

Winning Amazon KDD Cup'23

Recommender Two Stage Model

Task 1 & 2 Stage One

Task 1 & 2 Stage Two

Task 3

Amazon KDD Cup '23
Shopping Session Dataset
Build Multilingual Recommendation Systems

💰 \$21,000 Cash + \$10,500 AWS Prize Pool
ACM SIGKDD 2023 Workshop
Credit Point

by Amazon Search

Participants

Rank	Team	Score	Submission Trend
01	NVIDIA-Merlin 3.0	3,000	
02	MOTV-REC 9.0	9,000	2
03	Canary 19.0	19,000	8

1st Place Team NVIDIA-Merlin
4x KAGGLE GRANDMASTERS OF NVIDIA (KGMON)
1x Nvidia Merlin Recommender Systems

Benedikt Schifferer, Chris Deotte, Kazuki Onodera, Gilberto Titericz, Jean-Francois Puget

Three Tasks, Three Competitions!

Task 1: Next Product Recommendation
Amazon KDD Cup 2023
@ 318k # 5763

Task 2: Next Product Recommendation for Underrepresented Languages
Amazon KDD Cup 2023
@ 112k # 2872

Task 3: Next Product Title Generation
Amazon KDD Cup 2023
@ 9626 # 1961

→ **Task 1** - predicting the next engaged product for sessions from English, German, and Japanese

→ **Task 2** - predicting the next engaged product for sessions from French, Italian, and Spanish, where transfer learning techniques are encouraged

→ **Task 3** - predicting the title for the next engaged product

Session-based recommendation
Question - What Will Users do in Future?

time

item 1001, item 5391, item 4432, item 6788, item 7321, item 6543, item 1234, ???

(one example test user session)

- Above is 1 example user session
- Train data from past. Test data in the future.
- Train data has 3.6 million unique user sessions!
- Train data has 1.4 million unique items (i.e. products)!
- Each item has 11 properties including Title, Brand, Color, Price, etc
- Task 1 & 2 Metric is MRR@100
- Task 3 Metric is BLEU

Two Stage Recommender Pipeline

Items (millions) → Stage 1 Candidate generation → Candidates (hundreds) → Stage 2 Ranking → Recs (hundred)

Stage 1 → **Stage 2**

Co-Visitation, BERT Text Embedding Sim, Text Similarity Scores, SWIFT/Swing Similarity, Neural Network: CNN Scores, Neural Network: GRU Scores, Neural Network: Transformers

Ensemble Candidates¹⁾ → Feature Engineering²⁾ → Train Reranker → Ensemble

Session Features, Candidate Features, Co-Visitation Item Features, Session-Item Features

1) Final candidates can use a subset of all potential candidates from stage 1 depending on the solution
2) Scores from candidate generation can be used as session-item features

Stage 1 - BERT Embeddings & Text Similarity Scores

Item Title, Item Embedding, Session, KNN search, 100 candidates

- Concatenate product, title, price, locale and all other properties.
- Use a pretrained LLM to extract vector embeddings from the last layer.
- Use Nearest Neighbors search to find top 100 most similar items of last item in a session.

Stage 1 - Finetune User Session and Item embeddings with CNN

Transfer Learning: Embedding table is initialized with a pretrained model. This pattern is shifted to the left. Predict last prev item from previous ones. 4000 random samples are used as negatives for each predicted embedding.

Maximize sim, Minimize sim, emb_s, emb_3, emb_2, emb_1, emb_0, e_n_0, item_3, item_2, item_1, item_0, neg, Prev items, Next item, random item

Stage 1 - Finetune User Session and Item embeddings with NVIDIA Merlin Transformers4Rec¹⁾

Transformers Block, MLP Block, User Session Representation (Embedding), Dot Product, Loss, Linear Projection, Embedding, Inputs at step 1, Order, Inputs Candidate Tower

1) <https://github.com/NVIDIA-Merlin/Transformers4Rec>

- Merlin Transformers4Rec provide an easy-to-use integration for Hugging Face for recommender systems with ~64 different architectures
- The XLNet architecture was trained with MLM masking strategy
- The input were item_id, item_features (e.g. Brand, color) and BERT text embeddings processed by a 1-layer MLP without activation function
- One variant was multi-task learning, where the model had to predict the masked item and the masked item features (e.g. brand)
- Different combination of two variants were trained (total 4)
 - Previous Item Only vs.
 - Previous Item + Next Item
 - Single Task vs. Multi-Task

Stage 1 - LLM Transfer Learning Scores

	UK	DE	JP	FR	IT	ES
Allenai-specter	0.196	0.196	0.149	0.258	0.241	0.259
Distiluse-base-multilingual-cased-v2	0.197	0.196	0.212	0.251	0.236	0.254
Bert-base-multilingual-uncased	0.195	0.195	0.214	0.255	0.239	0.256
Clip-ViT-B-32-multilingual-v1	0.197	0.195	0.212	0.252	0.236	0.254
Stb-xlm-r-multilingual	0.194	0.192	0.212	0.247	0.232	0.249

Combining the 5 LLM embeddings scores (MRR@100):

- Task1 trainset score: 0.209 / LB: 0.226
- Task2 trainset score: 0.261 / LB: 0.278

Stage 1 - Covisitation Matrices

Use all train user sessions

1	5	3	12	9		6	8	3
4	1	3	8			5	1	3
5	7	9	3	15	4	8	11	14

Create item Top-N lists (by counting how often items appear together)

1	2	3	4	5	6	7	8
5	12	7	15	8	11	6	3
3	8	6	3	2	8	1	8
9	7	5	5	1	3	2	2

Stage 1 - Covisitation Matrices

Generate user session candidates by Applying Top-N lists to each user session item

User Session: 1, 8, 3, 7, 12, 4

5	3	5	4	3	1
3	1	2	5	5	2
9	2	7	6	8	9

Candidate Items

Item	Count
5	4
3	3
2	3
1	2
9	2
7	1
4	1
6	1
8	1

Stage 2 - GBT Reranker Example Feature Engineering

User Session, Item, Feature 1 Probability item is correct brand for user, Feature 2 Probability item is correct color for user, Feature 3 Probability item is correct size for user, Feature 4 Probability item is correct model for user, Target

Stage 2 - GRU to Predict Item Properties Brand, Color, Size, Model per User Session

User Session, Brand Embedding 768, Color Embedding 768, Size Embedding 768, Model Embedding 768, GRU hidden size 768, Brand Dense Layer 768, Color Dense Layer 768, Size Dense Layer 768, Model Dense Layer 768, Brand Softmax, Color Softmax, Size Softmax, Model Softmax, Predict Brand probability, Predict Color probability, Predict Size probability, Predict Model probability

Stage 2 - Train GBT Reranker using 8xV100 GPU!

8xV100 DASK XGBoost 256 GB VRAM, GPU V100 32 GB VRAM, GPU V100 32 GB VRAM, GPU V100 32 GB VRAM, GPU V100 32 GB VRAM, GPU V100 32 GB VRAM, GPU V100 32 GB VRAM, GPU V100 32 GB VRAM

```

import xgboost as xgb

X_train = candidates[FEATURES]
y_train = candidates[TARGET]
dtrain = xgb.dask.DaskQuantileMatrix(
    client, X_train, y_train
)

model = xgb.dask.train(
    client, XGB_PARAMS, dtrain=dtrain
)

```

Train 360 million rows with ~150 features using 8xV100 DASK XGBoost for +0.002 MRR boost !!

Task 3 Baseline
Submitting User's Last History Item Title
Achieves LB 0.26553 - 22nd Place!

Second to Last History Item Title → Last History Item Title (Prediction) → Leaderboard LB = 0.26553

$BLEU\ score = \sqrt[4]{P_{1-gram}(y, \hat{y}) * P_{2-gram}(y, \hat{y}) * P_{3-gram}(y, \hat{y}) * P_{4-gram}(y, \hat{y}) * BP(y, \hat{y})}$

* Geometric mean of 1-gram, 2-gram, 3-gram and 4-gram precision scores.
BP = Brevity Penalty

Task 3 - Model One
Last Two Titles Classifier
Achieves LB 0.26673 (+0.00120) - 13th Place!

Last History Item Title, Second to Last History Item Title, Titles Candidates Top 1 to 10, BLEU score, # of Words, # intersection of words, # of product properties in history titles, Concatenate, Random Forest Classifier

Task 3 - Model Two
Removing Last Word Classifier using LLM
Achieves LB 0.27152 (+0.00480) - 1st Place!

Train XLM-RoBERTa-base to choose:

```

model = TFAutoModel.from_pretrained('xlm-roberta-base')
outputs = model(ids, attention_mask=att)[0]
outputs = tf.keras.layers.GlobalAveragePooling1D()(outputs)
outputs = tf.keras.layers.Dense(1, activation='sigmoid')(outputs)

```

ZOEON Clothing for Baby Doll, Outfits with Hat for 14-16 inches Dolls (35-43 cm) (Pink) ???

Predict Binary Classification if removing last word improves BLEU metric