JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 26 No. 11, pp. 149-156, November 2021
https://doi.org/10.9708/jksci.2021.26.11.149

A New Similarity Measure based on Separation of Common Ratings for Collaborative Filtering

Soojung Lee*

*Professor, Dept. of Computer Education, Gyeongin National University of Education, Anyang, Korea

[Abstract]

Among various implementation techniques of recommender systems, collaborative filtering selects nearest neighbors with high similarity based on past rating history, recommends products preferred by them, and has been successfully utilized by many commercial sites. Accurate estimation of similarity is an important factor that determines performance of the system. Various similarity measures have been developed, which are mostly based on integrating traditional similarity measures and several indices already developed. This study suggests a similarity measure of a novel approach. It separates the common rating area between two users by the magnitude of ratings, estimates similarity for each subarea, and integrates them with weights. This enables identifying similar subareas and reflecting it onto a final similarity value. Performance evaluation using two open datasets is conducted, resulting in that the proposed outperforms the previous one in terms of prediction accuracy, rank accuracy, and mean average precision especially with the dense dataset. The proposed similarity measure is expected to be utilized in various commercial systems for recommending products more suited to user preference.

▸ **Key words**: Similarity Measure, Collaborative Filtering, Recommender System, Nearest Neighbor

[요 약]

추천 시스템의 여러 구현 기법들 중 협력 필터링은 과거 평가 이력을 토대로 유사성이 높은 인접 이웃들을 선정하여, 그들이 선호했던 상품들을 추천하는데, 많은 상업 사이트에서 성공적으로 활용되고 있다. 유사도의 정확한 측정은 시스템의 성능을 좌우하는 매우 중요한 요소이다. 기존에 다양한 방식의 유사도 척도들이 개발되었는데, 대개 전통적인 유사도 척도와 기개발된 여러 계수들과의 통합 방식이었다. 본 연구에서는 새로운 방식의 유사도 척도를 제안한다. 두 사용자 간의 공통 평가 영역을 평가치 크기에 따라 분할하여 각 부분 영역별로 유사도를 측정하고 이들을 가중 통합함으로써, 유사한 영역이 구체적으로 파악되어 최종 유사도값에 반영된다. 두 종류의 개방형 데이터셋을 활용한 성능을 측정하였고, 그 결과 특히 밀집 데이터셋에서 제안 방법의 예측 정확도, 순위 정확도, 평균 정밀도 성능이 기존보다 우수하였다. 제안 척도는 다양한 상업 시스템에서 사용자들의 선호에 보다 적합한 상품을 추천하는데 유용하게 활용될 것으로 기대한다.

▸ **주제어**: 유사도 척도, 협력 필터링, 추천 시스템, 인접 이웃

• First Author: Soojung Lee, Corresponding Author: Soojung Lee
*Soojung Lee (sjlee@gin.ac.kr), Dept. of Computer Education, Gyeongin National University of Education
• Received: 2021. 09. 09, Revised: 2021. 10. 26, Accepted: 2021. 11. 12.

Copyright © 2021 The Korea Society of Computer and Information
http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

협력 필터링(collaborative filtering, CF) 기반의 추천 알고리즘은 지난 수십년간 학계의 많은 연구가 이루어진 대표적인 추천 기법들 중 하나이다. 사용자가 선호할 만한 상품을 알아내기 위하여, 유사한 평가 이력의 다른 사용자들을 참조하여 그들이 선호하였던 상품들을 추천한다. 이러한 방식은 사용자들 간의 유사한 항목 선호 경향이 미래에도 지속될 것이라는 가정을 기반으로 한다.

CF 방법은 평가 이력의 유사성 여부를 판단하는 일이 매우 중요하고, 또한 그러한 판단을 가능하게 하는 충분한 량의 평가 데이터가 존재해야 한다. 전자를 위하여, 다양한 유사도 척도가 개발되었다[1-2]. 전통적인 대표적 방법으로서 상관도 기반의 피어슨 상관(Pearson correlation), 스피어만 순위 상관(Spearman rank correlation), 켄델 타우 상관(Kendall's tau correlation) 등이 포함되며, 벡터 기반의 코사인 유사도(cosine similarity), 조정된 코사인 유사도(adjusted cosine similarity) 등이 포함된다[1].

추천 시스템은 대개 매우 많은 상품 정보를 유지 관리하고, 사용자들은 극히 일부 상품에 대해서 평가하기 때문에, 전체 평가 데이터는 매우 희박한 실정이다. 이러한 데이터 희소성 문제(data sparsity problem)는 타당한 유사도값을 산출하기 어렵게 한다. 따라서 데이터의 희소 정도를 반영한 새로운 유사도 척도들이 개발되었다[3-4]. 이 밖에도 새로운 사용자, 새로운 항목이 추가되면 관련 평가 이력의 부재로 인하여 이들에 대한 추천 리스트의 생성 문제가 발생한다(new user problem, new item problem, cold start problem). 이러한 모든 문제들은 시스템의 추천 성능을 저하시키는 주요 요인이다.

위에서 기술한 내용 이외에도 발생 가능한 또다른 문제는 데이터 확장성 문제(data scalability problem)로서, 시스템 사용자나 항목이 급증하게 되면, 사용자의 추천 리스트 산출 요구에 대한 실시간적 처리에 과부하가 발생한다. 이 문제에 대한 해결 방안으로서 특이성 분해(singular value decomposition)와 같은 차원 감소 기법, 주성분 분석(principle component analysis) 등의 요인 분석 기법을 포함할 수 있으나, 주요한 특성 정보가 손실될 가능성이 있다고 보고되었다[1].

기존 연구의 다양한 유사도 척도들은 기본적으로 두 사용자가 공통으로 평가한 항목들의 평가치 전체를 기반으로 유사도를 산출하였다. 본 연구에서는 기존과는 다른 새로운 차원의 접근 방식으로서, 공통 평가 항목들을 평가치 크기에 따라 분리하여 각 영역별로 유사도를 산출한 후 통합하는 방안을 제안한다. 즉, 고평가치가 부여된 공통항목들에 대한 두 사용자의 유사성과 저평가된 공통항목들에 대한 유사성을 구분하여 측정한다. 이같은 방식은 영역을 세분화하여 파악함으로써 보다 정확한 유사도값의 산출에 기여할 것으로 기대되며, 따라서 시스템의 예측 정확도나 추천 정확도 등의 성능을 향상시킬 수 있다. 두 종류의 개방형 데이터셋을 활용한 실험 결과, 제안 방법은 여러 성능 평가 척도에 있어서, 기존 방법을 능가함을 확인하였다.

논문의 구성은 다음과 같다. 2절에서는 문헌에 소개된 다양한 유사도 척도를 소개한다. 3절에서는 제안 방법을 설명하고 4절에서 성능 실험 결과를 제시하며, 5절에서 논문의 결론을 맺는다.

II. Related Works

유사한 인접 이웃을 구하기 위한 유사도 산출 과정에서 근본적으로 내재된 문제들 중 하나인 평가 데이터 희소성 문제를 해결하기 위한 다양한 시도들이 이루어져 왔다. 매우 단순하면서도 효과적인 방법으로써 자카드 계수(Jaccard index)가 개발되었는데, 이 계수는 두 사용자가 평가한 모든 항목들 중에서 공통으로 평가한 항목수의 비율을 말한다[3]. 기존에 개발된 유사도 척도와 자카드 계수를 접목하여 유사도를 산출하는 방법이 관련 연구의 주요 방향이라고 할 수 있는데[4-5], 이러한 방식들은 주로 희소 데이터 환경에서 기존의 성능을 향상시킨다고 보고되었다[5-6].

자카드 계수는 평균자승차이(mean squared differences, MSD), 피어슨 상관도 등 여러 종류의 유사도 척도들과 결합[3,5,7]되어 그 성능이 분석되었다. Zhu 외 3인은 자카드 계수 이외에 비수치적 평가치 구조를 추가 접목하는 새로운 유사도 척도를 개발하였다[8]. 자카드 계수와 유사한 개념으로서 공통평가항목을 고려한 지수인 트라이앵글 유사도(triangle similarity)도 활용되었는데, Iftikhar 외 4인은 이 지수의 단점을 극복하고자 평가치의 평균과 분산 정보를 반영하여 유사도를 산출하였다[9]. Sun 외 6인은 트라이앵글 유사도와 자카드 계수를 통합하여 예측 오류를 개선하려는 연구 결과를 발표하였다[4]. 이 밖에도 Mu 외 4인은 데이터 희소 문제를 해결하기 위한 방법으로, 헬링거 거리(Hellinger Distance)와 자카드 계수를 공통 피어슨 상관계수(Common Pearson Correlation Coefficient, COPC)와 결합함으로써 공통항목 평가개수의 영향을 낮추고 피어슨 상관도의 단점을 극

복하려고 시도하였다[10]. 자카드 계수는 평가치의 크기를 고려하지 않은 채 공통평가항목 개수의 상대적 비율만을 반영하므로, 이를 보완하여 평가치 범주에 따라 계수를 별도 정의하고 유전자 알고리즘을 활용하여 통합함으로써 성능 개선을 이룬 연구 결과도 발표되었다[11].

이와 같이 새롭게 개발된 유사도 척도들 대부분이 평가치 개수의 한계 및 전통적인 척도들의 단점 등을 극복하려는 시도의 결과이다. Al-Shamri는 자카드와 다이스 계수(Dice coefficient)를 일반화하여 Power 계수를 정의하고 이를 유사도 척도로 활용하였다[12]. Chen 외 2인은 두 사용자의 평가치 범위의 차이를 해결하기 위한 방법을 제시하였고, 이를 피어슨 상관도와 조정 코사인 유사도와 접목한 새로운 유사도 척도를 제안하였다[13]. Bag 외 2인은 자카드 계수를 다소 변형 및 개선하여 연관 자카드 유사도(relevant Jaccard similarity)를 정의하고 제안 방법이 F1 성능 측면에서 우수하다고 밝혔다[3]. 또한 Wang 외 2인은 평가치 기반의 유사도와 구조 기반의 유사도를 통합하였고 이 방법이 예측 성능 면에서 모델 기반의 추천 방법보다 우수함을 보였다[14].

이와 같이 여러 가지 특성의 계수를 통합하는 방식 외에 평가치의 확률 분포도 고려한 연구가 발표되었는데, Jain 외 2인은 K-L divergence 대신에 Bhattacharyya 계수로 대체하여 다목적 적합용의 유사도 척도를 개발하고 이를 활용한 항목 추천 방법을 제시하였다[15].

한편 다양한 방법이나 계수들을 효율적으로 통합하기 위한 노력이 진행되었는데, 대개 기존 진화 알고리즘을 통한 최적화 기법을 이용하였다. [16]에서는 사용자 간의 유사도를 유전자 알고리즘(genetic algorithm, GA)으로 정제하여 예측 평가치의 정확도를 높이는 연구를 진행하였고, 코사인 유사도에 방법을 적용하였을 때 가장 좋은 성능을 보인다고 하였다. [17]에서도 유전자 알고리즘을 활용하여 온톨로지의 속성의 중요도를 알아내어 추천 시스템에 사용하였다. [15]의 연구에서도 GA를 사용하여 항목을 추천하였는데, 새로운 크로스오버 기법을 제안하였다. 한편 예측 정확도와 추천 항목의 다양성을 모두 만족시키고자 GA를 활용한 연구도 발표되었다[18]. GA 뿐만 아니라 다른 종류의 진화 알고리즘도 추천 시스템에 활용되어 왔다. [19]에서는 반딧불 알고리즘(Firefly Algorithm)과 Cuckoo 알고리즘을 사용하여 클러스터링을 통한 추천 시스템을 제안하였다. [20]에서는 뻐꾸기 최적화 알고리즘(Cuckoo Optimization Algorithm)으로 두 사용자의 평가치 간에 최적화된 가중치를 구하여 유사도를 측정하는 방안을 개발하였다.

본 연구에서는 기존 연구에서와 같이 여러 척도들을 통합하거나 최적화 기법을 사용하는 것이 아닌 새로운 패러다임의 유사도 산출 방법을 제안한다. 사용자 평가치의 크기 영역별로 유사도 척도를 적용하여, 각 영역의 유사도값을 산출한 후에 이를 통합하는 방식으로서, 두 사용자 평가치의 유사한 영역을 보다 세밀히 파악 가능하다.

III. Proposed Methodology

1. Motivation

전통적인 유사도 척도에서는 두 사용자가 공통으로 평가한 항목들에 대한 평가치를 기준으로 유사도를 산출한다. 표 1은 임의의 세 명의 사용자들과 일곱 개의 항목들에 대한 사용자-항목 평가 매트릭스(user-item rating matrix)의 예시이다. 평가치의 허용 범위를 1부터 5로 가정하였을 때, 현 사용자 u는 i1부터 i3까지의 항목에 대해서는 높은 평가를, 나머지 항목들에 대해서는 낮은 평가치를 부여한 것을 알 수 있다.

설명의 편의 상, 모든 일곱 항목들을 세 사용자의 공통항목으로 간주하였다. 표 1에서는 평균자승차이(mean squared differences, MSD)에 따라 유사도 산출의 예를 나타냈다. MSD를 구하기 위해 각 평가치의 정규화를 우선 실시한다. 사용자 u의 평가치 r_u에 대한 정규화 값은 $(r_u-1)/(5-1)$로 계산하며, 이를 r_u'로 표기하였다. 사용자 v와 w에 대해서도 마찬가지로 정규화 값을 구하였다.

표 1에서 사용자 u와 v 간의 MSD값, 즉, MSD(u,v)는 $1-\sum(r_u'-r_v')^2/7 = 1-0.3125/7 \approx 0.9554$ 가 되며 MSD(u,w)도 동일한 값으로 산출된다. 두 사용자 쌍의 유사도값은 동일하지만, 이들 쌍의 평가치엔 주목할 만한 특성이 존재한다. 구체적으로, u와 v는 항목 i1~i3에 대해 상대적으로 매우 유사한 평가치를 부여하였는데, 이들 항목들은 u가 다른 항목들에 비해 4 이상의 높은 평가치를 부여한 항목들이다. 반면에 u와 w 간에는 각 항목에 대한 평가치 차이가 u의 평가치의 크기에 상관없이 고르게 분포되어 있다. 차이를 한눈에 알아보기 쉽도록 $|r_u-r_v|$, $|r_u-r_w|$ 행을 표에 포함하였다.

본 연구의 아이디어는 위 예시에 기반하여 개발되었다. 구체적으로, 임의의 두 사용자 간 평가치 차이가 전체 공통항목들 중에서 어느 부분에 집중되어 있는지 또는 어느 부분에서 차이가 거의 없는지를 고려한다. 표 1에서, u와 v가 높은 평가치를 부여하는 공통항목집합이 i1부터 i3까지 동일하므로, 만약 u가 또 다른 항목 x에 대해 높은 평

가치를 부여한다면, w 보다는 v가 x에 대해 높은 평가치를 부여할 가능성이 더 클 것으로 예상된다. 이를 MSD 유사도 산출에 반영하며, 따라서 MSD(u,v)는 MSD(u,w) 보다 큰 값이 산출되게 한다.

Table 1. Illustration of user-item matrix and similarity calculation

	i1	i2	i3	i4	i5	i6	i7
r_u	4	4	5	2	1	1	2
r_v	5	4	5	3	2	2	1
r_w	3	4	4	1	2	1	1
r_u'	0.75	0.75	1	0.25	0	0	0.25
r_v'	1	0.75	1	0.5	0.25	0.25	0
r_w'	0.5	0.75	0.75	0	0.25	0	0
MSD(u,v)	0.9554						
MSD(u,w)	0.9554						
$\|r_u-r_v\|$	1	0	0	1	1	1	1
$\|r_u-r_w\|$	1	0	1	1	1	0	1

2. Formulation

본 절에서는 두 사용자 간의 새로운 유사도 산출 방법을 소개한다. 제안 방법은 MSD에 기반하여 각 공통평가항목에 대한 두 사용자의 평가치 차이를 반영한다. 단, 고평가치가 부여된 항목들과 저평가 항목들을 나누어 유사도 척도를 각기 적용한 후 가중 통합하는 방식을 취한다.

시스템의 모든 항목 집합을 I, 사용자 u의 항목 i에 대한 평가치를 $r_{u,i}$, 항목에 대한 미평가치를 '-'라고 표시할 때, 공통평가항목 집합의 정의는 다음과 같다.

$$I(u,v)=\{i\in I \mid r_{u,i}\neq -,\ r_{v,i}\neq -\}$$

현 사용자의 평가치 관점에서 고평가와 저평가 공통항목집합은 기준값 θ에 따라 다음과 같이 정의한다.

$$I_H(u,v)=\{i\in I(u,v) \mid r_{u,i}\geq \theta\}$$

$$I_L(u,v)=\{i\in I(u,v) \mid r_{u,i}< \theta\}$$

위 두 가지 항목 집합에 대하여 각각 MSD 값을 구하여 통합함으로써 최종적인 제안 척도 MSDPT(MSD ParTition)를 다음과 같이 산출한다.

$$MSD_H(u,v)=1-\frac{1}{|I_H(u,v)|}\sum_{i\in I_H(u,v)}(r'_{u,i}-r'_{v,i})^2$$

$$MSD_L(u,v)=1-\frac{1}{|I_L(u,v)|}\sum_{i\in I_L(u,v)}(r'_{u,i}-r'_{v,i})^2$$

$$MSDPT(u,v)=MSD_H(u,v)\cdot\ w+ MSD_L(u,v)\cdot\ (1-w),\ 0<w<1$$

위에서 기준값 θ와 가중치 w는 시스템 파라미터로서 실험에 의해 결정한다.

3. Example

표 2는 제안한 유사도 척도 방법의 산출 예이다. θ 값은 4로 하였고, 다양한 w 값을 적용하였다. 앞 절의 각 공식을 적용한 과정을 일부 나열하면 다음과 같다.

$$I_H(u,v)=\{i1,i2,i3\}$$

$$I_L(u,v)=\{i4,i5,i6,i7\}$$

$$MSD_H(u,v)=1-\frac{1}{3}((0.75-0.5)^2+(0.75-0.25)^2+(1-1)^2)\approx\ 0.8958$$

전체 공통항목에 대한 MSD는 0.9375이지만, I_H 집합에 대한 유사도 값은 그보다 작고, I_L 집합에 대하여는 크다. 따라서, w값이 작을 때는 최종 유사도값에 대한 MSD_H의 영향이 작으므로 MSDPT 결과는 커지게 되며, 큰 w값을 사용하면 MSDPT는 원래의 MSD 보다 작게 된다.

Table 2. Example of MSDPT calculation

	i1	i2	i3	i4	i5	i6	i7
r_u	4	4	5	2	1	1	3
r_v	3	2	5	1	1	1	2
r_u′	0.75	0.75	1	0.25	0	0	0.5
r_v′	0.5	0.25	1	0	0	0	0.25
MSD	0.9375						
MSD_H	0.8958						
MSD_L				0.9688			
MSDPT (w=0.1)	0.9615						
MSDPT (w=0.3)	0.9469						
MSDPT (w=0.5)	0.9323						
MSDPT (w=0.7)	0.9177						
MSDPT (w=0.9)	0.9031						

IV. Performance Experiments

1. Design of Experiments

본 연구에서는 협력 필터링 기반의 추천 시스템의 성능을 점검하기 위하여 연구용 목적으로 널리 활용되는 오픈 데이터셋들 중 대표적인 MovieLens와 Jester를 이용하여 실험하였다. 두 셋은 표 3에서 제시한 바와 같이 서로 매우 다른 특성을 가진다. 따라서 제안 방법의 성능을 객관적으로 평가하기에 적합하다고 판단된다. 전체 데이터 중 80%는 유사도 척도를 활용하여 유사한 인접 사용자들을 구하기 위한 훈련 데이터로 사용하였고, 나머지 20%는 훈련의 결과로서 시스템의 성능을 다각도로 평가하기 위한 목적으로 사용하였다.

Table 3. Description of datasets

	MovieLens	Jester
# users	3000	3000
# items	3952	100
rating range	1 ~ 5 (integer type)	-10.0~+10.0 (real type)
total number of ratings	744,072	213,037
sparsity level	0.9372	0.2899
number of ratings per user	≥20	≥36

2. Performance Metrics

성능 측정을 위하여 CF 알고리즘에서 사용하는 대표적인 척도들을 통해 예측 정확도, 순위 정확도, 추천 정확도를 살펴보았다. 예측 정확도는 주로 평균절대오차(Mean Absolute Error, MAE)로 측정하는데, 이는 각 사용자의 실제 평가치와 시스템의 예측 평가치의 차이의 평균으로 정의한다. 예측 평가치는 유사도가 상대적으로 큰 인접이웃들의 해당 항목에 대한 평가치의 가중 합[1]으로 구하였다. 실험에 사용한 두 데이터셋의 평가치 범위가 다르므로, 본 연구에서는 정규화된 MAE, 즉 NMAE(Normalized MAE)를 활용한다.

추천 정확도를 알기 위하여 평균 정밀도(Mean Average Precision, MAP)를 도입하였는데, 이는 각 사용자를 위한 추천 리스트에서 각 순위에서의 재현율 대비 정밀도를 계산하여 전체 사용자에 대한 평균값을 구한 것이다[1,5]. 순위 정확도는 시스템에서 제공한 추천 항목 리스트의 각 항목 순위가 실제 사용자가 부여한 평가치 순위에 얼마나 부합되는지를 말한다. 측정을 위하여 관련 연구에서 널리 사용되는 nDCG((normalized Discounted Cumulative Gain)[1,21]를 도입하였다. 각 성능 지표의 구체적인 공식은 표 4에 제시하였다.

Table 4. Definition of performance metrics

Metric	Formula
NMAE	$\frac{1}{n}\sum_u \sum_i \lvert r_{u,i} - \tilde{r}_{u,i} \rvert / (\max rating - \min rating)$
MAP	$\frac{1}{\lvert U \rvert}\sum_u \frac{1}{n}\sum_k P@k \cdot rel(k)$ n: number of relevant items for u P@k: precision at rank k in the list rel(k): 1 if the item at rank k is relevant, 0 otherwise.
nDCG	$\frac{DCG}{IDCG}$ $DCG = \frac{1}{\lvert U \rvert}\sum_u \sum_k \frac{2^{r(k)}}{\log(k+1)}$ r(k): real rating of item at rank k IDCG: ideal DCG

성능 실험 대상은 원 유사도인 MSD와 MSDPT를 비교하였는데, 다양한 값의 w 가중치에 대하여 제안 방법의 성능을 측정하였다. 공통항목집합을 분리하는 기준값 θ을 설정하는데 있어서, 이는 고평가치의 판단 기준이므로 허용 평가데이타 범위의 중간값보다 상위 값으로 결정하는 것이 바람직하다. 따라서, MovieLens에서는 중간값인 3보다 큰 4.0으로 정하였고, 이는 전체 데이터 범위의 75% 선이므로, Jester 셋에서도 이와 유사한 경계선에 해당하는 값인 4.0으로 정하였다.

3. Performance Results using MovieLens

그림 1은 MovieLens 데이터셋을 사용한 실험 결과이다. 평가치를 참조할 인접 이웃수를 늘릴수록 평균절대오차 값은 가중치와 무관하게 점차 안정화됨을 확인할 수 있다. 가중치 변화에 따라 성능 결과는 확연히 차이를 보이는데, 고평가된 공통평가항목들에 대한 가중치가 가장 작은 경우, 즉, w=0.1인 경우에 가장 성능이 저조하였다. 반면에 가장 큰 가중치인 w=0.9일 때도 마찬가지로 저조한 성능을 보였다. 이는 고평가 또는 저평가 항목들의 비중을 매우 작게 할 경우에 타당한 인접 이웃들의 확보는 어렵다는 것을 의미한다. 그러나, 유사도 산출에 있어서 고평가 항목들의 비중을 어느 정도 크게 한 경우(w=0.7), 가장 좋은 MAE 성능을 보였다.

위에서 언급한 현상은 MAP 결과에서도 확인되었다. 평가치가 4 이상이면 사용자가 선호하는 항목으로 간주하였다. 실험 결과 극단값의 가중치에 대해 가장 저조한 성능을 보였고, 그 밖의 가중치에 대해서 MSD 보다 대체로 좋은 결과를 나타냈다. 다만 MAE에서와는 다르게 w=0.3인 경우에 w=0.7과 거의 대등한 성능 결과를 보였다.

순위 성능을 측정한 결과에서도 위와 마찬가지로 w=0.7과 0.3일 때 대체로 가장 좋은 성능을 보였는데, 주목할만한 점은 w=0.5일 때도 거의 대등한 우수한 성능을 보인 것이다. 반면에 MSD 결과는 다른 성능 척도 결과에서보다 상대적으로 저조하여 w=0.1의 경우와 거의 유사하였다. 따라서, 제안 방법을 사용한 추천 리스트의 결과가 순위 성능 면에서 더욱 우수함을 알 수 있다.

4. Performance Results using Jester

그림 2는 Jester 데이터셋을 활용한 성능 결과를 나타낸다. 평균절대오차의 성능은 MovieLens를 활용했을 때보다 전반적으로 더 우수한 수치를 보인다. 그 이유는 Jester 셋은 매우 밀집한 평가 데이터를 포함하므로, 유사도 산출

의 정확성이 더 뛰어나기 때문이다. 가중치 변화에 따라 성능 차이는 뚜렷이 드러나는데, 특히 w=0.9에 대하여 가장 저조한 성능을 보였다. 반면에 w=0.1일 때 이보다는 더 좋은 성능을 보였다. w=0.1은 저평가 공통항목에 대한 비중을 고평가 공통항목에 대한 비중 보다 9배로 크게 한 것이므로, 저평가 공통항목에 대한 두 사용자의 평가치 차이를 더욱 중요시할 때 더 우수한 성능을 가져온다는 의미이다. 결론적으로, w=0.5인 경우에 기존의 MSD를 넘어서는 가장 좋은 결과를 보였다.

그림 2에서 평균 정밀도를 측정한 결과도 MAE 결과와 거의 유사하여, 극단의 w 값에 대하여 가장 저조한 성능을 보였고, w=0.5일 때 가장 우수하였다. 추천 항목에 대한 사용자의 적합성 판정 기준 평가치는 4.0으로 하였다. 정밀도가 높다는 것은 시스템이 추천한 항목 리스트 내에 사용자에게 적합한 항목들이 많다는 것을 의미한다. 순위 성능 결과에서도 위와 비슷한 양상을 보였는데, 다만 평균 정밀도 결과에서보다 각 방법들 간의 성능 차이가 다소 적게 나타났다. 결론적으로, 저평가와 고평가 공통항목들에 대해 각기 별도의 유사도값을 산출한 후 동일한 가중치를 부여하여 통합하는 제안 방법은 고밀도 데이터셋에서 가장 우수한 성능을 보인 것을 알 수 있다.

V. Conclusions

협력 필터링 기반의 추천 시스템에서는 현 사용자가 선호할 만한 상품을 알아내기 위하여 이웃 사용자들의 상품 선호 이력을 참조한다. 이들이 선호하였던 상품들을 종합적으로 판단하여 추천하기 위함이다. 따라서 어떠한 이웃을 선정하는가는 시스템의 성능에 매우 중요한 일이고, 이를 위하여 다양한 유사도 척도가 개발되어 현 사용자와 평가 이력 측면에서 유사도가 가장 큰 이웃들을 선정하는데 사용되어 왔다.

본 연구에서는 기존의 개발 관점과는 다른 새로운 접근 방식의 유사도 척도를 제안하였다. 두 사용자 간의 평가치 차이가 전체 공통항목들 중에서 어떠한 크기 영역에서 발생하는지를 고려하였다. 즉, 고평가 공통항목들과 저평가 공통항목들에 대한 유사성을 구분하여 측정한 후 이를 가중 통합하는 방식이다. 제안 방법은 두 종류의 개방형 데이터셋을 통하여 성능 평가하였으며, 그 결과 공통항목 평가치를 구분하지 않는 기존 방식에 비해 우수한 예측 정확도, 순위 정확도, 평균 정밀도를 보였으며, 이는 밀집 데이터셋에서 더 확실히 증명되었다.

본 연구 결과는 협력 필터링 기반의 다양한 상업용 추천 시스템에서 사용자의 선호에 더욱 적합한 상품들을 추천하는 유용하게 활용될 수 있다. 또한 데이터셋의 희소성과 무관하게 우수한 성능을 보였으므로, 그 활용 가치 및 범위가 더욱 크다고 할 수 있다. 단, 기존의 유사도 척도들 중에서 평균차승차이만을 실험에 도입하였으므로, 다른 종류의 유사도 척도에도 제안 방법을 적용하여 성능 향상을 추구함이 향후 연구 과제로 진행될 예정이다.

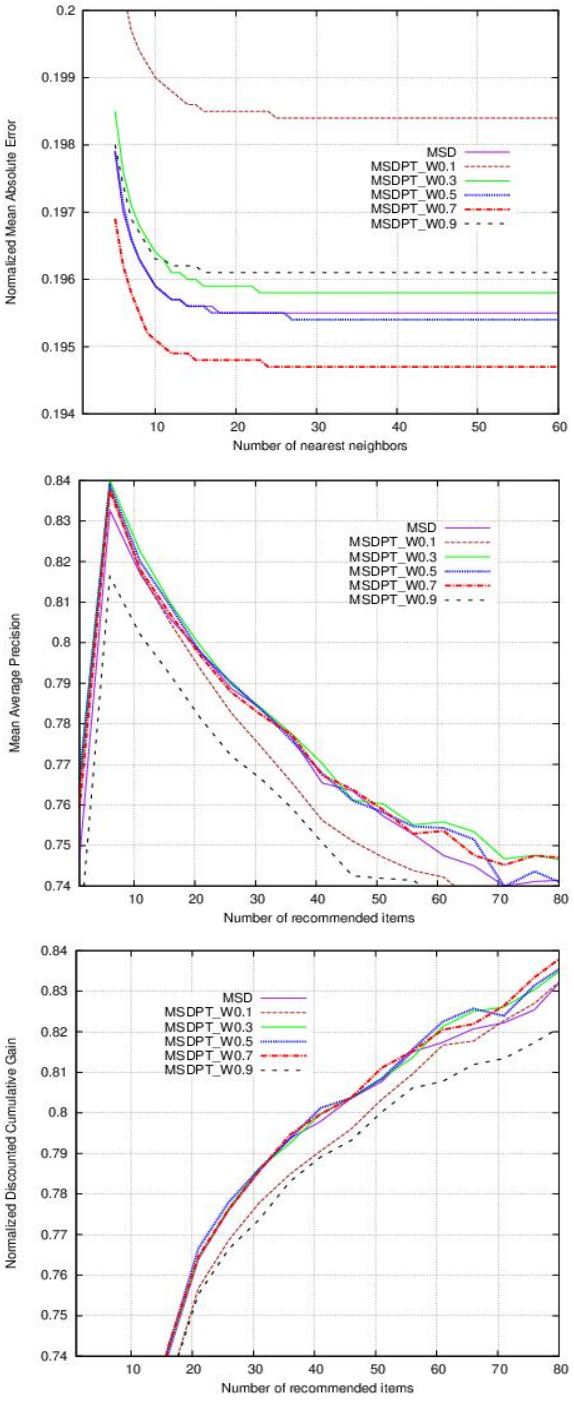

Fig. 1. Performance results using MovieLens

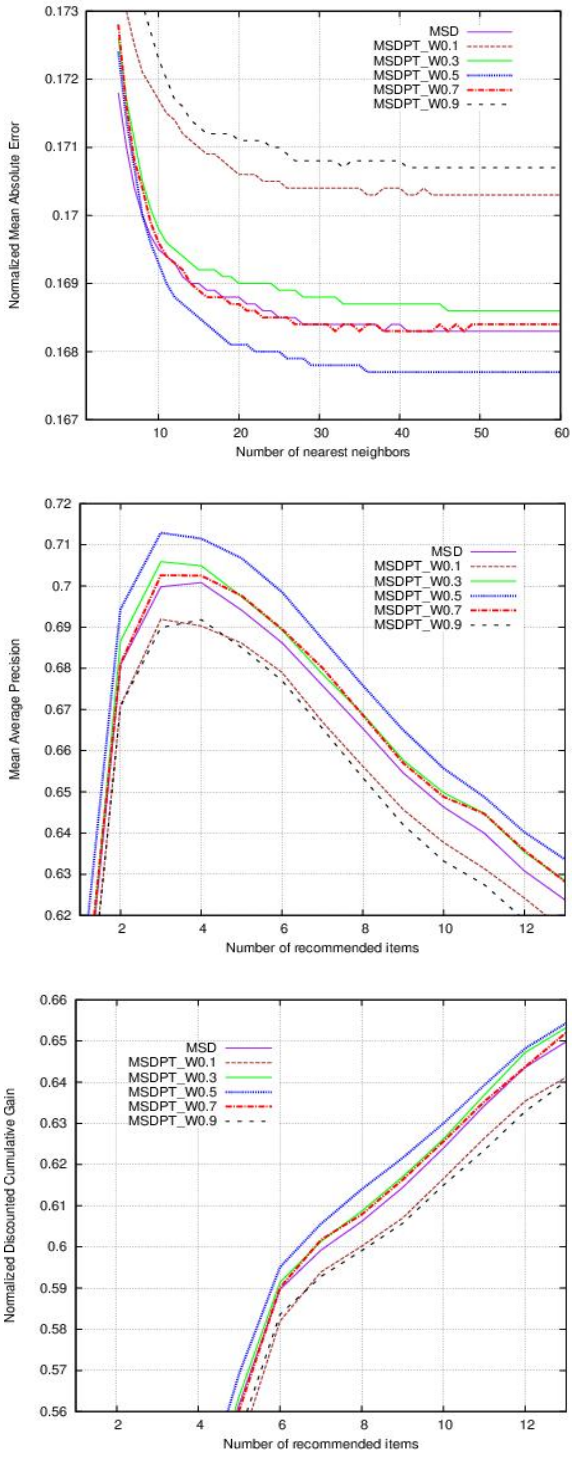

Fig. 2. Performance results using Jester

REFERENCES

[1] M. Jalili, S. Ahmadian, M. Izadi, P. Moradi, and M. Salehi, "Evaluating Collaborative Filtering Recommender Algorithms: A Survey," IEEE Access, Vol. 6, pp. 74003-74024, 2018. DOI: 10.1109/ACCESS.2018.2883742

[2] B. Shao, X. Li, and G. Bian, "A Survey of Research Hotspots and Frontier Trends of Recommendation Systems from the Perspective of Knowledge Graph," Expert Systems with Applications, Vol. 165, 2021. DOI: 10.1016/j.eswa.2020.113764

[3] S. Bag, S. K. Kumar, and M. K. Tiwari, "An Efficient Recommendation Generation using Relevant Jaccard Similarity," Information Sciences, Vol. 483, pp. 53-64, 2019. DOI: 10.1016/j.ins.2019.01.023

[4] S.-B. Sun, Z.-H. Zhang, X.-L. Dong, H.-R. Zhang, T.-J. Li, L. Zhang, and F. Min, "Integrating Triangle and Jaccard Similarities for Recommendation," PLoS ONE, Vol. 12, No. 8, 2017. DOI: 10.1371/journal.pone.0183570

[5] J. Bobadilla, F. Serradilla, and J. Bernal, "A New Ccollaborative Filtering Metric that Improves the Behavior of Recommender Systems," Knowledge-Based Systems, Vol. 23, No. 6, pp. 520-528, 2010. DOI: 10.1016/j.knosys.2010.03.009

[6] S. Kosub, "A Note on the Triangle Inequality for the Jaccard Distance," Pattern Recognition Letters, Vol. 120, pp. 36-38, 2019. DOI: 10.1016/j.patrec.2018.12.007

[7] K. G. Saranya, G. S. Sadasivam, and M. Chandralekha, "Performance Comparison of Different Similarity Measures for Collaborative Filtering Technique," Indian Journal of Science and Technology, Vol. 9, No. 29, Aug. 2016. DOI: 10.17485/ijst/2016/v9i29/91060

[8] B. Zhu, R. Hurtado, J. Bobadilla, and F. Ortega, "An Efficient Recommender System Method Based on the Numerical Relevances and the Non-Numerical Structures of the Ratings," IEEE Access, Vol. 6, pp. 49935-49954, 2018. DOI: 10.1109/ACCESS.2018.2868464

[9] A. Iftikhar, M. A. Ghazanfar, M. Ayub, Z. Mehmood, and M. Maqsood, "An Improved Product Recommendation Method for Collaborative Filtering," IEEE Access, Vol. 8, pp. 123841-123857, 2020. DOI: 10.1109/ACCESS.2020.3005953

[10] Y. Mu, N. Xiao, R. Tang, L. Luo, and X. Yin, "An Efficient Similarity Measure for Collaborative Filtering," Procedia Computer Science, Vol. 147, pp. 416-421, 2019. DOI: 10.1016/j.procs.2019.01.258

[11] S. Lee, "Improving Jaccard Index using Genetic Algorithms for Collaborative Filtering," Lecture Notes on Computer Science, 10385, pp. 378-385, 2017. DOI: 10.1007/978-3-319-61824-1_41

[12] M. Y. H. Al-Shamri, "Power Coefficient as a Similarity Measure for Memory-based Collaborative Recommender Systems," Expert Systems with Applications, Vol. 41, No. 13, pp. 5680-5688, 2014. DOI: 10.1016/j.eswa.2014.03.025

[13] H. Chen, Z. Li, and W. Hu, "An Improved Ccollaborative Recommendation Algorithm based on Optimized User Similarity," The Journal of Supercomputing, Vol. 72, No. 7, pp. 2565-2578, 2016. DOI: 10.1007/s11227-015-1518-5

[14] D. Wang, Y. Yih, and M. Ventresca, "Improving Neighbor-based

CF by Using a Hybrid Similarity Measurement," Expert Systems with Applications, Vol. 160, pp. 113651, 2020. DOI: 10.1016/j.eswa.2020.113651

[15] A. Jain, P. K. Singh, and J. Dhar, "Multi-objective Item Evaluation for Diverse as well as Novel Item Recommendations," Expert Systems with Applications, Vol. 139, 2020. DOI: 10.1016/j.eswa.2019.112857

[16] Y. Ar and E. Bostanci, "A Genetic Algorithm Solution to the Collaborative Filtering Problem," Expert Systems With Applications, Vol. 61, pp. 122−128, 2016. DOI: 10.1016/j.eswa.2016.05.021

[17] G. Lv, C. Hu, and S. Chen, "Research on Recommender System based on Ontology and Genetic Algorithm," Neurocomputing, Vol. 187, pp. 92-97, 2016. DOI: 10.1016/j.neucom.2015.09.113

[18] N. E. I. Karabadji, S. Beldjoudi, H. Seridi, S. Aridhi, and W. Dhifli, "Improving Memory-based User Collaborative Filtering with Evolutionary Multi-objective Optimization," Expert Systems with Applications, Vol. 98, pp. 153-165, 2018. DOI: 10.1016/j.eswa.2018.01.015

[19] R. Rashidi, K. Khamforoosh, and A. Sheikhahmadi, "Proposing Improved Meta-heuristic Algorithms for Clustering and Separating Users in the Recommender Systems," Electronic Commerce Research, 2021. DOI: 10.1007/s10660-021-09478-9

[20] M. Hatami and S. Pashazadeh, "Improving Results and Performance of Collaborative Filtering-based Recommender Systems using Cuckoo Optimization Algorithm," International Journal of Computer Applications, Vol. 88, No. 16, pp. 46-51, 2014. DOI: 10.5120/15440-3981

[21] L. Baltrunas, T. Makcinskas, and F. Ricci, "Group Recommendation with Rank Aggregation and Collaborative Filtering," Proceedings of the ACM Conference on Recommender Systems, pp. 119-126, 2010. DOI: 10.1145/1864708.1864733

Authors

Soojung Lee received the B.S. degree in Mathematics Education from Ewha Woman's University, Korea in 1985. She received M.S. and Ph.D. degrees in Computer Science from Texas A&M University in 1990 and 1994, respectively. Dr. Lee joined the faculty of the Department of Computer Education at Gyeongin National University of Education, Gyunggi-do, Korea, in 1998, as a professor. She is interested in recommender systems, information filtering, data mining techniques, and computer education.