고객-제품 구매여부 데이터를 이용한 제품 추천 방안

A Product Recommendation Scheme using Binary User-Item Matrix

이 종 석* 권 준 범** 전 치 혁**

{jongseok, samson, chjun}@postech.ac.kr

포항공과대학교 *정보통신대학원 **산업공학과

경상북도 포항시 남구 효자동 산31

Abstract

As internet commerce grows, many company has begun to use a CF (Collaborative Filtering) as a Recommender System. To achieve an accuracy of CF, we need to obtain sufficient account of voting scores from customers. Moreover, those scores may not be consistent. To overcome this problem, we propose a new recommendation scheme using binary user-item matrix, which represents whether a user purchases a product instead of using the voting scores. Through the experiment regarding this new scheme, a better accuracy is demonstrated.

1. 서론

전자상거래가 활발해지면서 인터넷을 기반으로 한 고객관계관리, 즉 e-CRM에 대한 관심이 증가하고 있다[1]. 특히 B2C 업체들은 기존 고객 관리와 수익성 증대를 위해 제품 추천시스템 도입에 적극성을 띄고 있다.

추천시스템 중 가장 대표적인 것은 D.Goldberg(1992)가 이름붙인 협동적 필터링(Collaborative Filtering)이다[5]. 이 알고리즘의 목적은 온라인상에서 제품을 구입하려는 고객에게 이전에 구입했던 제품의 평가기록(voting history)과 타고객의 평가기록과의 유사성을 바탕으로 관련 제품을 추천해주는 것이다.

여러 가지 추천시스템 중 협동적필터링이 각광을 받는 이유는 첫째, 정보검색이나 정보필터링과 달리 제품의 사전분류가 필요치 않아 알고리즘 적용이 간단하며 둘째, 고객이 미처 생각지 못했던 맘에 드는 제품도 추천되므로 고객감동을 제공할 수 있기 때문이다.

여기에 아마존이 NetPerception[9]솔루션을 사용해 'Who bought' 서비스를 시작하면서부터 협동적 필터링은 더 큰 관심을 모았다.[2]

협동적 필터링의 단점으로는 추천을 위해 일정 이상의 고객평가가 있어야 한다는 점이다. 따라서 신규고객 유치를 위해서는 다른 마케팅 전략을 적용해야 하며, 구매가 몇 차례 이뤄진 후부터 협동적 필터링을 적용하는 것이 바람직하다.

좋은 추천시스템의 요건은 추천의 정확성과 빠른 응답속도이다. 그러나 고객의 평가치를 바탕으로 추천하는 전통적 의미의 협동적 필터링은 이 두 가지 요건에 취약함을 보이고 있다. 제품을 구입한 고객 중에 사용 후 평가를 하는 고객은 그다지 많지 않다. 이러한 평가치의 희소성(sparsity) 문제로 인해 추천의 정확도가 떨어진다. 또한 전자상거래 업체가 취급하는 제품의 수를 증가시키거나 고객의 수가 늘어날수록 추천제품을 계산하는데 걸리는 시간이 증가하게 되며(scalability) 그 결과 실시간 추천서비스를 제공할 수 없게 된다.

본 논문에서는 위의 두 문제 중 평가치의 희소성 문제를 극복할 수 있는 방법으로 고객-제품 구매여부 데이터를 이용한 추천법을 제안한다.

본 논문의 구성은 다음과 같다. 2절에서는 평가치의 희소성 문제를 해결하기 위한 관련연구와 본 논문에서 제시하는 새로운 기법을 소개한다. 3, 4절에서는 전통적인 협동적 필터링과 새로운 기법을 적용한 협동적 필터링의 정확도를 실험을 통해 평가한다. 끝으로 5절에서는 결론을 맺는다.

2. 관련연구

고객평가 데이터의 희소성을 극복하기 위해 많은 연구가 진행됐다. 가장 간단한 방법으로는 평가를 하지 않은 것에 가장 낮은 점수를 할당하거나 해당고객이 평가한 기록

의 평균값을 주는 것 등이다. 그러나 이것은 유기적인 연관성이 없어서 추천의 정확도를 높이지 못한다.

대안으로 고객평가 행렬의 차원을 낮춰주는 SVD(Singular Value Decomposition) 방법이 주로 이용된다.[4, 7]

이상의 방법들은 결측치(missing value)를 추정을 통해 대입하는 것에 주력하고 있다. 그러나 실제 데이터를 보면, 아무리 적극적인 고객이라고 할지라도 업체가 다루는 제품의 1%로도 평가하지 않은 것으로 나타나 있다. 즉, 위와 같은 방법으로 접근을 하면 알고 있는 적은 데이터로 모르는 다수의 결측치를 추정하는 셈이므로 추천의 신뢰성이 떨어진다.

또한 사용한 제품에 대한 고객의 평가가 모두 다른 시점에 이뤄지므로 평가때마다 기준이 달라져, 평가치의 일관성(Consistency)에 의문이 제기된다.

따라서 본 논문에서는 고객의 평가치를 입력값으로 사용하지 않는 대신에 고객이 구입한 제품에는 1을 그렇지 않은 제품에는 0을 할당해서 추천하는 시스템을 만들었다. 이 방법의 장점은 첫째 고객의 주관적인 평가치를 수집하는 비용이 들지 않고 둘째, 입력값의 증가로 데이터의 희소성이 감소한다는 점이다. 실험결과 추천의 정확성도 높아졌다.

3. 새로운 방법의 제안

user-item의 평가행렬을 이용하는 협동적 필터링의 평가치 희소성 문제를 해결하기 위해 user-item의 이분 행렬(Binary Matrix)을 이용한다.

3.1 이분 행렬의 사용

기존의 협동적 필터링에 사용된 평가행렬은 아래 표와 같은 구조를 가진다.

<표 1> user-item rating matrix

	i_1	i_2	i_3	⋯	i_j	⋯	i_m
u_1	R		R				R
u_2			R		R		
⋮							
u_a	R		R				
⋮							
u_n		R					R

위 표에서 보듯이 각 셀은 제품에 대한 고객의 평가점수 R을 가지거나, 그렇지 않은 경우는 모두 결측치로 여겨진다. 위 행렬을 사용하는 경우 사용자의 저조한 평가 참여로 인해 데이터 희소성이 높은 편이며, 이는 부정확한 추천으로 이어진다. 따라서 사용자의 평가에 의존하지 않는 새로운 방법을 제안하고자 한다.

제안하고자 하는 행렬은 단순히 고객의 특정 제품 구매여부를 1(구매)과 0(비구매)으로 표현한 이분 행렬이며, <표 2>와 같은 구조를 가지게 된다.

<표 2> user-item binary matrix

	i_1	i_2	i_3	⋯	i_j	⋯	i_m
u_1	1	1	0	⋯	1	⋯	0
u_2	0	1	0	⋯	1	⋯	1
⋮	⋮	⋮	⋮		⋮		⋮
u_a	1	1	1	⋯		⋯	1
⋮	⋮	⋮	⋮		⋮		⋮
u_n	0	0	1	⋯	1	⋯	0

이분 행렬은 구매여부에 따라 1과 0을 할당했기 때문에, 결측치가 하나도 없다. 따라서 분석을 위해 결측치를 추정할 필요가 없어졌다. 또한 기존의 협동적 필터링에서는 실제 물건을 구입하고도 평가를 하지 않은 고객의 정보는 이용할 수 없었지만, 새로운 방법을 적용하면 이러한 값에 1을 할당하게 되므로 더 많은 정보를 이용할 수 있게 됐다. 즉, <표2>의 행렬을 차지하는 1의 비율이 <표1>의 평가 행렬의 R의 비율보다 높아졌고, 이러한 요인이 추천의 정확도에 기여할 것이라 판단했다.

3.2 수정된 수식의 사용

기존의 협동적 필터링에서 고객 i의 제품 j에 대한 평가치 $v_{i,j}$를 바탕으로 active user a의 특정 아이템 j에 대한 예측 값 $P_{a,j}$는 식 (1)을 통해 계산된다.

$$P_{a,j} = \bar{v}_a + \kappa \sum_{i=1}^{n} w(a,i)(v_{i,j} - \bar{v}_i) \quad (1)$$

$$\kappa = 1 / \left| \sum_{i=1}^{n} w(a,i) \right| \quad (2)$$

$$\bar{v}_i = \left(\sum_{j \in I_i} v_{i,j} \right) / |I_i| \quad (3)$$

$$w(a,i) = \frac{\sum_j (v_{a,j} - \bar{v}_a)(v_{i,j} - \bar{v}_i)}{\sqrt{\sum_j (v_{a,j} - \bar{v}_a)^2 \sum_j (v_{i,j} - \bar{v}_i)^2}} \quad (4)$$

위 식에서 고객간의 유사성(Pearson Correlation Coefficient를 사용하는 경우)을 나타내는 $w(a,i)$에서 j는 고객 a와 고객 i가

공통적으로 평가한 제품을 가리키며, 평균평가치(mean vote)를 나타내는 식(3)의 I_i는 고객 i가 평가한 제품집합을 의미한다.

하지만, 이분 행렬을 이용할 경우 식(5), 식(6) 과 같이 수정될 필요가 있다.

$$\bar{v}_i = (\sum_{j \in I} v_{i,j}) / |I| \quad (5)$$

$$w(a,i) = \frac{\sum_{j \in I} (v_{a,j} - \bar{v}_a)(v_{i,j} - \bar{v}_i)}{\sqrt{\sum_{j \in I} (v_{a,j} - \bar{v}_a)^2 \sum_{j \in I} (v_{i,j} - \bar{v}_i)^2}} \quad (6)$$

즉, 이분행렬을 구성하는 값은 오직 1과 0뿐이므로, 유사성과 고객의 평균평가치가 모든 제품집합, I에 대해서 계산이 되어져야 한다. 또한, 식(1)은 식(7)과 같이 수정돼야 한다. 이는 $v_{i,j}$의 값이 1과 0뿐이므로 즉, 고객마다의 평가경향이 달라지는 문제가 발생하지 않기 때문이다.

$$P_{a,j} = \kappa \sum_{i=1}^{n} w(a,i) \cdot v_{i,j} \quad (7)$$

4. 수치예제

4.1 실험데이터

본 논문에서 제안하는 방법을 실험하기 위한 데이터는 DEC Systems Research Center에서 제공하는 EachMovie data set[10]을 사용하였다. 이 데이터는 72,916명의 사용자가 1,628개의 영화와 비디오에 대해서 평가한 것으로서 평가의 단계는 (0.2,0.4,0.6,0.8,1.0)의 5단계로 이루어져 있다. 데이터의 양이 너무 많은 관계로 본 실험에서는 254명이 757개의 영화에 대하여 10,895개의 평가를 한 데이터를 사용하여 실험을 하였다.

4.2 실험방법 및 정확도 평가

정확도를 측정하기 위한 척도로는 정보검색에서 사용되는 Precision을 사용하였으며, 이는 추천 대상자에게 추천되어진 Top_N의 제품에 대한 매칭 비율을 나타낸다. N은 Top_N을 구성하는 제품의 개수이다.

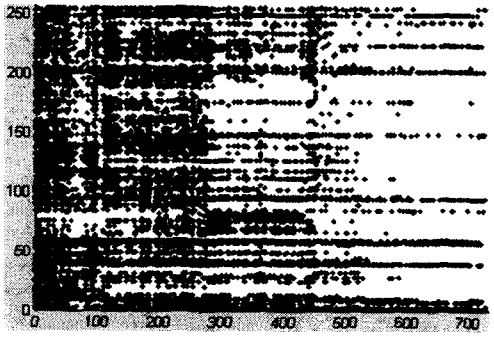
$$Precision = |test \cap Top\_N| / N \quad (8)$$

<표 3> Precision of traditional CF

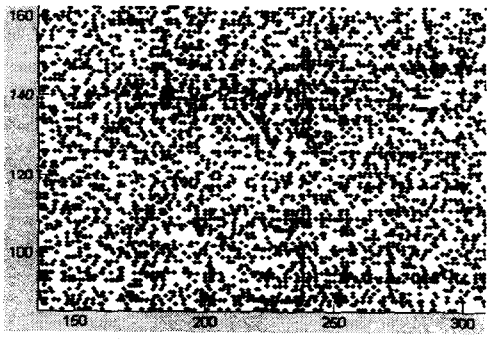
Top_N	5	6	7	8	평균
Precision(%)	32.46	32.60	31.88	31.16	32.03

<표 3>은 기존의 협동적 필터링의 실험결과를 나타낸 것이다. 평균 32%의 Precision을 보이고 있다.

본 논문에서 제안하는 방법을 실험하기 위해서는 영화에 대한 평가 외에 영화를 보았는지 유무를 나타내는 데이터가 필요하다. 그러나 EachMovie data set에는 그러한 데이터가 존재하지 않아서 본 실험을 위해 1 값을 랜덤 생성하여 본래의 데이터위에 덮어 쓰는 방법을 고안하여 synthetic data를 생성하였다.



<그림 1> Plot of original data
10895/(254×757)×100=5.67% density



<그림 2> Plot of synthetic data
26.74% density

<그림 1>과 <그림 2>는 각각 본래 데이터와 새롭게 생성된 데이터를 Plot해 본 것이다. 부득이 <그림 2>의 경우 데이터의 일부분만을 표시한 것이다. 데이터를 새로 생성할 때 density를 약 25%로 잡았다. 그 이유는 평균적으로 물건을 구입하고, 평가를 하는 비율이 20%정도밖에 되지 않을 것이라는 가정하에 density를 5.67%에서 이의 5배 정도로 잡은 것이다. (실제로는 평가비율이 더 낮아서, density의 비율이 5배가 넘어갈 것이라고 예상한다.)

제안된 수식과 데이터로 예측한 값은 0과 1사이의 연속적인 값을 갖게 된다. 따라서 test set의 각 유저에 대하여 예측값이 높은 순으로 Top_N 제품을 추천할 수 있으며 정확도를 측정하는데 있어서 기존의 방법과 동일한 척도를 사용할 수 있게 된다. 즉, 각 사용자에 대하여 Top_N의 제품에 대해서 1의 값을 부여

하고, 나머지 아이템에 대해서는 0의 값을 부여하는 방법으로 예측된 행렬을 생성할 수 있으며 그런 후 식(8)의 척도에 적용한다. <표 4>는 제안된 방법에 의한 필터링의 Precision을 나타낸 것이다. 평균 67.90(%)의 Precision으로써 기존의 방법에 비해 두 배 이상의 Precision을 보임을 알 수 있다.

<표 4> Precision of new method

Top_N	5	6	7	8	평균
Precision(%)	69.47	67.54	67.48	67.12	67.90

5. 결론 및 토론

본 논문에서 고객평가 데이터의 희소성을 줄이기 위한 방법으로 고객평가 대신 제품의 구매여부만을 나타내는 이분행렬을 적용해 보았다. EachMovie data set과 synthetic data를 혼합해 실험한 결과 본 논문에서 제시한 방법의 정확도가 기존의 방법보다 두 배 정도 높아졌다. 추후 연구를 통해 본 논문에서 제안한 방법론을 실제 데이터에 적용해 보고자 한다.

이분행렬 방법론을 적용하게 되면 고객의 주관적인 평가치를 수집하는데 걸리는 시간과 비용을 줄일 수 있게 되고, 더 많은 정보를 사용할 수 있게 됨으로써 추천시스템의 예측력이 증가되리라 본다.

또한, 사용된 데이터가 이분 행렬이라는 관점에서 보면 연관룰(Association Rule)의 Co-occurrence matrix의 생성이 가능하며, 따라서 연관룰과 이분 행렬을 이용한 협동적 필터링을 혼합한 추천시스템의 연구도 가능할 것이다.[8]

참고문헌

[1]안현철, 한인구, "데이터 마이닝을 활용한 인터넷 쇼핑몰의 상품 추천 시스템 개발", 한국과학기술원 MGSM 02137, (2002)

[2]황병연, "개선된 추천을 위해 클러스트링을 이용한 협동적필터링 에이전트 시스템의 성능", 한국정보처리학회 논문지 제7권 제5호 1599-1608, (2000)

[3]Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl, "Item-Based Collaborative Filtering Recommendation Algorithms", *ACM* 1-58113-348-0/01/0005, (2001)

[4]Daniel Billsus and Michael Pazzani. "Learning collaborative information filters", In *Proc. 15th International Conf. on Machine Learning*, pages 46-54. Morgan Kaufmann, San Francisco, CA, 1998(1998)

[5]David Goldberg, David Nichols, Brian Oki, and Douglas Terry. "Using collaborative filtering to weave an information tapestry" *Communications of the ACM*, 35(12):61-70, (1992)

[6]John S. Breese, David Heckerman, Carl Kadie, "Empirical Analysis of Predictive Algorithms for Collaborative Filtering", Microsoft Research Technical Report, MSR-TR-98-12, (1998)

[7]Michael Pryor "The effect of singular value decomposition on collaborative filtering", Dartmouth College CS Technical Report, (1998)

[8]Weiyang Lin, Sergio A.Alvarez, Carolina Ruiz, "Collaborative Recommendation via Adaptive Association Rule Mining", Workshop Web Mining for E-Commerce (WEBKDD'00).

[9]http://www.netperceptions.com

[10]http://www.research.digital.com/SRC/eachmovie/