

## 추천 시스템 기법 연구동향 분석

손지은<sup>1</sup> · 김성범<sup>1\*</sup> · 김현중<sup>2</sup> · 조성준<sup>2</sup>

<sup>1</sup>고려대학교 산업경영공학과 <sup>2</sup>서울대학교 산업공학과

## Review and Analysis of Recommender Systems

Jieun Son<sup>1</sup> · Seoung Bum Kim<sup>1</sup> · Hyunjoong Kim<sup>2</sup> · Sungzoon Cho<sup>2</sup>

<sup>1</sup>Department of Industrial Management Engineering, Korea University

<sup>2</sup>Department of Industrial Engineering, Seoul National University

The explosive growth of the world-wide-web and the emergence of e-commerce has led to the development of recommender systems. Recommender systems are personalized information filtering used to identify a set of items that will be of interest to a certain user. This paper reviews recommender systems and presents their pros and cons.

**Keywords:** Recommender system, Content-based approach, Collaborative filtering, Data mining

### 1. 서론

정보 통신의 발달과 더불어 고객이 상품을 구매하는 패턴 또한 빠르게 변화하고 있다. 과거에는 대부분 매장에서 직접 물건을 구매하였고 이때, 판매자와의 직접적인 소통을 통해 원하는 물건을 보고 체험한 뒤 선택하였으나 최근에는 TV나 컴퓨터를 비롯한 각종 스마트 기기의 발달로 홈쇼핑이나 인터넷 쇼핑이 생활화 되어 언제 어디서나 간편하게 물건을 구매할 수 있게 되었다. 그러나 고객의 입장에서 다양한 아이템 종류와 사전지식의 부족으로 인해 자신이 원하는 물건을 구매하는데 있어 어려움이 존재하며, 판매자 입장에서 개인의 선호도를 고려해 적절한 아이템을 추천해주고 이를 구매로 연결시키는 것은 이윤창출과 직결되므로 적절한 아이템 추천의 중요성이 대두되고 있다(Das et al., 2013). 가장 간단한 추천 방식으로는 베스트셀러기반 추천방법, 최소질의대상 상품결정 방법 등이 있다. 베스트셀러기반 추천방법은 판매량이 많은 순서대로 상품을 추천하는 방식으로 고객의 개인정보 없이도 신속한 추천이 가능하지만 개인의 취향을 고려하지 못하기 때문에 개인화된 추천이 불가능하다는 문제점을 가지고 있다

(Sarwar et al., 2000). 최소질의대상 상품결정 방법은 고객에게 직접 설문하여 얻은 정보를 통해 선호도를 파악하고 상품을 추천하는 방식이다. 개인화 된 추천이 가능하고 간편한 알고리즘으로 구현될 수 있으나 고객이 응답을 불완전하고 부정확하게 할 경우 추천결과에 대한 신뢰성이 저하될 수 있는 문제점을 가지고 있다(Schein et al., 2002; Yu et al., 2004). 따라서 최근에는 개인화 된 추천 시스템을 구현하기 위해 정보필터링 방법과 연관성 분석 등 다양한 추천 기법이 연구되고 있다(Schafer et al., 1999; Sarwar et al., 2000; Jin et al., 2010).

<Figure 1>은 다양한 추천 시스템의 종류를 보여주고 있다. 본 논문에서는 다양한 추천 시스템의 종류와 각각의 특징 및 장·단점에 대해 살펴보고 실제 적용사례를 알아봄으로써, 추천 시스템 기법의 발전 흐름과 동향을 파악하고자 하였다. 먼저, 논문의 제 2장, 제3장에서는 정보필터링 기법인 콘텐츠기반 접근방식과 협력필터링에 대해서 알아보았고 이들의 단점을 보완하기 위해 두 기법을 결합한 하이브리드 추천 시스템을 제 4장에서 소개하였다. 이어 제 5장에서는 연관성 분석을 바탕으로 연구된 추천 시스템에 대하여 설명하였으며 제 6장에서는 시간적 역동성을 고려한 추천 시스템, 제 7장에서는 다

\* 연락저자 : 김성범 교수, 136-701 서울시 성북구 안암동 5가 1번지 고려대학교 산업경영공학과, Tel : 02-3290-3397, Fax : 02-929-5888, E-mail : sbkim1@korea.ac.kr

2014년 7월 15일 접수; 2014년 11월 3일 수정본 접수; 2014년 11월 18일 게재 확정.

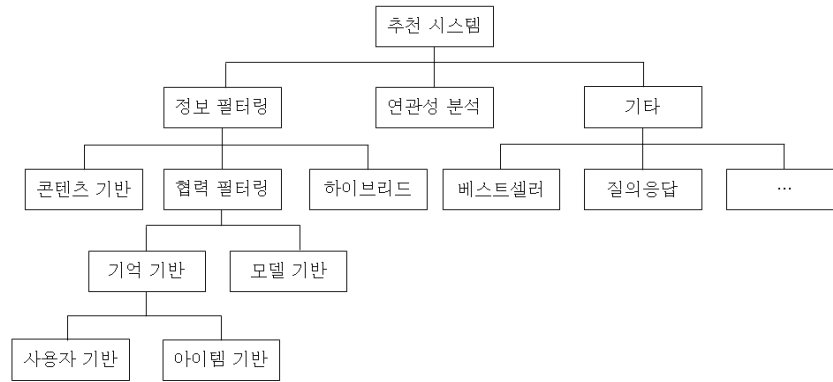


Figure 1. The categorization of recommender systems

양성 향상을 위한 추천 시스템을 설명하였다. 제 8장에서는 추천 시스템의 성능을 평가하는 기법에 대하여 다루었고, 제 9장에서는 실제로 추천 시스템을 성공적으로 적용한 사례에 대하여 살펴보았으며, 제 10장에서는 추천 시스템을 구현할 수 있는 소프트웨어를 소개하였다. 마지막으로 제 11장 결론에서는 본 논문을 요약한 후, 향후 연구방향에 대해 언급하였다.

## 2. 콘텐츠기반 접근방식

콘텐츠기반 접근방식은 정보검색기술에 바탕을 둔 시스템으로, 아이템의 콘텐츠를 직접 분석하여 아이템과 아이템 혹은 아이템과 사용자 선호도 간 유사성을 분석하여 이를 토대로 고객에게 아이템을 추천해주는 방식이다(Wu *et al.*, 2000). <Figure 2>는 콘텐츠기반 접근방식 추천 기법을 보여주고 있다. 사용자가 직접 입력한 프로파일 정보나, 사용자가 아이템에 대해 평가한 점수 혹은 과거 구매내역을 바탕으로 생성된 정보를 통해 선호하는 아이템을 파악한 다음, 미리 선정된 기준을 통해 분류된 아이템 카테고리 and 사용자의 선호 아이템 간의 유사도를 측정한다. 마지막으로, 유사도가 가장 높게 나타난 '카테고리 5'에 해당하는 아이템을 추천 대상 고객에게 추천한다(Lang, 1995). 본 방식은 분석의 용이함과 메타데이터의 가용성으로 인해 영화, 음악, 도서뿐만 아니라 텍스트 기반의 뉴스나

인터넷 기사 등을 추천하는데 널리 쓰이고 있다(Pazzani *et al.*, 2007).

### 2.1 콘텐츠기반 접근방식의 특징

#### (1) 독립적 정보 활용

협력필터링에서 추천 대상 고객과 취향이 비슷한 이웃 사용자를 찾기 위해 여러 사람의 평가점수 데이터가 필요한 반면, 콘텐츠기반 접근방식에서는 추천 대상 고객의 선호도를 파악하기 위해 자신만의 과거 구매이력이나 프로파일 정보가 이용된다(Pazzani *et al.*, 2007). 이는 추천하고자 하는 고객의 독립적인 정보만을 필요로 하므로, 다른 사용자의 정보가 부족할 경우에도 유용하게 쓰일 수 있는 추천 기법으로 알려져 있다. 그러나 자신의 과거 구매이력을 통해 선호도를 파악해야 할 경우, 선호도를 파악하기에 충분한 아이템 평가점수가 존재해야 한다. 과거 구매이력이 부족할 경우 추천의 성능을 보장할 수 없으며, 구매이력과 프로파일 정보가 모두 존재하지 않는다면 콘텐츠기반 접근방식 추천 시스템을 구현하는 것이 불가능하다는 단점이 있다(Bobadilla *et al.*, 2013).

#### (2) 새로운 아이템

새롭게 추가된 아이템에 대하여 아직 구매가 이루어지지 않아 점수가 존재하지 않을 경우, 협력필터링 방식에서는 누군

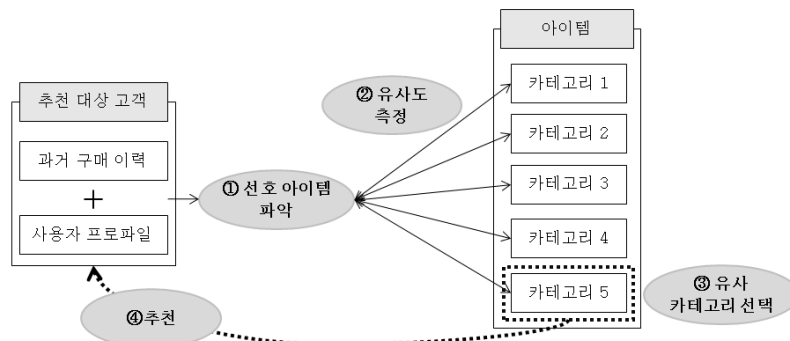


Figure 2. The algorithm of content-based approach

가가 점수를 주기 전까지는 해당 아이템은 추천 리스트에 포함될 수 없으며, 이러한 문제를 First rater라고 한다(Su *et al.*, 2009). 평가점수가 존재하지 않는 아이템이 많을 경우, 협력필터링은 과거에 출시된 아이템만 반복적으로 추천하며 추천 리스트에 포함되는 아이템 개수 또한 제한적이므로 사용자의 추천 만족도를 감소시킬 수 있다. 그러나 콘텐츠기반 접근방식에서는 아이템에 대한 평가점수가 존재하지 않더라도 아이템 간의 속성을 파악하여 카테고리에 할당된다. 최종적으로, 사용자가 선호하는 아이템 속성과 유사도가 높은 카테고리의 아이템들이 사용자에게 추천되기 때문에 First rater 문제가 발생하지 않는다(Lops *et al.*, 2011).

### (3) 과도한 특수화(Over Specialization)

콘텐츠기반 접근방식은 추천대상이 되는 고객의 과거 구매 이력을 바탕으로 유사한 제품을 찾아주기 때문에 다른 사용자들의 취향이나 선호도를 반영하지 못하고 이전에 구매한 상품과 너무 비슷한 상품들만이 추천된다는 문제점을 가지고 있다. 즉, 새로운 속성의 아이템을 접할 기회가 없으므로 추천 상품의 다양성을 보장할 수 없으며 이러한 문제를 과도한 특수화라고 한다(Balabanović *et al.*, 1997). 이를 해결하기 위해 추천 과정에서 유전자 알고리즘을 적용하여 무작위 요소를 추가하거나 돌연변이 방식을 사용하는 기법이 제안되었고, 이는 실험을 통해 기존 콘텐츠기반 접근방식보다 우수한 성능을 보임을 입증하였다(Sheth *et al.*, 1993).

## 2.2 콘텐츠기반 접근방식

### (1) 아이템 속성 분석

아이템은 속성들의 집합으로 표현될 수 있으며, 이는 크게 구조적 데이터와 비 구조적 데이터로 나눌 수 있다.

구조적 데이터는 아이템의 속성이 명확히 정의되어 있고 이에 따른 아이템의 속성 값이 구조적으로 입력되어있는 데이터로써, 아이템을 잘 표현할 수 있는 속성의 개수가 비교적 적고 아이템마다 각 속성에 해당하는 값이 모두 존재해야 한다. 예를 들어, 주연배우, 감독, 개봉일, 장르, 배급사 등에 대한 속성 값을 가지고 있는 영화 데이터베이스는 구조적 데이터의 좋은 예시이다. 이는 기존의 K-means와 같은 기계학습 알고리즘을 적용하여 비슷한 속성의 아이템으로 구성된 카테고리를 생성할 수 있다(Pazzani *et al.*, 2007).

반면, 비 구조적 데이터는 이미지, 소리, 텍스트와 같이 구조화된 데이터 관리 시스템에 포함할 수 없는 데이터이다. 이러한 비 구조적 데이터는 최근 들어 급증하고 있지만 구조적 데이터에 비해 속성 정의가 어렵다는 단점이 있다. 이러한 어려움을 극복하고자 다양한 연구가 진행되고 있으며 특히, 뉴스, 책, 웹 페이지와 같이 텍스트로 이루어진 콘텐츠에 대한 연구가 활발히 이루어지고 있다. 텍스트 데이터에서 콘텐츠기반 접근 방식은 크게 키워드 분석과 의미 분석으로 나눌 수 있다. 먼저 키워드 분석은 각 텍스트 아이템에서 키워드를 추출한 뒤, 아

이템간의 키워드를 비교하여 유사도를 계산한다. 즉, 사용자가 선호하는 아이템과 키워드 유사도가 높은 아이템을 추천 리스트에 포함시킨다. 키워드를 추출하는 대표적인 방법은 TF-IDF이며 식 (1)을 통해 계산할 수 있다(Salton *et al.*, 1988).

$$TF-IDF(t_k, d_j) = TF(t_k, d_j) \times IDF \quad (1)$$

$j$ 번째 아이템  $d$ 에서  $k$ 번째 단어  $t$ 의 TF-IDF은 TF(Term Frequency)와 IDF(Inverse Document Frequency)의 곱으로 나타난다. TF는 한 아이템 내에서 특정 단어가 출현한 빈도수이며, 이는 문서 내에서 많이 출현하는 키워드는 상대적으로 더 중요하다라는 가정을 바탕으로 하고 있다. TF에서는 정규화를 위해 아이템 내부에서의 단어 출현 빈도수를 모든 단어의 총 출현 빈도수로 나누어 주며 식 (2)와 같다

$$TF(t_k, d_j) = \frac{f_{k,j}}{\max_z f_{z,j}} \quad (2)$$

그러나, 문서 내에서 출현 빈도가 높더라도 불용어에 해당하는 확률이 크기 때문에, 이러한 문제를 보완하기 위한 IDF을 포함해야 한다.

$$IDF(t_k, d_j) = \log \frac{N}{n_k} \quad (3)$$

식 (3)에서,  $N$ 은 전체 아이템 개수이며  $n_k$ 는  $k$ 번째 단어  $t$ 가 출현한 아이템의 개수이다. 이는 상대적으로 적은 아이템에서 출현한 단어일수록 높은 IDF 값을 갖게 되므로 키워드로 추출될 확률이 높다는 것을 의미한다. 각 단어는 0부터 1사이의 값을 갖는 단어 가중치로 나타낼 수 있으며 식 (4)와 같다(Lops *et al.*, 2011).

$$\text{단어 가중치}(t_k, d_j) = \frac{TF-IDF(t_k, d_j)}{\sqrt{\sum_{s=1}^{|T|} TF-IDF(t_k, d_j)^2}} \quad (4)$$

위와 같은 과정을 통해 단어 가중치가 높은 상위  $N$ 개의 단어가 키워드로 선택되면, <Table 1>과 같은 아이템-키워드 행렬을 구성한다.

Table 1. Example of 'item-keyword' matrix

	키워드 1	키워드 2	키워드 3	키워드 4
아이템 1	0.1	0	0.5	0
아이템 2	0	0.9	0	0
아이템 3	0	0.8	0	0.1
아이템 4	0	0.1	0.2	0.3
아이템 5	0.1	0.1	0	0.7

아이템-키워드 행렬을 통해 아이템간의 유사도를 계산하기 위해 대표적으로 사용되는 유사도 측정치로는 코사인 유사도(cosine similarity)가 있으며 식 (5)를 통해 계산되어진다(Tata *et al.*, 2007).

$$\text{코사인 유사도}(d_i, d_j) = \frac{\sum_k w_{ki} \times w_{kj}}{\sqrt{\sum_k w_{ki}^2} \times \sqrt{\sum_k w_{kj}^2}} \quad (5)$$

이때,  $w_{ki}$ 는  $i$ 번째 아이템에서  $k$ 번째 키워드에 대한 단어 가중치이며  $w_{kj}$ 는  $j$ 번째 아이템에서  $k$ 번째 키워드에 대한 단어 가중치이다.

Table 2. Example of cosine similarity among the items

	아이템 1	아이템 2	아이템 3	아이템 4	아이템 5
아이템 1	1	0	0	0.52	0.02
아이템 2	0	1	0.99	0.26	0.14
아이템 3	0	0.99	1	0.36	0.26
아이템 4	0.52	0.26	0.36	1	0.82
아이템 5	0.02	0.14	0.26	0.82	1

<Table 2>는 <Table 1>의 아이템-키워드 행렬에서 아이템 간의 유사도를 측정하기 위해 코사인 유사도를 적용한 결과이다. 예를 들어, ‘아이템 3’을 선호하는 사용자에게는 ‘아이템 3’과 가장 유사한 ‘아이템 2’를 추천하게 된다. 그러나 키워드를 기반으로 아이템의 유사도를 평가하는 방법은 다음과 같은 문제점을 야기시킬 수 있다. 먼저 하나의 단어가 다양한 의미를 가지는 경우 실제로 유사하지 않은 아이템이 추천될 수 있으며, 반대로 다양한 단어로 표현되지만 실제로 같은 의미를 가지는 경우에는 내용적으로 유사한 아이템일지라도 유사도가 낮으므로 추천 리스트에서 누락될 수 있다(Wartena *et al.*, 2011). 이러한 문제를 해결하기 위해서 의미론 분석의 중요성이 대두되고 있으며, 단어의 단순 스트링 비교가 아닌 내용 파악을 목적으로 하므로 보다 정확도 높은 추천 결과를 도출할 수 있다. 의미론 분석에서는 품사 별 동의어 관계를 링크로 연결한 대형 네트워크 형태의 워드넷(Wordnet)이 가장 널리 이용되고 있다(Miller, 1995; Jiang *et al.*, 1997; Ahsae *et al.*, 2014).

## (2) 사용자 선호도 학습

콘텐츠 기반 접근방식에서 사용자의 선호도를 파악하기 위해 사용되는 첫 번째 방법은 사용자가 직접 흥미 분야에 대해 입력한 프로파일 정보를 이용하는 것이다. 시스템이 사용자에게 영화 장르, 스포츠 팀, 뉴스 분야 등에 대한 카테고리 영역을 보여주면, 사용자는 관심분야의 체크상자를 표시하여 프로파일 정보를 제공한다. 때로는 보다 상세한 선호도 정보를 수집하기 위해, 시스템이 제공하는 질문에 대해 사용자가 자유롭게 텍스트 형식으로 기재할 수 있는 방식을 취하기도 한다. 프로파일

정보의 활용은 자신의 과거 구매이력이나 다른 사용자의 아이템 평가점수 데이터가 존재하지 않더라도 사용자의 선호도를 쉽게 파악할 수 있는 방법이며 평가점수 데이터가 부족한 경우에 널리 쓰인다(Lü *et al.*, 2012). 그러나 정보를 정확하게 입력하는 사용자를 확보하는 것이 어렵고, 정확히 입력하였다고 해도 추후에 사용자의 선호 경향이 변화할 경우 올바른 추천 리스트 제공이 어렵다는 한계점이 있다. 또한 사용자가 선호하는 카테고리 내에서 아이템들의 중요도를 판단할 수 없기 때문에 정교한 추천 성능을 보장할 수 없다.

두 번째 방법은, 자신의 과거 구매이력 정보를 분석하는 것으로서 최근 혹은 자주 구입한 아이템을 통해 선호도를 파악하는 것이다. 이때 나이브 베이즈모델, 의사결정나무, K-근접 이웃 기법 등과 같은 기계학습을 이용하여 선호도를 정확하게 파악하여 추천의 성능을 향상시키고자 하는 다양한 연구가 제안되었다(Pazzan *et al.*, 2007; Lops *et al.*, 2011).

## 3. 협력필터링

협력필터링은 추천 시스템 중에서 현재까지 가장 우수한 성능을 나타낸다고 알려진 기법으로써, Goldberg *et al.*(1992)에 의해 그 개념이 처음으로 소개되었다. 이후 미네소타 대학의 GroupLens를 시작으로 인터넷 서점인 Amazon.com, CD 상점인 CD Now.com, 인터넷 영화 추천 사이트인 MovieFinder 등과 같은 다양한 분야에서 사용되고 있다(Resnick *et al.*, 1994; Konstan *et al.*, 1997; Schafer *et al.*, 2001). 협력필터링은 ‘특정 아이템에 대해 선호도가 유사한 고객들은 다른 아이템에 대해서도 비슷한 선호도를 보일 것’이라는 기본 가정을 바탕으로 사용자 혹은 아이템간 유사도를 기반으로 선호도를 예측하는 방법이다. 콘텐츠 기반 접근방식이 사용자와 아이템 정보에만 의존하여 선호도를 예측하는 반면 협력필터링은 사용자가 아이템에 대해 평가한 정보를 사용해 선호도를 예측한다는 것이 가장 큰 차이점이다. 즉, 추천대상이 되는 고객과 취향이 비슷한 사용자를 선정하고 그들이 선호하는 아이템을 추천대상 고객에게 추천하기 때문에 추천되는 아이템의 다양성을 보장할 수 있다.

<Figure 3>(a)는 사용자 기반 협력필터링의 기본적인 개념을 나타내고 있다. 추천 대상 고객이 선정되면, 구매 이력을 바탕으로 추천 대상 고객과 다른 사용자들 간의 유사도를 측정한다. 즉, 구매한 아이템이 일치할수록 취향이 비슷한 것이므로 높은 유사도를 갖게 되며, 모든 사용자들과 유사도를 측정했을 때 유사도가 가장 높은 사용자를 이웃으로 선택한다. 예를 들어, <Figure 3>(a)의 경우 추천 대상 고객과 취향이 가장 비슷한 사용자는 ‘아이템 1’, ‘아이템 3’, ‘아이템 5’를 구매한 ‘사용자 b’이다. 협력필터링의 마지막 단계에서는, 아이템 추천 단계로써 유사도 측정을 통해 선택된 ‘사용자 b’는 구매하였으나 추천 대상 고객은 아직 구매하지 않은 ‘아이템 8’을 최종적으로 선택하여 추천 대상 고객에게 추천한다. <Figure 3>(b)는 아

이템 기반 협력필터링의 기본 개념으로써 추천 대상 아이템을 기준으로 유사한 아이템을 선정한 뒤, 선정된 ‘아이템 2’는 구매하였으나 추천 대상 아이템은 구매하지 않은 ‘사용자 e’에게 추천 대상 아이템을 최종적으로 추천하게 된다.

이와 같은 협력필터링은 크게 기억 기반 협력필터링과 모델 기반 협력필터링으로 나뉜다(Breese *et al.*, 1998; Resnick *et al.*, 1994). 기억 기반 협력필터링은 앞서 설명한 방식으로써 사용자 간 유사도를 계산한 뒤 유사도가 높은 사용자가 선택한 아이템을 추천해주는 방식이다. 모델기반 협력필터링은 기억 기반 협력필터링 방식의 과정을 기본으로 하되 군집화, 분류, 예측의 단계에서 기계학습 또는 데이터마이닝 기법을 활용하는 것이다.

이는 기존의 기억 기반 협력필터링 적용 시 발생하는 다양한 문제점을 보완하기 위해 제안된 기법으로, 사용되는 모델은 확률 및 통계 이론에 근거한 베이지안, 선형 회귀분석, 마코프 결정 프로세스 등이다(Breese *et al.*, 1998; Hofmann, 2004). 본 제 3장에서는 먼저 협력필터링의 특징에 대해 설명하고 제 3.2절과 제 3.3절에서 각각 기억 기반 협력필터링과 모델기반 협력필터링에 대해 상세히 기술하였다.

### 3.1 협력필터링의 특징

#### (1) 데이터 희소성(Sparsity)

협력필터링에서는 추천 대상 고객의 선호도를 예측하기 위

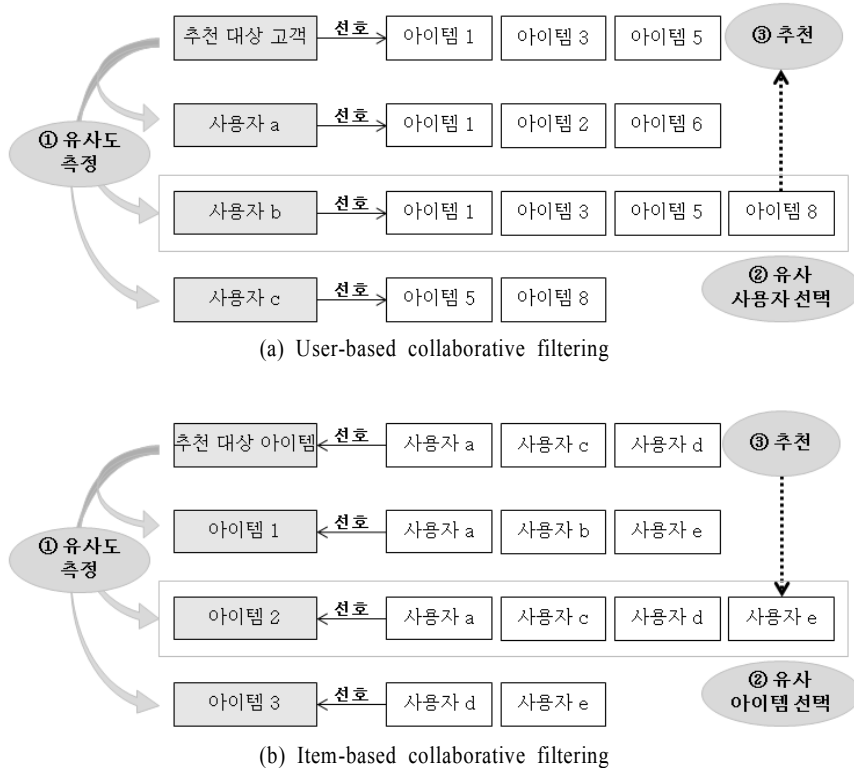
해 <Table 3>과 같이 사용자-아이템 행렬이 사용되며 행렬 값은 사용자의 아이템에 대한 만족도를 점수로 나타낸 것이다.

**Table 3.** Example of ‘user-item’ matrix

	아이템 1	아이템 2	아이템 3	아이템 4
사용자 a	4	?	5	
사용자 b	4	2	1	
사용자 c	3		2	
사용자 d	4	4		
사용자 e				

<Table 3>에서 ‘사용자 a’는 ‘아이템 1’과 ‘아이템 3’에 각각 4점과 5점의 평가 값을 주었다. 그러나 이와 같은 사용자-아이템 행렬에서의 평가 값은 상품의 인기도에 따라 데이터가 편중되거나 아직 평가 값을 입력하지 않은 새로운 사용자가 존재할 수 있다. 이런 경우, 유사도를 측정할 만한 충분한 데이터가 존재하지 않기 때문에 선호도를 예측하는 것이 불가능하며 이를 Cold start라고 한다. 또한 ‘아이템 4’와 아직 구매가 이루어지지 않아 점수가 존재하지 않을 경우, 누군가가 점수를 주기 전까지는 추천이 이루어 질수 없으며 이를 First rater 라고 한다. 이처럼 데이터의 부족으로 인해 발생하는 문제를 총칭하여 데이터 희소성이라 하며 이러한 문제점을 해결하기 위해 다양한 연구가 진행되었다(Su *et al.*, 2009).

Lee *et al.*(2010a)는 논문추천 시스템에서 평가정보나 이용정

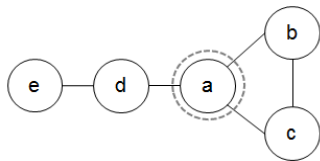


**Figure 3.** The algorithm of collaborative filtering

보가 부족한 신규논문을 사용자에게 추천하기 위해서 논문 이용 정보 대신 논문의 키워드를 활용하는 방식을 제안하였다. 신규논문이 입수되면 신규논문의 키워드와 사용자들이 기존에 이용한 논문의 키워드를 비교하여 각 사용자들에게 추천될 확률을 계산한 뒤, 가장 추천될 확률이 높은  $N$ 명의 사용자들에게 신규논문을 추천하게 된다. 이때 사용된 알고리즘은 문서 분류에 있어 대표적으로 사용되는 나이브 베이즈 모델이다. 또한 <Table 3>의 ‘사용자 e’와 같은 새로운 고객에 대해서는 사용자의 취향을 파악할 만한 구매 이력정보가 존재하지 않기 때문에 추천목록을 제공하는 것이 어렵다. 이를 해결하기 위해 Shin *et al.*(2012)은 협력필터링 기법과 사회연결망 기법의 중심성을 결합한 분석을 시도하였다. 먼저, 기존 고객들 간 유사도를 기반으로 네트워크를 생성한 뒤, 중심성이 높은 사용자들을 새로운 고객의 이웃으로 정하고, 그들이 선택한 아이템을 새로운 고객에게 추천하게 된다. 이처럼, 네트워크를 활용한 기법은 기존의 협력필터링 기법과 달리 구매성향이 서로 다른 사용자들의 정보를 모두 활용하여 추천의 정확성을 향상시켰다. 특히 중심성이 높은 고객이 선택한 상품을 신규고객에게 추천해 주는 방식은 많은 사람들과 비슷한 선호도를 갖는 고객의 취향은 대중적이고 신뢰성이 높다는 가정에서 출발한다. 우선 중심도를 갖기 위해서는 <Table 4>(a)와 같이 사용자간 유사도 행렬을 구한 뒤, 이를 네트워크로 표현하여 각 노드의 중심성을 계산하게 된다. <Table 4>에서는 중심성이 높은 ‘사용자 a’가 구매한 아이템 목록을 신규고객에게 추천하게 된다.

	사용자 a	사용자 b	사용자 c	사용자 d	사용자 e
사용자 a	1	0.7	0.8	0.8	0
사용자 b	0.7	1	0.6	0	0
사용자 c	0.4	0.6	1	0	0
사용자 d	0.8	0	0	1	0.9
사용자 e	0	0	0	0.9	1

(a) Similarity matrix



(b) User network

**Figure 4.** Example of recommendation for new customers using collaborative filtering and social network analysis

Kim *et al.*(2003)는 신규고객의 과거 구매이력은 없지만 사용자들의 프로필 정보가 있을 경우, 프로필 정보를 기반으로 사용자간의 유사도를 측정하여 선호도가 비슷한 사용자를 결정한 뒤, 그들의 선호도를 통해 신규고객의 선호도를 예측하는 알고리즘을 개발하였다. 이 연구는 과거 구매이력이 없는 신규고객에 대해서도 개인화된 추천목록을 제공할 수 있다는 장

점이 있지만 사용자의 프로필 정보가 있어야만 활용이 가능하다는 한계점이 있다.

Ko *et al.*(2014)는 사용자를 ‘주요 사용자’와 ‘일반 사용자’로 분류하는 방법을 제안하였는데 ‘주요 사용자’란 많은 아이템을 평가했거나 구매한 사용자로서 이들로부터 얻은 사용자-아이템 행렬에는 공백이 거의 없고, 이는 기존의 데이터 희소성 문제점을 해결하였다. 먼저 ‘주요 사용자’의 데이터로부터 선호하는 아이템 간 유사성을 계산하여 아이템을 군집화 하고 같은 방식으로 ‘일반 사용자’를 군집화 한 다음, ‘주요 사용자’와 ‘일반 사용자’의 유사한 군집끼리 매칭 시켜 ‘주요 사용자’ 군집에는 존재하지만 ‘일반 사용자’ 군집에는 없는 상품을 ‘일반 사용자’에게 추천한다.

이 외에도 데이터 희소성 문제점을 근본적으로 해결하기 위해 데이터의 차원을 축소하는 방법이 연구되고 있는데 대표적으로 특이값 분해(Singular Value Decomposition; SVD)가 있으며 이는 중요하지 않은 사용자나 아이템을 사용자-아이템 행렬에서 직접 제거하여 행렬의 차원을 축소시키는 방법이다(Billsus *et al.*, 1998; Sarwar *et al.*, 2000). Berry *et al.*(1995)는 이렇게 차원 축소된 데이터는 원본 데이터보다 노이즈가 적어 선호도 예측에 있어 우수한 성능을 나타냄을 확인하였다. Lee *et al.*(2014)은 순위 기반 추천 시스템의 사용자-아이템 행렬에서, 부분적으로 낮은 순위는 갖는 데이터에 대해 손실함수의 개념을 적용하여 추천의 성능을 향상시켰다.

## (2) 확장성(Scalability)

최근 상품의 종류가 다양해지고 사용자가 늘어남에 따라 사용자-아이템 데이터 셋의 크기가 커지고 있으며, 이에 따라 추천 알고리즘 구현 시 컴퓨터 계산량이 급증하게 되어 많은 시간과 비용이 소모되는 문제점이 발생하고 있다. 보통 사용자의 구매행동이 발생하는 즉시 사용자-아이템 행렬이 갱신되며, 대부분의 시스템에서는 이와 같이 실시간으로 갱신되는 정보를 통해 신속하게 새로운 예측 값을 계산하여 고객에게 추천을 진행한다. 즉, 즉각적인 피드백 구조가 유지되기 위해서 알고리즘의 빠른 계산 속도는 매우 중요한 사항이다. 그러나 모든 데이터를 사용하여 선호도를 예측하는 방식은 계산량이 많을 뿐만 아니라 예측력의 정확도를 저하시킬 수 있다(Su *et al.*, 2009).

Park *et al.*(2014)은 유사도가 큰 순서대로  $N$ 개의 사용자 혹은 아이템만을 선정하여 선호도를 예측하는 방법을 제안하였다. 가장 유사한  $K$ 개의 개체를 찾기 위해  $K$ -근접이웃 그래프를 생성하였으며 MovieLens 데이터 셋을 통해 높은 추천 만족도를 확인하였다. 특히, 전처리를 포함한 계산 속도를 비교하였을 때, 모든 사용자 혹은 아이템을 사용하여 계산했던 기존의 협력필터링 기법보다 빠른 계산 속도를 나타냄을 알 수 있었다.

## (3) Grey sheep

협력필터링 접근방식은 사람들 취향에 경향과 패턴이 존재

한다는 가정을 하고 있으나 일관성 없는 의견을 가진 사용자들의 데이터는 경향과 패턴을 파악함에 있어 방해가 되며 예측 정확도를 떨어뜨린다. Claypool *et al.*(1999)는 이러한 문제점을 해결하기 위해 콘텐츠기반 접근방식과 협력필터링 접근방식을 모두 사용하여 가중 평균값을 예측 값으로 활용하는 방법을 제안하였다. 처음 시작할 때에는 두 개의 방식에 동일한 가중치를 주어 점수를 예측한 뒤, 두 값의 차이가 큰 사용자의 가중치를 낮게 주어 예측 오차를 줄이는 방법이다. 즉, 사용자의 점수 데이터에 기반 하여 수정된 가중치를 반복적으로 적용함으로써 Grey sheep 문제점을 해결할 수 있다.

#### (4) Shilling Attack

고객들이 아이템에 대해 자유롭게 평가한 사용자-아이템 행렬은 협력필터링에서 고객들의 선호도를 평가하는 중요한 지표로 활용된다. 그러나 아이템이 잘 팔리기 원하는 생산자의 의도적으로 해당 아이템에 긍정적인 평가점수를 입력하여 추천 리스트에 포함되도록 하거나, 반대로 경쟁업체의 특정 아이템에 대하여 부정적인 평가점수를 입력함으로써 다른 고객에게 추천되지 않도록 할 수 있다. 이처럼 악의적으로 평가점수를 입력하여 추천 시스템의 정상적인 작동을 방해하는 행위를 Shilling Attack이라고 한다(Lam *et al.*, 2004). 2001년, Sony Pictures는 새로 출시된 영화를 홍보하기 위해 실제로 존재하지 않는 평가를 조작하였으며, 아마존과 이베이 등 수많은 기업에서도 Shilling Attack 사례가 잇따라 발표되어 문제의 심각성이 대두되었다(Lam *et al.*, 2004). 이후, 많은 연구를 통해 추천 알고리즘이 평범한 수준의 공격에도 쉽게 노출될 수 있다는 사실이 증명되었다(Mobasher *et al.*, 2007). Burke *et al.*(2006)은 공격의 유형을 분석하고 분류기준을 제시하였으며, 조작된 평가점수가 추천 결과에 미치는 영향을 최소화하기 위한 추천 시스템 모델 구축 방법을 제안하였다. 또한, 모델의 신뢰성과 강건함을 높이는 방법 이외에도 허위 사용자 탐지 기법에 관한 연구가 진행되고 있으며, 기계학습 알고리즘을 통해 비정상적인 행위를 미리 학습한 뒤 공격을 탐지하는 기법이 대표적이다(Lee *et al.*, 2012; Cao *et al.*, 2013).

### 3.2 기억 기반 협력필터링

기억 기반 협력필터링 기법은 <Figure 3>에서 설명한 바와 같이, 사용자 기반 협력필터링과 아이템 기반 협력필터링으로 나눌 수 있다. 사용자 기반 협력필터링은 사용자가 입력한 선호도 정보를 이용하여 고객과 유사한 성향을 갖는 이웃 사용자를 선별한 뒤, 선별된 이웃들이 공통적으로 선호하는 아이템을 고객에게 추천해주는 방식이다. 아이템 기반 협력필터링은 특정 아이템이 기준이 되어, 사용자들에 의해 평가된 점수가 유사한 아이템을 이웃 아이템으로 선정한 다음, 이웃 아이템을 평가한 점수를 바탕으로 추천 대상 고객이 특정 아이템 대해 갖게 될 선호도를 예측한다(Sarwar *et al.*, 2001).

#### (1) 유사도 측정

사용자 혹은 아이템간의 유사도를 계산하는 것은 기억 기반 협력필터링 단계에서 가장 중요한 단계이다. 같은 항목에 같은 점수를 준 두 고객이 존재할 경우, 이들의 유사도는 높으며 비슷한 취향을 갖고 있다고 볼 수 있다(Billsus *et al.*, 1998; Herlocker *et al.*, 2000). 즉, 추천대상 고객과 다른 고객들과의 구매 이력을 비교했을 때, 추천대상 고객과 같은 물건을 구매하거나 선호하는 물건이 같은 고객일수록 높은 유사도를 갖게 된다(Sarwar *et al.*, 2001). 대표적인 유사도 측정 척도로는 점수를 기반으로 하는 피어슨 상관계수(Pearson correlation coefficient)와 코사인 유사도가 있으며, 순위를 기반으로 하는 스피어만 순위 상관계수(Spearman's rank correlation coefficient)가 있다(Resnick *et al.*, 1994; Salton *et al.*, 1983; Herlocker *et al.*, 2004).

피어슨 상관계수는 두 개의 속성간의 상관성을 계산하여 1에 가까울수록 양의 상관관계, -1에 가까울수록 음의 상관관계가 있으며, 0은 상관관계가 없음을 나타낸다. 사용자 기반 협력필터링에서 피어슨 상관계수를 통해 두 사용자  $a$ 와  $b$ 의 유사도  $w_{a,b}$ 는 식 (6)와 같이 계산할 수 있다.

$$w_{a,b} = \frac{\sum_{i \in I} (r_{a,i} - \bar{r}_a)(r_{b,i} - \bar{r}_b)}{\sqrt{\sum_{i \in I} (r_{a,i} - \bar{r}_a)^2} \sqrt{\sum_{i \in I} (r_{b,i} - \bar{r}_b)^2}} \quad (6)$$

여기서  $I$ 는 아이템의 전체 집합이며  $r_{a,i}$ 는 사용자  $a$ 가 특정 아이템  $i$ 를 평가한 점수,  $\bar{r}_a$ 는 사용자  $a$ 가 전체 아이템에 준 평균 점수이다.

또한, 아이템 기반 협력필터링에서 피어슨 상관계수를 통해 두 아이템  $i$ 와  $j$ 의 유사도  $w_{i,j}$ 는 식 (7)와 같이 계산할 수 있다.

$$w_{i,j} = \frac{\sum_{u \in U} (r_{u,i} - \bar{r}_i)(r_{u,j} - \bar{r}_j)}{\sqrt{\sum_{u \in U} (r_{u,i} - \bar{r}_i)^2} \sqrt{\sum_{u \in U} (r_{u,j} - \bar{r}_j)^2}} \quad (7)$$

여기서,  $U$ 는 아이템  $i$ 와  $j$ 에 모두 점수를 준 사용자 전체 집합이며,  $r_{u,i}$ 는 사용자  $u$ 가 아이템  $i$ 에 준 점수,  $\bar{r}_i$ 는 전체 사용자가 아이템  $i$ 에 준 평균 점수이다.

코사인 유사도는 내적공간의 두 벡터 간 각도의 코사인 값을 이용하여 두 벡터가 얼마나 유사한지 측정하는 방법으로 식 (8)을 통해 계산되어지며 결과 값이 1이면 같은 성향, 0이면 다른 성향임을 의미한다.

$$w = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{\|\vec{i}\| * \|\vec{j}\|} \quad (8)$$

여기서,  $i$ 와  $j$ 는 사용자 혹은 아이템이며 콘텐츠 이용 횟수나 구매 횟수 등이 벡터로써 사용될 수 있다. 그러나 다양한 고객



들이 서로 다른 평가 척도를 사용할 경우, 코사인 유사도를 사용하여 유사정도를 파악하는 것이 어렵다는 문제점을 가지고 있다. 즉, 사용자  $a$ 는 가장 선호하는 영화에 대해 4점을 주지만 사용자  $b$ 는 5점을 줄 경우, 두 사용자의 실제 선호도가 일치하더라도 코사인 유사도는 다르게 나타날 수 있다.

Sarwar *et al.*(2001)는 이러한 단점을 해결하기 위해 보완 코사인 유사도(Adjusted cosine similarity)를 제안하였다. 식 (9)는 아이템 기반 협력필터링에서 두 아이템  $i$ 와  $j$ 의 보완 코사인 유사도  $w_{i,j}$ 를 구하는 식이다.

$$w_{i,j} = \frac{\sum_{u \in U} (r_{u,i} - \bar{r}_u)(r_{u,j} - \bar{r}_u)}{\sqrt{\sum_{u \in U} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{u \in U} (r_{u,j} - \bar{r}_u)^2}} \quad (9)$$

여기서,  $U$ 는 아이템  $i$ 와  $j$ 에 모두 점수를 준 사용자의 전체 집합이며  $r_{u,i}$ 는 사용자  $u$ 가 아이템  $i$ 에 준 점수,  $\bar{r}_u$ 는 사용자  $u$ 가 전체 아이템에 대해 평가한 점수의 평균값이다. 이는 사용자의 평균 평가점수 대비 특정 아이템에 대한 선호도를 파악할 수 있기 때문에 정규화된 유사도 값을 얻을 수 있으며, 유사도 측정의 정확도를 향상시킬 수 있다.

스피어만 순위 상관계수는 사용자  $a$ 와  $b$ 의 점수를 각각 순위로 변환한 뒤, 차이를 통해 유사도를 측정하게 되며 식 (10)를 통해 계산되어진다.

$$w_{a,b} = 1 - \frac{6 \sum_{i \in I} d_i^2}{n(n^2 - 1)} \quad (10)$$

여기서,  $d_i$ 는 아이템  $i$ 에 대해 두 사용자가 평가한 점수의 순위 차이를 나타내고  $n$ 은 전체 아이템의 개수이며 유사도의 결과 값은 1에서 -1까지의 범위를 갖게 된다. 순위 상관계수는 점수의 분포가 매우 극단적일 경우 유용하게 사용될 수 있는 장점을 지닌 반면, 사용자가 여러 아이템에 대하여 같은 평가점수를 준 경우에는 유사도를 측정하기 어렵다는 한계점을 지니고 있다.

## (2) 선호도 예측

유사도 측정을 통해 추천 대상 고객과 유사한 사용자를 선택한 후, 이들의 과거 구매 이력을 바탕으로 추천 대상 고객의 취향을 예측할 수 있다. 예측 값을 얻기 위한 기법으로는 가중합과 단순가중평균이 있으며 가중합은 추천 대상 고객  $a$ 가 아이템  $i$ 에 대해 갖는 예측 선호도를 나타내고 식 (11)와 같이 계산할 수 있다(Resnick *et al.*, 1994).

$$P_{a,i} = \bar{r}_a + \frac{\sum_{u \in U} (r_{u,i} - \bar{r}_u) \times w_{a,u}}{\sum_{u \in U} |w_{a,u}|} \quad (11)$$

여기서  $\bar{r}_a$ 는 추천 대상 고객이 모든 아이템에 준 점수의 평균이며  $\bar{r}_u$ 는 다른 사용자가 모든 아이템에 준 평균 점수이다.  $w_{a,u}$ 는 추천 대상 고객  $a$ 와 사용자  $u$ 사이의 유사도에 따른 가중치를 나타내는 것으로써 추천 대상 고객과 사용자간의 유사도가 높을수록 큰 가중치를 부여한다.

아이템 기반 협력필터링에서는 단순가중평균 기법을 통해 점수를 예측하며 식 (12)과 같이 계산할 수 있다(Sarwar *et al.*, 2001).

$$P_{a,i} = \frac{\sum_{n \in N} r_{a,n} \times w_{i,n}}{\sum_{n \in N} |w_{i,n}|} \quad (12)$$

여기서  $w_{i,n}$ 은 예측하고자 하는 아이템과 다른 아이템과의 유사도를 가중치로 부여하며, 이는 예측하고자 하는 아이템과 유사한 아이템의 점수를 보다 크게 반영하여 예측 값을 계산하는 방식이다.

Park *et al.*(2014)은 예측 시 가중치를 주는 방법으로 정보검색과 텍스트마이닝 분야에서 널리 쓰이는 TF-IDF(Term Frequency-Inverse Document Frequency)를 활용하였다(Salton *et al.*, 1998). TF는 특정 단어가 하나의 문서 내에서 나타난 빈도이며 DF는 전체 문서군 중에서 특정 단어를 포함하는 문서의 빈도로서, TF-IDF가 높은 단어일수록 적은 수의 문서에 등장하지만 각 문서 내에서의 빈도수는 높다. 추천 시스템에서는 이러한 개념을 적용하여 문서를 아이템으로 평가 점수를 빈도로 나타낼 수 있으며, 이는 특정 사용자가 다른 아이템에 평가한 점수에 비해 특정 아이템에 높은 점수를 주었을 경우 큰 가중치를 주게 된다.

Lee *et al.*(2010b)는 평가시기를 고려해 점수 반영 시 가중치를 적용하는 알고리즘을 제안하였으며 그 결과 예측 정확도를 향상시킬 수 있었다. 이 알고리즘에서는 먼저 유사도 측정을 통해 선정된 유사그룹 내에서 연관성이 적은 사용자들을 제거시켰으며, 평가점수는 사용자의 평가시기에 따라 가중치를 달리 적용하였다. 제안 방법을 Movielens 데이터 셋에 적용시켜 성능을 검증해 본 결과 기존의 협력필터링 기법보다 14.9% 정도 예측정확도가 향상되었음을 확인하였다.

## 3.2.3 상위 $N$ 개 아이템 추천

아이템 추천 단계에서는 추천 대상 고객의 선호도가 가장 높을 것이라 예상되는 상위  $N$ 개의 아이템을 최종적으로 선택하여 추천 대상 고객에게 아이템 목록을 제공한다(Karypis *et al.*, 2001). 이러한 기법은 주로 ‘구매·비 구매’와 같은 분류 문제에 많이 활용되고 있으나, 구매 빈도수 대신 선호 점수를 사용함으로써 선호도 예측 문제에도 활용될 수 있다(Deshpande *et al.*, 2004; Cremonesi *et al.*, 2010).

<Figure 5>는 사용자 기반 상위  $N$ 개 아이템 추천 기법을 보여주고 있다. 추천 대상 고객과 유사도를 측정한 뒤,  $K$ 명의 유사



한 이웃을 선택하고 그들이 구매한 아이템들에 대해 구매 횟수를 계산한다. 최종적으로 추천 대상 고객이 구매하지 않은 아이템들 중에서 구매 빈도수가 높은 순서에 따라 상위  $N$ 개가 추천된다. 즉, ‘아이템 5’는 추천 대상 고객과 구매 패턴이 가장 유사한 3명의 사용자가 가장 많이 구매한 아이템이며 그 다음으로 많이 구매한 아이템은 ‘아이템 1’이다. 이때, ‘아이템 1’은 추천 대상 고객이 이미 구매한 아이템으로 최종 추천 아이템 목록에서 제외된다(Cremonesi *et al.*, 2010).

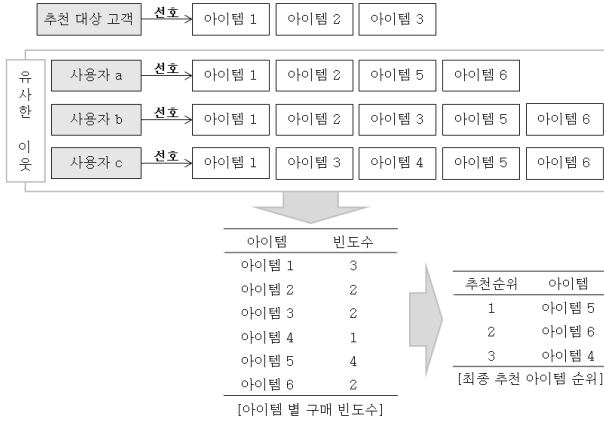


Figure 5. User-based top-n recommendation algorithm

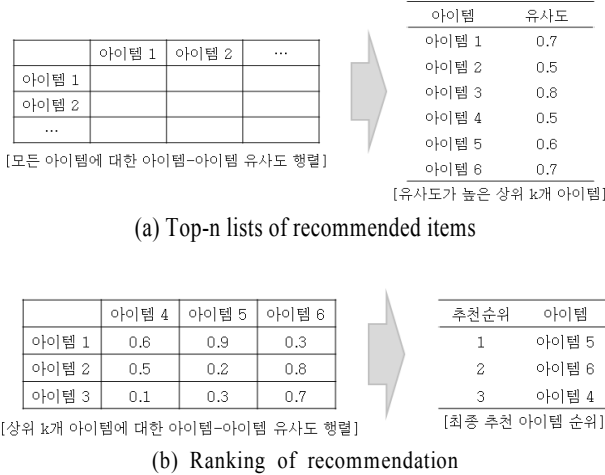


Figure 6. Item-based top-n recommendation algorithm

<Figure 6>은 아이템 기반 상위  $N$ 개 아이템 추천 기법을 보여주고 있다. 먼저, 전체 아이템을 대상으로 아이템-아이템 행렬을 만들며 여기서 행렬 값은 아이템간의 유사도로써 함께 구매되어질수록 큰 값을 갖게 된다. 다음 유사도가 높은 순서대로  $K$ 개의 아이템을 선정한 뒤, 다시 아이템-아이템 행렬을 생성하며 이 때,  $K$ 개의 아이템은 추천 대상 고객이 구매한 아이템과 구매하지 않은 아이템으로 나누어 각각 행과 열로 구성한다. 예를 들어, 유사도가 높은 6개 아이템 중에서 추천 대상 고객이 아이템 1, 아이템 2, 아이템 3을 이미 구매했다면 행렬은 <Figure 6>(b)과 같이 생성된다. 마지막으로, 유사도가 가

장 큰  $N$ 개의 아이템이 추천 대상 고객에게 추천된다(Karypis *et al.*, 2001; Deshpande *et al.*, 2004).

### 3.3 모델 기반 협력필터링

앞서 설명한 기억 기반 협력필터링은 모델을 구축하지 않고, 추천이 요구될 때마다 휴리스틱 기법을 통해 결과를 도출하는 ‘lazy learning’이다(Adomavicius, 2005). 이는 사용자가 설정해야 하는 파라미터 개수가 적다는 장점이 있는 반면 데이터 희소성이나 Grey sheep과 같은 문제점이 나타나고 있다(Herlocker *et al.*, 2000). 특히, 실제 데이터에 적용 할 경우 앞서 언급한 문제가 더욱 심각하게 나타내기 때문에 이를 해결하고자 하는 연구가 진행 중이다(Carrillo *et al.*, 2013). 그 가운데, 학습데이터를 통해 모델을 구축한 뒤, 사용자에게 아이템을 추천하는 모델기반 협력필터링의 우수한 성능이 검증되었다. 즉, 기존의 기억 기반 협력필터링의 유사도 측정, 선호도 예측 등의 과정에서 기계학습과 데이터마이닝 기법을 적용함으로써 기억 기반 협력필터링이 갖고 있는 여러 가지 문제점을 보완할 수 있게 되었다(Breese *et al.*, 1998; Hofmann, 2004). 최근 들어, 데이터에 내재되어 있는 복잡한 패턴을 발견할 수 있는 적절한 모델을 적용하여 선호도 예측 정확도를 높이하고자 하는 연구가 활발히 진행 중이며, 실제 데이터에 적용했을 때 성능이 우수하다는 점에서 효용성을 인정받고 있다(Miyahara *et al.*, 2000; Su *et al.*, 2006; O'Connor *et al.*, 1999; Sarwar *et al.*, 2002; Nikovski *et al.*, 2006; Tipping *et al.*, 1999). 일반적으로 사용자 선호도를 범주화로 나타내는 데이터에서는 분류모델이 사용되며, 연속형 데이터에서는 회귀모델이나 특이값 분해 기법이 사용된다.

#### (1) 나이브 베이즈

나이브 베이즈 모델은 문서분류에서 가장 우수한 성능을 보이고 있는 알고리즘으로 베이즈 정리에 이론적 근거를 두었으며 주어진 문서를 입력 받은 뒤 그것이 각 카테고리에 할당될 확률을 계산하는 방법이다(Domingos *et al.*, 1997). 나이브 베이즈 학습기법은 분류 대상문서에 대해 가장 가능성이 높은 분류 클래스를 정하기 위해 다음 식 (13) 과 같이 계산한다.

$$Class = \arg \max_{j \in class \text{ Set}} p(class_j) \prod_0 p(X_0 = x_0 | class) \quad (13)$$

즉, 사용자가 과거에 선택한 문서의 키워드나 평점 및 선호도 점수를 활용하여 각 클래스로 분류 될 확률을 계산하는 것이며, 이 때 분류 클래스는 사용자에게 의해 다양하게 정의될 수 있다.

Miyahara *et al.*(2000)는 다중 그룹 데이터를 이진 그룹 데이터로 변환한 뒤, 나이브 베이즈 모델을 적용한 협업필터링 기법을 제안하였다. 그러나 데이터 변환 과정에서 다중 그룹에 대한 정보 손실이 크고 계산 속도가 느리다는 문제점을 발견하였다. 이를 보완하기 위해 Su *et al.*(2006)는 다중 그룹에 나이

브 베이즈 모델을 적용하였고, 그 결과 피어슨 상관계수를 적용한 협력필터링보다 예측 정확도는 낮지만 빠른 계산 속도를 나타냄을 확인하였다.

Lee *et al.*(2010a)는 기존 논문에 대해서는 협력필터링 기법을 통해 추천을 하고 새로운 논문에 대해서는 나이브 베이즈 모델로 클래스를 구분하여 추천하는 기법을 제안하였다. 이는 나이브 베이즈 모델을 통해 평가 정보나 이용정보가 부족한 신규 콘텐츠 추천 문제를 해결하였고 콘텐츠기반 접근방식의 가장 큰 문제점인 과도한 특성화 문제의 해결 가능성을 제시하였다.

## (2) 군집화

군집화란 개체간의 유사성을 척도로 하여 비슷한 특성을 가진 개체들이 함께 그룹 지어지도록 하는 기법이다(Han *et al.*, 2006). 대표적인 알고리즘으로는 *K*-means, DBSCAN, OPTICS가 있으며 이러한 군집화 기법은 협력필터링 기법에서 다양하게 활용되고 있다. 특히, 기억 기반 협력필터링에서 유사도를 측정하기 전에 군집화를 먼저 수행한 다음, 같은 군집 내에서 유사도를 계산하는 기법이 연구되었다(O'Connor *et al.*, 1999; Sarwar *et al.*, 2002). 이 방법은 기존 협력필터링에서 유사도를 측정하는 단계에 앞서, 유사한 그룹을 나누는 과정이 추가되었고, 이에 따라 예측 정확도가 높아지는 장점을 가지고 있다. 그러나 데이터를 분할함에 따라 분할 된 데이터의 데이터 희소성 문제가 대두되었고 따라서 실제 데이터에 적용할 경우 성능이 좋지 않음을 확인 할 수 있다. 추후 이러한 문제를 극복할 수 있는 적절한 군집화 기법의 적용과 데이터 차원축소 과정 연구가 진행되어야 할 것이다.

## 3.4 차원 축소

협업필터링에서는 추천 대상 고객과 취향이 비슷한 사용자를 선정하여 그들이 구매한 제품을 추천하게 되는데 사용자가 구매한 상품이 너무 적으면 취향 분석을 통한 상품 추천이 거의 불가능하게 되어 추천 시스템의 성능을 저하시키는 요인이 될 수 있다. 또한 고객과 상품수가 많을 경우 유사도와 선호도 계산에 있어 많은 계산 비용이 발생한다. 이러한 문제를 개선하기 위해 정보검색(Information Retrieval, IR) 분야에서 사용되는 Latent Semantic Indexing(LSI) 기법이 적용되었다(Harshman, 1990; Berry *et al.*, 1995). LSI는 용어와 문서 간의 연관성을 분석하고 정보를 얻기 위해 고차원의 행렬을 저차원의 행렬로 축소시키는 기법으로 추천 시스템의 사용자-아이템 행렬에서도 적용할 수 있다. 즉, 상품의 개수를 줄여 차원을 축소함으로써 취향분석의 정확성을 높이고 빠른 속도로 결과를 도출할 수 있다. 이때 사용된 차원축소 기법으로는 기존행렬을 세 개의 행렬로 분해하는 특이값 분해가 있다(Billsus *et al.*, 1998; Sarwar *et al.*, 2000). 일반적으로, 특이값 분해는 더 작게 분해된 행렬을 이용하여 최적의 근사형을 제공하기 위해 사용되며 다음

식 (14)를 통해 수행할 수 있다.

$$M = U \Sigma V^T \quad (14)$$

여기서, 원본행렬  $M$ 은  $U$ ,  $\Sigma$ ,  $V$ 로 분해되며,  $U$ 와  $V$ 는 직각 행렬이고  $\Sigma$ 은 대각행렬이다. 추천 시스템에서는 각각의 분해된 행렬을 이용하기 위해 특이값 분해 행렬들에서  $K$ 개의 속성을 선택하여 문제에 적용하게 되며 이를 Truncated-SVD라 부르고 식 (15)와 같이 계산할 수 있다. 즉  $K$ 에 따라 차원 축소의 정도가 결정된다(Berry *et al.*, 1995).

$$M_k = U_k \Sigma_k V_k^T \quad (15)$$

Jeong *et al.*(2001)은 원본행렬과 차원축소 된 행렬의 추천 정확도를 비교한 결과, 차원수가 너무 작으면 계산 속도는 빨라 지지만 추천 정확도가 떨어짐을 확인하였는데 이는 차원 축소 시, 적절한 수의 차원 선택이 중요함을 설명해주고 있다. 또한 적절한 차원 수는 자료의 개수와 사용자간의 유사성 등 다양한 요인의 영향을 받으므로 데이터의 특성을 파악하는 것이 중요함을 강조하였다. Lee *et al.*(2002)는 특이값 분해를 통해 변환된 자료 사용이 때에 따라 원래 속성 정보를 상실할 수 있기 때문에, 원본 속성의 가치를 발견하는데 오히려 방해가 될 수 있다는 문제점을 제기하였다. 이를 해결하기 위해 1차적으로 군집화를 진행한 뒤, 엔트로피 가중치와 특이값 분해를 동시에 적용하여 군집의 중요한 속성을 발견할 수 있는 알고리즘을 제안하였다. 이는 행렬의 차원은 축소하되, 원본의 속성을 발견할 수 있는 방법이며 실험을 통해 기존 협력필터링 기법보다 높은 추천정확도를 보여주었다.

## 3.5 그 밖의 협력필터링 기반 추천 시스템

오늘날 스마트 기기와 GPS의 발달로 추천 대상이 되는 고객의 위치정보를 손쉽게 수집할 수 있다. Noh *et al.*(2014)는 뉴스를 읽는 독자의 흥미 대상이 되는 기사는 그들의 위치에 따라 변화한다는 가정을 두고 연구를 시작하였다. 즉, 독자의 취향 뿐만 아니라 기사를 읽을 때 어느 지역에 위치하고 있는지를 함께 고려하여 추천하는 것이 추천 만족도를 향상시킬 수 있다. Noh *et al.*(2014)의 연구에서는 스마트 폰의 GPS 기능을 통해 독자가 뉴스를 읽을 때, 기사를 클릭한 위치의 정보를 수집한 다음 뉴스의 주제와 지역 간의 관계를 Latent dirichlet allocation(LDA) 토픽 모델링을 통해 분석하였다(Blei *et al.*, 2003). 위치 정보를 이용한 또 다른 연구로써 Yin *et al.*(2014)은 LDA 알고리즘을 사용하여 지역에 따른 선호도를 고려하고 이에 따라 추천목록을 선정하는 방식을 제안하였다. 최근에는 이벤트나 지역을 기반으로 한 네트워크가 발달되어 있으므로 추천 대상 고객이 현재 어디에 위치해 있느냐에 따라 추천목록을 달리 해야 함을 강조하고 있다.

지역정보 뿐만 아니라, 상황, 사회관계, 시간 등 수많은 정보의 수집이 가능해짐에 따라 추천 시스템에 사용되는 데이터의 변수가 늘어나고 있다. 일반적으로 구성변수가 많아질수록 계산의 복잡성이 증가되고 예측 성능과 추천 유효성을 저해할 수 있어 변수에 대한 중요도를 파악하고 추천 성능을 극대화시키는 것이 필요하다. Park *et al.*(2012)는 각종상황, 사회관계, 시간 등을 각각 변수로 두어 수많은 변수 중에서 추천 만족도에 큰 영향을 미치는 변수만을 선택하여 선호도 예측에 사용하였다. 그 결과 구매이력 데이터만을 활용하거나 모든 정보를 활용하였을 때보다 추천 성능이 우수하였고, 특히 필요한 정보만을 사용함으로써 계산비용을 절감할 수 있었다.

#### 4. 하이브리드 시스템

사용자에게 만족도 높은 추천결과를 제공하기 위해서는 사용자의 현재 상황을 인지하고 선호도의 특성을 정확히 파악한 뒤, 이에 적합한 아이템을 추천하여야 한다. 어떤 경우에는 지금까지 보았던 영화와 분위기가 비슷한 영화를 찾을 수도 있고, 혹은 지금까지 본 영화에 등장했던 배우가 나온 다른 영화를 찾을 수도 있다. 즉, 추천 시스템에서 사용되는 알고리즘들은 각각 추천의 기준이 다르기 때문에 추천 시스템을 적용하고자 하는 분야의 특성에 알맞은 알고리즘을 선택하고 추천 성능을 향상시킬 수 있는 가능한 많은 정보를 이용하여야 한다(McNee *et al.*, 2006; Ducheneaut *et al.*, 2009).

앞서 살펴본 바와 같이 콘텐츠기반 접근방식과 협업필터링 방식은 각각의 장·단점을 가지고 있다. 콘텐츠기반 접근방식은 평가가 되지 않은 아이템을 추천할 수 있지만, 과도한 특수화 경향이 있으며 반대로 협업필터링은 높은 세렌디피티(*serendipity*)를 보여주지만, 사용자가 평가하지 않은 아이템에 대해서는 추천이 되지 않는다. 따라서 최근에는 각 방식의 장점을 극대화하면서 단점은 보완하고 다양한 정보를 효과적으로 활용할 수 있는 하이브리드 추천 시스템에 대한 연구가 중요시되고 있다(Burke, 2002; Al *et al.*, 2002; Barragáns-Martínez *et al.*, 2010).

Bruke(2002; 2007)은 추천 시스템의 하이브리드 전략에 대하여 다음과 같이 분류하였다. 첫째는 다른 추천 기준을 지닌 여러 개의 알고리즘을 학습한 뒤, 각 알고리즘이 아이템 추천 점수의 가중평균합을 구하는 방법이다. 이 방법은 여러 추천 알고리즘들의 결과를 전반적으로 이용할 수 있는 장점이 있지만, 각 추천 점수를 정규화하여야 하며, 각 추천 결과의 가중치를 잘 정의하여야 한다. 두 번째 전략은 학습된 여러 개의 추천 엔진 중에서 현재의 상황에 가장 적절한 추천 엔진을 선택하는 방법이다. 이는 상황에 가장 적합한 추천 기준을 이용할 수 있지만, 현재의 상황을 인지하기 위한 추가적인 계산이 필요하다. 세 번째 전략은 각 알고리즘들의 추천 결과를 혼합하여 보여주는 방법으로 추천 결과의 다양성을 높게 보여줄 수 있으며,

네 번째로 각각의 알고리즘에 사용되는 모든 변수를 하나의 알고리즘의 변수로 병합하여 이용할 수도 있다. 다섯 번째로 한 알고리즘이 추천한 아이템을 다음 알고리즘의 후보로 이용하여 각 단계별로 더 세밀하게 추천하거나 한 알고리즘의 추천 점수를 다른 알고리즘의 변수로 이용하는 것이다. 마지막으로 각각의 알고리즘의 추천 점수를 바탕으로 메타 알고리즘을 학습하는 앙상블 방법 역시 이용될 수 있다.

Adomavicius and Tuzhilin(2005)는 하이브리드 전략을 모델의 형태에 따라 크게 네 가지로 분류하였다. 첫째는 독립된 추천 결과를 조합하는 것이다. 이는 조합 방법에 따라 Bruke(2002; 2007)의 첫 번째, 두 번째, 그리고 세 번째 전략에 해당한다. 둘째는 콘텐츠기반 정보를 협업필터링에 적용하는 것으로, 사용자의 평가점수가 아닌 콘텐츠기반 사용자 프로파일을 이용하는 방법들이 해당한다. 셋째는 LSI, PLSI와 같은 알고리즘을 이용하여 협업필터링의 정보를 콘텐츠기반 접근방식에 융합하는 것이다. 마지막으로 협업필터링과 콘텐츠기반 접근방식을 동시에 고려하는 단일 모델을 구축하는 방법이 있다. 이는 사용자  $i$ 의 아이템  $j$ 에 대한 예측점수  $r_{ij}$ 를 다음과 같이 정의한다.

$$r_{ij} = x_{ij}\mu + z_i\gamma_i + w_j\lambda_j + e_{ij} \quad (16)$$

여기서,  $x_{ij}$ 는 사용자-아이템간의 정보이고,  $z_i$ 와  $w_j$ 는 각각 사용자와 아이템의 변수이다. 그리고  $\mu, \gamma_i, \lambda_j, e_{ij}$ 는 각각의 항목에 해당하는 상수 혹은 변수들이다. 단일 모델을 학습하기 위해서 MCMC와 같은 추정모델이나 베이지안 학습법이 이용되기도 한다.

McNee *et al.*(2006)은 논문 추천 시스템에서 콘텐츠기반과 협업필터링에 사용할 수 있는 알고리즘을 제시하였으며, 각 알고리즘의 특성과 장·단점을 비교하였다. 이웃기반 협업필터링을 이용할 경우에는 세렌디피티가 높은 경향이 있으며, 나이트 베이지안 판별기를 이용할 경우에는 선택한 논문이 많이 참조된 논문을 먼저 추천된다. 사용자-아이템 행렬을 차원 축소 방법 중 하나인 PLSI를 통해 학습한 뒤, 이를 바탕으로 논문을 추천할 경우에는 선택한 논문과 분야적으로 매우 유사한 논문들이 우선적으로 추천되었다. 또한 TF-IDF를 이용하여 콘텐츠기반 추천을 할 경우에는 선택한 논문들과 내용이 매우 유사한 논문들이 우선적으로 추천되었다. 이와 같이 선택한 알고리즘에 따라 추천의 기준이 달라지기 때문에 상황에 적합한 알고리즘을 현명하게 선택할 수 있어야 한다.

앞서 살펴본 바와 같이, 성격이 다른 추천 알고리즘을 어떻게 조합하느냐에 따라서 해결할 수 있는 문제는 다양하다. Claypool *et al.*(1999)는 여러 추천 결과에 대해 적절한 가중치를 계산하기 위하여 초기 가중치를 모두 동일하게 설정한 뒤, 추천 결과에 대한 사용자 피드백 정보를 활용하여 가장 설명력이 좋은 가중치를 실시간으로 조절하는 방법을 제안하였다. 이와 비슷하게 Fan *et al.*(2014)는 MovieLens 데이터에 대해서

콘텐츠기반 접근방식과 협업필터링의 추천 점수를 40% : 60%로 선형 결합할 경우 가장 높은 추천 만족도를 나타냄을 확인하였다. Kim *et al.*(2002)는 두 가지 방식을 병합하여 동시에 적용하는 것이 아니라, 구매이력이 존재하는 사용자에게 대해서는 협력필터링을 통해 추천 하고, 구매이력이 없어서 협력필터링 기법의 적용이 불가능한 사용자에게 대해서는 사용자 프로필을 사용하여 유사도를 측정하는 기법을 제안하였다. 이 방법은 새로운 고객에게 보다 효과적인 추천을 할 수 있다는 장점을 가지고 있다.

협업필터링의 경우 사용자-아이템 행렬의 밀도가 매우 낮기 때문에, 코사인거리나 피어슨 상관계수를 이용할 경우, 유사한 사용자가 존재하지 않아 유사도 값을 도출할 수 없는 데이터 희소성 문제가 발생한다. 이를 해결하기 위하여 Melville *et al.*(2002)는 아이템을 선택한 사용자 정보를 통해, 아이템 간의 유사도를 계산하였다. 이를 이용하여 사용자-아이템 행렬에 각 사용자가 선택한 아이템과 유사한 다른 아이템을 가상으로 선택한 것으로 여기는 유사 사용자 벡터를 정의하였다. 그 뒤, 밀도가 높아진 사용자-아이템 행렬을 이용하여 이웃기반 협업필터링을 수행하였다. Goldberg *et al.*(2001)은 데이터 희소성 문제를 해결하기 위하여 PCA를 수행한 뒤, 이를 바탕으로 클러스터링을 수행하여 유사한 이웃을 찾았다. SimRank는 데이터 희소성 문제를 해결하는 또 다른 방법이다(Jeh and Widom, 2002). SimRank는 사용자와 아이템의 이분 그래프를 구축하여 사용자-아이템의 유사도를 정의한다. 두 사용자의 유사도는 각 사용자가 선택한 아이템 간의 평균 유사도로 정의되고, 아이템 간의 유사도는 이를 선택한 사용자간의 평균 유사도로 회귀적 정의된다. 따라서, SimRank를 이용할 경우 동일한 아이템을 선택하지 않더라도 유사한 아이템을 선택할 경우 두 사용자가 높은 유사도를 지닐 수 있다. Sawant(2013)는 이분 그래프에 클러스터링을 수행한 뒤, random walks model을 적용하여 개인 사용자 혹은 클러스터 간의 영향도를 계산하여 협업필터링을 수행함으로써 데이터 희소성 문제를 해결함과 동시에 사용자간의 추천 영향력을 계산하였다.

협업필터링에서 이용하는 사용자-아이템 행렬 외의 다른 정보를 함께 이용하는 하이브리드 추천 연구들도 이루어졌다. Vozalis and Margaritis(2004)는 이웃기반 협업필터링을 통하여 비슷한 이웃 후보군을 선택한 뒤, 인구통계학(demographic) 정보를 이용하여 추천할 사용자와 유사한 이웃들의 정보만을 이용하는 연쇄방법을 제안하였다. Chow *et al.*(2014)는 이분 그래프에 사용자의 선호정보를 결합하여 Personalized PageRank를 적용한 개인화 추천 알고리즘을 고안하였고, Basilico and Hofmann(2004)는 사용자-아이템 행렬과 사용자-아이템의 특성 변수들을 모두 종합하여 사용자-아이템의 유사도 커널을 정의하는 프레임워크를 제안하였다. Ganu *et al.*(2009)은 아이템에 리뷰가 작성되는 경우에는 각 아이템의 특성이나 사용자의 선호를 더욱 자세히 분석할 수 있다고 가정하였다. 따라서, 리뷰 데이터로부터 추출된 토픽과 의미 벡터를 이용하여 이웃기반 협업필터

링을 보강하였으며, 궁극적으로 사용자나 아이템의 해당 리뷰 유무에 따라 적절한 추천 알고리즘을 선택할 수 있는 방법을 제안한 뒤 그 성능을 입증하였다. McAuley and Leskovec(2013)는 리뷰 데이터로부터 토픽을 추출 시, 기존의 LDA와 같은 토픽 모델링 기법은 사용자 평가 점수를 이용하지 않는 점을 보완하기 위하여 리뷰 데이터와 점수를 모두 이용하여 토픽을 추출하는 Hidden Factors as Topics(HFT)를 제안하였다. 그리고 이를 이용한 리뷰 데이터 기반의 점수 예측 시스템의 성능을 평가하였다. Ling *et al.*(2014)는 사용자들이 선택하는 아이템 들에는 숨겨진 변수가 있다고 가정할 뒤, 숨겨진 변수를 리뷰 데이터로부터 추출하였다. 그 뒤, 사용자가 선택한 아이템과 유사한 속성을 지닌 다른 아이템을 추천하였다.

이와 같이 하이브리드 추천 방식은 협업필터링과 콘텐츠기반 접근방식의 장·단점을 파악하여, 사용자가 현재 필요로 하는 아이템을 효율적으로 찾기 위한 다양한 정보를 최대한 활용한다. 또한, 이 과정에서 발생하는 다양한 문제들을 해결하기 위한 추가적인 방법들이 함께 결합 된다. 하지만 각 알고리즘과 데이터로부터 얻을 수 있는 정보는 각각 다르기 때문에, 하이브리드 추천 시스템을 설계하는 과정에서 어떤 목적의 추천을 하려는지 명확하게 정의해야 하며, 그 목적에 적합한 알고리즘과 데이터를 활용하여야 한다.

## 5. 연관성 분석

장바구니 분석으로도 불리는 연관성 분석은 데이터마이닝의 대표적인 기법 중 하나로써 개인화 추천 서비스에 널리 활용되고 있다(Oh, 2004). 연관성 분석의 핵심은 조건부 확률로써 “사건 A가 일어났을 때, 사건 B가 일어나는 것”을 의미하며 추천 시스템에서는 “고객이 A에 대해서 만족(혹은 구매)할 경우, B에 대해서 만족 한다”를 의미한다. 이때 A의 정보로써 고객의 나이, 성별, 지역과 같은 프로필정보나 과거 구매이력 혹은 설문지 응답으로부터 얻은 취향 정보 등을 모두 활용할 수 있으며, B는 추천 대상 고객에게 추천할 항목으로 정의할 수 있다. 연관규칙에 대한 판단 기준은 전체 발생 빈도수를 의미하는 지지도와, 조건부 확률을 통해 항목 간 관련 정도를 측정하는 신뢰도가 사용된다. 연관성 분석은 항목의 동시출현 빈도수를 기반으로 추천항목의 우선순위가 정해지므로 데이터 희소성과 Grey sheep 문제를 해결할 수 있다. 그러나 아이템 수와 사용자의 수가 많을 경우, 데이터 희소성 문제점이 대두되어 연산이 복잡해지고 계산량이 많아질 수 있다.

Kim *et al.*(2012)는 이러한 문제를 해결하고자 지지도와 신뢰도 계산 시 임계값을 설정하여 특정 항목집단에 대해서만 분석을 시행하는 방법을 제안하였다. 또한, Jin *et al.*(2010)는 Recency Frequency Monetary(RFM)기법을 사용해 고객 그룹을 세분화 한 뒤, 그룹 내에서 연관성 분석을 수행하여 교환구매 전략을 수립하였다. 연관규칙 없이 임의로 추천했을 때와 RFM



점수대 별로 연관규칙을 생성하여 추천했을 때의 만족도를 평가한 결과 RFM 점수가 높은 고객들로부터 생성된 연관규칙에 대해 높은 만족도를 나타냄을 확인하였다.

## 6. 시간적 역동성을 고려한 추천 시스템

추천 시스템의 요소인 아이템과 사용자들의 특징은 시간이 지남에 따라 변화한다. 예를 들어, 온라인 쇼핑물의 상품은 계절적, 혹은 이벤트에 의한 일시적인 유행이 발생하고, 영화나 음악의 경우에는 오래 전 영화보다는 최근에 발매된 콘텐츠들의 소비가 많은 것이 일반적이다. 특히 YouTube 처럼 서비스 내부아이템의 수명주기가 매우 짧은 경우에는 한 시점에서 집중적으로 아이템들이 소비되고, 어느 정도 시간이 지나면 거의 소비가 이뤄지지 않기도 한다(Baluja *et al.*, 2008). 에어컨이 한 겨울에는 판매되지 않는 것처럼 계절에 따라 판매되는 상품이 다른 현상을 컨셉 이동(concept drifting)이라 한다. 그 뿐 아니라 여성용 화장품만 구매하던 고객이 결혼 후에는 아이 용품을 구매하듯이 사람의 취향 역시 시간이 지남에 따라 변화한다. 이러한 시간적 역동성을 추천 시스템 내부에 반영하기 위한 많은 연구들이 제안되었다.

시간적 역동성을 반영하는 한 가지 방법으로 일정 기간 동안의 데이터만을 사용하여 추천 시스템을 학습하는 무빙 윈도우(moving window)를 사용하거나, 각 상품이 구매된 시점을 기준으로 시간의 흐름만큼 가중치를 줄여주는 감쇄 함수를 이용할 수 있다. 무빙 윈도우를 사용할 경우, 데이터를 지속적으로 무빙 윈도우 내의 데이터만 사용하기 때문에 학습 비용이 줄어드는 장점이 있지만, 전체 데이터를 활용하지 않는다는 점과 적절한 윈도우 사이즈를 정해야 하는 한계점이 있다.

Koren(2010)은 감쇄 함수를 협업필터링과 행렬 분할 알고리즘에 내재하는 방법을 제안하였다. 그는 Netflix Price 데이터를 이용하여 제안된 방법론에 대한 성능 평가를 하였는데 흥미롭게도 사용자의 선호나 트렌드가 바뀐다는 일반적 믿음과는 반대로, 시간에 따른 가중치를 변화하지 않았을 때 모델의 예측력이 가장 높게 나왔다. 이는 두 가지 방향으로 해석할 수 있는데, 첫째는 특정 도메인에 대해서는 오래된 정보라도 모두 이용하는 것이 추천 성능을 더 높일 수 있고, 두 번째는 역으로 사용된 추천 엔진의 방법론이 시간적 역동성을 잘 반영하지 못한다는 것이다.

사용자의 선호를 장기적 선호와 단기적 선호로 나눈 뒤, 이를 조합하는 형식으로 실시간으로 사용자의 선호를 반영한 추천 엔진을 만드는 방법도 제안되었다(Xiang *et al.*, 2010). 아이템의 구매 시간을 몇 개의 시점으로 묶은 뒤 <사용자, 아이템, 시즌>의 마디를 지니는 tripartite 그래프를 형성한 뒤, random walk 모델을 이용하여 각 선호자의 장기적 선호와 한 시점에서의 단기적 선호를 계산하였다.

Lathia *et al.*(2009)는 각 시점에 가장 적합한 추천 모델을 선

택하여 이를 반영하는 방법을 제안하였다. 그는 사전에 여러 개의 협업필터링 모델  $M = \{M_1, \dots, M_m\}$ 을 학습한 뒤, 't-1' 시점에서의 추천 결과를 이용하여 각 협업필터링 모델들을 테스트하고 추천 성능이 가장 좋은 모델  $M_t$ 를 t시점의 추천 모델로 사용하였다. 이와 같은 형태의 추천엔진은 하나의 추천 엔진으로는 잘 반영되지 않는 시즌별 특정 추천 규칙 등을 여러 개의 추천 모델에 나누어 학습시킴으로서, 각 시점에 적합한 추천 모델을 유동적으로 선택하는 효과를 얻을 수 있었다.

이와 같이 시간적 역동성을 반영하는 추천 시스템의 경우 평가 방법 역시 역동성을 반영하는 형태로 진행되어야 한다. 일반적으로 학습 데이터와 평가데이터를 나누기 위해서 사용자의 아이템 구매기록 행렬이나 점수평가 행렬에서 임의로 정보를 지운 뒤, 이를 얼마나 잘 예측하는지 정확도를 계산한다. 그러나 사용자의 아이템 구매나 평가는 시간이 흐름에 따라서 순차적으로 이뤄지기 때문에 Baluja *et al.*(2008)는 평가 데이터를 시간 기준으로 나누는 방식을 제안하였다. 그들은 YouTube의 동영상 추천 엔진의 성능을 평가하기 위하여 92일 동안의 데이터를 수집한 후 초반 46일간의 데이터를 학습데이터로 이용하여 47일째 콘텐츠를 추천하였다. 추천 시스템의 종합적 성능은 각 시점에서 precision과 recall 값을 구한 후 이를 평균하여 측정하였다. 이러한 평가 과정에는 몇 가지 데이터 정제 과정이 필요하다. 그 예로 시간이 지남에 따라서 학습기간에는 존재하지 않으나 테스트기간에 새로 유입된 사용자나 콘텐츠들이 존재하는 경우이다. 이 경우 가장 간단한 방법은 학습기간과 테스트기간에 모두 등장한 콘텐츠와 사용자만을 고려하거나, 다른 방법의 적용을 적용해야 한다.

## 7. 다양성 향상을 위한 추천 시스템

추천 시스템의 성능 향상 연구는 주로 정확성을 높이는 방향으로 이뤄져 왔으나, 그 외 다른 관점에서 추천 시스템을 바라보는 연구들이 제안되었으며 그 중 하나는 다양성이 높은 추천 시스템을 만드는 것이다.

오래 전부터 추천 시스템은 사용자의 선호에 적합한 아이템을 추천함으로써 시스템 전체의 아이템 소비에 다양성을 높여주는지, 혹은 반대로 자주 노출된 아이템을 우선적으로 제공함으로써 아이템 노출의 빈익빈 부익부 현상을 가져오는지에 대한 논의가 이뤄져 왔다. Fleder *et al.*(2009)는 협업필터링의 과정을 시뮬레이션으로 재현한 뒤, 추천 시스템이 전체 사용자 소비의 다양성을 감소시킬 위험이 있음을 언급했다. 그럼에도 불구하고 개인 사용자의 입장에서 추천 시스템에 의하여 자신의 선택의 폭이 다양하게 느껴지는 것은 개인이 선택하는 아이템의 개수가 단순히 늘어났기 때문이라는 지적도 있다.

일반적으로 다양한 아이템들이 추천되었을 때 추천 시스템에 대한 만족도는 높아지는 것으로 알려져 있다. 만약 추천의 정확도는 높다고 하더라도, 모든 사용자에게 비슷한 아이템을

## 분류

추천한다거나, 한 사용자가 매번 비슷한 아이템을 추천받는다면, 추천 시스템에 대한 만족도나 신뢰도는 떨어질 것이다. 하지만 무작위로 다양한 아이템을 추천하는 시스템 역시 사용자에게 충분한 만족도를 줄 수 없다. 이를 반영하기 위해 Lathia *et al.*(2010)는 일정 수준 정확도를 유지하며 다양성을 추구하는 것이 만족도를 가장 높일 수 있음을 언급하였다.

추천 시스템의 다양성을 높이는 연구는 각 개인에게 추천되는 아이템리스트의 다양성을 높이는 것과 시스템이 추천하는 아이템 전체의 다양성을 높이는 두 가지 방향으로 연구가 되고 있으나, 일반적으로 전자의 연구가 더 활발히 진행 중이다(Adomavicius *et al.*, 2012).

개인 사용자 추천 리스트의 다양성을 높이는 연구 중 하나로 Adomavicius *et al.*(2012)는 임의의 추천 엔진을 사용하여 아이템리스트를 추천한 뒤, 사전에 설정한 추천 점수의 하한을 넘은 아이템에 대하여 유명도의 역순으로 아이템 순위를 계산하여 상위 N개의 아이템을 선택함으로써 추천 점수가 높으면 서도 사용자들에게 다양한 아이템을 방법을 제안하였다. Ziegler *et al.*(2005)는 사용자의 관심 영역을 최대한 포함하는 추천 아이템리스트를 만들기 위하여 내부 리스트 유사도 개념을 제안하였는데 이는 아이템 간의 유사도가 코드로 아이템의 순서를 정함으로써 사용자들에게 비슷한 종류의 아이템들이 등장하지 않도록 하였다. Zhang *et al.*(2008)은 이러한 문제를 각 아이템 간 거리행렬을 계산한 뒤, 이를 이용하여 추천 리스트에 포함된 아이템의 거리를 최대화하는 최적화 문제로 정리하였다.

시스템이 추천하는 아이템 전체의 다양성을 높이기 위한 연구도 수행되었다. Park *et al.*(2008)은 추천 시스템의 다양성이 낮은 이유로 롱테일에 위치한 아이템들은 협업필터링에서 정보가 부족하여 추천이 잘 되지 않기 때문이라고 주장하였다. 이를 반영하기 위해 롱테일에 위치한 아이템을 군집화한 후, 한 군집 안에서 다른 아이템의 평점이나 추천 점수 등을 추측하여 리뷰나 평점이 부족한 롱테일의 아이템의 정보를 추정하였다. Schein *et al.*(2002)는 특히 새로 개봉한 영화의 경우 일시적으로 롱테일에 위치할 수 있는데 이러한 문제를 해결하기 위하여 한 영화에 출연한 배우들의 추천 점수를 계산한 뒤, 이들의 합을 영화의 추천 점수로 이용하는 콘텐츠기반 추천 방법을 제안하였다.

## 8. 평가 방법

추천 시스템 기법으로부터 얻어진 추천 결과의 효용성을 파악할 수 있는 다양한 성능 평가방법이 제안되었다. 평가방법은 크게 데이터의 유형과 평가 목적에 따라서 나눌 수 있다. 데이터 유형이 연속형 데이터인 경우에는 예측 정확도로 평가하며, 범주형 데이터인 경우 분류 정확도로 평가한다. 또한 평가 목적에 따라 추천 시스템의 추천 정확도, 다양성, 의외성 등으

로 평가할 수 있다. 추천 정확도는 아이템의 점수를 예측하는 알고리즘이 사용되고 실제 선호도와 예측 값의 차이로 계산하며 널리 사용되는 방법으로 Root Mean Squared Error(RMSE), Mean Average Error(MAE) 등이 있다. 분류 정확도는 선호도가 높을 것이라 예측한 상위 N개의 아이템에 대해 추천 성능을 평가할 때 사용되며 대표적으로 Precision, Recall, F1기법, Receiver operating characteristic(ROC) 등이 있다. 다양성 및 의외성의 측정을 위해서는 아직 통용되는 수리적 공식이 없기 때문에 얼마나 많은 아이템들이 추천이 되는지에 관련된 커버리지(coverage), 각 사용자에게 얼마나 흔하지 않는 아이템을 추천하는지에 대한 세렌디피티 혹은 특이성(novelty), 그리고 얼마나 다양한 아이템을 추천하는지에 대한 다양성(diversity)에 대한 개념들을 수리적으로 표현하기 위한 연구들이 제안되었다.

추천 시스템의 목적은 사용자가 선택할 가능성이 높은 아이템을 추천하는 것이지만, 더 나아가 사용자에게 충분한 만족감을 주어 시스템에 대한 신뢰도를 향상시킴으로써 오래도록 시스템을 이용할 수 있도록 함에 있다. 그렇기 때문에 좋은 추천 시스템의 기준은 추천 성능의 정확도 뿐 아니라, 심리적 요소나 인터페이스적 요소 역시 포함되어야 하며, 이를 위하여 통계 및 정보과학 뿐 아니라 휴먼컴퓨터 인터랙션 분야에서도 추천 시스템과 관련된 연구가 함께 진행되어야 한다.

### 8.1 점수 예측 알고리즘의 평가방법

점수 예측 알고리즘의 예측 성능을 평가하기 위해서는 예측 점수와 실제 점수의 차이를 비교하는 것이 일반적이다. 본 논문에서는 이를 바탕으로 가장 널리 쓰이는 평가척도에 대해 알아보았다. 먼저 Mean Squared Error(MSE)는 각각의 예측 점수와 실제 점수의 차이를 제곱한 후 이를 평균한 값으로써 식 (17)와 같다.

$$MSE = \frac{1}{N} \sum (p_{i,j} - r_{i,j})^2 \quad (17)$$

여기서,  $p_{i,j}$ 는 예측 점수이고  $r_{i,j}$ 는 실제 점수며, N은 총 데이터의 개수이다. MSE는 오차에 제곱을 취함으로써 오차가 큰 값에 대해 가중치를 높게 부여하는 평가방법이다.

Root Mean Squared Error(RMSE)는 Netflix prize에서 사용된 평가 방법으로 식 (18)와 같다.

$$RMSE = \sqrt{\frac{1}{N} \sum (p_{i,j} - r_{i,j})^2} \quad (18)$$

RMSE는 MSE 값의 제곱근으로써, MSE 값에 비하여 예측오차가 큰 관측치에 대해 상대적으로 적은 가중치를 부여한다.

MAE는 식 (19)을 통해 계산할 수 있다(Goldberg *et al.*, 2001; Herlocker *et al.*, 2004).

1. 정밀도
2. 재현율
3. F1 점수
4. TP, TN, FP, FN

$$MAE = \frac{\sum |p_{i,j} - r_{i,j}|}{N} \quad (19)$$

여기서,  $N$ 은 전체 평가 대상의 개수이며,  $P_{i,j}$ 는 예측 점수,  $r_{i,j}$ 는 실제 점수이다. 즉, 예측 점수와 실제 점수간의 차이에 절대값을 취한 뒤, 이들의 합을 전체 평가 대상의 개수로 나눠줌으로써 얻을 수 있다. MAE는 오차의 절대값의 평균이며, 오차의 크기에 상관없이 모두 같은 가중치를 갖게 된다.

다양한 종류의 데이터는 각기 다른 점수 스케일을 가질 수 있기 때문에 이를 정규화 시키기 위한 목적으로 Normalized mean absolute error(NMAE)가 고안되었으며 이를 식 (20)에서 보여주고 있다(Herlocker et al., 1999; Goldberg et al., 2001). NMAE는 MAE값에 최대점수와 최소점수의 차로 나누어 줌으로써 정규화된 결과를 도출할 수 있다.

$$NMAE = \frac{MAE}{r_{\max} - r_{\min}} \quad (20)$$

## 8.2 아이템 추천 알고리즘의 평가방법

선호 정도를 알아보는 것이 아니라 ‘구매-비 구매’, ‘상품 관람-비 관람’ 등 분류가 목적인 문제에서는 기본적으로 <Table 4>와 같은 혼돈행렬을 기반으로 평가가 이루어진다. ‘a’는 추천한 아이템이 실제로 구매된 경우, ‘d’는 추천하지 않은 아이템이 실제로도 구매되어지지 않은 경우, ‘c’는 추천하였으나 고객이 구매하지 않은 경우, ‘b’는 추천하지 않은 아이템을 고객이 구매한 경우를 나타낸다.

Table 4. 2-by-2 confusion matrix

실제/예측	추천	비 추천
구매	a	b
비 구매	c	d

가장 일반적으로 분류 성능을 평가하는 척도는 Mis classification ratio이며, 전체 아이템 중에서 추천과 비 추천을 옳게 분류한 확률로 식 (21)와 같이 나타낼 수 있다.

$$Mis\ classification\ ratio = \frac{a+d}{a+b+c+d} \quad (21)$$

또한, 추천 시스템은 많은 아이템으로부터 소수의 추천 아이템을 선정하는 것으로 ‘정보검색’의 개념과 유사하기 때문에 정보검색 성능 평가에 사용되는 방법이 사용되기도 한다. 가장 잘 알려진 평가방법으로는 Precision과 Recall이 있으며 이를 다음 식 (22), 식 (23)에서 보여주고 있다(Salton et al., 1983; Frakes et al., 1992).

$$Precision = \frac{\text{옳게 추천한아이템 개수}}{\text{추천한 전체아이템 개수}} = \frac{a}{a+c} \quad (22)$$

$$Recall = \frac{\text{옳게 추천한아이템 개수}}{\text{고객이 실제로 구매한아이템 개수}} = \frac{a}{a+b} \quad (23)$$

F-measure는 Precision과 Recall의 정보를 모두 반영하여 하나의 값으로 산출하기 위해 고안되었으며 다음 식 (24)와 같다.

$$F-measure = \frac{2 \times Precision \times Recall}{Precision + Recall} = \frac{2}{1/Precision + 1/Recall} \quad (24)$$

추천 상품 수가 커질수록 recall 값은 증가하지만 precision 값은 감소하게 되며 이러한 상충관계까지 고려하였을 때 분류의 효율성을 평가하는 척도로 F-measure가 사용될 수 있다. F-measure 값이 ‘1’에 근접할수록 recall 값과 precision 값 모두 높다는 것을 의미하며, ‘0’에 근접할수록 둘 중 하나의 값이 상대적으로 낮은 값을 의미한다.

## 8.3 기타 정확도 기반 평가방법

많은 연구에서 정확도 기반 평가방법이 사용되지만, 실제 현업에서는 lift나 hit rate와 같이 실제로 추천 시스템으로부터 얻을 수 있는 이익을 측정하는 지표들을 더 선호하기도 한다 (Konstan et al., 2012; Rashid et al., 2006).

Rashid et al.(2006)은 실제로 2점의 점수를 받은 아이템에 대하여 5점으로 예측하는 것과, 5점의 아이템에 대하여 2점으로 예측하는 것은 서로 비용이 다를 수 있지만 MAE 등의 지표로는 이를 반영할 수 없기 때문에 효용함수를 이용한 평가방법을 제안하였다. 실제점수( $R$ )와 예측 점수( $\hat{R}$ )의 차이에 의한 효용행렬을 만든 뒤, 식 (25)을 이용하여 시스템의 효용을 계산할 수 있다.

$$EU = \sum_{1 \leq i,j \leq 10} U(\hat{R}_i, R_j) P(\hat{R}_i, R_j) \quad (25)$$

여기서,  $P(\hat{R}_i, R_j)$ 는  $j$ 점을  $i$ 점으로 예측한 확률이고  $U(\hat{R}_i, R_j)$ 은  $j$ 점을  $i$ 점으로 예측했을 때의 효용이다.

Deshpande et al.(2004)는 시스템의 성능을 측정하기 위한 방법으로 사용자들이 추천된 아이템을 선택하는 행위를 수치화한 hit-rate와 hit-rank를 제안하였다. hit-rate는 추천되었던 아이템  $n$ 개 중에서 사용자가 실제로 선택한 아이템의 개수의 비율이며 식 (26)과 같이 계산될 수 있다.

$$hit-rate = \frac{\text{Number of hits}}{n} \quad (26)$$

hit-rank는 선택한 아이템의 순위를 가중치로 이용한 방법으



로 아이템  $i(1 \leq i \leq 10)$  의 위치인  $p_i$  역수의 평균으로 정의하였으며 다음 식 (27)과 같다.

$$hit-rank = \frac{1}{n} \sum_{i=1}^h \frac{1}{p_i} \quad (27)$$

#### 8.4 다양성 기반 평가방법

성공적인 추천 시스템에 있어 정확도 향상에만 초점을 두는 것은 한계가 있다는 의견이 대두되었고, 그들은 추천 시스템이 혼하지 않으면서도 흥미롭고 다양한 아이템을 추천해줄 수 있을 때 소비자의 만족도가 향상된다고 주장하였다(Ziegler et al., 2005; Adomavicius et al., 2012; Lathia et al., 2010; Ziegler et al., 2005; Zhang et al., 2008; Zhou et al., 2010; McNee et al., 2006; Konstan et al., 2012). 하지만 정확도만으로는 이러한 비수치적 정보들을 잘 반영할 수 없기 때문에 새로운 측정지표가 필요하며, 이러한 지표들은 사용자의 만족과 같은 심리적이거나 인지적인 경우가 많기 때문에 수치적으로 표현하는데 한계가 있다.

커버리지는 전체 아이템 중에서 추천 시스템이 추천하는 아이템의 비율을 뜻하며, 추천되는 아이템의 종류가 다양할수록 커버리지가 높게 나타난다. 반대로 콘텐츠기반 접근방식에서 과도한 특수화나 협력필터링에서 데이터 희소성과 같은 문제점이 있는 경우에는 전체 아이템 중에서 추천이 이루어지는 아이템이 매우 한정적이므로 낮은 커버리지 값을 나타내게 된다(Good et al., 1999; Herlocker et al., 1999; Sarwar et al., 1998; Herlocker et al., 2004).

세렌디피티는 본래 예상치 못한 행동이나 결과로부터 새로운 발견을 한다는 뜻으로써 추천 시스템에서는 의외의 아이템을 추천해주었을 때 오히려 고객의 만족도를 높일 수 있음을 나타낸다. 즉, 선호도가 높을 것이라고 예측한 아이템은 고객에게 뻔하고 지겨울 수 있으므로 추천 목록 중, 일정 비율은 새로운 아이템을 포함하여 다양성을 유지하고 고객의 만족도를 높여야 한다(Sarwar et al., 2001).

Lathia et al.(2010)는 한 사용자가 한 시점에 추천 리스트  $L_1$  을 추천 받고, 다음 시점에  $L_2$ 를 추천 받았을 경우  $L_2$ 에서  $L_1$ 에 들어있지 않는 아이템의 개수( $L_2-L_1$ )를 추천 개수인  $N$ 개로 나눈 지표를 제안하였으며 식 (28)과 같다.

$$diversity(L_1, L_2, N) = \frac{|L_2 - L_1|}{N} \quad (28)$$

다양성 지표는 두 개의 추천 리스트의 차이만 표현할 수 있기 때문에 이를 확장한 특이성 지표가 제안되었다, 이는 한 시점  $t$ 에 추천된 아이템 셋  $A_t$ 에서 등장하지 않는 아이템의 비율을 의미한다.  $A_t$ 를 한 사용자의 기존 추천 내역으로 정의할 경우, 시간이 변함에 따라 얼마나 다양한 아이템을 추천하는지에 대

한 지표로 해석할 수 있으며 식 (29)을 통해 계산할 수 있다.

$$novelty(L_1, N) = \frac{|L_2 - A_t|}{N} \quad (29)$$

Fleder et al.(2009)는 경제적 불평등지수로 자주 사용되는 Gini 계수를 이용하여 추천 시스템의 다양성을 측정하였다. 추천 시스템의 각 아이템 판매량(혹은 선택 빈도, 유명도)을 로렌츠 곡선으로 표현한 뒤, Gini 계수를 계산하였는데 이를 통하여 실제 시스템에서의 아이템 소비가 불균형적으로 일어나는 경우 다양성이 적다고 판단하였다.

Zhou et al.(2010)는 다른 사용자와의 추천된 아이템을 비교하여 추천 시스템의 개인화된 정도를 간접적으로 추정하는 지표를 제안하였으며 식(30)과 같다.

$$h_{ij}(L) = 1 - \frac{q_{ij}(L)}{L} \quad (30)$$

여기서, 사용자  $i$ 와  $j$ 의 추천된 아이템 중에서 공통된 아이템의 비율  $\frac{q_{ij}(L)}{L}$ 을 통하여 두 사용자의 추천 리스트간의 이질성을 계산하였으며, 모든 사용자에게 대하여  $h_{ij}(L)$ 을 계산한 뒤, 평균을 취함으로써 각 사용자의 개인화 정도를 표현하였다.

Vargas et al.(2011)는 아이템의 특이성은 특정 문맥에서 계산되어야 한다고 주장하였다. 이는 단순히 인지도의 역수로써 특이성을 정의하는 것이 아니라, 특정 사용자 집단에서 한 아이템이 얼마나 선택되는지, 혹은 특정 시점이나 영화 장르와 같이 특정한 상황  $\theta$ 에서 각각 특이성을 정의해야 함을 의미한다. 이를 이용하여 한 사용자에게는 특정 아이템만을 추천하는 경향 등을 수치적으로 표현할 수 있으며 식 (31)과 같다.

$$nov(i|\theta) = 1 - p(seen|i, \theta) \quad (31)$$

이 때,  $p(seen|i, \theta)$ 는 문맥  $\theta$ 일 때 아이템  $i$ 가 추천되는 확률이다.

Murakami et al.(2008)는 세렌디피티를 측정하는 지표로서 unexpectedness를 제안하였으며 식 (32)을 통해 계산되어진다.

$$unexpectedness = \frac{1}{N} \sum_{i=1}^N \max(Pr(s_i) - Prim(s_i), 0) \quad (32)$$

$$\times isrel(s_i) \times \frac{count(i)}{i}$$

여기서,  $Prim(s_i)$ 는 기존 추천 모델의 추천 점수이고  $Pr(s_i)$ 는 테스트 하려는 추천 모델의 추천 점수이며,  $isrel(s_i)$ 는 사용자의 적절성을 나타낸다. 즉, 두 모델 차이에 사용자의 적절성을 결합하여 기존모델에 비해 사용자에게 유용하면서도 새로운 아이템들이 추천되는지를 설명할 수 있다. 또한, 추천 리스트의 순서에 의한 가중치인  $\frac{count(i)}{i}$ 를 추가하여 계산할

수 있다.

Ge *et al.*(2010)는 추천 시스템이 추천하는 아이템의 종류의 다양성과 한 사용자에게 반복적으로 추천이 이뤄질 때의 다양성, 그리고 의외성을 수리적으로 표현하기 위하여 예측 커버리지(prediction coverage)를 제안하였으며 식 (33)과 같다.

$$prediction\ coverage = \frac{|I_p|}{|I|} \quad (33)$$

여기서,  $|I|$ 는 전체 아이템 집합의 크기이며,  $|I_p|$ 는 시스템이 추천하는 아이템 집합의 크기이다.

## 8.5 기타 평가방법에 대한 논의

협력필터링에서 통계 모델을 기반으로 학습 알고리즘을 사용할 경우에는 학습 데이터의 양이 선호도 예측에 큰 영향을 끼치는데 이를 ‘학습률’이라고 한다. ‘학습률’은 크게 ‘전체 학습률’, ‘아이템 당 학습률’, ‘사용자 당 학습률’로 나눌 수 있으며 데이터 희소성 문제가 있을 경우에 예측결과를 신뢰할 수 있는가를 판단하기 위해 제안되었다(Herlocker, 2004).

또한, 최근에는 실시간으로 쏟아지는 방대한 양의 평가 데이터를 빠르게 피드백 하여 고객에게 새로운 추천 목록을 제공하는 것에 대한 연구가 활발해지고 있다(Fortuna *et al.*, 2010). 즉, 추천 시스템을 실제 문제에 보다 효율적으로 적용하기 위해서는 알고리즘이 얼마만큼 빨리 데이터를 학습하고 분석하여 고객의 선호도를 예측하는가에 대한 평가도 함께 이루어져야 한다.

이처럼 다양한 평가방법을 통해 추천 알고리즘의 성능을 판단할 수 있지만 궁극적인 목적은 고객의 만족도를 극대화시키기 위한 것이므로 때로는 고객에게 추천 만족도를 평가하도록 하여 성능을 판단하기도 한다(kim *et al.*, 2013). 이러한 방법은 다양하고 복잡한 요소들에 대한 고려 없이 평가가 가능하다는 장점이 있지만 주관적인 고객의 의견에만 의존해야 한다는 문제점을 가지고 있다.

## 9. 적용 사례

### 9.1 해외 적용 사례

#### (1) GroupLens Research

GroupLens는 1992년에 설립된 미국 미네소타 주립대학 컴퓨터 공학과 소속 연구실로, 1994년 이후 추천 시스템을 자동화하여 구현하였다(Resnick *et al.*, 1994). 추천 기법으로써 사용자-아이템 행렬을 통해 취향이 유사한 이웃 고객을 선정하고 이웃 고객의 구매이력을 바탕으로 선호도를 예측하는 방식을 제안하였으며, 이러한 기법을 협력필터링이라 지칭하게 되었다. 연구에 사용된 대표적인 데이터는 총 1,684편의 영화에 대해

서 943명의 고객들이 선호도를 평가한 MovieLens 데이터와 22,862개의 인터넷상 뉴스에 대해서 250명의 고객들이 선호도를 평가한 NetNews 데이터이다. GroupLens Research의 연구는 협력필터링을 이용한 추천 시스템의 성공 가능성을 보여준 최초의 시도이며 이후 협력필터링을 적용한 수많은 연구에 큰 기여를 하였다(Konstan *et al.*, 1997).

#### (2) Amazon.com

Amazon.com은 1994년 인터넷 상에서 책을 팔기 시작한 뒤, 대표적인 인터넷 기업으로 성장한 이후 CD, 약품 등 아이템을 추가하여 사업의 영역을 확장하고 있다. 상품 페이지를 보기만 해도 그와 관련된 다른 상품을 추천하고 있으며 전체 판매의 약 35%가 추천 시스템을 통해 이루어지고 있다. Amazon의 추천 시스템은 협력필터링 기법을 바탕으로 개발된 상품 간 협조 필터링이다. 이는 크게 2단계의 과정을 거치는데, 먼저 사용자가 구입하려고 하는 상품들 간의 유사성을 수치화하기 위해 아이템-아이템 행렬을 만든 후 아이템간의 상관관계를 계산한다. 그 다음 구입대상 상품과 관련 있는 모든 상품들의 유사점에 관한 데이터베이스를 참조하여 사용자가 구매한 상품간의 유사점을 찾고 이와 비슷한 다른 상품을 추천해 주는 방식이다(Linden *et al.*, 2003). 또한 2010년에는 페이스북과의 제휴를 맺어 페이스북에 공개된 고객의 개인정보 또는 네트워크 정보를 아마존의 구매정보와 다양한 형태로 결합하여 개인화된 추천 아이템 목록을 제공하고 있다.

#### (3) Netflix

DVD 대여와 온라인 영화감상을 주 사업으로 하는 Netflix는 2009년 Netflix Prize 경진대회를 개최하였다(Netflix Prize, 2009). 이는 자사가 그 당시 갖추고 있는 추천 시스템 보다 10% 이상 향상된 성능의 알고리즘을 제안할 시 100만 달러의 상금을 수여하는 획기적인 대회였다. 제안된 대표적인 알고리즘으로는 SVD, 제약적 볼츠만 머신, K-근접 이웃 등이었으며, 이 대회의 우승팀은 세 가지 알고리즘을 기반으로 하여 100여 개 이상의 예측 알고리즘을 학습한 후, 이들을 앙상블한 기법을 제안하였다. Netflix는 자사의 성공 요인을 신뢰성 있는 추천 시스템이라고 주장하며 추천만족도를 보다 향상시키기 노력하고 있다.

#### (4) 그 밖의 사례

1995년 넷스케이프 커뮤니케이터가 출시되면서 본격적인 인터넷 시대가 시작되었고, 이에 따라 인터넷 수요의 급증과 상업화가 시작되었다(Dean *et al.*, 1996). 인터넷 상업화 초기에는 앞서 설명한 음악CD 혹은 도서판매가 주를 이루었으나 오늘날에는 의류, 식품, 약품에 이르기까지 그 범위가 확장되고 있다.

CDNow는 1994년, 음악 CD를 온라인상에서 판매하는 사업을 시작한 이후 2001년 아마존에 인수되기까지 다양한 방식으

로 CD 추천 서비스를 제공하였다. ‘Album Advisor’와 ‘Related Artists’를 통해 고객이 선택한 아이템과 유사한 가수 혹은 장르를 추천하는데 이는 콘텐츠기반 접근방식을 사용한 방법이다. 특히 ‘My CDNow’는 고객이 음악을 구입하여 자신만의 음악 바구니 구성한 뒤, 음악에 대해 좋고 싫음을 평가할 수 있도록 함으로써 구매여부 뿐만 아니라 구매 후 선호도까지 고려하여 보다 정교한 예측을 가능케 했다. 또한 고객이 평가한 데이터를 실시간으로 업데이트하는 등, 지속적인 시스템을 향상을 위해 노력하였다. 그밖에도 ‘Top 100’ 기능을 통해 베스트 셀러 CD를 나열하는 등 다양하게 추천 시스템을 활용하였다(Schafer *et al.*, 1999).

건강식품, 영양제, 화장품 등 다양한 제품을 판매하는 미국의 Drugstore.com은 ‘Advisor’ 기능을 통해 사용자의 프로필과 선호도를 모두 고려한 하이브리드 추천 시스템을 적용하여 만족도 높은 추천 시스템을 제공하고 있다. 특히 협력필터링 기법에서 가장 큰 문제점 중 하나로 꼽히는 First rate 문제를 해결하기 위해 신제품의 샘플을 고객에게 배송하여 체험하게 한 후, 해당 아이템에 대한 후기를 남기도록 하였다. 데이터에만 의존하여 문제를 해결하려던 기존 방식에서 벗어나, 회사가 적극적으로 개입하여 고객의 후기를 유도하는 새로운 시도라고 할 수 있겠다.

개인경매 사이트에서부터 시작한 이베이(eBay)는 모든 종류의 신제품, 중고제품을 취급하며 세계적인 기업들을 인수한 글로벌 기업이다. 이베이는 중고거래의 특성을 고려하여 추천 시스템을 구성하였는데 이는 구매자가 물품을 구매한 뒤, 물품에 대한 만족도를 기입하게 되고 이는 판매자의 프로파일 정보로 사용되고 있다. 즉, 판매자는 구매자들에 의해 평가되고 이러한 데이터를 기반으로 구매자-판매자 행렬을 생성할 수 있다. 기존에는 고객에게 아이템을 추천해 주는 방식이 대부분이었던 반면, 이베이는 고객에게 판매자를 추천해 주는 방식을 적용하였다(Schafer *et al.*, 2001).

페이스북이나 트위터와 같은 소셜 네트워킹 서비스(SNS)에서는 고객에게 친구를 추천해 주기 위해 네트워크를 기반으로 하여 친구의 친구를 고객에게 추천하거나 고객이 직접 작성한 프로필 정보를 기반으로 공통의 관심사를 갖는 사람을 친구로 추천하게 된다. 트위터의 경우에는 먼저 ‘Circle of Trust’라 불리는 추천 후보를 선정 한 뒤, 그 안에서 다시 SALSA(Stochastic Approach for Link-Structure Analysis)라는 추천 알고리즘을 수행하여 고객에게 추천 친구 목록을 제공한다(Gupta *et al.*, 2013). SNS에서는 친구 추천 뿐만 아니라 네트워크 정보를 기존의 상품 구매이력 정보와 결합하여 상품을 추천해주는 기법에 대한 연구 또한 활발히 진행되고 있다(Kim, 2013a).

Montaner *et al.*(2003)은 앞서 설명한 GroupLens와 아마존을 비롯한 해외의 37개 추천 시스템에 대하여 참고문헌과 도메인을 나타내었다. 뿐만 아니라, 각 시스템에서 사용한 데이터 구성 방법과 추천 기법 등 8개의 항목에 대해 상세히 기술하여 추천 알고리즘에 대한 이해와 비교가 용이하도록 구성하였다.

## 9.2 국내 적용 사례

### (1) 알라딘

알라딘은 1999년 인터넷 서점으로 출발하여 2000년부터 국내 최초로 도서 추천 서비스를 시작하였으며 현재 알라딘에서 제공하고 있는 추천 시스템의 이름은 ‘추천 마법사’로써 크게 세 개의 서비스로 구성되어 있다. 먼저 ‘마법사의 선택’은 고객의 과거 구매이력과 고객이 직접 작성한 관심리스트를 바탕으로 관심지수가 높은 분야에서의 추천 도서와 신간도서를 함께 보여준다. 또한 ‘신간 알리미’를 통해 고객이 과거 구매한 작가의 신간을 알려주며 ‘서재이웃의 선택’은 관심 있는 서재를 등록한 뒤 취향이 비슷한 사람들이 선택한 도서를 보여주는 서비스로써 협력필터링을 기반으로 하고 있다. 알라딘은 더욱 개인화 된 추천 도서 목록을 제공하기 위해 ‘추천 마법사’의 이용환경을 직접 설정할 수 있도록 구성하였으며, 특히 관심 목록을 자유롭게 추가 및 삭제할 수 있도록 하여 구매이력에만 의존하여 분석하던 것보다 많은 양의 학습 데이터를 확보하여 선호도 예측의 정확도를 향상시켰다.

### (2) 왓차

2012년, 영화추천 어플 왓차를 개발한 프로그램스 사는 영화 사이트나 SNS 등의 데이터를 실시간으로 분석하여 각 고객에게 개인화 된 추천을 하고 있다. 고객이 최소 20개 이상의 영화에 대해 별점을 매긴 이후 협력필터링 기법을 바탕으로 예측한 추천 영화 목록을 제공하게 되며, 개인의 취향을 반영하기 위해 고객의 프로필 정보도 사용된다. 특히, 다양한 영화를 추천해주기 위해 유명한 영화, 인지도가 낮은 영화, 개인취향에 맞는 영화 등을 골고루 안내함으로써 추천목록은 편하고 지루하다는 인식으로부터 탈피하려 노력하였다. 특히, 영화를 추천해주는 것에 그치지 않고 어떤 이유로 고객에게 추천이 이루어졌는지에 대한 정보를 함께 제공함으로써 추천의 신뢰도를 향상시키고 고객들의 흥미를 유발하였다. 향후, 더욱 정교한 알고리즘을 구현하기 위해 모델기반의 협력필터링 분야의 다양한 기법들에 대한 연구가 진행 중이며, 기존의 미흡한 부분을 보완하여 영화뿐만 아니라 드라마, 도서 등으로 영역을 확장하기 위한 노력이 계속되고 있다.

### (3) 다음 쇼핑하우

다음 쇼핑하우는 2006년부터 포털 다음에서 제공하는 쇼핑 메타검색 서비스로써 여러 쇼핑물들의 상품 데이터를 가져와서 고객에게 맞는 상품을 찾고, 가격을 비교해주고 있다. 쇼핑하우는 협력필터링을 바탕으로 추천이 이루어지며, 상품 구매이력이 없는 Cold start 제품은 콘텐츠기반의 추천방식을 적용하여 추천을 하도록 하였다. 또한 상품은 같은 카테고리 내에서 추천하는 것을 기본으로 하지만 해당 카테고리의 데이터가 희박할 경우에는 상위 카테고리로 범위를 확장시켜 추천을 진행한다. 협력필터링 이외에도 다양한 속성을 바탕으로 연관성을 분석한 뒤, 연관성 점수가 높은 상품을 추천하는 기법도 활



용하고 있다. 그러나 현재 쇼핑하우는 완전한 개인화는 적용되어 있지 않아 카테고리 마다 개인화 정도가 다른데 이는 분석 데이터의 양이 다를 뿐만 아니라 개인 정보의 보안문제를 갖고 있기 때문이다.

## 10. 추천 시스템 소프트웨어

### 10.1 추천 시스템 소프트웨어의 종류

추천 시스템의 발전과 더불어 Java, C++을 비롯해 다양한 언어로 추천 시스템 알고리즘을 구현한 소프트웨어들이 개발되고 있다. <Table 5>는 알고리즘의 코드를 제공하는 추천 시스템 소프트웨어의 이름과 구현에 사용된 언어, 다운로드 URL을 나타내고 있다. 데이터 특성과 구현하고자 하는 언어에 따라 다양한 서비스를 제공받을 수 있어 추천 시스템을 직접 구현하는데 매우 유용하다(Hahsler *et al.*, 2011).

Table 5. Recommender system software

소프트웨어	언어	URL
Recommenderlab	R	<a href="http://cran.r-project.org/web/packages/recommenderlab/index.html">http://cran.r-project.org/web/packages/recommenderlab/index.html</a>
MyMediaLite	C#, Mono	<a href="http://mymedialite.net/documentation/index.html">http://mymedialite.net/documentation/index.html</a>
LensKit	Java	<a href="http://lenskit.grouplens.org/">http://lenskit.grouplens.org/</a>
Duine	Java	<a href="http://www.duineframework.org/">http://www.duineframework.org/</a>
Crab	Python	<a href="http://muricoca.github.io/crab/">http://muricoca.github.io/crab/</a>
Waffles	C++	<a href="http://waffles.sourceforge.net/">http://waffles.sourceforge.net/</a>
ApacheMahout	Java	<a href="http://mahout.apache.org/">http://mahout.apache.org/</a>
easyrec	Java	<a href="http://www.easyrec.org/">http://www.easyrec.org/</a>

### 10.2 ‘Recommenderlab’ 구현

본 논문에서는 앞서 소개한 다양한 추천 시스템 소프트웨어 중에서, ‘R’에서 제공하는 ‘Recommenderlab’ 패키지를 통해 다양한 추천 기법을 구현한 후, 각각의 성능을 비교하였다. ‘Recommenderlab’은 분류와 예측 문제에 모두 적용이 가능하며 본 논문에서는 평가점수를 예측하는 문제를 다루었다.

#### (1) 데이터

실험에 사용된 데이터는 Goldberg *et al.*(2001)가 협력필터링의 성능을 평가하기 위해 사용한 ‘Jester’ 데이터로 1999년부터 2003년까지 ‘Jester Online Joke Recommender System’에서 수집되었다. <Figure 7>은 ‘Jester Online Joke Recommender System’ 화면으로써, 사용자가 추천받은 유머를 읽은 후 흥미로운 정도를 평가하게 된다. 시스템은 평가 정보를 바탕으로 사용자의 선호도를 예측한 뒤, 흥미를 가질 만한 유머를 그 다음 화면에서 보여주는 과정을 반복한다. 본 실험에서 사용한 데이

터는 ‘Jester Online Joke Recommender System’ 사용자 중에서 500명의 사용자가 100개의 유머에 대해 평가한 데이터이다. 총 362,106개의 평가 값이 기록되어있으며 평가 값의 범위는 -10부터 10까지이다. 실제로 ‘Jester Online Joke Recommender System’은 이러한 평가 데이터를 사용한 협력필터링 추천 방식을 통해 사용자의 선호도를 파악하고 그에 맞는 유머를 추천해주고 있다(Hahsler *et al.*, 2011).

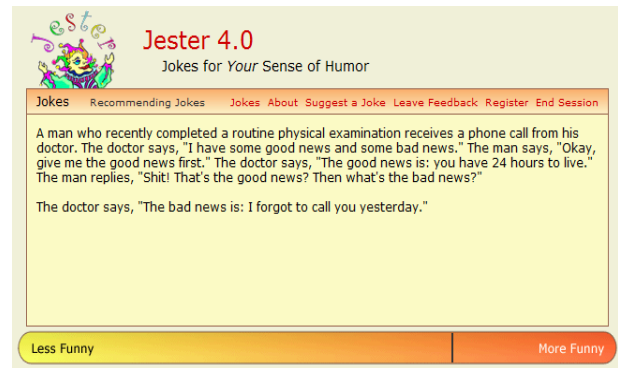


Figure 7. Jester online joke recommender system

#### (2) 예측 및 평가

```
R > data(Jester)

## 500 명의 사용자를 90 : 10 의 비율로 학습 데이터 셋과 테스트
## 데이터 셋으로 나눔
R > e <- evaluationScheme(Jester5k[1 : 500], method = "split",
train = 0.9, k = 1, given = 15, goodRating = 5)

## 학습 데이터 셋을 이용하여 사용자 기반 협력필터링 모델 생성
R > r <- Recommender(getData(e, "train"), "UBCF")

## 테스트 데이터 셋의 평가 값을 이용하여 예측
R > p <- predict(r, getData(e, "known"), type = "ratings")

## 평가 결과 확인
R > calcPredictionError(p, getData(e, "unknown"))
```

Figure 8. The source code of ‘Recommenderlab’

<Figure 8>은 ‘Recommenderlab’을 통해 사용자의 선호도를 예측하기 위한 절차를 ‘R’ 코드와 함께 설명한 그림이다. 먼저 ‘Jester’ 데이터를 불러온 후, 학습데이터 셋과 평가 데이터 셋을 90:10의 비율로 나눈 다음 학습데이터 셋을 통해 선호도 예측 모델을 생성하고 평가데이터 셋을 통해 성능을 평가하였다. <Figure 8>에서는 선호도 예측 모델을 생성 시 사용자 기반 협력필터링(UBCF)을 선택하였으며, 50명의 사용자에게 대해 선호도를 예측한 결과 MAE는 3.77, MSE는 22.05, RMSE는 4.69로 나타났다. ‘Recommenderlab’의 ‘Recommender’ 명령어에서는 모델 선택 옵션을 통해 사용자 기반 협력필터링(UBCF) 뿐만 아니라, 아이템 기반 협력필터링(IBCF), 주성분 분석(PCA), 무

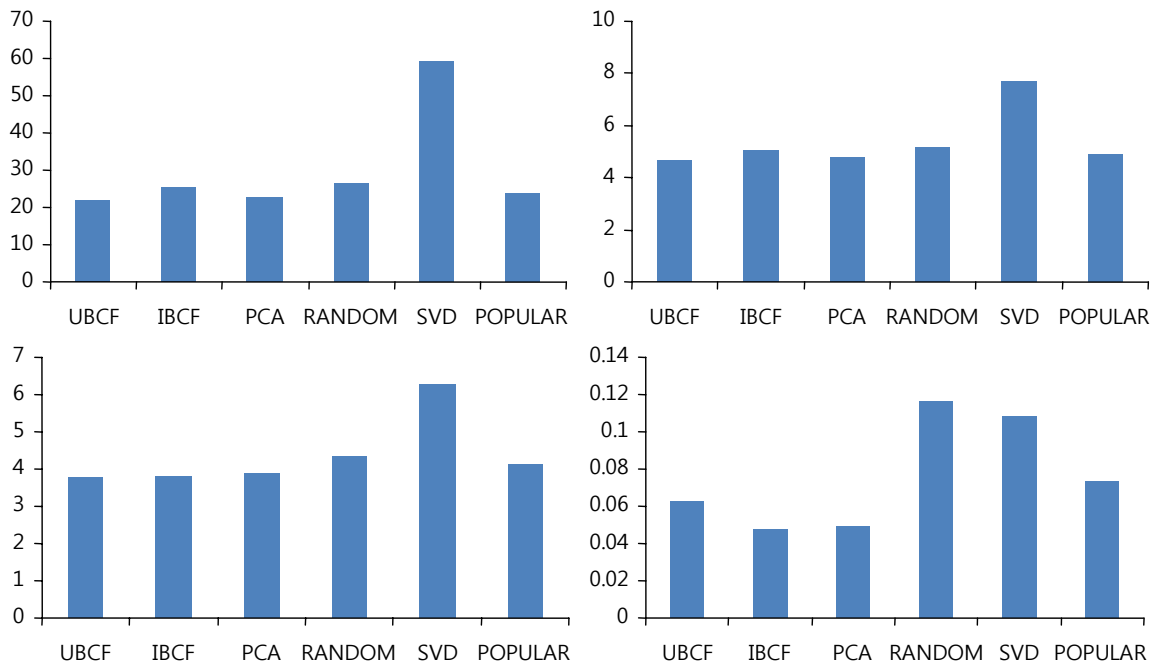


Figure 9. The performance Evaluations of recommender systems on 'Jester'

작위 선택(RANDOM), Singular Value Decomposition(SVD), 상위  $N$ 개 아이템(POPULAR) 등의 다양한 기법을 선택할 수 있다. 각 기법을 'Jester' 데이터에 적용한 뒤, 예측 성능을 비교한 결과는 <Figure 9>와 같다.

사용자 기반 협력필터링(UBCF)은 MSE, RMSE, MAE 평가 척도에 대해서 다른 기법보다 우수한 성능을 나타냄을 확인하였으나, MAE 값을 정규화 시킨 NMAE 평가척도에 대해서는 아이템 기반 협력필터링(IBCF)을 이용한 알고리즘의 성능이 가장 우수하게 나타났다. 이는 NMAE가 예측 오차 값을 정규화시킨 결과이므로 아이템 기반 협력필터링(IBCF)를 통해 예측을 하는 것이 각기 다른 스케일의 점수에 대해서도 예측 성능이 우수하다는 것을 나타낸다. 반면, 무작위 선택(RANDOM)을 통해 예측을 할 경우, MSE, RMSE, MAE 평가척도에 대해서 다른 기법과 성능의 차이가 크지 않지만 NMAE 평가척도에 대해서는 6개의 추천 기법 중에서 가장 좋지 않은 추천 성능을 나타내고 있다. 즉, 다양한 스케일을 고려하지 못한 채, 일괄적인 점수를 예측한다면 오차 측면에서는 우수하더라도 개인화된 추천 만족도에 있어 좋은 성능을 보장할 수 없다는 결론을 얻을 수 있다. 연구자는 예측오차가 큰 관측치에 대해 어느 정도의 가중치를 부여할 것인지, 혹은 데이터 특성에 맞는 평가 척도는 무엇인지 등에 대한 고민을 통해 적절한 평가척도를 선택해야 한다.

## 11. 결 론

인터넷을 이용한 상품 구매가 일상화되면서 고객이 원하는 상

품을 빠르고 정확하게 파악하여 추천해 주는 것은 고객의 만족도를 향상시키고 기업의 판매 이익을 극대화시킬 수 있다. 이에 따라 상품 추천의 중요성이 대두되고 있으며, 적절한 상품을 추천해 줄 수 있는 다양한 기법에 대한 연구가 활발히 진행 중이다.

본 논문에서는 다양한 추천 시스템의 특징 및 장·단점에 대해 살펴보고 실제 적용사례를 알아봄으로써, 추천 시스템 기법의 동향을 파악하였다. 주로 사용되는 추천 시스템 방식은 콘텐츠기반 접근방식과 협력필터링, 이들을 결합한 하이브리드 방식, 그리고 연관성 분석 등이 있다. 콘텐츠기반 접근방식은 사용자가 직접 입력한 프로파일 정보 및 과거 구매이력을 바탕으로 선호하는 아이템을 파악한 뒤, 해당 아이템이 속한 카테고리를 추천해주는 방식으로 다른 사용자의 정보가 없어도 추천이 가능하다는 장점이 있지만 추천 상품의 다양성을 보장할 수 없는 문제점을 가지고 있다. 협력필터링은 추천 대상 고객과 구매패턴이 유사한 사용자를 선정하여 그들이 선택한 아이템을 추천해주는 방식으로 개인화 된 추천이 가능하다는 장점을 지닌 반면 사용자들의 구매이력 데이터가 충분치 못할 경우 추천이 불가능하다는 한계점을 가지고 있다. 따라서 콘텐츠기반 접근방식과 협력필터링을 결합하여 각각의 단점은 보완하고 장점은 극대화 시킨 하이브리드 방식에 대한 연구가 활발히 이루어지고 있으며 데이터 특성에 맞춰 다양한 방식으로 결합된 기법은 우수한 성능과 함께 다양한 분야에서 사용되고 있다.

수많은 기법들이 제안됨에 따라 알고리즘의 성능을 평가할 수 있는 평가척도에 대한 연구도 중요시되고 있다. 사용자가 아이템에 대해 선호 정도를 평가하는 예측문제에 있어서는 예

측 점수와 실제 점수의 차이를 통해 정확도를 평가하며 점수 예측이 아닌 ‘구매-비 구매’와 같은 분류문제에 있어서는 Accuracy, Precision, Recall, F-measure 등을 통해 정확도를 평가하고 있고 정확도를 평가하는 방법 이외에도 추천목록의 다양성 및 의의성을 평가하기 위한 방법들에 대한 연구가 진행 중이다.

또한, 본 논문에서는 실제로 추천 시스템 알고리즘을 적용한 성공적 사례들을 살펴봄으로써, 추천 알고리즘의 중요성과 효용성을 파악하였다. 인터넷 서점인 Amazon.com, CD 상점인 CDNow.com, 인터넷 영화 추천 사이트인 MovieFinder 부터 화장품 및 건강식품을 판매하는 Drugstore.com와 유머를 추천해주는 Jester Online Joke Recommender System 등에 이르기까지 다양한 분야에서 추천 시스템이 성공적으로 활용되고 있다. 인터넷 상에서 판매되는 상품의 종류가 다양해질수록 각 아이템의 특성을 반영한 추천 시스템에 대한 필요성이 대두되고 있으며, 아이템의 속성정보나 구매이력 이외의 다양한 정보를 추천에 활용하여 추천의 정확성을 향상시키고자 하는 노력이 계속되고 있다.

이와 같은 추천 시스템의 발달과 함께 기업 및 교육기관에서는 다양한 프로젝트들이 수행되어지고 있고 이를 소프트웨어로 공개함으로써 누구나 쉽게 추천 시스템을 구현할 수 있도록 하고 있다. 본 논문에서는 이러한 소프트웨어 중에서 ‘R’에서 지원하는 ‘Recommenderlab’을 활용하여 다양한 추천 기법을 구현하였으며 코드와 함께 그 절차를 설명하였다.

최근 들어, 추천 시스템은 데이터의 양적인 증가, 상품 종류의 다양화, 비정형 데이터에 대한 전처리 등에 대한 문제를 해결하기 위해 다양한 시도가 이루어지고 있으나, 실제 데이터에 대한 적용사례가 적거나 혹은 특정 사례에만 적용 가능한 기법을 제안하는 연구가 대부분이므로 이에 대한 한계를 극복하는 것이 시급한 실정이다. 향후, 이러한 문제점을 해결하고, 다양한 특성을 지닌 데이터에 적용할 수 있는 보다 일반적인 추천 시스템 기법에 대한 연구가 많이 행해져야 할 것이다.

## 참고문헌

- Adomavicius, G., Sankaranarayanan, R., Sen, S., and Tuzhilin, A. (2005), Incorporating contextual information in recommender systems using a multidimensional approach, *ACM Transactions on Information Systems (TOIS)*, **23**(1), 103-145.
- Adomavicius, G. and Tuzhilin, A. (2005), Toward the next generation of recommender systems : A survey of the state-of-the-art and possible extensions, *Knowledge and Data Engineering, IEEE Transactions on*, **17**(6), 734-749.
- Adomavicius, G. and Kwon, Y. (2012), Improving aggregate recommendation diversity using ranking-based techniques, *Knowledge and Data Engineering, IEEE Transactions on*, **24**(5), 896-911.
- Ahsae, M. G., Naghibzadeh, M., and Naeini, S. E. Y. (2014), Semantic similarity assessment of words using weighted WordNet, *International Journal of Machine Learning and Cybernetics*, **5**(3), 479-490.
- Al Mamunur Rashid, S. K. L., Karypis, G., and Riedl, J. (2006), Clust KNN : a highly scalable hybrid model-and memory-based CF algorithm, *Proceeding of WebKDD*.
- Balabanović, M. and Shoham, Y. (1997), Fab : content-based, collaborative recommendation, *Communications of the ACM*, **40**(3), 66-72.
- Basilico, J. and Hofmann, T. (2004), Unifying collaborative and content-based filtering, In *Proceedings of the twenty-first international conference on Machine learning*, 9.
- Baluja, S., Seth, R., Sivakumar, D., Jing, Y., Yagnik, J., Kumar, S., and Aly, M. (2008), Video suggestion and discovery for youtube : taking random walks through the view graph, In *Proceedings of the 17th international conference on World Wide Web*, 895-904.
- Barragáns-Martínez, A. B., Costa-Montenegro, E., Burguillo, J. C., Rey-López, M., Mikić-Fonte, F. A., and Peleteiro, A. (2010), A hybrid content-based and item-based collaborative filtering approach to recommend TV programs enhanced with singular value decomposition, *Information Sciences*, **180**(22), 4290-4311.
- Berry, M. W., Dumais, S. T., and O'Brien, G. W. (1995), Using linear algebra for intelligent information retrieval, *SIAM review*, **37**(4), 573-595.
- Billsus, D. and Pazzani, M. J. (1998), Learning Collaborative Information Filters, In *ICML*, **98**, 46-54.
- Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003), Latent dirichlet allocation, *the Journal of machine Learning research*, **3**, 993-1022.
- Bobadilla, J., Ortega, F., Hernando, A., and Gutiérrez, A. (2013), Recommender systems survey, *Knowledge-Based Systems*, **46**, 109-132.
- Breese, J. S., Heckerman, D., and Kadie, C. (1998), Empirical analysis of predictive algorithms for collaborative filtering, In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, 43-52.
- Burke, R. (2002), Hybrid recommender systems : Survey and experiments, *User modeling and user-adapted interaction*, **12**(4), 331-370.
- Burke, R. (2007), Hybrid web recommender systems, In *The adaptive web*, 377-408.
- Carrillo, D., López, V. F., and Moreno, M. N. (2013), Multi-label Classification for Recommender Systems, In *Trends in Practical Applications of Agents and Multi agent Systems*, 181-188.
- Celma, Ò. and Herrera, P. (2008), A new approach to evaluating novel recommendations, In *Proceedings of the 2008 ACM conference on Recommender systems*, 179-186.
- Chow, A. and Manai, G. (2014), *HybridRank : A Hybrid Content-Based Approach To Mobile Game Recommendations*, CBRRecSys.
- Claypool, M., Gokhale, A., Miranda, T., Murnikov, P., Netes, D., and Sartin, M. (1999), Combining content-based and collaborative filters in an online newspaper, In *Proceedings of ACM SIGIR workshop on recommender systems*, 60.
- Cremonesi, P., Koren, Y., and Turrin, R. (2010), Performance of recommender algorithms on top-n recommendation tasks, In *Proceedings of the fourth ACM conference on Recommender systems*, 39-46.
- Das, M., De Francisci Morales, G., Gionis, A., and Weber, I. (2013), Learning to question : Leveraging user preferences for shopping advice, In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 203-211.
- Dean, D., Felten, E. W., and Wallach, D. S. (1996), Java security : From HotJava to Netscape and beyond, In *Security and Privacy, Proceedings, IEEE Symposium*, 190-200.
- Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., and Harshman, R. A. (1990), Indexing by latent semantic analysis, *JASIS*, **41**(6), 391-407.
- Deshpande, M. and Karypis, G. (2004), Item-based top-n recommen-

- dation algorithms, *ACM Transactions on Information Systems (TOIS)*, **22**(1), 143-177.
- Domingos, P. and Pazzani, M. (1997), On the optimality of the simple Bayesian classifier under zero-one loss, *Machine learning*, **29**(2/3), 103-130.
- Ducheneaut, N., Partridge, K., Huang, Q., Price, B., Roberts, M., Chi, E. H., and Begole, B. (2009), Collaborative filtering is not enough? Experiments with a mixed-model recommender for leisure activities, In *User Modeling, Adaptation, and Personalization*, 295-306.
- Fan, J. and Pan, W. (2014), An Improved Collaborative Filtering Algorithm Combining Content-based Algorithm and User Activity, *Big Data and Smart Computing (BIGCOMP)*, 88-91.
- Fleder, D. and Hosanagar, K. (2009), Blockbuster culture's next rise or fall : The impact of recommender systems on sales diversity, *Management science*, **55**(5), 697-712.
- Fortuna, B., Fortuna, C., and Mladenović, D. (2010), Real-time news recommender system, In *Machine Learning and Knowledge Discovery In Databases*, 583-586.
- Frakes, W. B. and Baeza-Yates, R. (1992), *Information retrieval : data structures and algorithms*.
- Ganu, G., Elhadad, N., and Marian, A. (2009), Beyond the Stars : Improving Rating Predictions using Review Text Content, In *WebDB*.
- Ge, M., Delgado-Battenfeld, C., and Jannach, D. (2010), Beyond accuracy : evaluating recommender systems by coverage and serendipity, In *Proceedings of the fourth ACM conference on Recommender systems*, 257-260.
- Ghazanfar, M. and Prugel-Bennett, A. (2010), An Improved Switching Hybrid Recommender System Using Naive Bayes Classifier and Collaborative Filtering.
- Goldberg, D., Nichols, D., Oki, B. M., and Terry, D. (1992), Using collaborative filtering to weave an information tapestry, *Communications of the ACM*, **35**(12), 61-70.
- Goldberg, K., Roeder, T., Gupta, D., and Perkins, C. (2001), Eigentaste : A constant time collaborative filtering algorithm, *Information Retrieval*, **4**(2), 133-151.
- Good, N., Schafer, J. B., Konstan, J. A., Borchers, A., Sarwar, B., Herlocker, J., and Riedl, J. (1999), Combining collaborative filtering with personal agents for better recommendations, In *AAAI/IAAI*, 439-446.
- Gupta, P., Goel, A., Lin, J., Sharma, A., Wang, D., and Zadeh, R. (2013), Wtf : The who to follow service at twitter, In *Proceedings of the 22nd international conference on World Wide Web*, 505-514.
- Han, J., Kamber, M., and Pei, J. (2006), *Datamining : concepts and techniques*.
- Hahsler, M. (2011), *Recommenderlab : A Framework for Developing and Testing Recommendation Algorithms*.
- Herlocker, J. L., Konstan, J. A., Borchers, A., and Riedl, J. (1999), An algorithmic framework for performing collaborative filtering, In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 230-237.
- Herlocker, J. L., Konstan, J. A., and Riedl, J. (2000), Explaining collaborative filtering recommendations, In *Proceedings of the ACM conference on Computer supported cooperative work*, 241-250.
- Herlocker, J. L., Konstan, J. A., Terveen, L. G., and Riedl, J. T. (2004), Evaluating collaborative filtering recommender systems, *ACM Transactions on Information Systems (TOIS)*, **22**(1), 5-53.
- Hofmann, T. (2004), Latent semantic models for collaborative filtering, *ACM Transactions on Information Systems (TOIS)*, **22**(1), 89-115.
- Jeh, G. and Widom, J. (2002), SimRank : a measure of structural-context similarity, In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 538-543.
- Jeong, J. and Rhee, P. K. (2003), A Collaborative Filtering using SVD on Low-Dimensional Space, *Korea Information Processing Society*, **10B**(3), 273-280.
- Jiang, J. J. and Conrath, D. W. (1997), *Semantic similarity based on corpus statistics and lexical taxonomy*, arXiv preprint cmp-lg/9709008.
- Jim, B. W., Cho, Y. S., and Ryu, K. H. (2010), Personalized e-Commerce Recommendation System using RFM method and Association Rules, *Korea society of computer information*, **15**(12), 227-235.
- Karypis, G. (2001), Evaluation of item-based top-n recommendation algorithms, In *Proceedings of the tenth international conference on Information and knowledge management*, 247-254.
- Kim, B. M., Li, Q., Kim, S. G., Lim, E. K., and Kim, J. Y. (2003), A New Approach Combining Content-based Filtering and Collaborative Filtering for Recommender Systems, *Journal of Korean Institute of Intelligent Systems*, **31**(3), 332-342.
- Kim, J. D. (2013), Development of Social Commerce and Recommendations, *Korea Multimedia Society*, **17**(1), 28-36.
- Kim, N. K. and Yong, S. Y. (2013), Bayesian network based Music Recommendation System considering Multi-Criteria Decision Making, *The Journal of digital policy and management*, **11**(3), 345-352.
- Kim, Y. (2010), A Study on Design and Implementation of Personalized Information Recommendation System based on Apriori Algorithm, *Korea Biblia Society for Library and Information Science*, **23**(4), 283-308.
- Ko, H. G., Kim, E., Ko, I. Y., and Chang, D. (2014), Semantically-based Recommendation by using Semantic Clusters of Users' Viewing History, *Big Data and Smart Computing (BIGCOMP)*, 83-87.
- Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., and Riedl, J. (1997), GroupLens : applying collaborative filtering to Usenet news, *Communications of the ACM*, **40**(3), 77-87.
- Koren, Y. (2010), Collaborative filtering with temporal dynamics, *Communications of the ACM*, **53**(4), 89-97.
- Lang, K. (1995), Newsweeper : Learning to filter netnews, In *Proceedings of the Twelfth International Conference on Machine Learning*.
- Lathia, N., Hailes, S., and Capra, L. (2009), Temporal collaborative filtering with adaptive neighbourhoods, In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, 796-797.
- Lathia, N., Hailes, S., Capra, L., and Amatriain, X. (2010), Temporal diversity in recommender systems, In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, 210-217.
- Lee, J., Bengio, S., Kim, S., Lebanon, G., and Singer, Y. (2014), Local collaborative ranking, In *Proceedings of the 23rd international conference on World wide web*, 85-96.
- Lee, S. G., Lee, B. S., Bak, B. Y., and Hwang, H. K. (2010), A Study of Intelligent Recommendation System based on Naive Bayes Text Classification and Collaborative Filtering, *Journal of Information Management*, **41**(4), 227-249.
- Lee, S. I. and Lee, S. Y. (2010), A Collaborative Filtering-based Recommendation System with Relative Classification and Estimation Revision based on Time, *Journal of Korean Institute of Intelligent Systems*, **20**(2), 189-194.
- Lee, Y. S. and Lee, S. (2002), Cluster Feature Selection using Entropy Weighting and SVD, *Journal of The Korean Institute of Information Scientists and Engineers*, **29**(4), 248-257.
- Linden, G., Smith, B., and York, J. (2003), Amazon.com recommendation



- tions : Item-to-item collaborative filtering, *Internet Computing, IEEE*, 7(1), 76-80.
- Ling, G., Lyu, M. R., and King, I. (2014), Ratings meet reviews, a combined approach to recommend, In *Proceedings of the 8th ACM Conference on Recommender systems*, 105-112.
- Lops, P., De Gemmis, M., and Semeraro, G. (2011), Content-based recommender systems : State of the art and trends, In *Recommender systems handbook*, 73-105.
- Lü, L., Medo, M., Yeung, C. H., Zhang, Y. C., Zhang, Z. K., and Zhou, T. (2012), Recommender systems, *Physics Reports*, 519(1), 1-49.
- McAuley, J. and Leskovec, J. (2013), Hidden factors and hidden topics : understanding rating dimensions with review text, In *Proceedings of the 7th ACM conference on Recommender systems*, 165-172.
- McNee, S. M., Riedl, J., and Konstan, J. A. (2006), Being accurate is not enough : how accuracy metrics have hurt recommender systems, In *CHI extended abstracts on Human factors in computing systems*, 1097-1101.
- McNee, S. M., Kapoor, N., and Konstan, J. A. (2006), Don't look stupid : avoiding pitfalls when recommending research papers, In *Proceedings of the 20th anniversary conference on Computer supported cooperative work*, 171-180.
- Melville, P., Mooney, R. J., and Nagarajan, R. (2002), Content-boosted collaborative filtering for improved recommendations, In *AAAI/IAAI*, 187-192.
- Miller, G. A. (1995), WordNet : a lexical database for English, *Communications of the ACM*, 38(11), 39-41.
- Miyahara, K. and Pazzani, M. J. (2000), Collaborative filtering with the simple Bayesian classifier, In *PRICAI 2000 Topics in Artificial Intelligence*, 679-689.
- Murakami, T., Mori, K., and Orihara, R. (2008), Metrics for evaluating the serendipity of recommendation lists, In *New frontiers in artificial intelligence*, 40-46.
- Netflix Prize, <http://netflixprize.com>.
- Nikovski, D. and Kulev, V. (2006), Induction of compact decision trees for personalized recommendation, In *Proceedings of the ACM symposium on Applied computing*, 575-581.
- Noh, Y., Oh, Y., and Park, S. (2014), A Location-based Personalized News Recommendation, *Big Data and Smart Computing (BIGCOMP)*, 99-104.
- O'Connor, M. and Herlocker, J. (1999), Clustering items for collaborative filtering, In *Proceedings of the ACM SIGIR workshop on recommender systems*, 128.
- Oh, J. Y. (2004), *Design of Recommendation system Using Association Rule in e-Commerce*, Myongji University.
- Park, K. S. and Moon, N. M. (2012), Multidimensional Optimization Model of Music Recommender Systems, *information processing society journal*, 19B(3), 155-164.
- Park, Y. J. and Tuzhilin, A. (2008), The long tail of recommender systems and how to leverage it, In *Proceedings of the ACM conference on Recommender systems*, 11-18.
- Park, Y., Park, S., and Lee, S. (2014), Fast Collaborative Filtering with a k-Nearest Neighbor Graph, *KDD'13*.
- Pazzani, M. J. and Billsus, D. (2007), Content-based recommendation systems, In *The adaptive web*, 325-341.
- Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. (1994), GroupLens : an open architecture for collaborative filtering of news, In *Proceedings of the ACM conference on Computer supported cooperative work*, 175-186.
- Salton, G. and McGill, M. J. (1983), *Introduction to modern information retrieval*.
- Salton, G. and Buckley, C. (1988), Term-weighting approaches in automatic text retrieval, *Information processing and management*, 24(5), 513-523.
- Sarwar, B. M., Konstan, J. A., Borchers, A., Herlocker, J., Miller, B., and Riedl, J. (1998), Using filtering agents to improve prediction quality in the grouplens research collaborative filtering system, In *Proceedings of the 1998 ACM conference on Computer supported cooperative work*, 345-354.
- Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2000), Application of dimensionality reduction in recommender system-a case study (No. TR-00-043), Minnesota Univ Minneapolis Dept of Computer Science.
- Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2000), Analysis of recommendation algorithms for e-commerce, In *Proceedings of the 2nd ACM conference on Electronic commerce*, 158-167.
- Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2001), Item-based collaborative filtering recommendation algorithms, In *Proceedings of the 10th international conference on World Wide Web*, 285-295.
- Sarwar, B. M., Karypis, G., Konstan, J., and Riedl, J. (2002), Recommender systems for large-scale e-commerce : Scalable neighborhood formation using clustering, In *Proceedings of the fifth international conference on computer and information technology*, 1.
- Sawant, S. (2013), Collaborative Filtering using Weighted BiPartite Graph Projection, *A Recommendation System for Yelp*.
- Schafer, J. B., Konstan, J., and Riedl, J. (1999), Recommender systems in e-commerce, In *Proceedings of the 1st ACM conference on Electronic commerce*, 158-166.
- Schafer, J. B., Konstan, J. A., and Riedl, J. (2001), E-commerce recommendation applications, In *Applications of Data Mining to Electronic Commerce*, 115-153.
- Schein, A. I., Popescul, A., Ungar, L. H., and Pennock, D. M. (2002), Methods and metrics for cold-start recommendations, In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, 253-260.
- Sheth, B. and Maes, P. (1993), Evolving agents for personalized information filtering, In *Artificial Intelligence for Applications*, 345-352.
- Shin, C. H., Lee, J. W., Yang, H. N., and Choi, I. Y. (2012), The Research on Recommender for New Customers Using Collaborative Filtering and Social Network Analysis, *Korea Intelligent Information System Society*, 18(4), 19-42.
- Su, X. and Khoshgoftaar, T. M. (2006), Collaborative filtering for multi-class data using belief nets algorithms, In *Tools with Artificial Intelligence*, 497-504.
- Su, X. and Khoshgoftaar, T. M. (2009), A survey of collaborative filtering techniques, *Advances in artificial intelligence*, 4.
- Tata, S. and Patel, J. M. (2007), Estimating the selectivity of tf-idf based cosine similarity predicates, *ACM SIGMOD Record*, 36(2), 7-12.
- Tipping, M. E. and Bishop, C. M. (1999), Probabilistic principal component analysis, *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 61(3), 611-622.
- Vargas, S. and Castells, P. (2011), Rank and relevance in novelty and diversity metrics for recommender systems, In *Proceedings of the fifth ACM conference on Recommender systems*, 109-116.
- Vozalis, M. and Margaritis, K. G. (2004), Collaborative filtering enhanced by demographic correlation, In *AIAl Symposium on Professional Practice in AI, of the 18th World Computer Congress*.
- Wartena, C., Slakhorst, W., Wibbels, M., Gantner, Z., Freudenthaler, C., Newell, C., and BBC R&D, L. (2011), Keyword-Based TV Program

- Recommendation, In *Work shop chairs*, **15**.
- Wu, Y. H. and Chen, A. L. (2000), Index structures of user profiles for efficient web page filtering services, In *2012 IEEE 32nd International Conference on Distributed Computing Systems*, 644-644.
- Xiang, L., Yuan, Q., Zhao, S., Chen, L., Zhang, X., Yang, Q., and Sun, J. (2010), Temporal recommendation on graphs via long-and short-term preference fusion. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, 723-732.
- Yin, H., Sun, Y., Cui, B., Hu, Z., and Chen, L. (2013), LCARS : A Location-Content-Aware Recommender System, *KDD*.
- Yu, K., Schwaighofer, A., Tresp, V., Xu, X., and Kriegel, H. P. (2004), Probabilistic memory-based collaborative filtering, *Knowledge and Data Engineering, IEEE Transactions*, **16**(1), 56-69.
- Zhang, M. and Hurley, N. (2008), Avoiding monotony: improving the diversity of recommendation lists, In *Proceedings of the ACM conference on Recommender systems*, 123-130.
- Ziegler, C. N., McNee, S. M., Konstan, J. A., and Lausen, G. (2005), Improving recommendation lists through topic diversification, In *Proceedings of the 14th international conference on World Wide Web*, 22-32.
- Zhou, T., Kuscsik, Z., Liu, J. G., Medo, M., Wakeling, J. R., and Zhang, Y. C. (2010), Solving the apparent diversity-accuracy dilemma of recommender systems, *Proceedings of the National Academy of Sciences*, **107**(10), 4511-4515.