kaggle days

14-15 NOVEMBER 2022
PARIS

ACM Recsys Challenge 2022 – Fashion Recommendations

Team NVIDIA Third Place Solution
Gilberto Titericz (Giba)

Organizers

LOGIC AI kaggle LVMH

SEPHORA LOUIS VUITTON Dior Christian Dior PARFUMS

CELINE GIVENCHY BVLGARI MoëtHennessy TIFFANY & CO.

About me

Giba

RAPIDS at NVIDIA
Curitiba, State of Paraná, Brazil
Joined 10 years ago · last seen in the past day
https://rapids.ai/

Followers 11075
Following 31

Competitions Grandmaster

Home | Competitions (248) | Datasets (20) | Code (1,160) | Discussion (998) | Followers (11,075) | Notifications | Account | Edit Public Profile

Competitions Grandmaster		
Current Rank 9 of 193,585	Highest Rank 1	
61	47	31
PetFinder.my - Pawpulari... · 10 months ago · Top 1%	1st of 3537	
Santander Value Predicti... · 4 years ago · Top 1%	1st of 4463	
Melbourne University AE... · 6 years ago · Top 1%	1st of 477	

Datasets Contributor		
Unranked		
0	0	2
Homecredit Best Solution · 4 years ago	9 votes	
segmentation-models-01... · 2 years ago	8 votes	
pytorch_pretrained_1 · a year ago	5 votes	

Notebooks Grandmaster		
Current Rank 110 of 243,120	Highest Rank 23	
17	19	29
The Property by Giba · 4 years ago	388 votes	
Building and Visualizing ... · 3 years ago	216 votes	
Giba R + data.table + Si... · 3 years ago	196 votes	

Discussion Grandmaster		
Current Rank 50 of 313,263	Highest Rank 6	
108	118	520
Data Scientist Hero · 7 years ago	525 votes	
1st PLACE - WINNER SO... · 7 years ago	475 votes	
1st Place Removed Soluti... · 2 years ago	389 votes	

Recsys Challenge 2022

The ACM Conference Series on
Recommender Systems

HOME RECSYS 2023 PAST CONFERENCES HONORS BLOG CONTACT

Timeline

When?	What?
7 March, 2022	**Start RecSys Challenge** Release dataset
14 March, 2022	**Submission System Open** Leaderboard live
14 June, 2022	**End RecSys Challenge**
21 June, 2022	**Final Leaderboard & Winners** EasyChair open for submissions
28 June, 2022	**Code Upload** Upload code of the final predictions
14 July, 2022	**Paper Submission Due**
1 August, 2022	**Paper Acceptance Notifications**
14 August, 2022	**Camera-Ready Papers**

RecSys Challenge 2022

About Participation Timeline Program Organization RecSys 2022

About

The RecSys Challenge 2022 is organized by Nick Landia (Dressipi), Bruce Ferwerda (Jönköping University, Sweden), Saikishore Kalloori (ETH Zürich, Switzerland), and Abhishek Srivastava (IIM Visakhapatnam, India).

Dressipi

Dressipi are the fashion-AI experts, providing product and outfit recommendations to leading global retailers.

Our recommendations enable retailers to create new product discovery experiences that are personalized and inspiring and can be used at all steps of the shopper journey.

Our algorithms enable retailers to make better buying and merchandising decisions by more accurately forecasting product demand and size ratios.

Our focus is to provide the world's best apparel recommendations and predictions. We do this by taking a domain specific approach across the data we collect and create, how we structure that data and the models we build. Everything we do is optimized to handle the nuances of fashion.

We work with brands across the US, UK, Europe, and Australia and outperform every competitor when A/B tested.

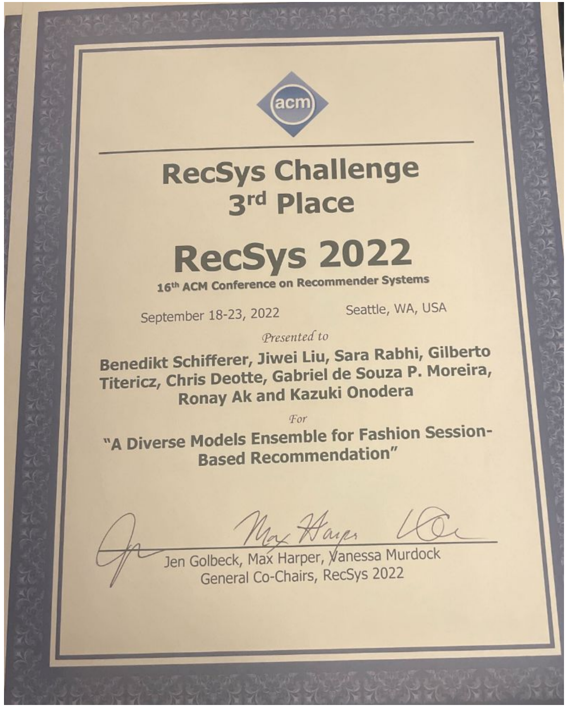
Nvidia Team

Benedikt Schifferer
DL Engineer

Chris Deotte
Sr. Data Scientist

Gabriel Moreira
Sr. Research Scientist

Gilberto Titericz
Sr. Data Scientist



acm

RecSys Challenge
3rd Place

RecSys 2022
16th ACM Conference on Recommender Systems

September 18-23, 2022 Seattle, WA, USA

Presented to

Benedikt Schifferer, Jiwei Liu, Sara Rabhi, Gilberto Titericz, Chris Deotte, Gabriel de Souza P. Moreira, Ronay Ak and Kazuki Onodera

For

"A Diverse Models Ensemble for Fashion Session-Based Recommendation"

Jen Golbeck, Max Harper, Vanessa Murdock
General Co-Chairs, RecSys 2022

Jiwei Liu
Sr. Data Scientist

Kazuki Onodera
Sr. Data Scientist

Ronay Ak
Sr. Data Scientist

Sara Rabhi
Research Scientist

Recommender Systems

Personalization Engine of Online Services

DIGITAL CONTENT

4.3B Watch Videos Online

E-COMMERCE

3.7B Shop Online

SOCIAL MEDIA

4.3B Active Users

DIGITAL ADVERTISING

4.7B Internet Users

"Already, 35 percent of what consumers purchase on Amazon and 75 percent of what they watch on Netflix come from product recommendations based on such algorithms."

Source: McKinsey

Session-based Recommendation Task

Users behaviour for different sessions might be very distinct

Session #1 - Looking for TVs

15 days later...

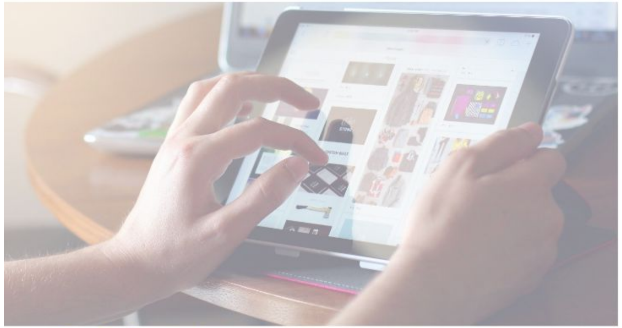


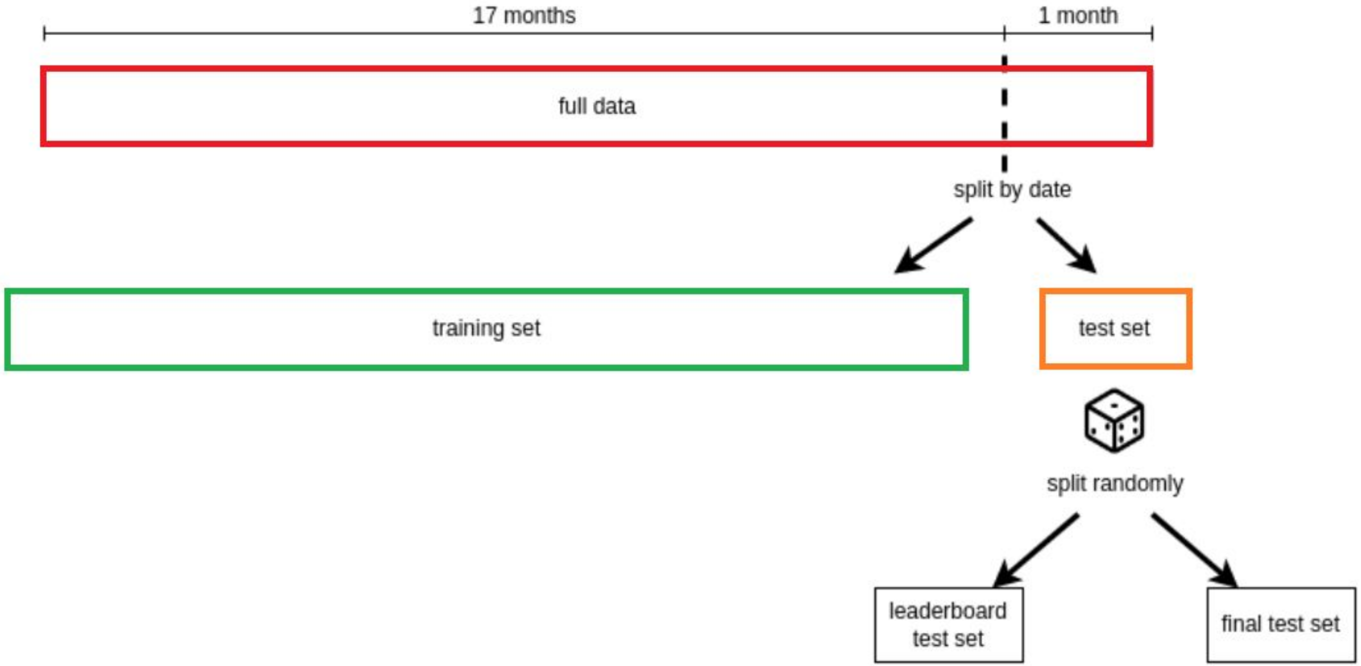
Photo by Taras Shypka on Unsplash

Session #2 - Browsing smartphones

Session Flow

Fig 1: Example Session and Purchase Data

Data Split



* Not allowed to fit models using test data.

Competition Dataset

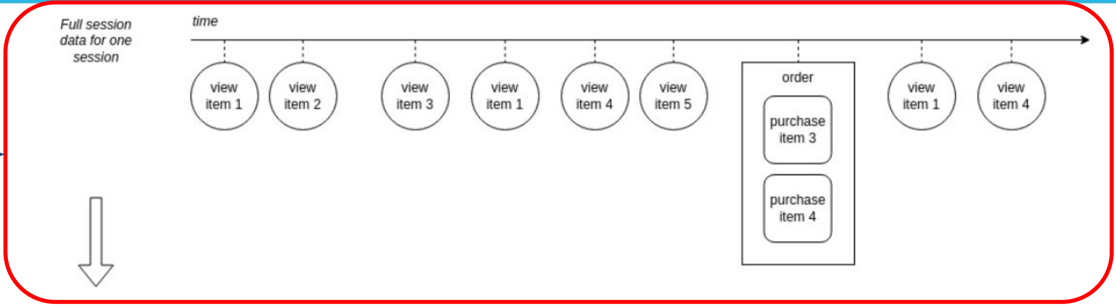
Train set:
- Number of sessions: 1,000,000
- Date range: Jan-2020 to May-2021 (514 days)
- Number of items: 23,496
- Average item views per section: 4.74

Test set leaderboard:
- Number of sessions: 50,000
- Date range: Jun-2021 (30 days)
- Number of items: 5,647 (65 new items not in trainset)
- Average item views per section: 4.52

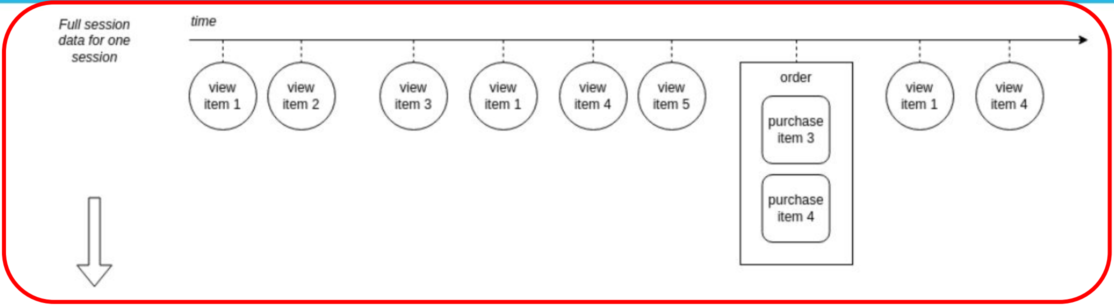
Test set final leaderboard:
- Number of sessions: 50,000
- Date range: Jun-2021 (30 days)
- Number of items: 5,648 (67 new items not in trainset)
- Average item views per section: 4.58

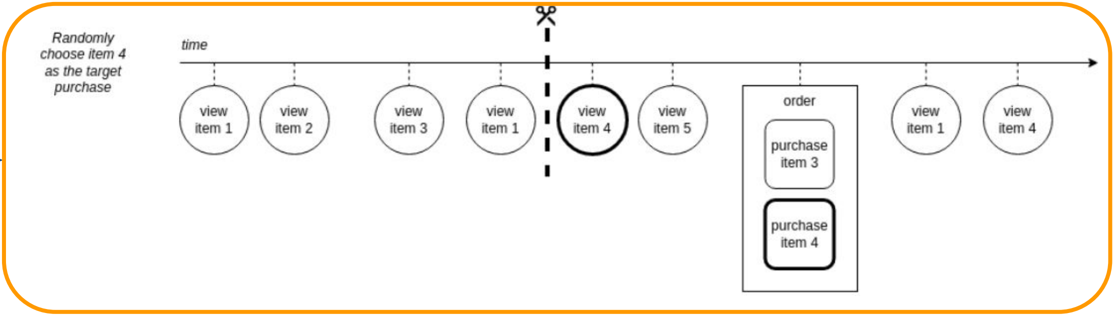
Data Cleaning

Original raw Session Data



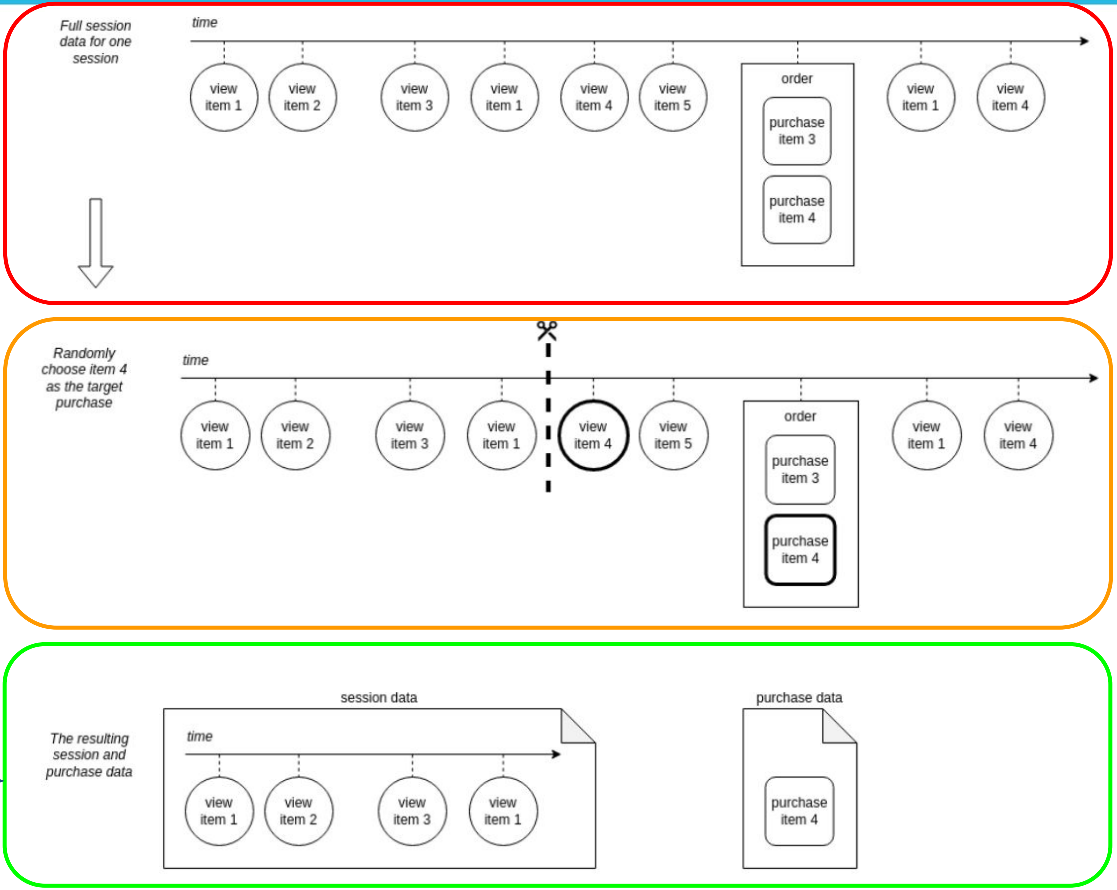
Data Cleaning





Select Target and Cut history

Data Cleaning



Trainset
Clean Data

Train set:

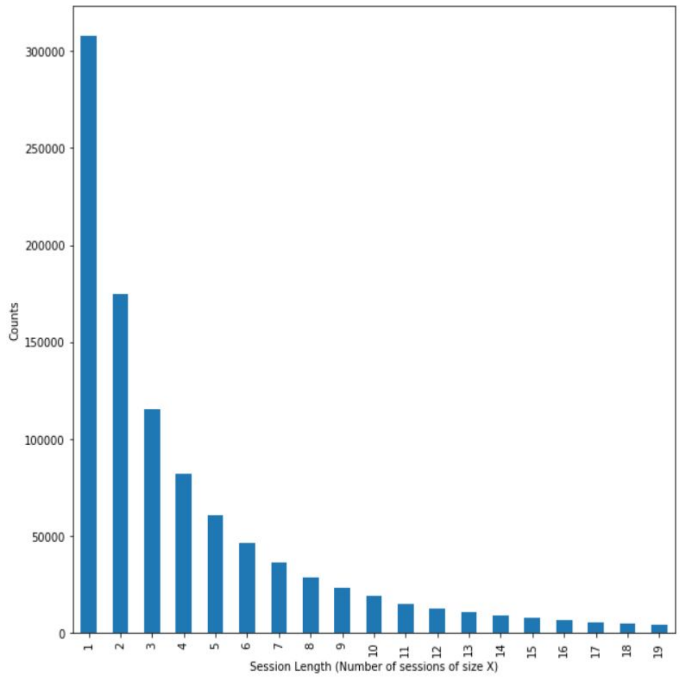
```
train.loc[train.session_id == 18].reset_index(drop=True)
```

	session_id	item_id	date	purchase_item_id	purchase_date
0	18	4026	2020-08-26 19:15:47.232	24911	2020-08-26 19:20:32.049
1	18	2507	2020-08-26 19:16:31.211	24911	2020-08-26 19:20:32.049
2	18	18316	2020-08-26 19:18:30.833	24911	2020-08-26 19:20:32.049

INPUT DATA

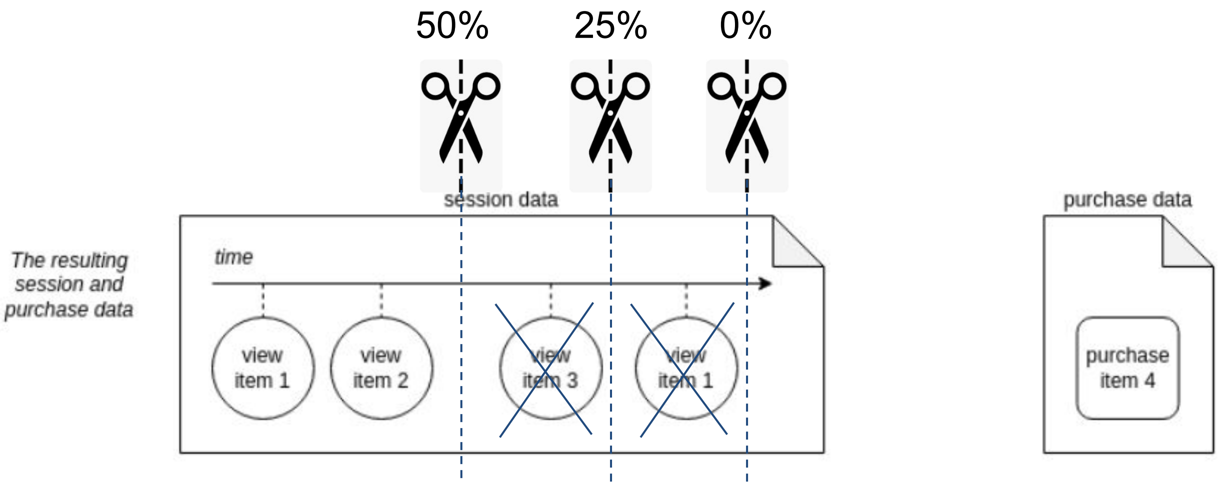
LABEL

Session Length Histogram



Test set

- Random cut sessions from 0 to 50%



Item Description:

- All item descriptions are anonymized

item_id	feature_category_id	feature_value_id
2	56	365
2	62	801
2	68	351
2	33	802
2	72	75
2	29	123
2	16	38
2	50	76
2	61	462
2	53	6
2	7	394
2	69	885
2	47	123

Evaluation Metric

Mean Reciprocal Rank @100 (MRR@100)

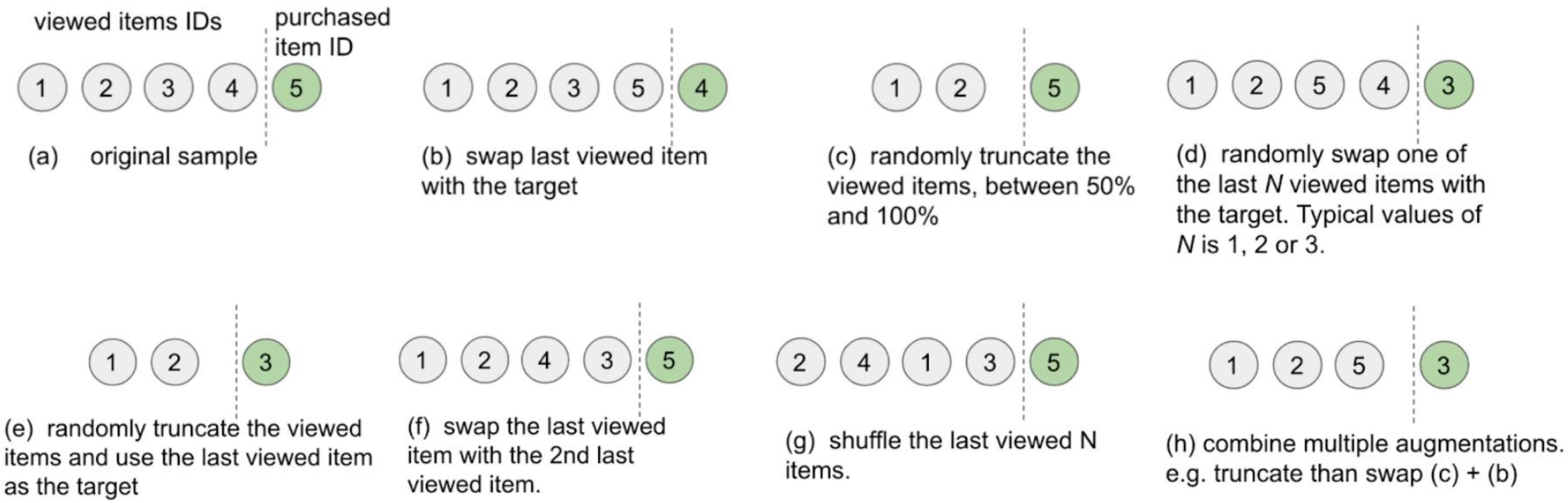
$$\text{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}_i}$$

https://en.wikipedia.org/wiki/Mean_reciprocal_rank

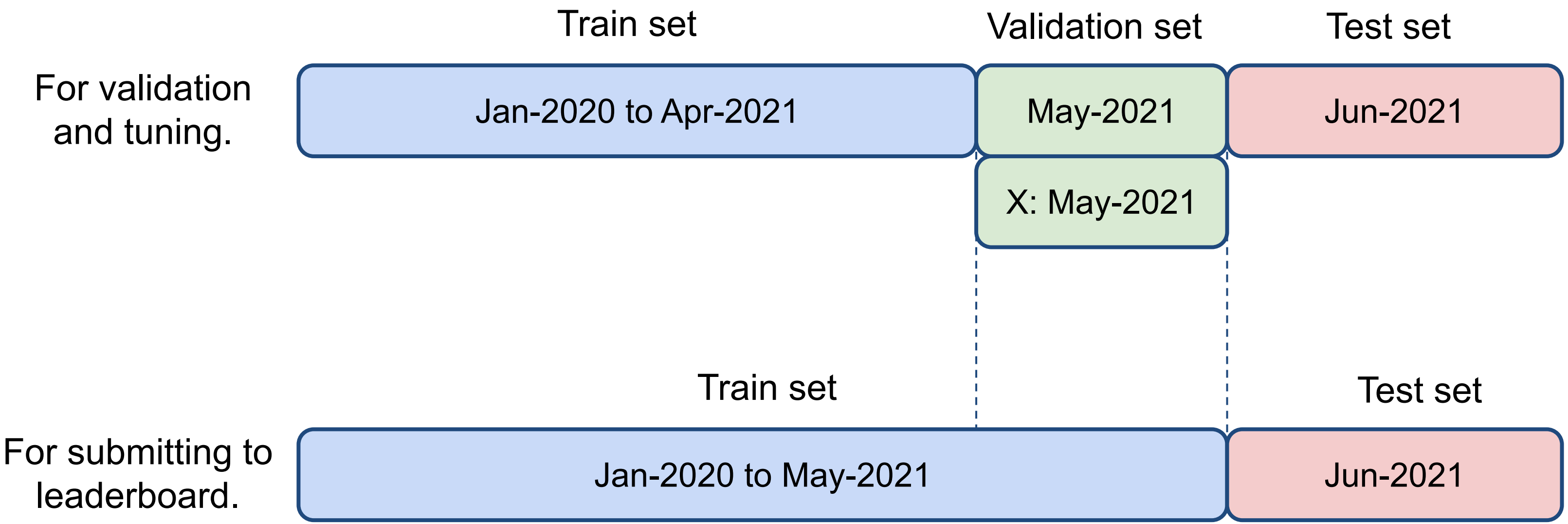
Ground Truth	Prediction	Rank	Reciprocal Rank
shirt	pants, socks, **shirt**	3	⅓ = 0.333
pants	**pants**, shirt, skirt	1	1/1 = 1.000

MRR@3 = (1 + 0.333)/2 ~ 0.67

Data Augmentation



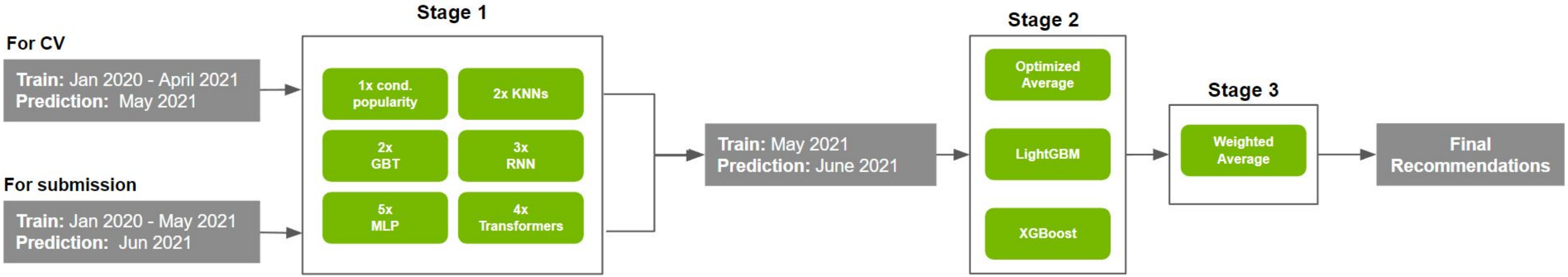
Validation Strategy:



* validation set X is random truncated 50% of items to match the test set distribution.

SOLUTION

3-stage pipeline of ensembling and stacking



- Last month of training data as local validation set
- Random 50% truncation of validation sessions
- Full re-training for submission

- 17 single models
- Optimize performance and diversity
- 6 models groups

- 3 stacking models
- Stage-1 candidates and in-session features as inputs
- Weighted average as final recommendation score

1. Time Weighted Conditional Popularity Model (Baseline):

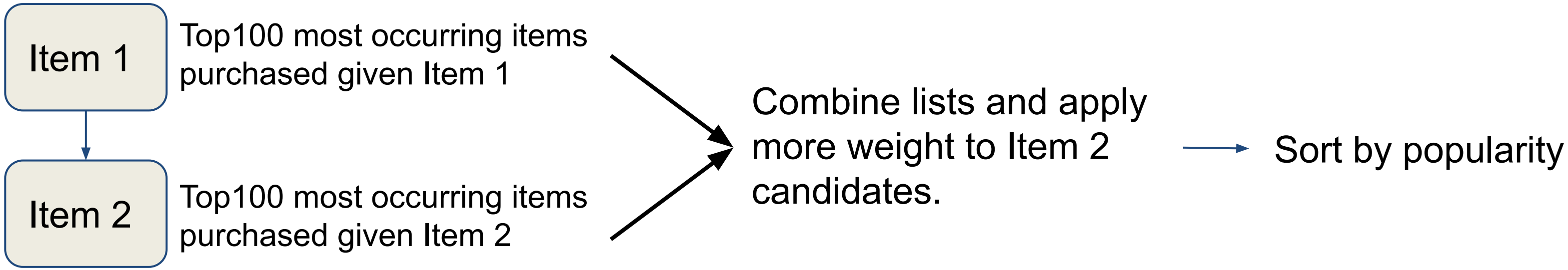
- Only two features: ‘item_id’ and ‘timestamp’.
- Uses data augmentation in trainset.

Algorithm:

- Calculate co-occurrence popularity of purchased **item_id** given previous viewed **item_ids.**

- Weight by time decay. Last viewed item_id gives more weight.

- For each item in the session, retrieve a list of top 100 most co-occurring items. Combine all top 100 for all items in the session and apply a time decay factor before sorting by popularity.



2. Session based Nearest Neighbors:

- Only two features: ‘item_id’ and ‘timestamp’.

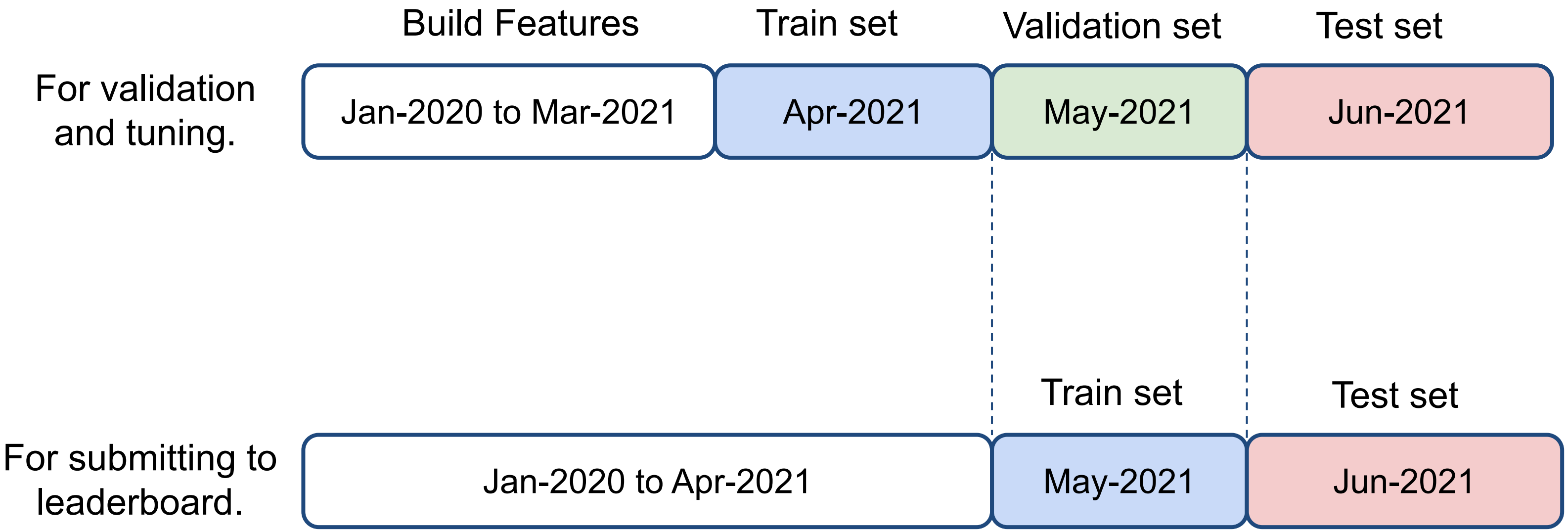
- Uses data augmentation in trainset.

- SkNN and VSTAN algorithms: Retrieve a list o similar sessions.
    - Fit on 4 periods of trainset: last 120 days, last 90 days, last 60 days, last 30 days.
    - Ensemble predictions scores for the 4 fits.

https://arxiv.org/abs/2012.08793: Session-based k-NNs with Semantic Suggestions for Next-item Prediction
https://doi.org/10.1145/3331184.3331322: Sequence and Time Aware Neighborhood for Session-based Recommendations: STAN

3. Gradient Boosting Decision Trees:

- Uses data augmentation in trainset.
- Convert the task to binary classification.
- Use 200 candidates item for each session.



3. Gradient Boosting Decision Trees:

- Train set with ~80k sessions x 200 candidates = 16M rows.

Binary Dataset

Session_id	Candidate number	Target Item_id	**Binary Target**	Other features (F.E.)
1	1	8	**0**	xxx
1	2	10	**1**	xxx
1	3	55	**0**	xxx
1	…	…		xxx
1	200	43	**0**	xxx

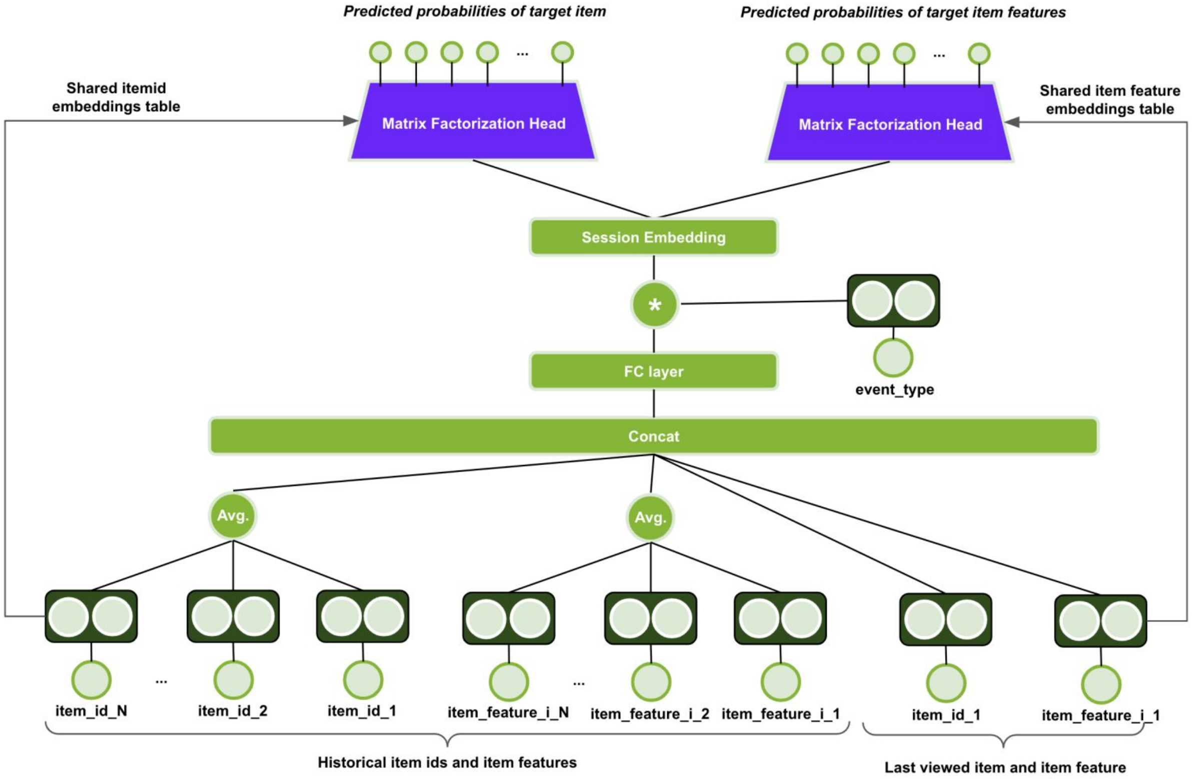
Features Based in Candidate Item:

- Purchase count of last N days.
- View Count of last N days
- Min, mean, max and std of purchase count of item pairs by day in last N days.
- Min, mean, max and std of view count of item pairs by day in last N days.
- Number of unique session containing the item.
- Purchase and view counts based in item properties at different time windows.

3. Gradient Boosting Decision Trees:

- XGBoost: using binary classification objective.
- LightGBM: using lambda rank objective.
- Hyperparameters tuned based in validation set metrics.

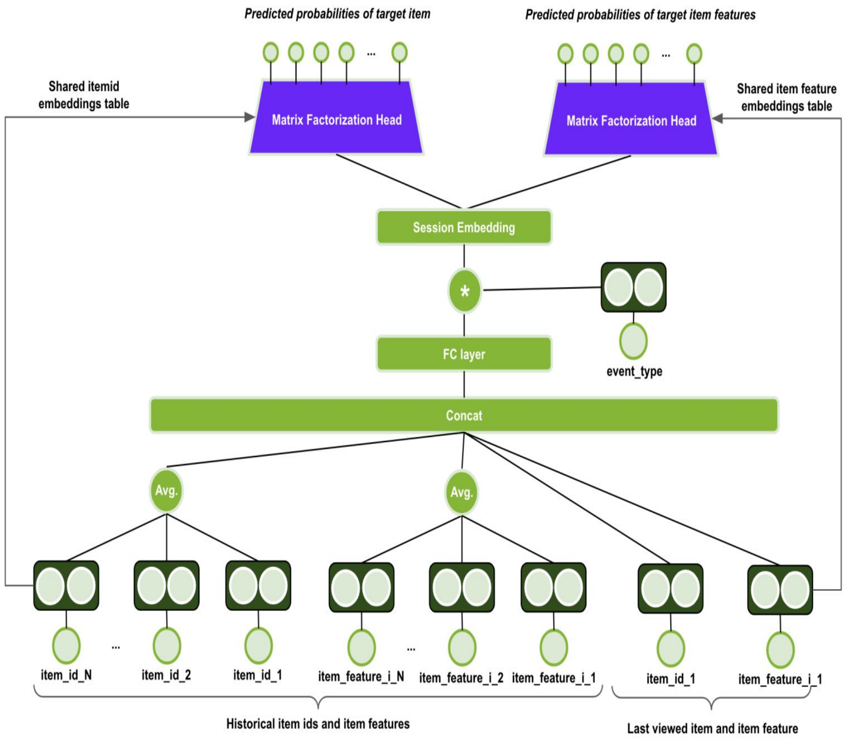
4. MultiLayer Perceptron (MLP):



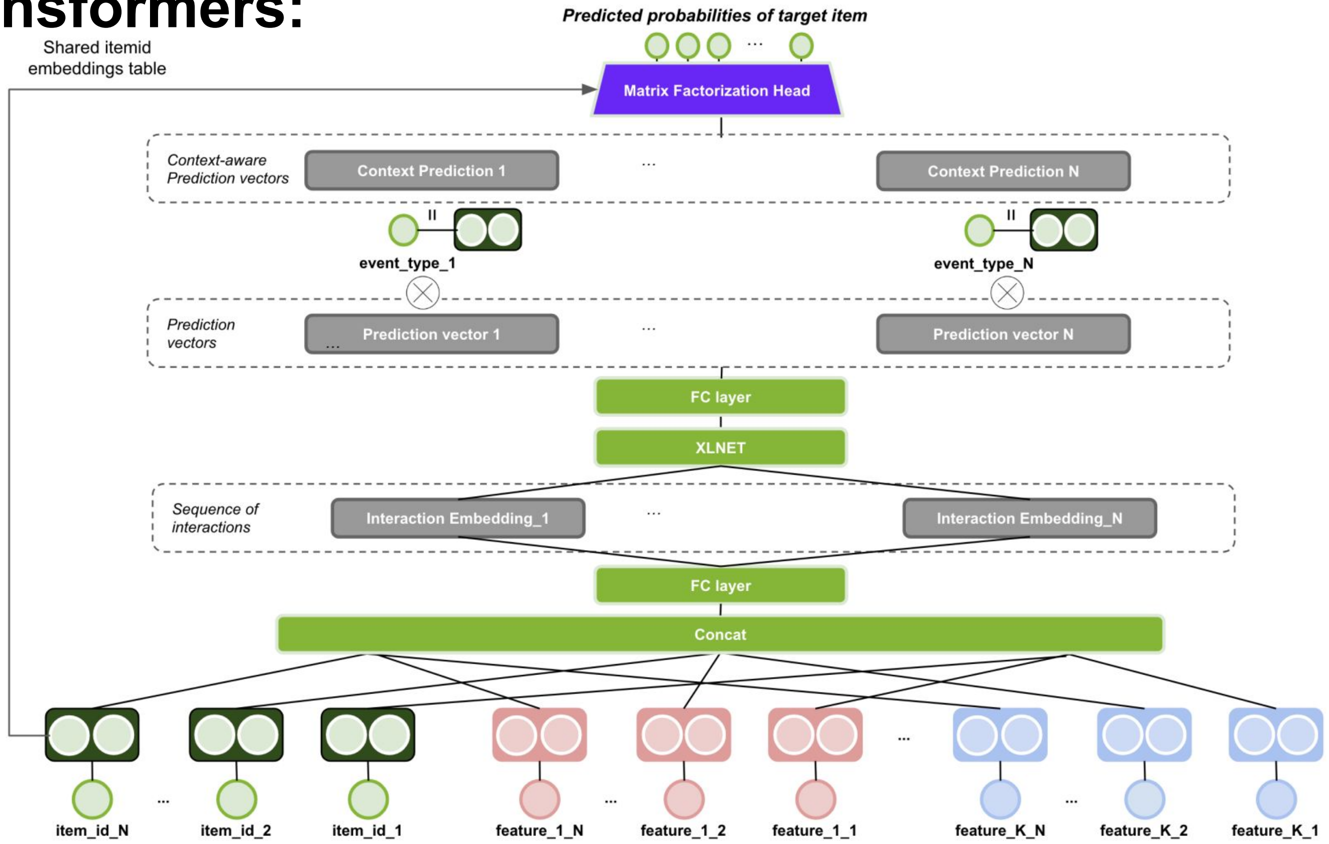
4. MultiLayer Perceptron (MLP):

- Trained on two tasks:
  - Predict final purchased item
  - Predict next item viewed or purchased.

- Multi task loss: Predict target item and target item features
- Bagging 6 times (run with different seeds)
- Stepwise learning rate decay

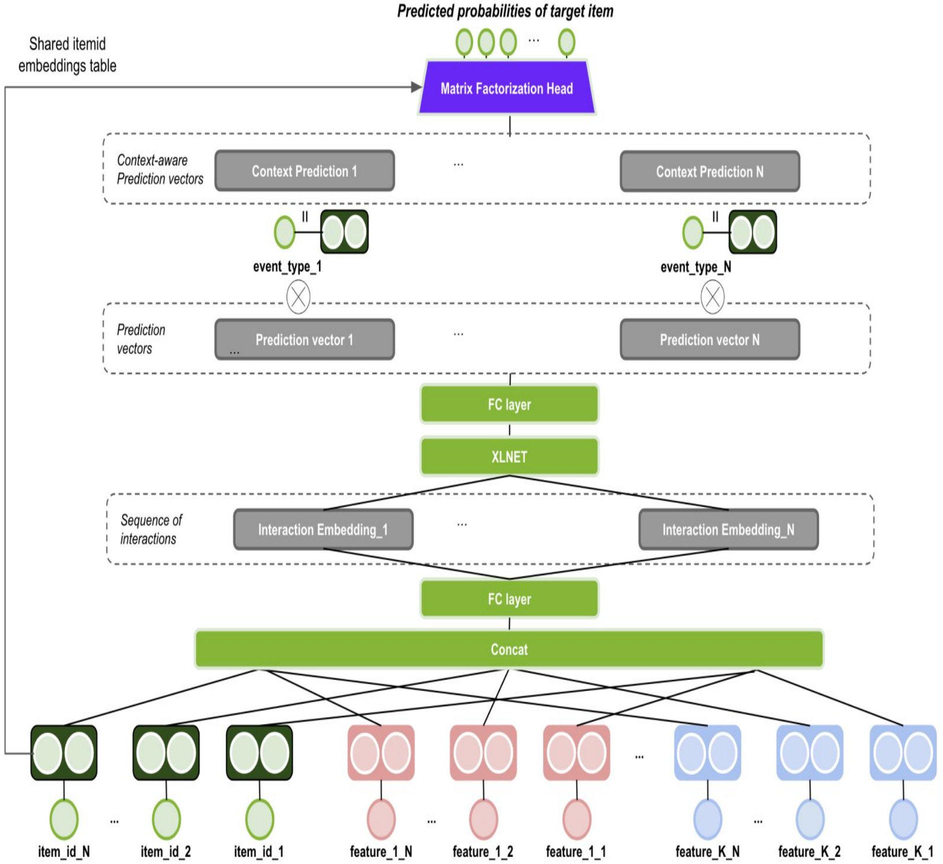


5. Transformers:

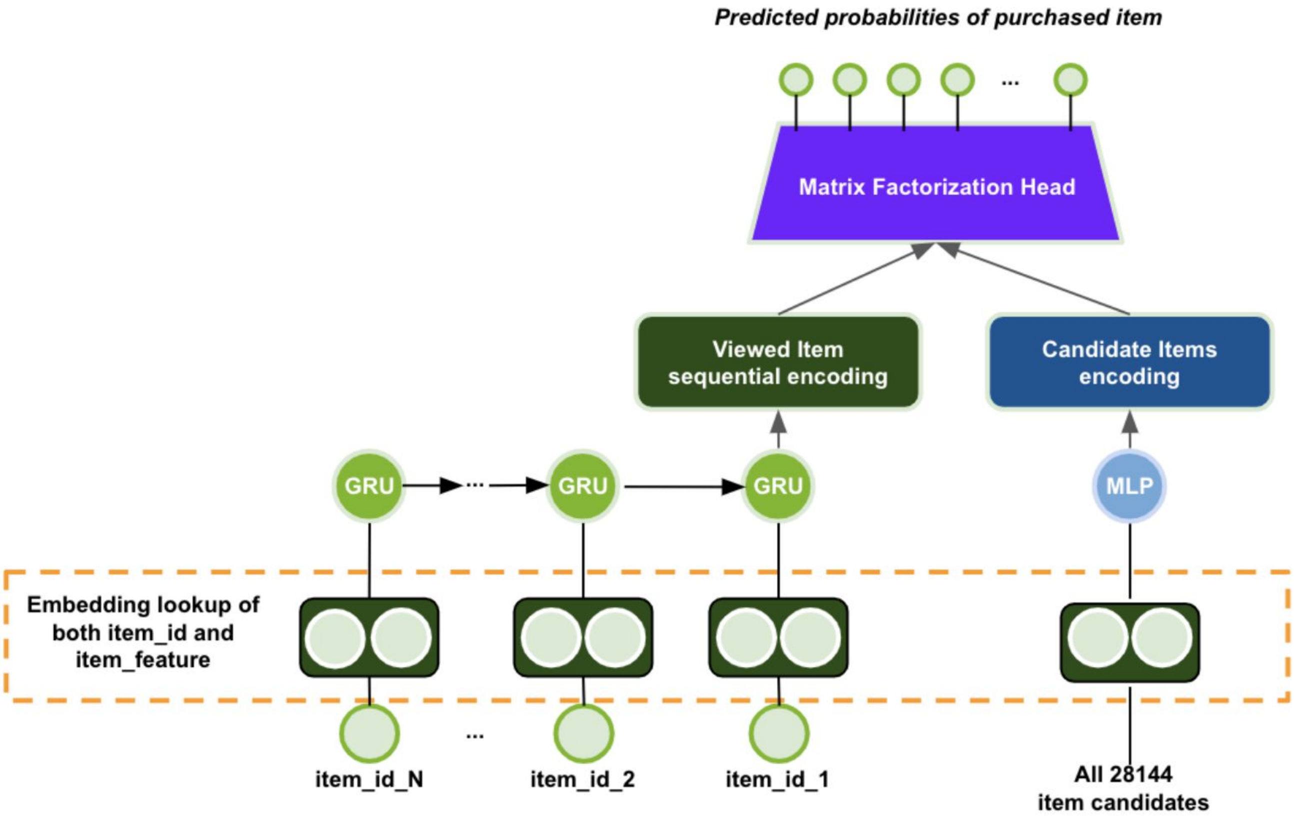


5. Transformers:

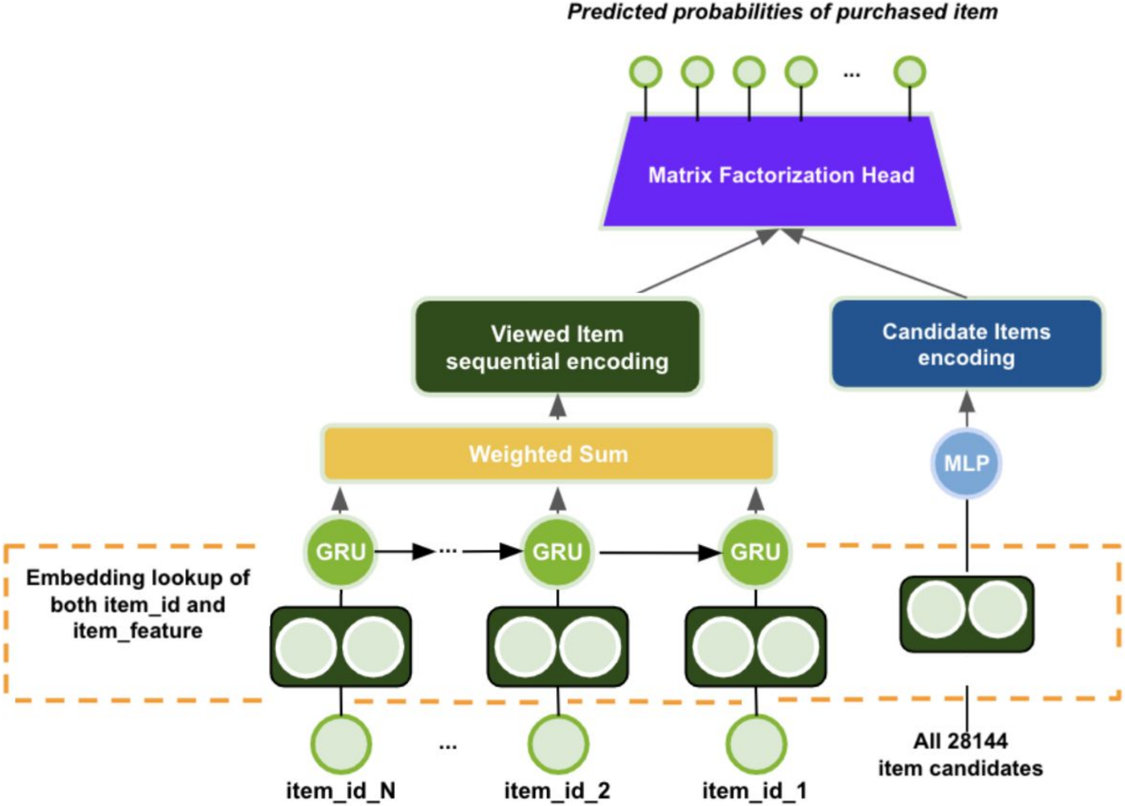
- Transformers4Rec library
- XLNet architecture
- Training using event_type flag boost score by +0.003



6. Recursive Neural Networks (RNN):

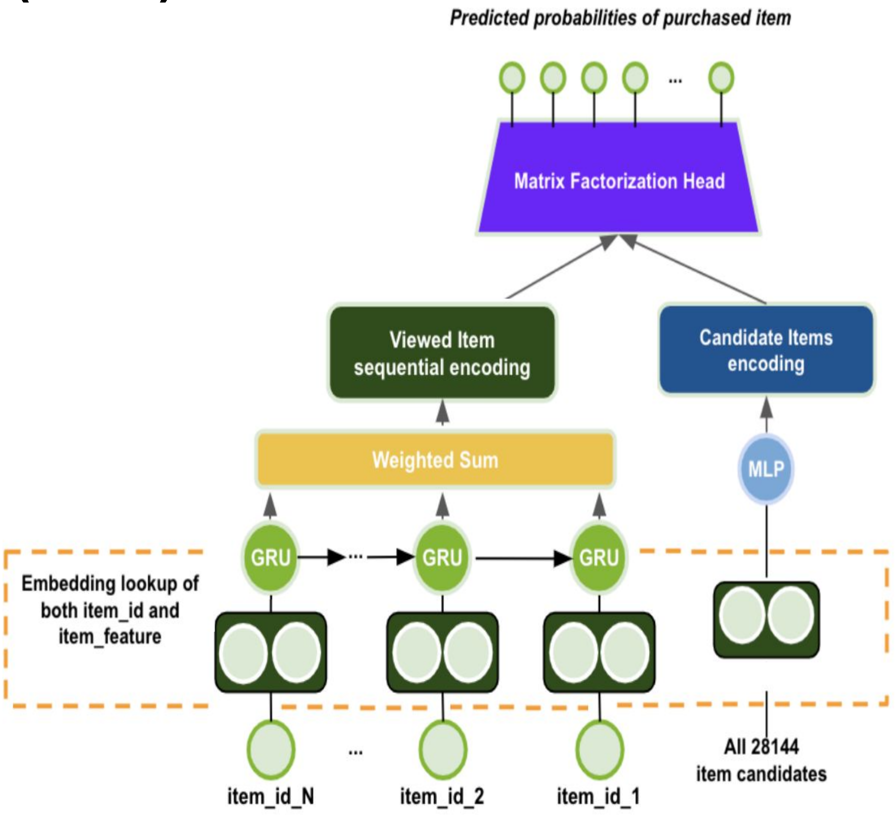


6. Recursive Neural Networks (RNN):



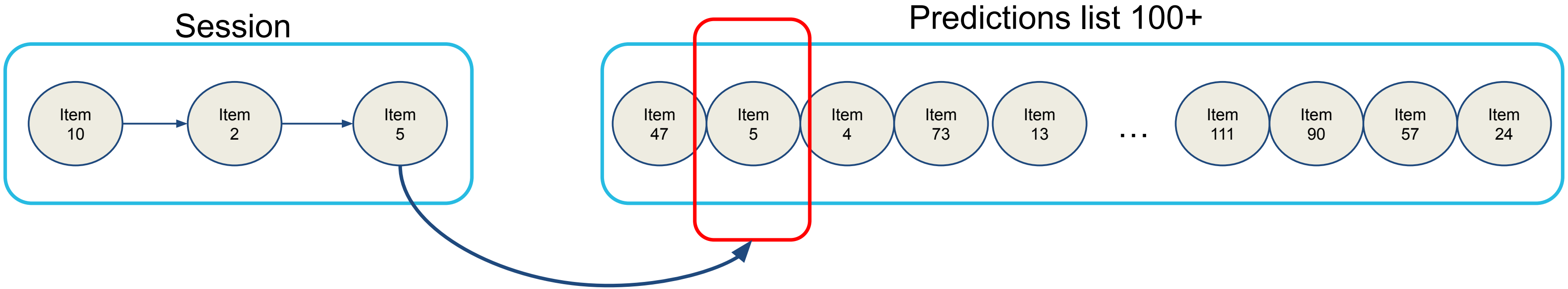
6. Recursive Neural Networks (RNN):

- Uses unidirecional GRU.
- Weighted Sum GRU embeddings.
- RNN are good for short sequences.
- Achieved best MRR@100 leaderboard metric over all single models: 0.2044



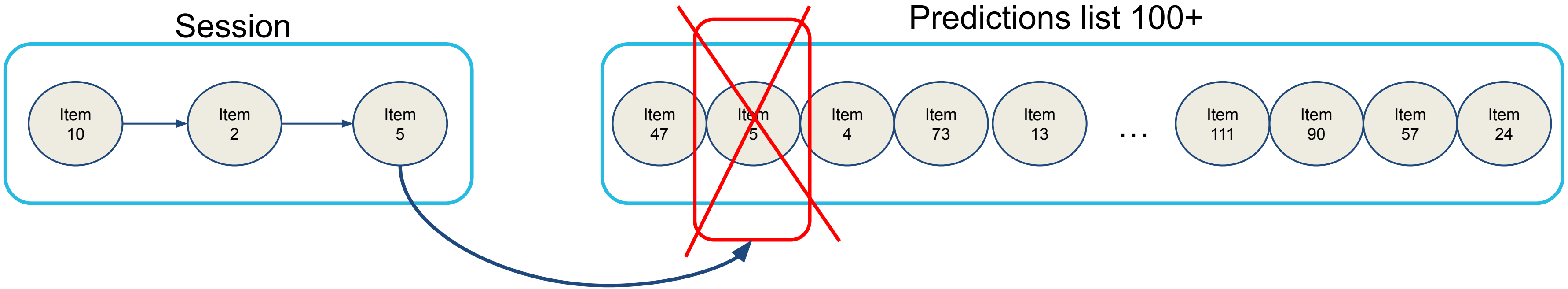
Post Processing:

- Filter out from the list of predictions any item previously viewed. (Boost +0.001 the metric)



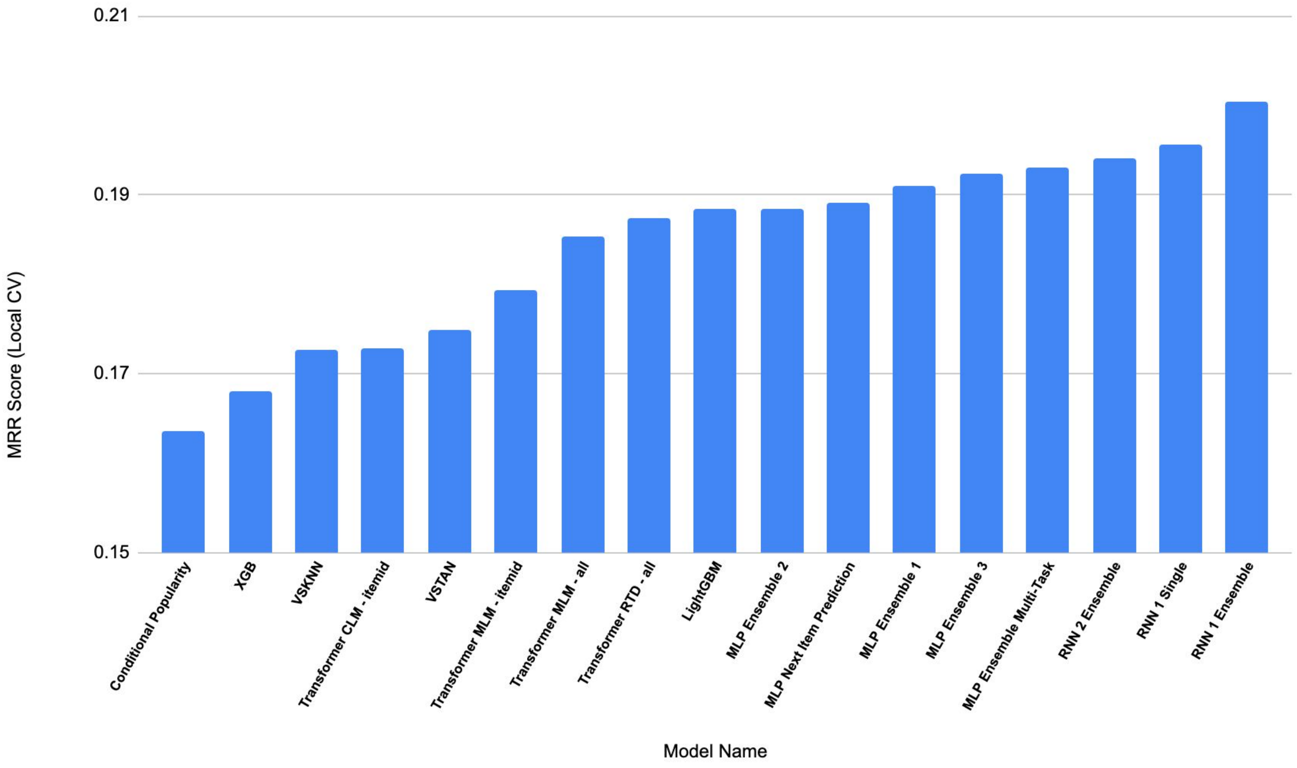
Post Processing:

- Filter out from the list of predictions any item previously viewed.
  (Boost +0.001 the metric)



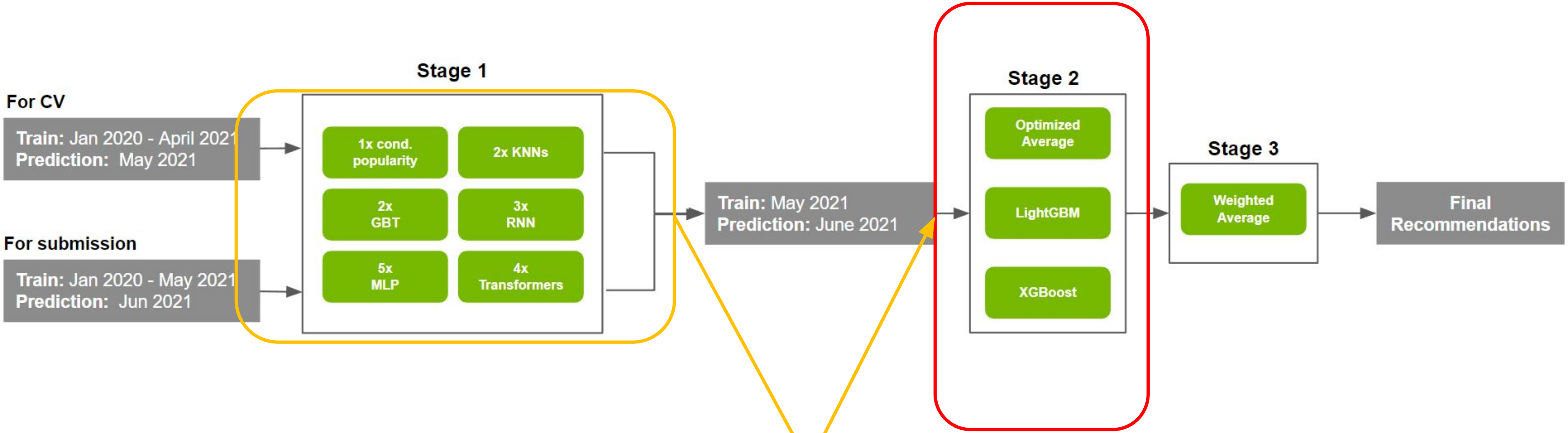
Single Models Scores :

Model	MRR@100
Conditional Popularity	0.164
VSTAN	0.174
Transformer4 Rec	0.185
LightGBM	0.187
MLP	0.194
RNN	0.204



Stacking Ensemble

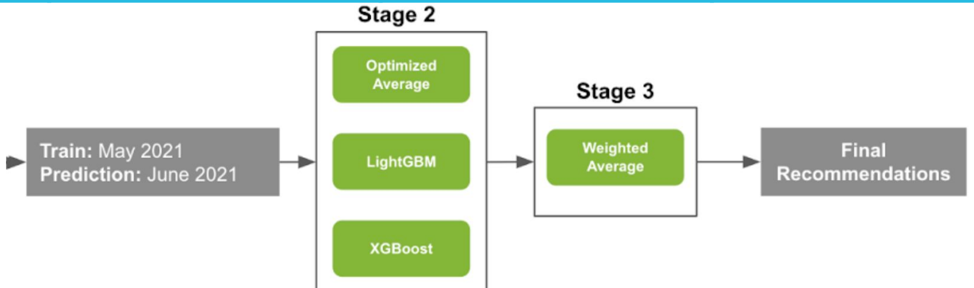
Levels 2 and 3



Build a Stage 2 dataset using item candidates predictions from Stage 1 models.

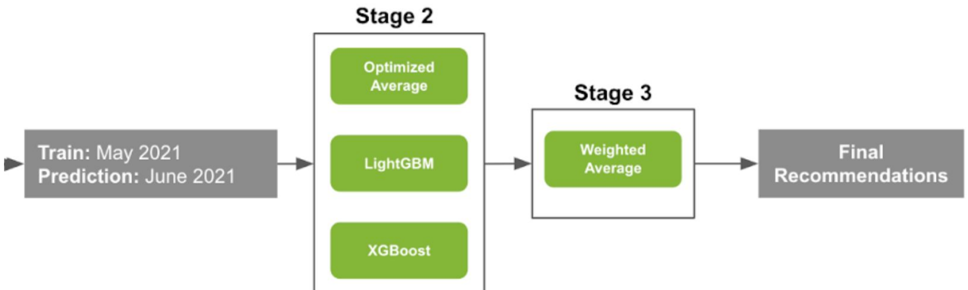
Ensembling Strategy

	Stage 2 - Optimization algorithm	Stage 2 - LightGBM	Stage 2 - XGBoost	Stage 3 - Weighted Average
Cross-validation strategy	No CV	25 random CV splits based on session-id (85%-15%)	3 seeds of temporal 4-fold splits	No CV
Objective function	Original MRR@100	Lambda Rank	Binary Cross-Entropy	Original MRR + Conservative constraint
Item candidates filtering strategy	Use all candidates from Stage1	Item-ids in at least two Stage1 models	Item-ids in at least two Stage1 models	-



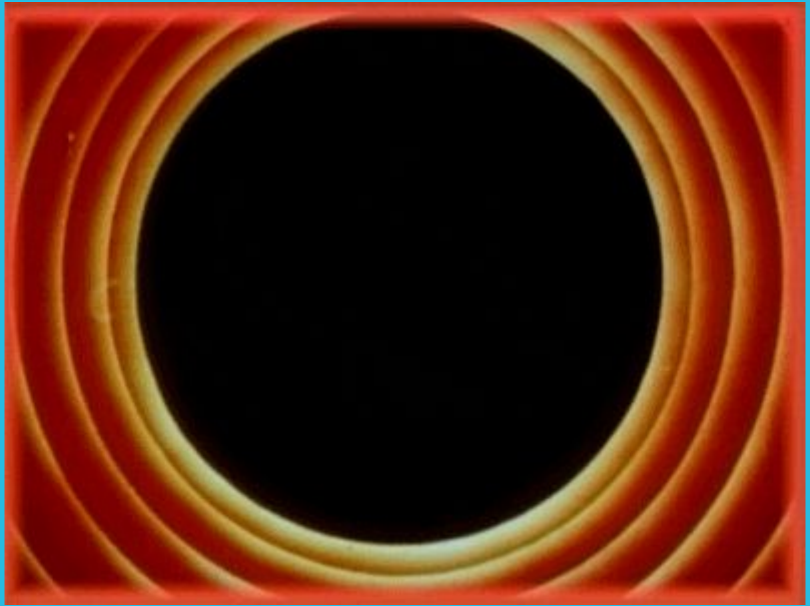
Models Ensemble Scores

	Stage 2			Stage 3
	Optimized Weighted Average	XGBoost	LightGBM	Weighted Average
Public LB	0.2073	0.2074	0.2076	0.2082
Final LB	-	-	-	0.2086



QUESTIONS?

THANKS FOR ATTENDING!



www.linkedin.com/in/giba1
https://twitter.com/giba1

Blog: https://medium.com/nvidia-merlin/building-a-diverse-models-ensemble-for-fashion-session-based-recommendation-for-recsys2022-2419d2182c4c
Paper: https://github.com/NVIDIA-Merlin/competitions/blob/main/RecSys2022_Challenge