스템은 비슷한 흥미를 갖는 사용자의 수가 적을 경우 정확한 사용자의 흥미를 예측하기 어렵다는 단점과 새로운 사용자가 자신의 흥미에 대한 정보 제공을 하기 않았을 경우 추천을 할 수 없다는 초기 평가 문제의 문제점을 갖는다[2].

본 논문에서는 협력적 여과 시스템의 단점을 해결하기 위하여 연관 아이템 트리를 이용한 추천 에이전트를 제안한다. 제안된 추천 에이전트에서는 사용자가 평가한 값만을 기반으로 추천을 하지 않기 때문에 초기 평가 문제나 비슷한 흥미를 갖는 사용자의 수가 적을 경우 발생하는 문제점을 보완할 수 있으며, 또한 내용 기반 여과 시스템이 갖는 사용자가 접근하였던 아이템만을 기반으로 추천할 경우에 발생하는 문제점 역시 보완할 수 있다. 그룹별 연관 아이템 트리를 생성하기 위한 전처리로 벡터 공간 모델과 K-means 알고리즘을 이용하여 사용자를 군집시킨다. 다음으로, 엔트로피를 이용하여 그룹의 대표 평가값을 추출한다[3]. 군집된 그룹별로 아이템간의 상호정보량을 계산하고, 계산한 아이템간의 상호정보량과 각 그룹의 대표 평가값을 기반으로 연관 아이템 트리를 생성한다. 새로운 사용자는 유사도 평가에 의해 그룹으로 분류되며, 각 그룹에 저장된 연관 아이템 트리를 이용하여 새로운 아이템을 추천 받을 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 사용자를 군집하고 군집의 대표 평가값을 추출하는 방법을 기술하며, 3장에서는 아이템을 대상으로 상호정보량을 계산하는 방법과 그룹별로 연관 아이템 트리를 생성하는 방법, 그리고 4장에서는 새로운 사용자에게 아이템을 추천하는 방법을 기술한다. 5장에서는 MovieLens 시스템에서 사용되는 영화 데이터를 대상으로 제안된 방법의 성능 평가를 하고, 6장에서는 결론을 기술한다.

## 2. 사용자 군집과 그룹의 대표 평가값 추출

본 장에서는 연관 아이템 트리를 생성하기 위한 전처리로 사용자 군집을 하는 방법과 군집의 대표 평가값을 추출하는 방법을 기술한다[3].

### 2.1 벡터 공간 모델과 K-means 알고리즘을 이용한 사용자 군집

본 논문에서 제안된 방법은 벡터 공간 모델[4,5]을 기반으로 사용자 프로파일을 정의하고 사용자를 군집시킨다. 사용자 프로파일을 정의하기 위하여 m명의 사용자와 n개의 아이템을 갖는 m × n차원의 사용자-아이템 행렬에서 전체 평가값의 평균을 구하여 평균이상의 평가를 받은 아이템을 대상으로 프로파일을 구성한다. 사용자 $U_i$가 아이템 $i_j$에 대하여 평가한 값을 $r_{ij}$로 정의한다. 단, 평균 이하의 평가값은 0으로 정의한다. 식 (1)은 사용자 $U_i$의 프로파일($C_{ui}$)을 정의하는 식이다.

$$C_{ui} = \{r_{i1}, r_{i2}, r_{i3}, \cdots, r_{in}\} \quad (1)$$

두 사용자 프로파일에서 중첩된 아이템이 많을수록 사용자간의 유사도는 높아지고, 적을수록 유사도는 낮아진다. 반면, 프로파일에 구성된 평가값의 불균형을 해결하기 위해 각 사용자의 벡터 길이를 1로 동일하게 하는 벡터 길이 정규화를 한다. 식 (2)는 정규화된 프로파일 $C'_{ui}$을 정의하는 식이다.

$$C'_{ui} = \{w_{i1}, w_{i2}, w_{i3}, \cdots, w_{in}\} \quad (2)$$

식 (2)로 정의된 사용자 프로파일을 기반으로 벡터 공간 모델에 의하여 사용자간의 유사도를 계산한다. 이와 같은 원리에 따라 두 사용자 $U_i$와 $U_j$ 간의 벡터 유사도를 식 (3)에 의해 구한다.

$$Sim(U_i, U_j) = \sum_{R_k} w_{iR_k} \times w_{jR_k} \quad (3)$$

식 (3)에서 $R_k$은 사용자 $U_i$와 $U_j$의 프로파일에 공통으로 속한 아이템을 의미한다. $w_{jRk}$는 사용자 $U_i$의 프로파일에 포함된 아이템 $R_k$의 가중치를 의미하며, $w_{jRk}$는 사용자 $U_j$의 프로파일에 포함된 아이템 $R_k$의 가중치를 의미한다.

식 (3)에 의해 계산한 사용자 간의 유사도를 기반으로 K-means 알고리즘[6]을 적용하여 사용자를 군집한다.

### 2.2 군집의 대표 평가값 추출

아이템에 대한 그룹의 대표 평가값을 추출하기 위해 사용자가 아이템에 대해 평가한 값에 사용자의 엔트로피 가중치를 곱한다. 사용자가 아이템에 대해 평가한 정보는 엔트로피를 이용하여 추출한다. 계산된 엔트로피를 아이템의 평가값에 적용함으로써 오류가 있는 아이템의 평가값을 보완할 수 있다. 이를 위한 식은 식 (4)와 같다. 식 (4)는 아이템 $i_j$의 대표 평가값($R_{Ij}$)을 추출한다.

$$RI_j = \sum_i p_{ui,j} \cdot H'_{ui} \quad (4)$$

식 (4)는 군집내의 모든 사용자가 아이템 j에 대해 평가한 값에 유클리디언 길이를 이용하여 정규화한 사용자의 엔트로피($H'_{ui}$)를 곱하여 모두 더한 값이다.

표 1은 그룹의 대표 평가값 추출을 위한 사용자-아이템 행렬의 예이다. 표 1의 자료는 GroupLens 시스템[7]의 영화 추천 시스템에 사용된 자료로부터 무작위로 50명의 사용자와 30명의 아이템을 추출하고, 자료 중에서 결측치의 수가 적은 순서로 사용자 20명, 아이템 13개를 선택한 자료이다.

표 1 사용자-아이템 행렬의 예

| | $i_1$ | $i_2$ | $i_3$ | $i_4$ | $i_5$ | $i_6$ | $i_7$ | $i_8$ | $i_9$ | $i_{10}$ | $i_{11}$ | $i_{12}$ | $i_{13}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $u_1$ | 1 | 0.8 | 0.8 | 1 | 0.2 | 0.6 | 0.8 | 0.8 | | 0.8 | 0.8 | 1 | 0.8 |
| $u_2$ | 0.8 | 0.4 | 0.4 | 0.8 | 0.8 | 0 | 1 | 0.6 | 0.6 | 1 | 0 | | 0.6 |
| $u_3$ | 1 | 0.8 | 0.8 | 0.8 | 0.8 | 0.6 | 0.6 | 0.8 | 0.6 | 0.8 | 0 | 1 | 1 |
| $u_4$ | 0.8 | | | 1 | 0.8 | 0.6 | 0.8 | 1 | 0.8 | 1 | 0.8 | 1 | |
| $u_5$ | 1 | 1 | 0.8 | 1 | | 0 | 1 | 0.6 | | 1 | 1 | 1 | 1 |
| $u_6$ | 0.2 | 0 | 0.4 | 1 | | 0 | 0.6 | 0 | 0.4 | 0 | 0 | 0 | |
| $u_7$ | | 0.6 | 0.6 | 0.2 | | 0 | 1 | 0.8 | 0.2 | 1 | 0 | 0.6 | 0.6 |
| $u_8$ | 1 | 0.8 | 0.8 | 0.8 | | 1 | 0.8 | 0.8 | | 1 | 1 | 0.8 | |
| $u_9$ | 0.8 | | 0.6 | 0.8 | | | 1 | 0.4 | 0.4 | 0.8 | 1 | 1 | 0.8 |
| $u_{10}$ | 0.6 | | 0.4 | 0.8 | 0.4 | 0 | 0.4 | 1 | 0 | 0.8 | 0.8 | 1 | 0.4 |
| $u_{11}$ | | 0.8 | 0.8 | | | 0 | | | 0.6 | | | | |
| $u_{12}$ | 1 | 0.6 | 0.8 | | | 0.4 | | | 0.6 | | | | 0.8 |
| $u_{13}$ | 0.8 | 0.8 | 0.8 | 1 | | | 1 | 1 | | 0.8 | | 1 | |
| $u_{14}$ | 0.8 | | 0.8 | | 0.8 | 0.6 | 0.8 | 0.6 | | 1 | 0.8 | 0.6 | 0.8 |
| $u_{15}$ | 1 | 0.6 | 0.4 | | | 0 | 0.8 | 1 | 0 | 0.8 | | 0.8 | |
| $u_{16}$ | 1 | 0.6 | 0.8 | | | 0.2 | 0.4 | 0.4 | 0.6 | | 0.6 | 0.6 | |
| $u_{17}$ | 0.6 | 0 | 0.6 | 1 | | 0 | | 0.6 | | 0.6 | 0.8 | 0.4 | |
| $u_{18}$ | | 0.4 | 0.8 | 0.8 | | 0.2 | | | 0.2 | 0.8 | 0.8 | 1 | |
| $u_{19}$ | 0 | 0.6 | 0.6 | | | 0.2 | 0.6 | 0.8 | | 0.6 | | 0.8 | 0.4 |
| $u_{20}$ | 0.2 | 0.6 | 0.6 | | | 0.2 | | | 0.8 | 1 | 0 | | |

표 2 그룹의 대표 평가값 추출

| | Class1 | Class2 | Class3 | Class4 | Class5 |
|---|---|---|---|---|---|
| $i_1$ | 0.85 | 0.27 | 0.43 | 0.56 | 0.56 |
| $i_2$ | 0.45 | 0.07 | 0.60 | 0.64 | 0.21 |
| $i_3$ | 0.85 | 0.19 | 0.67 | 0.64 | 0.52 |
| $i_4$ | 0.70 | 0.34 | 0.52 | 0.51 | 0.47 |
| $i_5$ | 0.25 | 0.13 | 0.25 | 0.37 | 0.00 |
| $i_6$ | 1.00 | 0.00 | 0.05 | 0.39 | 0.07 |
| $i_7$ | 0.45 | 0.21 | 0.76 | 0.62 | 0.27 |
| $i_8$ | 0.50 | 0.38 | 0.61 | 0.78 | 0.30 |
| $i_9$ | 0.32 | 0.00 | 0.36 | 0.48 | 0.30 |
| $i_{10}$ | 0.69 | 0.34 | 0.96 | 0.90 | 0.16 |
| $i_{11}$ | 0.69 | 0.34 | 0.20 | 0.14 | 0.42 |
| $i_{12}$ | 0.68 | 0.41 | 0.58 | 0.83 | 0.32 |
| $i_{13}$ | 0.89 | 0.21 | 0.37 | 0.42 | 0.00 |

표 2는 표 1을 기반으로 사용자를 5개의 그룹으로 군집시킨 후 식 (4)를 이용하여 그룹별로 대표 평가값을 구한 결과이다. 표 2에서 각 그룹은 {Class1, Class2, Class3, Class4, Class5}로 표기한다.

## 3. 연관 아이템 트리 생성

본 장에서는 아이템간의 연관도를 계산하고 군집별 대표 평가값을 이용하여 연관 아이템 트리를 생성하는 방법을 기술한다.

### 3.1 상호정보량을 이용한 아이템간의 연관도 계산

자연언어 처리 기술 중에서 두 단어의 동시출현빈도를 계산하기 위하여 특정한 공식 또는 계수를 사용하는데, 이때 사용하는 공식을 유사계수 또는 연관성 척도라고 한다. 코사인 계수를 비롯한 다양한 연관성 척도가 오랫동안 사용되어 왔지만 Shannon의 정보이론[8]에서 유래한 상호정보량을 Church & Hanks[9]가 연어 분석에 적용한 이후부터는 많은 연구에서 상호정보량의 공식을 이용하고 있다[10].

상호정보량은 두 사건 x와 y가 동시에 출현할 확률과 각각 독립적으로 출현할 확률 사이의 비율로, 식 (5)를 이용하여 구할 수 있다.

$$MI(x,y) = \log_2 \frac{p(x,y)}{p(x)p(y)} \tag{5}$$

식 (5)에서 p(x,y)는 x와 y가 동일 사용자에서 함께 평가된 경우의 빈도수를 나타내며, p(x)와 p(y)는 전체 아이템 수 n에 대한 아이템 x와 y의 상대빈도를 나타낸다. 이러한 개념을 기반으로 p(x,y), p(x), 그리고 p(y)는 식 (6)과 식 (7)로 정의된다.

$$p(x,y) = \frac{f(x,y)}{n} \tag{6}$$

$$p(x) = \frac{f(x)}{n}, p(y) = \frac{f(y)}{n} \tag{7}$$

반면, 이와 같이 정의한 경우 f(x,y)가 0인 경우가 발생한다. 즉, 공통적으로 평가된 아이템들의 빈도가 0인 경우이다. 따라서 식 (5)는 식 (8)로 재정의한다.

$$MI(x,y) = \log_2 \frac{p(x,y)}{p(x)p(y)} \quad if \ldots p(x,y) \neq 0$$

$$MI(x,y) = -1 \quad if \ldots p(x,y) = 0 \tag{8}$$

아이템간의 상호정보량을 계산하기 위하여 표 1로부터 사용자에게 흥미롭다고 평가된 아이템만을 추출하여 표 1를 재구성한다. 사용자에게 흥미롭다고 평가된 아이템들의 기준은 전체 평가값의 평균을 계산한 후 평균 이상인 아이템만을 사용자가 흥미롭다고 정의한다. 표 3은 식 (8)을 이용하여 아이템간의 상호정보량을 계산한 결과이다.

### 3.2 그룹의 트리 생성

본 절에서는 표 3과 같이 계산한 아이템간의 상호정보량과 표 2의 그룹별 대표 평가값을 이용하여 아이템간의 관계를 나타내는 연관 아이템 트리를 생성한다. 그림 1은 표 2의 첫 번째 그룹인 Class1에 부여된 대표 평가값과 표 3의 아이템간의 상호정보량을 기반으로 트리를 생성한 결과를 보인다. 이를 위하여, 표 2에 부여된 그룹의 대표 평가값을 기반으로 아이템을 내림차순으로 정렬한다. 그 결과, [$i_6$->$i_{13}$->$i_1$->$i_3$->$i_4$->$i_{10}$->$i_{11}$->$i_{12}$->$i_8$->$i_2$->$i_7$->$i_9$->$i_5$]의 순서로 정렬된 아이템과 표 3에 나타난 상호정보량을 이용하여 트리를 생성해 나간다. 그림 1은 $i_6$, $i_{13}$, $i_1$, $i_3$, $i_4$를 루트 노드로 하는 각각의 서브 트리를 생성한 결과이다.

표 3에서의 상호정보량값은 0과 1의 사이에 있으므로, 그 중간값인 0.5를 기준으로 0.5보다 큰 아이템을 선택하여 트리를 완성해간다.

그림 1과 같은 방법으로 생성한 $i_6$, $i_{13}$, $i_1$, $i_3$, $i_4$를 루트 노드도 하는 서브 트리 외에 $i_7$, $i_{10}$.. 등의 순으로 각각의 아이템에 대해 상호정보량에 따라 트리를 생성해간다. 각각의 아이템에 대해 생성된 서브 트리를 모두 연관시켜서, 최종적으로 연관 아이템 트리를 그림 2와 같이 생성한다. 생성된 연관 아이템 트리는 상단 부분에서는 높은 연관도를 보이나 하단부분에서는 연관도가 점차 낮아짐을 볼 수 있다.

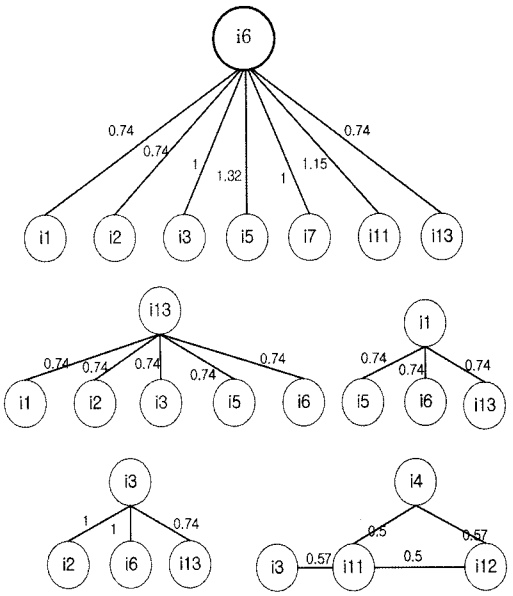


그림 1 $i_6$, $i_{13}$, $i_1$, $i_3$, $i_4$를 루트 노드로 하는 각각의 서브 트리

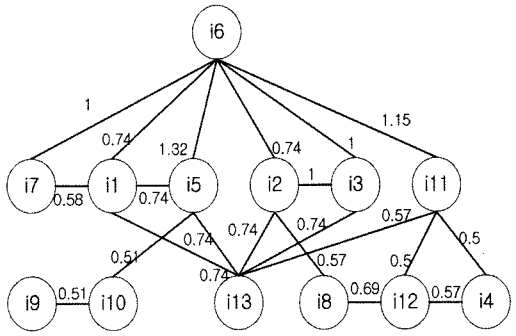


그림 2 Class1의 연관 아이템 트리

표 3 아이템간의 상호정보량

| | $i_1$ | $i_2$ | $i_3$ | $i_4$ | $i_5$ | $i_6$ | $i_7$ | $i_8$ | $i_9$ | $i_{10}$ | $i_{11}$ | $i_{12}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $i_1$ | | | | | | | | | | | | |
| $i_2$ | 0.474 | | | | | | | | | | | |
| $i_3$ | 0.222 | 1.000 | | | | | | | | | | |
| $i_4$ | 0.085 | 0.278 | -0.138 | | | | | | | | | |
| $i_5$ | 0.737 | -0.263 | 0.000 | 0.447 | | | | | | | | |
| $i_6$ | 0.737 | 0.737 | 1.000 | -0.138 | 1.322 | | | | | | | |
| $i_7$ | 0.585 | 0.415 | 0.000 | -0.138 | 0.585 | 1.000 | | | | | | |
| $i_8$ | 0.152 | 0.567 | -0.170 | 0.015 | 0.152 | 0.152 | 0.415 | | | | | |
| $i_9$ | -0.263 | -1.000 | -1.000 | -0.138 | 1.322 | -1.000 | 0.000 | 0.152 | | | | |
| $i_{10}$ | 0.252 | 0.252 | 0.000 | 0.225 | 0.515 | 0.515 | 0.363 | 0.152 | 0.515 | | | |
| $i_{11}$ | 0.152 | 0.152 | 0.152 | 0.500 | 0.152 | 1.152 | 0.415 | -0.018 | 0.152 | 0.345 | | |
| $i_{12}$ | 0.278 | 0.599 | 0.126 | 0.573 | 0.447 | -0.138 | 0.348 | 0.693 | -0.138 | 0.377 | 0.500 | |
| $i_{13}$ | 0.737 | 0.737 | 0.737 | -0.138 | 0.737 | 0.737 | 0.415 | -0.433 | -1.000 | 0.252 | 0.567 | 0.278 |

## 4. 연관 아이템 트리를 이용한 아이템 추천

새로운 사용자와 그룹의 대표 평가값과의 유사도를 계산한 후, 가장 유사도가 높은 그룹으로 사용자를 분류한다. 다음으로, 분류된 그룹의 연관 아이템 트리를 이용하여 아이템을 추천한다.

그룹의 대표 평가값과 새로운 사용자가 입력한 평가값과의 유사도를 비교하기 위하여 유사도 비교 방법 중 가장 많이 사용되는 코사인 유사도를 이용한다[11]. 새로운 사용자를 그룹으로 분류하기 위하여 그룹의 대표 평가값과 새로운 사용자의 유사도를 계산하기 위한 식은 식 (9)와 같다.

$$\cos ine(class_k, U_i) = \frac{\sum_{j=1}^{n} ci_j \cdot ru_j}{\sqrt{\sum_{j=1}^{n} ci_j^{\,2}} \cdot \sqrt{\sum_{j=1}^{n} ru_j^{\,2}}} \quad (9)$$

새로운 사용자 $U_a$가 표 4와 같이 평가를 하였을 경우, 식 (9)를 이용하여 각 그룹과의 유사도를 계산하여 그림 3과 같은 결과를 보인다.

표 4 새로운 사용자 Ua의 평가값

| | $i_1$ | $i_2$ | $i_3$ | $i_4$ | $i_5$ | $i_6$ | $i_7$ | $i_8$ | $i_9$ | $i_{10}$ | $i_{11}$ | $i_{12}$ | $i_{13}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $U_a$ | 1 | 0.6 | 0.8 | 0.8 | 0.2 | 1 | 0.8 | 0.6 | | 0.6 | 1 | 0.6 | |

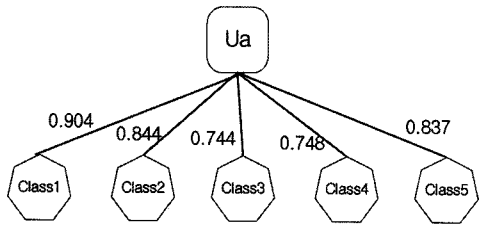


그림 3 사용자 $U_a$와 각 그룹과의 유사도

표 4의 $U_a$는 그림 3의 유사도 값에 따라 Class1로 분류하였다. $U_a$가 평가한 값을 기반으로 연관 아이템 트리를 이용하여 사용자가 평가하지 않은 아이템에 대하여 결측치를 예측한다. 사용자 $U_a$가 가장 높게 평가한 아이템은 $i_1$, $i_6$, $i_{11}$이다. 그림 2에서 $i_1$, $i_6$, $i_{11}$ 노드들이 가장 높은 연관도를 갖는 노드들을 상하로 순회해서 트리의 루트 노드나 최하단 노드까지 도달하도록 한다. 표 5는 $i_1$, $i_6$, $i_{11}$의 노드들을 검색하여 연관 아이템을 검색한 결과이다.

$i_1$ 노드는 $i_6$, $i_5$, $i_{13}$의 노드들에 동일하게 높은 연관도를 나타낸다. 또한, 연관도가 높은 아이템의 방향으로 최상 노드와 최하위 노드로 이동할 경우 $i_6$ 노드로 이동

표 5 $i_1$, $i_6$, $i_{11}$ 노드의 연관 아이템

| 상위의 평가값 노드 | 연관 아이템 트리 순회 |
|---|---|
| $i_1$ | (1) $i_6$(0.74) (2) $i_5$(0.74) -> $i_{13}$(0.74) (3) $i_{13}$(0.74) |
| $i_6$ | (1) $i_{11}$(1.15) -> $i_{12}$(0.5), $i_4$(0,5) |
| $i_{11}$ | (1) $i_6$(1.15) |

하고, 다음으로 $i_5$ 노드로, $i_5$ 노드는 $i_{13}$ 노드로 이동한다. $I_{13}$ 노드는 최하위 노드이므로 더 이상 이동하지 않는다. $i_6$ 노드는 가장 높은 노드인 $i_{11}$로, $i_{11}$ 노드는 동일한 연관도를 보이는 $i_4$ 노드와 $i_{12}$ 노드로 이동한다.

이와 같이 트리를 순회하여 $U_a$가 평가하지 않은 아이템들 중 연관도가 높은 아이템을 추천한다. 즉, 표 5에서와 같이 연관 아이템 트리의 순회 결과, $i_{13}$ 노드가 $i_1$ 노드와 0.74의 연관도를 나타내므로 $i_{13}$ 노드는 $i_1$ 노드와 비슷한 평가를 하였다고 예측한다. 따라서, $i_{13}$은 $i_1$과 같은 평가값인 1로 결측치를 예측한다. 평가값이 1인 경우는 사용자가 아이템에 대하여 흥미를 느끼고 있다고 판단되므로 추천을 한다.

연관 아이템 트리를 이용하여 아이템의 결측치를 예측하는 방법이 타당하다는 것을 수학적 귀납법을 통해 증명하고자 한다. 수학적 귀납법은 명제 $P_1$, $P_2$, $P_3$, …, $P_k$, ..., $P_n$이 사실이라고 할 때 $P_{n+1}$의 경우에도 적용된다는 것을 보이는 것이다. 명제 $P_k$와 명제 $P_{n+1}$은 다음과 같이 정의된다.

$P_k$: 아이템 $i_k$의 평가값이 결측되어 그 값을 예상하였을 때, 그 결과는 실제 평가값과 같은 값을 갖는다.
$P_{n+1}$: 아이템 $i_{n+1}$의 평가값이 결측되어 그 값을 예상하였을 때, 그 결과는 실제 평가값과 같은 값을 갖는다는 것을 증명한다.

수학적 귀납법에 의한 증명의 기초단계로서 n=1인 경우, 즉 $P_1$을 증명하고, 귀납가정 $P_1$, $P_2$, $P_3$, …, $P_k$, ..., $P_n$이 성립한다고 가정한다. 마지막으로, n+1의 경우에도 성립하는 것을 보인다.

귀납가정 $P_1$, $P_2$, $P_3$, …, $P_k$, ..., $P_n$ 중 명제 $P_k$의 증명을 위하여 사용자 $U_a$가 $i_{11}$에 대하여 평가를 하지 않았다고 가정한다. 이와 같이 가정했을 경우 $U_a$와 각 그룹과의 유사도는 {0.859,0.732,0.808,0.854,0.810}이다. 이와 같은 유사도를 기반으로 사용자 $U_a$는 Class1로 분류되었다. 다음으로, 연관 아이템 트리를 이용하여 사용자가 흥미를 느낄 것이라고 예상되는 아이템을 추천할 경우, $i_1$과 $i_6$ 노드를 중심으로 연관 아이템 트리를 순회한다. 표 5의 결과를 이용하여 $i_1$과 $i_6$ 노드를 중심으로 한 연관 아이템 트리의 순회 결과를 보면, $i_6$은 $i_{11}$과 가장

높은 연관도를 보인다. 따라서, $i_6$에 대한 평가값 1을 $i_{11}$에 배정함으로써 표 5에 나타난 $i_{13}$과 더불어 $i_{11}$을 사용자에게 추천할 수 있다. 따라서 명제 $P_k$의 가정은 성립한다.

## 5. 성능 평가

본 논문에서 제안된 방법을 실험하기 위한 데이터 집합으로 DEC 시스템 연구 센터에서 제공하는 EachMovie 협력적 여과 데이터 집합으로 데이터를 정제하여 추출한 MovieLens 데이터 집합을 사용하였다[10]. DEC는 18개월 동안 협력적 여과 시스템의 성능을 실험하기 위하여 EachMovie 추천 서비스를 실시하였으며, 그 결과로 72,916명의 사용자들이 1,628개의 영화에 대하여 28십만 개의 평가값을 갖는 데이터 집합을 수집하였다. MovieLens 시스템의 데이터 집합은 이러한 EachMovie 데이터 집합으로 데이터를 정제하여 체계적으로 수집한 것으로서, 기계 학습이나 산술적인 연구 프로젝트를 위하여 사용되어 왔다. 이러한 MovieLens 데이터 집합은 여러 연구가에 의해 사용되어 왔다. MovieLens 데이터 집합은 현재 43000명의 사용자가 3500개 이상의 영화에 대하여 평가한 데이터를 보유하고 있다.

본 논문에서는 제안된 방법의 성능을 평가하기 위하여 결측치가 적은 500개의 아이템을 추출하였다. 그리고 500개의 아이템에 대하여 평가를 한 5,000명의 사용자를 무작위로 추출하였다. 5,000명의 사용자들 중에서 500개의 아이템에 대하여 적어도 5개의 평가를 한 사용자를 추출한 결과 4,320명의 사용자를 추출하였다. 4,320명의 사용자 중 80%는 훈련집합으로 나머지 20%는 테스트 집합으로 사용하였다.

본 논문에서 제안한 방법은 추천의 결과가 사용자에게 적합함의 유무를 판단하므로 분류의 평가 척도를 사용한다. 분류의 평가 척도는 시스템이 추천한 아이템이 올바른가의 유무를 결정한다. 따라서 분류는 사용자가 이진의 평가값을 갖는다고 가정하고 좋은(good) 아이템을 찾을 수 있게 하는 척도이다. 추천은 좋은 값으로 평가될 것이라 예정되는 상위의 아이템을 예측하는 작업이라고 재정의될 수 있다. 이와 같은 분류의 평가 척도를 이용하는 방법은 정확도(Precision)와 재현률(Recall), 그리고 ROC 민감도 등이 있다.

정확도와 재현률은 정보 검색 시스템의 성능을 평가하는 매우 유용한 평가 척도 중 하나이다. 1968년에 Cleverdon[12]이 이와 같은 척도를 제안했으며, 그 이후로 많이 사용되고 있다. 또한 Billsus & Pazzani[13], Basu[14], 그리고 Sawar[15,16] 등의 여러 연구에서도 추천 시스템의 성능을 평가하기 위하여 정확도와 재현율을 사용하였다[17]

표 6 2 × 2 아이템의 분류

| | 선택 | 비선택 | 통계 |
|---|---|---|---|
| 적합 | nos | nol | no |
| 비적합 | nxs | nxl | nx |
| 합 | ns | nl | n |

정확도와 재현률은 2×2의 표 6로부터 계산할 수 있다.

추천된 아이템 집합은 적합한지 부적합한지의 두 분류 중 하나로 분류되어야 한다. 만약 평가 범위가 이진값이 아니라면 그들을 이진의 평가값으로 전환하는 작업이 필요하다. 본 논문에서는 0.5 이상의 값을 나타낼 경우 적합하다고 판단하고 그 이하일 경우 부적합하다고 판단하였다.

정확도는 선택된 아이템이 적합할 확률로, 선택된 아이템의 수에 대한 적합한 아이템의 수의 비율로 정의된다. 식 (10)은 정확도를 정의하는 식이다.

$$P = \frac{n_{os}}{n_s} \tag{10}$$

재현율은 식 (11)과 같이 적합할 가능성이 있는 아이템에 대한 실제로 적합한 아이템의 비율로서 정의된다. 즉, 재현율은 적합한 아이템이 선택될 확률을 나타낸다.

$$R = \frac{n_{os}}{n_o} \tag{11}$$

이와 같은 이론을 바탕으로 top-N의 추천 성능을 평가하기 위하여 식 (10)과 식 (11)를 수정한다. 아이템을 테스트 집합과 top-N의 집합으로 분류한다[12]. 이 두 집합에 모두 나타나는 집합은 hit 집합이라고 정의한다. 따라서 식 (10)과 식 (11)은 식 (12)로 재정의한다.

$$R_1 = \frac{hit\text{집합의크기}}{\text{테스트집합의크기}} = \frac{|tset \cap top_N|}{|tset|}$$

$$P_1 = \frac{hit\text{집합의크기}}{topN\text{집합의크기}} = \frac{|tset \cap top_N|}{N} \tag{12}$$

반면, 이와 같은 정확도와 재현율은 상반되는 경향이 있어 n의 수가 증가하면 재현율은 높아지나, 정확도는 감소하는 경향이 있다. 따라서 재현율과 정확도를 병합한 수식인 식 (13)의 F1 평가 척도[18]를 사용한다. F1 평가 척도는 재현율과 정확도에 같은 가중치를 부여하여 성능을 평가하는 기준이다.

$$F_1 = \frac{2R_1P_1}{R_1 + P_1} \tag{13}$$

분류의 평가 척도를 이용한 또 하나의 척도인 ROC 민감도는 추천 시스템이 사용자에게 높은 성능의 추천이 가능한지의 정도를 측정한다[19]. 예측된 아이템의 ROC 민감도는 그래프의 곡선을 이용하여 여과 시스템의 성능

을 측정한다. 곡선 아래 영역은 여과 시스템이 좋은 아이템을 보유할수록 증가한다. 여기서 좋은 아이템과 나쁜 아이템을 결정하는 것이 필요하다. ROC 민감도 측정은 추천 시스템이 좋은 아이템을 얼마나 많이 추천할 수 있는 가를 나타낸다. 특히, 1.0은 완전 여과라고 할 수 있으며, 0.5는 임의의 여과라고 할 수 있다[19].

본 논문에서는 제안된 방법(Tree_RA)의 성능을 평가하기 위하여 상호정보량의 변화에 따라 변화되는 ROC 민감도를 보인다. 또한, 각 그룹에 속한 사용자수의 변화에 따른 F1 측정인자를 계산함으로써 성능을 평가한다. 이와 같은 방법은 SVD를 이용하고 있는 방법(SVD_RA)[16]과 연관 규칙을 사용하고 있는 방법(AR_RA)[20]과 성능을 비교한다.

표 7은 상호정보량을 0.2에서 0.9까지 변화시킴에 따른 ROC 민감도를 나타낸다. 본 논문에서는 좋은 아이템과 나쁜 아이템을 구분하기 위해 사용자 자신의 평가값을 사용한다. ROC 민감도에서 좋은 아이템과 나쁜 아이템의 기준은 평가값이 0.5보다 클 경우 좋은 아이템으로, 그 외의 경우는 나쁜 아이템으로 정의하였다.

그림 4는 표 6을 기반으로 하고 있으며, Tree-RA의 ROC 민감도의 변화 곡선을 나타낸다.

표 7과 그림 4에서의 결과와 같이 제안된 방법 Tree-RA는 상호정보량이 0.5일 경우 최고의 ROC를 나타내며, 상호정보량이 0.5보다 클 경우 점차 그 값이

표 7 상호정보량에 따른 ROC 민감도의 변화

| 상호정보량 | ROC-0.5 |
|---|---|
| 0.2 | 0.6898 |
| 0.3 | 0.6923 |
| 0.4 | 0.7133 |
| 0.5 | 0.7331 |
| 0.6 | 0.7329 |
| 0.7 | 0.7235 |
| 0.8 | 0.6701 |
| 0.9 | 0.5932 |

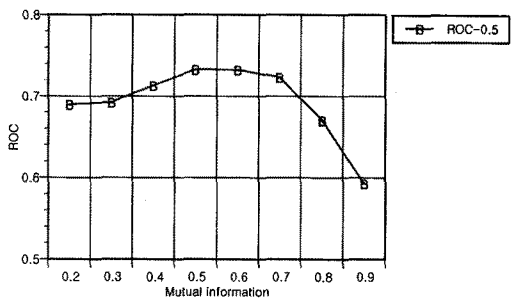


그림 4 표 7을 기반으로 하는 Tree_RA의 ROC 민감도의 변화 곡선

표 8 이웃의 수 변화에 따른 성능

| 이웃의수 | SVD_RA | AR_RA | Tree_RA |
|---|---|---|---|
| 30 | 0.2013 | 0.1994 | 0.2101 |
| 60 | 0.2015 | 0.1998 | 0.2132 |
| 90 | 0.2019 | 0.2014 | 0.2176 |
| 120 | 0.2103 | 0.2019 | 0.2181 |
| 150 | 0.2123 | 0.2021 | 0.2183 |
| 180 | 0.2128 | 0.2034 | 0.2187 |
| 210 | 0.2129 | 0.2041 | 0.2191 |
| 240 | 0.2130 | 0.2089 | 0.2193 |
| 270 | 0.2133 | 0.2091 | 0.2201 |
| 300 | 0.2139 | 0.2102 | 0.2203 |
| 평균 | 0.2093 | 0.2040 | 0.2175 |

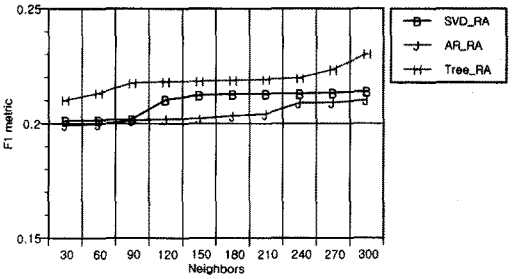


그림 5 표 8을 기반으로 하는 F1 척도의 결과

감소함을 볼 수 있다. 이와 같은 결과는 상호정보량이 임계값보다 클 경우 유사한 아이템의 수가 적어짐에 따라 추천할 수 있는 아이템의 수가 적어지고, 이에 따라 추천의 정확도가 낮아짐을 볼 수 있다. 따라서 상호정보량의 임계값을 0.5로 정하는 것이 추천의 정확도를 가장 높일 수 있음을 보였다.

표 8과 그림 5는 이웃의 수를 변화시킴에 따른 F1값의 변화를 나타내는 표와 그림이다.

표 8과 그림 5에서와 같이 제안된 방법 Tree_RA는 SVD_RA와 AR_RA보다 높은 F1 값을 보인다. 즉, SVD_RA와 같이 사용자 군집 기법만을 사용한 방법보다는 아이템간의 연관도를 반영한 Tree_RA 방법이 높은 성능을 보이며, 또한 연관도만을 고려한 AR_RA보다도 성능이 높다. 특히, 이웃의 수가 커짐에 따라 F1의 값은 점차 높아짐을 볼 수 있다. SVD_RA는 이웃의 수가 적을 때는 AR_RA와 비슷한 F1값을 보이나 이웃의 수가 커짐에 따라 AR_RA보다 높은 F1 값을 보인다.

## 6. 결 론

협력적 여과 시스템은 사용자가 아이템에 대해 평가한 정보를 기반으로 가장 적합한 아이템을 사용자에게 추천한다. 본 논문에서는 연관 아이템 트리를 이용한 추천 에이전트를 제안하였다. 제안된 방법은 사용자가 평

가한 정보를 기반으로 사용자를 군집시켜며, 군집된 그룹별로 아이템간의 상호정보량을 계산하여 연관 아이템 트리를 생성한다. 따라서 사용자 정보와 아이템 정보를 병합한 정보를 이용하여 새로운 사용자에게 아이템을 추천하는 장점을 갖는다. 또한, 아이템 기반의 연관 아이템 트리를 이용하여 추천을 함으로써 초기에 아이템에 대하여 평가한 정보가 부족한 사용자에게 보다 정확도가 높은 아이템을 추천할 수 있다는 장점을 갖는다.

향후, 보다 효율적인 그룹의 대표 평가값을 추출하는 방법을 연구하는 것이 필요하다.

## 참 고 문 헌


[1] Burke, R., "Knowledge-Based Recommender Systems," Encyclopedia of Library and Information Systems, Vol. 69, supplement 32, A. Kent, ed., Marcel Dekker, 2000.

[2] Wei, Y. Z., Moreau, L. and Jennings, N. R., "Learning users' interests by quality classification in market-based recommender systems," IEEE Trans on Knowledge and Data Engineering, Vol.17, No.12, pp. 1678-1688, 2005.

[3] 고수정, "사용자-상품 행렬의 최적화와 협력적 사용자 프로파일을 이용한 그룹의 대표 선호도 추출", 정보과학회 논문지, 제32권, 제7호, 2005.

[4] Salton, G., Wong, A., and Yang, C. S., "A vector space model for automatic indexing," Communications of ACM, Vol.18, No.11, 1975.

[5] Rijsbergen, V. and Joost, C., *Information Retrieval*, Butterworths, London-second edition, 1979.

[6] Alsabti, K., Ranka, S., and Singh, V., "An Efficient K-Means Clustering Algorithm," http://www.cise.ufl.edu/ranka/, 1997.

[7] MovieLens collaborative filtering data set, Http://www.cs.umn.edu/Research/GroupLens/index.html, GROUPLENS RESEARCH PROJECT, 2000.

[8] Shannon, C. E., "A mathematical theory of communication," Bell System Technical Journal, Vol. 27, pp. 379-423, 1948.

[9] Church, K. W. and P. Hanks, "Word association norms, mutual information, and lexicography," Computational Linguistics, Vol.16, No.1, 1990.

[10] 이재윤, "상호정보량의 정규화에 대한 연구", 한국문헌정보학회지, 제37권, 제4호, 2003.

[11] Kim, H., Lee, H., and Seo, J., "Improving FAQ Retrieval Using Query Log Clustering in Latent Semantic Space," In Proc. Of AIRS 2005, 2005.

[12] Cleverdon, C. and Kean, M., "Factors Determining the Performance of Indexing Systems," Aslib Cranfield Research Project, Cranfield, England, 1968.

[13] Billisus, D. and Pazzani, M. J., "Learning Collaborative information Filters," In proc. Of the 15th National Conference on Artificail Intelligence (AAAI-98), 1998.

[14] Basu, C., Hirsh, H., and Cohen, W. W., "Recommendation as classification:using social and content-based information in recommendation," In Proc. Of the 15th National Conference on Artificial Intelligence(AAAI-98), 1998.

[15] Sawar, B. M., Karypis, G., Konstan, J. A., and Ried l, J., "Analysis of recommendation algorithms for E_commerce," In Proc. Of the 2nd ACM Conference on Electric Commerce, 2000.

[16] Sawar, B. M., Karypis, G., Konstan, J. A., and Ried l, J., "Application of dimensionality reduction in recommender system - A case study," In Proc. Of the ACM WebKDD, 2000.

[17] Herlocker, J., Konstan, J., Terveen, L., and Riedl, J., "Evaluating Collaborative Filtering Recommender Systems," ACM Transactions on Information Systems, Vol.22, No.1, ACM Press, 2004.

[18] Yang, Y. and Liu, X., "A Re-examination of Text Categorization Methods," In Proc. Of ACMSIGIR'99, 1999.

[19] Mui, L., Ang, C., and Mohtashemi, M., "A Probabilistic Model for Collaborative Sanctioning," MIT LCS Technical Memorandum 617, 2001.

[20] Shyu, M., Haruechaiyasak, C., Chen, S., and Zhao, N., "Collaborative Filtering via Association Rule Mining from User Access Sequences," In Proc. of the International Workshop on Challenges in Web Information Retrieval and Integration, in conjunction with ICDE 2005, 2005.



고 수 정

1990년 인하대학교 전자계산학과 졸업(학사). 1997년 인하대학교 전자계산교육(석사). 2002년 인하대학교 전자계산공학과(박사). 2003년~2004년 Univ. of Illinois at Urbana Champaign Post Doc. 2004년~2005년 Colorado State University Research Scientist. 2005년~현재 인덕대학 컴퓨터소프트웨어과 교수. 관심분야는 데이터마이닝, 정보검색, 기계학습