ISSN 2288-4866 (Print)
ISSN 2288-4882 (Online)
http://www.jiisonline.org
J Intell Inform Syst 2015 March: 21(1): 119~142 http://dx.doi.org/10.13088/jiis.2015.21.1.119

추천 시스템의 성능 안정성을 위한 예측적 군집화 기반 협업 필터링 기법

이오준
중앙대학교 컴퓨터공학과
(concerto9203@gmail.com)

유은순
단국대학교 미디어콘텐츠연구원
(tesniere@naver.com)

사용자의 취향과 선호도를 고려하여 정보를 제공하는 추천 시스템의 중요성이 높아졌다. 이를 위해 다양한 기법들이 제안되었는데, 비교적 도메인의 제약이 적은 협업 필터링이 널리 사용되고 있다. 협업 필터링의 한 종류인 모델 기반 협업 필터링은 기계학습이나 데이터 마이닝 모델을 협업 필터링에 접목한 방법이다. 이는 희박성 문제와 확장성 문제 등의 협업 필터링의 근본적인 한계를 개선하지만, 모델 생성 비용이 높고 성능/확장성 트레이드오프가 발생한다는 한계점을 갖는다. 성능/확장성 트레이드오프는 희박성 문제의 일종인 적용범위 감소 문제를 발생시킨다. 또한, 높은 모델 생성 비용은 도메인 환경 변화의 누적으로 인한 성능 불안정의 원인이 된다. 본 연구에서는 이 문제를 해결하기 위해, 군집화 기반 협업 필터링에 마르코프 전이확률모델과 퍼지 군집화의 개념을 접목하여, 적용범위 감소 문제와 성능 불안정성 문제를 해결한 예측적 군집화 기반 협업 필터링 기법을 제안한다. 이 기법은 첫째, 사용자 기호(Preference)의 변화를 추적하여 정적인 모델과 동적인 사용자간의 괴리 해소를 통해 성능 불안정 문제를 개선한다. 둘째, 전이확률과 군집 소속 확률에 기반한 적용범위 확장으로 적용범위 감소 문제를 개선한다. 제안하는 기법의 검증은 각각 성능 불안정성 문제와 확장성/성능 트레이드오프 문제에 대한 강건성(robustness)시험을 통해 이뤄졌다. 제안하는 기법은 기존 기법들에 비해 성능의 향상 폭은 미미하다. 또한 데이터의 변동 정도를 나타내는 지표인 표준 편차의 측면에서도 의미 있는 개선을 보이지 못하였다. 하지만, 성능의 변동 폭을 나타내는 범위의 측면에서는 기존 기법들에 비해 개선을 보였다. 첫 번째 실험에서는 모델 생성 전후의 성능 변동폭에서 51.31%의 개선을, 두 번째 실험에서는 군집 수 변화에 따른 성능 변동폭에서 36.05%의 개선을 보였다. 이는 제안하는 기법이 성능의 향상을 보여주지는 못하지만, 성능 안정성의 측면에서는 기존의 기법들을 개선하고 있음을 의미한다.

주제어 : 추천시스템, 적응형 시스템, 협업 필터링, 하이브리드 필터링, 클러스터링

논문접수일 : 2015년 2월 10일 논문수정일 : 2015년 3월 2일 게재확정일 : 2015년 3월 20일
투고유형 : 국문급행 교신저자 : 유은순

1. 서론

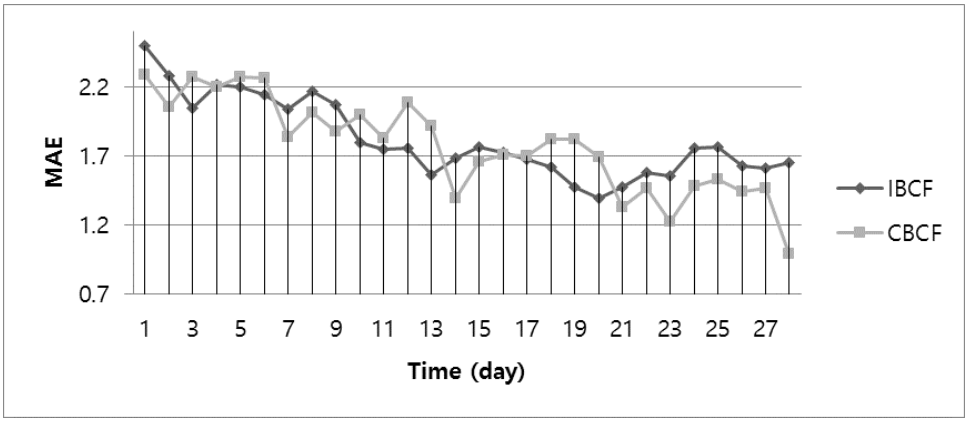
정보의 양이 폭발적으로 증가함에 따라 사용자들이 인터넷에서 필요한 정보를 찾는데 많은 어려움을 겪고 있다. 정보 과부하로 인해 발생하는 문제들을 해결하기 위해 사용자의 취향과 선호도를 고려하여 사용자에게 맞는 정보를 제공해주는 추천시스템이 그 어느 때보다도 중요해졌다. 현재 아마존(Linden et. al., 2003), 구글(Das et. al., 2003), 넷플릭스(Bennet and Lanning, 2007), 티보(Ali and Stam, 2004) 그리고 야후(Park and Pennock, 2007)와 같은 선도 기업들은 이미 개인

* 이 논문은 2013년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (NRF-2013R1A1A2057943)

화된 추천이 가능한 추천 시스템을 운용하고 있다. 이 시스템들의 주요한 요구사항은 추천의 성능(performance)과 시스템의 확장성(scalability)이다. 이 요구사항들을 충족시키기 위해, 내용 기반 필터링(CBF, Content-Based Filtering), 인구통계학적 필터링(DF, Demographic Filtering), 협업 필터링(CF, Collaborative Filtering) 등의 기법들이 제안되었다. 이 중, CBF와 DF는 외부 정보를 필요로 한다는 한계점으로 인해, 다양한 도메인에 적용이 불가능하다. 따라서 비교적 도메인의 제약이 적은 CF가 널리 사용되고 있다. (Bhosale et al., 2015; Bobadilla et al., 2013; Lee et al., 2012)

CF(Su et al., 2009; Hameed et al., 2012; Natarajan et al., 2013; Joshi et al., 2015; Cho and Cho, 2007; Im and Kim 2012) 기법들은 크게 메모리 기반(Memory-based) CF, 모델 기반(Model-based) CF, 하이브리드(Hybrid) CF로 나눌 수 있다. 이 중, 모델 기반 CF는 베이지안(Bayesian) 모델이나 군집화 모델, 의존성 네트워크(dependency network) 등의 모델을 사용해서 CF의 단점을 보완한 방법이다. 이는 희박성(sparsity) 문제와 확장성 문제 등을 개선하며, 예측 성능을 높일 수 있다. 하지만, 모델 생성 비용이 크고(expensive model-building) 성능과 확장성 간의 트레이드오프(trade-off)가 발생한다. 성능과 확장성 간의 트레이드오프는 희박성 문제의 일종인 적용범위 감소(reduced coverage) 문제에 기인한다. 또한, 높은 모델 생성 비용은 성능 불안정 문제의 원인이 된다. 이는 높은 비용으로 인해 도메인 환경의 변화를 모델에 즉각적으로 반영할 수 없기 때문이다. 반영 되지 못한 도메인 환경 변화의 누적은 시스템의 성능을 저하시킨다. (Lee and Baek, 2014)

<Figure 1>과 <Table 1>은 불안정성 문제를 증명하기 위한 실험의 결과이다. 메모리 기반 CF의 일종인 아이템 기반 CF(IBCF, Item-based CF)와 모델 기반 CF의 일종인 군집화 기반 CF(CBCF, Clustering-based CF)를 데이터 희박성 문제의 영향이 최소화된 환경에서 성능을 비교하였다. IBCF 기법은 코사인 유사도를 이용해 구현하였으며, CBCF는 아이템을 코사인 유사도에 따라, K-최근접 이웃(K-NN, K-Nearest Neighborhood) 알고리즘을 이용해 군집화하여 유사 아이템 집합을 대신하는 방법으로 구현하



〈Figure 1〉 MAE of IBCF and CBCF according to Service Time

였다. 군집의 수는 베이지안 정보 기준(BIC, Bayesian Information Criteria)를 기준으로 결정하였다. 실험 데이터로는 MovieLens 1M 데이터 셋을 이용하였다. 이것은 3952개의 영화에 대해 6040명의 사용자들이 입력한 1,000,209개의 평가 점수 데이터 셋이다. 테스트 데이터로는 이 데이터 셋의 평가점수 중 시간 순으로 마지막 28만개를 28일간, 1일당 10,000회의 서비스 요청을 가정하고 입력하였으며, 나머지 데이터를 훈련 데이터로 사용하였다. 실험은 각 기법의 성능을 선호도 예측치와 사용자의 평가 점수간 평균절대오차(MAE, Means Absolute Error)를 이용해 측정하였다. CBCF의 경우 7일을 주기로 군집화를 수행한다고 가정하였다. <Figure 1>은 각 기법의 시간의 흐름에 따른 MAE 변화를 표시한 그래프이다. <Table 1>은 각의 기법의 MAE의 평균과 표준편차, 범위를 나타낸 표이다.

〈Table 1〉 Average, Standard Deviation and Range of MAE for IBCF and CBCF

	IBCF	CBCF
Average	1.814	1.770
Standard Deviation	0.283	0.345
Range	1.099	1.292

<Figure 1>과 <Table 1>은 CBCF가 IBCF에 비해 평균적으로 약간 높은 성능을 보이지만, 희박성 문제의 영향이 제한된 상황에서 CBCF가 IBCF에 비해 불안정한 신뢰도를 보임을 나타낸다. CBCF는 IBCF에 비해 평균 0.044(2.426%) 개선된 성능을 보이지만, 성능의 표준편차와 범위는 각각 0.062, 0.193 더 높다. 또한, CBCF의 경우 군집화 시점에서부터 성능이 서서히 저하되며, 다음 군집화 시점에서 다시 개선되는 모습을 볼 수 있다. 이는 CBCF가 모델 재생성 비용으로 인한 성능 불안정성 문제를 갖고 있음을 증명한다.

Lee(Lee et al., 2014)는 이 문제들을 해결하기 위해 적응형 군집화 기반 협업 필터링(ACCF, Adaptive Clustering based CF)를 제안하였다. 이 기법은 사용자 또는 아이템의 추가와 평가 점수의 입력에 따라 해당 사용자나 아이템의 군집을 지역적으로 재배치하는 방법을 사용하였다. 하지만, 이는 시스템의 부담을 가중시키며, CBCF의 신뢰도 불안정성을 완전히 해결하지 못한다.

본 연구에서는 위의 문제를 해결하기 위해, CBCF에 마르코프(Markov) 전이확률모델(transition probability model)과 퍼지 군집화(Fuzzy Clustering)의 개념을 접목하여, 적용범위 감소 문제와 성능 불안정성 문제를 해결한 예측적 군집화 기반 CF(PCCF, Predictive Clustering-based CF) 기법을 제안한다. 이 기법은 첫째, 사용자 기호(Preference)의 변화를 추적하여 정적인 모델과 동적인 사용자간의 괴리 해소를 통해 성능 불안정 문제를 개선한다. 둘째, 전이확률과 군집 소속 확률에 기반한 적용범위 확장으로 적용범위 감소 문제를 개선한다.

제안하는 기법은 4개의 과정으로 이뤄진다. 첫째, 기호 군집화(Preference-Clustering) 과정을 통해 사용자들의 기호를 정규화하며, 둘째, 기호 전이 탐지(Preference Transition Detection) 과정을 통해 사용자들의 평가점수 입력으로부터 사용자들의 기호 변화를 탐지한다. 셋째, 성향 군집화(Propensity-Clustering) 과정을 통해 사용자들의 기호 변화 패턴(성향)를 통해 사용자들의 성향을 정규화한다. 넷째, 선호도 예측(Preference Prediction) 과정을 통해 선호도 예측 모델을 만들고 사용자

들의 아이템에 대한 선호도를 예측한다.

본 논문의 구성은 다음과 같다: 2장에서는 본 연구의 관련 연구들에 대해 살펴보고, 3장에서는 본 논문에서 제안하는 PCCF 기법에 대해 상세히 기술한다. 4장에서는 본 연구의 타당성과 유효성을 입증하기 위한 실험 및 검증 방법과 그 결과를 기술하고 분석한다. 5장에서는 본 논문에서 제시하는 내용을 정리하고 향후 연구 방향을 기술한다.

2. 관련 연구

CBCF는 적용범위 감소 문제와 성능 불안정 문제에도 불구하고, CF의 확장성 문제와 희박성의 문제를 개선하는데 유용한 방법이다. 때문에 위의 두 가지 문제를 해결하기 위한 다양한 방법들이 제안되고 있다. 하지만 군집화 방법의 개선에만 치중하여 문제의 근본적인 원인을 해결하지 못하거나, CBF와의 접목으로 인해 사용자의 프로파일이나 설문조사와 같은 외부 데이터(external data)를 필요로 하는 또 다른 문제점을 낳았다.

Wen와 Zhou(Wen and Zhou, 2012)는 아이템을 동적으로 군집화하는 방법을 통해 아이템의 추가나 삭제, 수정으로 인한 추천시스템의 성능 저하를 해결하였다. 하지만 단일 클러스터 안에 과다한 아이템이 존재하여 적용범위 감소 문제가 발생할 가능성이 있다.

Gong(Gong, 2010)은 사용자 군집과 아이템 군집을 결합하는 방법을 제안했다. 이는 아이템에 대한 사용자의 순위에 기반하여 사용자를 군집화하고, 대상 사용자와 각각의 사용자 군집이 갖고 있는 군집 중심(cluster center) 간의 유사도(similarity)에 기반하여 대상 사용자의 유사 사용자 집합을 생성하는 방법이다. 하지만 이는 적용범위 감소 문제를 발생시킬 가능성이 있으며, 사용자/아이템의 추가, 삭제, 수정이 어렵다는 문제점이 있다.

Li 와 Dong(Li and Dong, 2010)은 확률 군집화(probabilistic clustering) 모델 기반의 CF를 제안하였다. 이 방법의 핵심은 퍼지 군집화(fuzzy clustering)를 기반으로 사용자와 아이템을 군집화 하는 것이다. 이 방법은 추천 시스템의 성능을 개선하고 적용범위 감소 문제를 일부분 해소할 수 있다. 하지만 사용자의 속성이나 행위에 따른 사용자 범주화(categorization)를 추가로 요구한다.

Pham(Pham et al, 2011)은 아이템에 대한 사용자들의 평가점수(rating) 대신 사용자들의 사회관계망(social network)을 분석하여 유사한 사용자들의 그룹을 찾아내는 사회관계(social relationship) 기반 CF 기법을 제안하였다. 하지만 다양한 유형의 사회 관계망 각각에 대한 분석이 필요하며, 추가적인 외부정보를 필요로 한다는 문제가 있다.

Bellogin와 Parapar(Bellogin and Parapar, 2012)는 그래프 분할 기반의 군집화 기법인 정규분할(N-Cut, Normalized Cut)을 이용하여 유사 사용자 집합을 구성하는 방법을 제안하였다. 이는 기존의 CF보다 개선된 성능을 보이지만, 적용범위 감소 문제를 해결하지는 못했다.

Simon(Simon et al., 2013)은 사용자들의 비적극성에 기인하는 희박성 문제를 해결하기 위해 비명시적인 사용자 피드백을 활용하는 방법을 제안하였다. 이 방법은 사용자들의 구매 내역(history)을 계층 분할 고차원 비모수 군집화(high-dimensional, parameter-free, divisive hierarchical

clustering)를 이용해 분석한다. 이는 데이터 희박성 문제 해결에 효과적이지만, 비명시적인 피드백이 항상 사용자의 선호에 대한 정확한 정보를 제공하지는 않는다는 문제점이 있다.

Zhou(Zhou et al., 2013)는 의미적 관계 분석에 기반한 CBCF를 제안하였다. 이는 아이템과 사용자, 이용 내역간의 연관관계(correlation)와 의미적 관계(semantic relationship)를 벡터 공간에서 기술하고 벡터들을 퍼지 C-평균 알고리즘을 기반으로 군집화하는 방법이다. 이는 유사한 서비스들을 군집화함으로써 서비스 검색 엔진의 성능을 향상시켰지만 의미적 상호운용성을 지원하기 위해서는 도메인 온톨로지가 필요하며, 매개변수가 부족한 다른 서비스에는 적합하지 않다.

Li와 Murata(Li and Murata, 2012)는 다차원(Multi-dimensional) 군집화 기반 CF를 제안하였다. 이는 백그라운드 데이터로 구성된 아이템/사용자 프로파일을 기반으로 이들을 군집화하고 군집 정리(clustering pruning)과정을 거친 후, 이웃의 가중치 평균을 통해 선호도를 예측하는 방법이다. 이러한 방법은 추천의 성능을 유지하면서 아이템의 다양성이 증가할 때도 성능 균형을 유지하는 장점이 있지만, 모델 기반 CF의 한계점을 개선하지 못하고 있다.

George와 Merugu(George and Merugu, 2005)는 가중치 기반 이중 군집화(weighted co-clustering) 알고리즘을 이용한 CF를 제안하였다. 이는 이중 군집화를 통해 유사 아이템/사용자 집합을 동시에 생성하고 이중 군집들의 평균 순위에 기반하여 선호도를 예측하는 방법이다. 이 방법은 CBCF의 확장성을 개선하지만, 적용범위 감소 문제와 성능 불안정성 문제에는 효과를 보이지 못한다.

Zhirao(Zhirao, 2011)는 커뮤니티 기반의 CF을 제안했다. 이것은 같은 집단에 속해 있는 사용자들은 유사한 취향을 갖고 있다는 가정에 기반한다. 이것은 유사 사용자 집합 구성의 범위를 좁히는 효과적인 방법으로 데이터의 희소성 문제를 어느 정도 해결할 수 있지만, 추가적인 외부 정보를 필요로 한다.

Tseng(Tseng et. al., 2011)는 클라우드 모델을 이용한 기본 투표 전략(Default voting schema)을 제안했다. 클라우드 모델은 사용자가 과거에 상품에 대해 매긴 순위를 이용하여 사용자의 전체적인 선호도를 분석하는 방법이다. 이 방법은 사용자의 관심과 선호도를 좀 더 정확하게 기술하고 데이터의 희소성을 감소시키는 효과를 보이지만 사용자에게 추가 정보를 요구한다는 단점을 갖는다.

Khoshneshin (Khoshneshin et al., 2010)는 점진적 이중 군집화를 통한 증분 CF(ICFEC, Incremental Collaborative Filtering via co-Clustering)을 제안하였다. 이 방법은 반복적 알고리즘을 사용하는 기존의 지역 최적 군집화 기반 CF에 비해 개선된 성능을 보인다. 또한 CBCF의 한계인 운용 중 모델 변경의 어려움을 진화적 군집화를 이용해 개선하였다. 하지만 이 방법은 시스템 구동 중, 사용자나 아이템의 추가/삭제 만을 지원할 뿐, 성능 불안정성 문제에 대한 해결책을 제공하진 못한다.

3. 예측적 군집화 기반 협업 필터링 기법

TV 프로그램 추천 시스템 S와 매일 오후 7시에서 9시에 TV를 시청하는 사용자 A가 있다고 가정하자. A가 주중에는 드라마를 주로 시청하지만, 주말에는 리얼리티 쇼를 주로 시청한다고

할 때, S가 사용자의 변화를 고려하지 않는다면, S는 평균적으로 드라마를 더 많이 이용한 A의 이용내역을 기반으로 주말에도 A에게 드라마를 추천할 것이다. 이는 A의 S에 대한 서비스 만족도를 저하시킬 수 있다.

이 장에서는 위의 문제를 해결하기 위한 예측적 군집화 기반 협업 필터링(PCCF, Predictive Clustering-based Collaborative Filtering) 기법의 개념을 기술한다. 이 기법은 사용자의 기호 변화를 전이확률모델을 이용해 추적한다. 이는 첫째, 정적인 모델과 동적인 사용자간의 괴리 해소를 통한 성능 불안정 문제 해결과 둘째, 전이확률에 기반한 적용범위 확장으로 적용범위 감소 문제 해결을 목적으로 한다. 다음은 본 논문에서 제안하는 사용자 선호도 예측 모델이다. 먼저, 이 모델은 다음과 같은 가정을 근간으로 한다.

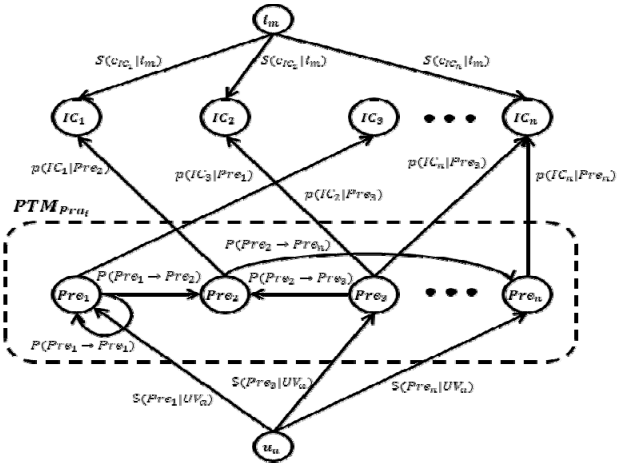
- 가정. 사용자의 기호는 지속적으로 변화하며, 이 변화의 패턴이 사용자의 성향이다.

본 연구에서는 위 가정을 바탕으로, 사용자의 기호를 범주화하고, 사용자의 기호 변화를 탐지하여 기호 전이모델을 구축하며, 사용자들의 기호 전이 패턴을 범주화하며, 이에 따라 사용자의 기호 변화를 추적한다. 다음은 제안하는 사용자 선호도 예측 모델을 구성하는 주요 요소들에 대한 정의다.

- 정의 1 (기호 군집). 기호 군집은 유사한 기호를 가진 사용자들의 집합으로, 유사한 아이템에 대해 유사한 평가를 한 사용자일수록 같은 기호 군집에 속할 가능성이 높다.
- 정의 2 (기호 전이모델). 기호 전이모델은 사용자의 기호 변화를 마르코프 전이확률모델을 응용하여, 사용자들의 기호 전이와 전이의 소요시간을 확률적으로 나타낸 것이다.
- 정의 3 (성향 군집). 성향 군집은 유사한 성향을 가진 사용자들의 집합으로, 유사한 기호 전이모델을 가진 사용자일수록 같은 성향 군집에 속할 가능성이 높다.

<Figure 2>는 위에서 정의된 요소들을 바탕으로 정의된 사용자 선호도 예측 모델의 도시이다.

u_a는 a번째 사용자를, i_m은 m번째 아이템을, IC_n은 n번째 아이템 군집을, Pre_n은 n번째 기호를, Pro_n은 n번째 성향을, $PTNpro_n$은 n번째 성향의 사용자들의 기호 전이모델을 나타낼 때, $S(Pre_i|UV_a)$는 u_a와 Pre_i간의 유사도, $P(Pre_i \rightarrow Pre_j)$는 Pre_i에서 Pre_j로의 전이확률, $p(Pre_j|IC_k)$는 Pre_j에 속한 사용자들의 에 IC_k대한 선호도, $S(c_{IC_k}|i_m)$는 i_m과 IC_k의



〈Figure 2〉 Probabilistic Modeling of Preference of User for Item

중심간의 유사도이다. 이때, u_a의 i_m에 대해 선호도 예측치, $p_{u_a i_m}(predicted)$는 <Equation 1>과 같이 추정할 수 있다.

$$p_{u_a,i_m}(predicted) = \sum_i \sum_j \sum_k S(Pre_i|UV_a)P(Pre_i \rightarrow Pre_j)p(Pre_j|IC_k)S(c_{IC_k}|i_m)$$

〈Equation 1〉

제안하는 기법은 4개의 과정으로 구성된다.

1) 기호 군집화: 사용자들의 기호를 정규화하기 위해 사용자들을 기호에 따라 군집화하고 각 기호를 나타내는 기호 벡터를 생성한다.
2) 기호 전이 탐지: 기호 군집화 과정에서 생성된 기호 벡터들을 바탕으로 사용자들의 기호 변화를 탐지하고 그것을 기호 전이 벡터로 나타낸다.
3) 성향 군집화: 사용자들의 성향을 정규화하기 위해 사용자들을 기호 전이 패턴, 즉 성향에 따라 군집화하고 각 성향을 나타내는 기호 전이모델을 생성한다.
4) 선호도 예측: 기호 벡터와 기호 전이모델을 바탕으로 사용자 선호도 예측모델을 생성하고, 사용자의 각 아이템에 대한 선호도를 예측한다.

3.1. 기호 군집화

기호 군집화는 사용자들을 각 아이템 군집에 대한 선호도에 따라 군집화하여, 사용자들의 기호를 정규화하기 위한 과정이다. 이 과정은 아이템 군집화와 사용자 군집화, 기호 벡터 생성의 세 단계로 나뉜다. 아이템 군집화는 사용자들의 소비 경향에 따라 아이템들을 범주화하는 과정이다. 사용자 군집화는 사용자들을 기호에 따라 군집화하여 각 기호를 나타내는 군집을 도출하는 과정이다. 마지막으로 기호 벡터 생성은 사용자들의 각 기호들의 대푯값을 결정하는 과정이다.

3.1.1. 아이템 군집화

아이템 군집화는 사용자들이 입력한 평가 점수를 기반으로 추정된 아이템들간의 유사도를 기준으로 이뤄진다. 특정한 두 아이템간의 유사도는 두 아이템을 모두 평가한 적이 있는 사용자들의 평가 점수들을 바탕으로 코사인 유사도를 이용해 구한다. 그 과정은 <Equation 2>과 같다.

$$IS_{i,j} = \frac{\sum_{u \epsilon U} (R_{u,i} - \overline{R_i})(R_{u,j} - \overline{R_j})}{\sqrt{\sum_{u \epsilon U} (R_{u,i} - \overline{R_i})^2} \sqrt{\sum_{u \epsilon U} (R_{u,j} - \overline{R_j})^2}}$$

〈Equation 2〉

$IS_{i,j}$가 i_i와 i_j간의 유사도일 때, U는 아이템 i와 j를 모두를 평가한 적이 있는 사용자들의 집합, $R_{u,i}$는 사용자 u의 아이템 i에 대한 평가 점수, $\overline{R_i}$는 사용자 집합 U의 아이템 i에 대한 평가 점수의 평균이다.

아이템 군집화 알고리즘은 K-NN 알고리즘을 기반으로 한다. 이때 원소간의 거리는 원소간의 유사도의 역수가 된다. 군집의 수는 BIC를 기준으로 결정한다. <Table 2>는 아이템 군집화 과정의 세부 알고리즘이다.

〈Table 2〉 Algorithm of Item-Clustering

Algorithm of Item-Clustering

IC_i: i-th item cluster
i_j: j-th item
Ic_i: center of IC_i
N_{ICi}: number of items in IC_i
$S_{i,j}$: similarity between i_i and i_j
 	
While $\mu(Inter\ Cluster\ Deviation) < \alpha$,
For $j = 1 \rightarrow j = (Number\ of\ Element)$
For $i = 1 \rightarrow i = k(Number\ of\ Cluster)$
IF $S_{Ic_i,j} > S_{Ic_{i-1},j}$
$x_j \in IC_i$
End IF
End For
End For
For $i = 1 \rightarrow i = k(Number\ of\ Cluster)$
For $j = 1 \rightarrow j = N_{IC_i}$
IF $\sum_k^{N_{IC_i}} S_{j,k} > \sum_k^{N_{IC_i}} S_{j-1,k}$
$Ic_i = i_j$
End If
End For
End For
End While

3.1.2. 사용자 군집화

사용자 군집화는 아이템 군집과 사용자들의 평가 점수를 바탕으로 구성된 특성 벡터와 사용자간 유사도를 기준으로 이뤄진다. 특성 벡터는 사용자의 각 아이템 군집에 속한 아이템들에 대한 평가 점수의 평균으로 구성되며, 그 차원의 수는 아이템 군집의 수와 같다. 그 과정은 <Equation 3>과 같다.

$$\overrightarrow{UV_i} = \left(\overline{R_{i,IC_1}}, \overline{R_{i,IC_2}}, \cdots, \overline{R_{i,IC_n}}\right)$$

〈Equation 3〉

$\overrightarrow{UV_i}$가 u_i의 특성벡터일 때, n은 아이템 군집의 수, $\overline{R_{i,IC_j}}$은 u_i의 IC_j에 속한 모든 아이템에 대한 평가 점수의 평균이다.

특정한 두 사용자간의 유사도는 두 사용자에 의해 모두 평가된 적이 있는 아이템들에 대한 평가 점수들을 바탕으로 코사인 유사도를 이용해 구한다. 그 과정은 <Equation 4>와 같다.

$$US_{a,u} = \frac{\sum_{i \epsilon I} (R_{a,i} - \overline{R_a})(R_{u,i} - \overline{R_u})}{\sqrt{\sum_{i \epsilon I} (R_{a,i} - \overline{R_a})^2} \sqrt{\sum_{i \epsilon I} (R_{u,i} - \overline{R_u})^2}}$$

〈Equation 4〉

$US_{a,u}$가 u_a와 u_u간의 유사도일 때, I는 사용자 α와 u모두에 의해 평가된 적이 있는 아이템들의 집합, $R_{a,i}$는 사용자 a의 아이템 i에 대한 평가 점수, $\overline{R_a}$는 아이템 집합 I에 대한 사용자 a의 평가 점수의 평균이다.

사용자 군집화에는 기대치 최대화(EM, Expectation Maximization) 알고리즘을 기반으로 가우시안-베이지안(Gaussian-Bayesian) 확률모델을 이용한다. 군집의 수는 BIC를 기준으로 결정한다. <Table 3>은 사용자 군집화 과정의 세부 알고리즘이다.

〈Table 3〉 Algorithm of Preference-Clustering

Algorithm of Preference-Clustering
UC_i : i-th preference cluster u_j : j-th user Uc_i : center of Uc_i μ_i : average of elements in Uc_i σ_i : standard deviation of elements in Uc_i $P(UC_i)$: probability of random user is included in Uc_i $P(UC_i \mid u_j)$: probability of u_j is element of Uc_i $S_{i,j}$: similarity between u_i and u_j L_{UCiuj} : likelihood between u_j and UC_i

While μ*(Inter Cluster Deviation)* < α,
For $j=1 \rightarrow j =$ *(Number of Element)*
For $i=1 \rightarrow i=k$ *(Number of Cluster)*

$$P(UC_i|u_j) = P(u_j|UC_i)\,P(UC_i)/\,P(u_j)$$

$$\cong \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{(u_i-\mu_i)^2}{2\sigma^2}} P(UC_i)$$

For $u_l \in UC_i$,

$$L_{UC_i,u_j} = P(UC_i|u_j) \sum_{l}^{N_{UC_i}} S_{j,l}$$

End For
$u_j \in UC_i$, when $L_{Uc_i u_j}$ has maximum value.
End For
Following the changed model, calculate each value of Uc_i, μ_i,σ_i, and $P(UC_i)$ of each clusters.
End For
End While

3.1.3. 기호 벡터

기호 벡터는 사용자 군집화를 통해 나타난 사용자들의 기호의 대푯값으로 각 사용자 군집에 속한 사용자들의 특성 벡터의 평균이다. <Equation 5>은 기호 벡터를 구하는 방법이다.

$$\overrightarrow{Pre_i} = \sum_{j\in UC_i}^{N_{UC_i}} \overrightarrow{UV_j} \Big/ N_{UC_i} = \frac{1}{N_{UC_i}} \sum_{j\in UC_i}^{N_{UC_i}} (\overline{R_{j,IC_1}}, \overline{R_{j,IC_2}}, \cdots, \overline{R_{j,IC_n}})$$

〈Equation 5〉

$\overrightarrow{Pre_i}$가 i번째 기호 Pre_i의 특성벡터일 때, N_{UC_i}는 사용자 군집 UC_i의 원소의 수, $\overrightarrow{UV_j}$는 u_i의 특성벡터, n은 아이템 군집의 수, $\overline{R_{i,IC_j}}$은 u_i의 IC_j에 속한 모든 아이템에 대한 평가 점수의 평균이다.

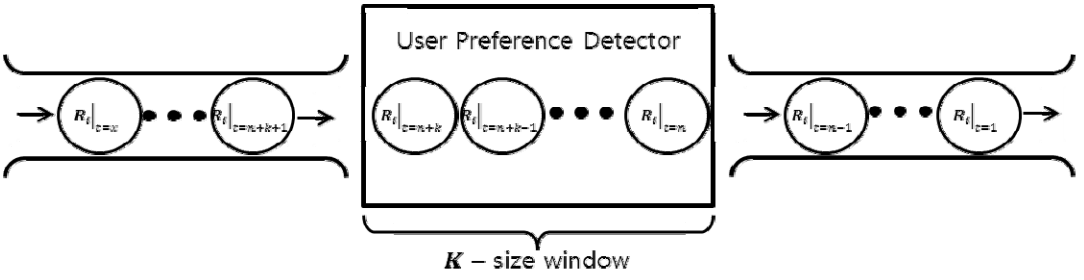
3.2. 사용자 기호 전이 탐지

사용자 기호 전이 탐지는 사용자의 기호 성향을 분석하기 위하여, 사용자가 입력한 평가 점수의 시퀀스로부터 각 시점에서의 사용자의 기호를 추정하고 그 전이를 탐지하는 과정이다. 이 과정은 기호전이 탐지와 기호 전이 벡터 생성, 두 단계로 나뉜다.

3.1.1. 기호 전이 탐지

기호 전이 탐지에는 사용자의 평가점수 입력을 특정한 크기의 윈도우로 관측하는 방법이 사용된다. <Figure 3>은 기호 전이의 탐지 방법을 도시한 것이다.

이는 사용자의 평가 점수를 바탕으로 해당 시점의 사용자 특성벡터의 추정을 기반으로 한다. $R_{i,n}$를 u_i가 시간 n에 입력한 평가 점수, k를 윈



〈Figure 3〉 Model of User Preference Transition Detection

도우의 크기라 할 때, $R_i|_{t=n+k}$가 입력된 시점에서, u_i의 사용자 특성벡터, $\overrightarrow{UV_i}|_{t=n+k}$의 m번째 항은 $R_{i,n}$부터 $R_{i,n+k}$까지의 평가점수 중, IC_m의 원소인 평가점수들의 평균으로 추정된다. 사용자의 기호는 $\overrightarrow{UV_i}|_{t=n+k}$와 가장 높은 유사도를 보이는 기호 벡터를 통해 추정된다. <Table 4>는 사용자 기호 전이탐지 과정의 세부 알고리즘이다.

〈Table 4〉 Algorithm of User Preference Transition Detection

Algorithm of User Preference Transition Detection

$R_i|_{t=n}$: rating point inserted by u_i at time, n

$\mu(R_i)|_{t=n\rightarrow n+k}$: average of rating point inserted by u_i at time, to time, $n+k$

IF $R_i|_{t=n+k}$ is inserted,

$\mu(R_i)|_{t=n\rightarrow n+k} = \mu(R_i)|_{t=n-1\rightarrow n+k-1} + \frac{R_i|_{t=n+k}}{k} - \frac{R_i|_{t=n-1}}{k}$

For $j = 1 \rightarrow j = (Number\ of\ Preference - Clusters)$

$S\left(\mu(R_i)|_{t=n\rightarrow n+k}, \overrightarrow{Pre_j}\right) = \left(\frac{\mu(R_i)|_{t=n\rightarrow n+k} \cdot \overrightarrow{Pre_j}}{\|\mu(R_i)|_{t=n\rightarrow n+k}\| \|\overrightarrow{Pre_j}\|}\right)$

IF $S\left(\mu(R_i)|_{t=n\rightarrow n+k}, \overrightarrow{Pre_j}\right) > S\left(\mu(R_i)|_{t=n\rightarrow n+k}, \overrightarrow{Pre_{j-1}}\right)$, $u_i|_{t=n+k} \in Pre_j$

End IF

End For

End IF

3.2.1. 기호 전이 벡터

사용자의 기호 전이 벡터는 성향 군집화의 기반이 된다. 기호 전이 시퀀스에는 전이된 기호의 기호 벡터, 전이 시점의 사용자 벡터, 전이 소요시간이 포함된다. 사용자 기호 전이 벡터는 <Equation 6>와 같이 정의된다.

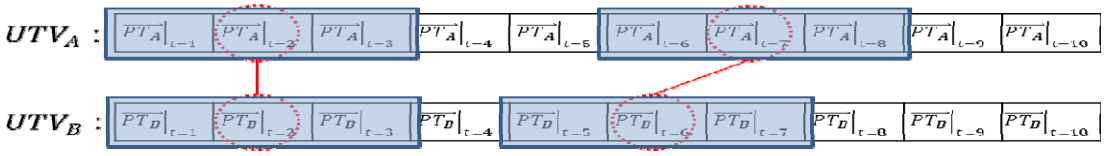
$$\overrightarrow{UTV_i} = \left(\overrightarrow{PT_i}|_{t=1}, \cdots, \overrightarrow{PT_i}|_{t=n}\right) = \begin{pmatrix} \overrightarrow{Pre}|_{t=1}^{i} & \cdots & \overrightarrow{Pre}|_{t=n}^{i} \\ \overrightarrow{UV}|_{t=1}^{i} & \cdots & \overrightarrow{UV}|_{t=n}^{i} \\ \Delta t|_{t=1}^{i} & \cdots & \Delta t|_{t=n}^{i} \end{pmatrix}$$

〈Equation 6〉

$\overrightarrow{UTV_i}$를 사용자 i의 기호 전이 벡터, $\overrightarrow{PT_i}|_{t=n}$을 시점, $t=n$에서 사용자 i의 기호 전이라고 할 때, $\overrightarrow{PT_i}|_{t=n}$는 각 전이 시점에서의 $\overrightarrow{PT_i}|_{t=n}$의 시퀀스로 이뤄진다. $\overrightarrow{PT_i}|_{t=n}$는 다음의 세 가지 요소들로 구성된다. $\overrightarrow{Pre}|_{t=n}^{i}$는 $t=n$에서 사용자 i의 기호의 특성 나타내는 기호 벡터, $\overrightarrow{UV}|_{t=n}^{i}$는 $t=n$에서 사용자 i의 특성을 나타내는 사용자 특성 벡터, $\Delta t|_{t=n}^{i}$는 $t=n$에서 발생한 사용자 의 기호 전이가 일어나기까지의 소요시간, 즉 이전의 기호 전이와의 시간적 간격을 의미한다.

3.3. 성향 군집화

성향 군집화는 사용자들을 기호 전이의 유사도에 따라 군집화하여 사용자들의 기호 변화의 패턴을 정규화하는 과정이다. 제안하는 기법에서는 이 패턴을 사용자의 성향으로 본다. 이 과정은 기호 전이 유사도를 추정하고, 이 유사도를 바탕으로 군집화를 통해 기호 전이의 패턴(성향)을 찾아내며, 추출된 성향들을 나타내는 기호 전이 모델을 구성하는 세 단계로 나뉜다.



〈Figure 4〉 Algorithm of Preference Transition Sequence Similarity

3.3.1. 기호 전이 유사도 추정

기호 전이 유사도는 기호의 전이 순서와 전이 간 시간 간격을 바탕으로 추정된다. 유사도의 추정에는 윈도우가 사용된다. 유사도를 측정하고자 하는 두 사용자 벡터에서 서로 가장 유사한 기호를 가지고 있었던 시점을 쌍으로 묶고 유사도 추정의 기점으로 삼는다. 그리고 이 쌍의 전후를 윈도우를 이용해 탐색하여 그 전이 과정 또한 유사한지 탐색한다. <Figure 4>는 UTV_A와 UTV_B간의 유사도를 구하는 과정을 나타낸 것이다.

기호 전이 유사도의 추정은 기호 벡터간의 유사도, 기호 벡터와 사용자 벡터간의 유사도, 기호 전이 순서와 시간 간격을 바탕으로 이뤄진다. <Table 5>는 기호 전이 유사도 추정 알고리즘의 상세한 기술이다.

〈Table 5〉 Algorithm of Preference Transition Sequence Similarity

Algorithm of Preference Transition Sequence Similarity
UTV_i: user preference transition vector of th User N_{UTV_i}: number of elements in UTV_i $S(\overrightarrow{Pre_i}, \overrightarrow{Pre_j})$: similarity between i th preference and j th preference $S(\overrightarrow{UV_i}, \overrightarrow{UV_j})$: similarity between i th user and j th user $PTS_{i,j}$: preference transition similarity between i th user and j th user

For $n = 1 \rightarrow n = N_{UTV_i}$

For $m = 1 \rightarrow m = N_{UTV_j}$

$$S\left(\overrightarrow{Pre}\Big|_{t=n}^{i}, \overrightarrow{Pre}\Big|_{t=m}^{j}\right) = S\left(\overrightarrow{Pre}\Big|_{t=n}^{i}\Big|\overrightarrow{UV}\Big|_{t=m}^{j}\right) S\left(\overrightarrow{Pre}\Big|_{t=m}^{j}\Big|\overrightarrow{UV}\Big|_{t=n}^{i}\right) S\left(\overrightarrow{UV}\Big|_{t=n}^{i}, \overrightarrow{UV}\Big|_{t=m}^{j}\right)$$

$$= \left(\frac{\overrightarrow{Pre}\Big|_{t=n}^{i} \cdot \overrightarrow{UV}\Big|_{t=m}^{j}}{\left\|\overrightarrow{Pre}\Big|_{t=n}^{i}\right\| \left\|\overrightarrow{UV}\Big|_{t=m}^{j}\right\|}\right)\left(\frac{\overrightarrow{Pre}\Big|_{t=m}^{j} \cdot \overrightarrow{UV}\Big|_{t=n}^{i}}{\left\|\overrightarrow{Pre}\Big|_{t=m}^{j}\right\| \left\|\overrightarrow{UV}\Big|_{t=n}^{i}\right\|}\right)\left(\frac{\overrightarrow{UV}\Big|_{t=n}^{i} \cdot \overrightarrow{UV}\Big|_{t=m}^{j}}{\left\|\overrightarrow{UV}\Big|_{t=n}^{i}\right\| \left\|\overrightarrow{UV}\Big|_{t=m}^{j}\right\|}\right)$$

End For

IF $S\left(\overrightarrow{Pre}\Big|_{t=n}^{i}, \overrightarrow{Pre}\Big|_{t=m}^{j}\right) > S\left(\overrightarrow{Pre}\Big|_{t=n}^{i}, \overrightarrow{Pre}\Big|_{t=m-1}^{j}\right)$

$$pair\left(\overrightarrow{Pre}\Big|_{t=n}^{i}\right) = \overrightarrow{Pre}\Big|_{t=m}^{j}$$

End IF

End For

For $n = 1 \rightarrow n = N_{UTV_i}$

$$\overrightarrow{Pre}\Big|_{t=x}^{j} = pair\left(\overrightarrow{Pre}\Big|_{t=n}^{i}\right)$$

$$PTS_{i,j} = PTS_{i,j} + \frac{S\left(\overrightarrow{Pre}\Big|_{t=n-1}^{i}, \overrightarrow{Pre}\Big|_{t=x-1}^{j}\right) S\left(\overrightarrow{Pre}\Big|_{t=n}^{i}, \overrightarrow{Pre}\Big|_{t=x}^{j}\right) S\left(\overrightarrow{Pre}\Big|_{t=n+1}^{i}, \overrightarrow{Pre}\Big|_{t=x+1}^{j}\right)}{\left|\Delta t\right|_{t=n-1}^{i} - \Delta t\Big|_{t=x-1}^{i}\left|\left|\Delta t\right|_{t=n}^{i} - \Delta t\right|_{t=x}^{i}\right|}$$

End For

3.3.2. 기호 전이 패턴 군집화

기호 전이 패턴 군집화는 앞서 추정된 기호 전이 유사도, 를 바탕으로 변형된 K-NN 알고리즘을 이용해 수행된다. K-NN 알고리즘은 원소간 거리의 총합이 최소가 되는 군집의 중심을 찾지만 이 알고리즘에서는 원소간 유사도의 합이 최대가 되게 하는 군집의 중심을 찾는다. 군집의 수는 BIC를 기준으로 결정한다. <Table 6>은 기호 전이 패턴 군집화 알고리즘을 기술한 것이다.

〈Table 6〉 Algorithm of Propensity-Clustering

Algorithm of Propensity-Clustering

$Pro - C_i$: i-th propensity cluster

$Pro - c_i$: center of $Pro - C_i$

While $\mu(Inter\ Cluster\ Deviation) < \alpha$,

  For $j = 1 \rightarrow j = (Number\ of\ Element)$

    For $i = 1 \rightarrow i = k(Number\ of\ Cluster)$

      IF $PTS_{Pro-c_i,j} > PTS_{Pro-c_{i-1},j}$

        $u_j \in Pro - C_i$

      End IF

    End For

    Following the changed model, calculate each value of $Pro - c_i$ of each clusters.

End For

For $i = 1 \rightarrow i = k(Number\ of\ Cluster)$

  For $j = 1 \rightarrow j = N_{Pro-C_i}$

    If $\sum_k^{N_{Pro-C_i}} PTS_{j,k} > \sum_k^{N_{Pro-C_i}} PTS_{j-1,k}$

      $Pro - c_i = UTV_j$

    End If

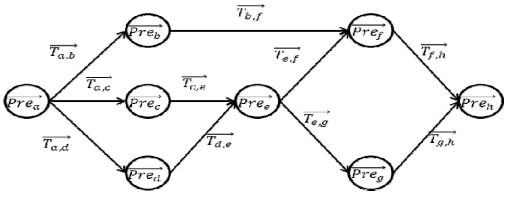
  End For

 End For

End While

3.3.3. 기호 전이 모델

기호 전이 모델은 각 성향의 사용자들의 기호 전이 패턴을 마르코프 모델의 형을 빌려 나타낸 것이다. 여기서 각 노드는 해당 성향 군집 내의 사용자들의 기호가 되고, 간선은 노드간의 전이 확률이 된다. 제안하는 모델의 경우 전이 확률은 전이에 걸리는 시간에 관한 가우시안 확률 모델이 된다. <Figure 5>는 확률 전이 모델을 도시한 것이다.



〈Figure 5〉 Preference Transition Model

$\overrightarrow{T_{a,b}}$는 Pre_a에서 Pre_b로의 전이 확률을 나타내는 벡터로써, 산술적 확률을 바탕으로 한 전이 확률과 전이 시간 간격에 대한 가우시안 확률 모델의 정보를 담고 있다. <Equation 7>은 전이 확률 모델의 간선의 정의이다.

$$\overrightarrow{T_{a,b}} = \begin{pmatrix} P(Pre_a \rightarrow Pre_b) \\ \mu\left(t_{Pre_a \rightarrow Pre_b}\right) \\ \sigma\left(t_{Pre_a \rightarrow Pre_b}\right) \end{pmatrix} = \begin{pmatrix} N_{Pre_a \rightarrow Pre_b} / N_{Pre_a} \\ \sum_i \sum_j \Delta t|_{t=j}^{i} / N_{Pre_a} \\ \sum_i \sum_j \left(\Delta t|_{t=j}^{i} - \mu\left(t_{Pre_a \rightarrow Pre_b}\right)\right) / N_{Pre_a} - 1 \end{pmatrix}$$

〈Equation 7〉

N_{Pre_a}가 Pre_a에 속해 있던 사용자들의 수, $N_{Pre_a \rightarrow Pre_b}$가 Pre_a에서 Pre_b로 전이된 사람의 수일 때, $P(Pre_a \rightarrow Pre_b)$는 Pre_a에 속한 사람들이 Pre_b로 전이되는 비율을, $\mu\left(t_{Pre_a \rightarrow Pre_b}\right)$는

전이의 시간 간격의 평균을, $\sigma(t_{Pre_a \to Pre_b})$는 시간 간격의 표준 편차를 말한다.

3.4. 선호도 예측

선호도 예측 과정은 앞에서 구성된 아이템 군집, 사용자 벡터, 기호 벡터, 기호 전이 모델 등을 바탕으로 사용자의 특정한 아이템에 대한 선호도를 예측하는 과정이다. 이 모델은 본론의 서두에서 제시한 <Figure 1>과 같으며, <Equation 1>과 같이 나타낼 수 있다. 다음은 선호도 예측 모델의 구축을 위해, <Equation 1>의 각 항의 값을 추정하는 방법을 제시한다.

3.4.1. 기호-사용자 유사도

첫 번째 항은 기호와 사용자 간의 유사도를 말한다. 이 값이 1에 가까울수록 사용자가 해당 기호에 속할 확률이 높아지며, -1에 가까울수록 낮아진다. 이는 기호 벡터와 사용자 벡터 간의 코사인 유사도를 통해 구할 수 있다. 그 과정은 <Equation 8>과 같다.

$$S(\overrightarrow{Pre_i}|\overrightarrow{UV_a}) = \left(\frac{\overrightarrow{Pre_i} \cdot \overrightarrow{UV_a}}{\|\overrightarrow{Pre_i}\|\|\overrightarrow{UV_a}\|}\right)$$

〈Equation 8〉

3.4.2. 기호 전이 확률

두 번째 항은 기호 간의 전이 확률을 말한다. 이는 사용자가 속한 성향의 기호 전이 모델을 기반으로 가우시안-베이지안 모델을 이용해 추정된다. 먼저, 기호 전이 모델의 간선에 있는 전이 시간 간격의 평균과 표준편차로부터 가우시안 확률분포모델을 생성하고, 베이지안 모델을 이용하여 해당 시간에서 기호 전이 확률을 추정한다. $P(Pre_i \to Pre_j, t)$가 특정한 성향의 사용자들에 대한 특정한 시점 t에서의 Pre_i에서 Pre_j로의 전이 확률일 때, 이것을 구하는 과정은 <Equation 9>와 같다.

$$P(Pre_i \to Pre_j, t) = P(Pre_i \to Pre_j|t) = P(t|Pre_i \to Pre_j)P(Pre_a \to Pre_b)$$

$$\cong \frac{1}{\sqrt{2\pi}\sigma(t_{Pre_a \to Pre_b})} e^{-\frac{\left(t-\mu(t_{Pre_a \to Pre_b})\right)^2}{2\sigma(t_{Pre_a \to Pre_b})^2}} P(Pre_a \to Pre_b)$$

〈Equation 9〉

3.4.3 기호 군집-아이템 군집 선호도

기호 벡터는 특정한 기호의 특성을 표현하기 위해 각 아이템 군집들에 대한 기호 군집 내 사용자들의 선호도의 평균으로 구성된다. 번째 기호의 사용자들의 번째 아이템 군집에 대한 선호도의 대푯값은 곧, 번째 기호 벡터의 번째 항의 값이다.

$$p(Pre_j|IC_k) = \overrightarrow{Pre_j}[j]$$

〈Equation 10〉

3.4.4. 아이템-아이템 군집 유사도

특정한 아이템이 아이템 군집에 포함되어 있을 확률은 아이템 군집의 중심과 아이템 간의 유사도로 측정될 수 있다. 유사도 측정 방법은 <Equation 2>에서 제시한 아이템간 유사도 측정 방법과 같다. <Equation 11>은 아이템-아이템 군

집 유사도의 측정 과정이다.

$$S(c_{IC_k}|i_m) = \frac{\sum_{u \epsilon U} (R_{u,c_{IC_k}} - \overline{R_{c_{IC_k}}})(R_{u,i_m} - \overline{R_{i_m}})}{\sqrt{\sum_{u \epsilon U} (R_{u,c_{IC_k}} - \overline{R_{c_{IC_k}}})^2} \sqrt{\sum_{u \epsilon U} (R_{u,i_m} - \overline{R_{i_m}})^2}}$$

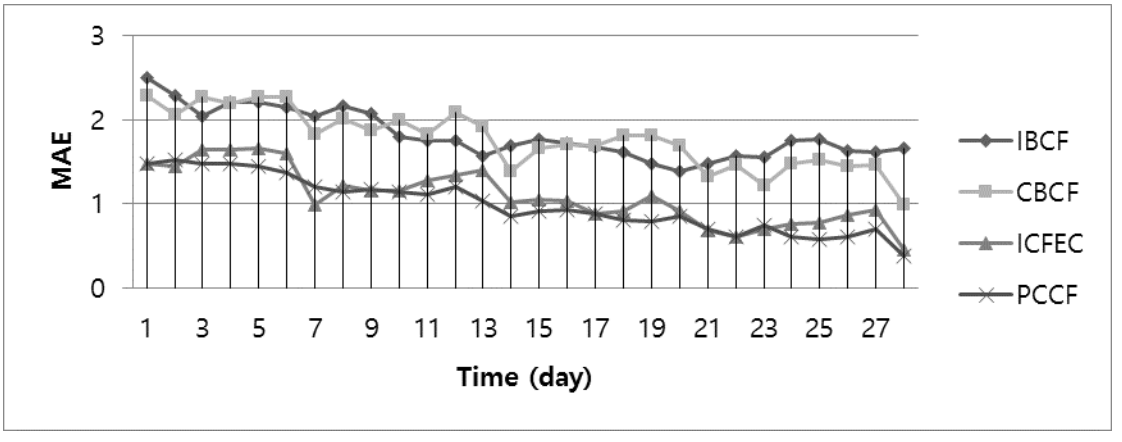
〈Equation 11〉

4. 실험 결과 및 검증

본 연구에서는 두 가지 실험을 통해 제안하는 기법이 모델 기반 CF의 성능 불안정성과 적용범위 감소 문제를 개선하였음을 검증한다. 첫 번째 실험은 데이터 희박성을 최소화한 환경에서 IBCF와 CBCF, ICFEC, PCCF를 이용해 구현된 각각의 추천 시스템들의 성능을 비교 분석하는 것이다. 이는 모델 재생성 시점 사이의 성능 변동 비교를 통해 제안하는 기법이 성능 불안정 문제를 개선하고 있음을 검증하기 위함이다. 두 번째 실험은 CBCF, ICFEC, PCCF의 군집의 개수를 최적의 개수에서 좌우로 조절하며 시스템의 성능 변화를 비교 분석하는 것이다. 적용범위 감소 문제는 모델 기반 CF에서 확장성을 위해 추천 대상 아이템/사용자의 범위를 지나치게 한정할 경우 나타난다. 이는 CBCF의 경우에서는 군집을 지나치게 세분화하여 추천 대상이 되는 아이템/사용자 군집의 크기가 지나치게 작아지는 것을 말한다. 본 연구에서는 군집 모델 상의 군집의 수를 변화시키며, 이때의 성능 변동을 비교함으로써 제안하는 기법이 적용범위 감소 문제에 대해 강건성을 보임을 입증한다.

실험 데이터로는 MovieLens 1M 데이터 셋을 사용했다. 이것은 3952개의 영화에 대한 6040명의 사용자들의 1,000,209개의 평가점수 데이터 셋이다. 테스트 데이터로는 이 데이터 셋 중, 시간 순으로 마지막 28만개를 28일간, 1일당 10,000회의 서비스 요청을 가정하였으며, 나머지 데이터를 훈련 데이터로 사용하였다. 성능 평가 기준으로는 선호도 예측치와 사용자의 평가점수 간의 MAE를 사용하였다.

실험 환경은 다음과 같다. 각 알고리즘은 Visual C++를 이용해 구현되었으며, 운영체제는 Windows7을 사용하였고, 데이터베이스로는 MySQL



〈Figure 6〉 MAE of IBCF and CBCF according to Service Time

〈Table 7〉 Average, Standard Deviation and Range of MAE for IBCF, CBCF, ICFEC, and PCCF

	IBCF	CBCF	ICFEC	PCCF
Average	1.814	1.770	1.097	0.991
Standard Deviation	0.283	0.345	0.336	0.326
Range	1.099	1.292	1.209	1.142

〈Table 8〉 Gap of MAE between before and after of Clustering in CBCF, ICFEC, and PCCF

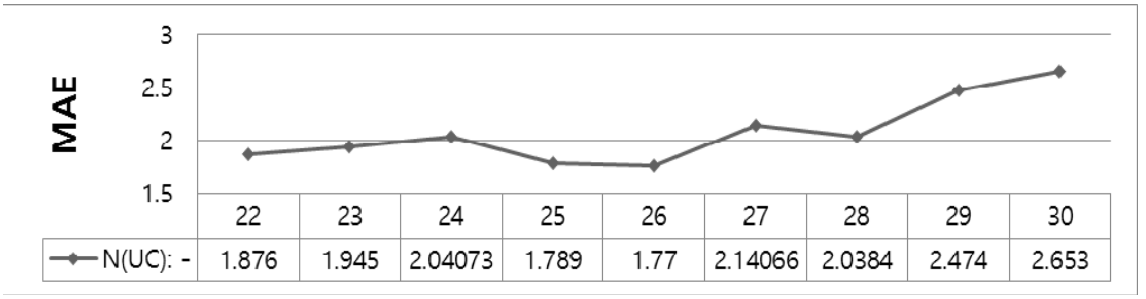
	CBCF	ICFEC	PCCF
1st weekend(6th,7th)	0.428	0.596	0.172
2nd weekend(13th,14th)	0.529	0.381	0.185
3rd weekend(20th,21th)	0.361	0.224	0.147
4th weekend(27th,28th)	0.470	0.484	0.319
Average	0.447	0.421	0.205

5.5를 사용하였다. 모든 알고리즘은 동일한 사양을 가진 같은 시스템 상에서 실험되었다. 또한, IBCF는 코사인 유사도를 이용해 구현하였으며, CBCF는 아이템을 코사인 유사도에 따라, K-NN을 이용해 군집화하여 유사 아이템 집합을 대신하는 방법으로 구현하였다. 군집화 기법을 사용하는 알고리즘들은 BIC를 이용하여 군집의 수를 결정하였으며, 군집화는 7일에 한번, 사용자들의 이용이 없을 때, 수행하는 것으로 가정하였다.

<Figure 6>과 <Table 7>, <Table 8>은 첫 번째 실험의 결과를 나타낸 그림과 표이다. <Figure 6>은 각 기법의 시간의 흐름에 따른 MAE를 표시한 그래프이며, <Table 7>은 각각의 기법의 MAE의 평균과 표준편차, 범위를 나타낸 표이며, <Table 8>은 군집화를 이용한 알고리즘들의 군집화 전후의 성능 변동 폭을 나타낸 표이다.

ICFEC와 PCCF는 기존의 두 기법, IBCF와 CBCF에 대해서는 큰 폭의 성능 향상을 보인다. 성능 향상의 폭은 ICFEC의 경우 CBCF에 비해 0.673(38.02%), PCCF의 경우 0.779(44.01%)를 보였다. 하지만, PCCF는 ICFEC에 비해 0.106(9.66%)의 개선을 보이는데 그쳤다. 이는 PCCF가 ICFEC에 비해 비교적 복잡한 구조를 가지고 있음에 볼 때, 미미한 성능의 향상이라 할 수 있다. 또한 표준편차의 경우에는 CBCF와 ICFEC, PCCF 모두가 IBCF 보다 높은 수치를 보였다. 또한, ICFEC와 PCCF는 CBCF에 비해 0.009(2.68%), 0.019(5.51%) 개선되어, 개선의 정도가 미미했다. PCCF와 ICFEC의 비교에서도 PCCF는 0.010(2.98%)의 근소한 개선에 그쳤다. 하지만, PCCF의 성능 불안정성에 대한 개선은 범위의 측면에서 보면 명확히 드러난다. CBCF와 ICFEC, PCCF 모두 여전히 IBCF에 비해 높은 범위를 보여주고 있다. 또한 ICFEC는 CBCF에 비해 0.083(6.42%)의 개선에 그쳤다. 하지만, PCCF는 CBCF와 ICFEC에 비하여 각각 0.150(11.61%), 0.067(5.54%)의 성능 향상을 보였다. 이는 PCCF가 CBCF의 성능 불안정성 문제를 다소 개선하였음을 보인다. PCCF의 성능 불안정성에 대한 개선은 <Table 8>에서 보다 명확히 드러난다. <Table 8>은 군집화 시점 전후에서 각 추천 기법들의 성능 변화를 나타낸 표이다.

이는 PCCF가 모델 생성의 어려움으로 인한 도메인 환경 변화의 누적에 대해 다른 기법들보다 영향을 적게 받음을 보인다. ICFEC의 경우 CBCF와 거의 유사한 폭의 성능 변화를 보이지만, PCCF의 경우에는 CBCF에 비해 0.242(54.14%), ICFEC에 비해 0.216(51.31%) 개선된 성능을 보인다. 이는 결과적으로 PCCF가 여타 기법들에 비해 추천의 성능에 대한 개선은 미미하지만, CBCF의 성능 불안정성 문제를 개선하고 있음을 알 수 있다.

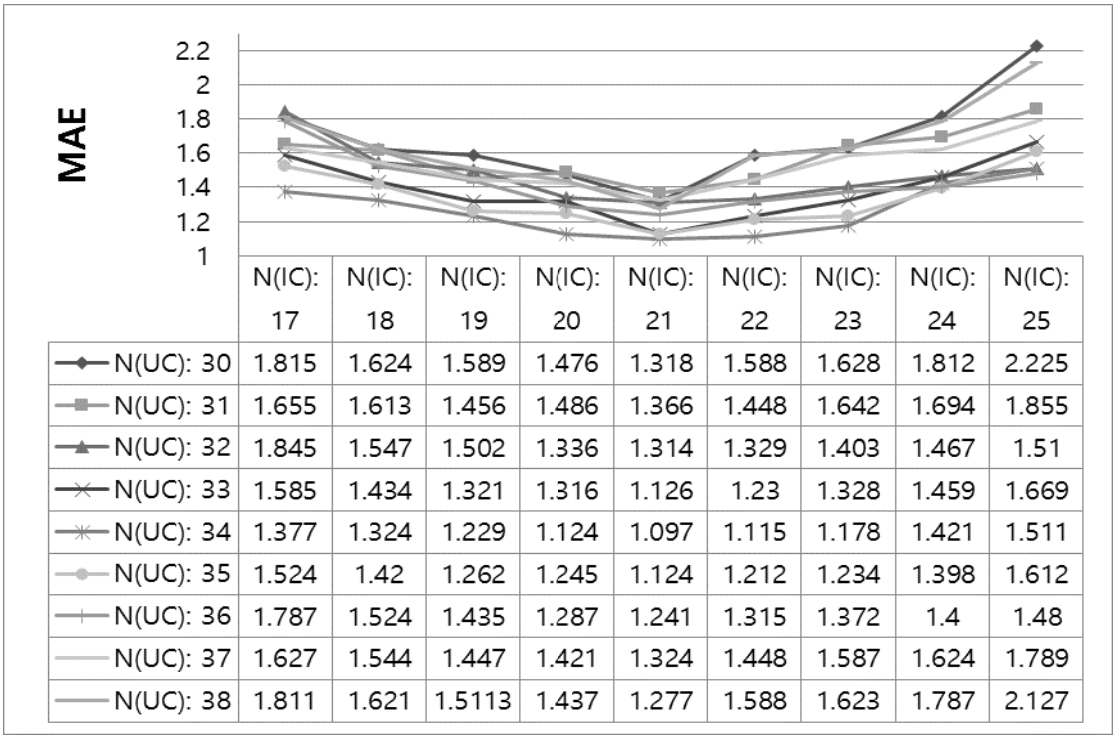


〈Figure 7〉 MAE of CBCF according to Number of Clusters

〈Table 9〉 Optimal Number of Clusters of CBCF, ICFEC, PCCF

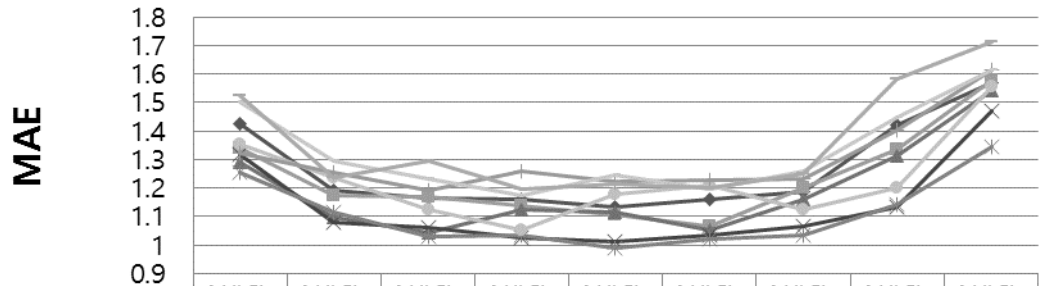
	CBCF	ICFEC	PCCF
Number of Item Clusters	26	21	37
Number of User Clusters	-	34	46

두 번째 실험은 각 기법에서 최적 군집의 수를 *k*라 할 때, *k*를 기준으로, 아이템 군집과 사용자 군집 모두에 대해 군집의 수가 *k-4*에서 *k+4*의 범위에서 MAE의 평균을 측정하였다. <Figure 7>부터 <Figure 9>와 <Table 9>, <Table 10>은 두 번째 실험의 결과를 나타낸 그래프와 표이다.



	N(IC): 17	N(IC): 18	N(IC): 19	N(IC): 20	N(IC): 21	N(IC): 22	N(IC): 23	N(IC): 24	N(IC): 25
N(UC): 30	1.815	1.624	1.589	1.476	1.318	1.588	1.628	1.812	2.225
N(UC): 31	1.655	1.613	1.456	1.486	1.366	1.448	1.642	1.694	1.855
N(UC): 32	1.845	1.547	1.502	1.336	1.314	1.329	1.403	1.467	1.51
N(UC): 33	1.585	1.434	1.321	1.316	1.126	1.23	1.328	1.459	1.669
N(UC): 34	1.377	1.324	1.229	1.124	1.097	1.115	1.178	1.421	1.511
N(UC): 35	1.524	1.42	1.262	1.245	1.124	1.212	1.234	1.398	1.612
N(UC): 36	1.787	1.524	1.435	1.287	1.241	1.315	1.372	1.4	1.48
N(UC): 37	1.627	1.544	1.447	1.421	1.324	1.448	1.587	1.624	1.789
N(UC): 38	1.811	1.621	1.5113	1.437	1.277	1.588	1.623	1.787	2.127

〈Figure 8〉 MAE of ICFEC according to Number of Clusters



	N(IC): 33	N(IC): 34	N(IC): 35	N(IC): 36	N(IC): 37	N(IC): 38	N(IC): 39	N(IC): 40	N(IC): 41
N(UC): 42	1.425	1.193	1.165	1.16	1.134	1.16	1.189	1.42	1.57
N(UC): 43	1.34	1.176	1.17	1.14	1.107	1.065	1.2	1.337	1.578
N(UC): 44	1.29	1.097	1.045	1.124	1.117	1.054	1.161	1.312	1.541
N(UC): 45	1.317	1.08	1.06	1.024	1.014	1.034	1.068	1.132	1.472
N(UC): 46	1.257	1.114	1.028	1.034	0.991	1.021	1.034	1.144	1.344
N(UC): 47	1.353	1.239	1.123	1.052	1.178	1.211	1.124	1.2	1.558
N(UC): 48	1.322	1.257	1.19	1.261	1.224	1.229	1.234	1.405	1.613
N(UC): 49	1.504	1.296	1.234	1.174	1.245	1.195	1.261	1.448	1.616
N(UC): 50	1.524	1.237	1.295	1.196	1.216	1.202	1.249	1.583	1.713

〈Figure 9〉 MAE of PCCF according to Number of Clusters

〈Table 10〉 Average, Standard Deviation and Range of MAE for CBCF, ICFEC, and PCCF

	CBCF	ICFEC	PCCF
Average	2.081	1.480	1.240
Standard Deviation	0.302	0.218	0.168
Range	0.883	1.129	0.722

<Figure 7>부터 <Figure 9>는 각 기법의 군집의 수의 변화에 따른 MAE의 변화를 표시한 그래프이며, <Table 9>는 각 기법의 최적의 군집의 개수를 나타낸 표이다. <Table 10>은 각의 기법의 MAE의 평균과 표준편차, 범위를 나타낸 표이다. <Figure 7>부터 <Figure 9>에서 그래프의 가로축은 아이템 군집의 개수이며, 범례의 $N(UC)$는 사용자 군집의 개수를 말한다.

군집의 수를 변화시키며 확장성과 성능간 트레이드오프를 시험한 2번째 실험에서, ICFEC와 PCCF는 모두 CBCF에 비해 개선된 성능을 보인다. 성능 측면에서 ICFEC는 CBCF에 비해 0.601(21.39%), PCCF는 0.841(40.41%) 개선되었다. 또한 PCCF는 ICFEC에 비해서도 0.240 (16.22%) 개선된 성능을 보인다. 표준편차 측면에서 ICFEC는 CBCF에 비해 0.084(27.81%), PCCF는 0.134(44.37)%의 개선되었다. PCCF와 ICFEEC의 비교에서도 PCCF는 0.050(22.94%) 개선된 성능을 보였다. 하지만 범위의 측면에서

ICFEC는 CBCF에 비해 오히려 0.246(27.86%) 증가하였다. 하지만, PCCF는 0.161(18.23%) 개선된 성능을 보인다. 이것은 ICFEC에 비해서는 0.407(36.05%) 개선된 수치이다. 이는 PCCF가 확장성과 성능간 트레이드오프 문제에 대해서도 강건성을 보이기 있음을 말한다.

본 연구에서 제안하는 기법의 검증을 위해 수행한 두 가지 실험은 각각 성능 불안정성 문제와 확장성과 성능의 트레이드오프 문제에 대한 강건성을 시험하기 위한 것이었다. PCCF는 기존 기법들에 비해 성능이 다소 향상되었지만, 이는 PCCF 기법의 복잡성에 비교해 볼 때, 의미 있는 성능 향상이라고 볼 수는 없다. 또한 PCCF는 데이터의 변동 정도를 나타내는 지표인 표준 편차의 측면에서도 의미 있는 개선을 보이지 못하였다. 하지만, 성능의 변동 폭을 나타내는 범위의 측면에서는 다른 기법들에 비해 개선을 보였다. 첫 번째 실험에서는 모델 생성 전후의 성능 변동폭에서 51.31%의 개선을, 두 번째 실험에서는 군집 수 변화에 따른 성능 변동폭에서 36.05%의 개선을 보였다. 이는 PCCF가 성능의 향상을 보여주지는 못하지만, 성능 안정성의 측면에서는 기존의 기법들을 개선하고 있음을 말한다.

5. 결론

본 연구에서는 모델 기반 CF 기법들의 공통적인 문제점인 모델 생성 고비용으로 인한 성능 불안정 문제와 성능과 확장성 간의 트레이드오프 문제로 인한 적용범위 감소 문제 해결을 위해 예측적 군집화 기반의 협업 필터링(PCCF)을 제안하였다. 이 기법은 CBCF에 마르코프 전이확률 모델과 퍼지 군집화의 개념을 접목하여 사용자의 기호 변화를 예측함으로써 도메인 환경의 변화와 모델간의 괴리를 축소하고 성능 불안정성 문제를 개선한다. 또한 전이확률모델을 기반으로 선호도 예측의 적용범위를 동적으로 변화할 수 있도록 하여 적용범위 감소 문제를 개선한다.

PCCF는 4개의 과정으로 구성된다. 첫째, 기호 군집화 과정을 통해 사용자들의 기호를 정규화하며, 둘째, 기호 전이 탐지 과정을 통해 사용자들의 평가점수 입력으로부터 사용자들의 기호 변화를 탐지한다. 셋째, 성향 군집화 과정을 통해 사용자들의 기호 변화 패턴(성향)를 통해 사용자들의 성향을 정규화한다. 넷째, 선호도 예측 과정을 통해 선호도 예측 모델을 만들고 사용자들의 아이템에 대한 선호도를 예측한다.

제안하는 기법의 검증은 각각 성능 불안정성 문제와 확장성/성능 트레이드오프 문제에 대한 강건성 실험을 통해 진행되었다. 첫 번째 실험은 데이터 희박성을 최소화한 환경에서 IBCF와 CBCF, ICFEC, PCCF를 이용해 구현된 각각의 추천 시스템들의 성능을 비교 분석하였다. 두 번째 실험은 CBCF, ICFEC, PCCF의 군집의 개수를 최적의 개수에서 가감하며 시스템의 성능 변화를 비교 분석하였다. 실험 결과, 제안하는 기법은 기존 기법들에 비해 성능의 향상 폭은 미미하였다. 또한 데이터의 변동 정도를 나타내는 지표인 표준 편차의 측면에서도 의미 있는 개선을 보이지 못하였다. 하지만, 성능의 변동 폭을 나타내는 범위의 측면에서는 기존 기법들에 비해 뚜렷한 개선을 보였다. 첫 번째 실험에서는 모델 생성 전후의 성능 변동폭에서 51.31%의 개선을, 두 번째 실험에서는 군집 수 변화에 따른 성능 변동폭에서 36.05%의 개선을 보였다. 이는 제안하는 기법이 성능의 향상을 보여주지는 못하지만, 성능 안정성의 측면에서는 기존의 기법들을 개선하고 있음을 의미하는 것이다.

본 연구의 향후 방향은 기존 기법들에 비해 뚜렷한 향상을 보이지 못한 추천 성능을 개선하는 것을 목적으로 나아갈 것이다. 기법의 복잡성에 비해 비교적 낮은 성능의 원인은 단순한 군집화 알고리즘의 사용으로 인해, 인접 군집간 경계 결정의 정확도가 낮았기 때문으로 판단된다. 따라서, 향후 연구에서는 고차원 비모수 군집화 알고리즘 혹은 딥러닝(Deep Learning) 기반 학습 모델을 도입하여, 추천의 성능을 개선하는 방향의 연구를 진행할 것이다.

참고문헌(References)

Ali, K. and W. Van Stam, "Tivo: Making show recommendations using a distributed collaborative filtering architecture," *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, (2004), 394~401.

Bellogin, A. and J. Parapar, "Using graph partitioning techniques for neighbor selection in user-based collaborative filtering," *Proceedings of the sixth ACM conference on Recommender systems*, ACM, (2012), 213~216.

Bennet, J. and S. Lanning, "The netflix prize," *Proceedings of KDD Cup and Workshop*, (2007). Available at http://www.netflixprize.com/ (Accessed 20 March, 2015).

Bhosale, N. S. and S. S. Pande. "A Survey on Recommendation System for Big Data Applications," *Data Mining and Knowledge Engineering*, Vol.7, No.1(2015), 42~44.

Bobadilla, J., F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowledge-Based Systems*, Vol. 46(2013), 109~132.

Cho, Y.-B., and Y. -H. Cho, "Considering Customer Buying Sequences to Enhance the Quality of Collaborative Filtering," *Journal of Intelligence and Information Systems,* Vol.13, No.2(2007), 69~80

Das, A. S., M. Datar, A. Garg, A., and S. Rajaram, "Google news personalization: Scalable online collaborative filtering," *Proceedings of the 16th international conference on World Wide Web*, ACM, (2003), 271~280.

George, T., and S. Merugu, "A scalable collaborative filtering framework based on co-clustering," *Proceedings of the Fifth IEEE International Conference on Data Mining*, IEEE, (2005), 4.

Gong, S., "A collaborative filtering recommendation algorithm based on user clustering and item clustering," *Journal of Software*, Vol.*5, No.*7 (2010), 745~752.

Hameed, M. A., O. A. Jadaan, and S. Ramachandram, "Collaborative Filtering Based Recommendation System: A survey," *International Journal on Computer Science & Engineering,* Vol. 4, No.5(2012).

Im, I. and B. H. Kim, "The Effect of the Personalized Settings for CF-Based Recommender Systems," *Journal of Intelligence and Information Systems,* Vol.18, No.2(2012), 131~141.

Joshi, R. C. and R. S. Paswan, "A Survey Paper on Clustering-based Collaborative Filtering Approach to Generate Recommendations," *International Journal of Science and Research*, Vol.4, No.1(2015), 1395~1398.

Khoshneshin, M. and W. N. Street, "Incremental collaborative filtering via evolutionary co-clustering," *Proceedings of the fourth ACM*

conference on Recommender systems, ACM, (2010), 325~328.

Lee, J., M. Sun, and G. Lebanon, "A comparative study of collaborative filtering algorithms," *arXiv preprint arXiv*:1205.3193, (2012), 1~27.

Lee, O. -J., M. -S. Hong, W. -j. Lee, and J. -D. Lee, "Scalable Collaborative Filtering Technique based on Adaptive Clustering ," *Journal of Intelligence and Information Systems*, Vol.20, No.2(2014), 73~92.

Lee, O. -J. and Y. -t. Baek, "Hybrid Preference Prediction Technique Using Weighting based Data Reliability for Collaborative Filtering Recommendation System," *Journal of the Korea Society of Computer and Information*, Vol.19, No.5 (2014), 61~69.

Renaud-Deputter, S., T. Xiong, and S. Wang, "Combining collaborative filtering and clustering for implicit recommender system," *Proceedings of 2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA),* IEEE, (2013), 748~755.

Li, Q. and Z. Dong, "Research of collaborative filtering algorithm based on the probabilistic clustering model," *Proceedings of 2010 5th International Conference on Computer Science and Education (ICCSE)*, IEEE, (2010), 380~ 383.

Li, X. and T. Murata, "Using multidimensional clustering based collaborative filtering approach improving recommendation diversity," *Proceedings of 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, IEEE, Vol. 3(2012), 169~174.

Linden, G., B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Computing*, (2003), 76 ~80.

Natarajan, N., D. Shin, and I. S. Dhillon, "Which app will you use next?: Collaborative filtering with interactional context," *Proceedings of the 7th ACM conference on Recommender systems*, ACM, (2013), 201~208.

Park, S. T. and D. M. Pennock, "Applying collaborative filtering techniques to movie search for better ranking and browsing," *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, (2007), 550~559.

Pham, M. C., Y. Cao, R. Klamma, and M. Jarke, "A Clustering Approach for Collaborative Filtering Recommendation Using Social Network Analysis," *J. UCS*, Vol.17, No.4 (2011), 583~604.

Su, X. and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in artificial intelligence*, (2009), 4.

Tseng, K. C., C. S. Hwang, and Y. C. Su, "Using Cloud Model for Default Voting in Collaborative Filtering," *Journal of Convergence Information Technology (JCIT)* Vol.6, No.12 (2011), 68~74

Wen, J. and W. Zhou, "An improved item-based collaborative filtering algorithm based on clustering method," *Journal of Computational Information Systems*, Vol.8, No.2(2012), 571~ 578.

Zhirao, J., "Based on Java Technology System and Implement the Personalized Recommendations of the system," Jilin: Jilin University, 2011.

Zhou, Z., M. Sellami, W. Gaaloul, M. Barhamgi, and B. Defude, "Data providing services

clustering and management for facilitating service discovery and replacement," *IEEE Transactions on Automation Science and Engineering*, Vol. 10, No. 4(2013), 1131~1146.

Abstract

Predictive Clustering-based Collaborative Filtering Technique for Performance-Stability of Recommendation System

O-Joun Lee* · Eun-Soon You**

With the explosive growth in the volume of information, Internet users are experiencing considerable difficulties in obtaining necessary information online. Against this backdrop, ever-greater importance is being placed on a recommender system that provides information catered to user preferences and tastes in an attempt to address issues associated with information overload. To this end, a number of techniques have been proposed, including content-based filtering (CBF), demographic filtering (DF) and collaborative filtering (CF). Among them, CBF and DF require external information and thus cannot be applied to a variety of domains. CF, on the other hand, is widely used since it is relatively free from the domain constraint.

The CF technique is broadly classified into memory-based CF, model-based CF and hybrid CF. Model-based CF addresses the drawbacks of CF by considering the Bayesian model, clustering model or dependency network model. This filtering technique not only improves the sparsity and scalability issues but also boosts predictive performance. However, it involves expensive model-building and results in a tradeoff between performance and scalability. Such tradeoff is attributed to reduced coverage, which is a type of sparsity issues. In addition, expensive model-building may lead to performance instability since changes in the domain environment cannot be immediately incorporated into the model due to high costs involved. Cumulative changes in the domain environment that have failed to be reflected eventually undermine system performance.

This study incorporates the Markov model of transition probabilities and the concept of fuzzy clustering with CBCF to propose predictive clustering-based CF (PCCF) that solves the issues of reduced coverage and of unstable performance. The method improves performance instability by tracking the

* School of Computer Engineering, Chung-Ang University
E-mail : concerto9203@gmail.com
** Corresponding author: Eun-Soon You
Institute of Media Content, Dankook University
Tel: +82-31-8005-2387, Fax: +82-31-8021-7422, E-mail : tesniere@naver.com

changes in user preferences and bridging the gap between the static model and dynamic users. Furthermore, the issue of reduced coverage also improves by expanding the coverage based on transition probabilities and clustering probabilities.

The proposed method consists of four processes. First, user preferences are normalized in preference clustering. Second, changes in user preferences are detected from review score entries during preference transition detection. Third, user propensities are normalized using patterns of changes (propensities) in user preferences in propensity clustering. Lastly, the preference prediction model is developed to predict user preferences for items during preference prediction.

The proposed method has been validated by testing the robustness of performance instability and scalability-performance tradeoff. The initial test compared and analyzed the performance of individual recommender systems each enabled by IBCF, CBCF, ICFEC and PCCF under an environment where data sparsity had been minimized. The following test adjusted the optimal number of clusters in CBCF, ICFEC and PCCF for a comparative analysis of subsequent changes in the system performance. The test results revealed that the suggested method produced insignificant improvement in performance in comparison with the existing techniques. In addition, it failed to achieve significant improvement in the standard deviation that indicates the degree of data fluctuation. Notwithstanding, it resulted in marked improvement over the existing techniques in terms of range that indicates the level of performance fluctuation. The level of performance fluctuation before and after the model generation improved by 51.31% in the initial test. Then in the following test, there has been 36.05% improvement in the level of performance fluctuation driven by the changes in the number of clusters. This signifies that the proposed method, despite the slight performance improvement, clearly offers better performance stability compared to the existing techniques.

Further research on this study will be directed toward enhancing the recommendation performance that failed to demonstrate significant improvement over the existing techniques. The future research will consider the introduction of a high-dimensional parameter-free clustering algorithm or deep learning-based model in order to improve performance in recommendations.

Key Words : Recommendation System, Adaptive System, Collaborative Filtering, Hybrid Filtering, Clustering

Received : February 10, 2015 Revised : March 2, 2015 Accepted : March 20, 2015
Type of Submission : Fast Track Corresponding Author : Eun-Soon You

저 자 소 개

이 오 준

단국대학교 소프트웨어학과를 2015년에 졸업하고(공학사), 현재 중앙대학교 컴퓨터공학과 석박사 통합과정에 재학 중이다. 연구 관심 분야는 적응형 시스템, 개인화 맞춤형 시스템, 추천 시스템, 데이터 마이닝, 비즈니스 인텔리젼스 등이다.

유 은 순

인하대학교 불어불문학과를 졸업하였고, 2007년에 프랑스 Franche-Comté 대학교에서 언어학 박사학위를 취득하였다. 2012년부터 현재까지 단국대학교 미디어콘텐츠연구원 리서치 펠로우로 재직 중이다. 연구 관심분야는 빅데이터, 소셜미디어, 스토리텔링 등이다.