JKIICE Journal of the Korea Institute of Information and Communication Engineering

한국정보통신학회논문지(J. Korea Inst. Inf. Commun. Eng.) Vol. 17, No. 11 : 2602~2608 Nov. 2013

항목 간 선호도 차이를 이용한 영화 추천 방법

오세창[1] · 최민[2*]

A Movie Recommendation Method Using Rating Difference Between Items

Se-Chang Oh[1] · Min Choi[2*]

[1] Department of Information & Communication, Sejong Cyber University, Seoul 143-150, Korea
[2] Department of Information and Communication Engineering, Chungbuk National University, Cheongju 361-763, Korea

요 약

영화 추천 문제에 대한 해법으로 사용자 기반 추천 방법과 항목 기반 추천 방법이 연구되어왔다. 그러나 이들은 각각 희박성의 문제와 사용자의 선호도를 반영하지 못한다는 문제를 안고 있다. 이러한 문제들을 해결하기 위해서 유사도의 개념을 이용해 두 가지 방법을 조합하는 연구가 있으나 계산해야 할 파라메타 수가 많아 현실적으로 희박성의 문제에서 자유롭지 못하다. 본 연구에서는 이러한 문제를 보완하기 위하여 항목 간 선호도 차이를 이용한 추천 방법을 제안한다. 이 방법은 계산해야 할 파라메타 수가 적어 희박성의 문제에서 비교적 자유롭다. 또한 파라메타 계산에 사용자들이 평가한 선호도를 반영함으로써 보다 정확한 결과를 얻을 수 있다. 실험 결과 제안된 방법은 초기에는 오류가 크지만 빠르게 성능이 안정화되는 것을 보여준다. 또한 유사도를 이용한 기존의 추천 방법과 비교하여 평균 오류를 0.0538 낮추는 결과를 보였다.

ABSTRACT

User-based and item-based method have been developed as the solutions of the movie recommendation problem. However, these methods are faced with the sparsity problem and the problem of not reflecting user's rating respectively. In order to solve these problems, there is a research on the combination of the two methods using the concept of similarity. In reality, it is not free from the problem of sparsity, since it has a lot of parameters to be calculated. In this study, we propose a recommendation method using rating difference between items in order to complement this problem. This method is relatively free from the problem of sparsity, since it has less parameters to be calculated. And it can get more accurate results by reflecting the users rating to calculate the parameters. In experiments for the proposed method, the initial error is large, but the performance has been quickly stabilized after. In addition, it showed a 0.0538 lower average error compared to the existing method using similarity.

키워드 : 협업 필터링, 추천 시스템, 데이터 마이닝, 희박성, 전자 상거래

Key word : Collaborative Filtering, Recommender System, Data Mining, Sparsity, Electronic Commerce

접수일자 : 2013. 09. 17 심사완료일자 : 2013. 10. 09 게재확정일자 : 2013. 10. 21

*** Corresponding Author** Min Choi (E-mail:mchoi@cbnu.ac.kr, Tel:+82-43-261-3367)
Department of Information and Communication Engineering, Chungbuk National University, Cheongju 361-763, Korea

Open Access http://dx.doi.org/10.6109/jkiice.2013.17.11.2602 print ISSN: 2234-4772 online ISSN: 2288-4165

Ⅰ. 서 론

영화 추천 문제는 상품 추천 문제 중 대표적인 문제로, 이 문제를 해결하기 위한 방법으로는 크게 항목에 기반을 둔 방식과, 사용자에 기반을 둔 방식으로 나눌 수 있다. 또한 이 두 가지 방법을 결합한 혼합 방식도 연구되고 있다.

항목에 기반을 둔 방식은 내용 기반 방식이라고도 하며, 사용자의 선호도가 높은 항목과 특징이 비슷한 새로운 항목을 추천한다 [1]. 여기에서 비슷한 항목을 찾기 위해서 특징을 사용해 분류하거나, 항목들을 군집화하기도 한다. 이 방식은 데이터가 충분치 않은 상황에서도 비교적 잘 작동한다는 장점이 있다. 반면 이 방법은 항목의 본질적인 특성을 나타내는 특징들을 사용하면 이것이 선호도와 연관성이 높을 것이라는 가정을 전제로 한다. 그러나 이를 사용한 항목 간 유사도 지표는 실제로 사용자들의 선호도를 전혀 반영하지 못한다는 단점이 있다.

사용자에 기반을 둔 방식은 협업 필터링이라고도 하며, 사용자들 간의 유사도를 미리 조사하여, 목표 사용자와 유사한 사용자들이 좋게 평가한 항목들을 추천하는 방식이다 [2]. 이 때 사용자들 간의 유사도는 항목에 대한 사용자의 선호도 정보를 사용해서 구함으로써 사용자의 취향을 충분히 반영할 수 있다 [3]. 그러나 이 방식은 데이터가 충분치 않은 상황에서는 사용자 간의 유사도 계산이 불완전해 신뢰도를 확보하기 어렵다. 이러한 문제를 완화하기 위해 인구통계학적 정보를 사용하여 사용자들을 분류하기도 한다 [4].

혼합 방식은 항목에 기반을 둔 방식과 사용자에 기반을 둔 방식을 결합함으로써 두 방식의 장점을 취하려는 새로운 시도이다. 영화 추천 문제에서 혼합 방식을 적용한 예로서 항목 기반 방식인 장르 정보와 사용자 기반 방식인 사용자의 주소 정보를 같이 사용한 예가 있다 [5]. 그러나 같은 장르에 속한 영화들이 비슷한 평을 받을 것이라는 가정은 근거가 약하고, 인구통계학적 분류가 선호도가 비슷한 사람들의 분류라고 보기 어렵다는 점에서 보다 근본적인 접근이 이루어져야 한다. 논문 [6]에서는 사용자들이 평가한 선호도 정보만으로 사용자 간, 항목 간 유사도를 구하고, 이를 사용해 사용자 기반 협업 필터링과 항목 기반 협업 필터링 방법을 적절히 통합하는 방법을 제안하였다. 그러나 이 방법에서도 희박성의 문제는 잘 해결되고 있지 못하다.

따라서 본 연구에서는 희박성의 문제를 효과적으로 해결하기 위해서 사용자들의 선호도 정보를 근거로 항목 간 거리를 구하고, 이를 이용한 추천 방법을 제안한다.

Ⅱ. 관련 연구

이 장에서는 논문 [6]에서 제안한 선호도를 이용한 영화 추천 방법을 소개하고, 이 방법의 장단점을 분석한다. 이 방법에서는 피어슨 상관 계수를 사용해서 항목 간 그리고 사용자 간 유사도를 구하고 [7], 이를 사용해 유사 항목 집합과 유사 사용자 집합을 구한 다음, 최종적으로 유사집합에 속한 사용자 또는 항목의 선호도를 참고하여 선호도 예측치를 구한다. 이때 앞에서 계산된 유사도는 최종 선호도 계산을 위해 각각의 예측치를 결합할 때 각 예측치에 대한 중요도로 사용된다.

2.1. 유사도 계산

이 방식에서는 먼저 사용자들이 항목에 대해 선호도를 평가한 정보를 바탕으로 사용자 간의 유사도와 항목 간의 유사도를 구한다. 먼저 사용자 기반 협업 필터링에서 사용자 a와 사용자 u 간의 유사도 Sim(a, u)는 다음과 같은 수식에 의해 구해진다.

$$Sim(a,u) = \frac{\mathrm{Min}(|I_a \cap I_u|, \gamma)}{\gamma} \cdot \frac{\sum_{i \in I_a \cap I_u} (r_{a,i} - \bar{r}_a) \cdot (r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i \in I_a \cap I_u} (r_{a,i} - \bar{r}_a)^2} \cdot \sqrt{\sum_{i \in I_a \cap I_u} (r_{u,i} - \bar{r}_u)^2}} \quad (1)$$

이 식에서 I_u는 사용자 u가 평가한 항목들의 집합을, $|I_a \cap I_u|$는 사용자 u와 사용자 a가 공통으로 평가한 항목 집합의 크기를, $r_{u,i}$는 사용자 u가 항목 i에 대해 평가한 선호도를, $\bar{r}_u$는 사용자 u의 평균 선호도를 각각 나타낸다. 또한 상수 γ는 $|I_a \cap I_u|$에 대한 문턱치로, 초기에 항목 집합의 크기가 작은 상태에서 유사도를 과대평가하는 것을 막기 위해 사용한다. 이 식의 의미는 사용자 a와 사용자 u가 공통적으로 평가한 모든 항목 i에 대해서 두 사용자가 각각 자신의 평균적인 선호도에 비해서 어

떻게 평가했는지를 피어슨 상관 계수를 사용해 비교한 것이다.

다음으로 항목 기반 협업 필터링에서 항목 *i*와 항목 *j* 간의 유사도 *Sim(i, j)*는 다음과 같은 수식에 의해 구해진다.

$$Sim(i,j) = \frac{\mathrm{Min}(|U_i \cap U_j|, \delta)}{\delta} \cdot \frac{\sum_{u \in U_i \cap U_j} (r_{u,i} - \bar{r}_i) \cdot (r_{u,j} - \bar{r}_j)}{\sqrt{\sum_{u \in U_i \cap U_j} (r_{u,i} - \bar{r}_i)^2} \cdot \sqrt{\sum_{u \in U_i \cap U_j} (r_{u,j} - \bar{r}_j)^2}} \quad (2)$$

이 식에서 U_i는 항목 i를 평가한 사용자들의 집합을, $\bar{r}_i$는 항목 *i*의 평균 선호도를 각각 나타낸다. 또한 상수 δ는 $|I_i \cap I_j|$에 대한 문턱치로, 초기에 사용자 집합의 크기가 작은 상태에서 유사도를 과대평가하는 것을 막기 위해 사용한다. 이 식의 의미는 항목 *i*와 항목 *j*를 모두 평가한 모든 사용자 *u*가두 항목을 각각의 평균적인 선호도에 비해서 어떻게 평가했는지를 피어슨 상관계수를 사용해 비교한 것이다.

2.2. 유사 그룹 선택

앞에서 구한 유사도를 바탕으로 특정 사용자와 성향이 비슷한 사용자들의 집합과, 특정 항목과 유사한 성향을 가진 항목들의 집합을 정의할 수 있다. 먼저 사용자 *u*와 성향이 비슷한 사용자들의 집합 *S(u)*를 다음과 같이 구한다.

$$S(u) = \{u_a | \mathrm{Sim}(u_a, u) > \eta, u_a \neq u\} \quad (3)$$

이 식에서 상수 η는 유사도에 대한 문턱치로, 사용자 *u*와 유사한 성향을 가진 사용자들을 선택하기 위한 기준이 된다.

다음으로 항목 *i*와 성향이 비슷한 항목들의 집합 *S(i)*를 다음과 같이 구한다.

$$S(i) = \{i_k | \mathrm{Sim}(i_k, i) > \theta, i_k \neq i\} \quad (4)$$

이 식에서 상수 Θ는 유사도에 대한 문턱치로, 항목 *i*와 유사한 성향을 가진 항목들을 선택하기 위한 기준이 된다.

2.3. 선호도 예측

이 유사 사용자 집합과 유사 항목 집합에 속한 사용자와 항목을 사용해서 새로운 항목에 대한 선호도 예측 결과를 구하기 위한 식은 다음과 같다.

$$\hat{r}_{u,i} = \lambda \cdot \left(\bar{r}_u + \frac{\sum_{u_a \in S(u)} Sim(u_a, u) \cdot (r_{u_a,i} - \bar{r}_{u_a})}{\sum_{u_a \in S(u)} Sim(u_a, u)} \right) + (1-\lambda) \cdot \left(\bar{r}_i + \frac{\sum_{i_k \in S(i)} Sim(i_k, i) \cdot (r_{u,i_k} - \bar{r}_{i_k})}{\sum_{i_k \in S(i)} Sim(i_k, i)} \right) \quad (5)$$

이 식에서 상수 λ는 사용자 기반의 선호도 예측 값과 항목 기반의 선호도 예측 값을 결합하기 위한 비율이다. 즉 λ가 0.5 보다 작으면 항목 기반의 선호도 예측을 더 중요하게 보고, λ가 0.5보다 크면 사용자 기반의 선호도 예측을 더 중요하게 보는 것이다. 이 식에서 첫 번째 항의 의미는 사용자 *u*의 유사 그룹에 속한 사용자 u_a가 항목 *i*에 대해 평가한 선호도와 자신의 평균적 선호도와의 차이를 가중 평균한 것이다. 여기에서 *u*와 u_a의 유사도를 가중치로 사용한다. 마찬가지로 두 번째 항의 의미는 항목 *i*의 유사 그룹에 속한 항목 i_k가 사용자 *u*에 의해 평가된 선호도와 이 항목의 평균적 선호도와의 차이를 가중 평균한 것이다. 여기에서 *i*와 i_k의 유사도를 가중치로 사용한다.

2.4. 알고리즘 분석

논문 [6]에서 제안한 기존 방법의 실험결과에 따르면 이 방법은 유사도 융합 방법 [8], 스무딩 및 클러스터 기반의 피어슨 상관계수 방법 [9], 측면 모델 [10], 개인 특성 진단 방법 [11], 사용자 기반 피어슨 상관계수 방법 [12] 등 높은 성능의 다양한 영화 추천 방법들에 비해서 오류가 적은 것으로 나타난다.

이 방법에서 사용하는 전체적인 알고리즘은 다음과 같이 기술된다.

이 알고리즘의 근본적인 문제는 매번 계산해야 할 파라메타의 수가 *Sim(a, u)*와 *Sim(i, j)*에 대해 각각 *u*u* 개와 *i*i* 개, 그리고 $\bar{r}_u$와 $\bar{r}_i$에 대해 각각 *u* 개와 *i* 개로 많다는 점이다. 이는 희박성의 문제가 존재함을 의미한다.

표 1. 기존 방법에서 사용된 알고리즘
Table. 1 Algorithm Used in Former Method

```
for all data tuples (u, i, rate, time)
  user_set = {a|I_a ∩ I_u ≠ ∅};
  for all a ∈ user_set
      calculate Sim(a,u) using equation (1);
      select S(u) using equation (3);
  item_set = {j|U_i ∩ U_j ≠ ∅};
  for all j ∈ item_set
      calculate Sim(i,j) using equation (2);
      select S(i) using equation (4);
  calculate r̂_{u,i} using equation (5);
```

또한 이 알고리즘을 구성하는 식 자체에도 문제가 있는데, 식 (1)과 식 (2)의 경우 분모가 0이 될 수가 있다는 점이다. 이 경우 유사도 값이 구해질 수 없다. 실제로 충분한 데이터를 사용해 식을 계산할 수 없는 초기뿐만 아니라 그 이후에도 새로운 사용자와 항목이 추가될 때마다 이 식들의 분모가 0이 되는 경우가 나타나며, 정상적으로 값이 나오더라도 유사도가 낮아 식 (3)과 식 (4)에서 공집합이 구해지는 경우가 상당히 많다. 이는 식 (5)에서 결국 $\bar{r}_u$와 $\bar{r}_i$만을 근거로 선호도를 예측할 수밖에 없게 되며, 그 결과 예측 성능의 저하로 이어진다. 실제로 논문 [6]에서 사용한 문턱치를 그대로 사용하여 실험해보면 식 (3), (4)에서 유사 집합이 공집합이 되는 경우가 각각 99.78%와 98.4%로 나타난다.

마지막으로 이 방법에서는 식 (1), (2), (3), (4) 등 중간 단계에서 계산된 결과들의 적합성을 판단하기 위해, 그리고 식 (5)의 최종적으로 계산된 항들을 결합할 때 결합 비율을 정하기 위해 상수를 사용하였다. 이 상수들은 어떤 절대적인 기준에 의해 정해지는 것이 아니라 실험적으로 정해질 수밖에 없다. 이는 부분적으로는 부적절한 수치일 수 있으며, 따라서 선호도 예측 성능은 제한될 수밖에 없다.

Ⅲ. 제안하는 방법

본 연구에서는 기존 연구 [6]에서 소개한 방법의 문제점들을 효과적으로 개선한 새로운 선호도를 예측 방법을 제안하고자 한다. 이를 위해 먼저 계산해야 할 파라메타의 개수를 줄이고, 계산된 항들을 결합할 때는 상수 대신 적절한 중요도 지표를 사용한다. 이 방법은 항목 간 선호도의 차이를 이용하는 방법으로, 항목 기반 방법이면서도 사용자들의 선호도를 충분히 반영하여 예측치를 구한다.

3.1. 항목 간 선호도의 차이 계산

두 항목 i와 j에 대한 사용자들의 선호도 차이의 평균 $d_{i,j}$는 다음 식에 의해 구할 수 있다.

$$d_{i,j} = \frac{\sum_{a \in U_i \cap U_j} (r_{a,i} - r_{a,j})}{|U_i \cap U_j|} \tag{6}$$

이 식은 주어진 두 항목을 모두 평가했던 각 사용자가 두 항목의 차이를 어떻게 평가했는지에 대한 정보를 통합한 것이라고 볼 수 있다.

3.2. 선호도 예측

식 (6)에서 계산한 선호도의 차이를 이용하면 새로운 항목 i에 대한 사용자 u의 선호도의 예측치 $\hat{r}_{u,i}$를 구할 수 있는데, 이는 다음 식에 의해서 계산될 수 있다.

$$\hat{r}_{u,i} = \frac{\sum_{j \in I_u} (r_{u,j} + d_{i,j})}{|I_u|} \tag{7}$$

이 식에서 항목 j를 이용해 새로운 항목 i에 대한 선호도 예측치를 구할 수 있는데, 이는 사용자 u가 이전에 평가한 항목 j에 대한 선호도 $r_{u,j}$에 항목 i와 항목 j의 선호도의 차이 $d_{i,j}$를 더함으로써 구한다. 이렇게 구한 각 항목 j를 이용한 예측 치들을 단순 평균함으로써 $\hat{r}_{u,i}$를 구할 수 있는데, 이 경우 각각의 예측 치들에 대한 중요도는 모두 상수 1로 적용한 것과 같다. 이렇게 되면 각 항목 j의 상대적인 중요도를 고려하지 못한다는 문제가 있다.

이 문제는 아래 식과 같이 $d_{i,j}$를 구할 때 사용된 데이터의 수 $|U_i \cap U_j|$를 항목 j의 중요도로 보고 가중 평균을 구함으로써 해결할 수 있다.

$$\hat{r}_{u,i} = \frac{\sum_{j \in I_u} \{(r_{u,j} + d_{i,j}) \cdot |U_i \cap U_j|\}}{\sum_{j \in I_u} |U_i \cap U_j|} \quad (8)$$

이는 $d_{i,j}$ 를 구할 때 보다 많은 데이터를 사용해서 계산할수록 신뢰도가 높기 때문이다.

3.3. 알고리즘 분석

제안하는 방법에서 사용하는 알고리즘은 다음과 같이 기술된다.

표 2. 항목 간 선호도의 차이를 이용한 예측 알고리즘
Table. 2 Prediction Algorithm using Difference between Item Ratings

```
for all data tuples (u, i, rate, time)
  for all j ∈ I_u && j ≠ i
      calculate d_{i,j} using equation (6);
  calculate r̂_{u,i} using equation (8);
```

이 방법에서 매번 계산해야 할 파라메타의 수는 $d_{i,j}$ 에 대해 *i*j* 개이다. 이는 논문 [6]에서 사용된 방법에 비해 훨씬 적은 수치로 수행 속도는 물론이고 예측의 정확도 면에서도 상대적으로 유리하다. 또한 사용된 수식 모두 분모가 0이 되는 경우나 합산해야 할 대상 집합이 공집합이 되는 등의 문제가 처음 시작할 때를 제외하고는 전혀 발생하지 않는다. 따라서 논문 [6]에서 사용된 방법에 비해서 상대적으로 안정된 예측 결과를 얻을 수 있다. 마지막으로 식 (8)에서 각 항목 j를 이용한 선호도 예측치 $(r_{u,j} + d_{i,j})$의 상대적 중요도를 정하기 위해 상수 대신 사용된 데이터의 수로 사용하였다.

Ⅳ. 실험 결과

4.1. 실험 환경

본 연구에서 사용한 실험 데이터는 MovieLens 100K dataset [13]으로 사용자의 수가 943명, 항목 즉, 영화의 수가 1,682편, 선호도 평가 데이터의 수가 100,000개이다. 특히 선호도 평가 데이터는 튜플 (사용자 ID, 항목 ID, 선호도, 시간 정보)로 구성되어 있다.

실험 방법은 [6]에서 사용한 방법과 같이 데이터를 둘로 나누고, 먼저 훈련 데이터로 훈련한 후 테스트 데이터를 사용하여 성능을 측정하는 방식을 취하지 않았다. 대신 본 논문에서는 처음부터 각 튜플에 대해 선호도 예측을 하여 그 결과를 성능 측정 지표에 누적하고, 그 데이터를 사용해서 부분적인 훈련을 하는 방법을 반복적으로 진행하였다. 이는 실제로 영화 추천 문제에 적용할 때 지속적으로 새로운 사용자와 새로운 영화가 추가되기 때문에 충분한 데이터를 사용해 미리 훈련시키는 것이 현실적이지 않기 때문이다.

실험 데이터는 시간 정보를 기준으로 정렬한 다음 앞에서부터 차례로 하나씩 꺼내어 선호도 예측을 하였다. 즉, t 시점에서의 선호도 예측은 1부터 t-1 시점까지 들어온 데이터를 근거로 이루어진다.

4.2. 선호도 예측 성능의 평가

본 연구에서 사용한 성능 측정 지표는 MAE (Mean Absolute Error)로 다음 식과 같이 정의된다.

$$\text{MAE} = \frac{\sum_{u,i} |r_{u,i} - \hat{r}_{u,i}|}{N} \quad (9)$$

즉, 주어진 선호도 평가 데이터에 들어있는 선호도 $r_{u,i}$와 식 (8)에 의해 구해진 선호도 $\hat{r}_{u,i}$의 차이를 누적하고 데이터의 수로 나누어 평균을 구한다.

다음 그림은 본 논문에서 제안하는 방법의 각 구간별 성능을 MAE 지표로 나타낸 것이다.

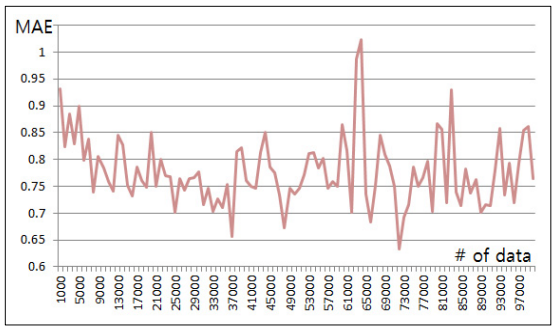

그림 1. 제안하는 방법의 구간 선호도 예측 성능
Fig. 1 Regional Performance of Proposed Rating Prediction Method

이 그래프에서는 전체 데이터를 앞에서부터 1000개씩 나누어 각 구간별로 MAE를 따로 구했다. 그래프를 보면 전체적으로는 초기에 MAE 값이 꾸준히 감소한 이후로 0.7 ~ 0.85의 범위에서 변하고 있으나 중간에 예외적인 구간이 존재한다. 이는 사용자나 항목의 특성이 크게 바뀌는 구간으로 해석되며, 전체적인 성능을 판단하는데 방해가 된다.

따라서 이러한 부분적인 변화의 영향을 줄이고 전체적인 예측 성능을 확인하기 위해서는 구간을 따로 나누지 않고, 다음 그림과 같이 전체 구간에 대한 MAE를 구하는 것이 필요하다.

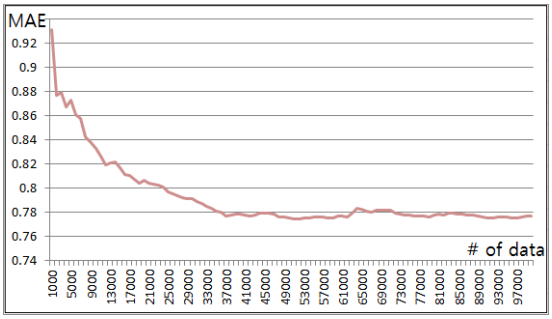

그림 2. 제안하는 방법의 전체 선호도 예측 성능
Fig. 2 Whole Performance of Proposed Rating Prediction Method

이 그래프는 가로 축의 각 시점에 대해 처음부터 그 시점까지의 누적된 모든 데이터를 범위로 MAE를 구한 것이다. 그래프를 보면 처음에는 MAE가 높다가 데이터의 수가 증가하면서 MAE가 하락하여 일정한 수준을 유지한다. 이는 초기에는 선호도 예측에 참고할 만한 데이터가 적어서 예측의 정확도가 떨어졌지만 점차 경험이 쌓이면서 이를 바탕으로 보다 정확한 예측이 가능해 짐을 보여준다.

4.3. 기존 방법과의 비교

본 논문에서 제안하는 방법과 기존 논문 [6]에서 사용한 방법을 비교한 결과는 다음 그림과 같다.

그래프에서 Former Method로 표기된 곡선은 기존 논문 [6]에서 사용한 방법의 성능을 나타내고, Proposed Method로 표기된 곡선은 식 (8)에 기술된 방법의 성능을 나타낸다. 그림에서 보는 바와 같이 기존 논문에서 소개한 방법은 시간이 지나면서 MAE가 지속적으로 증가하고 있으나, 본 논문에서 제안하는 방법은 일정한 수준을 유지하고 있음을 알 수 있다.

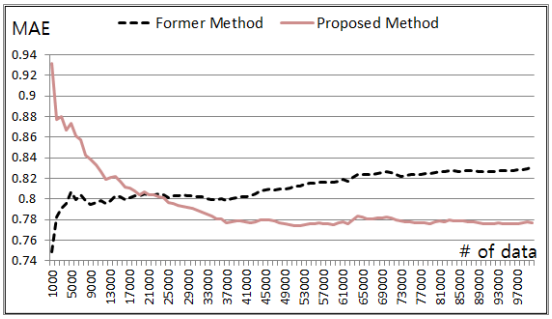

그림 3. 기존 방법과의 비교
Fig. 3 Comparison with Former Method

결과적으로 100,000개의 데이터에 대해서 성능평가를 한 결과 본 논문에서 제안하는 방법은 기존 논문에서 소개한 방법에 비해 MAE가 0.0538 낮게 나타난다.

V. 결 론

본 연구에서는 항목 간 선호도 차이를 이용한 영화 추천 방법을 제안하였다. 이 방법은 계산해야 할 파라메타 수가 적어 희박성의 문제에서 비교적 자유롭고, 파라메타를 계산할 때 사용자들이 평가한 선호도 정보를 사용함으로써 보다 정확한 결과를 얻을 수 있다. 실험 결과 제안한 방법은 초기에는 오류가 크지만 빠르게 성능이 안정화되는 것을 보여준다. 또한 유사도를 이용한 기존의 추천 방법과 비교하여 평균 오류를 0.0538 낮추는 결과를 보였다.

앞으로 예측 결과를 개선하기 위해 보다 다양한 예측 방법들을 통합하기 위한 방법을 개발하고자 한다. 또한 좀 더 큰 데이터를 사용한 실험을 통해서 보다 현실적으로 의미 있는 영화 추천 방법을 개발하고자 한다.

REFERENCES

[1] S. H. Jo, "Weight Recommendation Technique Based on Item Quality To Improve Performance of New User Recommendation and Recommendation on The Web," Ph.

D. dissertation, Hannam University Graduation School, 2008.

[2] S. J. Lee and T. R. Jeon, G. D, Baek, S. S. Kim, "A Movie Rating Prediction System of User Propensity Analysis based on Collaborative Filtering and Fuzzy System," *Journal of Korean institute of intelligent systems*, vol. 19, no. 2, pp. 242-247, 2009.

[3] Hee-Choon Lee, Seok-Jun Lee, Sun-Ok Kim, "A Study on improvements of prediction accuracy using additional information in collaborative filtering," in *Proceeding of The KITS Conference 2009*, pp. 349-352, 2009.

[4] G.Lekakos and G.M.Giaglis, "Improving the Prediction Accuracy of Recommendation Algorithms : Approaches Anchored on Human Factors," *Interacting with Computers*, vol. 18, pp. 410-431. 2006.

[5] Kyung-Rog Kim, Jaehee Byeon, Nammee Moon, "Collaborative Filtering Design Using Genre Similarity and Preffered Genre," in *Proceeding of The KSCI Conference 2011*, vol. 16, no. 4, pp. 161-170, April 2011.

[6] Hao Ma, Irwin King and Michael R. Lyu, "Effective Missing Data Prediction for Collaborative Filtering," *in Proceeding of SIGIR 2007*, pp. 39-46, 2007.

[7] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "Grouplens: An open architecture for collaborative filtering of netnews," *in Proceeding of ACM Conference on Computer Supported Cooperative Work*, 1994.

[8] J. Wang, A. P. de Vries, and M. J. Reinders, "Unifying user-based and item-based collaborative filtering approaches by similarity fusion," *in Proceeding of SIGIR*, 2006.

[9] G.-R. Xue, C. Lin, Q. Yang, W. Xi, H.-J. Zeng, Y. Yu, and Z. Chen, "Scalable collaborative filtering using cluster-based smoothing," *in Proceeding of SIGIR*, 2005.

[10] T. Hofmann, "Latent semantic models for collaborative filtering," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 89 - 115, 2004.

[11] D. M. Pennock, E. Horvitz, S. Lawrence, and C. L. Giles, "Collaborative filtering by personality diagnosis: A hybrid memory- and model-based approach," *in Proceeding of UAI*, 2000.

[12] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," *in Proceeding of UAI*, 1998.

[13] GroupLens Research. MovieLens Data Sets [Internet]. Available: http://www.grouplens.org/node/73.

오세창(Se-Chang Oh)

1997년 2월 KAIST 전산학과 공학박사
1995년 3월 ~ 1999년 1월 LG종합기술원 선임연구원
1999년 2월 ~ 2000년 2월 (주)인지소프트 이사
2000년 3월 ~ 2003년 8월 아주대학교 정보통신 전문대학원 조교수 대우
2004년 1월 ~ 현재 세종사이버대학교 정보통신학과 조교수
※관심분야 : 패턴인식, 데이터 마이닝, 빅 데이터

최민(Min Choi)

2001년 2월 : 광운대학교 전자계산학과 학사
2003년 2월 : 한국과학기술원 전산학과 석사
2009년 2월 : 한국과학기술원 전산학과 박사
2008년 3월~2010년 2월 : 삼성전자 책임연구원
2010년 3월~2011년 8월 : 원광대학교 교수
2011년 9월~현재 : 충북대학교 교수
※관심분야 : 임베디드 시스템, 운영체제, 컴퓨터구조