

# Graph Neural Network 기반 추천시스템 성능 향상을 위한 유저 행동 시간 기반 데이터 사용 방안 분석

이상민\*<sup>1</sup>, 김희찬\*<sup>2</sup>, 김남기\*\*

## Analyzing the Use of Time-based User Behavior Data to Improve the Performance of Graph Neural Network-based Recommendation Systems

Sang-Min Lee\*<sup>1</sup>, Hee-Chan Kim\*<sup>2</sup>, and Namgi Kim\*\*

---

본 연구는 경기도의 경기도 지역협력연구센터 사업의 일환으로 수행하였음  
[GRRC경기2020-B04, 영상 및 네트워크 기반 지능정보 제조 서비스 연구]  
본 연구는 2023학년도 경기대학교 대학원 연구원장학생 장학금 지원에 의하여 수행되었음

---

### 요 약

모바일 기기의 발전으로 인해 해당 산업군의 비약적으로 발전했으며 이로 인해 사용자들은 이전에 비해 손쉽게 다양한 추천을 받을 수 있게 되었다. 하지만 잘못된 추천의 경우 오히려 사용자 경험 품질을 저해시키는 문제가 발생한다. 이러한 문제를 극복하기 위해서 본 연구에서는 현재 사용자가 원하는 아이템을 보다 정확히 추천하기 위해 시간 기반의 데이터가 추천시스템에 미치는 영향을 실험을 통해 분석하였다. 실험 결과 F1, Precision@20 점수는 전체 데이터를 활용하는 것보다 최근 데이터만을 이용하는 것이 더 높은 점수를 보였으나, Recall@20 점수의 경우 오히려 낮은 점수를 보였다. 이러한 결과로 유추하였을 때 현재의 사용자가 원하는 아이템의 예측 정확도 높이기 위해서는 데이터의 개수보다, 시간을 기준으로 하여 데이터를 분류하여 학습하는 것이 보다 높은 성능을 낼 수 있음을 확인하였다.

### Abstract

The evolution of mobile devices has led to leaps and bounds in this industry, allowing users to get recommendations more easily than ever before. However, in the case of incorrect recommendations, the quality of the user experience is compromised. In order to overcome this problem, this study analyzed the effect of time-based data on the recommendation system through experiments in order to more accurately recommend items that users currently want. The experimental results showed that the F1 and Precision@20 scores were higher using only recent data than using all data, but the Recall@20 score was lower. Inferring from these results, we found that to improve the prediction accuracy of items desired by current users, it is better to learn by classifying data based on time rather than the number of data.

### Keywords

time-based database, recommender systems, graph neural network, graph-based recommendation

---

\* 경기대학교 컴퓨터과학과 석사과정  
- ORCID<sup>1</sup>: <https://orcid.org/0000-0002-7720-8622>  
- ORCID<sup>2</sup>: <https://orcid.org/0009-0002-4930-3486>  
\*\* 경기대학교 AI컴퓨터공학부 교수(교신저자)  
- ORCID: <https://orcid.org/0000-0002-0077-6576>

• Received: Mar. 24, 2023, Revised: Apr. 14, 2023, Accepted: Apr. 17, 2023  
• Corresponding Author: Namgi Kim  
Division of Computer Science and Engineering, Kyonggi University, 154-42,  
Gwanggyosan-ro, Yeongtong-gu, Suwon-si, Gyeonggi-do, Korea  
Tel.: +82-31-249-9962, Email: [ngkim@kgu.ac.kr](mailto:ngkim@kgu.ac.kr)

## 1. 서 론

멀티미디어 응용 프로그램과 모바일 기기의 발달로 1990년 대비 인터넷 사용자 수가 기하급수적으로 증가하면서 데이터 쓰나미의 경고음이 울리고 있다. 이에 따르면 2022년까지 무선 및 모바일 기기에서의 트래픽이 전체 IP 트래픽의 71%를 차지할 것으로 예상되며, 전 세계 모바일 데이터 트래픽은 정적 IP 트래픽보다 거의 2배 빠르게 성장할 것으로 예상된다[1]. 이러한 모바일 기기의 발달은 eCommerce, OTT(Over The Top) 서비스 및 실시간 스트리밍 서비스와 같은 서비스에 대한 모바일 액세스를 용이하게 만들어 많은 사용자를 유치하고, 이와 같은 산업의 발전에 큰 영향을 미쳤다. 그러나 다양한 선택지로 인해 사용자는 제품 또는 서비스를 선택하는 데 매우 혼란스러워 하며, 선택에 오랜 시간이 걸리므로 기업은 사용자 친화적인 서비스를 제공하여 기업 서비스의 경험 품질을 향상시켜 기존 사용자를 유지하고 새로운 사용자를 유치하기 위한 기술 개선의 필요성이 제기되었다. 이를 위한 한 가지 방법으로 추천 시스템을 사용하여 사용자에게 선호하는 제품이나 서비스를 인공 지능 및 기계 학습 기술을 사용하여 추천하는 것이다. 이러한 추천 시스템은 온라인 쇼핑, 음악 스트리밍, 비디오 스트리밍, 소셜 미디어 등 다양한 분야에서 널리 사용되며, 사용자의 취향 및 행동 패턴을 분석하여 최적의 추천을 제공함으로써 개인 맞춤형 서비스를 제공할 수 있다. 추천 시스템은 고객 만족도를 높이고 이를 통해 고객의 재구매율을 높여 매출 증가에 직접적인 영향을 줄 수 있으며, 성공적인 예로는 Netflix와 Amazon이 있다. 이러한 기업들은 추천 시스템을 사용하여 실제로 원하지 않는 제품을 고객에게 광고하는 것을 방지하여 고객 불만도를 최소화하고 기업 이미지를 개선하는 용도로 사용하였으며, 고객 만족도가 증가함에 따라 서비스 사용자는 더욱 증가하였다. 이렇듯 기업의 이윤을 극대화 하고 사용자 품질 경험 증진을 위해서는 서비스 품질 개선이 요구된다[2]-[6].

사용자 품질 경험 저해 문제를 해결하기 위해 추천시스템은 고객이 좋아할 만한 제품을 추천하여

고객 경험의 질을 높이고, 기업의 판매 활동을 촉진하는데 사용된다. 하지만 추천 결과가 부정확하거나 추천 결과가 고객의 취향이나 상황에 따라 변하지 않을 경우, 고객의 만족도가 저하되고, 사용자의 관심이 변화함에 따라 새로운 아이템을 추천하지 못하여 사용자 경험의 질이 떨어질 수 있다. 이를 해결하기 위해 최근에 구매한 상품이나 검색어를 기반으로 추천 결과를 제공하거나 최근에 검색하지 않은 제품이나 서비스를 추천하여 새로운 제품을 소개할 수 있다. 또한 제품이나 서비스의 인기 변화를 반영하여 급격히 인기가 오르는 특정 제품을 추천할 수도 있다. 이러한 개인화 추천 시스템은 사용자 취향의 변화를 고려할 수 있으며, 특정 기간 동안 검색한 제품이나 서비스를 기반으로 추천 결과를 제공함으로써 사용자의 관심에 맞는 추천 결과를 더욱 개선할 수 있다. 그러나 이 방법은 사용자와 아이템 간 상호작용을 모델링하여 추천하는 것이 아니라, 최근에 검색한 아이템과 유사한 아이템이나 최근에 인기가 상승한 아이템을 추천하는 방식이므로 장기적인 데이터를 사용하여 추천하는 것은 불가능하다. 따라서 딥러닝 기술을 사용하여 다양한 데이터를 활용하여 추천 결과의 정확도를 높이는 다양한 방법이 연구되고 있다[7]-[9].

본 연구에서는 유저와 아이템 간의 상호작용을 모델링하여 추천 결과를 제공하기 위해 GNN(Graph Neural Network)을 활용하였다. 기존 Graph-based Recommendation 연구에서는 전체 데이터를 모두 사용하여 추천 결과를 제공했으나, 사용자 행동 시간에 대한 분석이 진행되지 않았기 때문에 현재 사용자가 원하는 아이템에 대한 예측 성능이 부족한 문제가 있다. 이러한 문제를 해결하기 위해 사용자와 아이템 간의 관계 모델링에 사용되는 정보 중 시간 관련 정보가 그래프 기반 추천 시스템에 미치는 영향에 대해 집중적으로 분석하였다. 따라서 본 연구에서는 이전에 사용자가 좋아했던 아이템을 추천하는 것이 아닌 현재 사용자가 원하는 아이템을 보다 중심으로 추천하기 위해 사용자와 아이템 간의 관계 모델링에 사용되는 정보 중 시간 관련 정보에 대해 집중적으로 분석하였다.

## II. 관련 연구

추천시스템은 많은 분야에서 사용되고 있으며, 인공지능 분야에서도 큰 관심을 받고 있으며, 그 발전 동향은 크게 Collaborative Filtering, Content-based Filtering, Hybrid Recommendation, Graph-based Recommendation의 순으로 발전하였다. Graph-based Recommendation은 사용자와 아이템을 그래프로 표현하고, 그래프 분석 기술을 사용하여 추천을 수행하기에, Graph-based Recommendation 방식의 일종인 GNN를 적용하면 사용자와 아이템 간의 상호작용을 그래프 형태로 표현이 가능해진다. 이를 기반으로 좀 더 정확하고 개인화된 추천을 할 수 있게 되었으며, 최근 연구들에서는 GNN을 활용하여 추천시스템의 성능을 높이기 위한 다양한 방법들이 제안되고 있다. 이러한 GNN 기반 아키텍처로는 GraphSAGE[13], GCN[14], GAT[15] 등의 다양하게 있다. 이러한 GNN 기반 아키텍처는 전통적인 추천시스템 알고리즘들과 함께 사용될 수 있으며, GNN 기반에 아키텍처와 결합하여 추천시스템의 성능을 향상시키는 방법인, attention mechanism과 같은 기술을 활용하여 사용자와 아이템 간의 상호작용을 모델링하거나, autoencoder와 같은 기술을 활용하여 잠재요인(Latent factor)을 추출하여 보다 개인화된 추천을 제공하는 등의 연구들이 진행되고 있다. 그 밖에도 다양한 방식을 적용하여 Graph-based Recommendation의 추천시스템의 성능을 향상시키기 위한 다양한 연구들이 활발히 진행되고 있다. 본 연구에서는 현재까지 가장 좋은 성능을 보이고 있는 그래프 기반의 협력 필터링 기법을 활용하였으며, 해당 기법은 Amazon, CDNow, MovieFinder 등과 같은 다양한 분야의 기업에서 사용되고 있다. 그래프 기반의 협력 필터링은 “특정 아이템에 대해 선호도가 유사한 고객들은 다른 아이템에 대해서도 비슷한 선호도를 보일 것”이라는 기본 가정을 바탕으로 사용자 혹은 아이템 간 유사도를 기반으로 선호도를 예측하는 방법이다. 콘텐츠 기반의 추천 방식과의 가장 큰 차이점은 콘텐츠 기반 접근 방식이 사용자와 아이템 정보에만 의존하여 선호도를 예측하는 반면, 협력 필터링은 사용자가 아이템에 대해 평

가한 정보를 사용해 선호도를 예측한다는 것이다. 이러한 방식을 통해 추천을 받고자 하는 고객과 취향이 비슷한 사용자를 매핑할 수 있으며, 연결된 사용자들에게 알맞는 아이템을 추천함으로써 그룹마다 추천되는 아이템의 다양성을 보장할 수 있다. 예를 들어, 사용자 기반 협력 필터링은 구매 이력을 바탕으로 추천 대상 고객과 다른 사용자들 간의 유사도를 측정한 후, 유사도가 가장 높은 사용자를 이웃으로 선택하게 된다. 이웃 유저는 구매하였지만 추천을 받고자 하는 유저가 구매하지 않은 아이템을 추천하는 방식이다. 이와 같은 협력 필터링은 크게 기억 기반 협력 필터링과 모델 기반 협력 필터링으로 나뉘게 되는데, 기억 기반 협력 필터링은 위의 예시와 동일하게 유사도를 바탕으로 하여 아이템을 추천하는 방식을 말하며, 모델 기반 협력 필터링은 기억 기반 협력 필터링 방식을 기본으로 하며, 군집화, 분류, 예측의 단계에서 기계 학습 또는 데이터 마이닝 기법을 활용하는 것이다[10]-[16].

이러한 협력 필터링 기반의 추천 시스템의 성능을 높이기 위해 딥러닝을 적용하는 방안의 연구가 활발히 진행되고 있으며, 본 연구에서는 그래프 기반의 추천 시스템 모델에 시간 기반 데이터를 활용하여 연구를 진행하였다.

## III. 시간 정보를 활용한 추천시스템

### 3.1 실험 절차

본 연구에서는 최근 유저의 행동 데이터가 추천 시스템에 미치는 영향을 분석하기 위해, 시간을 가장 주요한 정보로 생각하고, 유저 행동 시간 정보를 포함하고 있는 MovieLens 100k, 1m 데이터를 기반으로 실험을 진행했다. 해당 데이터를 사용한 이유는, 표 3에 나타난 대로 Timestamp라는 속성이 있으며, 해당 데이터에는 사용자가 아이템에 대한 평점을 매긴 시간 정보를 의미한다. 이러한 기록은 사용자 interactions에 기록되어 있으며, 데이터 기간에 따른 성능 파악을 위한 실험을 진행하기 위해, 원본의 전체 데이터를 시간순으로 정렬한 후 가장 최근 데이터를 test data로 분리하여 현재의 사용자가 필요로 하는 아이템을 잘 추천하는지 분석했다.

이를 통해 현재 시점에서 시간에 따른 추천 성능을 비교 할 수 있다. 나머지 데이터는 기간별로 분리하여 학습을 진행하고, 이를 통해 학습 데이터의 기간과 총량에 따른 추천 성능을 비교했다. 모델 별 실험 과정을 그림 1을 통해 간략하게 도식화 하였으며, 이는 Graph-based 추천 시스템의 기본적인 학습 과정이다. 표 1, 표 2는 진행된 실험의 환경과 매개변수를 나타낸다. 3.2절에는 실험에 사용된 데이터에 대한 상세 정보를 나타내며, 표 3, 4, 5는 데이터 별 속성과 나누어진 기간을 표시하고, 표 6에는 각 데이터별 고유한 값의 개수를 나타낸다.

표 1. 실험 환경  
Table 1. Experimental environments

Device type	Property
CPU	Intel i9-12900k
GPU	Nvidia GeForce 3090 24GB
DRAM	64GB
OS	Ubuntu 18.04.06 LTS
Nvidia dirver	525.78.01
CUDA	12.0
CuDNN	11.6
Python	3.7.11
Pytorch	1.10.0

표 2. 실험 매개변수  
Table 2. Experiment parameters

Property	Parameters
Epoch	1000
Seed	2023
Optimizer	Adam
Learning late	0.001
Weight decay	0.0005
Embedding size	64
Random state	42

### 3.2 실험 데이터

MovieLens 데이터 세트는 미네소타 대학의 GroupLens 연구 프로젝트에서 수집하였으며, ml-100k는 1997년 9월 19일부터 1998년 4월 22일까지 약 7개월(216일)동안 수집되었으며, ml-1m는 2000년 4월 25일부터 2003년 2월 25일까지 약 3년(1037일)동안 수집된 데이터이다. 표 3, 4, 5은 각각 MovieLens 100k, 1m 데이터에 대한 속성 정보와 데이터를 기간별로 나누어 분리한 정보이다. 표 3은 MovieLens 100k 데이터에 대한 속성 정보를 담고 있으며, 표 4는 MovieLens 100k 데이터에 대한 기간별로 분리한 정보를 담고 있다. 마지막으로, 표 5는 MovieLens 1m 데이터에 대한 기간별로 분리한 정보를 담고 있다.

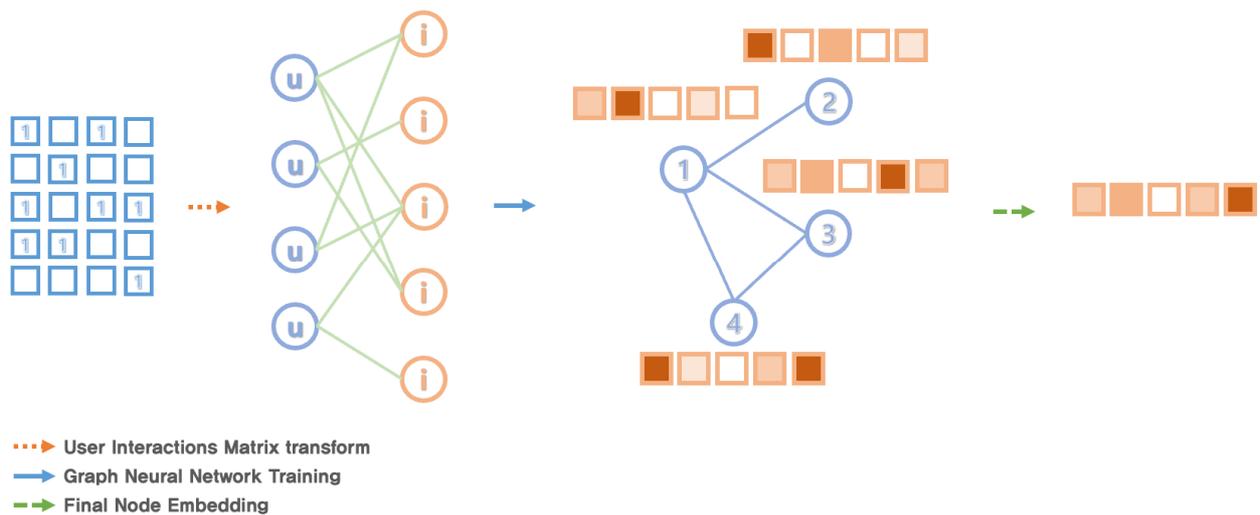


그림 1. 그래프 기반 추천시스템 구조 및 흐름  
Fig. 1. Graph-based recommendation system structure and flow

표 3. 실험 데이터 속성

Table 3. Experiment data properties

User interaction	User information	Item information
user id	user id	item id
item id	age	movie title
rating	gender	release year
timestamp	occupation	class
	zip code	

표 4. 실험에 사용된 ml-100k 기간

Table 4. ml-100k period used in the experiment

Dataset name	Period
Test	1998/03/31 ~ 1998/04/22 (23day)
ml_100k_1	1998/01/13 ~ 1998/03/31 (78day)
ml-100k_2	1997/12/22 ~ 1998/03/31 (100day)
ml-100k_3	1997/11/08 ~ 1998/03/31 (144day)
ml-100k_4	1997/09/20 ~ 1998/03/31 (193day)
ml-100k_random	1997/09/20 ~ 1998/03/31 (193day)

표 5. 실험에 사용된 ml-1m 기간

Table 5. ml-1m period used in the experiment

Dataset name	Period
Test	2000/12/29 ~ 2003/02/25 (789day)
ml_1m_1	2000/11/20 ~ 2000/12/29 (40day)
ml-1m_2	2000/10/31 ~ 2000/12/29 (60day)
ml-1m_3	2000/07/21 ~ 2000/12/29 (162day)
ml-1m_4	2000/04/25 ~ 2000/12/29 (249day)
ml-1m_random	2000/04/25 ~ 2000/12/29 (249day)

MovieLens 100k와 1m 데이터는 user id, item id, rating, timestamp의 4가지 속성으로 이루어져 있다. user id는 각 유저를 구분하기 위한 고유한 식별자이고, item id는 각 아이টে를 구분하기 위한 고유한 식별자이다. rating은 유저가 아이টে에 대해 매긴 평점 정보를 담고 있으며, timestamp 속성은 유저가 아이টে에 대한 rating을 매긴 시간 정보를 담고 있다. 표 6은 데이터 별 고유한 값의 개수를 나타내며, MovieLens 100k 데이터에서는 943명의 users와 1682개의 items, MovieLens 1m 데이터에서는 6040명의 users와 3706개의 items가 존재한다. 이러한 데이터는 각각의 user와 item에 대한 정보들을 가지고 있으며, 이러한 데이터를 기반으로 영화 추천 시스템을 구현할 수 있으며, 이러한 데이터는 다양한 분석에 유용하게 사용될 수 있다. 예를 들어, 이 데이터를 사용하여 사용자의 취향이나 아이টে의 특성을

분석할 수 있다. 이는 산업에서 매우 중요한 정보가 되며, 이를 바탕으로 영화 제작 및 마케팅 전략을 수립할 수 있다.

표 6. 분리된 데이터 집합당 고유 값

Table 6. Number of unique values per separated dataset

Dataset name	User	Item	Interaction
ml-100k_Test	166	1,343	4,806
ml-100k_1	407	1,533	14,495
ml-100k_2	505	1,553	19,548
ml-100k_3	743	1,593	24,354
ml-100k_4	867	1,637	44,476
ml-100k_random	866	1,474	22,445
ml-1m_Test	1,208	3,407	63,401
ml-1m_1	2,352	3,562	125,242
ml-1m_2	3,095	3,589	167,754
ml-1m_3	4,862	3,650	301,314
ml-1m_4	6,010	3,678	394,938
ml-1m_random	6,001	3,539	211,093

#### IV. 실험 및 성능 분석

본 연구의 실험 결과는 그림 2과 3를 통해 추세 파악이 가능하다. ml-100k\_1은 30,000개, ml-100k\_2는 40,000개, ml-100k\_3는 70,000개, ml-100k\_4는 90,000개, ml-100k\_random는 ml-100k\_1와 같은 30,000개의 사용자 interaction을 가지고 있으며, ml-1m의 경우 ml-100k의 10배 분량의 사용자 행동에 관한 데이터를 갖는다. ml-100k\_1, ml-1m\_1(빨간색), ml-100k\_2, ml-1m\_2(초록색), ml-100k\_3, ml-1m\_3(파란색), ml-100k\_4, ml-1m\_4(하늘색), ml-100k\_random, ml-1m\_random(노란색)으로 구분하였다. 실험 결과에 따르면 학습되는 데이터의 기간에 따라 추천 성능이 큰 차를 보인다는 점을 볼 수 있다. 실험 결과 Precision@20/Recall@20의 성능 추이는 명확하게 다르다는 것을 파악할 수 있으며, Precision@20는 내가 추천한 아이টে 20개 중에 실제 사용자가 관심있는 아이টে의 비율을 의미하고 Recall@20는 사용자가 관심있는 모든 아이টে 중에서 내가 추천한 아이টে 20개가 얼마나 포함되는지 비율을 의미한다. 데이터의 개수보다 최신의 데이터를 학습하는 것이 Precision@20의 정확도가 더욱 우수하며, 이는 실제 사용자가 관심있는 아이টে를 더 잘 예측한다는 것을 알 수 있습니다.

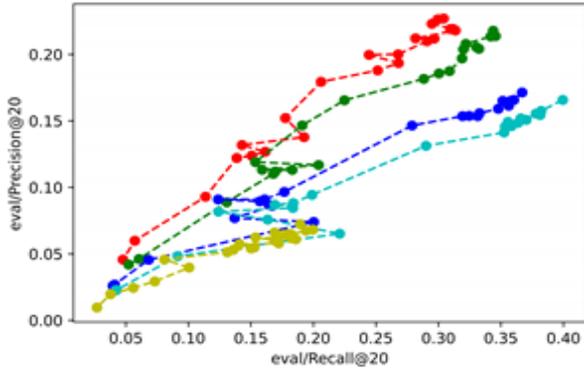


그림 2. ml-100k 시간 경과에 따른 정확도/리콜 추세  
Fig. 2. ml-100k Precision/Recall trend over time

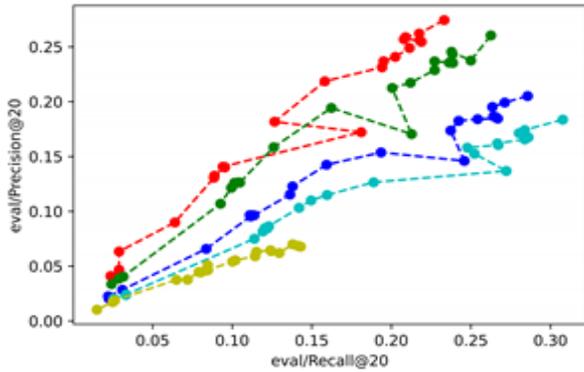


그림 3. ml-1m 시간 경과에 따른 정확도/리콜 추세  
Fig. 3. ml-1m Precision/Recall trend over time

반면 시간 구분 없이 모든 데이터를 학습에 사용하는 경우 Recall@20의 정확도 측면에서 더욱 우수하며, 이러한 학습 결과가 의미하는 바는 사용자가 관심있는 모든 아이템 중 모델이 추천한 아이템의 개수가 더 많다는 것을 의미한다. 두 개의 지표를 종합적으로 분석한 결과, 최신 데이터만을 통해 학습한 모델은 추천하여야 하는 개수에 대한 정보가 부족하여 추천의 성능이 떨어지는 것으로 분석되며, 전체 기간을 모두 학습한 모델의 경우 너무 많은 정보를 고려한 추천을 진행하였기에 오히려 성능이 떨어지는 것으로 분석된다. 표 7과 8의 1, 2, 3, 4는 기간의 따른 데이터를 의미하며 5는 1과 사용되는 데이터의 양을 맞추기 위해 전체 데이터에서 무작위하게 추출한 데이터를 의미하며, 5의 실험 결과가 선행 연구의 추천 시스템 성능 지표로 볼 수 있다. 표 7과 8은 Precision@20과 Recall@20을 모두 고려한 F1 score을 나타내고 있다. 대부분의 모델들이 더욱 많은 데이터를 사용한 3, 4의 경우보다 더욱

적은 데이터인 최신 데이터들만 학습에 사용하는 것이 F1 score 점수가 더욱 높아지는 것을 확인할 수 있다. 이는 최신 데이터를 학습시키는 것이 추천 시스템의 성능을 향상시키는 데에 더 효과적이라는 것을 의미한다. 결과적으로 사용자의 취향 변화나 아이템의 인기 변화를 빠르게 반영하기 위해서는 시간 정보를 중점적으로 사용해야 한다는 것을 의미한다. 또한, 이 추천 시스템은 Graph Neural Network를 이용하여 사용자와 아이템 간의 상호작용을 고려하여 추천을 진행하므로, 성능이 높다는 것을 보여준다.

$$Precision@K = \frac{1}{|U|} \sum_{u \in U} \frac{|\hat{R}(u) \cap R(u)|}{|\hat{R}(u)|} \quad (1)$$

$$Recall@K = \frac{1}{|U|} \sum_{u \in U} \frac{|\hat{R}(u) \cap R(u)|}{|R(u)|} \quad (2)$$

$$F_1 = 2 \times \frac{precision \times recall}{precision + recall} \quad (3)$$

표 7. 시간 경과에 따른 ml-100k F1 점수  
Table 7. ml-100k F1 score over time

model	1	2	3	4	5
CDAE	0.247	0.244	0.222	0.215	0.102
ConvNCF	0.102	0.106	0.099	0.104	0.057
DGCF	0.164	0.134	0.125	0.128	0.082
DMF	0.136	0.133	0.114	0.115	0.080
ENMF	0.242	0.253	0.226	0.220	0.095
FISM	0.142	0.140	0.114	0.116	0.081
GCMC	0.047	0.047	0.032	0.029	0.014
ItemKNN	0.254	0.249	0.212	0.206	0.089
LightGCN	0.137	0.132	0.116	0.119	0.076
LINE	0.192	0.166	0.105	0.099	0.058
MacridVAE	0.257	0.253	0.218	0.214	0.094
MultiDAE	0.257	0.264	0.227	0.211	0.087
MultiVAE	0.257	0.263	0.226	0.220	0.089
NAIS	0.215	0.230	0.207	0.202	0.092
NCEPLRec	0.161	0.149	0.108	0.101	0.042
NeuMF	0.225	0.223	0.209	0.210	0.100
NGCF	0.220	0.191	0.192	0.181	0.095
NDCF	0.229	0.047	0.031	0.030	0.026
Pop	0.059	0.052	0.055	0.063	0.034
RecVAE	0.260	0.267	0.234	0.234	0.095
SGL	0.258	0.253	0.225	0.223	0.105
SimpleX	0.244	0.233	0.210	0.208	0.092
SpectralCF	0.130	0.135	0.114	0.117	0.074

표 8. 시간 경과에 따른 ml-1m F1 점수

Table 8. ml-1m F1 score over time

model	1	2	3	4	5
CDAE	0.229	0.237	0.219	0.211	0.084
ConvNCF	0.106	0.109	0.125	0.127	0.057
DGCF	0.040	0.028	0.022	0.023	0.012
DMF	0.113	0.112	0.104	0.102	0.060
ENMF	0.238	0.241	0.230	0.216	0.085
FISM	0.114	0.115	0.104	0.100	0.060
GCMC	0.036	0.036	0.030	0.028	0.020
ItemKNN	0.231	0.232	0.208	0.193	0.082
LightGCN	0.075	0.099	0.074	0.090	0.048
LINE	0.030	0.033	0.022	0.028	0.022
MacridVAE	0.220	0.228	0.214	0.202	0.070
MultiDAE	0.235	0.244	0.219	0.209	0.092
MultiVAE	0.235	0.236	0.218	0.201	0.092
NAIS	0.211	0.215	0.201	0.190	0.078
NCEPLRec	0.177	0.189	0.183	0.182	0.049
NeuMF	0.184	0.177	0.150	0.134	0.072
NGCF	0.149	0.140	0.130	0.120	0.064
NNCF	0.036	0.036	0.030	0.028	0.020
Pop	0.106	0.110	0.103	0.097	0.057
RecVAE	0.252	0.261	0.239	0.230	0.092
SGL	0.230	0.242	0.224	0.213	0.093
SimpleX	0.214	0.206	0.171	0.152	0.084
SpectralCF	0.113	0.114	0.103	0.099	0.059

따라서, 이 연구에서 제안한 추천 시스템은 기존 추천 시스템 모델 성능 대비하여 성능 개선을 보여 주며, 최신 데이터를 기반으로 학습을 시키는 것이 추천 시스템의 성능을 향상시키는 데에 중요하다는 것을 알 수 있다.

## V. 결론 및 향후 과제

본 연구에서는 그래프 신경망 기반 추천 시스템의 성능 개선을 위해 유저 행동 시간 기반 데이터를 활용하는 방안을 분석하였다. 실험 결과, 학습되는 데이터의 기간에 따라 추천 성능이 큰 차이를 보인다는 것을 확인하였다. 최신 데이터만을 학습하는 모델은 추천할 개수에 대한 정보가 부족하여 추천의 성능이 떨어지는 반면, 전체 기간을 학습에 사용하는 모델은 너무 많은 정보를 고려한 추천을 하여 오히려 성능이 떨어지는 것으로 나타났다. 두 개의 지표를 종합적으로 분석한 결과, 최신 데이터만을 통해 학습한 모델은 Precision@20의 정확도가 높

아진다는 것을 발견하였다. 반면 시간 구분 없이 모든 데이터를 학습에 사용하는 경우 Recall@20의 정확도 측면에서 더욱 우수하였다. 이러한 결과를 토대로, 현재의 사용자가 원하는 추천을 진행하기 위해서는 전체 데이터를 통해 추천을 진행하는 것이 아닌, 사용자의 취향 변화, 새로운 아이템, 아이템의 인기 변화와 같은 정보를 포착할 수 있는 시간을 기반으로 한 데이터를 통해 실시간 추천을 진행하는 것이 유의미하다는 것을 알 수 있었다. 향후 과제로는, 최신 추세 학습이 가능한 학습 방법인 Continual Learning과 같은 방법을 통해 과거의 정보를 완전히 학습에 사용하지 않고, 과거의 데이터부터 시작하여 현재의 데이터에 맞게 적응하는 방법의 학습을 적용하는 것이 추천 성능 개선에 유의미할 것으로 보이기에 이러한 학습 방법을 통해 과거의 정보를 완전히 잃지 않으며, 최신 추세 학습이 가능하기에 Precision과 recall 두 가지의 수치를 높일 수 있는 방안에 대해 연구해 볼 필요가 있다.

## References

- [1] Cisco, "Cisco Visual Networking Index: Forecast and Trends, 2017 - 2022", Cisco public, 2019.
- [2] F. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook", Springer, pp. 1-35, Oct. 2010. [https://doi.org/10.1007/978-0-387-85820-3\\_1](https://doi.org/10.1007/978-0-387-85820-3_1).
- [3] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions", IEEE Transactions on Knowledge and Data Engineering, Vol. 17, No. 6, pp. 734-749, Jun. 2005. <https://doi.org/10.1109/TKDE.2005.99>.
- [4] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques", Advances in artificial intelligence, Oct. 2009. <https://doi.org/10.1155/2009/421425>.
- [5] R. Burke, "Hybrid recommender systems: Survey and experiments", User Modeling and User-Adapted Interaction, Vol. 12, No. 4, pp. 331-370, Nov. 2002. <https://doi.org/10.1023/A:1021240730564>.

[6] P. Resnick and H. R. Varian, "Recommender systems", *Communications of the ACM*, Vol. 40 No. 3, pp. 56-58, 1997.

[7] Y. Koren, S. Rendle, and R. Bell, "Advances in collaborative filtering", Springer, pp. 145-186, Nov. 2021. [https://doi.org/10.1007/978-1-0716-2197-4\\_3](https://doi.org/10.1007/978-1-0716-2197-4_3).

[8] C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles", Association for Computing Machinery, pp. 448-456, Aug. 2011. <https://doi.org/10.1145/2020408.2020480>.

[9] G. Adomavicius and A. Tuzhilin, "Context-aware recommender systems", Springer, pp. 217-253, 2015. [https://doi.org/10.1007/978-1-4899-7637-6\\_6](https://doi.org/10.1007/978-1-4899-7637-6_6).

[10] R. V. D. Berg, T. N. Kipf, and M. Welling, "Graph Convolutional Matrix Completion", *Proceedings of the 11th ACM International Conference on Web Search and Data Mining (WSDM)*, Jun. 2017. <https://doi.org/10.48550/arXiv.1706.02263>.

[11] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, "Graph Convolutional Neural Networks for Web-Scale Recommender Systems", Association for Computing Machinery, pp. 974-983, Jul. 2018. <https://doi.org/10.1145/3219819.3219890>.

[12] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks", *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 32, No. 1, pp. 4-24, Jan. 2021. <https://doi.org/10.1109/TNNLS.2020.2978386>.

[13] W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive Representation Learning on Large Graphs", *Advances in neural information processing systems*, Vol. 30, Jun. 2017. <https://doi.org/10.48550/arXiv.1706.02216>.

[14] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks", In *Proceedings of the International Conference on Learning Representations (ICLR)*, Sep. 2016.

<https://doi.org/10.48550/arXiv.1609.02907>.

[15] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph Attention Networks", arXiv preprint arXiv:1710.10903, Oct. 2017. <https://doi.org/10.48550/arXiv.1710.10903>.

[16] H. S. Oh, J. M. Jeong, Y. J. Nam, Y. D. Seo, and E. J. Lee, "Deep Learning-based Recommendation System using User Purchase History", *Journal of KIIT*, Vol. 20, No. 6, pp. 117-128, Jun. 2022. <https://doi.org/10.14801/jkiit.2022.20.6.117>.

### 저자소개

이 상 민 (Sang-Min Lee)



2022년 2월 : 경기대학교  
컴퓨터공학부(공학사)  
2022년 3월 ~ 현재 : 경기대학교  
컴퓨터과학과 석사과정  
관심분야 : 인공지능/머신러닝,  
네트워크, 추천시스템

김 희 찬 (Hee-Chan Kim)



2023년 2월 : 경기대학교  
AI컴퓨터공학부(공학사)  
2023년 3월 ~ 현재 : 경기대학교  
컴퓨터과학과 석사과정  
관심분야 : 강화학습, 메타러닝,  
네트워크

김 남 기 (Namgi Kim)



1997년 2월 : 서강대학교  
컴퓨터과학과(공학사)  
2000년 3월 : KAIST  
전산학과(공학석사)  
2005년 3월 : KAIST  
전산학과(공학박사)  
2007년 2월 : 삼성전자 통신연구소

책임연구원

2007년 3월 ~ 현재 : 경기대학교 컴퓨터공학부 교수  
관심분야 : 통신시스템, 네트워크