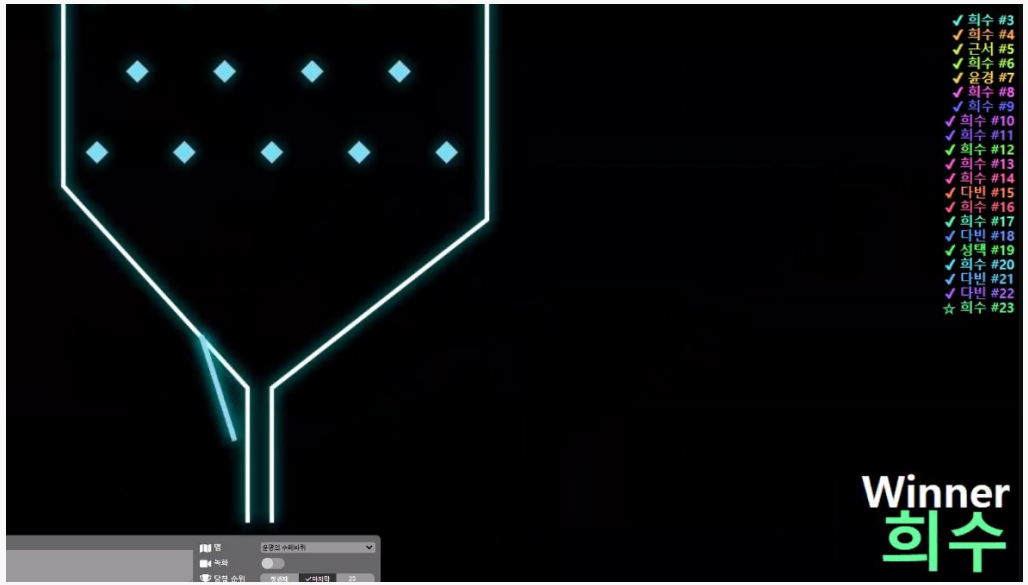
2024.11.08

Book Rating Prediction

RecSys_03 냉장고를 부탁해



Index

목차

01

Project Overview

프로젝트 개요

upstage AI Stages

AI 대회 / Book Rating Prediction

Book Rating Prediction

사용자의 책 평점 데이터를 바탕으로 사용자가 어떤 책을 더 선호할지 예측하는 태스크입니다.

#부스트캠프7기 #Tabular_RecSys

60 | 2024.10.30 10:00 ~ 2024.11.07 19:00 • 종료

RecSys Stages

책과 관련된 정보와 소비자의 정보, 그리고 소비자가 실제로 부여한 평점 데이터셋을 활용하여
각 사용자가 주어진 책에 대해 얼마나 평점을 부여할지 예측하는 문제에 도전했습니다.

Stacks

- Python

Tools

- Git
- WandB

Collaboration

- GitHub
- Notion
- Zoom
- Slack

01

프로젝트 개요

POINT.01 프로젝트 개요

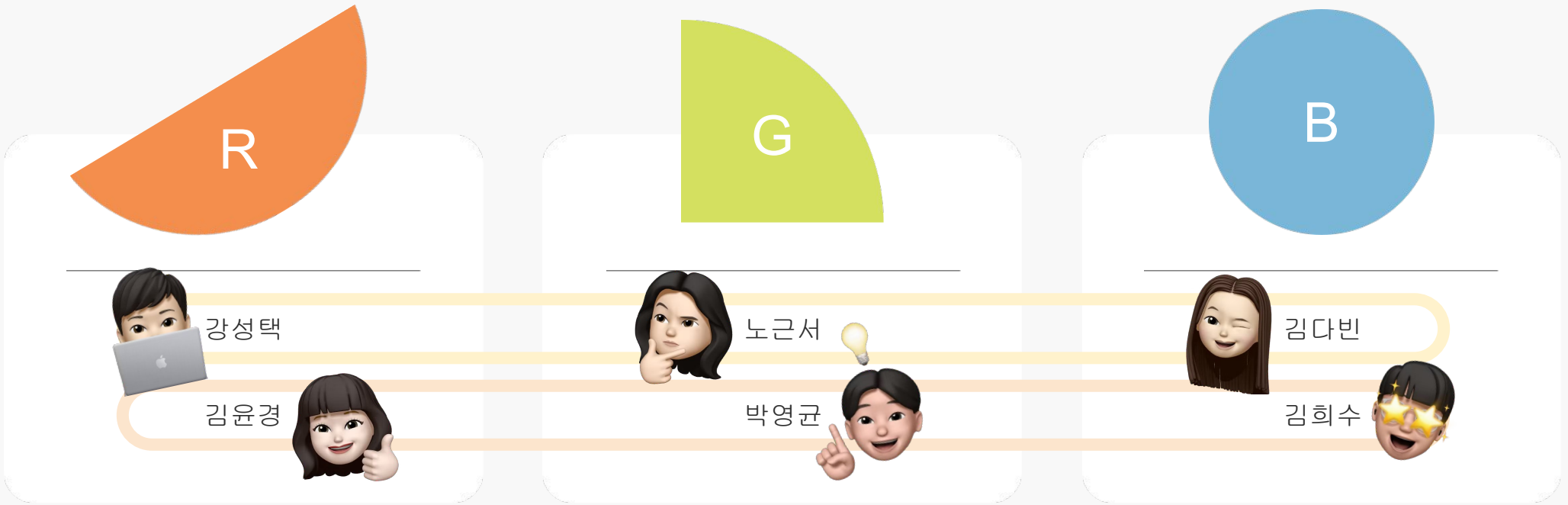
POINT.02 팀 구성 및 역할

01

Team Structure and Roles

팀 구성 및 역할

Data 전처리 팀 / Model 팀 (3:3) 으로 나눈 후, 각 팀마다 1명씩 짝지어 페어 프로그래밍



추가로, 협업을 위한 개인 역할 지정

- Notion 회의록 서기 : 성택(Data), 희수(Model)
- Github 관리자 : 영균(PM), 근서(Data), 윤경(Model)
- WandB 관리 및 기타 : 다빈

02

프로젝트 수행 절차 및 방법

POINT.01 프로젝트 수행 계획

POINT.02 협업 방식

02

Planning

프로젝트 수행 계획

1. Project Rule, Convention 설정
2. 프로젝트 개발 환경 구축
3. 데이터 이해 및 EDA
4. 데이터 전처리, 모델 코드 구현 및 모듈화
5. EDA별 실험
6. 모델 튜닝, 앙상블
7. 코드 리팩토링 & 최종 모듈화

02

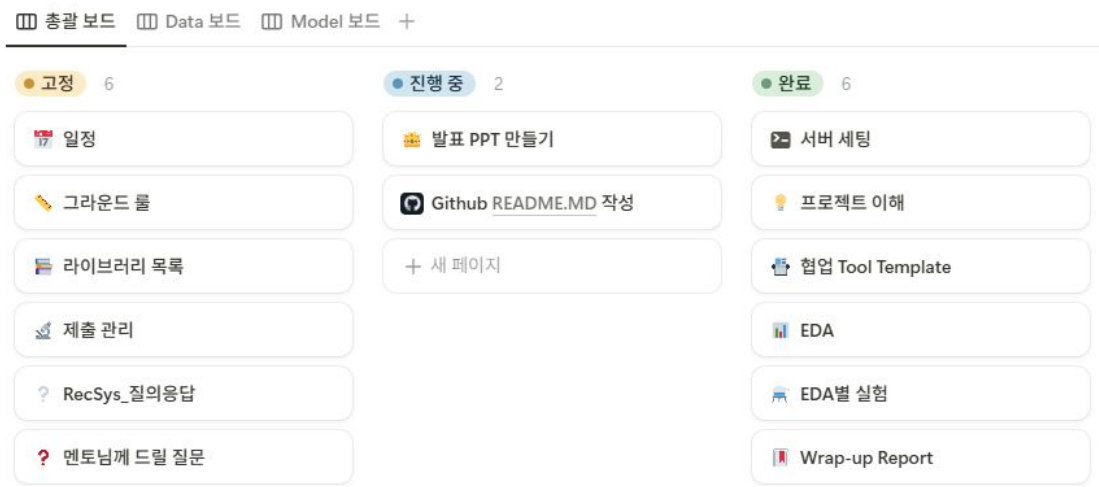
Collaboration

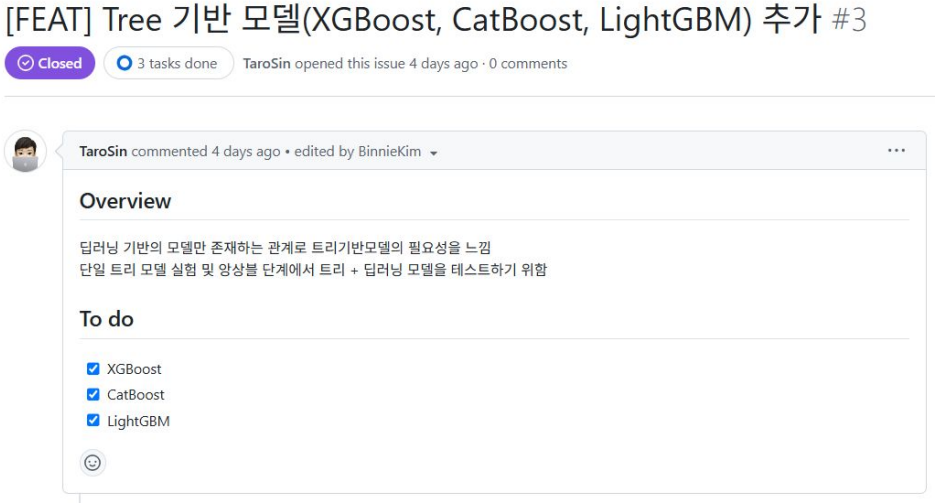
협업 방식

Issue 관리

노션 데이터베이스 보드 및 Github Issue로 관리

- 총괄 보드 : 전체 진행상황 관리 (실험관리 포함)
- Data 보드 : 데이터 전처리 함수 개발 진행상황 관리
- Model 보드 : Model 및 앙상블 개발 진행상황 관리





02

Collaboration

협업 방식

Github 관리 Github Convention에 따라 관리

main branch는 배포이력을 관리하기 위해 사용,
house branch는 기능 개발을 위한 branch들을 병합(merge)하기 위해 사용

모든 기능이 추가되고 버그가 수정되어 배포 가능한 안정적인 상태라면 *house* branch에 병합 (merge)

작업을 할 때에는 개인의 branch를 통해 작업

EDA

branch명 형식은 "**EDA-자기이름**" 으로 작성 ex) EDA-TaroSin

파일명 형식은 "**name_EDA**" 으로 작성 ex) TaroSin_EDA

데이터 전처리팀 branch 관리 규칙

```
book
└── data
```

시간 관계상 하나의 branch에서 진행

모델팀 branch 관리 규칙

```
book
└── model
    ├── model-modularization   # model 개발 및 모듈화 작업
    ├── model-stratifiedkfold  # stratifiedkfold 로직 개발
    ├── model-optuna           # optuna 로직 개발
    └── model-experiment       # 모델 실험
```

master(main) Branch에 Pull request를 하는 것이 아닌,
data Branch 또는 ***model*** Branch에 Pull request 요청

commit message는 아래와 같이 구분해서 작성 (한글)
ex) git commit -m "**docs**: {내용} 문서 작성"
ex) git commit -m "**feat**: {내용} 추가"
ex) git commit -m "**fix**: {내용} 수정"
ex) git commit -m "**test**: {내용} 테스트"

pull request merge 담당자 : data - **근서** / model - **윤경** / **최종** - **영균**
나머지는 ***house*** branch 건드리지 말 것!

merge commit message는 아래와 같이 작성
ex) "**merge**: {내용} 병합"

Issues, **Pull request**는 Template에 맞추어 작성 (커스텀 Labels 사용)
Issues → 작업 → PR 순으로 진행

02

Collaboration

협업 방식

코드 관리

Code Convention에 따라 관리

- 문자열을 처리할 때는 작은 따옴표를 사용하도록 합니다.
- 클래스명은 카멜케이스(CamelCase) 로 작성합니다.
  함수명, 변수명은 스네이크케이스(snake_case)로 작성합니다.
- 객체의 이름은 해당 객체의 기능을 잘 설명하는 것으로 정합니다.
- 가독성을 위해 한 줄에 하나의 문장만 작성합니다.
- 들여쓰기는 4 Space 대신 Tab을 사용합시다.
- 주석은 설명하려는 구문에 맞춰 들여쓰기 + 위에 작성 합니다.
  (데이터 전처리팀) 전처리별 구분 주석은 ###으로 한 줄 위에 작성 합니다.
- 키워드 인수를 나타낼 때나 주석이 없는 함수 매개변수의 기본값을 나타낼 때 기호 주위에 공백을 사용하지 마세요.
- 연산자 사이에는 공백을 추가하여 가독성을 높입니다.
- 콤마(,) 다음에 값이 올 경우 공백을 추가하여 가독성을 높입니다.

02

Collaboration

협업 방식

실험 관리

WandB 사용해 관리

Name (15 visualized)	Tags	Notes	User	Crea	End Tir	Valid RMSE	Valid MAE	Valid MSE	param	features	Runtim	State	pa
add harmonic_average_rating & fix steam_rating	submit	2.2480	geunsseo	24h ago	Nov 07 '24	1.76869	1.23938	3.12831	{"learning_rate":0.189	["user_id","isbn","age_rang	38m 3s	Finished	-
add harmonic average_rating & apply optuna	submit	2.2783	geunsseo	2d ago	Nov 07 '24	1.93416	1.43239	3.74103	{"learning_rate":0.189	["user_id","isbn","age_rang	3h 34m 2s	Finished	-
add harmonic_average_rate	submit	2.4284	geunsseo	2d ago	Nov 06 '24	1.94126	1.43911	3.76853	{"learning_rate":0.5,"d	["user_id","isbn","age_rang	21m 55s	Finished	-
add average_rating & apply optuna	submit	2.2845	geunsseo	2d ago	Nov 07 '24	1.91975	1.41613	3.68548	{"learning_rate":0.165	["user_id","isbn","age_rang	3h 38m 23	Finished	-
add average_rating	submit	2.4304	geunsseo	2d ago	Nov 06 '24	1.92832	1.42418	3.71847	{"learning_rate":0.5,"d	["user_id","isbn","age_rang	13m 38s	Finished	-
optuna_base	submit	2.1250	davinkeem	2d ago	Nov 06 '24	2.12387	1.59472	4.51086	{"learning_rate":0.163	["user_id","isbn","age_rang	2h 42m 59	Finished	-
new_base		Add notes	davinkeem	2d ago	Nov 06 '24	2.13056	1.60116	4.53929	{"learning_rate":0.5,"d	["user_id","isbn","age_rang	17m 49s	Finished	-
clip		Add notes	kyk709	2d ago	Nov 06 '24	2.13056	1.60116	4.53929	{"learning_rate":0.5,"d	["user_id","isbn","age_rang	12m 8s	Finished	-
user,book review_counts	submit	2.1267	kyk709	2d ago	Nov 06 '24	2.13056	1.60116	4.53929	{"learning_rate":0.5,"d	["user_id","isbn","age_rang	12m 12s	Finished	-
book_review_counts only		Add notes	davinkeem	2d ago	Nov 06 '24	2.13272	1.60029	4.5485	{"learning_rate":0.5,"d	["user_id","isbn","age_rang	16m 23s	Finished	-
use active_user		Add notes	kyk709	2d ago	Nov 06 '24	2.13438	1.59586	4.55558	{"learning_rate":0.5,"d	["user_id","isbn","age_rang	9m 42s	Finished	-

03

데이터 엔지니어링

POINT.01 EDA 및 전처리

POINT.02 피처 엔지니어링

03

EDA 및 전처리

EDA 진행 방식 : 6명 각자 EDA 진행 후 회의를 통해 최종 EDA 결정

- (DAY6~7) 각자 EDA 후 베이스라인 코드 작성해보기
  → 11/03(일) 20:00 EDA 회의 ("aistages 제출, PR까지 미리 하세요.")

(DAY8) - EDA day ~ (feat.모듈화)

- Data팀 : EDA 하면서 발주서 작성
- Model팀 : baseline 기반으로 구현

EDA 및 전처리

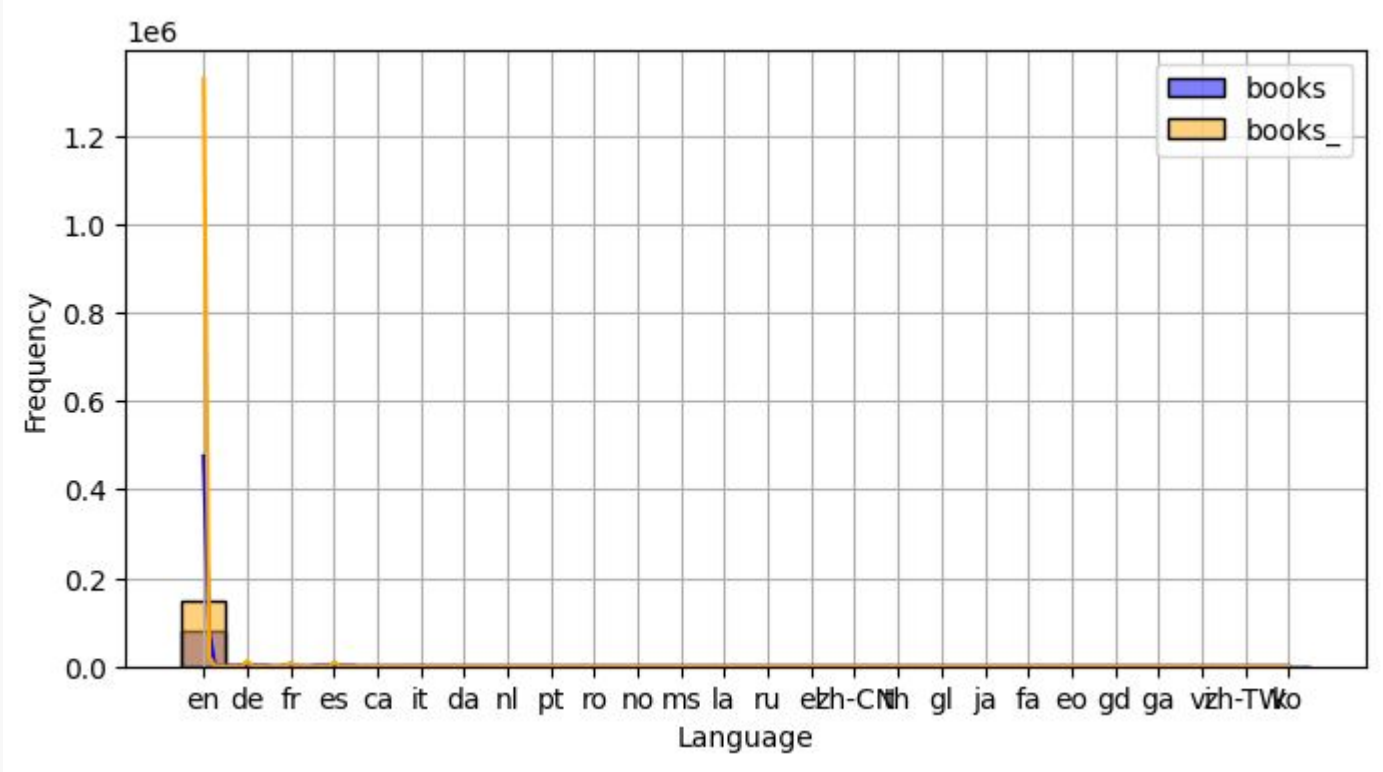


Figure 1. 언어별 빈도수

(파랑: 결측치 처리 전, 노랑: 결측치 최빈값으로 대체)

language 변수는 결측치 비율이 40% 이상 (44.946848)

최빈값으로 대체할 경우 데이터의 분포가 왜곡될 수 있어 ISBN 국가 코드를 활용해 결측치 대체

EDA 및 전처리

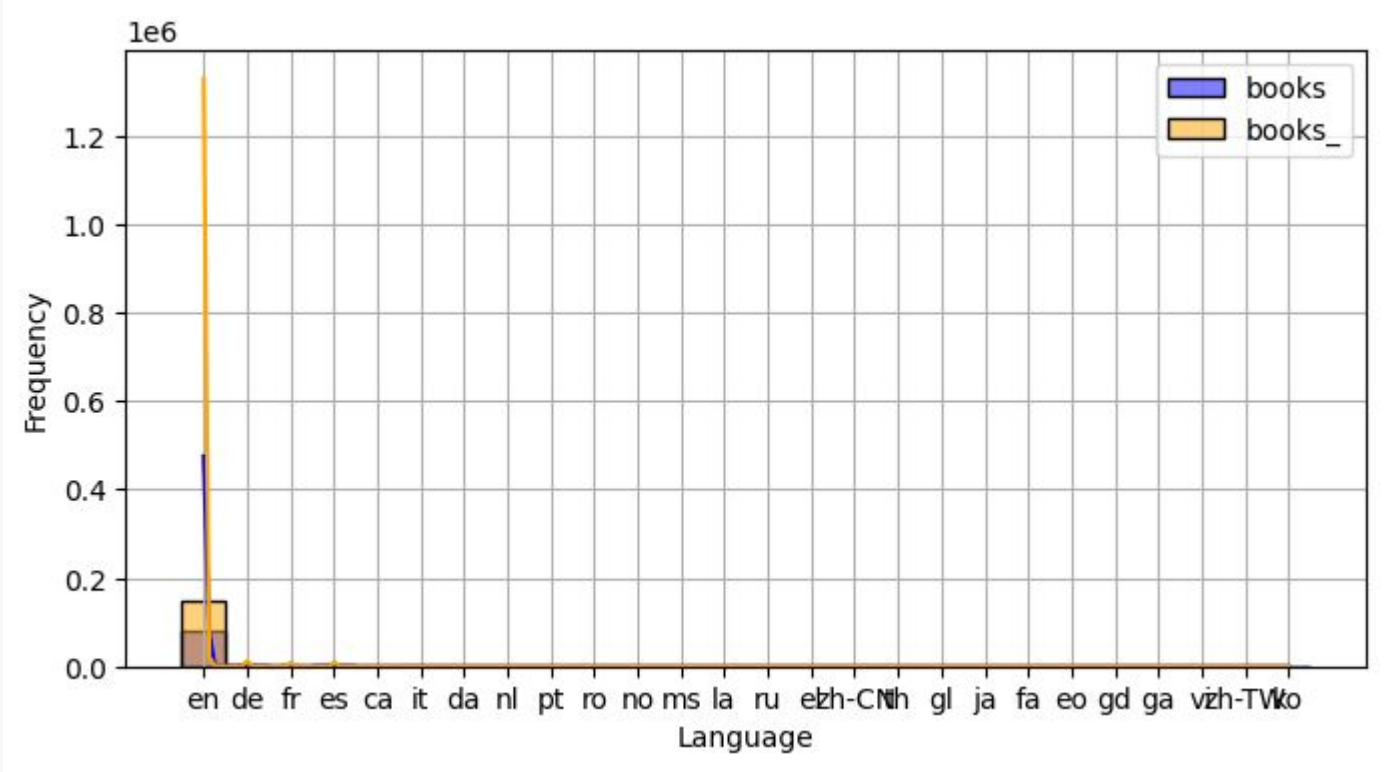


Figure 1. 언어별 빈도수

(파랑: 결측치 처리 전, 노랑: 결측치 최빈값으로 대체)

ISBN 국가 번호	언어 코드 (in books)	언어
0	en	영어권
3	de	독일어권
84	es	스페인어
2	fr	프랑스어권
88	it	이탈리아어
94	nl	네덜란드어
989	pt	포르투갈어
87	da	덴마크어
967 (말레이시아) 602 (인도네시아)	ms	말레이어
82	no	노르웨이어
7	zh-CN	중국어권
5	ru	러시아어
4	ja	일본어
606	ro	루마니아어
618	el	그리스어
974	th	태국어
600 (이란)	fa	페르시아어
604	vi	베트남어
89	ko	한국어

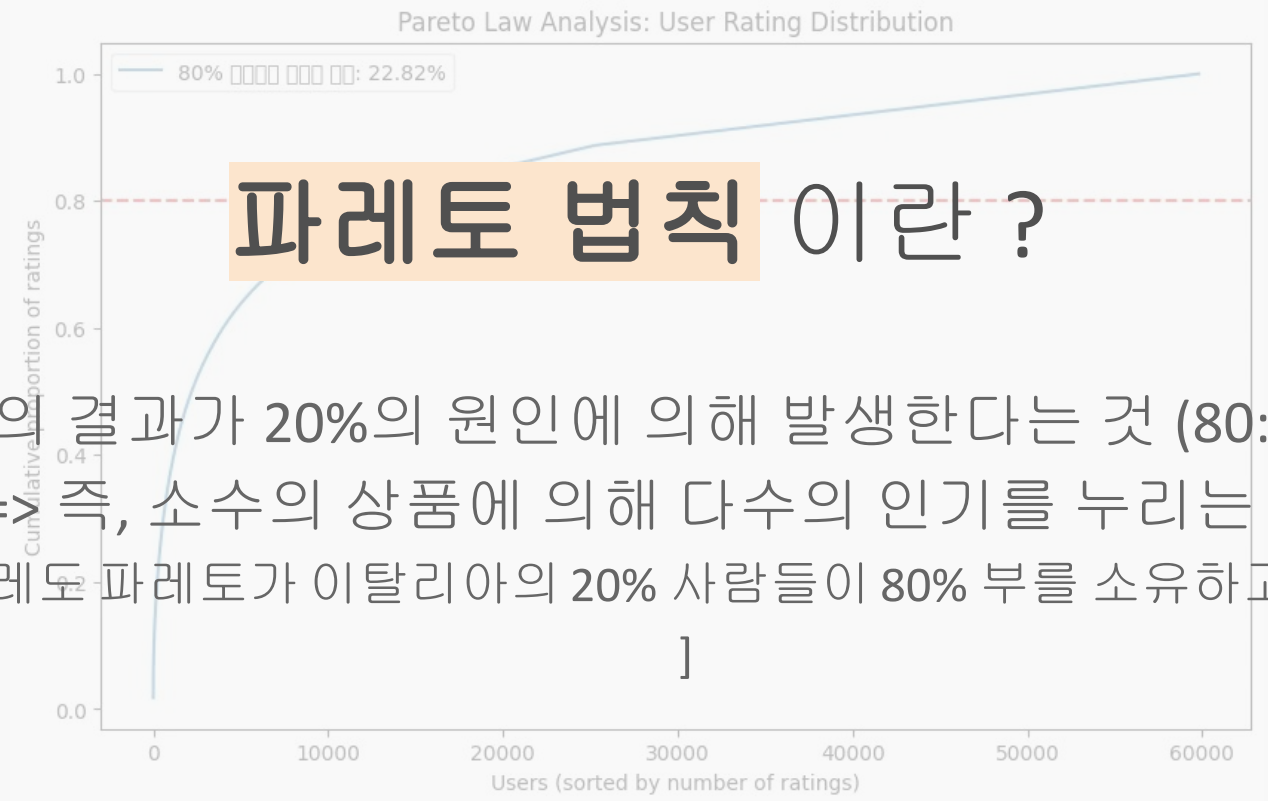
https://ko.wikipedia.org/wiki/%EA%B5%AD%EA%B0%80%EB%B3%84_ISBN

language 변수는 결측치 비율이 40% 이상 (44.946848)

최빈값으로 대체할 경우 데이터의 분포가 왜곡될 수 있어 ISBN 국가 코드를 활용해 결측치 대체

03

EDA 및 전처리



파레토 법칙 이란?

80%의 결과가 20%의 원인에 의해 발생한다는 것 (80:20 법칙)

=> 즉, 소수의 상품에 의해 다수의 인기를 누리는 구조

[경제학자 빌프레도 파레토가 이탈리아의 20% 사람들이 80% 부를 소유하고 있는 현상에서 유래]

Figure 2. 유저 수(유저별 리뷰 수를 기준으로 정렬)의 누적 비율

EDA 및 전처리

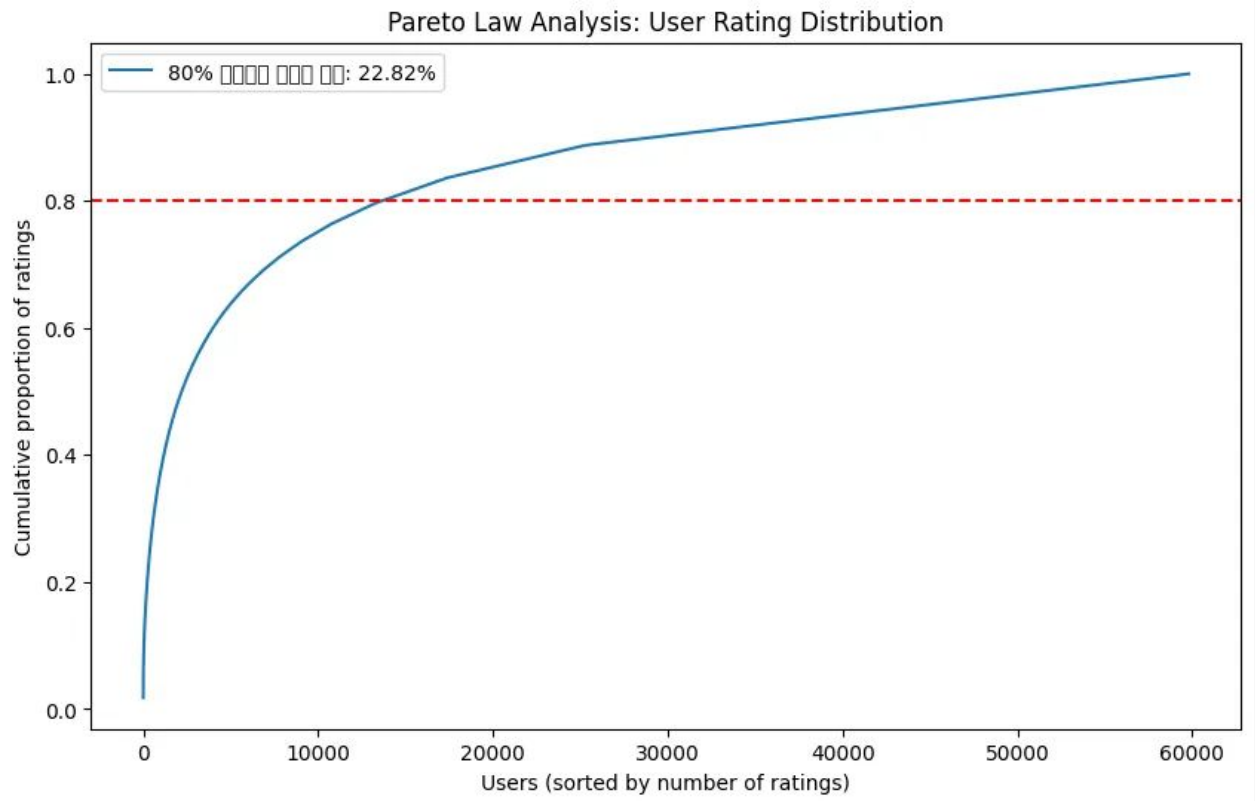


Figure 2. 유저 수(유저별 리뷰 수를 기준으로 정렬)의 누적 비율

유저 당 리뷰 수를 파레토 차트로 나타낸 결과, 22.82%의 유저가 전체 리뷰의 80%를 작성함

EDA 및 전처리

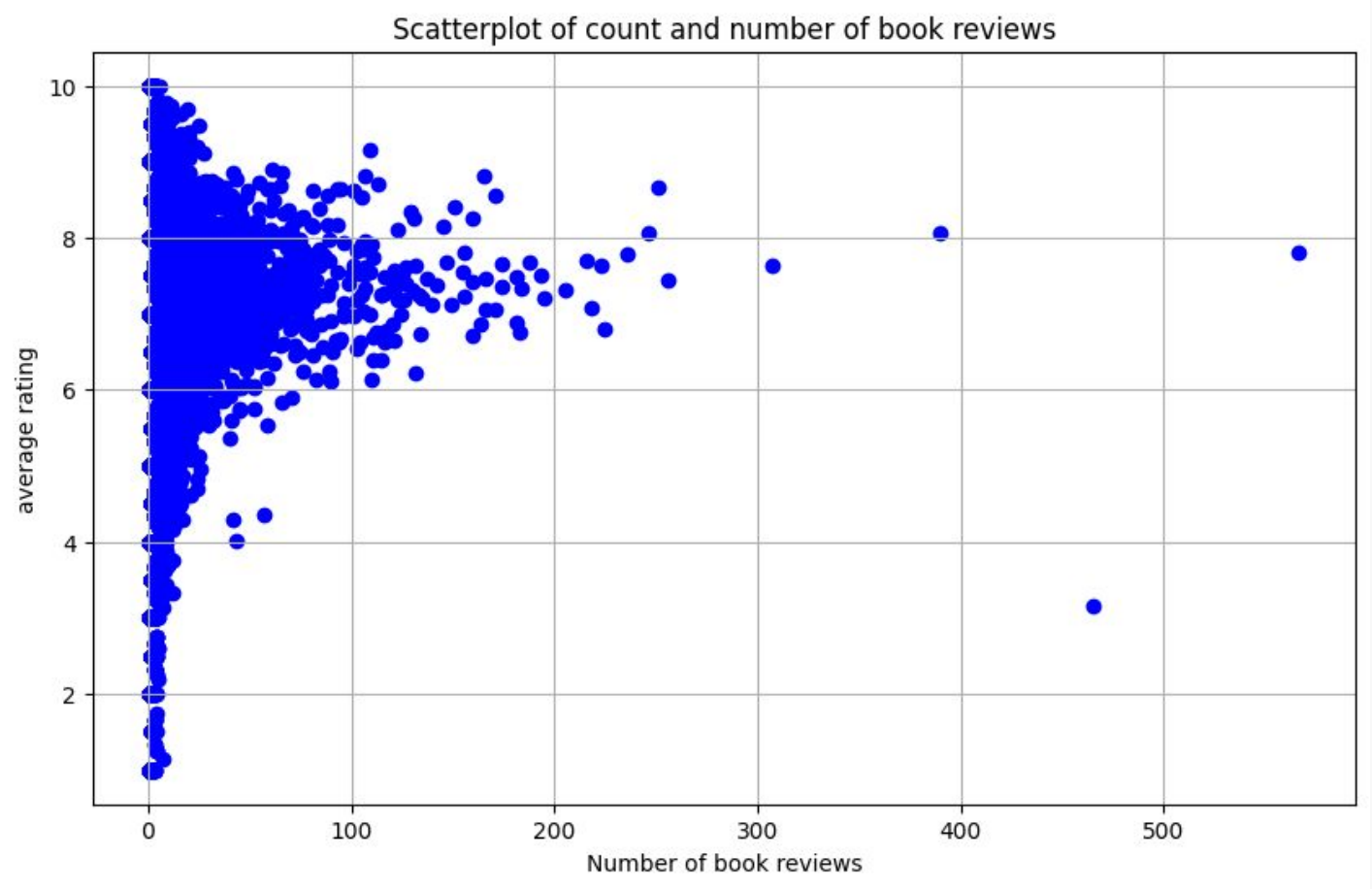


Figure 3. 책에 매겨진 리뷰 수와 평점의 산점도

– 리뷰 수가 늘어날수록 높은 평점의 빈도 증가한다는 가설 설정

03

EDA 및 전처리

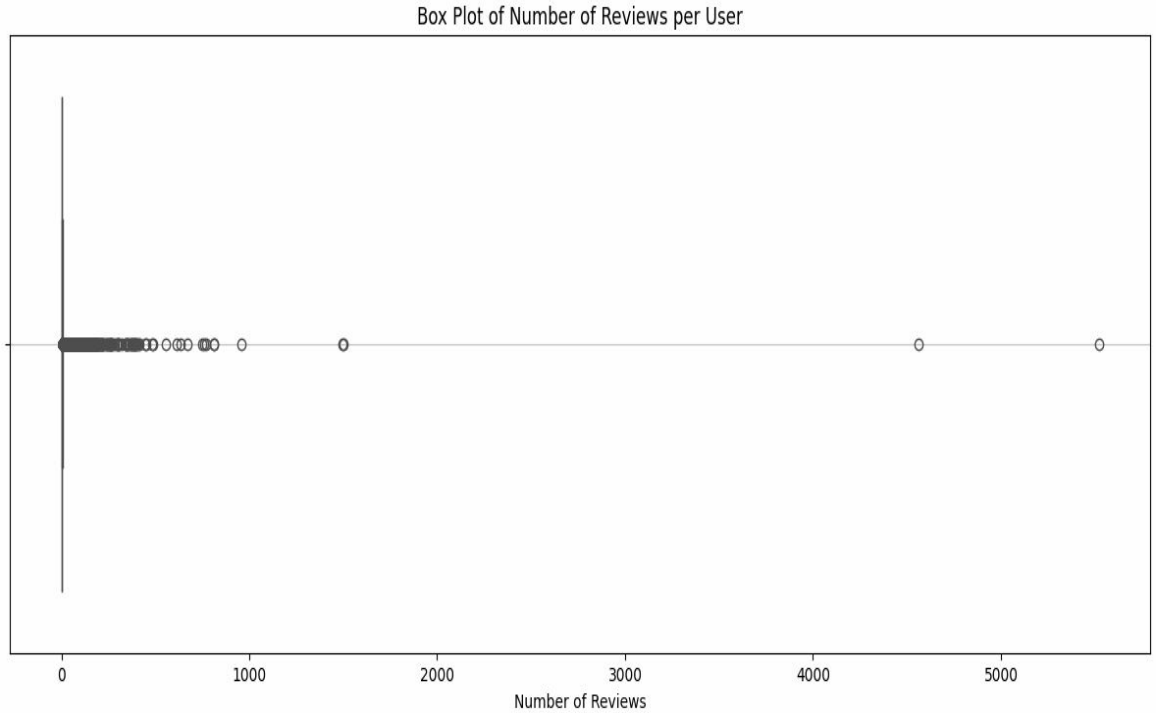


Figure 4. 유저별 리뷰 수의 상자 그림
- 극단적으로 적은 값의 비율이 매우 높다.

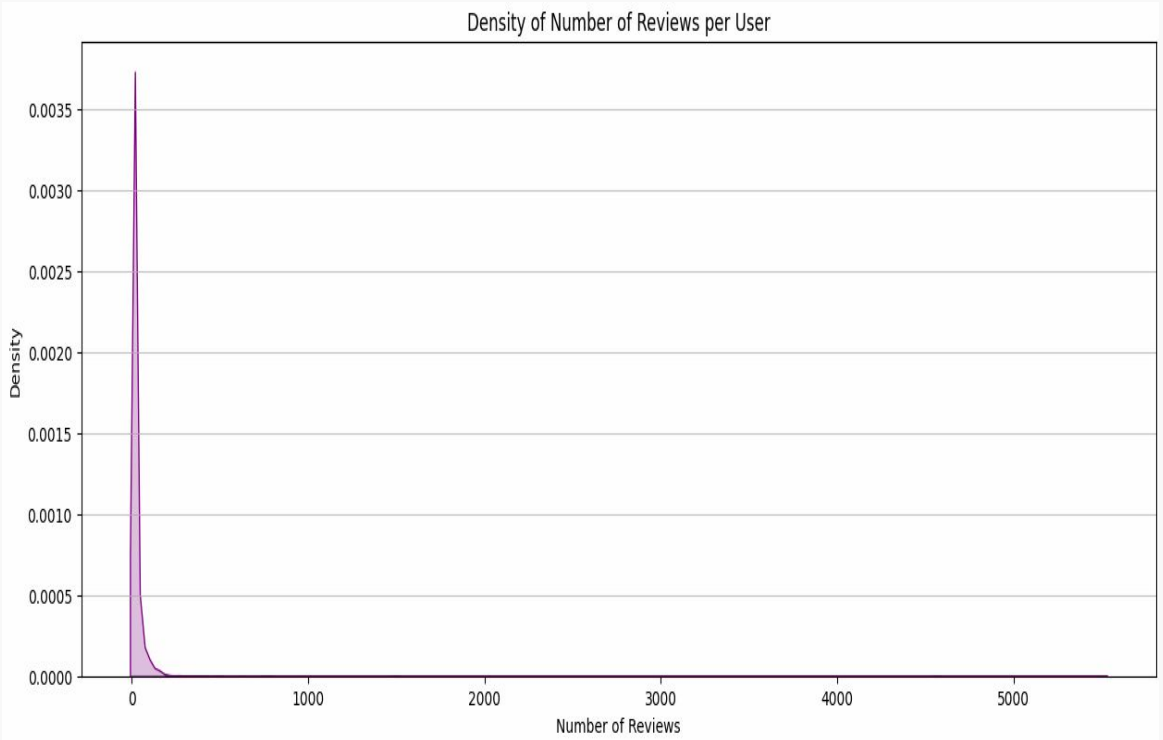


Figure 5. 유저별 리뷰 수의 분포 확인
- 극단적으로 적은 값의 비율이 높은 점을 감안해 KDE Plot 활용

03

EDA 및 전처리

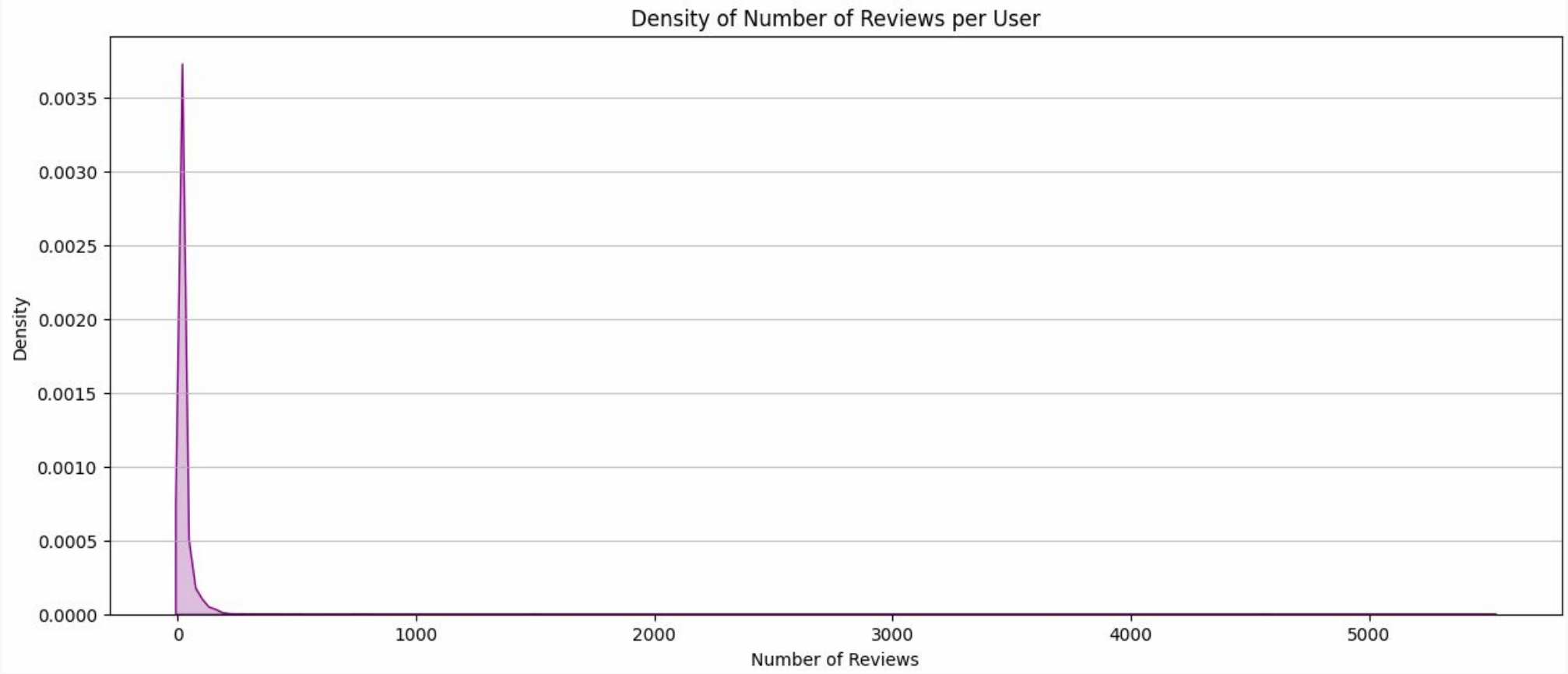


Figure 5. 유저별 리뷰 수의 분포 확인 - 극단적으로 적은 값의 비율이 높은 점을 감안해 KDE Plot 활용

03

EDA 및 전처리

스코어별 특징

- **산술평균**

  주로 데이터가 고르게 분포했을 때 유용, 하지만 평점에 극단적인 값이 많을수록 영향을 크게 받음

- **조화평균**

  작은 값에 더 민감하게 반응, 낮은 평점에 더 많은 가중치 부여

  평점 값이 매우 높은 값이 일부 있으면 조화평균은 산술평균보다 덜 반영하는 경향이 있음

03

EDA 및 전처리

스코어별 특징

- **베이지안 평균**

  평점 수가 적을수록 사전 확률로 정한 값(보통 전체 평점)을 가중치(보정값)를 더해 사용
  평점 수가 많아질수록 점차 실제 유저 평균으로 수렴

$$\text{Bayesian Average} = \frac{\sum x_i + m \cdot C}{n + m} \quad \text{where} \begin{cases} C = \text{total average}, \\ n = \#\ \text{of reviews}, \\ m = \text{constant} \end{cases}$$

03

EDA 및 전처리

스코어별 특징

- **Steam Rating Formula**

  평점 수가 적을수록 극단적인 값이 아닌 기준값으로 수렴하게 해
  평가가 적은 리뷰가 전체 평점에 미치는 영향을 줄임

$$\text{average rating} = \frac{\text{sum(rating)}}{\#\ \text{of reviews}}$$

$$\text{score} = \text{average rating} - (\text{average rating} - 5.5) \cdot 2^{-\log_{10}(\#\ \text{of reviews} + 1)}$$

03

EDA 및 전처리

구간별 다른 평점 적용

- 평점 수를 구간마다 다른 평균 스코어를 적용하여 매핑
  1. 평점 수가 20개 이상 → 산술평균 또는 조화평균 적용
  2. 평점 수가 10~20개 → 베이지안 평균 적용
  3. 평점 수가 1~9개 → Steam Rating Formula 적용

콜드 스타트

- 콜드 스타트 문제를 해결하기 위해 유저 데모그래픽 정보를 활용
- 공통 데모그래픽 정보를 갖는 평점을 매긴 유저의 리뷰 수에 따라 적절한 평균을 사용
  1. 평점 수가 10개 이상 → 산술평균 또는 조화평균 적용
  2. 평점 수가 10개 미만 → 베이지안 평균 또는 Steam Rating Formula 적용

03

EDA 및 전처리

Book 데이터 전처리 및 변수 생성

- book_title, book_author, publisher : 정규표현식 활용한 text 전처리
- year_of_publication → publication_range로 범주화

  1970년 이전과 2000년 이후는 하나로, 나머지는 5년 단위
- language : isbn을 활용한 결측치 처리 (없는 경우 최빈값으로)
- category → category_high로 범주화

  정규표현식 활용한 text 전처리, 첫번째 category 활용, 상위 카테고리로 범주화

03

EDA 및 전처리

User 데이터 전처리 및 변수 생성

- location → location_country, location_state, location_city로 str 분리

  결측치는 다른 변수 값으로 대체 후 나머지는 최빈값으로 대체

- age → age_range로 범주화

  평균으로 결측치 처리 후

  20대 미만과 60대 이상은 하나로, 나머지는 10대 단위로 범주화

03

Feature Engineering

피처 엔지니어링

User & Book 데이터를 활용한 변수 생성

- 리뷰 횟수(user_review_counts, book_review_counts)

  사용자의 리뷰 횟수로 활동성을 측정하는 피처와

  도서의 리뷰 횟수 책의 베스트셀러 정도를 나타내는 피처를 추가

- 유저별 평균 평점(average_rating)

  Steam Rating Formula와 베이지안 평균을 활용해 평점을 예측하는 피처를 추가

03

Feature Engineering

피처 엔지니어링

변수 선택

범주형 변수

- Book : isbn, book_title, book_author, publisher, language, high_category, publication_range
- User : user_id, age_range, location_country

수치형 변수

user_review_counts, book_review_counts

04

모델링 과정

POINT.01 모델 비교 및 분석

POINT.02 모델 튜닝

POINT.03 앙상블

POINT.04 최종 모델 선정

POINT.05 프로젝트 결과

04

Modeling

모델 비교 및 분석

CatBoost

범주형 변수를 효과적으로 처리할 수 있어
데이터셋의 특성에 맞는 빠르고 높은 성능 기대

XGBoost

강력한 부스팅 알고리즘을 사용하여
과적합을 방지하고 예측 정확도를 높이는 데 유리

Text_DeepFM

텍스트 데이터를 기반으로
추천 성능을 향상시키기 위해 선택

Image_DeepFM

이미지 데이터를 기반으로
추천 성능을 향상시키기 위해 선택

04

Modeling

모델 튜닝

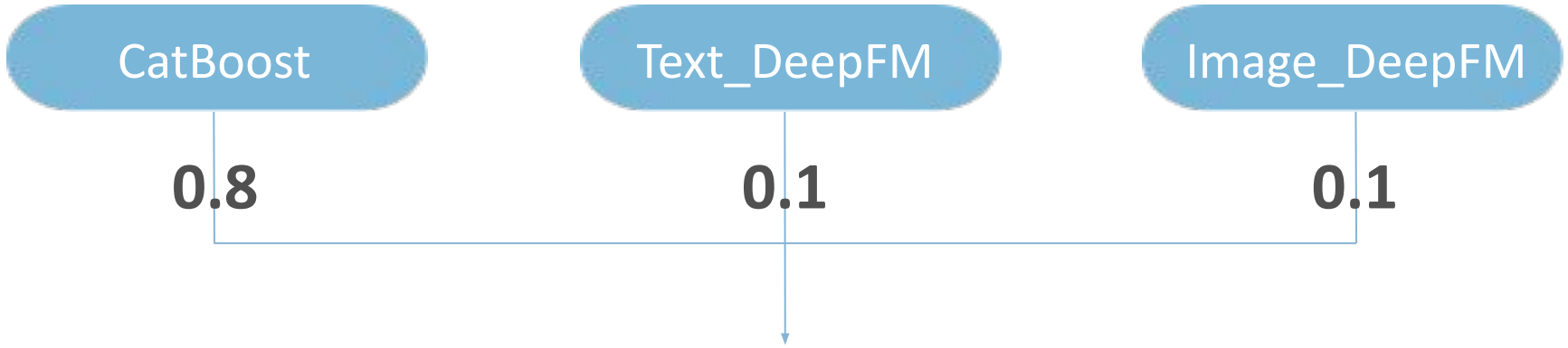
Stratified K-Fold 교차 검증 적용

Optuna를 활용한 하이퍼파라미터 튜닝

Weighted Ensemble 진행

Modeling

앙상블



각 모델로부터 얻은 서로 다른 예측값을 입력하여 앙상블의 성능을 극대화하는 전략 적용
더 높은 성능을 보인 모델에 더 높은 가중치를 부여하여 예측 정확도 최적화

04

최종 모델 선정

CatBoost

카테고리형 데이터와
구조화된 데이터를 처리하는 데
탁월한 성능을 보임

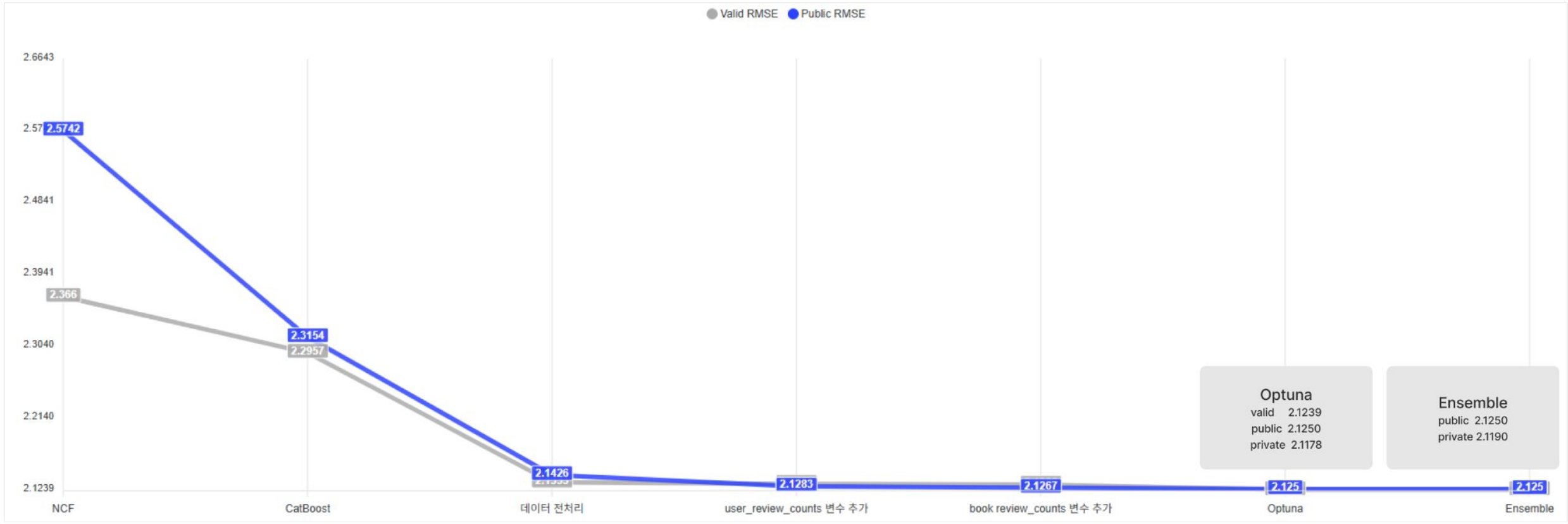
Weighted Ensemble

여러 모델의
강점을 결합하여
더 나은 예측 성능을 보임

04

Final Model

프로젝트 결과



Final Model

프로젝트 결과

☑	Catboost_stf...ptuna		2.1250 2.1178	2024.11.06 20:55	완료

제출 결과 1. Stratified K-Fold를 이용해 교차 검증을 적용한 CatBoost 단일 모델 학습 결과

☑	Ensemble_Wei...d_0.8		2.1250 2.1190	2024.11.07 00:39	완료

제출 결과 2. Catboost와 Image FM, Text FM에 각각 8 : 1 : 1의 비율로 소프트 보팅을 적용한 결과

1	RecSys_03조 🏆		2.1178	45	22h

최종 리더보드 순위 🎉1등!🎉

05

프로젝트 회고

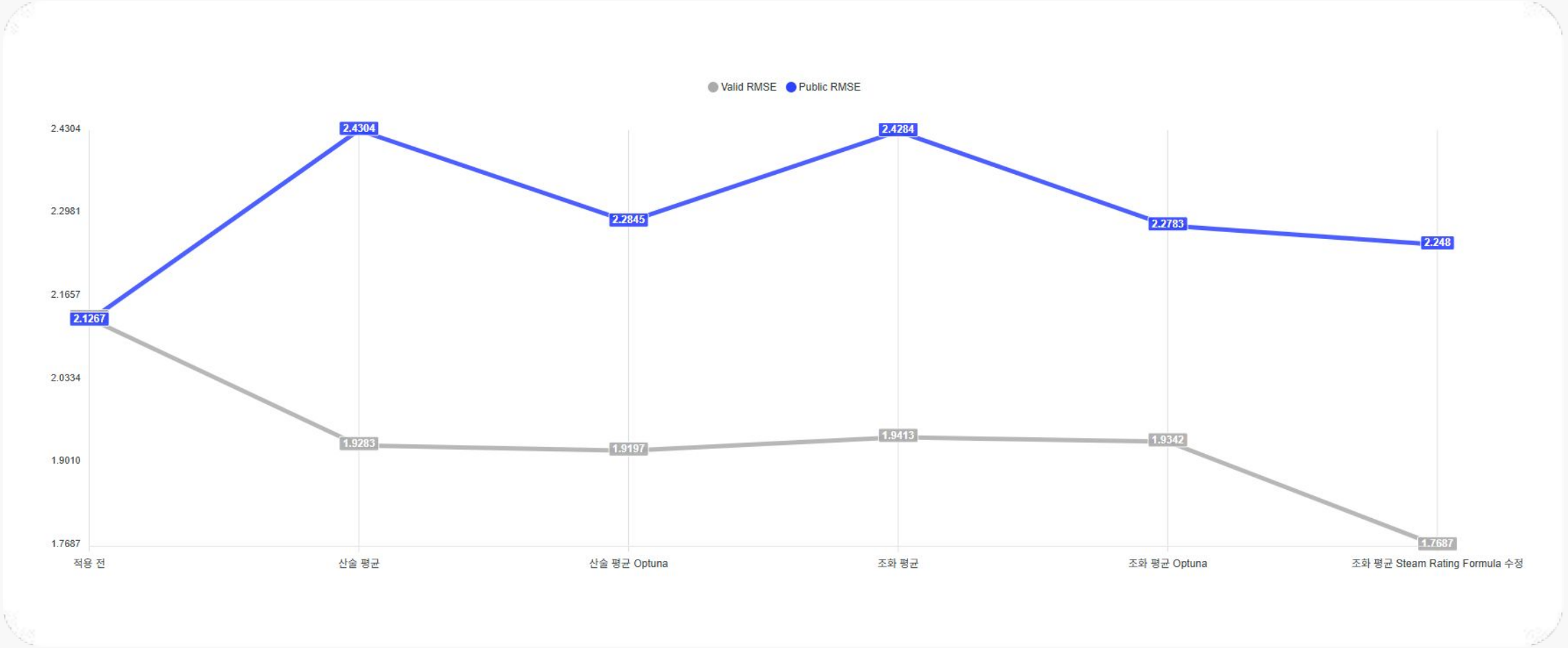
POINT.01 시행 착오

POINT.02 이번 프로젝트를 통해 배운 점

POINT.03 다음 프로젝트에 적용해볼 점

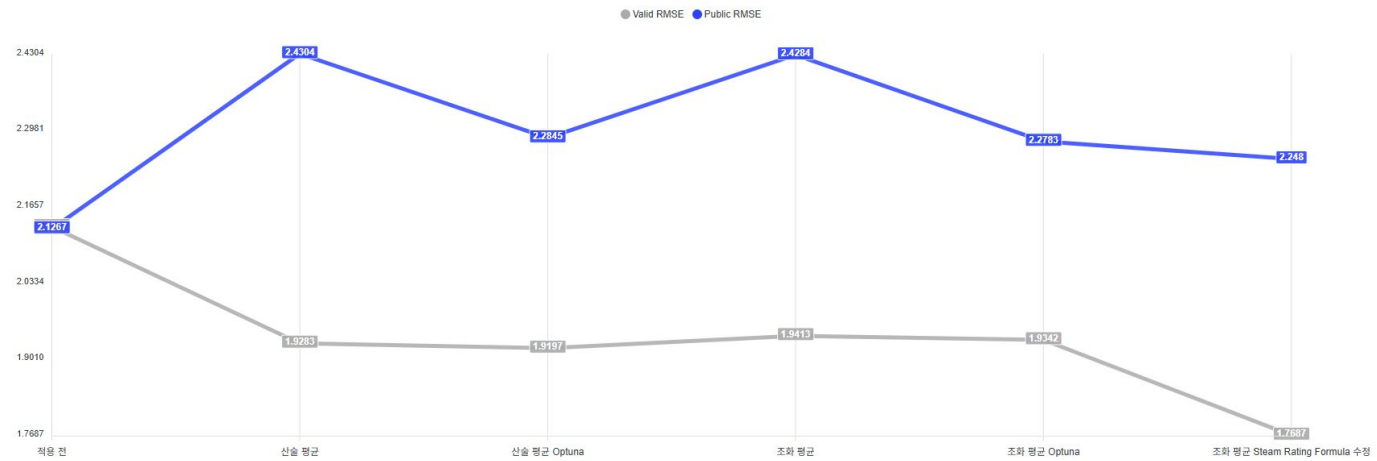
05

시행 착오


Valid RMSE
Public RMSE
2.4304
2.2981
2.1657
2.0334
1.9010
1.7687
2.1267
2.4304
2.2845
2.4284
2.2783
2.248
1.9283
1.9197
1.9413
1.9342
1.7687
적용 전
산술 평균
산술 평균 Optuna
조화 평균
조화 평균 Optuna
조화 평균 Steam Rating Formula 수정

05

시행 착오



	Valid RMSE	Public RMSE	Model
적용 전	2.1305	2.1267	CatBoost
조화 평균 사용	1.9413 ▼	2.4284 ▲	CatBoost
조화 평균 사용 + Optuna 적용	1.9342 ▼	2.2783 ▼	CatBoost
조화 평균 사용 + Steam Rating Formula 수정	1.7687 ▼	2.2480 ▼	CatBoost

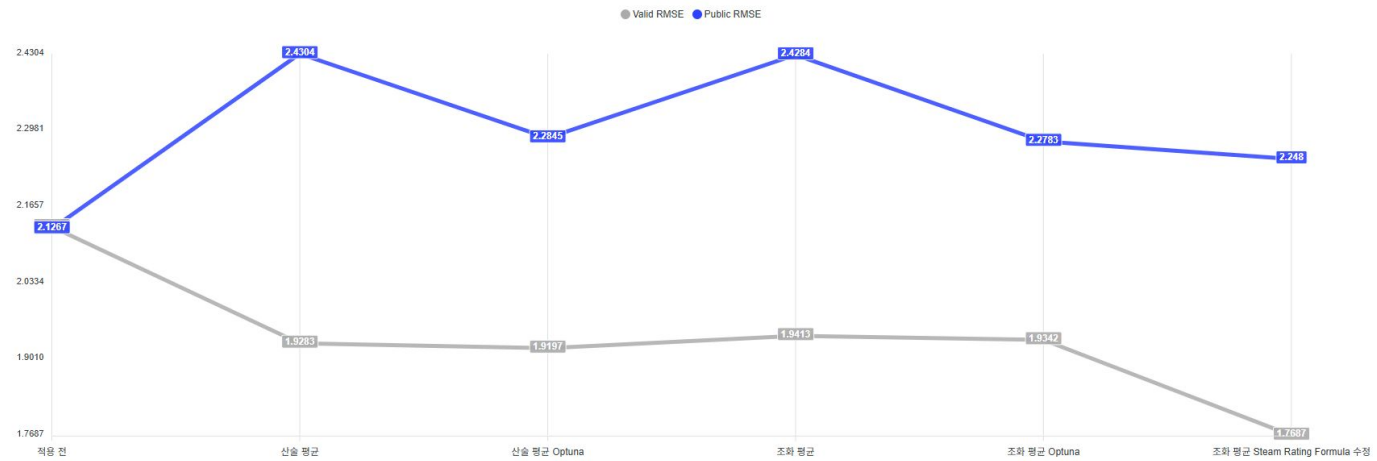
유저별 평균 평점(average_rating)

유저별 평균 평점은 다음과 같이 생성

- 유저별 평점 분포가 다름
- 유저별 평점 수가 극단적으로 적은 경우가 대부분(평점 수 상위 10%가 4개 이상)
- 다양한 평균 계산
  1. 산술평균
  2. 조화평균
  3. 베이지안 평균
  4. Steam Rating Formula
- 평점 수의 구간마다 각 평균들의 특징을 살려 각각 매핑

05

시행 착오



	Valid RMSE	Public RMSE	Model
적용 전	2.1305	2.1267	CatBoost
조화 평균 사용	1.9413 ▼	2.4284 ▲	CatBoost
조화 평균 사용 + Optuna 적용	1.9342 ▼	2.2783 ▼	CatBoost
조화 평균 사용 + Steam Rating Formula 수정	1.7687 ▼	2.2480 ▼	CatBoost

유저별 평균 평점(average_rating)

콜드 스타트 문제는 다음과 같이 적용

(test data에서 처음으로 등장하는 유저의 평점)

1. test data에서 처음 등장하는 데모그래픽 정보와 같은 정보를 갖는 유저를 train data에서 찾는다.
2. train data에서 미리 계산한 다양한 평균 평점을 활용할 수 있게 평균 평점의 평균으로 사용 (표본평균의 평균처럼)
3. 같은 데모그래픽 정보를 갖는 유저의 평점 수에 따라 다른 평점을 매핑
   - **평점 수 10 미만**: 베이지안 평균 또는 Steam Rating Formula
   - **평점 수 10 이상**: 산술평균 또는 조화평균

05

시행 착오

PCA_user_summary_dim0	PCA_user_summary_dim1	PCA_user_summary_dim2	PCA_user_summary_dim3	PCA_user_summary_dim4
-0.237793	3.276237	-3.307261	-0.641598	-0.321731
1.145325	-4.468685	0.425656	0.095989	-0.463568
2.392794	-3.693433	1.401775	-0.001178	-0.702486
0.479492	4.429990	-1.038597	0.740431	0.207224
2.660652	-3.330617	1.306553	-0.643068	-0.407529
...	...	...	...	...

PCA_book_summary_dim0	PCA_book_summary_dim1	PCA_book_summary_dim2	PCA_book_summary_dim3	PCA_book_summary_dim4
-2.464710	8.156643	0.819486	1.499137	0.454352
-2.464710	8.156643	0.819486	1.499137	0.454352
-2.841926	-2.430526	-1.083330	-0.183209	-0.279758
-4.188267	-1.405234	-0.586769	0.805591	-1.371812
-1.132310	5.041648	-0.570259	-0.440621	0.413250
...	...	...	...	...

summary 임베딩 벡터 추가

기존 CatBoost 학습에서는 summary의 임베딩 벡터를 변수로서 사용하지 않음

이에 baseline에서의 텍스트 임베딩을 거친 벡터를 변수로 추가

기존 벡터는 768차원 벡터 두 개로 이루어져 있어 PCA를 적용해 5차원 벡터 두 개로 변환

05

시행 착오

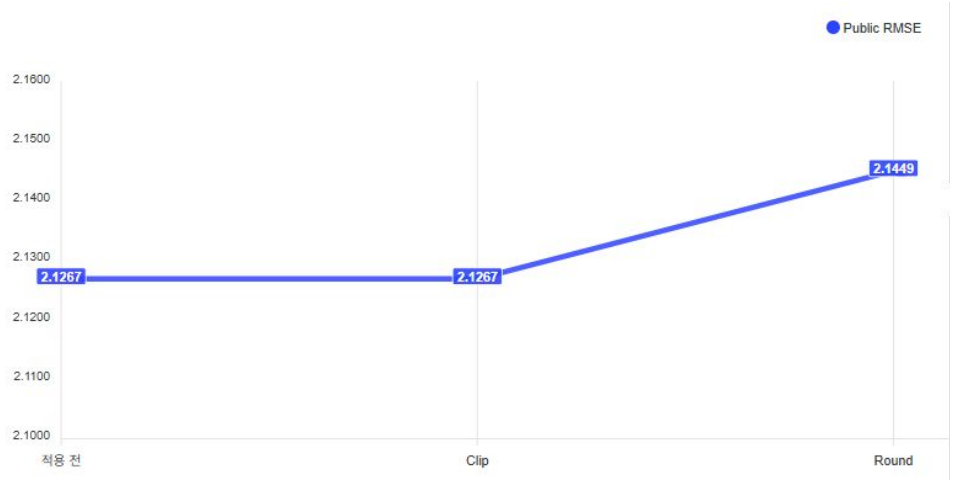
	Valid RMSE	Public RMSE	Model
적용 전	2.1306	2.1267	CatBoost
PCA를 활용한 임베딩 벡터 변수 추가	2.1288 ▼	2.4313 ▲	CatBoost

Public RMSE에서 성능이 오히려 낮아짐

➡ 트리 모델에서 임베딩 벡터의 특성을 잘 잡아내지 못한다!

05

시행 착오



	Public RMSE	Model
적용 전	2.1267	CatBoost
Clip(1,10) 사용	2.1267 (-)	CatBoost
Round 사용	2.1449 ▲	CatBoost

Round + Clip

- Clip() 사용하여 1 미만 값 1, 10 초과 10으로 매핑
- ex) 0.899 - > 1, 10.332 -> 10

- Round() 사용하여 소수점 첫째 자리에서 반올림
- ex) 7.333 -> 7, 8.66 -> 8

- 반올림의 경계에 있는 평점(.5로 끝나는)은 loss를 크게 증가
- 경계에 있지 않은 평점(x.5 초과 (x+1).5 미만)은 loss를 감소

05

이번 프로젝트를 통해 배운 점

3 GUNA!

모듈화

모듈화를 잘 하면 이렇게 좋GUNA!

WandB, 노션 데이터베이스

WandB와 노션 데이터베이스를
활용했더니 이렇게 좋GUNA!

도메인 지식

도메인 지식은 알면 알수록 중요하GUNA!

05

다음 프로젝트에 적용해볼 점

3 HAJA!

다양한 모델 실험 병행

가장 성능이 뛰어난 모델을 기준으로만 실험을 진행하는 것이 아닌 다양한 모델과 함께 실험을 진행하면서 기록HAJA!

프로젝트 구조

이번 프로젝트 구조를 참고하여 다음 프로젝트 진행HAJA!

모델 구조

모델을 단순히 점수 기준으로만 사용하지 말고, 모델 구조에 대한 이해를 한 후, 모델을 선정HAJA!

감사합니다