

# Level2 P-Stage Movie Recommendation

## RecSys-15조

### 1. 프로젝트 개요

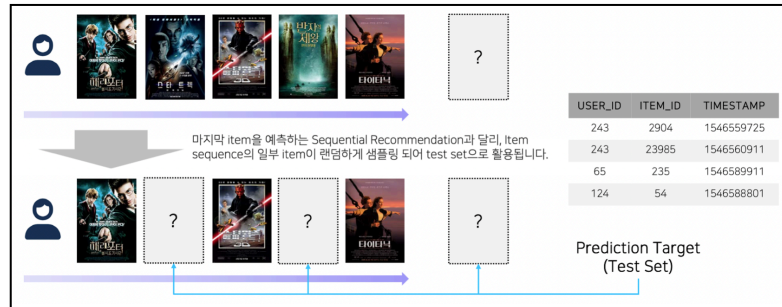
#### 1-1. 프로젝트 주제

사용자의 영화 시청 이력 데이터를 바탕으로 사용자가 다음에 시청할 영화와 좋아할 영화를 예측하는 모델을 만든다.

#### 1-2. 데이터셋 구조

train\_ratings.csv

- user id, item id, timestamp(초)로 구성 (5,154,471행)
- time-ordered sequence 중 일부 item이 누락되었으며 (drop out), 해당 누락 data를 예측



#### Side Information

- 영화의 제목, 개봉연도, 감독, 작가, 장르로 구성

#### 1-3. 평가 방법

- 각 유저당 10개의 영화를 추천 후 (normalized) Recall@10 계산 ( $0 \leq \text{Recall@10} \leq 1$ )

$$\text{Recall@K} = \frac{1}{|U|} \sum_{u \in U} \frac{|\{i \in I_u | \text{rank}_u(i) \leq K\}|}{\min(K, |I_u|)}$$

#### 1-4. 개발 환경 (개발 및 협업 툴)

- PyTorch (사용 라이브러리), Git (버전 관리), Asana (일정 관리), Slack (협업 관리)

### 2. 프로젝트 팀 구성 및 역할

- 소경학: EDA, 하이퍼 파라미터 실험, Model Ensemble 설계
- 임경연: BiVAE, H+Vamp 모델 실험 관리, Ensemble 진행
- 하동준: Microsoft Recommenders 모델 성능 전수 조사, SVD++ 구현, 앙상블 및 overfitting 방지
- 허민영: RippleNet, pinSAGE, recVAE 모델 실험
- 노건웅: Mult-VAE, RecVAE, EASE 모델 구성 및 하이퍼 파라미터 실험

### 3. 프로젝트 수행 절차 및 방법

#### 3-1. 프로젝트 사전 기획 및 문제 정의

4주 동안의 대회 기간임을 감안하여 초반 2주 정도는 데이터 EDA와 주어진 Baseline과 Special Mission 코드를 대회 task에 맞게 Top-K개 추천으로 수정하는데 초점을 맞췄다. timestamp와 영화 개봉연도 등에 이상치가 다수 존재 하는 등 대회 데이터가 sequential 모델을 사용하기에 적합하지 않았다. 결측치가 많이 존재하는 Side information을 제외하는 것이 타당하다고 판단하였고 AutoEncoder 계열이 높은 성능을 보임에 따라 결과적으로 다수의 VAE 모델을 사용하였다.

## 3-2. 프로젝트 수행 절차 및 과정

### 1,2주차(3/21~4/01)

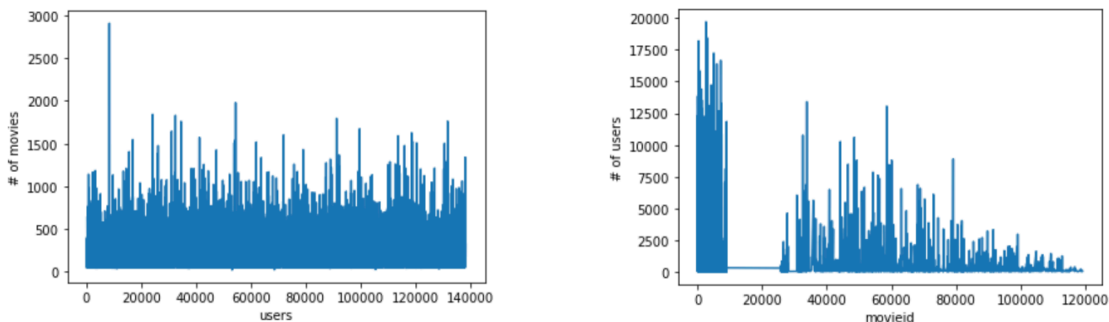
- (3/21~3/25) Baseline, Special Mission 활용하여 다수의 Sequential Model 실험 및 EDA
- (3/28~4/01) Microsoft Recommenders 등 외부 라이브러리 사용 및 GNN 계열 실험

### 3,4주차(4/04~4/14)

- (4/04~4/08) 성능 향상 기간으로 AutoEncoder 계열 모델 실험 및 Static Model 실험
- (4/11~4/14) 다수결, 가중치 부여 다수결 등의 앙상블 기법을 적용 후 최종 결과물 도출

## 3-3. 탐색적 자료 분석 (EDA) 및 전처리

해당 프로젝트는 Implicit Feedback 기반의 sequential recommendation 시나리오를 바탕으로 사용자의 time-ordered sequence에서 일부 item이 누락된 상황을 상정하고 있다. 고유 train user 수는 31360명, 고유 train item 수는 6807개이며 user당 몇 개의 영화를 시청했는지, 각 영화를 몇 명의 user가 시청했는지에 대해 시각화를 진행해보면 아래와 같다.



첫 번째 그래프의 분포를 살펴보면 한 명의 user가 최대 2912개, 최소 16개의 영화를 보았음을 알 수 있다.

두 번째 그래프의 분포를 살펴보면 총 31360명의 user 중 한 영화를 최대 19699명이 보았고, 최소 27명이 보았으며 평균적으로는 757명이 보았음을 알 수 있다.

본 competition에서는 특정 시점 이후의 데이터(sequential)와 특정 시점 이전의 일부 데이터(static)데이터를 임의로 추출하여 정답 데이터로 사용하기 때문에 sequential한 예측에 최적화된 모델과 static한 예측에 최적화된 모델을 다양하게 실험해보며 적합한 모델을 선정하는 것이 중점이라고 판단된다.

## 3-4. 모델 평가 및 개선 과정

Transformer, BPR, SVD, MF, FM, GNN 기반의 다양한 모델들을 실험해보았지만 AutoEncoder 기반의 모델들이 성능이 제일 좋았기 때문에 최종적으로 Mult-VAE, RecVAE, EASE를 가지고 실험을 진행하였다.

### 공통

- AutoEncoder 기반의 모델은 User-Item Matrix를 통해 학습하기 때문에 데이터 전처리를 통해 user id, item id만 남기고 나머지 feature는 모두 제외하였다.
- NDCG@100을 기준으로 best를 기록한 모델을 사용하였다. 대회 평가 Metric은 Recall@10이었지만 실제 제출을 했을 때 Recall@10 best 모델보다 NDCG@100 best 모델의 점수가 0.0001 ~ 0.0002 정도 더 높았다.

### 하이퍼 파라미터

- Mult-VAE: batch size를 64, 128, 256, 500으로 설정하고 learning rate를 0~1e-4 으로 설정해 하이퍼 파라미터 실험을 진행하였다. NDCG@100 점수가 가장 높은 모델은 batch size 500, learning rate 1e-4 모델이었고 최종적으로 0.1428을 기록하였다.
- RecVAE: batch size를 500으로 고정하고 learning rate를 5e-4 → 1e-4으로 변경했을 때 0.1490 → 0.1496로 증가하였다. 이후 batch size를 500 → 128로 줄였을 때 0.1496 → 0.1508으로 증가하였다. 학습시 사용되는 user 수 또한 성능에 영향을 미칠 것이라 생각하였다. heldout 유저수를 3000 → 2000으로 줄였을 때 0.1508 → 0.1518으로 증가하였다.
- EASE: regularization 정도를 250, 350, 400, 500, 600, 750, 2000으로 설정해 실험하였다. 250일 때 0.1594이었고 350, 400, 500, 600일 때 0.1600이었다. 750부터는 성능이 감소하여 0.1597이 되었고 2000일 때 0.1570을 기록하였다.

### 앙상블

Mult-VAE, RecVAE, EASE 모델의 Top-10 Recommendation 결과를 Voting 방식으로 앙상블하여 0.1609를 기록하였다.

## 4. 프로젝트 수행 결과

Model	Optimizer	Learning rate	Batch Size	Best epoch	Others	Recall@10
Mult-VAE	Adam	1e-4	500	717	anneal_cap : 0.2 dropout: 0.5 hidden-dim: 600 latent-dim: 200	0.1428
RecVAE	Adam	1e-4	128	132	hidden-dim : 600 latent-dim : 200 heldout user: 2000	0.1518
EASE					regularization : 500	0.1600

위에 나와 있는 모델 3개를 Model Ensemble 했을 때 Test는 0.1609, Hidden-Test는 0.1601을 기록하였다.

EASE 모델의 Regularization 파라미터를 변경하면서 (400,500,600,350) Recall@10이 0.1600인 모델 4개를 얻을 수 있었다. 같은 모델, 같은 Recall@10을 기록하였더라도 유저마다 적게는 1개에서 많게는 3개의 아이템은 다르게 추천을 하는 것을 확인하였다. 성능이 같기 때문에 그러한 1~3개의 아이템은 성능에 도움이 없다는 가설을 세우고 앙상블을 진행하였다.

결과적으로 총 313600개의 아이템 중 EASE 모델 4개가 동시에 추천했던 300611개의 아이템을 제거한 12989개의 아이템을 RecVAE, Mult-VAE의 High Rank에 위치한 아이템으로 대체하게 되었고 성능이 0.009가 오른 것을 통해 그 12989개의 영화 중 약 117개의 영화를 유저가 선택하였음을 알 수 있었다.

## 5. 자체 평가 의견

- 하동준: mask classification 대회에 비해 많은 모델을 사용해보았고, SOTA 모델들의 코드를 변경 활용해보기도 했다. 이상치 데이터 제외 및 Side Information 을 활용하여 좋은 성적을 내지 못한 것과 wandb 등의 툴 들을 사용 못해본 것이 아쉽다. DKT 에서는 조금 더 체계적인 실험 결과 정리가 필요할 것 같으나 많은 것을 얻은 대회인 것 같다.
- 허민영: 상대적으로 side information이 부족한 경우 GNN 모델의 사용은 적합하지 않다는 것을 깨닫게 되면서 추천시스템에서 Dataset에 따른 적합한 모델 선정이 무척 중요하다는 것을 알게 되었음. 또한 하이퍼 파라미터의 많은 실험 진행을 위해 다음 대회에서 WanB의 Sweep Tool을 사용해 많은 실험을 진행해보고 싶다고 느꼈음
- 임경연 : 이번 프로젝트를 통해 추천시스템에 대해서 더 많이 알게 되었다. 하지만 이전 이미지 분류 PJT와 달리 데이터 전처리에 대해서 많이 다뤄보지 못하고 그냥 모델이 넣고 좋은 모델을 찾는 것에만 몰두한 것 같아 Recsys의 데이터 전처리에 대해서 많이 공부하지 못한 게 아쉬웠다. 하지만 이번 대회를 통해 많이 성장할 수 있었다.
- 소경학 : Bert4Rec의 하이퍼 파라미터를 좀 더 체계적으로 바꿔보면서 실험을 했더라면 하는 아쉬움이 있습니다. 다양한 추천 모델을 사용해 보고 성능을 비교하면서 실험을 한 것이 기억에 남습니다. 다음 DKT 프로젝트에서는 모델을 적용하기 전에 EDA를 좀 더 꼼꼼히 해보고, EDA를 통해 나온 결과에 걸맞은 모델을 적용해 보도록 하겠습니다.
- 노건웅: 데이터를 직접 Pandas를 이용해 전처리를 해보고 추천 시스템 강의를 통해 배운 모델들에 적용하며 추천을 진행해본 것이 좋았던 것 같다. AutoEncoder 기반 모델들만을 이용해 앙상블을 진행하였는데 Sequential 모델도 더하여 앙상블을 진행하면 어떨까하는 아쉬움이 남기도 했다.

## 1. 이번 프로젝트에서 목표는 무엇이었는가?

이번 프로젝트에서 중점을 둔 사항은 3가지였다. 첫번째는 실제 데이터셋을 가공해보고 전처리해보는 것이었고 두번째는 부스트 캠프 강의에서 배운 추천시스템 모델들을 직접 활용해보는 것이었다. 세번째는 Data EDA를 다양하게 진행해보는 것이었다.

## 2. 나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?

- Pandas를 이용하여 데이터를 가공 및 전처리 진행
- BPR, LightFM, MF, BiVAE, Mult-VAE, RecVAE, EASE 등 다양한 모델들을 사용
- 모델들의 논문을 직접 읽어보며 다양한 실험을 진행함
- 하이퍼 파라미터들을 변경해보며 각 모델별 최적의 성능을 찾아내려고 노력함
- Data EDA를 진행하며 Director, Writer, Year에 있는 데이터 결측치들을 확인하였고 Year에 있는 결측치들을 직접 수정함
- Voting 방식으로 다양한 모델들의 앙상블을 진행

## 3. 나는 어떤 방식으로 모델을 개선했는가?

- MF → LightFM → BiVAE → BPR → Mutl-VAE → RecVAE → EASE로 모델들을 변경해보며 성능을 개선함
- batch size, learning rate, train에 쓰이는 user 수 등 하이퍼 파라미터들을 변경해보며 어떤 값이 모델의 성능이 올리는 지 실험함
- user id, Item id 외 genre, director, writer, year 정보와 같은 Side Information을 각각 적용해보며 모델의 성능이 좋게 나오는 지 실험함
- 일반 데이터셋과 Negative Sampling을 적용한 데이터셋을 비교해보며 어떤 샘플링 방법이 성능이 높은지 직접 실험함
- 점수가 높게 나온 모델들의 특징이 각 유저별 장르 분포를 잘 표현한다는 것을 깨닫고 이를 기준으로 여러가지 가설들을 세우고 각 모델들을 직접 평가해보며 모델의 성능을 높이려고 노력함

## 4. 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?

- 개인적으로 public data를 기준으로 0.1600을 달성하였다.
- Sequential 모델도 사용해봤지만 제대로 사용하지 못한다면 오히려 Static 모델보다 성능이 좋지 않다는 것을 알게 되었다.
- Side Information을 적절하게 쓰지 못하면 오히려 모델의 성능을 감소시킨다는 것을 알게 되었다.
- AutoEncoder 기반 모델이 성능이 가장 좋다는 것도 알게 되었다. Representation 성능이 다른 모델들보다 뛰어나기 때문이라고 조심스럽게 추측해본다.

## 5. 전과 비교해서, 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

- 마스크 이미지 분류 대회때보다 Data EDA에 시간을 조금 더 투자하였다. 그 결과 각 feature의 특성을 더 잘 이해하게 되었고 데이터 결측치도 직접 수정을 해볼 수 있었던 것 같다.
- Pandas를 다른 때보다 더 많이 사용해본 것 같다. 데이터 전처리에 필요했지만 모델들의 결과 분석을 위해 Pandas를 사용하면서 Pandas 사용 실력이 더 늘었던 것 같다.
- 다양한 GitHub Repository의 코드들을 대회에 직접 맞게 변경해보면서 오픈 소스를 활용해보는 경험도 가졌다.
- 성능이 오르진 않았지만 Learning Scheduler를 사용해보기도 하였다.

## 6. 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

- 결국 각 모델이 가진 성능은 한계가 있기 때문에 앙상블을 진행해 성능을 높여야했는데 개인적으로 제대로 된 설계 없이 앙상블을 진행해서 성능을 끌어올리지 못해서 아쉬움이 많이 남았다.
- 더 많은 모델을 앙상블 하지 못한 것도 아쉽다.
- Static 모델 뿐만 아니라 Sequential 모델도 같이 사용해서 앙상블을 진행하여야 했는데 그렇지 못해서 아쉬운 것 같다.
- Weight and Bias 를 이용하지 않은 것도 아쉽다.

## 7. 한계/교훈을 바탕으로 다음 P-Stage에서 스스로 새롭게 시도해볼 것은 무엇일까?

- 먼저 협업 톨과 룰을 제대로 설정해 효율을 높이고 싶다.
- 기존에도 Notion을 사용하긴 했지만 보다 체계적으로 정리하여 부족한 부분은 없는지 더 실험해볼 것은 없는지 꼼꼼히 정리해보고 싶다.
- 최대한 다양한 형태의 모델들을 앙상블하여 대회 점수를 높여보고 싶다.
- 확률이나 통계 등 수학적인 부분을 조금 더 공부해서 모델을 개선해보고 싶다.

## 1. 이번 프로젝트에서 목표는 무엇이었는가?

이번 프로젝트에선 추천시스템 모델을 구현하고 실제 대회에서 높은 순위 달성하는 것에 있었다.

## 2. 나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?

목표에 달성하기 위해서 데이터 EDA를 통해서 BERT4Rec과 같은 Sequence Model 보다 일반 FM, VAE 모델이 더 높은 성능이 달성할 수 있겠다고 파악했고, BiVAE 모델과 SOTA 모델 중 H+Vamp 모델을 사용했다. 또한 마지막 성능 향상을 위해 모델에 가중치를 주어 앙상블을 하는 시도까지 할 수 있었다.

## 3. 나는 어떤 방식으로 모델을 개선했는가?

모델 개선 방법은 우선 논문을 보고 해당 논문에서 사용한 하이퍼 파라미터를 시도해보고, 하이퍼 파라미터를 하나하나 변경해가면서 모델 성능이 오르는지 아니면 과적합되는지 파악했고, 최적 파라미터를 찾으려고 했다.

## 4. 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?

내가 만든 모델은 비록 다른 모델에 비해 높은 성능을 달성하지 못해서 결과적으로 사용하지 못했다. 하지만 이번 프로젝트를 진행하면서 느낀 점은 Data 전처리보다 어떤 모델을 발견하고 해당 모델을 빠르게 시도해서 가능성을 발견하는 것에 있다고 생각했다.

## 5. 전과 비교해서, 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

내가 새롭게 시도한 변화는 전에는 무작정 모델에 데이터를 집어넣는 시도를 처음부터 했지만, 이번에는 EDA를 통해서 데이터 특성을 파악하고 어떤 모델이 더 높은 성능을 달성할 것인지 파악하는 작업을 했다. 그래서 실제로 요즘 SOTA 모델인 S3Rec, Bert4Rec 이런 Sequence 모델이 높은 성능을 달성하지 못할 것이라고 파악했고 실제로 해당 모델이 높은 성능을 보여주지 못했다.

## 6. 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

한달동안 진행된 대회에서 집중을 못한 점이 아쉬웠다. 또한 기존 코드에 있는 모델에 대해서 구현을 했지만 기존 reference가 없는 모델에 대해서 완전 새롭게 코드를 짜는 것에 대해서 시도를 했으나 만들어내지 못했다. 이런 점이 아쉬웠고 한계라고 느꼈다.

## 7. 한계/교훈을 바탕으로 다음 P-Stage에서 스스로 새롭게 시도해볼 것은 무엇일까?

다음 P-Stage에선 무조건 처음부터 끝까지 스스로 모델 코드를 짜보려고 한다. 또한 이번 대회에선 Data 전처리를 하지 못했지만 실제로는 data 전처리가 필수라고 생각한다. 따라서 데이터 전처리에 대해서 더 많은 시간을 기울이려고 한다.

## 1. 이번 프로젝트에서 목표는 무엇이었는가?

- 주어진 Baseline Code를 습득하여 내 것으로 만들고 최대한 많은 것을 실험해보기
- Implicit feedback 기반의 행렬을 사용하기에 적합한 모델 탐색하기
- Side information을 어떻게 활용하여야 할지 분석하고 생각해보기

## 2. 나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?

### 개인 학습 측면

- Special Mission을 통해 EDA, Dataset 전처리 및 side information 간의 상관관계 분석
- Side information 간의 상관관계 분석을 통한 GNN 모델(KCGN, pinSAGE) 사용
- Side information (Genre, writer, director, timestamp)을 어떻게 적절히 사용하여야 할지 반복적으로 실험
- Top-K영화를 선정하기 위해 RecVAE 모델 사용
- Batch\_size, optimizer, loss\_function 등의 다양한 조합을 실험

### 공동 학습 측면

- Github를 활용하여 코드를 업로드 하며 협업하는 방법을 익힘
- 서로 질문이 있을 시 Slack과 Zoom을 활용하여 언제든지 도움을 주고받음
- Side information (Genre, writer, director, timestamp)을 어떻게 적절히 사용하여야 할지 반복적으로 실험
- 좋은 결과물을 종합해 앙상블

## 3. 나는 어떤 방식으로 모델을 개선했는가?

- Side information을 활용해보고자 여러가지 GNN 모델을 사용해보았으나 성능이 좋지 못하였음
- GNN 모델은 sequential한 Top-K 추천 과제에서는 적합하지 않다는 것을 깨닫고 RecVAE 모델을 통한 실험을 진행
- Hidden dim, latent dim, batch size, gamma, lr 등의 하이퍼 파라미터를 조정해가며 성능을 높이기 위해 여러가지 조합을 실험해보았음
- 여러 실험 결과 hidden dim 700, latent dim 300, batch 128, gamma 0.001, lr 0.0001의 조합에서 유의미한 결과를 얻을 수 있었음

## 4. 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?

- Side information 분석 결과 genre 정보를 사용하는 것이 적합하겠다고 판단하였으나 실제 결과는 너무 genre 쪽으로 치우치는 결과가 나타났음
- Side information이 적합하지 않을 때는 오히려 문제가 될 수 있음을 알게 되었음. → 이를 통해 Dataset을 좀 더 자세히 들여다보고 모델을 선정해야 하는 안목을 기를 수 있었음
- RecVAE 모델에서 hidden dim 500 → 700으로 조정했을 때, ndcg 값이 0.4475 → 0.4488로 향상되었으며 latent dim 200 → 300으로 조정했을 때, ndcg 값이 0.4488 → 0.4520으로 향상됨을 볼 수 있었음 batch size를 512 → 256 → 128 → 64로 줄이는 과정에서 128이 적합하다는 것을 알게 되었고 이때의 ndcg 값은 0.4598 이었음
- 이를 통해 최대한 많은 하이퍼 파라미터를 실험해보는 것의 중요성을 깨달았음

## 5. 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

- 더 많은 하이퍼 파라미터 실험을 해보지 못한 것이 아쉬움이 남음
- Side information을 활용하는 것에 한계를 느꼈을 때, 이미 많은 시간이 지나 있어서 더 많은 sequential한 모델을 사용해 보지 못한 것에 많은 아쉬움을 느꼈음

## 6. 한계/교훈을 바탕으로 다음 P-Stage에서 스스로 새롭게 시도해볼 것은 무엇일까?

- 시작 전 전반적인 흐름에 대한 계획을 수립하고 Dataset에 좀 더 적합한 모델을 찾아 다양한 실험을 진행해볼 것임
- 효율적인 하이퍼 파라미터를 찾기 위해 WandB의 Sweep Tool을 사용해볼 것임

## 1. 이번 프로젝트에서 목표는 무엇이었는가?

이번 프로젝트의 목표는 다양한 종류의 모델을 대회에 맞게 코드를 수정하고 사용하는 모델에 적합하게 데이터를 전처리 및 정제하는 것에 있었다. 주어진 Side Information 을 사용하는 모델에 맞게 어느 정도까지 포함할지 전처리 과정을 거치는 것이 주된 목표였다. 또한 외부 라이브러리 및 논문 구현 코드를 수정을 거쳐 최적의 모델을 설계하는 것에도 목적이 있었다.

## 2. 나는 내 학습목표를 달성하기 위해 어떤 방식으로 모델을 개선했는가?

- S3Rec, SASRec, Bert4Rec 등 제공된 코드를 대회에 맞게 수정하여 각각의 성능과 최적의 성능을 찾아봄
- Microsoft Recommenders 의 bpr, BiVAE 등 활용할 수 있는 모델을 사용하여 최적의 성능을 찾아봄
- multi-VAE, Rec-VAE, EASE 등 autoencoder 계열의 모델과 SVD++ Time-flipped 모델을 구현 코드를 사용하여 최적의 성능을 찾아봄
- 최고 성능을 보이는 EASE 모델을 k-fold 진행하여 output 을 앙상블 하여 overfitting 을 해결

## 3. 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?

- $k=5$ , k-fold cross validation 을 통해 5개의 submission.csv 을 얻었고 이를 단일 모델 최고 성능인 Recall@10 0.1600을 기록한 EASE(reg=500) 의 submission.csv 과 다수결 모델(threshold=3)을 사용하여 앙상블 하였다. 과정을 통해 public 데이터셋의 overfitting 을 방지하였고 private 데이터셋에서 Recall@10 0.1602 을 기록하여 단일 모델 기준 최고 성능을 달성하였다. 이를 통해 앙상블의 중요성과 함께 overfitting 을 방지하는 것의 중요함을 깨달을 수 있었다.

## 4. 전과 비교해서, 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

- 주어진 데이터를 활용할 수 있는 최적의 모델은 무엇인지, 그리고 모델에 따른 전처리 과정을 겪으며 데이터 자체에 대한 insight을 넓히려 노력했다. 무엇보다 SOTA 에 근접한 많은 모델들을 대회 task 에 맞게 코드 수정을 할 수 있었던 것에 이전 에 비해 실력 향상이 있었다.

## 5. 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

- timestamp 와 영화 개봉 년도의 이상치 등으로 인해 주어진 Side information 을 제대로 활용하지 못하였고, 현재 추천시스템의 트렌드인 graph neural network 계열의 모델들을 사용해 의미 있는 성능을 못 낸 것이 아쉽다.
- 각각의 모델들을 pipeline 으로 구현 하지 못하였고 이로 인해 wandb 등 분석 툴 들을 사용하여 체계적인 기록을 하지 못한 것이 아쉽다.

## 6. 한계/교훈을 바탕으로 다음 P-Stage에서 스스로 새롭게 시도해볼 것은 무엇일까?

- 데이터 EDA 을 통해 주어진 데이터를 가공하여 더 다양한 모델을 사용해 보고 싶다.
- pipeline code 를 제작 및 분석 툴을 사용하여 체계적인 성능 기록을 하고 싶다.
- Asana, 노션 등을 이용하여 팀원들과의 작업 일정 및 결과 기록을 더욱 체계적으로 하고 싶다.

## 1. 이번 프로젝트에서 목표는 무엇이었는가?

이번 프로젝트의 목표는 모델의 성능을 높이기 위한 방법들을 연구함에 있었다. 데이터 전처리부터 하이퍼 파라미터까지 EDA를 통해 얻은 정보들을 토대로 다양한 모델들을 설계해보고 적용해 보는 것이 주된 목표였다. 학습한 내용을 토대로 모델을 A부터 Z까지 설계하는 과정을 이해하는 것에도 목적이 있었다.

## 2. 나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가?

- 하이퍼 파라미터를 다양하게 바꿔보며 최적의 성능을 내는 수치를 찾아봄
- 여러 모델을 데이터셋에 걸맞게 수정하고 실험을 진행함
- 다양한 모델을 앙상블을 해보고 결과를 기록하고 분석함
- EDA를 통해 얻은 insight를 토대로 모델을 설계함

## 3. 나는 어떤 방식으로 모델을 개선했는가?

- 상위 20개 영화를 제거해보며 성능을 개선하였다.
- 높은 성능을 기록한 모델 (EASE, MultiVAE, RecVAE)을 앙상블을 통해 성능을 개선함
- explicit 데이터를 추가해서 모델에 맞게 꿈 데이터를 변환시켜봄
- batch size와 learning rate를 포함한 여러 파라미터들을 조절해보면서 최고의 효율을 낼 수 있는 값을 찾아봄

## 4. 내가 한 행동의 결과로 어떤 지점을 달성하고, 어떠한 깨달음을 얻었는가?

- RecVAE, MultiVAE, EASE를 앙상블을 통해 Recall@10은 0.1609를 달성하였다. 모델을 데이터에 맞게끔 설계를 해보면 구현 능력을 키웠다. 데이터 전처리를 할 때에도 우리가 목표로 하는 것이 무엇인지 확실히 인지하고 했을 때 더욱 효과적인 방법으로 성능을 개선할 수 있는 것을 알게 되었고, 시간 또한 절약할 수 있다는 것을 알게 되었다.

## 5. 전과 비교해서, 내가 새롭게 시도한 변화는 무엇이고, 어떤 효과가 있었는가?

- 문제 정의에 좀 더 시간을 투자했다. 문제 정의를 하지 않고 접근하였을 때 보다 실험을 하는데에 있어서 목적이 뚜렷하게 생겼고 실험을 통한 결과를 분석할 때도 좀 더 효과적으로 할 수 있게 되었다.

## 6. 마주한 한계는 무엇이며, 아쉬웠던 점은 무엇인가?

- 모델간의 차이를 뚜렷하게 밝혀내지 못한 것이 아쉬웠다. 제출횟수에 한계가 있어서 다양한 모델을 여러 경우의 수로 나누어 실험하지 못한 부분이 아쉬웠다.
- 실험결과에 대한 분석을 제대로 하지 못한 것이 아쉬웠다. 성능이 좋게 나온 이유와 안좋게 나온 이유를 뚜렷하게 밝혀내지 못하였다.
- 장르 information을 통해 모델의 성능을 개선시키지 못한 점이 아쉬웠다.
- 찾을 수 있는 모델 중을 데이터 셋에 걸맞게 변환시키지 못한 점이 아쉬웠다.

## 7. 한계/교훈을 바탕으로 다음 P-Stage에서 스스로 새롭게 시도해볼 것은 무엇일까?

- 문제를 올바르게 정의하고 진행할 방향을 정하는데 시간을 더욱 투자해보고 싶다.
- pretrained된 모델보다 문제에 걸맞는 방향으로 모델을 직접 설계해보고 싶다.
- 주제에 대한 파악을 심도 있게 하고, 올바른 방향을 정하고 설계해보고 싶다.
- 베이스라인 코드를 명확하게 분석해보고 코드를 다양하게 바꿔가며 내 것으로 만들어 보고 싶다.