사용자와 아이템의 혼합 협력적 필터링에서 Naïve Bayesian 알고리즘을 이용한 추천 방법

김용집*O, 정경용*, 한승진**, 고종철*, 이정현**
인하대학교 전자계산공학과*, 인하대학교 컴퓨터공학부**
{jobs76*O, dragon*}@nlsun.inha.ac.kr, kjc@bc.ac.kr*, {softman**, jhlee**}@inha.ac.kr

Recommendation Method using Naïve Bayesian algorithm in Hybrid User and Item based Collaborative Filtering

Yong-Jip Kim*O, Kyung-Yong Jung*, Seung-Jin Han**, Jong-Cheol Ko*, Jung-Hyun Lee**
{Dept.*, School**} of Computer Science & Engineering, Inha University

요 약

기존의 사용자 기반 협력적 필터링이 가지는 단점으로 지적되었던 희박성과 확장성의 문제를 아이템 기반 협력적 필터링 기법을 통하여 개선하려는 연구가 진행되어 왔다. 실제로 많은 성과가 있었지만, 여전히 명시적 데이터를 기반으로 하기 때문에 희박성이 존재하며, 아이템의 속성이 반영되지 않는 문제점이 있다. 본 논문에서는 기존의 아이템 기반 협력적 필터링의 문제점을 보완하기 위하여 사용자와 아이템의 혼합 협력적 필터링에서 Naive Bayesian 알고리즘을 이용한 추천 방법을 제안한다. 제안된 방법에서는 각 사용자와 아이템에 대한 유사도 검색 테이블을 생성한 후, Naive Bayesian 알고리즘으로 아이템을 예측 및 추천함으로써, 성능을 개선하였다. 성능 평가를 위해 기존의 아이템 기반 협력적 필터링 기술과 비교 평가하였다.

1 서론

하루가 다르게 증가하고 있는 인터넷의 방대한 정보는 경제적인 측면에서 그 관리의 효율성 또한 중요하다는 사실을 일깨워주고 있다. 과거에는 제공자 입장에서의 소극적 관리와 사용자 입장에서의 적극적 검색이 있었다면, 현재엔 그 양상이 반대로 변해야 한다는 것을 모두 인정한다. 이것은 정보의 검색 분야에서뿐만 아니라, 경제활동의 중추로 부상하는 전자상거래 분야에서 더욱 중요하다.

최근 세계적 인터넷 쇼핑몰인 Amazon.com을 비롯한 유수의 전자상거래 업체가 도입하고 있는 추천시스템이 바로 그 예라고 볼 수 있는데, 점차 방대해지는 사용자와 아이템에 관한 정보를 바탕으로 하는 추천 시스템이 가지는 문제점은 이미 많은 지적을 받아왔다. 대표적으로 희박성(Sparsity)문제와 확장성(Scalability)문제가 그것이며, 이것을 해결하기 위한 수많은 연구와 실험이 있어왔고 개선 여지 또한 많이 남아 있다[1].

본 논문에서는 기존의 추천 시스템이 가지는 문제점을 보완하기 위해 최근 기존 연구 대비 우수한 성능을 입증한 아이템 기반 협력 필터링 방법을 개선하고자 한다. 아이템 기반 알고리즘이 가지는 단점을 보완하기 위해 사용자의 접근 기록을 이용한 사용자간의 유사도를 참조하며, 이를 바탕으로 Naive Bayesian 알고리즘에 의한 추천을 시도한다. 제안된 방법의 성능평가를 위해 기존의 기술들과 비교 평가하였다.

2 관련 연구

2.1 아이템 기반 협력적 필터링

아이템 기반 협력적 필터링은 기존의 협력적 필터링들이 사용자들간의 유사도에 관심을 두었던 것과는 달리, 서로 다른 두 아이템에 대해 동시에 평가한 사용자들의 평가치로 계산한 아이템들간의 유사도를 기반으로 하는 추천방법이다. 아이템들간의 유사도를 구하기 위해 코사인 기반 유사도, 상관계수 기반 유사도, 개선된 코사인 유사도의 방법을 사용할 수 있는데, 이들 중 개선된 코사인 유사도가 정확성 측면에서 우수하다고 알려져 있다. 아이템들간의 유사도 $s_{i,N}$를 구했다면, 추천 대상 아이템 N에 대한 사용자 u의 평가 예측은 식(1)과 같은 가중치 합을 이용한 식으로 계산한다. $R_{u,N}$는 모든 유사 아이템들에 대한 사용자 u의 평가치를 말한다. 아이템 기반 협력적 필터링 방법은 사용자 기반 방법에 비하여 희박성과 확장성 측면에서 주목할만한 개선을 보여주었다[2]. 그럼에도 불구하고, 아이템에 대한 속성이 반영되지 않는 점과 여전히 명시적 평가 데이터에 의존하기 때문에 희박성의 영향을 많이 받는다는 점에서 그 개선 여지를 보여주고 있다.

$$P_{u,i} = \frac{\sum_{\text{all similarity items, N}} (s_{i,N} * R_{u,N})}{\sum_{\text{all similarity items, N}} (|s_{i,N}|)} \quad (1)$$

3 제안된 추천 시스템

본 논문에서 제안하는 추천 시스템의 전체 구성은 그림 1과 같다. 첫 번째 단계로 훈련 데이터 집합으로부터 사용자들이 각 아이템에 대하여 평가한 자료를 바탕으로 각 아이템간의 유사도를 계산한다. 이 계산을 근거로 아이템 유사도 검색 테이블을 생성한다. 동시에 동일한 훈련 데이터 집합으로부터 각 사용자가 아이템들에 접근한 기록을 이용하여 각 사용자 고유의 Interest-Behavior Matrix(I-B Matrix)를 생성한다. I-B Matrix는 사용자들간의 유사도를 측정하는 척도가 되며, 이것을 바탕으로 사용자 유사도 검색 테이블을 생성한다.

새로운 사용자의 추천 요구가 발생하였을 때, 아이템에 대한 새로운 평가치가 있거나 아이템 이용 내역이 추가 되었다면 이것은 훈련 데이터 집합에 추가 되어 사용자와 아이템 유사도 검색 테이블을 갱신하도록 한다. 두 번째 단계에서는 추천 요구에 대한 대응으로 아이템과 사용자 유사도 검색 테이블에서 가능성 있는 추천 후보 아이템과 유사 사용자 집단을 추출한다. 마지막 단계에서 추천 후보 아이템들과 유사 사용자들에 대해 Naive Bayesian 알고리즘을 적용하여 추천 후보 아이템에 대한 예측 및 추천을 수행한다.

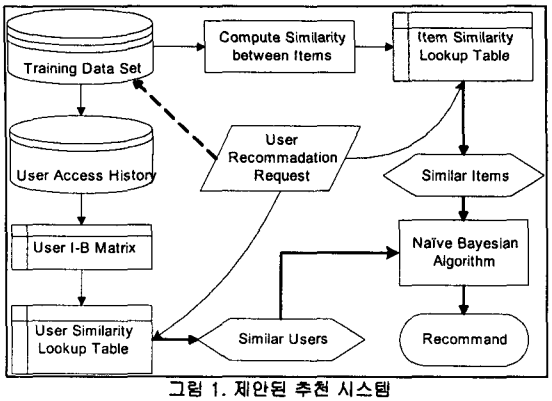

그림 1. 제안된 추천 시스템

3.1 아이템 기반 유사도 측정

아이템 유사도 검색 테이블을 생성하기 위한 아이템간의 유사도 측정의 기본 개념은 아이템 기반 협력적 필터링 알고리즘에서 출발한다. 임의의 사용자 u가 아이템 i와 함께 공통적으로 구매하거나 평가한 이력이 있는 아이템들은 서로 유사성을 가지고 있다는 가정을 하여 가장 유사성이 있다고 판단되는 아이템 집합 $\{i_1, i_2, \cdots, i_k\}$을 구하며, 동시에 아이템 i와 유사 아이템들 사이의 유사도 $\{s_{i1}, s_{i2}, \cdots s_{ik}\}$를 계산한다. 아이템들간의 유사도를 계산하기 위하여 개선된 코사인 유사도를 사용한다. 이것은 기존의 코사인 유사도 측정에 의한 계산이 서로 다른 사용자들 사이의 평가치의 차이를 계산에서 취급하지 않는다는 단점을 보완한 개선 방법이며, 기존의 코사인 기반 유사도와 비교해볼 때 정확성 측면에서 우수하다는 장점이 입증되었다[2].

아이템 i와 j에 대한 유사도 $sim(i,j)$는 식(2)로 계산된다.

$$sim(i,j) = \frac{\sum_{u \in U}(R_{u,i} - \bar{R}_u)(R_{u,j} - \bar{R}_u)}{\sqrt{\sum_{u \in U}(R_{u,i} - \bar{R}_u)^2}\sqrt{\sum_{u \in U}(R_{u,j} - \bar{R}_u)^2}} \quad (2)$$

$R_{u,i}$과 $R_{u,j}$는 아이템 i와 j에 대한 사용자 u의 평가치를 의미하며, $\bar{R}_u$는 사용자 u의 평가치를 평균한 값이다.

계산된 아이템들간의 유사도 값을 근거로 하여, 각 아이템과 유사한 아이템 집합과 그 유사도 값을 나타내는 아이템 유사도 검색 테이블을 생성한다. 각 아이템에 대응하는 유사 아이템 집합의 크기는 해당 유사도 값을 내림차순으로 정렬하여 k개의 원소를 갖도록 한다. 아이템 유사도 검색 테이블의 구성은 표 1과 같다.

표 1. 아이템 유사도 검색 테이블

Item	Similar Item Set
i_1	$(i_{17}, s_{1,17}), (i_8, s_{1,8}), \cdots, (i_{kth}, s_{1,kth})$
i_2	$(i_{209}, s_{2,209}), (i_4, s_{1,4}), \cdots, (i_{kth}, s_{2,kth})$
...	...
i_N	$(i_{92}, s_{N,92}), (i_{20}, s_{N,20}), \cdots, (i_{kth}, s_{N,kth})$

아이템 기반 유사도 측정방법은 사용자 기반 유사도 측정 방법과 비교해 볼 때 확장성 측면에서 장점을 가진다. 훈련 데이터를 통해 미리 계산된 유사도 값을 사용하기 때문에 우수한 성능과 높은 정확성을 일관성 있게 유지한다. 그러나, 명시적 데이터에 기반한 유사도 측정이며 아이템의 속성을 고려하지 않기 때문에 희박성 문제로부터 자유로울 수 없으며, 이러한 단점을 보완할 대책이 강구되어야 기존의 아이템 기반 추천 방법의 정확성을 향상시키는데 기여할 수 있다.

3.2 사용자 유사도 측정

명시적 데이터를 사용한 유사도 측정에서 기인하는 희박성 문제를 보완할 수 있는 방법으로써 묵시적 데이터, 즉 사용자가 아이템에 접근한 기록을 이용하여 유사한 사용자들을 검색한 후 추천에 참고 하는 방법이 사용될 수 있다. 우선 각 사용자들은 표 2와 같은 형태의 접근 기록을 가지게 된다. 이 접근 기록은 트랜잭션 번호와 각 트랜잭션에 있는 아이템 그룹으로 구성된다. 아이템 그룹은 아이템의 분류 속성에 따라 만들어진다. 이 접근 기록을 바탕으로 해당 사용자의 Interest Table을 작성하는데, 그 형태는 표 3과 같다. 이 테이블은 사용자가 각 아이템그룹에 대해 접근한 시작 트랜잭션(FT)과 마지막 트랜잭션(LT), 횟수(Count), 그리고 지지도(Support)를 기록한다. 지지도는 식(3)으로 계산된다.

$$\text{support} = \frac{\text{count}}{T_c - FT + 1} \quad (3)$$

T_c와 FT는 현재 트랜잭션의 번호와 시작 트랜잭션 번호를 의미한다.

표 2. 사용자의 접근 기록

Transaction	Item Groups in Transaction
T_1	A, C, E
T_2	B, C, E, F
...	...
T_x	A, G

표 3. Interest Table

Item Group	Count	FT	LT	Support
A	2	T_1	T_5	0.4
B	2	T_2	T_4	0.5
...	...	...	...	...
G	1	T_5	T_5	1

동일한 방법과 식을 사용하여 2개의 아이템에 대한 지지도를 나타내는 사용자의 Behavior Table을 표 4와 같은 형태로 생성한다. 이 테이블의 구성은 Interest Table 과 동일하다. 이렇게 하여 얻은 두 개의 테이블에는 모두 지지도가 기록되어 있다. 최소 지지도를 만족하는 아이템에 대하여 표 5와 같은 형태의 I-B Matrix를 구성할 수 있다. 지금까지의 과정을 통해 각 사용자는 고유의 I-B Matrix를 가지게 되며 유클리드 거리 계산식을 사용하여 서로 다른 사용자간의 유사성을 측정할 수 있다[3].

표 4. Behavior Table

Item Group Pair	Count	FT	LT	Support
AC	1	T_1	T_1	0.2
AE	1	T_1	T_1	0.2
...	...	...	...	...
EF	2	T_2	T_3	0.5

표 5. I-B Matrix

	A	B	C	D	E	F	G
A	0	0	0	0	0	0	1
B		0	0	0	0	0	0
C			1	1	0	1	0
D				0	0	1	0
E					0	0	0
F						1	0
G							1

각 사용자들간의 유사도를 알고 있다면 아이템 유사도 검색 테이블을 생성한 것과 같은 방법으로 표 6과 같은 사용자 유사도 검색 테이블을 생성할 수 있다. 다만 사용자의 유사도를 측정하는 척도로써 거리를 사용하였기 때문에 유사도와 거리는 반비례한다는 점이 다를 뿐이다.

표 6. 사용자 유사도 검색 테이블

User	Similar User Set
u_1	$(u_{17}, d_{1,17}), (u_8, d_{1,8}), \cdots, (u_{kth}, d_{1,kth})$
u_2	$(u_{209}, d_{2,209}), (u_4, d_{1,4}), \cdots, (u_{kth}, d_{2,kth})$
...	...
u_M	$(u_{92}, d_{M,92}), (u_{20}, d_{M,20}), \cdots, (u_{kth}, d_{M,kth})$

3.3 Naïve Bayesian 알고리즘을 사용한 추천

3.1절과 3.2절을 통해서 아이템 유사도 검색 테이블과 사용자 유사도 검색 테이블을 생성하였다. 사용자 u_c에 의한 추천 요구가 발생하였을 때, 우선 아이템 유사도 검색 테이블에서 사용자 u_c의 최

근 트랜잭션에서 접근이 있었던 아이템들과 유사한 k개의 아이템집합 $\{i_{c1}, i_{c2}, \cdots, i_{ck}\}$를 탐색한다. 이 과정에서 사용자 u_c가 접근했던 아이템과 중복되는 유사 아이템들은 제거된다. 동시에 사용자 유사도 검색 테이블에서 추천을 요구한 사용자와 유사한 m명의 사용자를 선택하며, 사용자 u_c와 유사한 사용자 $\{u_1, u_2, \cdots, u_m\}$의 최근 트랜잭션에서 접근이 있었던 아이템들의 합집합 $\{i_1, i_2, \cdots, i_n\}$을 구한다. 이상 3개의 집합들은 표 7의 형태로 구성된다.

표 7. Naive Bayesian 알고리즘을 적용하기 위한 테이블

	I_1	I_2	···	I_n	I_{c1}	I_{c2}	···	I_{ck}
U_1	0.8	0	···	0	0	0.2	···	1
U_2	0.6	1	···	0.4	1	0	···	0.8
···	···	···	···	···	···	···	···	···
U_m	0.2	0	···	1	0.2	0.6	···	0.6
U_c	0	0.6	···	1	?	?	···	?

아이템 i_k에 대한 사용자 u_c의 예측 및 추천은 식(4)의 Naïve Bayesian 알고리즘을 사용하여 사후확률(Posterior Probability) $P(C_{ik}|X)$ 를 최대화 함으로써 판단하게 된다.

$$P(C_{ik}|X) = \frac{P(X|C_{ik})P(C_{ik})}{P(X)} \quad (4)$$

X는 사용자 u_c가 가지는 조건, 즉 $\{i_1, i_2, \cdots, i_n\}$에 대한 평가를 나타내며, C_{ik}는 아이템 i_k의 클래스(0, 0.2, 0.4, 0.6, 0.8, 1)를 나타낸다. $P(C_{ik}|X)$는 조건 X일 경우 C_{ik}라는 클래스의 확률을 말한다. 사전확률(Prior Probability) $P(X)$는 일정하기 때문에 $P(X|C_{ik})P(C_{ik})$ 에 대해서만 고려한다면, 이것은 식(5) 로 표현한다.

$$P(X|C_{ik})P(C_{ik}) = P(C_{ik})\prod_{j=1}^{n} P(x_j|C_{ik}) \quad (5)$$

위의 식에 의한 각 추천 아이템$\{i_{c1}, i_{c2}, \cdots, i_{ck}\}$의 확률이 계산된다면, 각 아이템의 평가치를 예측할 수 있으며, 예측값의 크기에 따라 추천 순서를 결정할 수 있다.

4 실험 및 성능 평가

본 논문에서는 EachMovie 데이터[4] 중에서 1000명의 사용자와 1682편의 영화에 대한 정보, 그리고 각 사용자가 영화에 대해 평가한 항목들에 대해서 실험하였다. 그 중에서 80%의 데이터를 훈련 데이터로 사용하였으며 나머지 20%는 실험 데이터로 사용되었다. 먼저 각 아이템간의 유사도를 계산하여 표 8에서 제시한 유사도 검색 테이블을 데이터 베이스에 생성하였다.

표 8. 아이템 유사도 검색 테이블

Item_I	Item_J	Similarity
1	10	-0.271285645
60	338	0.405042944
205	1273	-0.93367832
424	1297	0.442048685
800	1031	0.162971873

각 아이템에 대한 사용자의 평가 데이터를 바탕으로 사용자의 접근 기록들을 생성하고, 표 9의 IB-Matrix와 표 10의 사용자 유사도 검색 테이블을 생성하였다.

표 9. IB-Matrix

User ID	IB_Matrix
875	0100001110001010000010010000000000001···
2198	0101101100011010101111100110111010010···
16409	0100011110001000000000100000000000000···
25373	1110111110001010001010000000000000001···
52932	0000001010001000001000010000100010000···

훈련 데이터 집합으로 생성한 아이템과 사용자 유사도 검색 테이블에서 실험 데이터에서 임의로 추출한 사용자 uc와 관련 있는 유사 아이템과 유사 사용자를 검색하여 Naive Bayesian 알고리즘을 표 11과 같이 적용하였다. 이상의 절차를 걸쳐서 기존의 아이템 기반 협력적 필터링에 의한 추천 방법과 본 논문에서 제안된 추천 방법의 성능을 모델 사이즈를 나타내는 매개변수 n에 따라 MAE[2]로써 비교하였고, 그림 2의 결과를 얻을 수 있었다.

표 10. 사용자 유사도 검색 테이블

User_I	User_J	Distance
1	2198	5.099019514
875	1918	2.828427125
1893	34880	5.567764363
36229	68778	3.31662479
58625	47222	3.464101615

표 11. Naive Bayesian 알고리즘 적용 예

	50	805	1604	329	480	420
30244	1	0	0	0.6	0.6	0
67217	0	0.8	0	0	0.4	0
6164	0.8	0	1	0.2	0	0
67008	0	1	0	0	0	0
68400	0	0	0	0	0	0
1918	0.6	0	0	?	?	?

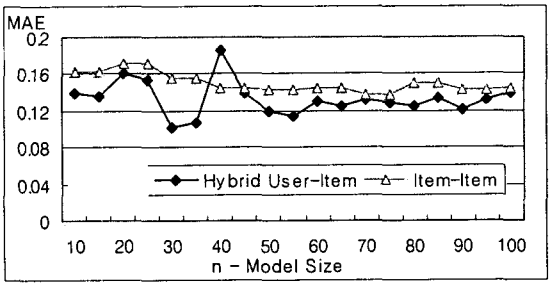

그림 2. MAE에 의한 성능 평가

그림 2를 통해 제안된 추천 방법이 기존의 아이템 기반 방법과 비교해 볼 때 대부분의 영역에서 정확성이 향상되었다는 것을 알 수 있다. 또한 n이 증가함에 따라서 정확성의 편차가 줄어드는 모습을 볼 수 있다. 그러나 n이 증가함에 따라 모델 사이즈의 증가와 데이터베이스 접근 횟수의 증가 폭이 기존의 방법보다 많아짐에 따라 속도 저하의 문제가 큰 것으로 확인되었으며 이를 보완해야 할 것으로 보인다.

5 결론

본 논문에서는 기존의 아이템 기반 협력적 필터링의 성능을 개선하고자 아이템과 사용자의 유사도 테이블을 참고하여 Naïve Bayesian 알고리즘을 적용하는 예측 및 추천 방법을 제안 하였다. 제안한 방법의 성능을 기존의 방법과 비교 실험한 결과 예측의 정확도가 향상되었으며, 본 논문에서 제안한 방식이 효과적임을 알 수 있다.

향후 연구과제로는 기존의 방법과 비교했을 때 정확성 측면 뿐만 아니라 속도 면에서도 효과적인 방법의 고안이 필요하며, 아이템 상호간의 유사도에 아이템의 속성을 고려한 가중치를 적용함으로써 유사 아이템 선정의 정확성을 높이도록 해야 할 것이다.

참고문헌

[1] G. Linden, B. Smith, J. York," Amazon.com recommendations : item-to-item collaborative filtering," Internet Computing, IEEE, Vol. 7, No. 1, pp. 76-80, 2003.

[2] B. Sarwar, G. Karypis, J. Konstan, J. Reidl, " Item-based collaborative filtering recommendation algorithms," Proc. of the 10th international conference on WWW, pp. 285-295, 2001.

[3] Hung-Chen Chen, Arbee L. P. Chen, " Collaborative Filtering and Algorithms : A music recommendation system based on music data grouping and user interests," Proc. of the 10th international conference on Information and knowledge management, pp. 231-238, 2001.

[4] P. McJones, EachMovie collaborative filtering dataset, URL:http://www.research.digital.com/SRC/eachmovie, 1997.