Multi-Objective Recommender System

OTTO 2022 – WSM Project 3

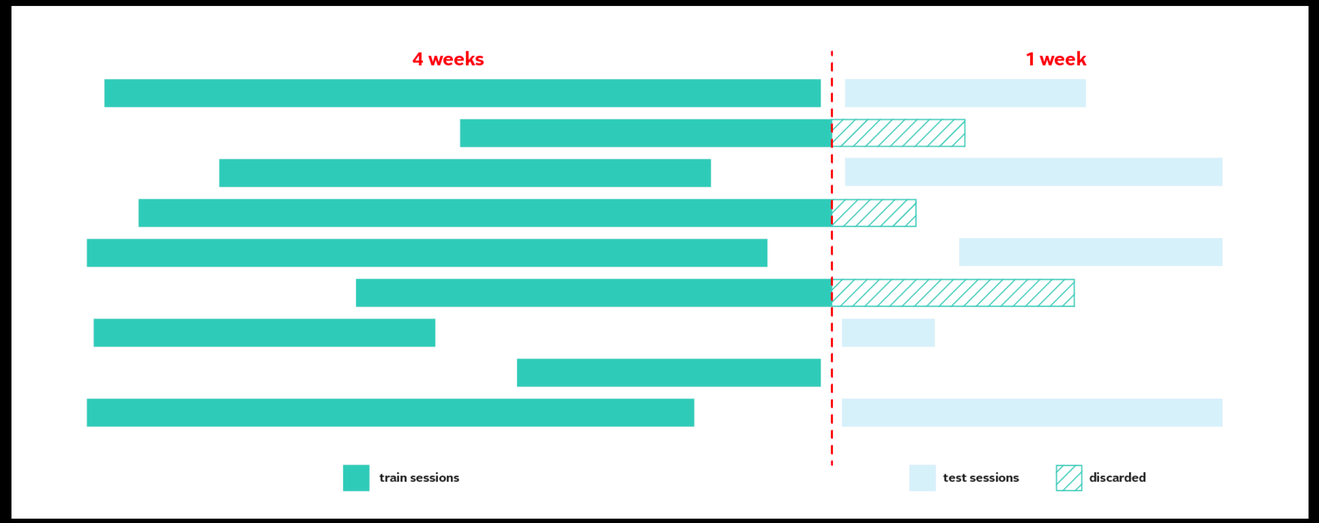
WSM_UTF8 | 2023/01/08

111753229 何子安｜111753152 王良文｜111753162 謝非諭｜111753213 江昀紘

Goal

Multi-Objective Recommender System

Previous
Events
in a
User Session



Clicks
Carts
Orders

https://www.kaggle.com/competitions/otto-recommender-system/overview

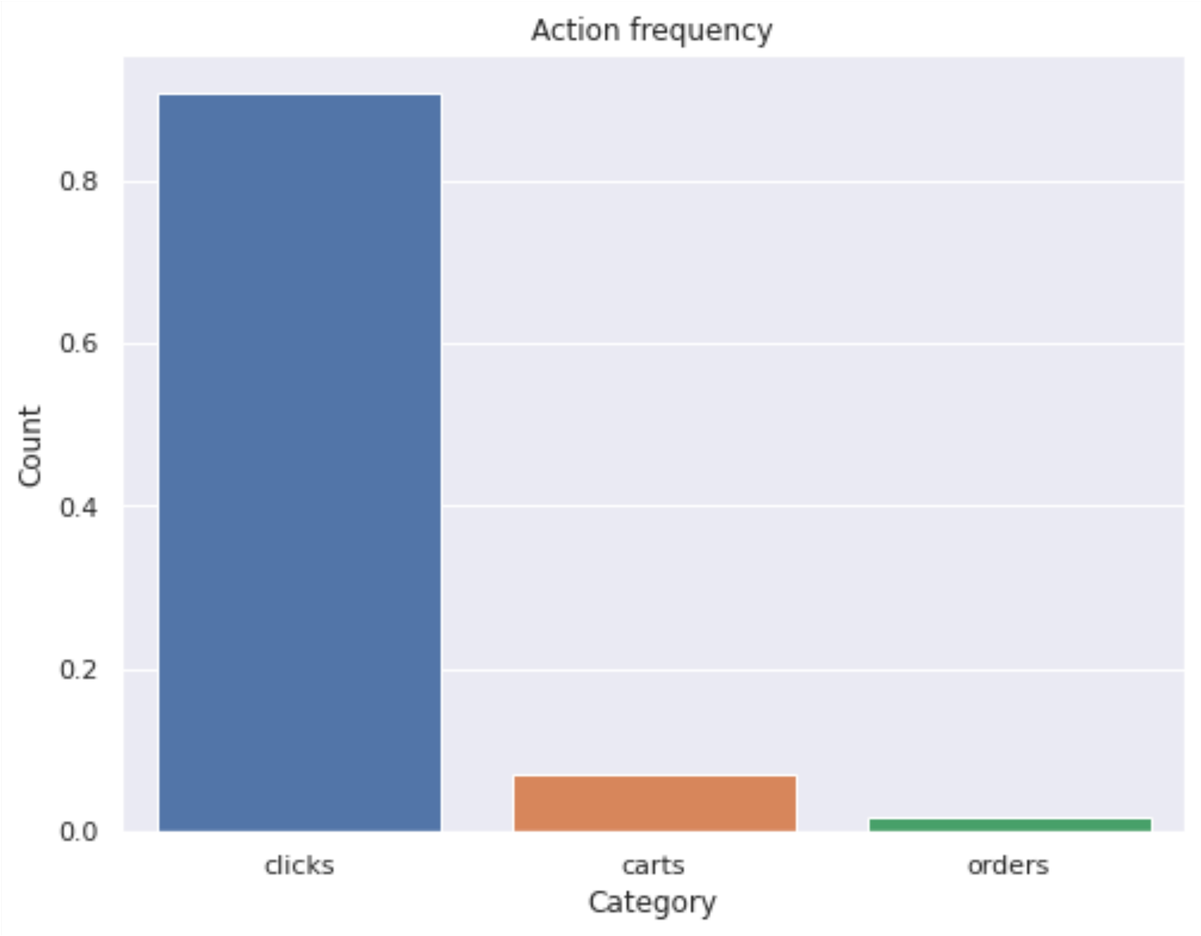
TOC

EDA & Data Visualization

Action (Behavior) frequency

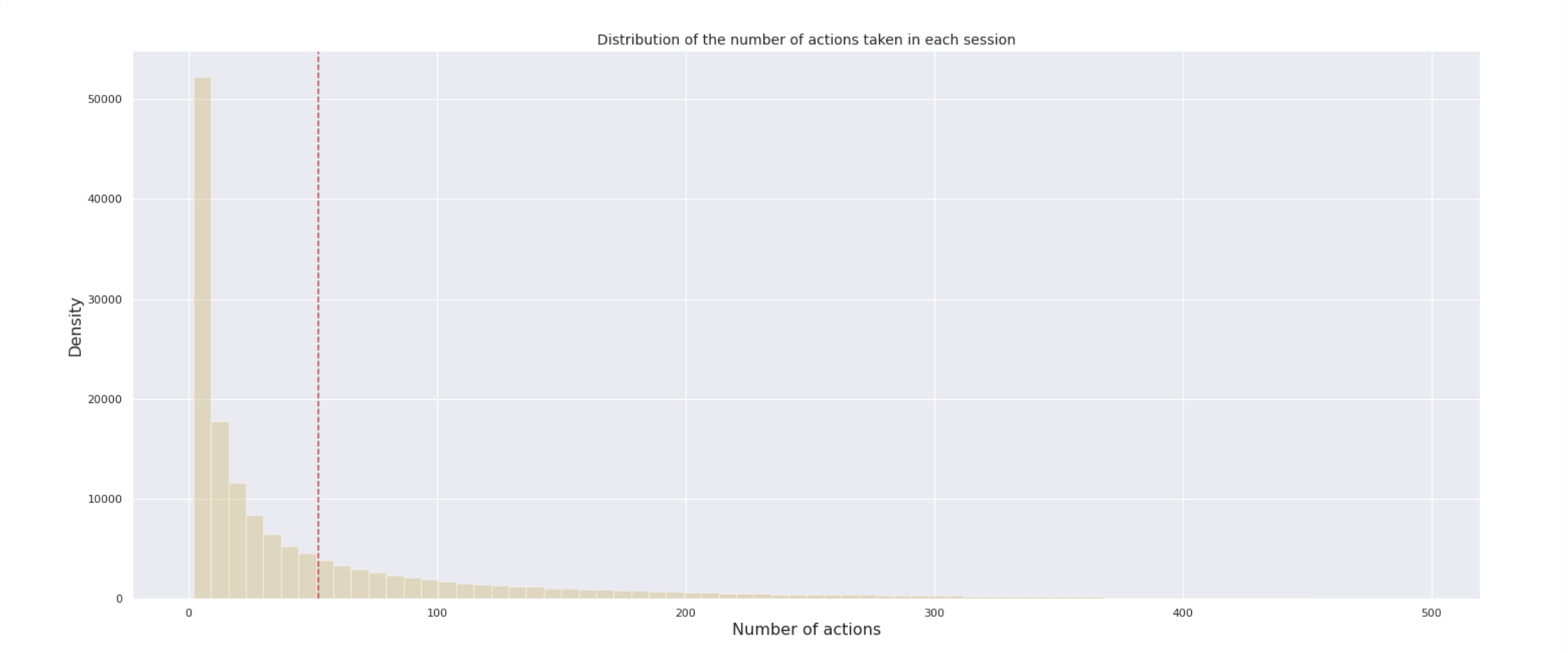
clicks, carts, orders

- behavior weight
- add bias (task's need)

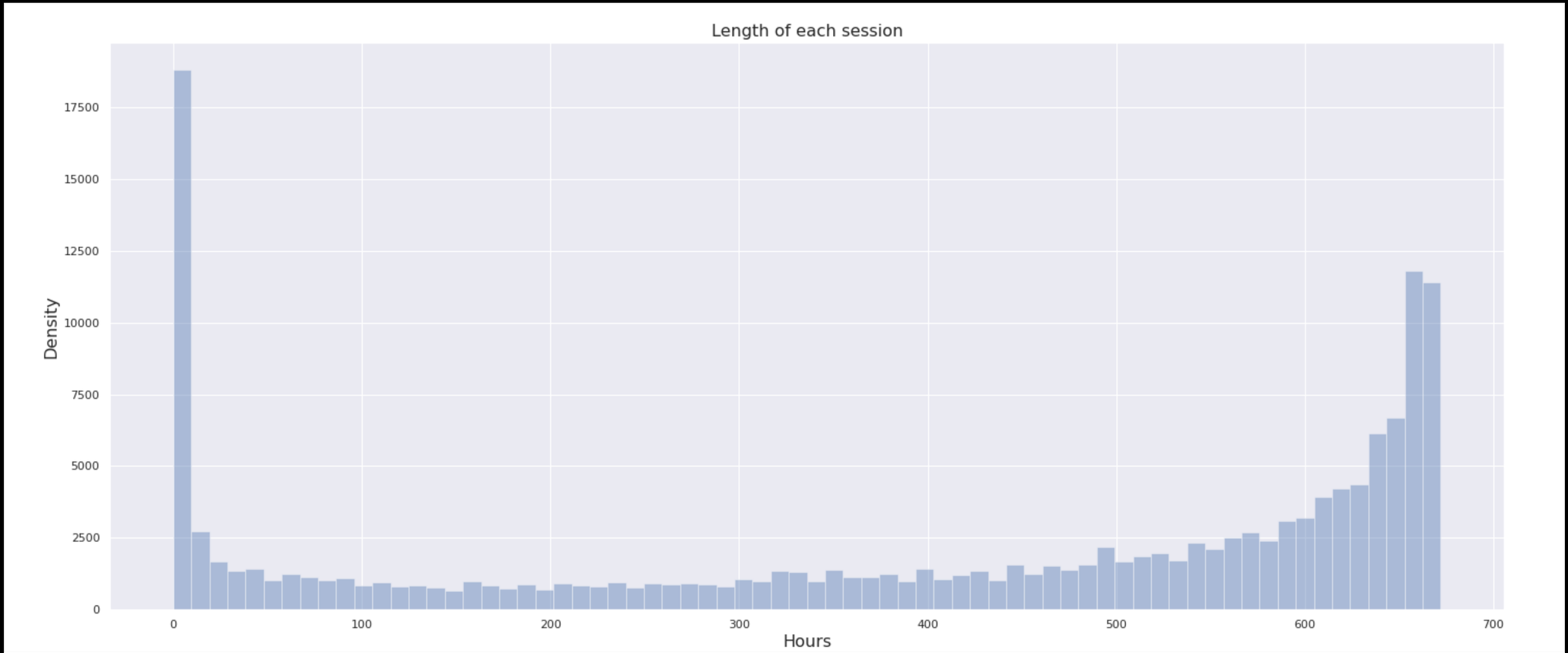


Distribution of the number of actions taken in each session

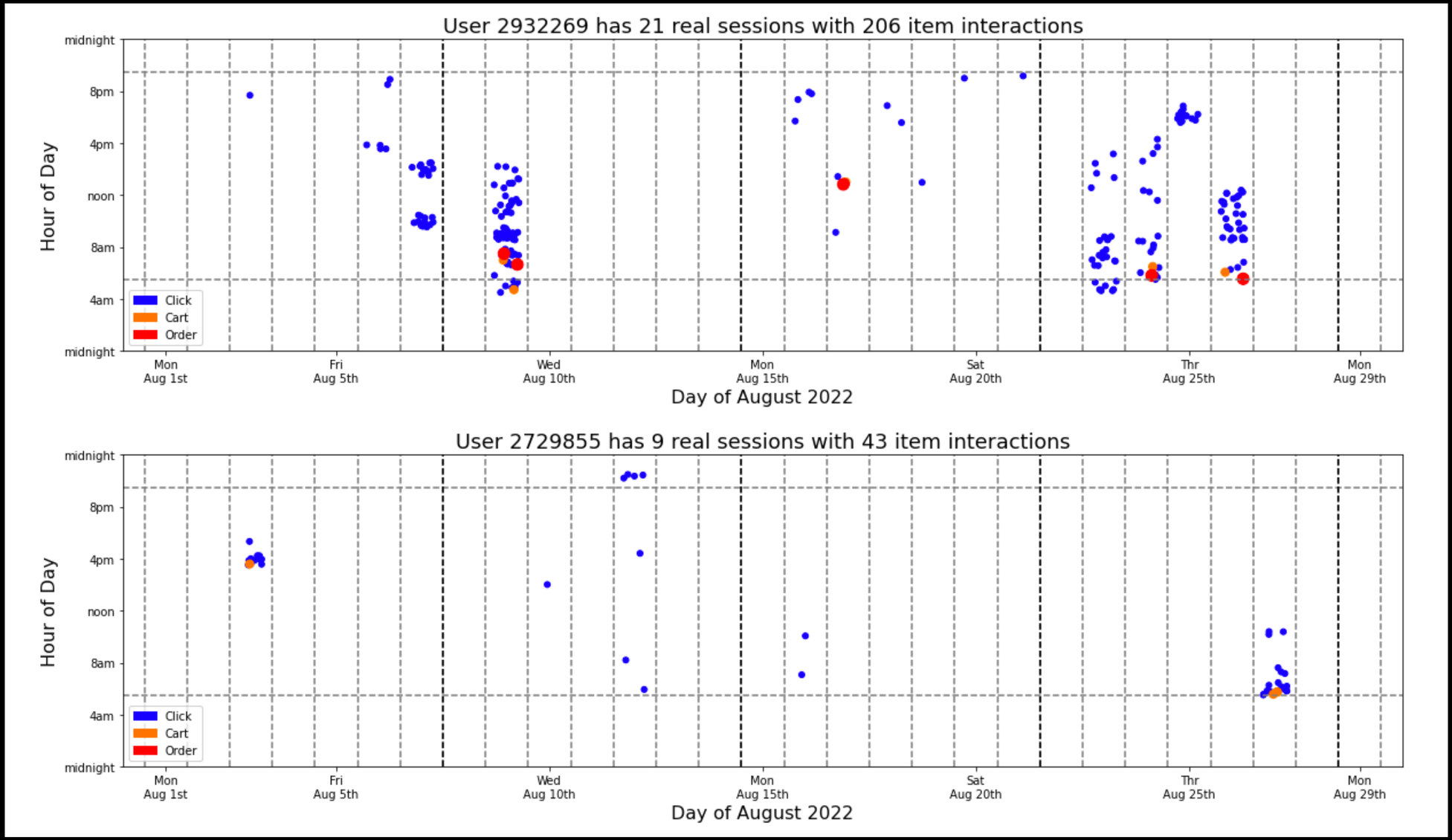
avg session long by actions, 80/20 rule



Length of each session by hours



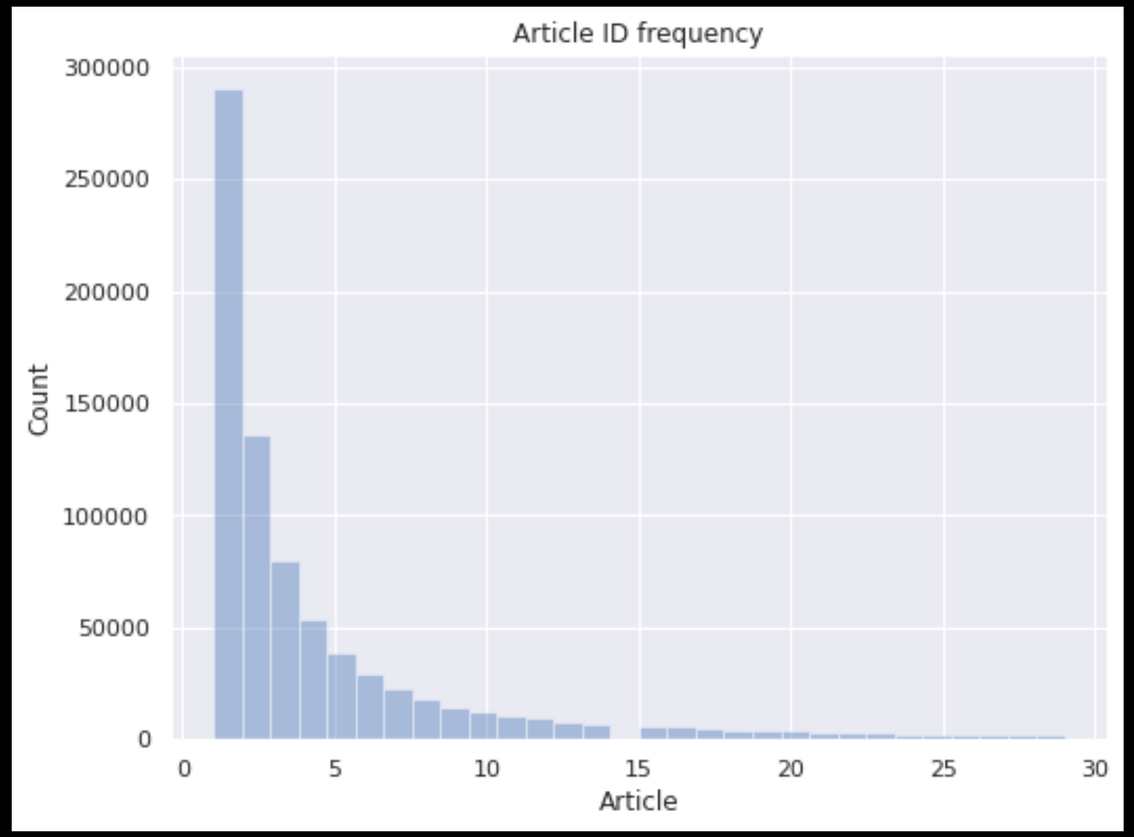
Real sessions and item interactions



Article ID frequency

aid

- # *aid* in total: 1,855,603
- 1,072,991 *aid's* frequency < 30



Models

Collaborative Filtering (CF) and Ensemble

111753152 王良文

Collaborative Filtering (CF)

Data Preprocessing

Step1: Use Compressed Sparse Row Matrices (CSR)

- *clicks_csr*, *carts_csr*, *orders_csr*

Step2: Create User Rating Matrix (URM)

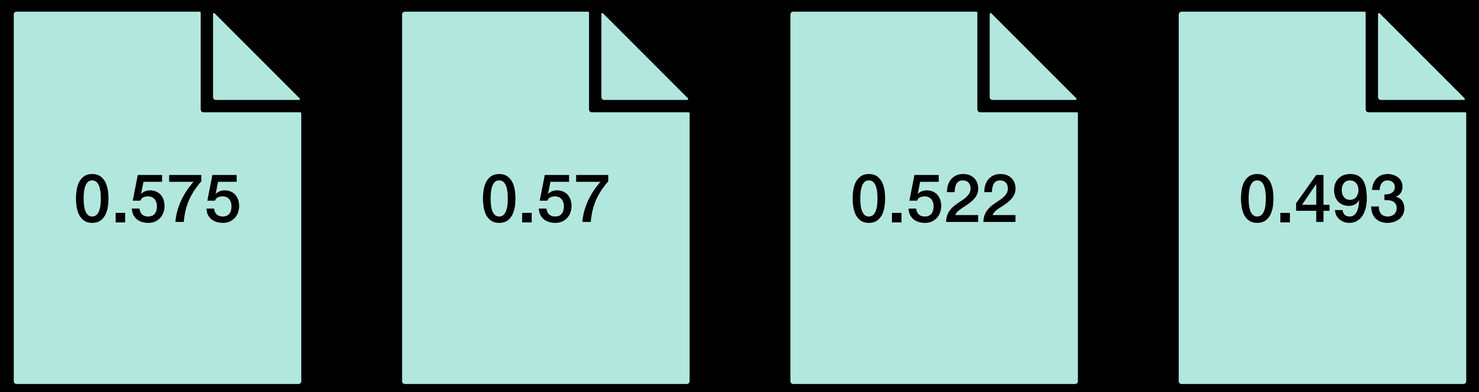
Step3: Use tf-idf to normalization URM

Collaborative Filtering (CF)

Model

Ensemble

Public Scores



Ensemble

Public Scores

weights	1	0.6	0.35	
scores	Final Scores 0.575	Final Scores 0.57	Final Scores 0.522	Final Scores 0.493

to vote and sort by vote sum: 0.554

Ensemble

Public Scores

weights	1	0.7	0.6	0.35
scores	Final Scores 0.575	Final Scores 0.57	Final Scores 0.522	Final Scores 0.493

to vote and sort by vote sum: 0.542

Word2Vec

111753162 謝非諭

Word2Vec

Data Preprocessing (with *polars*)

Step1: Group by aid

- *GroupBy.agg()*

Step2: Transform the data into list

- *to_list()*

Word2Vec

Model (with *gensim:CBOW*)

- In the CBOW model, the distributed representations of context are combined to predict the word in the middle

ItemA, ItemB, _____ , ItemD, ItemE

- Use annoy to find neighbor
  - (Euclidean)

Word2Vec

Hyperparameters

CBOW(*workers=8, window=9, vector_size=64, sg=0*)

weights	1	5	3
behavior	 Click	 Cart	 Order

best score: 0.519

Word2Vec

Difficult

Difficult: If find aids < 20

Solved: Find the most recent aid and look for its neighbors

ItemCF

111753213 江昀紘

ItemCF

Data Preprocessing

Step 1: Transform the data into Apache Parquet format

Step 2: Def *ItemSimilarityMatrix_fn*

Step 3: Normalization

ItemCF

Model

User-based CF

Item-based CF

top 100 similarity scores

best score: 0.517

ItemCF

Goods & Difficulties

Goods:

- Explainable and sensible

Difficulties:

- Need similar data
- Unpopular items are hard to recommend
- Require a big score sheet

Matrix Factorization

111753229 何子安

Matrix Factorization

Data Preprocessing

```
{
    "session": 42,
    "events": [
        { "aid": 0, "ts": 1661200010000, "type": "clicks" },
        { "aid": 1, "ts": 1661200020000, "type": "clicks" },
        { "aid": 2, "ts": 1661200030000, "type": "clicks" },
        { "aid": 2, "ts": 1661200040000, "type": "carts"  },
        { "aid": 3, "ts": 1661200050000, "type": "clicks" },
        { "aid": 3, "ts": 1661200060000, "type": "carts"  },
        { "aid": 4, "ts": 1661200070000, "type": "clicks" },
        { "aid": 2, "ts": 1661200080000, "type": "orders" },
        { "aid": 3, "ts": 1661200080000, "type": "orders" }
    ]
}
```

session	aid	ts	type
0	1517085	1659304800	0
0	1563459	1659304904	0
0	1309446	1659367439	0
0	16246	1659367719	0
0	1781822	1659367871	0
...	...	...	...
12899776	1737908	1661723987	0
12899777	384045	1661723976	0
12899777	384045	1661723986	0
12899778	561560	1661723983	0
12899778	32070	1661723994	0

Matrix Factorization

Data Preprocessing

session	aid
0	1517085
0	1563459
0	1309446
0	16246
0	1781822
...	...
12899777	384045
12899777	384045
12899778	561560
12899778	32070

session	aid
0	1517085
	1563459
	1309446
	16246
	1781822
...	...
12899777	384045
	384045
12899778	561560
	32070

aid	aid_next
1517085	1563459
1563459	1309446
1309446	16246
16246	1781822
1781822	
...	
384045	384045
384045	561560
561560	32070
32070	

drop

Matrix Factorization

Model

per batch

aid	aid_next
1517085	1563459
1563459	1309446
1309446	16246
16246	1781822
...	
384045	384045
384045	561560
561560	32070

target embedding

n_factors k

cardinality_aids

Nearest Neighbors

euclidean

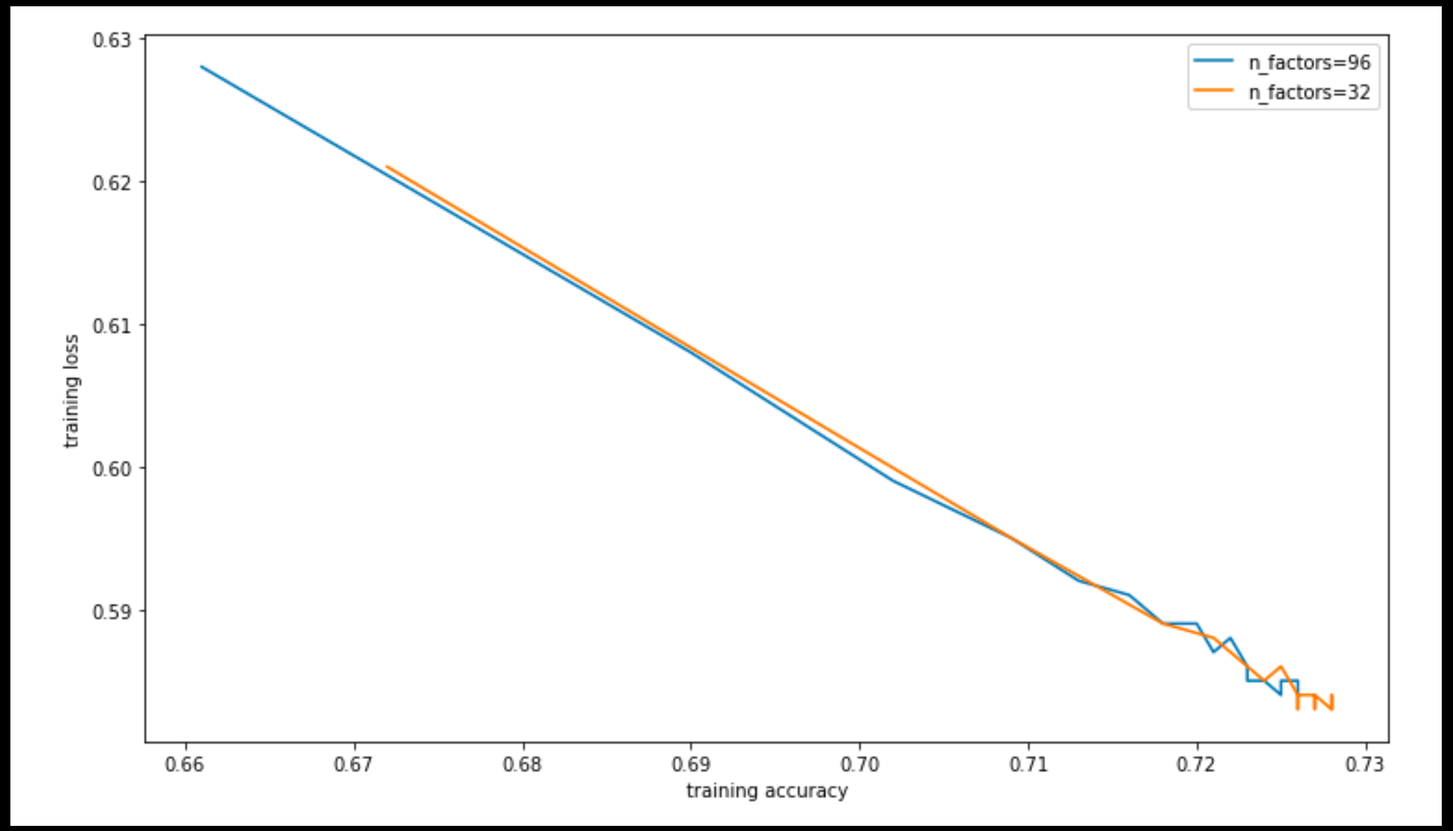
Matrix Factorization

Hyperparameters

batch size	*epochs*	*n factors*	*optimizer*
loss function	*metric*	*behavior's weight*	

Matrix Factorization

Model's Performance



best score: 0.499

Matrix Factorization

Model’s Improvements

- This model used *aid* straight, which makes no sense
- Replace *batch size* with average session long
- Add bias / intercepts, ex. popular items, rare items
- Similarity -> Distance
  - Metric Factorization: Recommendation beyond Matrix Factorization (2018)

Results

Model	Try	Ensemble 1	Ensemble 2
Ensemble	0.575	0.542	0.554
	0.57		
	0.522		
	0.493		
Word2Vec	0.519		
Item CF	0.517		
Matrix Factorization	0.499		

Difficulties & Learned

- Data preprocessing matter, which is hard
- To make better use of the TOP model, require model’s fundamental as well
- Ensemble and LearningToRank usually bring performance to next level
- Limitations of hardware
    - There is always bigger
    - *(TFRecord)*
- Stand on the shoulder of the giant

Thank you for your attention.