

모바일 기기에서 개인화 추천을 위한 실시간 선호도 예측 방법에 대한 연구

이학민[†], 엄종석^{**}

A Study on the Real-Time Preference Prediction for Personalized Recommendation on the Mobile Device

Hak Min Lee[†], Jong Seok Um^{**}

ABSTRACT

We propose a real time personalized recommendation algorithm on the mobile device. We use a unified collaborative filtering with reduced data. We use Fuzzy C-means clustering to obtain the reduced data and Kohonen SOM is applied to get initial values of the cluster centers. The proposed algorithm overcomes data sparsity since it extends data to the similar users and similar items. Also, it enables real time service on the mobile device since it reduces computing time by data clustering. Applying the suggested algorithm to the MovieLens data, we show that the suggested algorithm has reasonable performance in comparison with collaborative filtering. We developed Android-based smart-phone application, which recommends restaurants with coupons and restaurant information.

Key words: Unified Collaborative Filtering, Fuzzy C-Means, Real-Time Preference Prediction

1. 서 론

온라인 환경 및 스마트 기기의 보급, SNS 활성화, 그리고 사물 인터넷의 확산 등으로 정형 또는 비정형 데이터가 폭증하였고 기존의 데이터 저장, 관리, 분석기법은 한계를 보였다. 이렇게 축적된 대용량의 데이터를 의미있게 분석할 수 있는 분석기법 및 추천 분야에 대한 연구와 개발이 지속적으로 이루어지고 있으며, 축적된 데이터를 통해 제공되는 많은 정보에서 온라인 사용자들의 정보 검색 및 선택에 편리성을 제공하고자 개인화 추천 서비스가 개발되어 왔다 [1][2]. 개인화 추천이라는 것은 사용자의 웹페이지 사용 행태와 개인적인 관심 혹은 개인의 상황 정보로부터 얻은 정보를 이용하여 특정 개인이나 그룹에게

적합한 정보나 서비스를 온라인으로 제공하는 행위를 말하며, 개인화의 목적은 개인이 원하거나 혹은 필요한 정보를 명시적으로 질의하지 않아도 이전의 사용행태와 개인 정보를 이용하여 미리 예측하여 제공하는데 있다[3]. 개인화 추천은 사용자의 개인 정보, 사용자 선호도나 웹페이지 이용 기록 등의 축적된 대용량의 데이터를 사용하여 특정 아이템에 대한 선호도를 예측하고 예측된 선호도에 기반하여 아이템을 추천하는 것으로 넷플릭스, 구글 뉴스, 아마존과 같은 웹사이트에서 이미 적용되고 있다[4].

모바일 기기에서 개인화 추천은 많은 사용자들의 누적된 대용량의 데이터를 이용하기 때문에 실시간 선호도 추천에 한계가 있다. 본 연구에서는 사용자에게 사용자의 선호도에 기반하여 모바일 기기에

※ Corresponding Author : Jong Seok Um, Address: (02876) Samseongyo-ro 116, Seongbuk-gu, Seoul, Korea, TEL : +82-2-760-4133, FAX : +82-2-760-4488, E-mail : jsum@hansung.ac.kr
Receipt date : Jan. 13, 2017, Approval date : Jan. 31, 2017

[†] Digital Innovation Team, Shinhan Data System
(E-mail : ley0812@hansung.ac.kr)

^{**} Division of Computer Engineering, Hansung University
※ This research was financially supported by Hansung University.

서 실시간 추천이 가능하도록 만들기 위하여 Fuzzy C-Means Clustering을 이용하여 데이터의 수를 적절한 수의 대표 데이터로 군집화 하여 연산 시간을 단축하였다. 또한 군집화 결과는 누락 값으로 발생하는 데이터 희소성 문제와 데이터의 증가에 따른 scalability 문제를 해결할 수 있다. Fuzzy C-Means 군집화는 초기 값에 민감하여 목적함수의 지역 최소 값을 갖는 해로 수렴할 가능성이 있기 때문에 Konohen SOM을 이용하여 군집화에 필요한 군집 중심의 초기 값을 정하였다. 사용자 수와 아이템 수가 큰 경우 많은 누락 값이 존재하며, 또한 데이터의 일부분을 사용하는 협력적 필터링에서는 데이터의 희소성으로 인해 추천 결과에 대한 신뢰도가 낮을 수 있다. 이 문제를 해결하기 위해 기존의 대표적인 협력적 필터링의 두 가지 방법을 융합하여 선호도 예측에 유사한 사용자의 유사한 아이템에 대한 선호도를 추가적으로 사용하여 추천의 신뢰도를 높였다. 제2절에서는 협력적 필터링을 소개하였고, 제3절에서는 제안 알고리즘을 설명하였으며, 제4절에서는 Movie Lens 데이터를 이용하여 기존의 방법과 성능 비교를 수행하였고, 제5절에서는 제안 알고리즘을 모바일 기기에 적용하여 안드로이드 기반 추천시스템을 구현하였으며, 추천된 매장의 정보 제공 기능과 쿠폰을 제공하는 비콘 시스템도 같이 구현하였으며 끝으로 결론을 맺었다.

2. 관련연구

협력적 필터링은 Memory-based와 Model-based로 분류될 수 있다. Memory-based 방식은 기존에 있는 사용자의 평가 데이터를 이용하는 방법으로 사용자 기반의 협력적 필터링, 아이템 기반의 협력적 필터링 등이 있다. Model-based 방식은 훈련된 데이터를 사용하여 선호도 예측을 위한 모델을 생성하는 방법으로 Decision trees[5], Bayesian-belief Network 기반의 협력적 필터링[6], aspect model[7] 등이 있으며, 본 연구에서는 Memory-based 방법인 협력적 필터링을 수정한 융합된 협력적 필터링을 사용한다.

2.1 사용자 기반 협력적 필터링

사용자 기반 협력적 필터링은 사용자의 아이템에

대한 선호도 값을 예측할 때 유사한 사용자들의 아이템에 대한 선호도 값을 이용한다. 먼저 사용자 간의 유사도를 계산한 후에, 이 유사도를 가중치로 사용하여 사용자의 특정 아이템에 대한 선호도 값을 예측한다. 여기서 사용하는 유사도는 비선호의 의미까지 포함하기 때문에 코사인 유사도 대신 피어슨 상관계수를 사용하며, 이 상관계수를 가중치로 사용하여 선호도를 예측하였다. 사용자 k의 아이템 m에 대한 선호도 값 예측 수식은 다음 수식 (1)과 같다[8].

$$p_{k,m} = \bar{v}_k + \frac{\sum_{a=1}^n (v_{a,m} - \bar{v}_a) \times s_{k,a}}{\sum_{a=1}^n s_{k,a}} \quad (1)$$

n은 사용자의 수이고, $v_{a,m}$ 는 사용자 a의 아이템 m에 대한 선호도 값이며, $s_{k,a}$ 는 사용자 k와 사용자 a의 유사도이다. 사용자 k의 아이템 m에 대한 선호도 예측치는 사용자 k의 선호도 평균에 사용자 k와 사용자 a의 유사도를 가중치로 하여 사용자 a의 선호도를 수정한 것이다. 여기서 사용하는 사용자 a의 선호도는 사용자 a의 아이템 m에 대한 선호도에서 사용자 a의 선호도 평균을 빼 준 것으로 사용자 a의 평가 성향을 반영한 선호도이다.

2.2 아이템 기반 협력적 필터링

아이템 기반 협력적 필터링은 사용자의 아이템에 대한 선호도 값을 사용자 대신 아이템 간의 유사도를 이용하여 예측한다. 유사도 계산식은 사용자 기반 협력적 필터링과 마찬가지로 피어슨 상관계수를 사용하였다. 사용자 k의 아이템 m에 대한 선호도 값 예측 수식은 다음과 같다[9].

$$p_{k,m} = \frac{\sum_{all\ similar\ items,\ b} (s_{m,b} \times v_{k,b})}{\sum_{all\ similar\ items,\ b} (s_{m,b})} \quad (2)$$

위의 식에서는 m과 비슷한 상품들의 집합의 각 상품들에 대한 사용자 k의 선호도 값과 아이템 간의 유사도를 이용한다. $s_{m,b}$ 은 아이템 m과 다른 아이템 b의 유사도 값이고 $v_{k,b}$ 은 사용자 k의 아이템 b에 대한 선호도 값이다. 사용자 k의 아이템 m에 대한 선호도 예측치는 다른 아이템에 대한 사용자 k의 선호도에 아이템 m과 다른 아이템간의 유사도를 가중치로 준 가중평균 선호도이다.

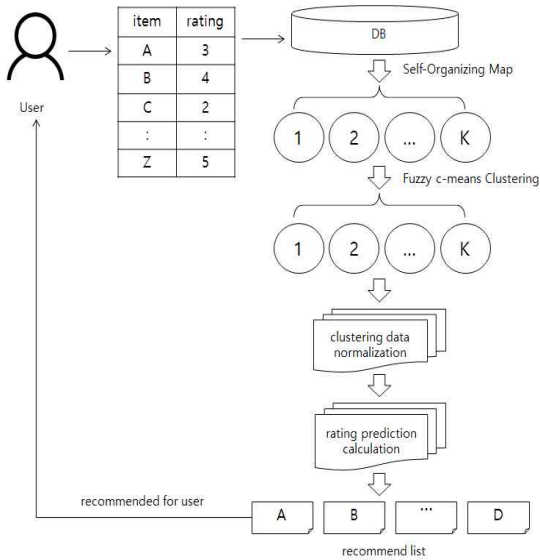


Fig. 1. Flowchart for recommend system.

3. 제안 알고리즘

3.1 시스템 구성도

시스템의 구성은 Fig. 1에서 볼 수 있듯이, 사용자가 만약 어떠한 아이템에 대해 선호도 값을 준다면 데이터베이스에 선호도가 저장되거나 갱신된다. 이렇게 사용자가 입력한 선호도 전체 데이터를 적절한 수의 군집으로 그룹화하기 위하여 Konohen SOM을 이용한 초기 군집을 형성하고 Fuzzy C-Means를 이용하여 군집화 시켜준다. 군집화 된 데이터를 정규화를 시켜주고 제안 알고리즘을 이용하여 아이템에 대한 선호도를 예측하여 우선순위가 높은 아이템들을 사용자에게 추천한다.

3.2 Self-Organizing Map(SOM)

추천 시스템의 연산 시간을 줄이기 위해 아이템들에 대해 유사한 선호도를 갖는 사용자들을 같은 집단으로 군집화 시켜준 후에 협력적 필터링을 적용한다. Fuzzy C-means 군집화 방법은 군집 초기 값과 잡음에 민감한 문제점이 있다. 군집 초기 값에 대한 문제점을 해결하기 위해 Konohen SOM 학습을 이용하여 군집의 초기 중심 벡터를 설정하였다[10]. SOM은 신경회로망의 일종으로 입력층과 경쟁층으로 구성되고, 하나의 벡터가 입력되면 경쟁층의 각 뉴런은 연결강도 벡터와 입력벡터의 거리를 계산하고, 가장 가까운 뉴런이 승리하여 입력벡터에 대하여 학습을 하게 된다. 아래 Fig. 2는 SOM의 수행 과정을 나타낸 것이다.

3.3 Fuzzy C-means 군집화

SOM을 수행한 결과 출력된 초기 중심 벡터를 통해 Fuzzy C-Means 알고리즘을 수행한다. Fuzzy C-Means 알고리즘은 데이터를 군집으로 분류할 때 하나의 군집에 속해져있는 각각의 데이터 점의 군집 소속 정도를 일일이 열거한 데이터 분류 알고리즘이다. 각각의 데이터는 하나 이상의 군집에 속할 수 있으며 소속 정도는 확률로 표시한다. Fuzzy C-Means는 다음과 같은 오류제곱의 합(Sum of squared error, SSE)을 최소화하는 것을 목적으로 한다.

$$SSE(C_1, C_2, \dots, C_k) = \sum_{j=1}^k \sum_{i=1}^m w_{ij}^p \text{dist}(x_i, c_j)^2 \quad (3)$$

여기서 c_j 는 j번째 군집의 중심점이며, p는 가중치의 영향을 조정하는 1과 ∞ 값을 가지는 지수이다. p

Step1. Set learning rate $\alpha(t)$ and initialize weight matrix.

Step2. While stopping condition do step Step3 to Step7.

Step3. For each input vector x, do step Step4 to Step6.

Step4. For each j, compute the squared Euclidean error distance as follows

$$D(j) = \sum (w_{ij} - x_i)^2, \text{ where } i = 1, \dots, n \text{ and } j = 1, \dots, n$$

Step5. Now find index j, when D(j) is minimum.

Step6. For all the unit j, update weights as follows

$$w_{ij}^{new} = w_{ij}^{old} + \alpha(t)[x_i - w_{ij}^{old}]$$

Step7. Test the stopping condition.

Fig. 2. Process of the SOM.

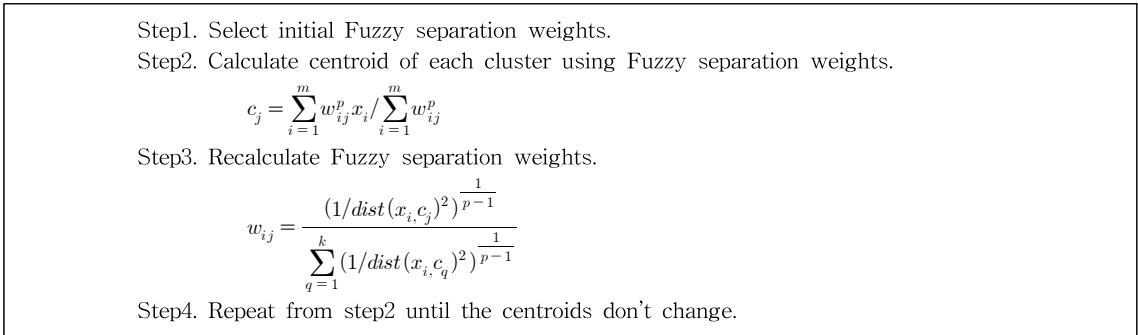


Fig. 3. Clustering process of the Fuzzy C-Means.

의 값이 1에 가깝게 정해지면 Fuzzy C-Means는 K-means와 유사한 행태를 보이고, 반대로 p의 값이 커지면 모든 군집의 중심점이 전체 데이터의 전역적인 중심점에 근접하게 된다. 우리는 가중치 갱신 수식을 간단하게 만들기 위해 p의 값을 2로 설정하였다. 중심점 c_j 와 점 x_i 사이의 거리는 Euclidean 거리로 계산한다. Fuzzy C-Means 군집화 과정은 다음 Fig. 3과 같다.

모든 군집화 과정이 종료될 때, k개 군집의 중심 벡터가 생성되고, 이 중심 벡터가 군집을 형성하는 사용자들의 축약된 대표 선호도가 되며, 이를 사용하여 선호도 예측을 수행한다. k개의 중심 벡터가 나온 상태에서 중심 벡터간의 거리를 측정하여 특정한 임계값 미만이라면 같은 중심 벡터로 인지하여 중심 벡터를 하나로 합쳐주었다.

3.4 선호도 예측

군집화 결과, 각 군집의 중심 벡터를 통해서 사용자의 아이템에 대한 선호도를 예측한다. 선호도를 예측하기 전에 모든 데이터를 정규화 한다. 모든 사용자의 평가 기준은 동일하지 않기 때문이다. 어떤 사용자는 극단적인 값을 통해 평가를 할 수 있고, 또 어떤 사용자는 중간 값에서 크게 벗어나지 않게 평가를 할 수 있다. 사용자 a의 아이템 b에 대한 선호도 정규화 식은 다음과 같다.

$$p_{k,m}(x_{a,b}) = x_{a,b} - (\bar{x}_a - \bar{x}_k) - (\bar{x}_b - \bar{x}_m) \quad (4)$$

정규화된 데이터를 사용하여, 알려지지 않은 선호도 값을 예측하기 위해 수정된 융합 협력적 필터링을 사용한다. 관련연구에서 언급했던 사용자 기반과 아이템 기반 협력적 필터링은 Fig. 4와 같이 전체 데이

터를 나타내는 User-Item 행렬에서 누락 값을 제외하고 k번째 행과 m번째 열인 데이터 즉, 전체 데이터 중에서 일부분을 사용하기 때문에 데이터 희소성 문제로 인해 추천에 대한 신뢰도가 낮을 수 있다. 본 연구에서는 이와 같은 데이터의 희소성을 극복하고, 예측의 정확성을 높이기 위해 유사한 사용자들의 유사한 아이템에 대한 선호도를 추가로 사용하여 사용자 선호도를 예측하였다.

Fig. 4에서 사용자 k의 아이템 m에 대한 선호도 예측 값 $\hat{x}_{k,m}$ 을 구하려고 한다고 했을 때, 사용자 기반 협력적 필터링은 유사한 사용자들의 아이템 선호도 값인 m열의 집합 $SU_{k,m} = \{x_{a,m} | u_a \in S_u(u_k)\}$ 의 값들을 사용한다. 아이템 기반 협력적 필터링은 사용자의 유사한 아이템들의 선호도 값인 k행의 집합 $SI_{k,m} = \{x_{k,b} | i_b \in S_i(i_m)\}$ 의 값들을 사용한다. 위의 두 가지 방법으로 선호도 값을 예측하면 데이터 희소성 문제가 발생할 가능성이 있어 추가적으로 User-Item 행렬의 나머지 부분인 유사한 사용자들의 유사한 아이템에 대한 선호도 값들의 집합인 $SUI_{k,m} = \{x_{a,b} | u_a \in S_u(u_k), i_b \in S_i(i_m), a \neq k, b \neq m\}$ 의 값

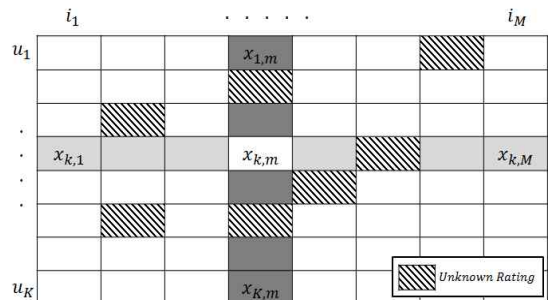


Fig. 4. User-Item Matrix.

들을 사용한다. 이 집합의 선호도 값들과 예측 값의 유사도를 구하기 위해 사용자 간 유사도와 아이템 간 유사도를 혼합하여 융합 협력적 필터링을 사용한다. 사용자 간 유사도를 $s_u(u_k, u_a)$ 라고 하고 아이템 간 유사도를 $s_i(i_m, i_b)$ 라고 했을 때 수식은 다음과 같다[11].

$$s_{ui}(x_{k,m}, x_{a,b}) = \frac{1}{\sqrt{(1/s_u(u_k, u_a))^2 + (1/s_i(i_m, i_b))^2}} \quad (5)$$

선호도 $x_{k,m}$ 값을 예측하는 수식은 아래 수식 (6)과 같다.

$$\hat{x}_{k,m} = \sum_{x_{ab}} p_{k,m}(x_{a,b}) w_{k,m}^{a,b} \quad (6)$$

$$w_{k,m}^{a,b} = \begin{cases} \frac{s_u(u_k, u_a)}{\sum_{x_{ab} \in SU} s_u(u_k, u_a)} \lambda(1-\delta) & x_{a,b} \in SU \\ \frac{s_i(i_m, i_b)}{\sum_{x_{ab} \in SI} s_i(i_m, i_b)} (1-\lambda)(1-\delta) & x_{a,b} \in SI \\ \frac{s_{ui}(x_{k,m}, x_{a,b})}{\sum_{x_{ab} \in SUI} s_{ui}(x_{k,m}, x_{a,b})} \delta & x_{a,b} \in SUI \end{cases}$$

(6)번 수식에서 $p_{k,m}(x_{a,b})$ 는 수식 (4)에 의해서 사용자 a의 아이템 b에 대한 정규화 된 선호도이며, 가중치인 $w_{k,m}^{a,b}$ 의 모든 값의 합은 1이 되어야 하며, $w_{k,m}^{a,b}$ 은 (6)번 수식에 따라 계산을 해준다. 이 가중치를 이용하여 세 가지 유사도를 혼합하여 선호도 값을 예측한다. 가중치 λ 와 δ 은 예측 방법을 제어할 수 있다. 사용자 기반 유사도와 아이템 기반 유사도는 λ 에 의해서 조절되고, δ 는 혼합된 유사도를 조절한다. λ 가 커지면 사용자 기반 유사도가 강조되고, λ 가 작아지면 아이템 기반 유사도가 강조된다. 만약 λ 가 1이라면 사용자 기반 협력적 필터링이고, 0이라면 아이템 기반 협력적 필터링이 된다. 한편 δ 가 0에 가까워지면 혼합된 유사도는 사용되지 않는다. 여기에서는 Jun Wang의 연구에서 교차검증을 통하여 나온 최적의 가중치인 $\lambda = 0.7$ 과 $\delta = 0.7$ 을 사용하여 실험하기로 하였다.

4. Simulation

제안된 알고리즘의 성능 평가를 위해 MAE(Mean Absolute Error)를 사용한다. MAE는 예측된 값들이 실제 값들과 평균적으로 어느 정도 유사한지 나타내는 지표이다. 추천 시스템은 MAE값이 작을수록 우

수하다고 평가한다. MAE 수식은 다음과 같다.

$$MAE = \frac{\sum_{k,m} |x_{k,m} - \hat{x}_{k,m}|}{L} \quad (7)$$

$x_{k,m}$ 은 실제 선호도 값이고, $\hat{x}_{k,m}$ 은 선호도 예측 값이다. L은 값의 총 개수이다. 여기에서는 미네소타 대학의 Group Lens의 MovieLens Dataset을 사용하였다[12]. 이 데이터는 1000명의 사용자가 1700개의 영화에 대해 1~5점으로 선호도 값을 입력하였다. 각 사용자는 최소 20개의 영화에 대해 선호도 값을 입력하였고, 총 100,000개의 평점 데이터들로 구성되어있다. 융합된 협력적 필터링의 성능을 검증하기 위해 MovieLens 데이터를 사용하여 10명의 사용자부터 150명의 사용자까지 100개의 아이템에 대한 예측 값을 구하여 MAE를 측정하였으며 결과가 Fig. 5와 Table 1에 있다.

실험 결과를 보면 사용자 기반 협력적 필터링(UBF)과 아이템 기반 협력적 필터링(IBF)보다 융합된 협력적 필터링(UIBF)의 성능이 우수하다는 것을 볼 수 있다. UBF의 MAE 값 평균은 0.7093이고, IBF는 0.815인 것에 반해 융합된 방법인 UIBF는 0.6656

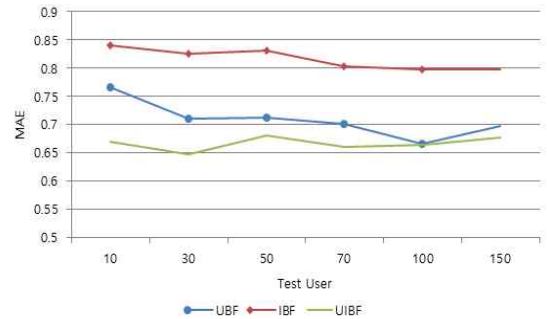


Fig. 5. MAE of three recommending methods(UBF, IBF, UIBF).

Table 1. MAE of three recommending methods(UBF, IBF, UIBF)

Test User	UBF	IBF	UIBF
10	0.7662	0.8393	0.669
30	0.7107	0.825	0.6464
50	0.7126	0.8294	0.68
70	0.7019	0.8024	0.6595
100	0.6661	0.7976	0.6629
150	0.6983	0.7967	0.6759

이라는 결과가 나왔다. UIBF가 UBF보다 6.1% 향상되었고, IBF보다 18.3% 향상되었다.

다음 실험은 군집화 후의 성능을 확인하기 위해 Konohen SOM과 Fuzzy C-means를 적용시켜 MAE 값을 조사하였다. 아이템의 개수는 100로 지정하고, 군집의 개수는 연산 전에 먼저 정해줘야 하기 때문에 사용자 군집 개수를 실험 사용자 수의 5%부터 시작하여 5%씩 증가시켜 25%까지 군집의 수를 증가해서 융합된 협력적 필터링을 통해 실험을 해보았다.

Fig 6과 Table 2에서 보듯이 전체적으로 군집의 개수가 증가할 경우 처음에는 MAE가 증가하다가 점점 안정적으로 감소하는 경향을 보였다. 군집화를 적용하기 전의 경우인 Table 2의 0%에서 100명의 사용자가 100개의 아이템에 대한 예측 결과가 0.6629 이 나온 것에 비해 10%로 군집화의 적용 후인 10개의 군집으로부터 나온 실험 결과가 최적 값이 0.6907 이 나와 약 0.028(4%)정도의 오차의 증가가 발생하였다.

군집화 적용 후 연산시간의 변화는 아래의 Fig. 7를 통해 볼 수 있다. 위에서 실험하였던 100명의 사용자에 대한 100개의 아이템의 예측 값을 군집화를 적용하기 전과 적용한 후 연산시간을 비교하였다.

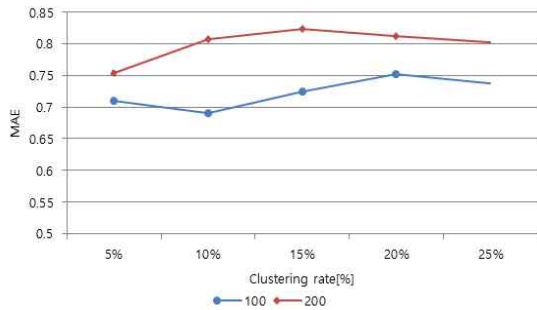


Fig. 6. MAE values according to clustering rate.

Table 2. MAE values according to clustering rate

Clustering rate	Users	
	100	200
0%	0.6629	0.6884
5%	0.7094	0.7543
10%	0.6907	0.8077
15%	0.7232	0.8234
20%	0.7515	0.8118
25%	0.7377	0.8031

Fig. 7에서 보듯이 적용되지 않은 방법의 시간보다 군집화 적용 후의 시간이 줄어든 것을 볼 수 있으며 군집의 개수가 증가할수록 연산시간이 증가한다.

다음 실험은 많은 사용자 데이터를 사용해 보았다. 100개의 아이템에 대해 300명의 사용자부터 사용자를 증가시켜 700명의 사용자를 대상으로 사용자의 수의 10%로 군집을 생성한 뒤, 융합된 협력적 필터링을 이용하여 선호도 예측 값을 구해보았다. 결과는 다음과 같다.

Fig. 8에서 보듯이 사용자의 수를 늘렸음에도 MAE값이 비교적 낮게 나와 우수한 성능을 입증하였다. 현실적으로 전체 데이터를 모두 활용하여 유사도를 구하여 계산하는 것에는 무리가 있다. 그러므로 비용을 고려한다면 군집화를 적용한 후 비교적 적은 데이터를 통해 선호도 값을 예측하여 실시간 추천이 가능하다.

5. 구현

본 논문에서 사용된 추천 시스템을 기반으로 모바일 어플리케이션을 구현해보았다. 이 프로그램은 사

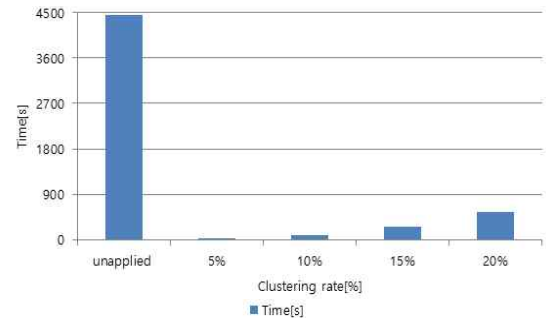


Fig. 7. Time according to clustering rate.

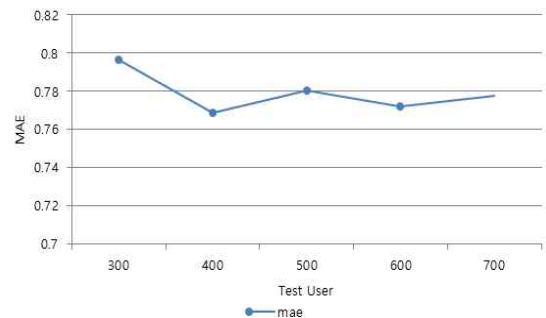


Fig. 8. MAE according to the number of users.

용자에게 매장 추천과 매장 정보 열람 그리고 쿠폰 제공 및 사용을 제공한다. 개발환경은 Windows 8이고 개발도구는 Android SDK를 사용하였다. 추천 시스템을 구축하기 위해 Amazon Web Service EC2 서버를 사용하였고, 효과적인 쿠폰 제공을 위해 블루투스 장치인 비콘을 사용하였다.

추천 시스템은 사용자가 매장에 대한 선호도 값을 주면 그 점수를 기반으로 추천을 해준다. 하지만 사용자가 초기에 평점을 단 한 개도 주지 않았다면 유사도 계산이 불가능하기 때문에 규칙 기반 필터링을 적용하였다. 규칙 기반 필터링은 사용자의 나이, 성별, 거주지를 토대로 유사한 사용자들의 선호도에 따라 평균을 사용하여 선호도 값이 높은 매장을 분야별로 추천을 해준다.

Fig. 9의 User테이블의 정보를 통해 규칙 기반 필터링을 실행하여 비슷한 유저의 값을 활용하여 매장을 추천해준다. 사용자가 선호도 값을 남기면 Rating 테이블에 레코드가 쌓이고 이 정보를 토대로 Kono-hen SOM과 Fuzzy C-Means Clustering이 실행된다. 군집화 된 데이터 행렬을 토대로 선호도 값 예측이 실행된다. 아마존 EC2 서버에서 연산을 실행한 후 사용자에게 추천 매장을 전송해준다.

초기 선호도 데이터를 위해 서울시 성북구 대학로에서 90명의 시민을 대상으로 90개의 매장 중 30개의

매장을 선택하게 하여 선호도 값을 매기도록 하여 선호도를 조사하였다.

Fig. 10의 구현화면에서 첫 번째 화면의 하단에서는 사용자에게 매장을 카테고리별로 추천을 해주고 있다. 가운데 화면에는 비콘 시스템을 통해 받은 쿠폰을 사용하는 창을 볼 수 있다. 마지막 화면은 해당 매장의 정보를 지도와 상세정보를 보여주고 있다.

6. 결 론

본 연구에서는 대용량의 데이터를 이용하여 모바일 기기에서 실시간 추천 시스템 구현하기 위하여 Kohonen SOM과 Fuzzy C-Means Clustering을 사용하여 데이터를 축약시켜 연산시간을 단축하였다. 데이터의 일부분을 사용하는 협력적 필터링이 갖는 데이터의 희소성을 해결하기 위해 사용자 기반 협력적 필터링과 아이템 기반 협력적 필터링을 융합하여 선호도 예측에 사용되는 데이터를 유사한 사용자의 유사한 아이템에 대한 선호도 데이터를 추가하여 데이터를 확장하였으며, 실험 결과에서 보듯이 실시간 추천 결과는 기존의 예측 값이 갖는 오차와 비슷한 결과를 보여 제안 알고리즘의 효용성을 입증하였다. 제안 알고리즘을 이용한 안드로이드 기반 모바일 앱을 제작하였으며 클라우드 서버를 이용하여 실시간 추천 시스템을 구현하였다. 향후 연구과제는 다양한 분야의 아이템에 대한 분야별 선호도를 이용하여 특정 분야 아이템에 대한 선호도를 주어진 조건으로 다른 분야 아이템에 대한 조건부 추천 알고리즘을 개발하는데 있다.

User	Store	Rating
user_id device_num name gender age address	store_id name location picture ...	user_id store_id value

Fig. 9. Database design structure.

REFERENCE

[1] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutierrez, "Recommender Systems Survey," *Knowledge-Based Systems*, Vol. 46, pp. 109-132, 2013.

[2] J.I. Kim, "A Recommendation Method of Similar Clothes on Intelligent Fashion Coordination System," *Journal of Korea Multimedia Society*, Vol. 12, No. 5, pp. 688-698, 2009.

[3] M. Eirnaki and M. Vazirgiannis, "Web Mining for Web Personalization," *ACM Transactions on Internet Technology*, Vol. 3, No. 1, pp. 1-

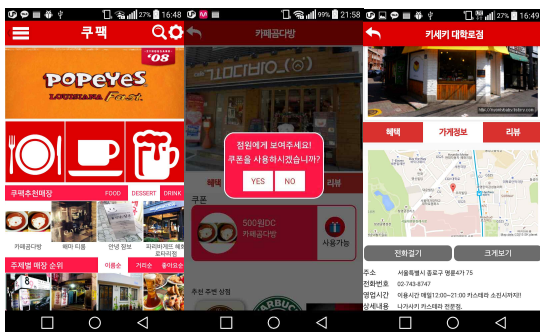


Fig. 10. Mobile application screens.

27, 2003.

[4] G. Linden, B. Smith, and J. York, "Amazon.com Recommendations: Item-to-Item Collaborative Filtering," *IEEE Internet Computing*, Vol. 7, No. 1, pp. 76-80, 2003.

[5] S. Breese, D. Heckerman, and C. Kadie, "Empirical Analysis of Predictive Algorithms for Collaborative Filtering," *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, pp.43-52, 1998.

[6] L.M. Campos, J.M. Fernandez-Luna, J.F. Huete, and M.A. Rueda-Morales, "Combining Content-based and Collaborative Recommendations: A Hybrid Approach Based on Bayesian Networks," *International Journal of Approximate Reasoning*, Vol. 51, Issue 7, pp. 785-799, 2010.

[7] T. Hofmann, "Latent Semantic Model for Collaborative Filtering," *ACM Transactions on Information Systems*, Vol. 22, No. 1, pp. 89-115, 2004.

[8] J.L. Herlocker, J.A. Konstan, A. Borchers, and J. Riedl "An Algorithmic Framework for Performing Collaborative Filtering," *Proceeding of Special Interest Group on Information Retrieval*, pp. 230-237, 1999.

[9] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based Collaborative Filtering Recommendation Algorithms," *Proceedings of the 10th international Conference on World Wide Web* , pp. 285-295, 2001.

[10] K. Zagoris, N. Papamarkos, and I. Koustoudis, "Color Reduction Using the Combination of the Kohonen Self-Organized Feature Map and the Gustafson-Kessel Fuzzy Algorithm," *Machine Learning and Data Mining in Pattern Recognition*, LNAI 4571, pp. 703-715, 2007.

[11] J. Wang, A.P. de Vries, and J.T. Reinders, "Unifying Userbased and Itembased Collaborative Filtering Approaches by Similarity Fusion," *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 6-11, 2006.

[12] MovieLens Dataset, Social Computing Research at the University of Minnesota. <http://www.grouplens.org/> (accessed Aug., 20, 2015).



이 학 민

2016년 2월 한성대학교 멀티미디어공학과 학사
 2016년 11월~현재 신한데이터시스템 디지털혁신팀
 관심분야: 패턴인식, 인공지능, 기계학습



엄 종 석

1982년 연세대학교 응용통계학과 학사
 1984년 연세대학교 응용통계학과 석사
 1991년 Ohio State University (Ph.D in Statistics)

1992년 8월~현재 한성대학교 컴퓨터공학부 교수
 관심분야: 데이터마이닝, 컴퓨터비전, 기계학습