

단계적 협업필터링을 이용한 추천시스템의 성능 향상

Performance Improvement of a Recommendation System using Stepwise Collaborative Filtering

이재식^a, 박석두^b

^a 아주대학교 경영대학 e-비즈니스학부
경기도 수원시 영통구 원천동 산 5번지
Tel: +82-31-219-2719, E-mail:leejsk@ajou.ac.kr

^b 유비쿼터스 컨버전스 연구소
경기도 수원시 영통구 원천동 산 5번지
Tel: +82-31-219-2308, E-mail:epitus@hanmail.net

Abstract

Recommendation system is one way of implementing personalized service. The collaborative filtering is one of the major techniques that have been employed for recommendation systems. It has proven its effectiveness in the recommendation systems for such domain as motion picture or music. However, it has some limitations, i.e., sparsity and scalability. In this research, as one way of overcoming such limitations, we proposed the stepwise collaborative filtering method. To show the practicality of our proposed method, we designed and implemented a movie recommendation system which we shall call Step_CF, and its performance was evaluated using MovieLens data. The performance of Step_CF was better than that of Basic_CF that was implemented using the original collaborative filtering method.

Keywords:

Recommendation System, Collaborative Filtering

1. 서론

최근 인터넷의 사용이 보편화되면서 서비스 제공자들이 고객 정보를 실시간으로 모니터링하는 것이 가능해지면서 다양한 개인화 서비스가 제공되고 있다. 개인화 서비스란 고객들이 필요로 하는 제품이나 서비스를 명시적으로 묻지 않고 제공하는 것을 뜻한다 [Mulvenna et al., 2000]. 개인화 서비스 중에서 추천시스템 (Recommendation System)은 목표고객에게 그가 좋아할 만한 서비스나 아이템을 추천해주는 서비스로서 Amazon이나 CD Now 등 인터넷 쇼핑몰에서 많이 사용되고 있다. 추천시스템은 다양한 기법을 통해 구현 될 수 있는데

최근 전자상거래 분야에서 쓰이는 기법 중에서 대표적인 것이 협업 필터링 (Collaborative Filtering)이다.

협업 필터링은 고객들의 프로파일정보를 활용하여 목표고객이 높게 평가할 것으로 예상되는 서비스나 아이템을 추천하는 기법으로 다음과 같은 과정으로 아이템을 추천한다. 먼저, 아이템들에 대한 고객의 평가치를 직간접적으로 수집하여 고객별 프로파일을 생성한다. 생성된 프로파일을 기반으로 목표고객과 유사한 성향을 보이는 고객들로 최근접이웃을 구성한 후 최근접이웃의 평가치를 이용하여 목표고객이 평가하지 않은 아이템의 평가치를 예측한다. 이렇게 예측된 평가치를 기반으로 목표고객이 높게 평가할 것이라고 예상되는 서비스나 아이템을 추천한다. 기존의 추천시스템의 기법들은 아이템간의 연관성을 파악할 때, 아이템의 속성을 사용하였다. 하지만, 음악이나 동영상 같은 무형의 아이템들은 제목, 제작자 등의 명시적인 속성만으로 그 특성을 기술하는데 한계가 있다. 협업 필터링은 이러한 아이템들을 대상으로도 적용할 수 있다는 장점이 있다.

하지만 고객의 프로파일 정보를 기반으로 추천을 하기 때문에 유사한 성향을 보이는 최근접이웃을 찾고, 이 아이템에 대한 평가치를 예측하기 위해서는 고객 및 구매 데이터가 충분히 축적되어야 한다. 또한 희박성(Sparsity), 확장성(Scalability) 그리고 투명성(Transparency) 등의 한계점이 지적되었다 [Sarwar, 2001; Li and Yamada, 2004; Yang et al., 2004]. 그리고 협업 필터링은 고객들이 경험하고 평가한 아이템을 대상으로 추천을 하기 때문에 신제품이 추천되기가 어렵다는 지적이 제시되었다 [Canny,

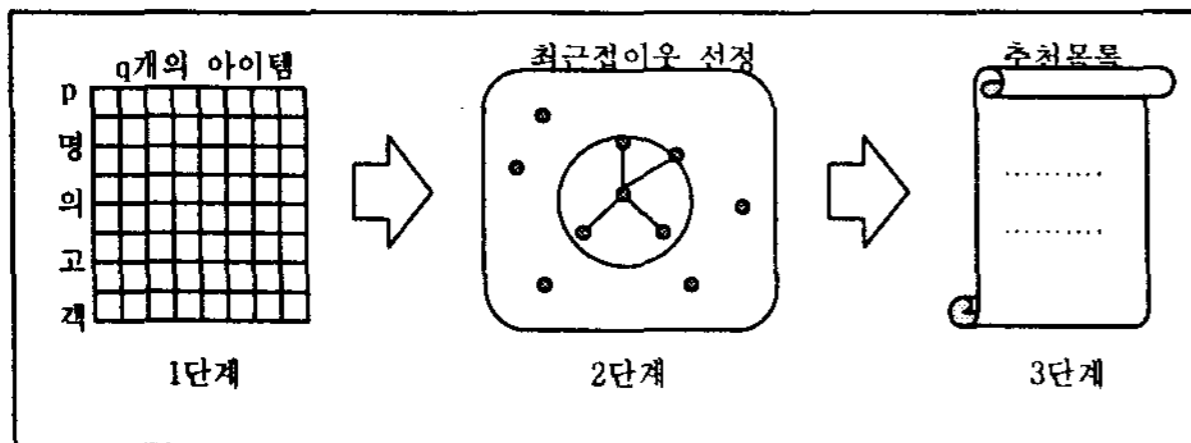
2002].

본 연구에서는 영화를 대상으로 추천시스템을 개발하였는데, 최종적으로 개별 영화를 추천하기 전에 영화 장르의 점수를 계산하여 이것을 활용함으로써 추천의 성능을 높였다. 본 연구는 다음과 같이 구성되었다. 제 2절에서는 협업 필터링에 대해서 기술한다. 제 3절에서는 본 연구에서 사용한 데이터인 영화 데이터의 설명 및 정제 과정을 설명하고, 전통적인 협업 필터링을 사용한 추천시스템에 대해서 기술한다. 제 4절에서는 본 연구에서 제안하는 단계적 협업 필터링을 사용한 추천시스템의 구축 및 성능에 대해서 기술하고, 제 5절에서 결론을 맺는다.

2. 협업 필터링

2.1. 협업 필터링의 과정

협업 필터링은 아이템에 대한 목표고객의 평가치와 다른 고객의 평가치를 이용하여 목표고객이 좋아할 만한 아이템을 추천하는 기법이다[Resnick et al., 1994; Lekakos and Giaglis, 2006]. 협업 필터링 기법으로 아이템을 추천하는 과정은 <그림 1>과 같이 크게 세 단계로 나뉘어진다.



<그림 1> 협업 필터링의 3단계

제 1단계: 평가치 매트릭스 준비

p명의 고객들이 q개의 아이템을 경험하고 부여한 평가치들을 정리한다. 고객이 각 아이템들을 경험하고 부여한 평가치를 정리하여 <표 1>과 같이 p×q의 고객×아이템 매트릭스를 만든다.

<표 1> 협업 필터링에서 고객×아이템 매트릭스

고객 \ 아이템	아이템 1	아이템 2	아이템 3	아이템 4
고객 A	$R_{A,1}$	\emptyset	$R_{A,3}$	$R_{A,4}$
고객 B	$R_{B,1}$	$R_{B,2}$	\emptyset	$R_{B,4}$
고객 C	$R_{C,1}$	\emptyset	$R_{C,3}$	\emptyset

여기서 $R_{A,1}$ 는 고객 A가 아이템 1에 부여한 평가치를 뜻한다. \emptyset 은 고객이 아이템을 평가하지 않았음을 뜻한다.

제 2단계: 최근접 이웃의 구성

고객들의 평가치를 이용하여 고객들간의 유사도를 계산한 후 최근접이웃을 구성한다. 유사도를 계산하기 위한 측정지수로는 일반적으로 Pearson Correlation Coefficient와 Cosine이 사용된다.

Pearson Correlation Coefficient $w(A, B)$ 는 두 고객 A, B에 의해 공통적으로 평가된 아이템들의 평가치를 이용하여 식(1)과 같이 계산한다[Resnick et al., 1994].

$$w(A, B) = \frac{\sum_{i=1}^q (R_{A,i} - \bar{R}_A)(R_{B,i} - \bar{R}_B)}{\sqrt{\sum_{i=1}^q (R_{A,i} - \bar{R}_A)^2 \sum_{i=1}^q (R_{B,i} - \bar{R}_B)^2}} \quad \text{식(1)}$$

여기서 $R_{A,i}$ 와 $R_{B,i}$ 는 고객 A와 B가 공통으로 평가한 아이템 i의 평가치를 뜻한다. 그리고 \bar{R}_A , \bar{R}_B 는 고객 A와 B의 이용 가능한 평가치들의 평균값을 뜻한다.

Cosine $\cos(\vec{A}, \vec{B})$ 은 고객 A와 B가 공통으로 평가한 아이템들의 평가치를 q차원 공간에 벡터화한 후 두 벡터 사이 각의 Cosine값을 식(2)와 같이 계산한다.

$$\cos(\vec{A}, \vec{B}) = \frac{\vec{A} \cdot \vec{B}}{\|\vec{A}\| \times \|\vec{B}\|} \quad \text{식(2)}$$

Herlocker et al.[1999]는 고객들간의 유사도를 계산할 때, Pearson Correlation Coefficient를 사용하는 것이 Cosine을 사용하는 것보다 높은 추천 성과를 보인다고 하였다. 그래서 본 연구에서는 최근접이웃을 구성하기 위해 Pearson Correlation Coefficient를 사용하였다.

목표고객에 대한 다른 고객들의 유사도를 계산한 후 목표고객의 최근접이웃을 다음의 두 가지 방법으로 구성할 수 있다. 첫 번째는 Threshold-based Selection [Shardanand and Maes, 1995]으로 목표고객과의 유사도가 미리 설정한 Threshold값 이상인 고객들로 최근접이웃을 구성하는 것이고, 두 번째는 k-NN 기법으로 목표고객과의 유사도가 높은 상위 k명의 고객들로 최근접이웃을 구성하는 것이다 [Resnick et al., 1994].

제 3단계: 추천목록의 생성

최근접이웃의 평가치를 이용하여 목표고객이 평가하지 않은 아이템의 평가치를 예측한 후 추천목록을 생성한다. 평가치 $R_{A,i}$ 는 고객 A의 아이템 i에 대한 평가치로 고객 A의 최근접이웃의 평가치들을 가중 평균하여 식(3)과 같이 예측한다.

$$R_{A,i} = \bar{R}_A + \frac{\sum_{j=1}^k w(A,j)(R_{j,i} - \bar{R}_j)}{\sum_{j=1}^k |w(A,j)|} \quad \text{식(3)}$$

여기서 \bar{R}_j 는 고객 A의 최근접이웃인 고객 j의 이용 가능한 평가치들의 평균값을 뜻한다.

목표고객이 평가하지 않은 아이템들의 평가치를 예측한 후 Top-N 기법으로 추천목록을 생성한다. 즉, 목표고객의 예측된 평가치 중에서 수치가 높은 상위 N개의 아이템을 목표고객에 대한 추천목록으로 생성한다.

2.2. 협업 필터링의 성능 측정

추천시스템의 성능을 측정하는 방법은 여러 가지가 있다[Sarwar et al., 2000]. 최근접이웃의 질과 같이 중간단계를 평가하는 것도 가능하나 본 연구에서는 추천시스템의 결과에 관심이 있기 때문에 예측 평가 지표와 추천 평가 지표만을 고려하였다.

2.2.1 예측 평가 지표

고객이 실제로 부여한 평가치와 추천 알고리즘에 의해 예측된 평가치의 차이로 추천시스템의 성능을 측정한다. 측정지표로 MAE(Mean Absolute Error)가 사용된다. MAE는 실제 고객의 평가치와 추천 알고리즘에 의해 예측된 평가치의 차이를 나타내는 지표로 식(4)와 같이 계산된다.

$$MAE = \frac{\sum_{i=1}^q |\text{실제 고객 평가치 } i - \text{예측된 평가치 } i|}{q} \quad \text{식(4)}$$

2.2.2 추천 평가 지표

Top-N 추천을 평가하기 위해 정보검색분야에서 사용되는 두 가지 측정지표인 Recall과 Precision이 사용된다. T 집합을 ‘고객이 실제로 경험하여 평가한 아이템들 중에서 추천시스템의 성능측정을 위해 따로 모아둔 아이템들의 집합’이라고 하고, R 집합을 ‘추천시스템에 의해 생성된 추천목록의 집합’이라고 하자. 한 아이템은 T 집합 또는 R 집합 또는 두 집합 모두에 속할 수 있다.

Recall은 목표고객이 실제로 경험하여 평가한 아이템들 중에서 추천시스템이 생성한 추천목록에 속하게 된 아이템의 비율로 식(5)와 같이 계산된다.

$$Recall = \frac{|T \cap R|}{|R|} \quad \text{식(5)}$$

Precision은 추천시스템이 생성한 추천목록의 아이템들 중에서 목표고객이 실제로 경험하여 평가한 아이템의 비율로 식(6)과 같이 계산된다.

$$Precision = \frac{|T \cap R|}{|R|} \quad \text{식(6)}$$

일반적으로 Recall과 Precision은 추천목록의 개수에 따라 상충관계에 있다. 예를 들어, 추천목록의 개수를 증가시키면 Recall은 분자가 증가하여 커지고 Precision은 분모가 증가하여 감소하게 된다. 이러한 상충관계 때문에 두 성능지표를 동시에 고려하는 F1이 식(7)과 같이 계산되어 사용된다.

$$F1 = \frac{2 \times Recall \times Precision}{Recall + Precision} \quad \text{식(7)}$$

2.3. 협업 필터링의 과제들

전통적인 협업 필터링은 희박성(Sparsity), 확장성(Scalability) 그리고 투명성(Transparency) 등의 한계점을 가지고 있다 [Sarwar, 2001; Li and Yamada, 2004; Yang et al., 2004].

희박성(Sparsity)은 고객이 아이템에 부여한 평가치의 양이 부족한 경우 추천의 성과가 떨어지는 문제점이다. 취급하는 아이템의 종류가 많은 Amazon이나 e-Bay 같은 대형 인터넷 쇼핑몰은 고객들이 실제로 경험하고 평가하는 아이템들이 전체의 1%도 안될 뿐만 아니라 평가를 하지 않는 고객들도 많다. 즉, 협업 필터링 과정 중 제 1단계에서 준비되는 고객×아이템 매트릭스의 대부분이 비어 있게 된다.

확장성(Scalability)은 고객과 거래데이터가 늘어남에 따라 목표고객의 최근접이웃을 찾기 위한 연산이 기하급수적으로 늘어난다는 문제점이다. 예를 들어, 협업 필터링이 만 명의 목표고객들에게 추천을 하고자 한다면, 만 명의 목표고객들의 최근접이웃을 찾기 위해 수십만 명의 고객데이터를 검색해야 한다. 추천시스템이 상대하는 고객의 수나 취급하는 아이템의 수가 늘어나면 최근접이웃을 구성하기 위한 계산의 양이 기하급수적으로 늘어나서 추천목록을 생성하기까지 오랜 시간이 걸려서 시스템의 효율성이 떨어진다. 확장성의 문제를 완화하기 위해 Li and Yamada[2004]는 고객들간의 유사도를 계산하는 대신 고객을 그룹화한 후 그룹들간의 유사도를 계산하여 추천목록을 생성하는 시스템을 제안하였다.

투명성(Transparency)은 추천 정보와 고객 선호도와와의 관계가 불명확하다는 문제점이다[Li and Yamada, 2004]. 고객은 아이템들이 어떻게 추천되었는지를 이해하는 것이 어려울 것이다. 투명성 문제를 완화하기 위해 김재경 등[2006]은 추천과정에서 발생한 고객 프로파일정보, 최근접이웃고객정보, 고객평가치정보, 추천목록정보 등과 웹로그 데이터, 아이템데이터, 시스템이 생성한 추천목록데이터를 이용하여 아이템조희비율,

장바구니에 담은 비율, 구매비율, 유명 매체 추천 등 20가지 유형의 설명기능을 추가한 추천시스템을 구현하였다.

3. 데이터 정제와 전통적 협업 필터링 모델

3.1. 원본 데이터

본 연구의 대상 영역은 영화로서, 미네소타 대학의 GroupLens Research Project에서 수집된 MovieLens 데이터를 사용하였다 [Resnick et al., 1994]. MovieLens 데이터는 1997년 9월에서 1998년 4월까지의 7개월의 조사기간 동안 수집된 1682편의 영화에 대한 943명의 고객들의 평가치로 구성되어있다. 조사대상인 1682편의 영화의 배포연도는 1922년부터 1998년까지이고 배포연도의 정보가 없는 1편을 제외한 각 연대별 분포는 <표 2> 와 같다.

<표 2> 조사대상 1682편의 연대별 편수

연대	1920	1930	1940	1950	1960	1970	1980	1990
편수	2	29	45	57	46	55	110	1337

MovieLens 데이터는 User ID와 Movie ID, 각 고객들이 관람한 영화에 대해 1점에서 5점까지의 5점 척도로 내린 평가치, 평가한 시점의 시간으로 구성되어있으며 총 레코드 수는 10000건이다. 각 영화는 미국의 최대 영화 데이터베이스인 IMDB(Internet Movie Database, <http://us.imdb.com>)의 기준에 따라 18개의 장르에 속해있다. 장르에 대한 정보가 없는 2편을 제외한 장르별 영화 수는 <표 3>과 같다. 한 영화는 최소 1개에서 최대 5개의 장르에 속해있다.

<표 3> 장르별 영화 수

장르	편수	장르	편수	장르	편수
Action	251	Documentary	50	Mystery	61
Adventure	135	Drama	725	Romance	247
Animation	42	Fantasy	22	Sci-Fi	101
Children	122	Film-Noir	24	Thriller	251
Comedy	505	Horror	92	War	71
Crime	109	Musical	56	Western	27

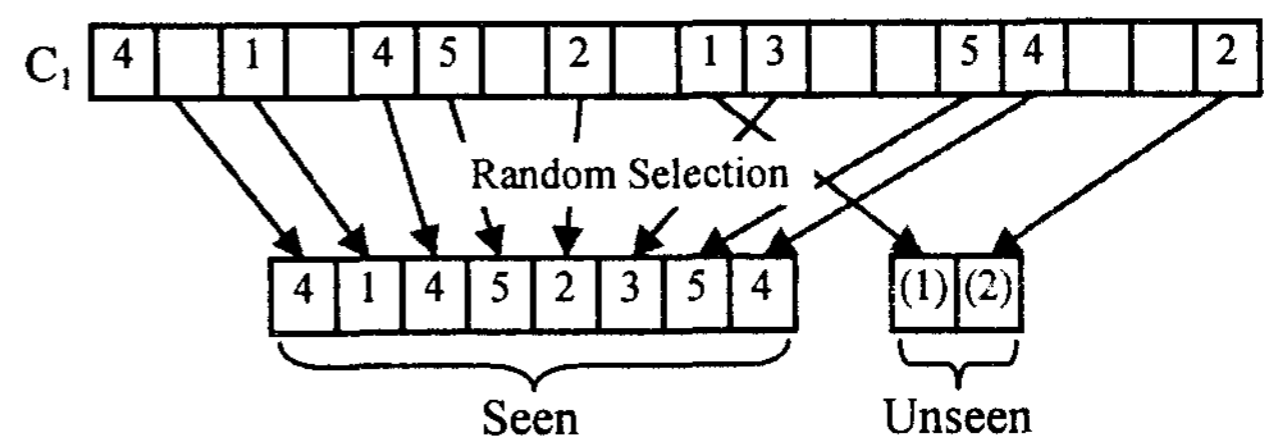
3.2. 원본 데이터에 대한 전통적 협업 필터링의 적용

본 연구에서는 먼저 원본 데이터에 대해서 전통적인 협업 필터링을 적용한 영화추천시스템을 구현하였다.

제 1 단계: 고객×영화의 매트릭스 준비

1692편의 영화들에 대해 943명의 고객들이 부여한

평가치 매트릭스를 준비하였다. 943명의 고객들로부터 무작위로 80%인 755명의 고객을 추출하여 학습데이터집합(Training Data Set)으로 사용하고 나머지 20%인 188명의 고객을 평가데이터집합(Test Data Set)으로 사용하였다. 평가데이터집합에 속하는 각 고객의 평가치들을 <그림 2>와 같이 8:2로 나누었다. 모든 평가치는 이미 주어져 있는 상태이다. 구현하는 추천시스템의 성능측정을 위해서 일부 아이템의 평가치를 모른다고 가정하고, 그 아이템의 평가치를 예측한 후에 그것을 원래 주어졌던 평가치와 비교해야 한다. 즉, 각 고객의 1682편의 영화에 대한 평가치들을 무작위로 80%와 20%로 나눈 후에 80%는 알려진 평가치로 사용하고 20%는 모른다고 가정하고 추천시스템이 예측해야 하는 평가치의 대상으로 사용하였다. 80%부분을 'Seen'으로, 20%부분을 'Unseen'으로 명명한다.



<그림 2> 추천시스템의 성능평가를 위한 각 고객들의 평가치의 분할

제 2 단계: 목표고객의 최근접이웃 구성

평가데이터집합의 고객을 목표고객으로 하고 이 목표고객들의 최근접이웃을 구성하였다. 목표고객의 Seen부분의 평가치를 이용하여 학습데이터집합의 고객들과의 유사도를 계산한 후 목표고객별 최근접이웃을 40명으로 구성하였다.

제 3 단계: 추천목록의 생성

목표고객의 Seen에 속하지 않는 아이템들의 평가치를 예측한 후 추천목록을 생성하였다. 생성된 추천목록과 Unseen의 데이터를 비교하여 추천시스템의 성능을 측정한 결과는 <표 4>와 같다.

<표 4> 원본 데이터에 대한 전통적인 협업 필터링의 적용 결과

	Recall	Precision	F1
측정값	0.062	0.096	0.075

Sarwar et al.[2000]가 동일한 데이터를 사용하여 구축한 협업 필터링 모델들의 F1이 0.120에서 0.240사이의 값을 보였지만, 본 연구에서 개발한 협업 필터링의 F1은 0.075로 매우 낮은 성능을 보였다. 그래서 원본 데이터에 대해서 데이터 정제 작업을 수행하였다.

3.3. 데이터 정제

데이터 정제 과정은 다음과 같다.

- ① 영화를 감상한 시점과 평가한 시점의 차이가 크면 왜곡된 평가를 할 수 있기 때문에 이를 완화하기 위해 1995년에서 1998년 사이에 배포된 영화들만을 고려대상으로 하였다.
- ② MovieLens 데이터에서 고객의 최소 영화 평가 수를 최소 20편으로 한 것과 같이 본 연구에서는 ①에서 선별된 영화들 중에서 최소 5%의 영화들에 평가치를 부여한 고객들만을 고려대상으로 하였다.
- ③ ②와 유사한 기준을 영화에도 적용하여, 최소 5%의 고객들로부터 평가치를 받은 영화들만을 고려대상으로 하였다.
- ④ ③의 고려대상의 장르별 영화 수가 20개 미만인 장르를 Etc로 분류하여 <표 5>와 같이 정리하였다. 18개의 장르가 11개의 장르로 축소되었고 한 영화가 속한 장르의 수는 <표 6>과 같다.

<표 5> 장르별 영화 수(정제 후)

장르 ID	장르	편수	장르 ID	장르	편수
2	Action	91	8	Mystery	23
3	Adventure	47	9	Romance	76
4	Children	27	10	Sci-Fi	35
5	Comedy	141	11	Thriller	102
6	Crime	35	12	Etc	8
7	Drama	166			

<표 6> 정제 후에 한 영화가 속한 장르의 수

	1	2	3	4	합계
영화 수	176	172	57	15	420

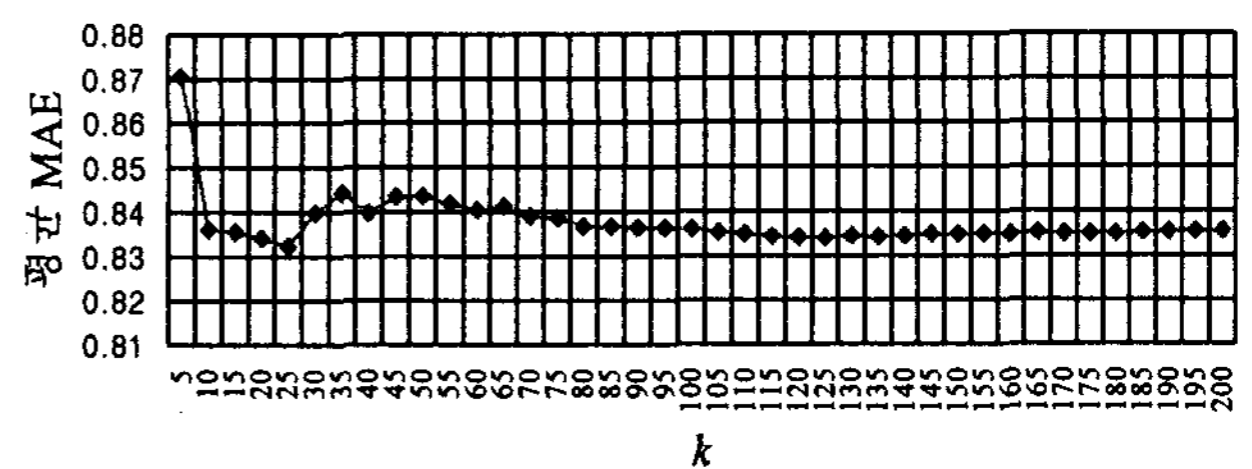
데이터 정제 작업의 결과, 고객은 총 388명, 영화는 총 420편, 장르는 총 11개가 되었다. 추천시스템을 개발하고 성능을 측정하기 위해 정제된 데이터를 Training Data Set(308명), Validation Data Set(40명), Test Data Set(40명)의 비율이 80%, 10%, 10%가 되도록 나누었다. 또한 샘플링에 따른 Bias를 없애기 위해 10-fold Cross Validation을 수행하였다. <그림 2>에서 설명한 바와 같이 Validation Data Set과 Test Data Set의 고객들의 평가치를 80%와 20%의 비율로 Seen부분과 Unseen부분으로 나누었다.

3.4. 최근접이웃의 수 설정

최근접이웃(k -NN: k -Nearest Neighbors)의 수인 k 를 설정하기 위하여 아래의 과정에 따라 k 를 증가시켜가면서 MAE의 변화를 관찰하였다. Validation Data Set의 고객들을 목표고객으로 설정하고 Training Data Set의 고객들 중에서 최근접이웃을 찾았다.

- ① n 을 1로 설정한다.
- ② k 를 $5 \times n$ 으로 설정한다
- ③ 목표고객의 Seen부분의 평가치들을 이용하여 Training Data Set의 고객들과의 유사도를 계산한다.
- ④ 유사도가 높은 상위 k 명으로 목표고객의 최근접이웃을 구성한다.
- ⑤ 구성된 최근접이웃의 평가치를 기반으로 목표고객의 Seen부분에 속하는 않는 아이템에 대한 평가치를 예측한다.
- ⑥ 목표고객의 Unseen부분의 실제 평가치와 ⑤에서 예측된 평가치로 MAE(Mean Absolute Error)를 계산한다.
- ⑦ Validation Data Set에 다른 목표고객이 남아있으면 ③으로 돌아간다.
- ⑧ Validation Data Set의 모든 목표고객들의 MAE가 구해졌으면 k 값에 대한 MAE의 평균을 계산한다.
- ⑨ k 값이 200보다 작으면 n 을 1증가시켜 ②로 돌아간다.

본 연구에서는 k 를 5에서부터 200까지 5단위로 증가시키면서 최근접이웃의 수의 변화에 따른 평균 MAE의 변화를 관찰하여 <그림 3>과 같이 정리하였다.

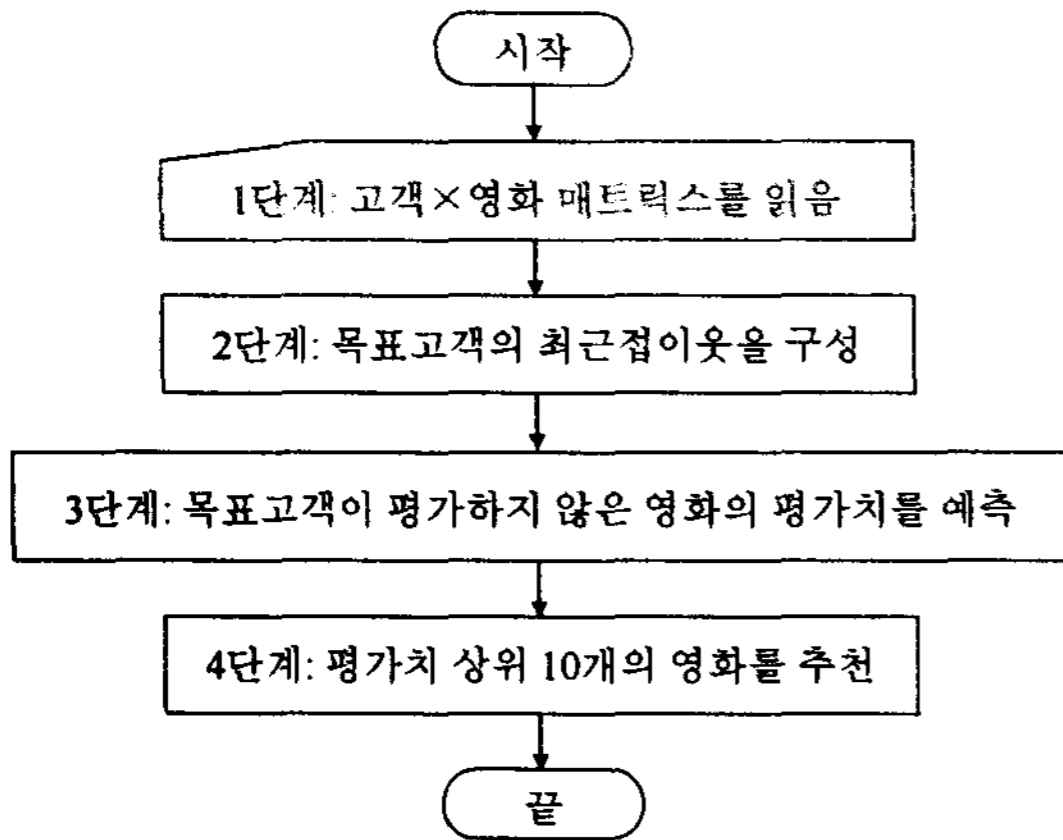


<그림 3> k 의 변화에 따른 평균 MAE의 변화

<그림 3>에서 보이는 바와 같이 최근접이웃의 수가 25명일 때, 평균 MAE가 제일 작았다. 따라서 본 연구에서는 최적의 최근접이웃의 수를 25명으로 설정하였다.

3.5. 전통적 협업 필터링 모델

전통적인 협업 필터링을 적용하여 구현한 추천시스템을 Basic_CF로 명명하였다. Basic_CF는 <그림 4>와 같이 진행된다.



<그림 4> Basic_CF의 순서도

<표 7>은 Test Data Set에 대한 추천 결과의 일부를 보여주고 있다.

<표 7> Test Data Set 고객별 추천 목록 (일부)

User ID	추천목록의 Movie ID									
	1	2	3	4	5	6	7	8	9	10
677	12	22	100	28	302	282	271	313	298	121
910	302	258	8	475	7	11	693	109	408	28
236	258	313	1	22	12	28	4	408	124	475
893	100	22	313	275	12	8	285	302	15	269
939	12	7	313	268	22	1	100	11	28	302
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
234	100	286	275	258	272	750	302	408	248	150
378	100	7	12	9	258	286	475	750	315	246
851	100	117	1	302	28	222	692	15	237	7
405	100	258	7	117	222	25	121	1	289	282
634	316	22	12	300	28	346	268	310	298	887

<표 7>에서 음영으로 처리된 부분은 추천목록과 Test Data Set의 Unseen 부분 양쪽에 속해 있는 Movie ID를 나타낸다. 즉, 추천이 적중한 Movie ID들이다. Test Data Set에 대한 Basic_CF의 10 Fold 성능 평균은 <표 8>과 같다.

<표 8> Basic_CF의 성능 (10 Fold 평균)

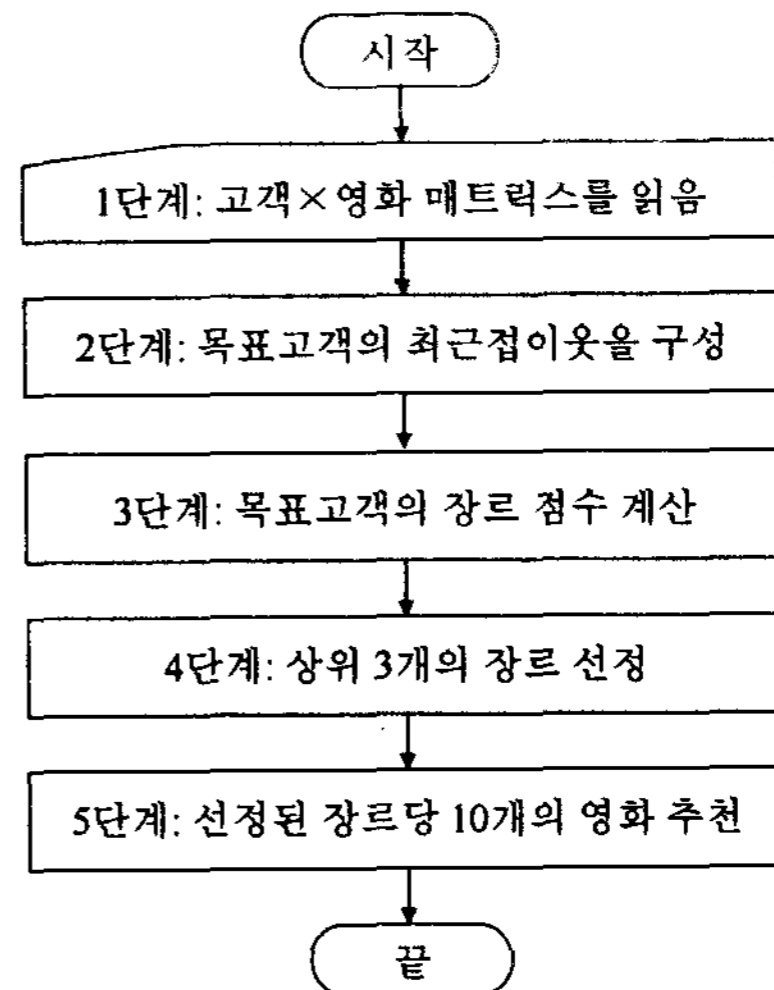
	Recall	Precision	F1
측정값	0.096	0.164	0.116

4. 단계적 협업 필터링시스템

4.1. 시스템의 설계

본 연구에서 제안하는 단계적 협업 필터링시스템 Step_CF(Stepwise Collaborative Filtering)는 최종적으로 영화를 추천하기 전에, 중간 단계에서 영화를 분류하는 기준인 장르의 점수를 계산하여 영화추천에 이용한다. 장르점수를 기반으로 선정된 장르의 영화를 추천하기 위해 최근접이웃을 새롭게 구성하여 영화를 추천하는 방법으로 Step_CF의

과정은 <그림 5>와 같다.



<그림 5> Step_CF의 순서도

<그림 5>의 제 3단계에서 목표고객의 장르점수는 최근접이웃의 평가치를 이용하여 계산되는데, 먼저 최근접이웃 j 의 장르 l 의 장르점수 $G_{j,l}$ 는 식(8)에 의해 계산된다.

$$G_{j,l} = \begin{cases} \frac{\sum_{i \in E_l^j} R_{i,l}}{|E_l^j|} & \text{if } |E_l^j| \neq \phi \\ 0 & \text{if } |E_l^j| = \phi \end{cases} \quad \text{식(8)}$$

여기서 E_l^j 은 최근접이웃 j 가 본 장르 l 에 속하는 영화의 집합을 뜻 한다.

장르의 개수를 L 이라 할 때, \bar{G}_j 는 최근접이웃 j 의 장르점수들을 평균한 값으로서 식(9)와 같이 계산된다.

$$\bar{G}_j = \frac{\sum_{l=1}^L G_{j,l}}{L} \quad \text{식(9)}$$

목표고객 A 의 장르 l 의 장르점수 $GS_{A,l}$ 은 식(10)과 같이 계산된다.

$$GS_{A,l} = \frac{\sum_{j=1}^k w(A,j) \frac{(G_{j,l} - \bar{G}_j)}{\sigma}}{\sum_{j=1}^k |w(A,j)|} \quad \text{식(10)}$$

여기서 k 는 최근접이웃의 수이고, σ 는 $G_{j,l}$ 의 표준편차를 뜻 한다.

장르 선정의 정확성은 Test Data Set 고객의 Unseen 부분의 아이템 중에서 평가치가 제일 높은 영화의 장르가 선정되었는가로 측정하였다. 선정된 장르들과 적중한 장르의 일부는 <표 9>와 같다.

<표 9> 최근접이웃의 평가치기반의 장르 선정 (일부)

User ID	장르점수											선정된 장르 ID		
	2	3	4	5	6	7	8	9	10	11	12			
677	0.18	0.11	-0.27	0.01	0.61	0.33	0.35	0.05	0.16	0.32	-1.85	6	8	7
910	0.06	-0.15	0.03	0.15	0.49	0.47	0.03	0.26	0.01	0.26	-1.61	6	7	9
236	0.08	0.06	0.10	0.06	0.28	0.55	-0.08	0.17	0.27	0.13	-1.61	7	6	10
893	0.27	-0.08	-0.25	0.19	0.59	0.64	-0.01	0.47	0.11	0.40	-2.34	7	6	9
939	0.15	-0.08	-0.29	0.05	0.37	0.50	0.19	0.28	0.22	0.25	-1.64	7	6	9
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
560	0.19	-0.17	0.00	0.11	0.53	0.60	0.26	0.46	0.01	0.40	-2.38	7	6	9
311	0.23	-0.04	0.04	0.16	0.19	0.40	0.27	0.42	0.11	0.35	-2.11	9	7	11
385	0.04	-0.23	0.07	0.05	0.57	0.43	0.10	0.07	0.28	0.13	-1.51	6	7	10

<표 9>의 '선정된 장르 ID' 열에서 음영으로 처리된 부분은 선정된 장르와 Test Data Set 고객의 Unseen 부분의 영화 중에서 평가치가 제일 높은 영화의 장르가 일치하는 것을 뜻한다. 이 일치 여부를 Test Data Set 고객 모두에 대하여 구한 후에,

이것들의 평균을 계산하여 장르 선정의 정확성을 측정하기 위한 지표로 사용하였다. 추천장르를 선정한 후, 각 장르에 속하는 영화를 10개 추천한 Step_CF의 결과의 일부는 <표 10>과 같다.

<표 10> Step_CF의 결과 (일부)

User ID	추천장르	추천목록의 Movie ID									
		1	2	3	4	5	6	7	8	9	10
677	6	12	100	302	273	293	11	327	332	346	628
	8	302	328	327	333	291	902	770	342	682	878
	7	22	100	28	282	313	258	293	31	692	327
910	6	302	11	24	248	273	770	346	129	547	1226
	7	258	8	475	7	693	28	22	4	285	223
	9	531	268	275	276	283	549	778	371	580	751
236	7	258	313	22	28	4	124	475	7	293	8
	6	12	293	302	11	17	24	346	248	770	844
	10	258	109	7	919	121	250	358	271	270	1240
893	7	100	22	313	275	8	285	15	28	223	475
	6	100	12	302	346	248	902	312	17	293	823
	9	313	275	283	276	268	306	751	311	580	692
939	7	7	313	268	22	100	28	272	475	223	315
	6	12	100	11	302	273	129	24	346	248	312
	9	313	268	129	246	724	286	111	328	751	276

<표 10>의 '추천목록의 Movie ID' 중에서 음영으로 처리된 부분은 Test Data Set 고객의 Unseen부분의 영화와 추천목록의 영화가 일치하는 것을 뜻한다.

4.2. 제안 모델의 성능 평가

제안한 모델인 Step_CF의 성능을 Basic_CF의 성능과 비교한 결과는 <표 11>과 같다.

영화추천의 중간단계인 장르선정의 정확성을 나타내는 '장르적중률'은 0.910으로 매우 높게 나타났다. 본 연구에서는 추천시스템의 성능지표로 F1값만을 사용하였다.

<표 11> Basic_CF와 Step_CF의 성능 비교

Fold	Basic_CF F1	Step_CF	
		장르적중률	F1
01	0.134	0.950	0.145
02	0.121	0.950	0.135
03	0.089	0.875	0.114
04	0.124	0.850	0.145
05	0.113	0.875	0.122
06	0.136	0.925	0.149
07	0.094	0.925	0.115
08	0.124	0.950	0.148
09	0.134	0.950	0.141
10	0.093	0.850	0.123
평균	0.116	0.910	0.134

Basic_CF에서는 10편의 영화를 추천하는 반면, Step_CF의 경우에는 각 장르별로 10편의 영화를 추천하기 때문에 총 30편의 영화를 추천한다. 따라서 추천의 개수에 따른 상충을 완화시키는 성능지표인 F1을 사용한 것이다. 성능을 비교해 볼 때, Step_CF는 0.134로서 Basic_CF의 0.116보다 0.017만큼 향상되었다. 이 같은 결과가 통계적으로 신뢰할 만한지를 알아보기 위해 Pairwise t-test를 수행하였고 그 결과는 <표 12>와 같다.

<표 12> 성능차이의 유의성 검증

대응쌍	대응차					t-Value	자유도	유의확률(양쪽)
	평균차	표준편차	차이의 표준오차	차이의 95% 신뢰구간				
				상한	하한			
Step_CF - Basic_CF	0.017	0.008	0.002	0.012	0.023	7.092	9	0.000

<표 12>에서 보는 바와 같이 t값이 7.092로서 유의수준 1%일 때의 t값인 2.821보다 크게 나타났다. 그러므로, 유의수준 1%에서 Step_CF의 성능이 Basic_CF의 성능보다 우수하다고 할 수 있다.

6. 결론 및 향후 연구

경쟁이 점점 치열해지는 시장경제에서 서비스 제공자들이 고객에게 맞춤 아이템이나 정보를 제공하는 것은 매출신장에 있어 중요한 요인 중의 하나이다. 이런 개인화 서비스를 가능하게 하는 방법 중의 하나가 바로 추천시스템이다. 추천시스템에서 사용되는 여러 기법들 중에서 전자상거래에서 성공으로 적용된 대표적인 기법은 협업 필터링이다. 협업 필터링은 명시적인 속성만으로 규정짓기 힘든 동영상이나 음악 같은 아이템들에도 효과적인 성능을 발휘한다는 장점이 있다.

본 연구는 이러한 협업필터링의 활용도를 넓히고자 단계적 협업 필터링을 제안하였다. 단계적 협업 필터링은 아이템을 최종적으로 추천하기 전에 아이템의 상위 카테고리에 대한 정보를 활용하는 방법이다. 본 연구에서는 영화 데이터를 사용하여 단계적 협업 필터링을 적용한 추천시스템인 Step_CF를 개발하였다. 실험 결과, 본 연구에서 제안한 Step_CF는 전통적인 협업 필터링을 적용하여 개발한 추천시스템인 Basic_CF보다 향상된 성능을 보였다. 또한 단계적 협업 필터링의 중간 단계인 장르추천의 적중률은 91%로서 매우 높은 적중률을 보였다. 본 연구에서 제안한 단계적 협업 필터링과 함께 고객과의 상호작용을 추가하는 방안들을 활용한다면 고객에게 보다 다양하고 만족스런 서비스를 제공할 수 있을 것이다.

Acknowledgement

본 연구는 21세기 프론티어 연구개발 사업의 일환으로 추진되고 있는 정보통신부의 유비쿼터스컴퓨팅및네트워크원천기반기술개발사업의 지원에 의한 것임.

References

- [1] 김재경, 이희애, 안도현, 조윤희, (2006). "설명기능을 추가한 협업 필터링 기반 개인별 상품추천시스템: WebCF-Exp," *경영학연구*, Vol. 35, No. 2, pp.493-519.
- [2] Canny, J. (2002). "Collaborative Filtering with Privacy," *Proc. IEEE Symp. on Security and Privacy*, pp. 45-57
- [3] Herlocker, J., J. A. Konstan, R. Borchers and J. Riedl, (1999). "An Algorithmic Framework for Performing Collaborative Filtering," *Proc. the 22nd ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 230-237.
- [4] Lekakos, G. and G. M. Giaglis (2006). "Improving the Prediction Accuracy of Recommendation Algorithms: Approaches Anchored on Human Factors," *Interacting with Computers*, Vol. 18, pp.410-431.
- [5] Li, P. and S. Yamada, (2004). "A Movie Recommender System Based on Inductive Learning," *IEEE Con. on Cybernetics and Intelligent Systems*, pp. 318-323.
- [6] Mulvenna, M., S. S. Anand and A. G. Büchner. (2000). "Personalization on the Net Using Web Mining: Introduction," *Communications of the ACM*, Vol. 43, No. 8, pp. 122-125.
- [7] Resnick, P., N. Iacovou, M. Suchak, P. Bergstrom and J. Riedl, (1994). "GroupLens: An Open Architecture for Collaborative Filtering of Netnews," *Proc. the ACM Conf. on Computer Supported Cooperative Work*, pp. 175-186.
- [8] Sarwar, B. (2001). *Sparsity, Scalability, and Distribution in Recommender Systems*, Ph.D. Diss., Dept. of Computer and Information Sciences, Minnesota Univ.
- [9] Sarwar, B., G. Karypis, J. A. Konstan and J. Riedl, (2000). "Application of Dimensionality Reduction in Recommender System: a Case Study," *Proc. the ACM WebKDD-2000 Workshop*.
- [10] Shardanand, U. and P. Maes, (1995). "Social Information Filtering: Algorithms for Automating 'Word of Mouth'," *Proc. the ACM CHI'95 Conf. on Human Factors in Computing Systems*, pp. 210-217.
- [11] Yang, W., Z. Weng and M. You, (2004). "An Improved Collaborative Filtering Method for Recommendations' Generation," *IEEE Int'l Conf. on Systems, Man and Cybernetics*, pp. 4135-4139.