



RecSys 01 - 공통알





김은혜

 kimeunh3



장원준

 jwj51720




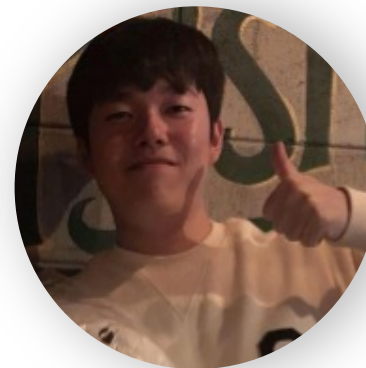
이수경

 4low1ives




정준환

 Jeong-Junhwan

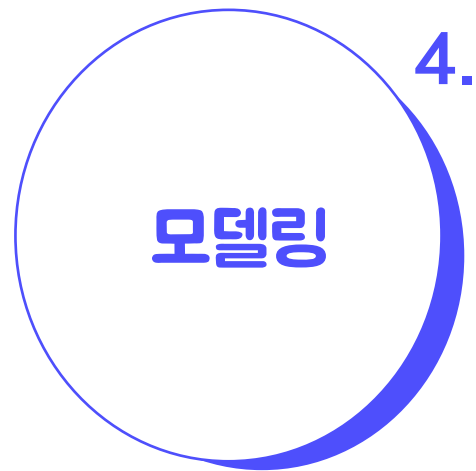


류명현

 ryubright

목차

Table of Contents



01. 목표/전략

목표

협업방식

타임라인

Goal

팀원 모두의
고른 성장

- 최종 프로젝트를 위해 팀원 모두가 프로젝트 전반에 대한 깊은 이해 필요
- 본인이 자신 있는 부분이 아니라 자신 없는 부분을 맡으며 부족한 능력 배양하기

1

베이스라인 코드
직접 작성

제공된 베이스라인,
파이토치 템플릿을 활용해
팀만의 베이스라인 구축

2

PM 로테이션제

이틀에 한 번씩 PM을
돌아가면서 말음으로써
프로젝트 전반에 대한 이해 도모

3

팀 로테이션제

팀을 나누어 운영함에 있어
이틀에 한 번씩 팀을 바꾸며
프로젝트 진행 속도 공유

4

강의와 실습 내용
온전히 파악하기

제공된 강의와 실습 내용의
EDA, 전처리 방식을 최대한
이해하며 활용하기

TEAM2
팀원 모두의
고른 성장

PM

TEAM1

TEAM2

TEAM1

베이스라인 코드 직접 작성

제공된 베이스라인,
파이토치 템플릿을 활용해
팀만의 베이스라인 구축

PM 로테이션제

이틀에 한 번씩 PM을
돌아가면서 말음으로써
프로젝트 전반에 대한 이해 도모

팀 로테이션제

팀을 나누어 운영함에 있어
이들에 한 번씩 팀을 바꾸며
프로젝트 진행 속도 공유

강의와 실습 내용
온전히 파악하기

제공된 강의와 실습 내용의
EDA, 전처리 방식을 최대한
이해하며 활용하기

01. 목표/전략

목표

협업방식

타임라인



Project와 Issue 활용

Milestones

브랜치 전략

Dkt 프로젝트 안에서 할 것, 진행 중인 것, 완료된 것, pr 등을 개별 이슈로 정리하여 프로젝트 진행 상황을 한 눈에 볼 수 있습니다.

@level2-RecSys-01-DKT-project

View 1 View 2 + New view

Filter by keyword or by field

Todo 0

In Progress 0

Done 37

- level2_dkt_recsys-level2-recsys-01 #2 [데이터] 전처리
- level2_dkt_recsys-level2-recsys-01 #1 [데이터] EDA
- level2_dkt_recsys-level2-recsys-01 #3 [베이스라인] 설계도 제작
- level2_dkt_recsys-level2-recsys-01 #36 [베이스라인] wandb sweep 활용하기
- level2_dkt_recsys-level2-recsys-01 #4 [베이스라인] 코드 구조 제작
- level2_dkt_recsys-level2-recsys-01 #5 [전처리] answer code 누적합 column 제작
- level2_dkt_recsys-level2-recsys-01 #6 [데이터] 아이템 (시험지) 데이터 생성
- level2_dkt_recsys-level2-recsys-01 #7 [전처리] 그래프 데이터 만들기
- level2_dkt_recsys-level2-recsys-01 #9

PR 30

- level2_dkt_recsys-level2-recsys-01 #8 add eda_ryu.ipynb
- level2_dkt_recsys-level2-recsys-01 #16 Add eda_sssu.ipynb
- level2_dkt_recsys-level2-recsys-01 #17 [베이스라인] 설계도 제작 완료
- level2_dkt_recsys-level2-recsys-01 #21 Update base code structure
- level2_dkt_recsys-level2-recsys-01 #27 update eda
- level2_dkt_recsys-level2-recsys-01 #29 Update Baseline
- level2_dkt_recsys-level2-recsys-01 #31 Update 11.22 preprocessing.ipynb
- level2_dkt_recsys-level2-recsys-01 #32 Update preprocessing.ipynb
- level2_dkt_recsys-level2-recsys-01 #33

01. 목표/전략

목표

협업방식

타임라인



Project와 Issue 활용

Milestones

브랜치 전략

Milestone이란, 여러개의 issue를 카테고리화 해서 나누는 단계

EDA, 전처리, 베이스라인, 모델링으로 나뉘어져 해당하는 milestone을 만들고 issue를 해당 milestone에 등록해 프로젝트의 완성도를 눈으로 볼 수 있도록 했습니다.

Labels

Milestones

New milestone

0 Open 4 Closed

Sort

전처리

Closed 6 days ago Last updated less than a minute ago

100% complete 0 open 12 closed

Edit Reopen Delete

베이스라인

Closed 3 minutes ago Last updated less than a minute ago

100% complete 0 open 16 closed

Edit Reopen Delete

모델링

Closed 2 days ago Last updated 1 day ago

100% complete 0 open 7 closed

Edit Reopen Delete

EDA

Closed 6 days ago Last updated 6 days ago

100% complete 0 open 5 closed

Edit Reopen Delete

01. 목표/전략

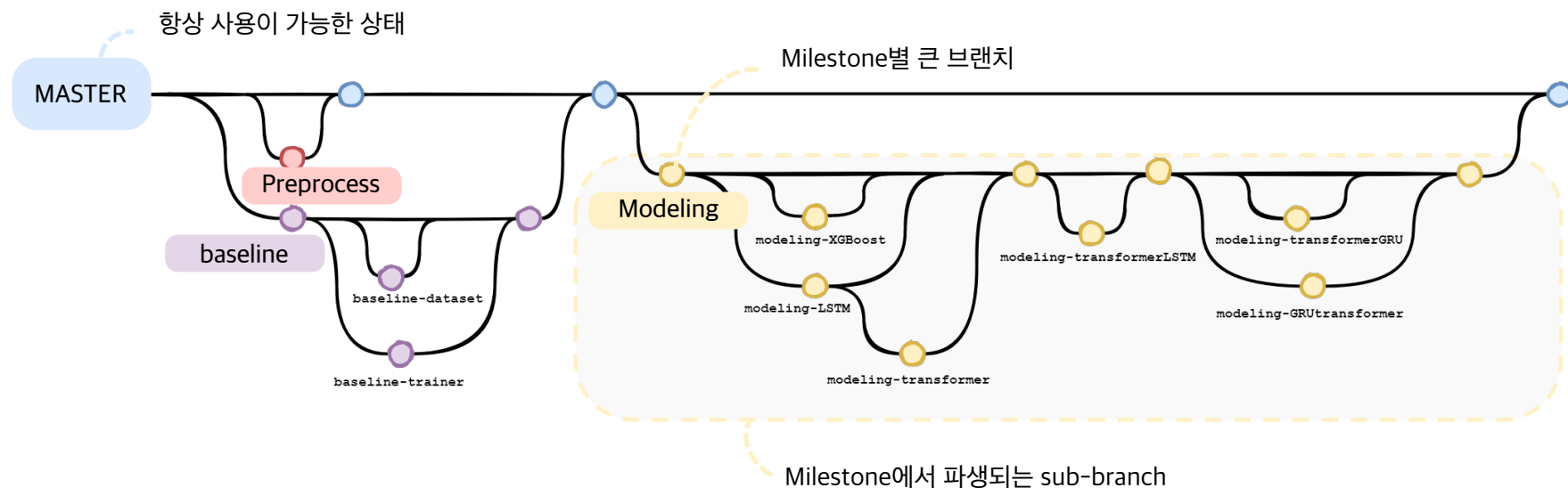
목표 협업방식 타임라인



Project와 Issue 활용

Milestones

브랜치 전략



01. 목표/전략

목표

협업방식

타임라인



데이터 버전 관리
























모델 결과 공유

w.  W&B



DKT dataset

Table +

 파일과 미디어	Aa 이름	 태그	 주인장	 설명
data_v1_1 train_v1.1.csv traintest_v1.1.csv	 data_v1.1	Data1.1	 장원준	[11/23] train_data.csv에서 feature 가공 및 추가
traintest_v1.2.zip	 data_v1.2	Data1.2	 류명현	
dataset_v1.3.zip	 data_v1.3	Data1.3	 류명현	
traintest_v2.1.csv	 data_v2.1	Data2.1	 수교	[11/25] elapsed_time 변경 반영
traintest_v2.2.csv	 data_v2.2	Data2.2	 류명현	[11/25] elapsed_time 변경에 따른 feature 수정
traintest_v3.0.csv	 data_v3.0	Data3.0	 준환 정	[12/06] elapsed_time 문제 해결
traintest_v3.1.csv	 data_v3.1	Data3.1	 준환 정	[12/07] 정답률 추가, 시간 추가
traintest_v3.2.csv	 data_v3.2	Data3.2	 준환 정	[12/08] 문제의 난이도, 유저의 실력 추가
traintest_v3.3.csv	 data_v3.3	Data3.3	 류명현	[12/08] elo feature 추가
traintest_v4.0.csv	 data_v4.0	Data4.0	 수교	[12/08] column명 정리

+ New

01. 목표/전략

목표

협업방식

타임라인



데이터 버전 관리

모델 결과 공유

w.  W&B

Aa wandb이름	🕒 모델	🕒 data_ver	☰ cat_col	☰ num_col	# val_auroc	# val_loss	# epoch	☰ 비교
 XGBoost	XGBoost	3.1	assessmentItemID test_id week_num KnowledgeTag question_number question_numslen test_month test_day test_cat testid	ALL	0.8427	0.775	69	
2022-12-07_20:15_sssu> ≤	GRUTransformer	3.1	assessmentItemID test_id question_number KnowledgeTag test_day test_month	elapsed_time ans_cumsum ans_cumavg time_question_median tag_acc elapsed_time_mean elapsed_time_median KnowledgeTag_et_mean KnowledgeTag_et_std test_acc test_hour hour_acc exp_tag KnowledgeTag_aC_mean	0.8499	0.3957	50	
 2022-12-07_20:09_ryubri ght	GTN	3.0			0.8434	0.3799	50	fold 10
 2022-12-07_22:04_eunhy e	GRUTransformer	3.1	assessmentItemID test_id question_number KnowledgeTag	elapsed_time ans_cumavg ans_cumsum time_question_median	0.8541	0.3135	50	

Timeline

11월

12월

16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 1 2 3 4 5 6 7 8

EDA

Preprocessing

Baseline

Debugging

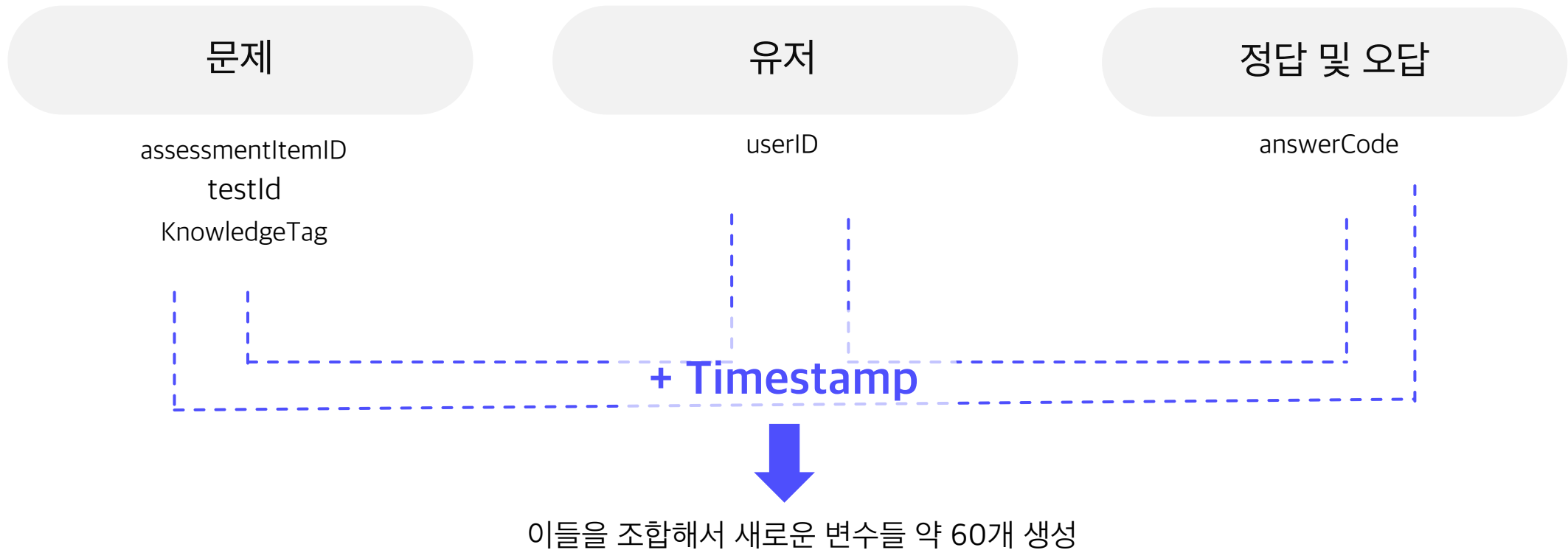
Modeling + Improvement

Ensemble

02. 데이터/전처리

Overview 파생변수

EDA를 통해 얻은 인사이트 보다는 직관적으로 접근 하여 피처를 생성하고자 함



'question_number', 'test_cat', 'test_day', 'test_hour', 'test_month', 'week_day', 'Timestamp_day', 'Timestamp_week', 'Timestamp_hour', 'question_numslen', 'post_Timestamp',
'elapsed_time', 'elapsed_time_log', 'elapsed_time_cat', 'test_acc', 'test_cat_acc', 'aID_acc', 'tag_acc', 'org_user_acc', 'week_day_acc', 'question_number_acc', 'shift', 'org_user_ans_cumsum',
'org_user_time_acc', 'test_time_acc', 'test_cat_time_acc', 'aID_time_acc', 'tag_time_acc', 'testId_et_mean', 'testId_et_std', 'test_cat_et_mean', 'test_cat_et_std', 'aID_et_mean', 'aID_et_std',
'KnowledgeTag_et_mean', 'KnowledgeTag_et_std', 'week_day_et_mean', 'week_day_et_std', 'testId_et_log_std', 'testId_et_log_mean', 'aID_et_log_std', 'aID_et_log_mean',
'KnowledgeTag_et_log_std', 'KnowledgeTag_et_log_mean', 'test_cat_et_log_std', 'test_cat_et_log_mean', 'aID_et_mean', 'aID_et_std', 'week_day_et_log_std', 'week_day_et_log_mean',
'org_user_exp_test', 'org_user_exp_tag', 'userID_theta', 'assessmentItemID_beta', 'KnowledgeTag_beta', 'testId_beta', 'test_cat_beta', 'question_number_beta', 'assessmentItemID_elo',
'testId_elo', 'KnowledgeTag_elo'

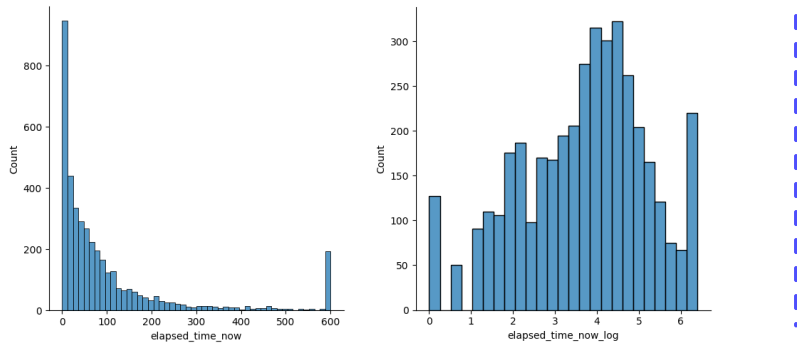
02. 데이터/전처리

Overview 파생변수

Elapsed time의 log transform

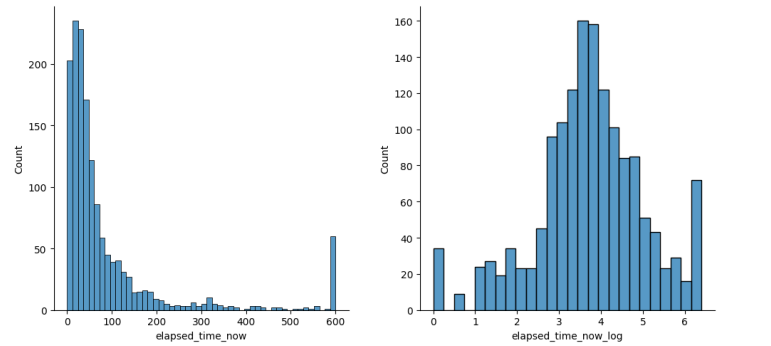
714

< KnowledgeTag의 elapsed time >



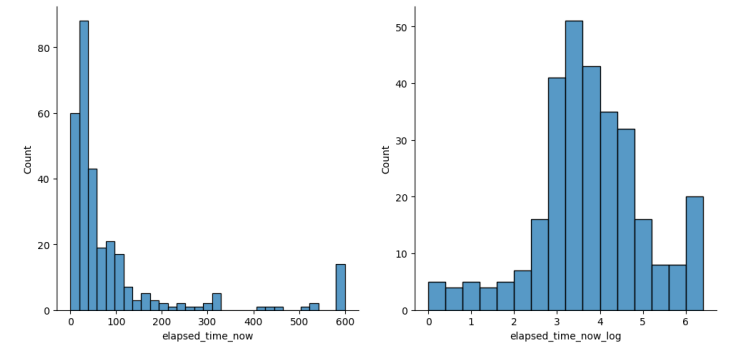
A020000172

< TestId의 elapsed time >



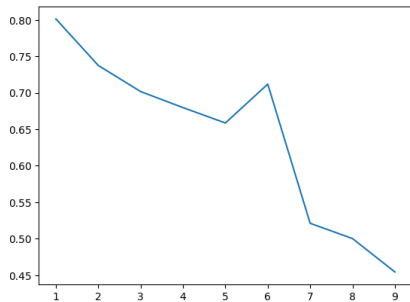
A020172001

< assessmentItemID의 elapsed time >

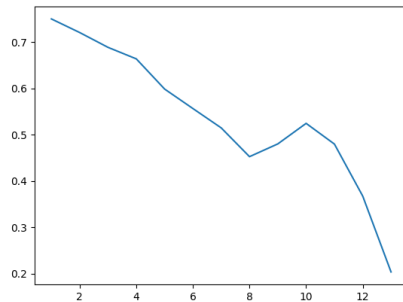


Mean과 median의 활용

< 시험지 대분류에 따른 정답률 >



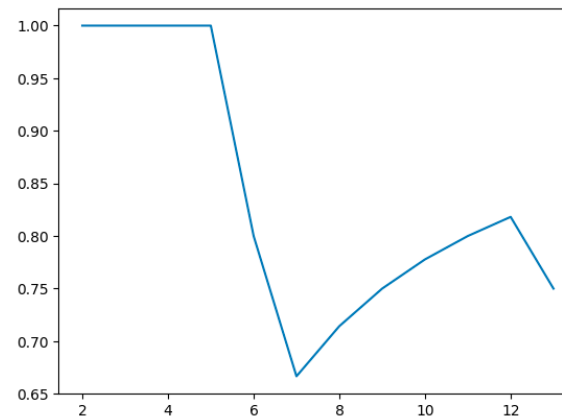
< 문제 번호에 따른 정답률 >



시험지 대분류/ 문제 번호에 따른 정답률에 유의미한 차이가 있다고 판단

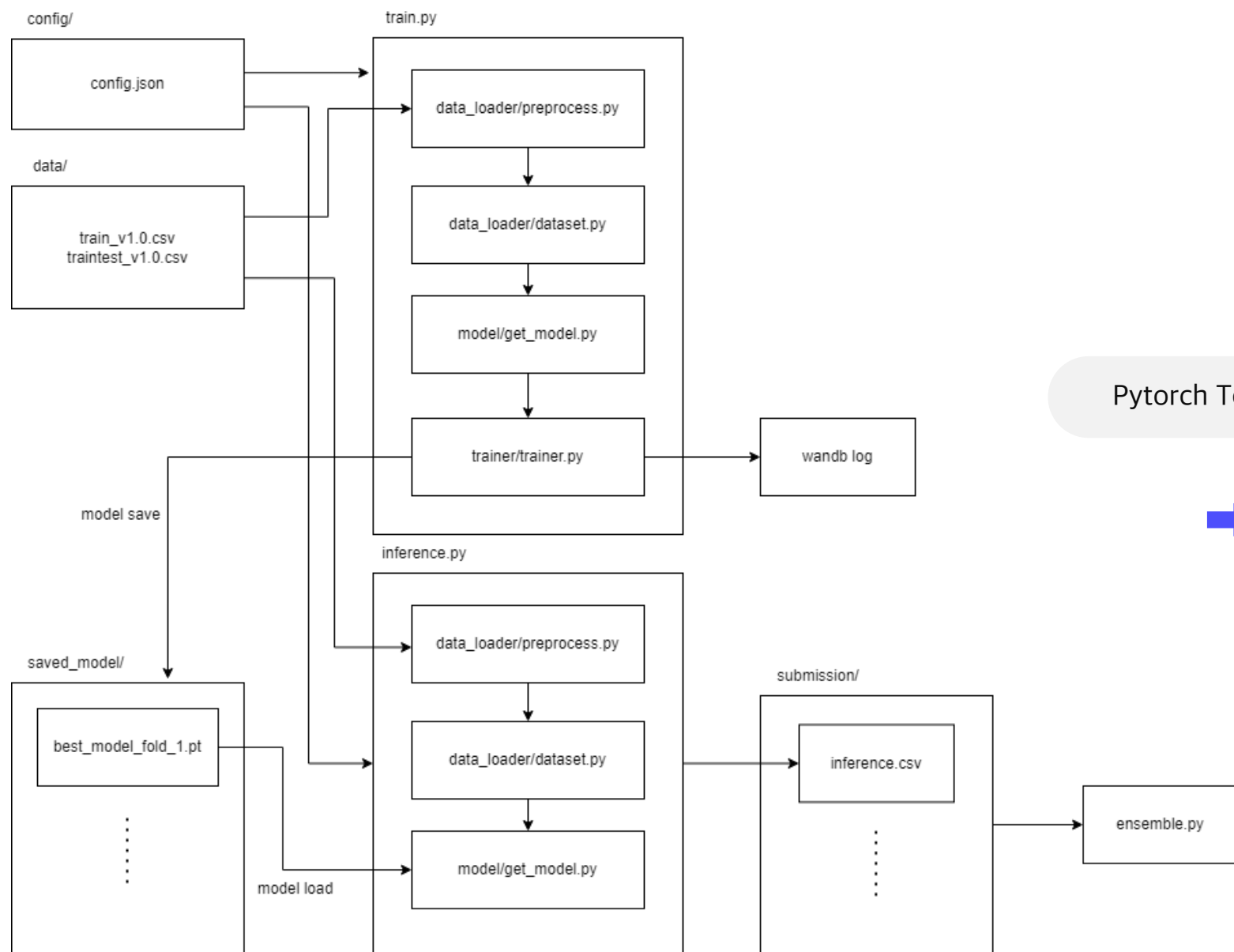
시간의 흐름을 고려한 피쳐

< 특정 시험의 시간 흐름에 따른 정답률 변화 양상 >



규칙성은 파악하지 못했지만 시간의 흐름에 따른 차이가 유의미하다고 판단하여 시간의 흐름에 따른 정답률/elapsed time 피쳐 또한 생성

03. 베이스라인



Pytorch Template

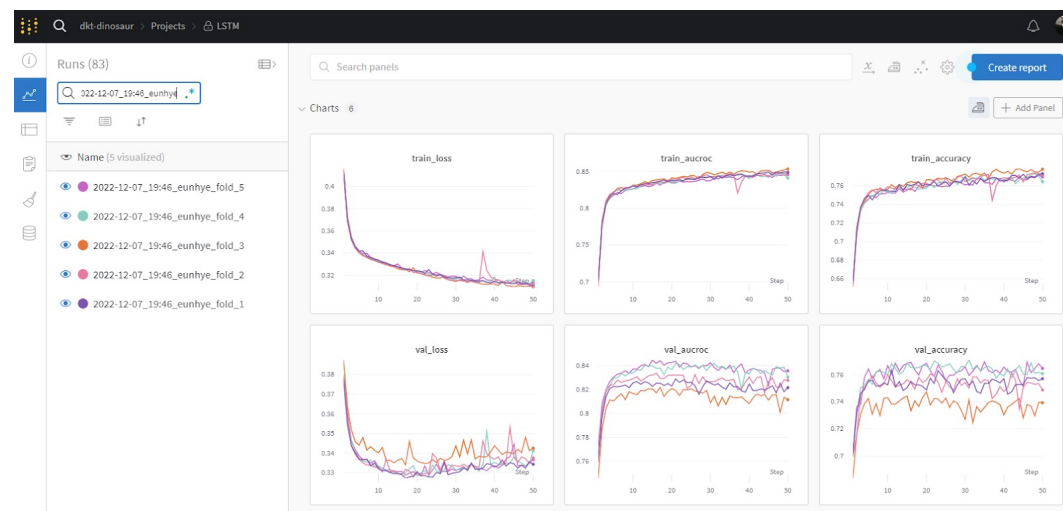


제공된 baseline code

➡ 팀만의 **고유한** baseline 구조 생성

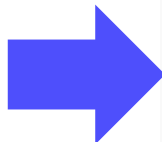
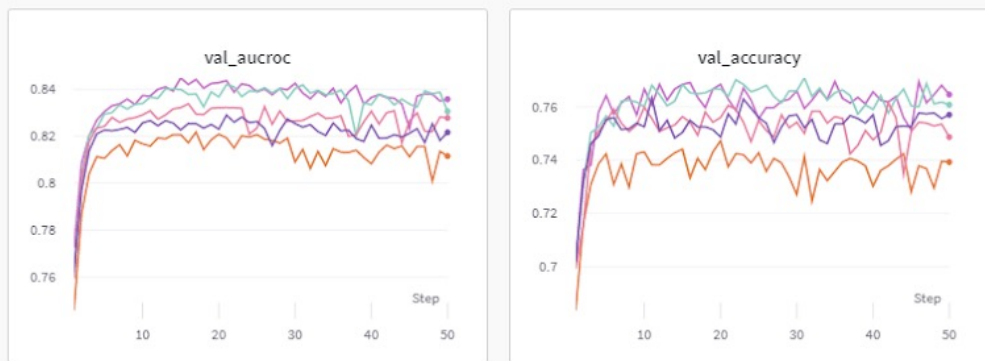


Projects				Create new project
Q Search	1-8 of 8			
Name	Last Run	Runs	Contributors	
GRUtransformer	16 hours ago	255	5	
GtnGRU	17 hours ago	49	2	
LSTM	18 hours ago	83	3	
transformerLSTM	18 hours ago	68	2	
transformerGRU	21 hours ago	102	3	
GTN	2022-12-07	41	1	
transformer	2022-12-05	98	3	
test-project	2022-12-02	150	5	

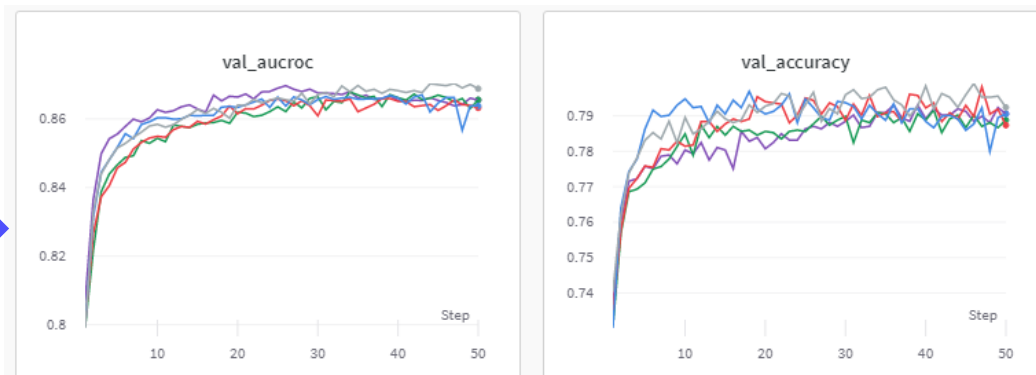


대분류 모델 프로젝트 -> 소분류 fold 단위로 wandb에 학습내역 출력
보수적 기록을 위해 최소 aucroc가 나온 fold를 노선에 기록

< k-fold shuffle 전 >



< k-fold shuffle 후 >



k-fold 진행 시 shuffle을 하지 않고 wandb 찍어본 결과
fold 별 경향성이 크게 다른 것 확인

public과의 align을 위해서는 valid 데이터를
잘 만드는 것이 중요하다고 판단함

k-fold shuffle 후 wandb 찍어본 결과
폴드 별 경향성 안정되는 것 확인

public 제출 결과와도 어느 정도 align됨

LSTM & Transformer

TransformerLSTM

TransformerGRU

GRUTransformer

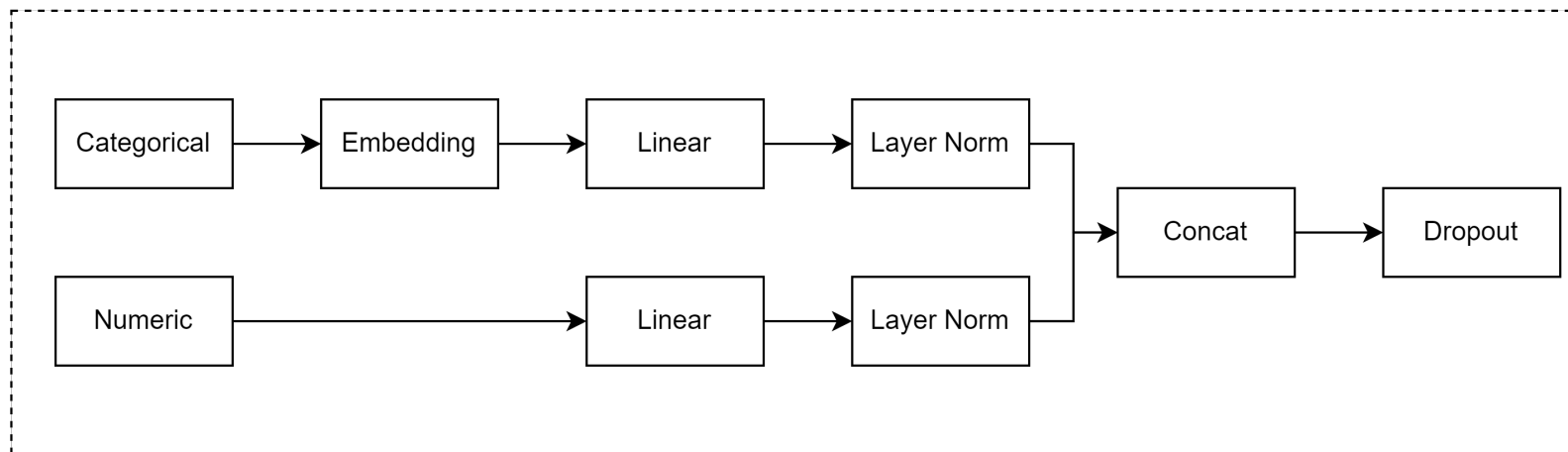
GTN

GTNGRU

강의에서 소개한 Riid대회 1등솔루션인

TransformerLSTM을 만들기 위한 Transformer, LSTM 단독 모델 제작

Feature Embedding



Categorical feature와 numerical feature를 나누어 embedding 한 후 concat

LSTM & Transformer

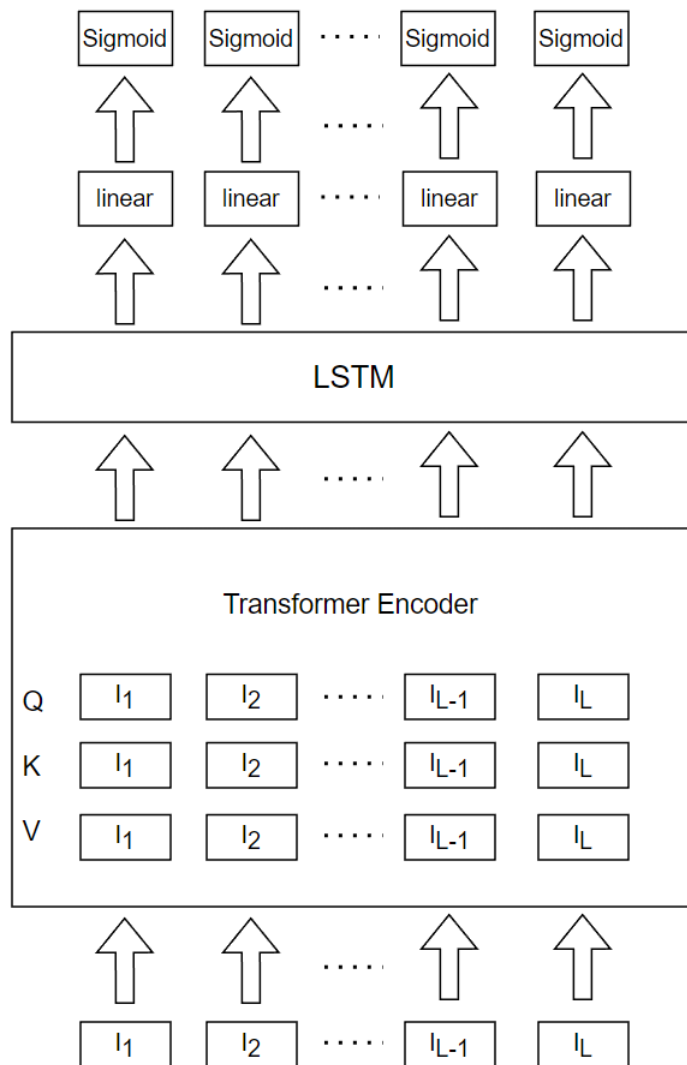
TransformerLSTM

TransformerGRU

GRUTransformer

GTN

GTNGRU



● Riid vs 우리대회

상대적인 데이터 부족

-> 전체 시퀀스에 대한 loss 계산

Sequential 특징을 반영하기 위한 LSTM 모델 병합

Positional Encoding을 제외한 transformer 모델

1. 마지막 문제를 예측하는 것이 목표인 만큼
2. LSTM 단에서 sequential 정보를 반영하는데 두 번 반영하면 좋지 않은 영향이 있을 것이라 판단

04. 모델링

Overview

모델소개

LSTM & Transformer

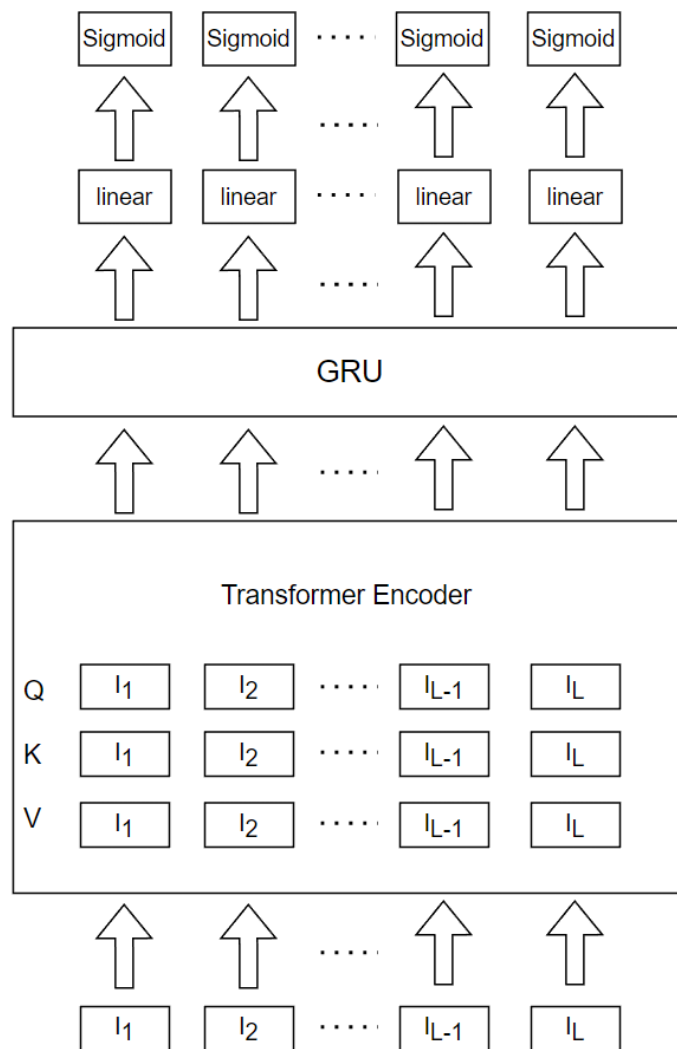
TransformerLSTM

TransformerGRU

GRUTransformer

GTN

GTNGRU



LSTM의 개선 모델인 GRU를 활용함으로써 모델 학습 속도 개선



학습 속도는 빠르지만 성능은 거의 유사함을 확인

04. 모델링

Overview 모델소개

LSTM & Transformer

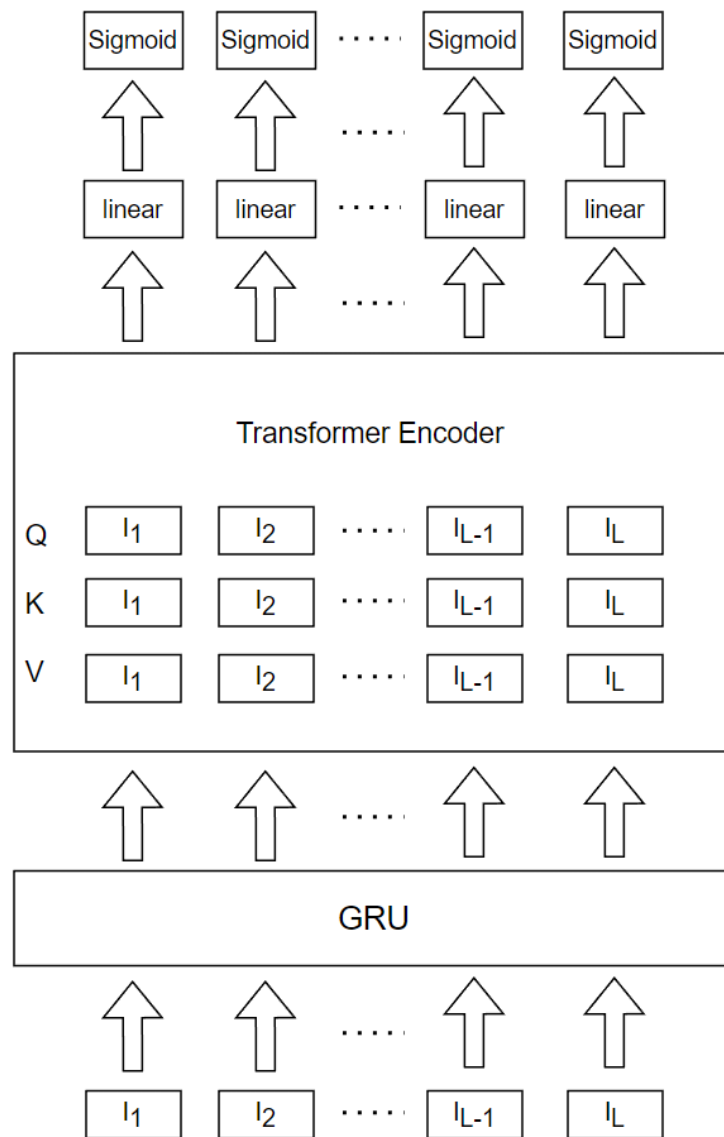
TransformerLSTM

TransformerGRU

GRUTransformer

GTN

GTNGRU



Sequential 정보를 반영하는 LSTM계열의 모델을
Positional encoding을 제거한 트랜스포머 보다 먼저 사용해 본다면?



TransformerGRU와 동일 조건으로 성능 비교한 결과
약간의 성능 개선

LSTM & Transformer

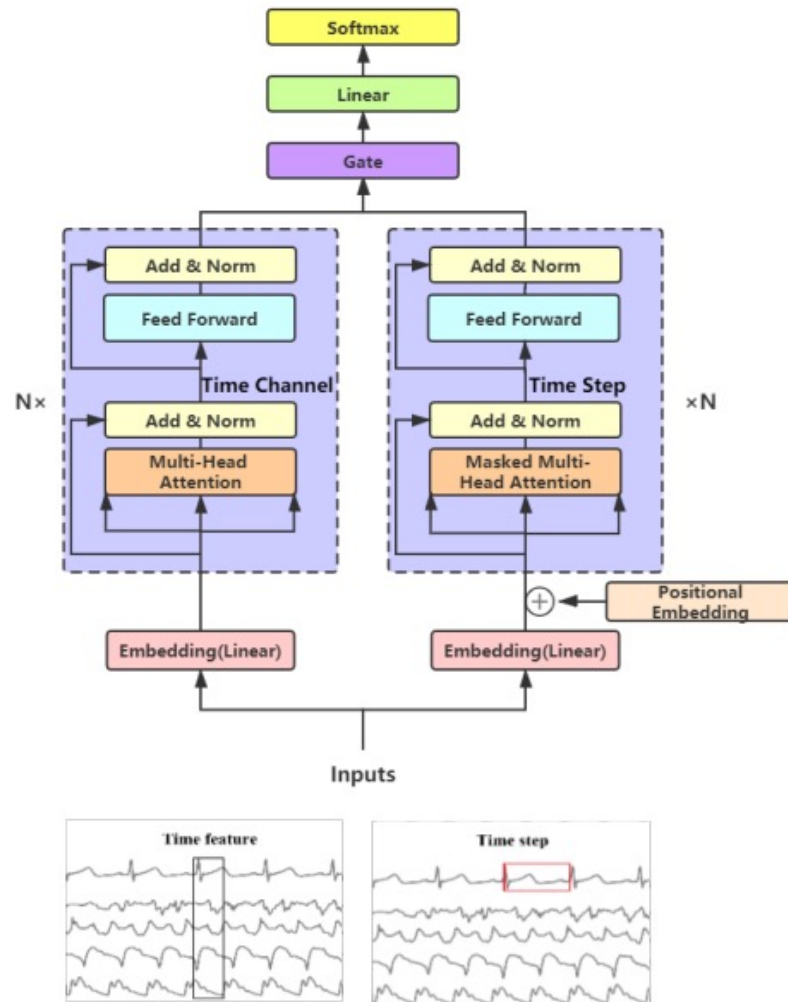
TransformerLSTM

TransformerGRU

GRUTransformer

GTN

GTNGRU



Transformer 성능이 준수한 것을 확인
Transformer 관련 모델을 더 써보는 것을 시도

Gated Transformer Network

Positional Embedding이 있는 모델과
Positional Embedding이 없는 모델을 concat 하는 모델



- LSTM 계열 모델을 사용하지 않아 속도 개선
- 기존 모델들과 비슷한 성능

LSTM & Transformer

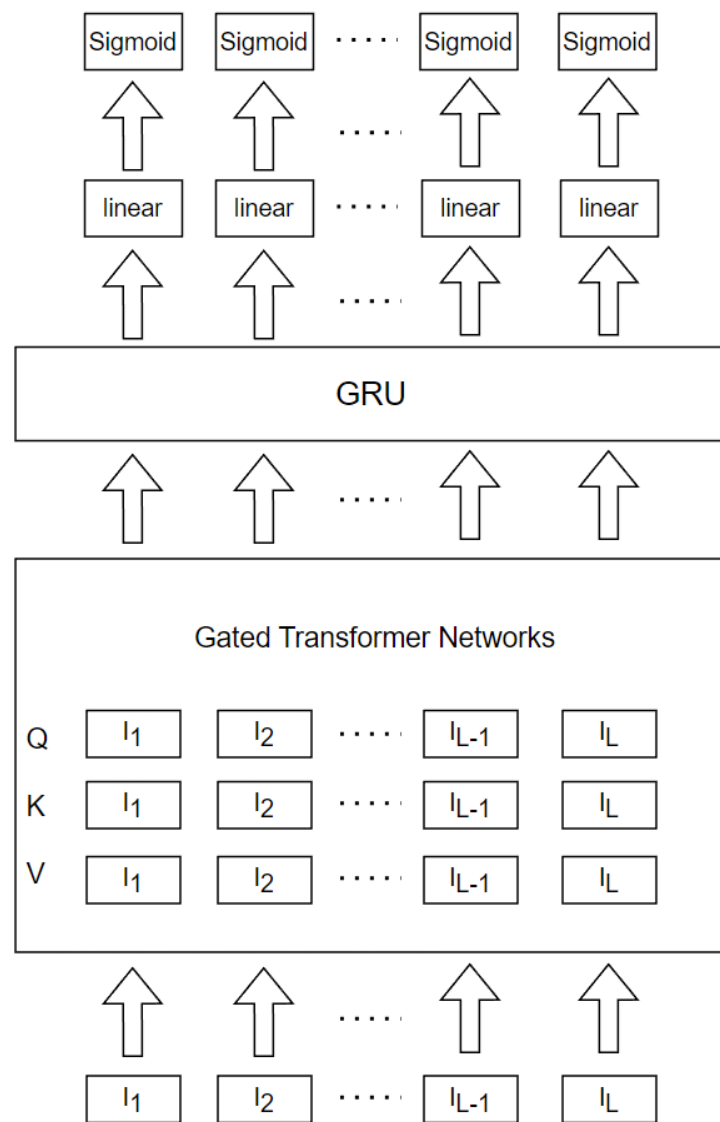
TransformerLSTM

TransformerGRU

GRUTransformer

GTN

GTNGRU



LSTM과 같이 sequential 특징을
반영하는 모델이 준수한 성능을 냈음

Gated Transformer Network + GRU

GTN 뒤에 GRU를 부착해 sequential 정보를 한 번 더 반영



- Auroc는 기존 모델들과 유사한 양상
- Accuracy가 불안정해 최종 모델로 채택 하지 않음

05. 성능개선

Feature Selection

Ensemble

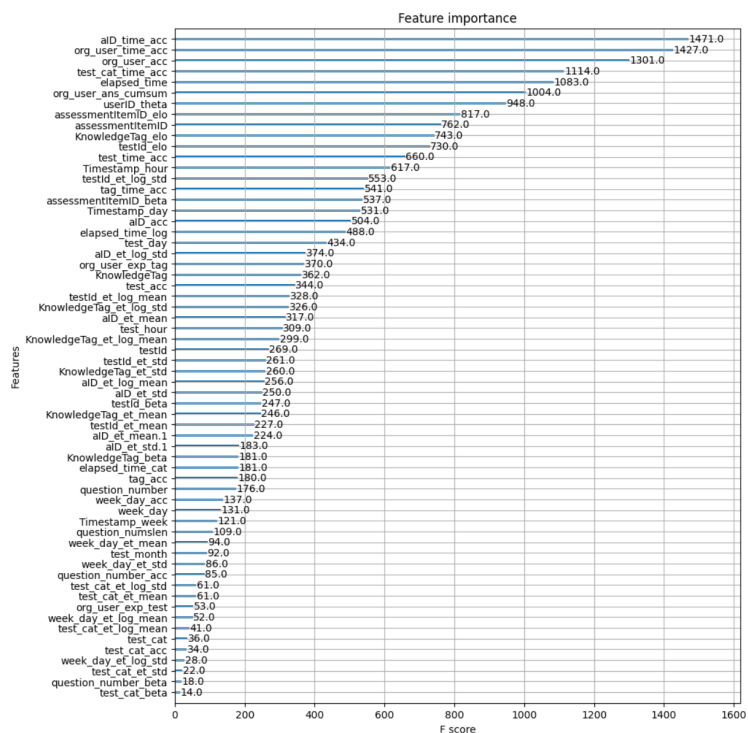
Parameter Tuning

Data Augmentation

약 60개 이상의 feature를 생성한 만큼 중복되거나 의미 없는 피처를 제거 할 필요

다양한 방법을 통해 feature들의 중요도 확인

XGBoost



RandomForest

```
std = np.std([tree.feature_importances_ for tr
indices = np.argsort(importances)[::-1]

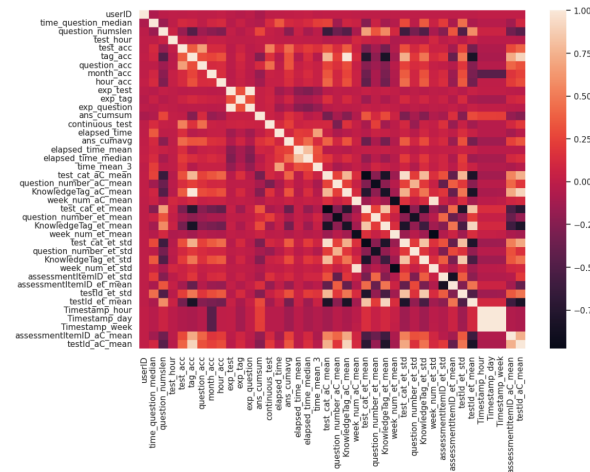
for f in range(data_X_vif.shape[1]):
    print(" {}. feature {} ({:.3f})".format(f,
```

1. feature elapsed_time (0.120)
2. feature elapsed_time_mean (0.094)
3. feature ans_cumsum (0.089)
4. feature elapsed_time_median (0.073)
5. feature time_question_median (0.072)
6. feature tag_acc (0.055)
7. feature KnowledgeTag_aC_mean (0.053)
8. feature test_hour (0.047)
9. feature hour_acc (0.047)
10. feature KnowledgeTag_et_mean (0.042)
11. feature month_acc (0.041)
12. feature exp_tag (0.039)
13. feature KnowledgeTag_et_std (0.036)
14. feature week_num_et_mean (0.025)
15. feature week_num_aC_mean (0.025)
16. feature week_num_et_std (0.025)
17. feature question_number_et_mean (0.020)
18. feature question_number_aC_mean (0.019)
19. feature question_number_et_std (0.016)
20. feature question_numslen (0.015)
21. feature test_cat_aC_mean (0.015)
22. feature test_cat_et_std (0.014)
23. feature test_cat_et_mean (0.012)
24. feature exp_question (0.003)
25. feature exp_test (0.003)

Stepwise Feature Selection



Feature Correlation



Ensemble

Inference 결과의 bias를 줄여주기 위함

유사한 성능을 내지만 correlation이 낮은 inference 결과끼리 앙상블 해야 한다!

비슷한 계열의 모델들 끼리의 correlation 결과는 모두 0.92~0.99

=> 실제로 LB 제출 결과 성능 개선이 이루어지지 않았음

```

43] ✓ 0.3s
...
lstm:1 0.9817474915695275
lstm:2 0.9844585710637431
lstm:3 0.9871839806768654
lstm:4 0.9798770249906582
lstm:5 0.9776009317032818
lstm:6 0.969559650598699
lstm:7 0.9684699425255836
lstm:8 0.9753999744427667
lstm:9 0.9507379571646466
  
```

상대적으로 성능은 조금 낮았으나 correlation이 낮은 XGBoost와의 앙상블

=> Private 점수 확인 결과 오히려 다른 앙상블 결과 보다 좋은 성능을 보임

with 1 :	0.8674299193178432		
with 2 :	0.8737356738910386		
with 3 :	0.8790969074440186		
with 4 :	0.897702086572		
with 5 :	0.87848705977	0.8266 → 0.8620	0.7473 → 0.7796
with 6 :	0.87951771192		
with 7 :	0.87338695046		
with 8 :	0.90538167356	0.8274 → 0.8620	0.7473 → 0.7796
with 9 :	0.86414851191		
with 10 :	0.8824410388	0.8274 → 0.8617	0.7500 → 0.7823
with 11 :	0.8757875519		
with 12 :	0.8777420657557229		
with 13 :	0.8736658379733481		

05. 성능개선

Feature Selection

Ensemble

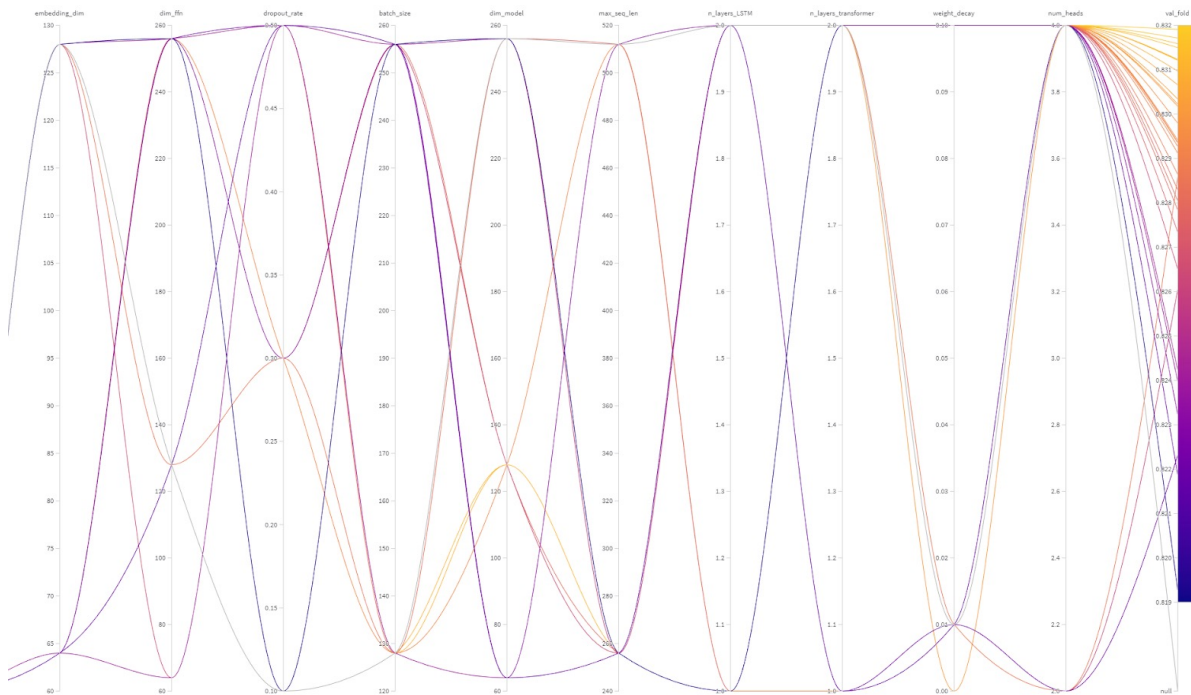
Parameter Tuning

Data Augmentation

Config parameter	Importance ① ↓	Correlation
batch_size	<div><div></div></div>	<div><div></div></div>
dim_model	<div><div></div></div>	<div><div></div></div>
dim_ffn	<div><div></div></div>	<div><div></div></div>
max_seq_len	<div><div></div></div>	<div><div></div></div>
dropout_rate	<div><div></div></div>	<div><div></div></div>
n_layers_LSTM	<div><div></div></div>	<div><div></div></div>

Wandb sweep을 이용해 파라미터 튜닝

- Feature의 importance와 성능과의 correlation 파악
- 파라미터의 대략적인 척도를 잡는데 도움이 됐음



05. 성능개선

Feature Selection

Ensemble

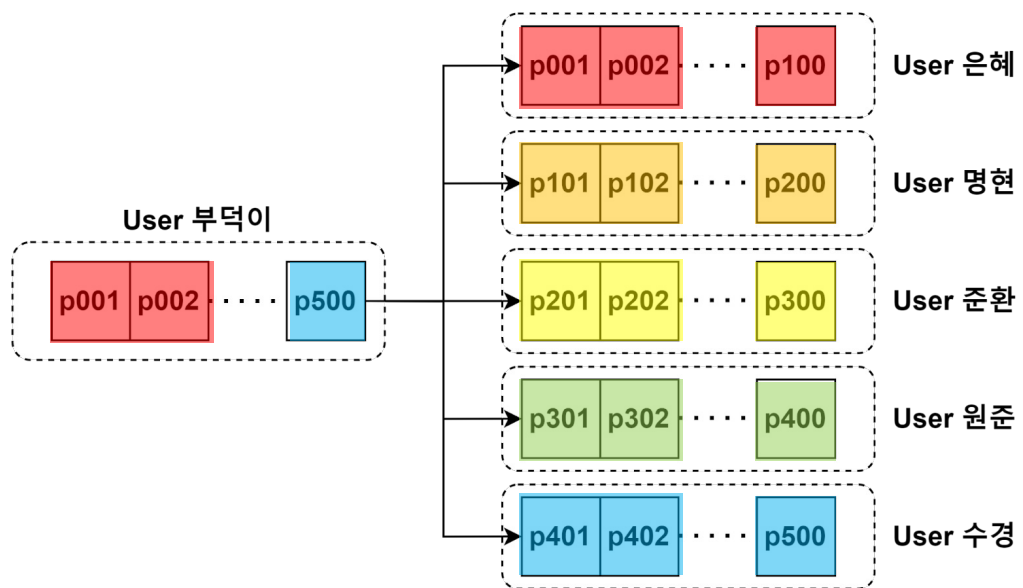
Parameter Tuning

Data Augmentation

두 가지 augmentation 진행

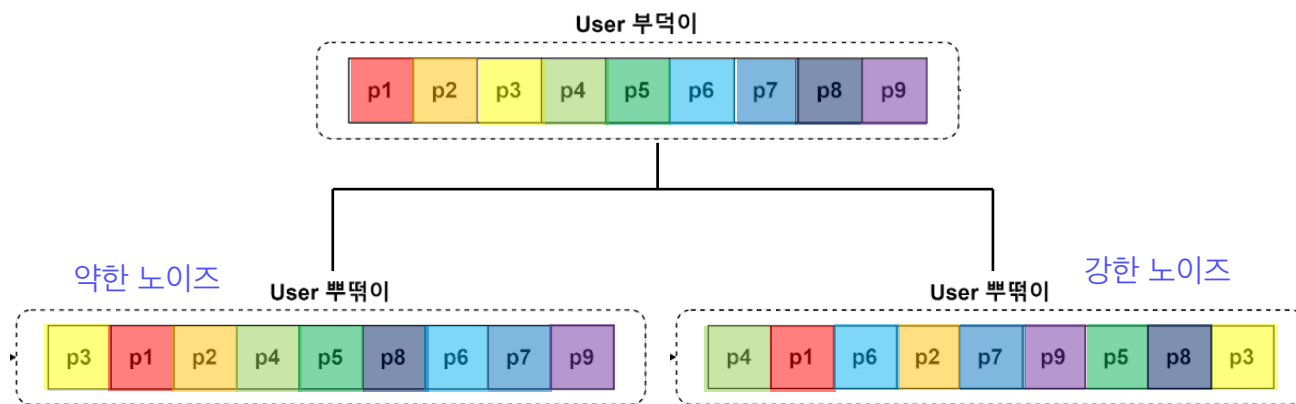
1.

max seq len보다 더 많은 문제를 푼 유저를
새로운 여러명의 유저로 구분



2.

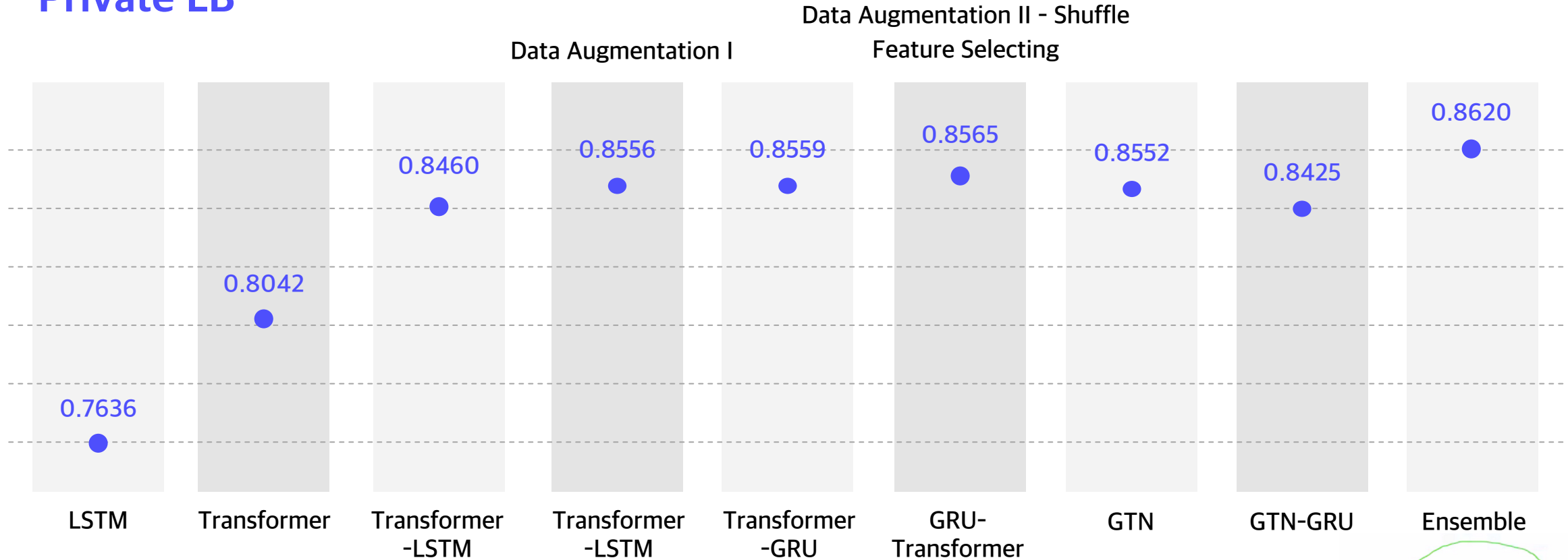
한 유저 안에서 문제의 순서를 섞은
새로운 유저 생성





이를 통해 좀 더 강건한 모델을 만들 수 있음

6. 최종 결과

Private LB



1 (3 ▲)	RecSys_01조		0.8563	0.7769	63
1	RecSys_01조		0.8563	0.7769	63

개구리 아니고 공룡입니다 ㅜㅜ

