Wrap-up Report

Book Rating Prediction

RecSys 6조

노관옥, 박경원, 이석규, 이진원, 장성준

개요

1. 프로젝트 개요

2. 팀 구성 역할

3. 수행 절차 및 방법

4. 수행 결과

5. 자체 평가 의견

6. 개인 회고

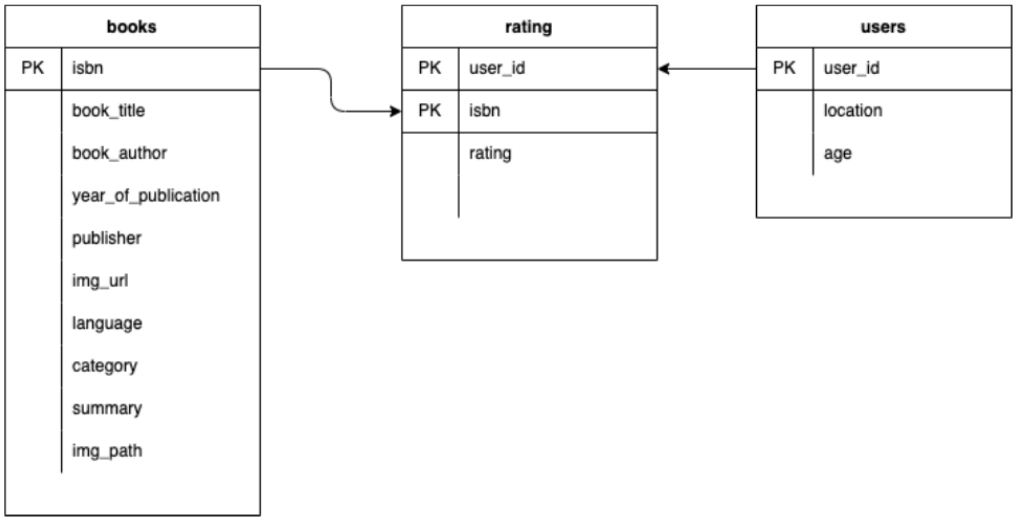
1-1. 프로젝트 개요

● 프로젝트 주제
- Book Rating Prediction
- 사용자의 책 평점 데이터를 바탕으로 사용자가 어떤 책을 더 선호할지 예측
- 평가 지표 : RMSE(Root Mean Squared Error)

● 활용 장비 및 재료
- 개발환경 (AI Stage Server)
  - OS : Ubuntu 18.04.5 LTS
  - GPU : Tesla V100-SXM2-32GB

- 협업 Tool
  - Github / Slack / Zoom

● 사용 데이터셋의 구조



1-2. 프로젝트 팀 구성 및 역할

이번 프로젝트에서는 대회가 처음인 팀원들을 위해 모든 과정을 경험하게 하는 데 중점을 두었습니다. 전체 팀원의 경험 향상에 초점을 두고 첫째 주는 팀원 역할을 나누지 않았습니다. 이를 통해 팀원 각자가 데이터 전처리, 모델 선정, 예측값 제출 등의 중요한 단계들을 직접 수행해보며 실질적인 학습과 경험을 쌓을 수 있는 기회를 제공했습니다.

프로젝트의 첫 주 주말에 전체 팀원이 모여 진행한 회의는 이러한 학습 과정을 더욱 강화하는 데 중요한 역할을 했습니다. 이 회의에서는 각자의 경험을 공유하고, 프로젝트의 진행 방향에 대해 집중적으로 논의했습니다.

마지막 주에는 성준님과 진원님의 멘토 역할해주셨습니다. 팀원들의 개별적인 질의응답 및 결과물에 대한 피드백을 제공함으로써 프로젝트의 질을 높이는 데 크게 기여했습니다. 덕분에 모든 팀원의 학습 경험과 개발 능력에 많은 도움이 되었습니다.

다음 프로젝트에선 프로젝트 시작하면서 팀원의 역할을 분배하고 협업 시너지를 끌어낼 예정입니다.

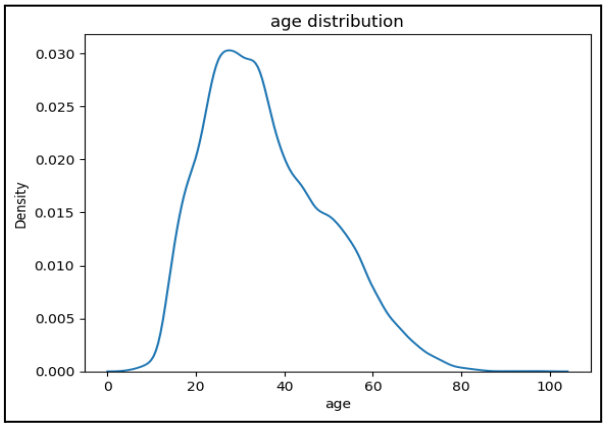
1-3. 프로젝트 수행 절차 및 방법

1) EDA

1. 전체 텍스트 데이터 처리

- 기본 모델에 포함된 텍스트 전처리 함수를 문자열 데이터에서 적용해 범주의 개수를 줄였습니다.

2. 사용자 연령대 분석

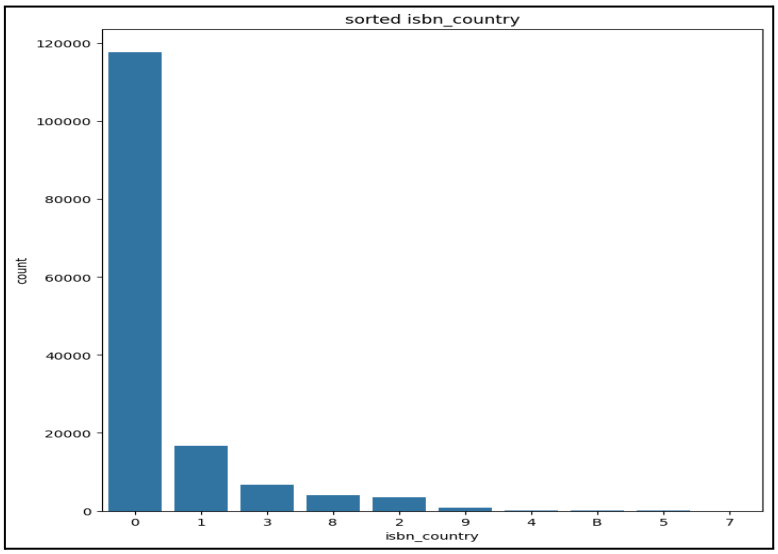



```
------------------------ ANOVA TEST -------------------------
H0 : 연령대 간 Rating의 평균이 동일하다.
H1 : 연령대 간 Rating의 평균이 적어도 하나는 동일하지 않다.

F-statistic : 58.92791
p-value : 2.49362383368452e-108

p-value < alpha이므로 유의수준 5% 하에서 H0를 기각한다.
즉, 연령대 간 Rating의 평균이 적어도 하나는 동일하지 않다.
```

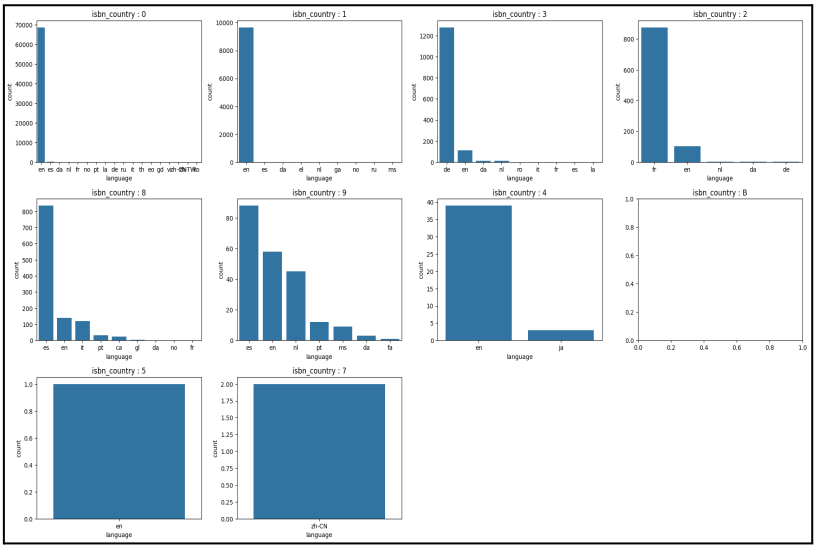
- 전체 이용자 데이터를 시각화하여 대부분의 이용자가 20대에서 40대 사이임을 확인했습니다.
- 성인 이전 연령대에서의 독서 취향 변화를 고려하여 성인 이전은 3년 단위, 성인 이후는 10년 단위로 범주화를 진행했습니다.
- 연령 데이터의 결측치는 평균 연령으로 대체했습니다.
- 범주화한 연령대 그룹 간의 Rating 평균 차이를 확인해보기 위해 ANOVA 검정을 수행하였고, 검정 결과 유의수준 5% 하에서 연령대 간의 Rating의 평균에 차이가 있음을 알 수 있었습니다.

3. 도서 **ISBN** 처리

- ISBN의 첫 번째 숫자를 국가 코드로 사용했습니다.



- 다른 자료에서는 두 번째 숫자까지를 국가 코드라고 하는 경우도 있었으나 첫 번째 숫자만을 국가 코드로 사용하는 것이 성능이 좋아 첫 번째 숫자만을 국가 코드로 사용
- 국가 코드별 언어 분포를 시각화하여 대부분에서 영어 사용이 우세함을 확인했습니다.
- 아마존 ASIN 데이터(국가 코드 'B')에서 언어 분포가 결측인 점을 고려하여 해당 결측치를 최빈값으로 대체했습니다.



- ISBN의 첫 자리를 새로운 국가 코드 특성으로 추가했습니다.

4. 도서 저자 처리

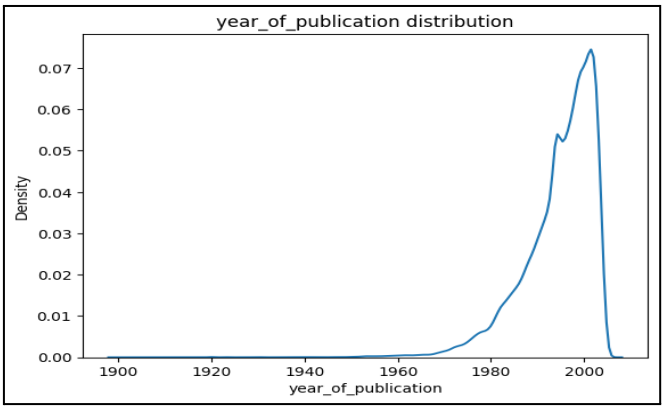
- 저자 칼럼의 결측치가 단 하나였으며, 해당 도서가 실제로 저자가 없음을 확인했습니다.

	isbn	book_title	book_author	year_of_publication	publisher
73737	0751352497	A+ Quiz Masters:01 Earth	NaN	1999.0	Dorling Kindersley

- 따라서 같은 출판사의 최빈값을 사용하여 결측치를 대체했습니다.

5. 출판 년도 처리

- 출판 년도 데이터를 시각화하여 대부분의 도서가 1980년 이후에 출판됨을 확인했습니다.

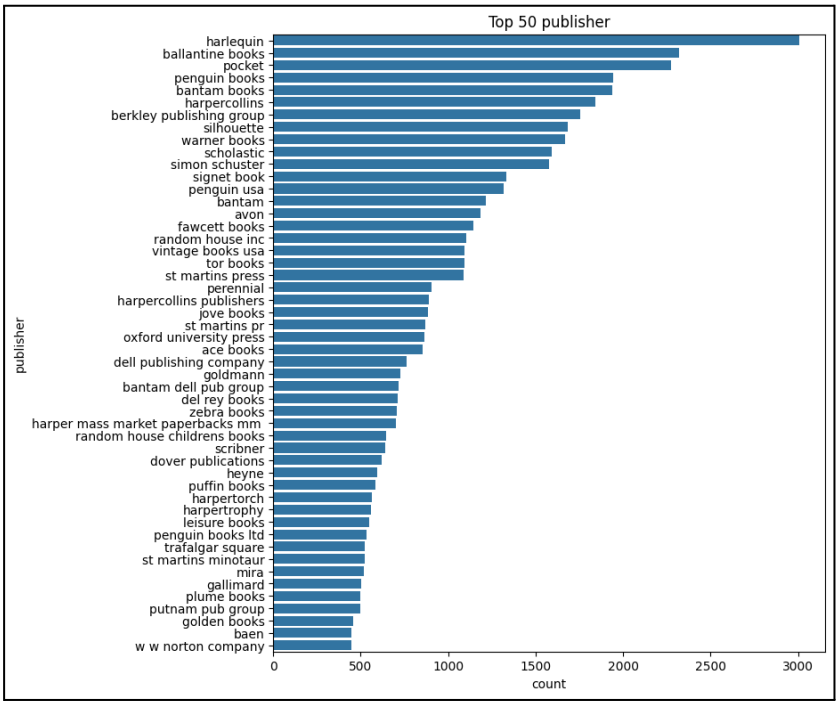



```
------------------------ ANOVA TEST ------------------------
H0 : 출판년도 그룹 간 Rating의 평균이 동일하다.
H1 : 출판년도 그룹 간 Rating의 평균이 적어도 하나는 동일하지 않다.

F-statistic : 66.49107
p-value : 5.347374704534725e-83

p-value < alpha이므로 유의수준 5% 하에서 H0를 기각한다.
즉, 출판년도 그룹 간 Rating의 평균이 적어도 하나는 동일하지 않다.
```

- 1970년 이전과 2000년 이후를 제외하고 나머지를 5년 단위로 범주화했습니다.
- 범주화한 출판년도 그룹 간의 Rating 평균 차이를 확인해보기 위해 ANOVA 검정을 수행하였고, 검정 결과 유의수준 5% 하에서 출판년도 간의 Rating의 평균에 차이가 있음을 알 수 있었습니다.

6. 출판사 처리

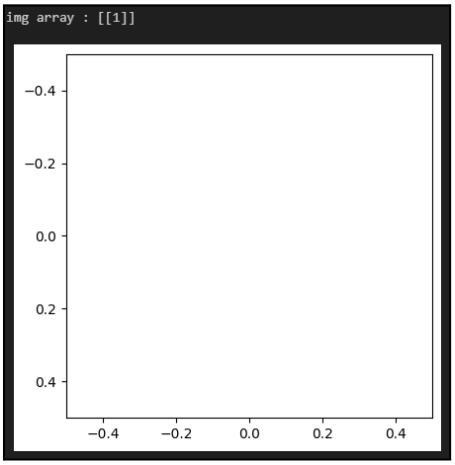
- 상위 50개 출판사를 시각화하여 대부분이 영어로 책을 출간함을 확인했습니다.



- publisher로 language의 결측치를 채우려고 했으나 isbn country를 활용하는 것보다 성능이 안 좋았습니다.

7. 이미지 **URL** 처리

- 책 표지 이미지가 없는 데이터가 존재함을 확인했습니다.



- Binarization된 이미지 배열 길이가 1인 경우를 결측치로 판단하였을 때, 전체 데이터의 28%인 41,802개가 결측치임을 확인했습니다.

8. 도서 카테고리 처리

- 카테고리 결측치가 전체의 약 44.95%를 차지하여, 결측치를 'fiction'으로 대체했습니다.

	category	count
1	fiction	39678
2	biography autobiography	3326
3	history	1927
4	religion	1824
5	nonfiction	1427
6	humor	1291

	category	count
1	fiction	33016
2	juvenile fiction	5835
3	biography autobiography	3326
4	history	1927
5	religion	1818
6	juvenile nonfiction	1418

- 책 표지 이미지와 책의 제목을 활용하여 결측치를 대체하려고 하였으나 이미지의 결측치가 너무 많아 사용하지 않았고 책의 제목은 MLP로 모델링을 진행하였으나 정확도가 46%로 오히려 예측에 안좋은 영향을 줄 것 같아 사용하지 않았습니다.
- category의 범주가 너무 다양하여, 상위 카테고리로 묶어 범주를 축소시켰으며, 5개 이하의 카테고리는 'others'로 분류하여 `category_high` 변수를 생성했습니다.

9. 책 요약 정보 처리

- 책 요약 정보의 결측치가 44.95%로 많아 적절한 대체 방안이 없어 제거하기로 결정했습니다.

```
summary 결측치 개수 : 67227
summary 결측치 비율 : 44.95%
```

2) 모델링 절차 및 결과

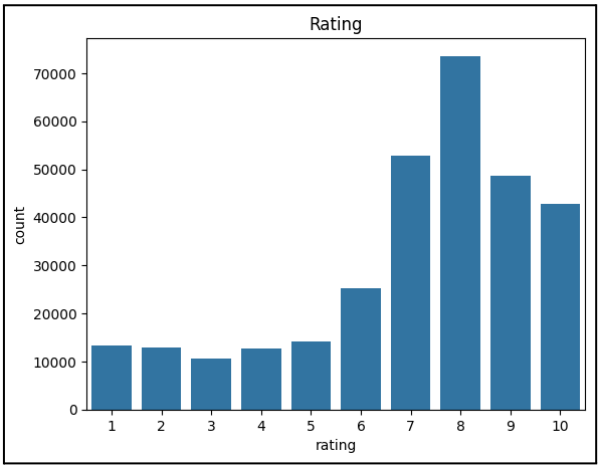
모델링 개요

- 본 프로젝트에서는 모든 변수를 범주화하여 범주형 변수에 대해 효과적인 Gradient Boosting 라이브러리인 Catboost를 활용했습니다.
  특히, 모든 변수를 범주화하기 위해 `publication_of_year`와 `age` 변수도 범주화하였습니다.
- HPO(Hyper Parameter Optimization)는 Optuna를 활용하여 수행했습니다.

Catboost 모델링

- CatBoost 모델에 `CatBoostPruningCallback`을 사용하여 HPO 도중 불필요한 실험을 중단하는 가지치기(Pruning) 기법을 적용했습니다.
- `CatBoostPruningCallback`은 GPU를 지원하지 않습니다.

- 일반적으로 Regression 문제에서는 연속형 Label에 대해 Stratified K-Fold를 지원하지 않지만, 본 프로젝트의 Rating이 이산형(정수)으로 되어있어 Stratified K-Fold를 사용할 수 있었습니다.
- Rating 값의 분포 차이가 크므로 Stratified K-Fold 하는 것이 좋다고 판단하였습니다,



CNN_FM, DeepCoNN 모델링

- 성능 향상을 위해 비정형 데이터(이미지, 텍스트)를 활용하는 CNN_FM과 DeepCoNN 모델을 학습하였습니다.
- 추후에, Catboost 모델과 앙상블을 진행하는 것이 좋다고 판단하였습니다.

1-4. 프로젝트 수행 결과

Feature Engineering (Catboost)

- 유저의 평균 평점인 avg_rating 변수의 Feature Importance가 다른 변수들에 비해 매우 높았지만 다른 변수들 보다 압도적으로 높아 제거를 하였더니 LB Score가 2.1881에서 2.1279로 개선된 것을 확인할 수 있었습니다.
- 유저의 책 리뷰 횟수인 review_counts 변수를 추가했을 때 LB Score가 2.1279에서 2.1257로 개선된 것을 확인할 수 있었습니다.
- isbn의 두 번째 자리까지를 국가 코드(isbn_country)로 사용한 변수를 추가했을 때 LB Score가 2.1257에서 2.1250으로 개선된 것을 확인할 수 있었습니다.
- publisher를 isbn의 출판 코드로 변경하는 것보다 기존의 publisher를 사용했을 때 LB Score가 2.1250에서 2.1235로 개선된 것을 확인할 수 있었습니다.
- isbn의 두 번째 자리까지를 isbn_country로 사용하는 것보다 첫 번째 자리만을 isbn_country로 사용하는 것이 LB Score가 2.1235에서 2.1226으로 개선된 것을 확인할 수 있었습니다.

비정형데이터 활용 **(CNN_FM, DeepCoNN)**

- 또한, 성능 향상을 위해 이미지와 텍스트 데이터를 활용하여 학습하는 CNN_FM과 DeepCoNN을 활용했습니다.
- Optuna를 활용한 HPO를 수행한 결과, CNN_FM과 DeepCoNN의 LB Score는 각각 2.1739, 2.2211로 Catboost 모델보다 많이 좋지 않았습니다.

앙상블 모델링

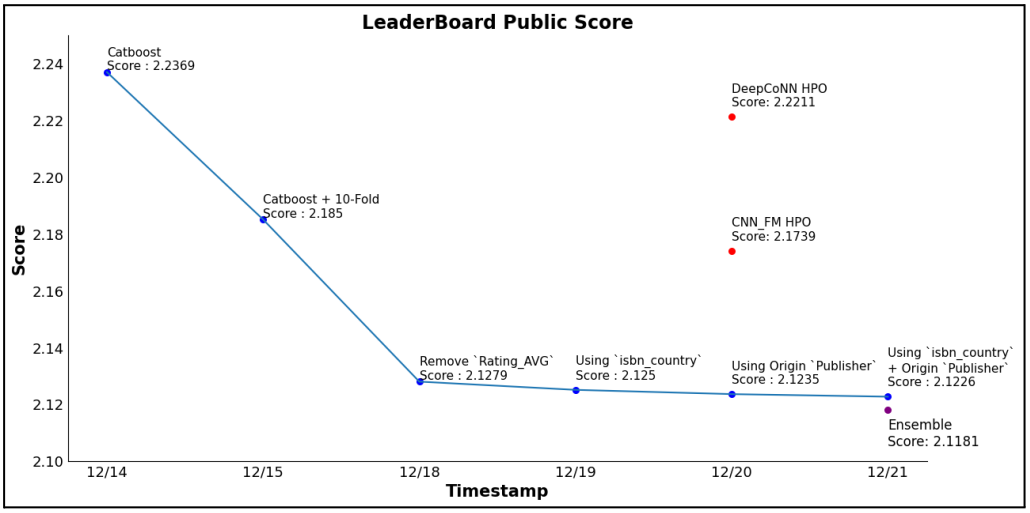
- 성능을 고려하여 가중치를 주어 앙상블을 진행하면 더 좋은 결과를 얻을 수 있을 것이라고 판단하였고, Catboost : CNN_FM : DeepCoNN = 0.8 : 0.15 : 0.05의 비율로 앙상블을 진행하였습니다.
- 결과적으로 LB Score는 2.1187로 성능이 소폭 향상되었습니다.

모델	Catboost	CNN_FM	DeepCoNN
LB Score	2.1226	2.1739	2.2211

Ensemble	
LB Score	**2.1187**

- Tree-Based 모델인 Catboost와 달리 Deep Learning 모델인 CNN_FM과 DeepCoNN은 비선형성을 고려한다는 점이, Catboost가 고려하지 못한 유의미한 특징을 얻을 수 있다고 생각하였습니다.

- 프로젝트의 과정
  - 탐색적 분석 및 전처리 (학습데이터 소개)
    - 각 변수별 전처리 방법을 고안
  - 모델 개요
    - 모델 선정 및 분석
    - 데이터가 대부분 범주형 변수
    - 범주형 변수에 대해 효과적인 Gradient Boosting 라이브러리인 Catboost
    - 이미지, 텍스트 데이터를 활용하는 CNN_FM, DeepCoNN
  - 모델 평가 및 개선
    - Optuna를 활용하여 HPO(Hyper Parameter Optimization) 수행
    - 다양한 Feature Engineering을 수행하며 모델의 성능을 향상
  - 시연 결과



1-5. 자체 평가 의견

○ 잘한 점들
- 열정도 1등, 성능도 1등 모든 것을 다 얻은 대회였다고 생각합니다.

○ 시도 했으나 잘 되지 않았던 것들
- book_title를 활용하여 category를 예측하였던 것이 성능이 좋지 않아 사용하지 못했다.
- 실수 데이터를 계속 embedding하려고 시도했다. 다음부턴 모델을 수정하던지 데이터를 범주형으로 바꿔야겠다.

○ 아쉬웠던 점들
- Feature Engineering을 진행할 때 트리 모델 기반의 Feature Importance의 Default 값인 'gain' 외에 지니계수 같은 다른 방법을 적용하지 못했다.
- 책의 표지를 사용해 category를 예측하려고 했으나 이미지 결측치가 많아 사용하지 못했는데 시간이 촉박하여 이미지 결측치를 처리하는 방법에 대해 생각하지 못했다.
- Wandb를 사용하지 못했다.

○ 프로젝트를 통해 배운 점 또는 시사점
- Wandb & Notion 등 협업 툴을 더 많이 사용해보고자 다짐하게 되었다.

노관옥

집중적으로 공부한 내용

1. ISBN 속성에 대한 연구를 했습니다.

학습 데이터셋에는 출판사와, 언어가 있었습니다. ISBN에는 이 속성들이 들어있죠

ISBN은 기본적으로 <국가>-<출판사>-<항목>-<확인숫자> 포맷으로 구성돼있는데 각 속성들은 '-' delimiter로 구분됩니다.

여기서 여기서 추출할 수 있는 건 국가와 출판사였습니다.

파이썬 패키지중 isbnlib의 mask 메서드를 사용해서 국가와 출판사를 쉽게 추출할 수 있었습니다.

하지만 평균 RMSE는 2.134…가 나오면서 isbn의 앞자리**1**자리만 추출한 결과보다 약간 낮은 결과가 나왔습니다**.**

→ Catboost같은 범주형 데이터에 강점이 있는 모델에서 결국 작은 단위로 내려가는 것보다 앞자리를 추출하는 것이 더 좋다는 추측을 했습니다.

2. Review Count 미션내용 반영

EDA 미션에서 heavy 유저들이 상대적으로 10점을 잘 안준다는 분석을 적용해봤습니다.

앞으로 하고 싶은 것

1. 실험 관리를 제대로 해보고 싶습니다 .

W&B를 진원님께서 도입을 해주셨지만 제대로 활용하지 않았던 것 같습니다. 다음 프로젝트땐 W&B를 구축하여 실험관리를 더 잘 해보고 싶습니다.

2. 데이터 프로세싱을 더 빠르게 하면 좋겠다는 생각이 들었습니다.

이번 프로젝트를 진행하면서 데이터 처리가 느리다는 느낌을 받아서 좀 더 효율적으로 처리하는 방식을 통해 시간을 많이 아껴보고 싶습니다.

회고

이번 대회에는 각자 전체 프로세스를 한 번 경험해보는 게 목적이었습니다.

이미 경험이 많은 팀원분이 있어서 코드를 보고 처음부터 끝까지 따라할 수 있었습니다.

덕분에 목표로 했던 제출을 성공적으로 했고, 첫 대회를 잘 마친 것 같아서 이후에는 역할 분담을 잘 해서 기존에 하고싶었던 MLOps에 기여하고 싶습니다

박경원

○ 학습목표
베이스라인 코드를 이해하기 위해 데이터 전처리 전체 모델을 한 번씩 학습시켜보고 조금씩 개선해 나가기. 데이터 EDA란 무엇인지 직접 전처리하고 팀원들과 공유했습니다.

○ 마주한 한계와 아쉬웠던 점
지난 6~7주간 빠르게 강의 듣고, 공부하면서 많이 배웠다고 생각했지만, 놓친 부분도 많은 것 같습니다. 베이스라인을 접하면서 모델 자체에 대한 이해나 전체적인 흐름에 대한 개념이 부족하다고 느꼈습니다. 이로 인해서 데이터를 조금씩 변경하고 모델에 넣으면서 생긴 오류들이 많이 발생했습니다. 이것에 대처하는데 시간을 많이 잡아 먹어, 모델에 집중할 시간이 부족하여 아쉽습니다.

○ 한계/교훈을 바탕으로 다음 프로젝트에서 시도해보고 싶은 점
이번 프로젝트에선 EDA에 시간을 많이 쏟았습니다. 다음 프로젝트에선 모델을 수정하는 것에 집중해서 좋은 결과를 얻어내고 싶습니다. 이렇게 해서 모델에 대한 이해도를 많이 높이고 싶습니다.

○ 데이터 개선과정
미션에선 사용자 지역 데이터는 3개 값으로 표현된 게 대부분이지만, 일부 데이터는 5개도 많았습니다. 그래서 열을 5개로 늘려서 결측치는 많더라도 조금 정확하게 나누고자 했습니다.
사용자가 책에 부여한 점수 개수, 점수의 최빈값 그리고 책의 평균점수 등 점수와 관련된 것들이 의미가 있을 것 같아 추가했습니다.

이렇게 데이터를 직접 전처리하는 과정에서 pandas 관련 숙련도가 많이 상승했습니다. 하지만 이것들이 loss를 줄이거나 하는 유의미한 결과를 만들어내지는 못한 것 같아 아쉬웠습니다. hyperparameter를 수정하는 것이 조금이나마 loss를 줄일 수 있었습니다.

○ 실패의 과정에서의 교훈
책 데이터에 언어가 결측치인 것들이 많았고 해당하는 부분을 채워넣으면 결과가 좋아질 것이라 예상했습니다. 'fasttext'를 사용해서 문장을 입력하면 그 언어와 0~1 신뢰도를 알 수 있었습니다. 이 값을 언어 열의 결측치에 채우고 신뢰도 열을 만들어서, 기존에 언어가 있는 채워진 행은 신뢰도 1, 결측된 곳은 fasttext의 신뢰도 값으로 채워넣었습니다.

그리고 모델을 돌려보려 했지만, 실수값을 임베딩 할 때 문제가 생겼습니다. 문자, 정수 데이터들을 모두 인덱싱 후 임베딩합니다. 하지만 실수형 데이터를 추가했을 때 임베딩할 방법을 몰라서 자료를 찾고 있었습니다. 이번 대회에 이것만 해결하고 결과에 많은 것 싶었지만, 시간 내에 할 수 없었습니다. 다음 프로젝트에선 좋은 결과를 만들 수 있길 노력하겠습니다!

이석규

목표 설정 및 진행 과정

딥러닝과 머신러닝 분야에서 처음부터 완벽한 능숙함을 기대하는 것은 비현실적이지만, 지속적인 탐구와 발전을 하기위해 노력했습니다. 이러한 인식 하에, 저는 다음과 같은 구체적인 학습 목표를 설정했습니다:

- **Competition** 경험.
- 이론적 지식을 실제 상황에 적용하는 능력 향상.
- 데이터 시각화 기술 연마.
- 탐색적 데이터 분석(**EDA**)능력 향상.

이 목표들을 추진하기 위해, 저는 여러 리소스와 방법을 활용했습니다. Data 분석 대회 에 익숙해지기 위해 Kaggle 대회에 참가했고, 동료들과의 피어세션시간을 통해 데이터 분석을 통한 feature 구성의 중요성을 깨달았습니다. 이론적으로 배운 모델과 API에 대한 깊은 이해를 목표로 삼았으며, 데이터 시각화 기법에 대해서는 **Seaborn**을 중점적으로 학습했습니다. 또한, EDA 과정에서는 단순한 가시적 분석을 넘어서, 서로 상호작용하는 데이터 특성들을 분석하는 능력을 개발했습니다.

마주한 도전과 개선의 여지

짧은 시간 내에 학습한 이론을 실제로 적용하는 과정은 여러 도전을 내포하고 있었습니다. 특히, 코드 작성의 실수가 시간 소모의 주요 원인이 되기도 했습니다. 이러한 경험을 통해, 문제 해결 방법의 개선과 실력 향상의 필요성을 깨달았습니다. 또한, 데이터 분석과 설계 단계에서 팀원과의 충분한 상의의 중요성을 인식했습니다.

이번 프로젝트는 개개인 전체가 모든 과정을 경험하기로 해서 개인적으로 모델링을 하며 스터디하는 방향성을 가지고 진행했지만, 다음 프로젝트부터 '팀'이라는 이점을 살려 협업과, 분업을 통한 시너지를 만들 계획입니다.

협업 과정 중 성공적인 요소

프로젝트 진행 과정에서의 중요한 성공 요인 중 하나는 멘토님의 도움으로 가능했던 주말 대면 모임이었습니다. 이 모임은 비록 짧은 시간이었지만, 팀원 간의 상호 이해를 증진시키고 창의적인 아이디어 교환에 크게 기여했습니다. 이러한 대면 시간은 서로의 생각과 전략을 더욱 효과적으로 공유할 수 있는 기회를 제공했으며, 이는 전체 프로젝트에 긍정적인 영향을 미쳤습니다.

부스트캠프의 특성상 모든 작업을 대면으로 수행하기는 어려울 수 있지만, 이번 경험을 바탕으로 앞으로도 간헐적인 대면 모임을 계획하여 프로젝트의 질을 높이고 팀원 간의 협력을 강화할 수 있을 것입니다. 이러한 대면 모임은 팀원들 간의 의사소통을 원활하게 하고, 개별적인 기술 및 아이디어를 효율적으로 통합하는 데 기여함으로써 프로젝트의

성공에 결정적인 역할을 할 수 있습니다.

다음 단계: 전략과 계획

현재 Kaggle 대회 참가를 통해 경쟁 환경에 대한 이해를 심화시키고 있습니다. 첫 프로젝트로 진행한 Titanic 과제에서 Data Visualization을 통해 bronze medal을 받게 되었습니다. 다른사람들에게 insight와 데이터 전처리의 초석을 제공하였습니다. 이를 통해 Project에서 Feature사용 방안, 수정 방안을 제시하려면 상대방을 설득함으로써 동의와 공감을 얻는 것도 실제 분석 능력만큼 중요한 것을 깨달았습니다.

이번 프로젝트에서는 2.4703(RMSE)에서 2.1733(RMSE)로 성능 향상을 이루었습니다. 이는 창의적인 EDA 과정과 데이터 전처리 전략의 성공적 적용에서 비롯되었습니다. 앞으로도 이러한 방법론을 활용하여, 데이터 분석 프로젝트에 팀원의 공감을 이끌 수 있는 신선한 접근 방식을 제시하고자 합니다.

22	Finished	석규	2.2403	상세 보기	2023-12-16 20:25			
45	Finished	석규	2.1733	상세 보기	2023-12-19 15:53			

장성준

나는 내 학습목표를 달성하기 위해 무엇을 어떻게 했는가**?**
EDA, Feature Engineering, modeling

마주한 한계는 무엇이며**,** 아쉬웠던 점은 무엇인가**?**
book category의 결측치를 책의 표지와 책 제목으로 채우려고 했으나 책 표지는 결측치가 많아 제외하였고 MLP로 책의 제목으로 책의 카테고리를 채우는 모델링을 해봤지만, 성능이 좋지 않아 사용하지 않고 최빈값인 'fiction'을 사용하였다.

user data의 location을 country, state, city로 나누어 전처리를 진행하였지만, 결측치와 유저가 직접 지역을 입력하는 데이터라서 일관적이지 못해 우선순위에 밀려 세심하게 전처리를 진행하지 못했다. 프로젝트가 다 끝난 뒤에 이런 데이터를 어떻게 처리하는지 찾아봐야 할 것 같다.

한계**/**교훈을 바탕으로 다음 프로젝트에서 시도해보고 싶은 점은 무엇인가**?**
다음 프로젝트부터는 MLOps tool인 wandb를 연동해 모델 실험 관리를 해보고 싶다.

나는 어떤 방식으로 모델을 개선했는가**?**
데이터 전처리와 feature engineering을 통해 성능을 개선했다.

isbn의 경우 국가 코드를 확인할 수 있었는데 첫 번째 2자리가 국가 코드라고 하는 경우도 있고 첫 번째 자리의 숫자만 국가 코드라고 하는 경우도 있어서 시각화 해본 결과 분포는 비슷했다. 하지만 모델링의 결과로 보면 첫 번째 자리만 국가 코드로 사용했을 때 RMSE가 2.1226으로 2.1235보다 작게 나와서 첫 번째 자리를 국가 코드로 선정하였다.

language의 경우 국가 코드로 결측치를 처리하는 방법과 publisher별로 처리하는 방법 2가지를 고려했다. 하지만 시각화해 본 결과 language의 분포를 isbn_country에서 잘 표현한다고 생각하여 language의 결측치를 isbn_country로 대체하였다.

user의 나이 분포 같은 경우 성인 이전 나이에 대해서는 취향이 급격하게 변하기 때문에 3년 단위로 범주화를 진행했고 이후에는 10년 단위로 범주화를 진행했다. 결측치는 평균 나이로 대체하였다.

새로운 feature를 생성하는 방법으로 성능을 개선해보 '책의 평균 평점', '유저의 평균 평점', '유저가 가장 많이 선택한 카테고리', '유저의 리뷰 횟수', '유저의 언어'를 새로운 feature로

추가해 봤지만 '유저의 리뷰 횟수'를 제외하고는 오히려 LB와 CV 간의 차이가 커지고 RMSE가 높아져서 제외하였다.

내가 해본 시도 중 어떠한 실패를 경험했는가**?** 실패의 과정에서 어떠한 교훈을 얻었는가**?**
카테고리 결측치를 채우려고 텍스트 데이터('저자', '책 제목')과 책 표지를 활용해 MultiModal을 사용하려 했지만 책 표지 결측치도 많이 있어 사용하지 못했고 텍스트 데이터를 활용하여 모델링을 해봤지만 성능이 좋지 않아 사용하지 못했다. 시간이 더 있었으면 이미지의 결측치를 어떻게 채울지 고민해 보고 모델링을 어떻게 해야 성능이 더 좋아질까에 대해 깊이 고민해 봤을 거 같다.

협업 과정에서 잘된 점**/** 아쉬웠던 점은 어떤 점이 있는가**?**
잘된 점
vscode의 remoete ssh 익스텐션을 사용해 리눅스 원격 서버와 연결까지는 성공했으나 Github 연동을 하는 과정에서 어려움을 느꼈다. 하지만 팀원들이 연동하는 방법을 상세하게 알려줘서 무사히 연동할 수 있어서 좋았다.

이번 대회에서 모든 팀원들이 전체 데이터에 대한 전반적인 EDA와 modeling을 경험하는 것이 목표였는데 모든 팀원들이 전반적인 데이터 분석 및 모델링 과정을 경험을 할 수 있어서 좋았다.

여러 가지 실험을 통해 Feature Engineering을 진행했는데 isbn_country와 review_counts를 추가하고 출판 년도와 유저의 나이대를 범주화하여 성능을 개선했다는 점이 좋았다.

아쉬운 점
이번 협업을 진행하면서 모델 성능에 대한 공유를 Slack이나 피어 세션 때 공유했는데 다음 프로젝트부터는 Wandb로 해야 할 필요성을 느꼈다.

발표할 것이라 생각하지 못해서 급하게 발표 자료를 만들어서 발표 자료의 디테일이 좋지 못했다. 다음부터는 세심하게 작성하고 검토를 해봐야겠다.

이진원

모델 성능 고도화 과정 속에서의 느낀 점

이번 대회에서는 기본적인 Feature Engineering을 진행하고, 모델링을 중점적으로 수행하였다.

대회를 진행하는 과정에서 모델의 하이퍼파라미터 튜닝보다는 **Critical**한 변수를 생성하거나**,** 전처리를 하는 것이 모델의 성능을 향상시키는 데에 더 중요하다고 판단하였다. 그래서, 다양한 변수 생성 및 변수별로 전처리 기법도 다르게 하며 모델링을 한 후 변수들의 최적의 조합을 찾기 위해 노력했다.

Baseline 코드로 주어진 모델의 성능은 크게 좋지 않았다. 따라서, 모델을 고려하지 않고 데이터를 기반으로 가장 최적의 모델이 무엇일지 생각하였고, year_of_publication과 age를 제외한 모든 변수들이 범주형 변수이므로, year_of_publication과 age도 범주형으로 변환하여 범주형 변수에 가장 효과적인 **Catboost** 모델을 주요 모델로 선정하였다.

또한, 다른 GBM(Gradient Boosting Machine) 계열의 모델인 LGBM과 XGBoost의 학습도 진행하였는데, LGBM의 경우 학습 속도는 빠르지만 성능이 잘 나오지 않았고, XGBoost는 학습 속도가 Catboost와 LGBM에 비해 현저히 느려 주요 모델로 선정하지 않았다.

Catboost 모델을 주요 모델로 선정한 이후, 다양한 변수와 전처리 기법을 고려하면서 학습을 진행하였고, Bayesian Optimization 기법을 사용하는 Optuna 라이브러리를 활용하여 Hyperparameter 최적화를 진행하였다. 모든 실험은 **Catboost** 모델 **+ Optuna HPO + 10-Fold Cross Validation**으로 진행하였고, 각각의 모델에 대해 Feature Importance를 확인하며 변수를 선정하였다.

다양한 변수를 고려하며 실험을 하는 도중 유저별 평균 평점(rating_avg)을 추가하여 모델을 학습시켰을 때, `avg_rating` 변수의 Feature Importance 값이 다른 변수에 비해 매우 높게 나왔으며, CV = 1.766 / LB = 2.185으로 리더보드 3등을 기록하였다. 하지만, LB-CV의 Gap이 너무 큰 것에 이상함을 느껴 동일한 Hyperparameter로 고정하여 `avg_rating` 변수를 제거한 후에 모델을 학습시켜보았을 때, **CV = 2.129 / LB = 2.1305**로 **LB-CV** 간의 **Gap**도 줄어듦과 동시에 리더보드 **1**등을 기록하였다. 여기서, 모델이 특정 변수의 지나치게 의존하면 모델의 학습이 더 잘 안될 수도 있다는 점을 깨닫게 되었다.

이와 더불어, 모델의 성능을 더욱 끌어올리기 위해 이미지 데이터와 텍스트 데이터도 활용해보고자 하였다. Baseline 코드에서 주어진 CNN_FM 모델과 DeepCoNN 모델의 성능을 끌어올리기 위해 Optuna를 활용하여 CNN_FM 모델과 DeepCoNN의 모델의 Hyperparameter 최적화를 진행하였다. 결과적으로, CNN_FM 모델의 경우 CV = 2.1675 / LB = 2.1739, DeepCoNN의 경우 CV = 2.2161 / LB = 2.2211으로 Baseline으로 주어진 모델보다 성능을 향상시켰지만, Catboost 모델의 성능에는 현저히 미치지 못하였다.

하지만, 모델의 성능이 좋지 않다고 모델을 사용하지 않는 것보다 앙상블을 하면 더욱 좋은 성능이 나올 것이라고 판단하였다. CNN_FM과 DeepCoNN은 딥러닝 기반 모델로, 변수 간 비선형성을 고려하기 때문에, Tree 기반 모델인 Catboost가 고려하지 못한 것들을 모델에 반영하였을 것이라고 생각하였다. 따라서, LB 성능에 따라 가중치를 주어 Catboost, CNN_FM, DeepCoNN 모델의 앙상블을 진행하였고 **LB = 2.1181**이라는 결과를 얻게 되었다.

길다면 길고, 짧다면 짧은 2주 간의 대회 기간 동안 여러가지 다양한 변수와 전처리 기법, 적어도 100번

이상의 모델링 Test를 함께 해준 팀원들 덕분에 좋은 성과를 내며 마무리할 수 있었다고 생각한다. 다음 대회에서는 Wandb와 Notion을 더욱 많이 활용하여 보다 더 완벽한 역할 분담과 팀플레이로 다시 한 번 성과를 내고 싶다.