Yandex Education

Университет Сириус

Научно-практический интенсив по воспроизведению State-of-the-Art научных результатов

Поиск подходящих кандидатов с применением на GPU

Статья Revisiting Neural Retrieval on Accelerators

Сириус 2024 г.

Наша команда

Соня-Аня Никифорова

Маша Розаева

Данил Мироманов

Илья Мурзин

Ментор: Костя Гордеев

План доклада

1. Что такое рекомендательная система? Этапы отбора рекомендаций
2. Обзор научной статьи
3. Постановка задачи в рамках интенсива: воспроизведение предлагаемых в статье алгоритмов (h-indexer, MoL)
4. Реализация и эксперименты
5. Наши результаты
6. Дальнейшие исследования
7. Список использованных источников

Рекомендательная система – это?

Пользователи (users, U) и товары (items, I). I >> U, много “холодных”

Хотим рекомендовать каждому пользователю наиболее релевантные товары, т.е. которые он посмотрит/положительно оценит/купит

Retrieval + Ranking

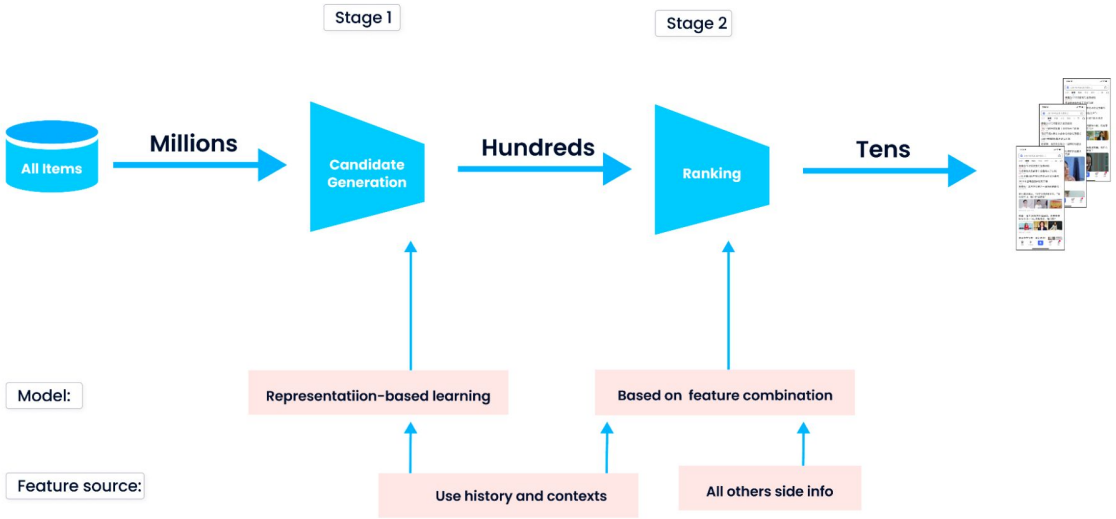
Оцениваем :

- HR@k – в топ-k попал хотя бы один релевантный товар
- Recall@k – доля релевантных товаров среди рекомендуемых
- MRR – обратное число к номеру первой позиции с релевантным товаром
- Еще много разных метрик

Этапы отбора рекомендаций

Нужно отбирать товары быстро, но качественно

Идея: легковесный средненький алгоритм + тяжеловесный хороший алгоритм



Revisiting Neural Retrieval on Accelerators (2023)

Jiaqi Zhai
jiaqiz@meta.com
Meta Platforms, Inc.
Menlo Park, CA, USA

Zhaojie Gong
zhaojieg@meta.com
Meta Platforms, Inc.
Menlo Park, CA, USA

Yueming Wang
yuemingw@meta.com
Meta Platforms, Inc.
Menlo Park, CA, USA

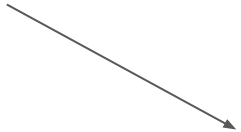
Xiao Sun
sunx@meta.com
Meta Platforms, Inc.
Menlo Park, CA, USA

Zheng Yan
zyan@meta.com
Meta Platforms, Inc.
Menlo Park, CA, USA

Fu Li
leaf123@meta.com
Meta Platforms, Inc.
Menlo Park, CA, USA

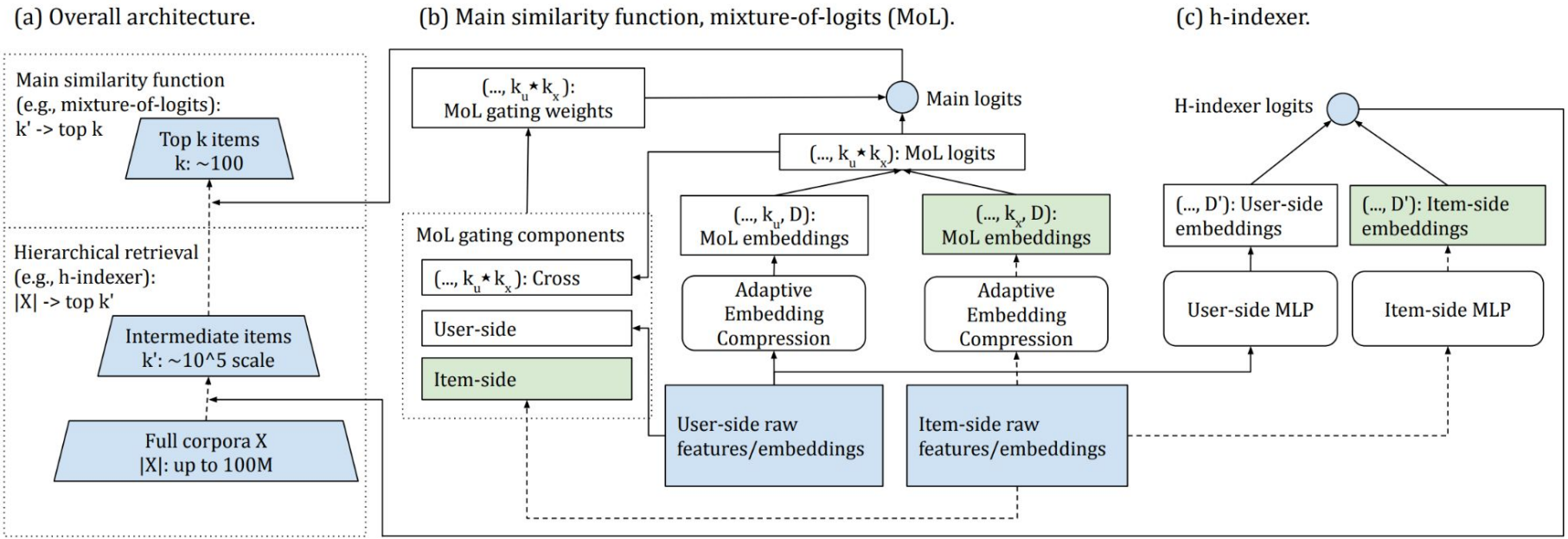
Xing Liu
xingl@meta.com
Meta Platforms, Inc.
Menlo Park, CA, USA

Dot Product – хорошо, но не супер качественно. Вот бы был Dot Product, но чтобы умный

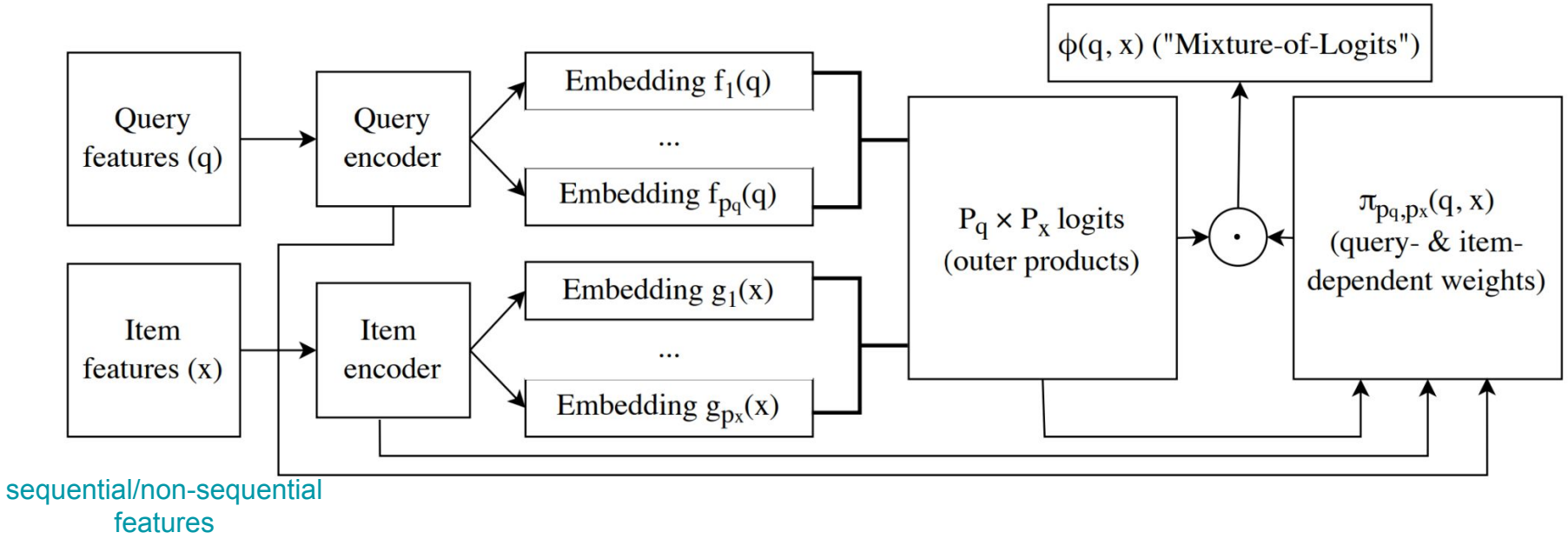


придумали MoL, h-indexer и то, как их оптимизировать

Retrieval с «умным» Dot-product



Алгоритм MoL (Mixture of Logits)



Bailu Ding, Jiaqi Zhai - Efficient Retrieval with Learned Similarities (2024)

Постановка задачи в рамках интенсива

1. Создать пайплайн для retrieval-части рекомендательной системы, включая обработку входных данных и расчет метрик
2. Имплементировать алгоритмы MoL и h-indexer, Sampled Softmax Loss, а также другие neural-based алгоритмы
3. Провести эксперименты и оценить эффективность алгоритмов
4. Изучить предлагаемые в статье способы оптимизаций

Данные

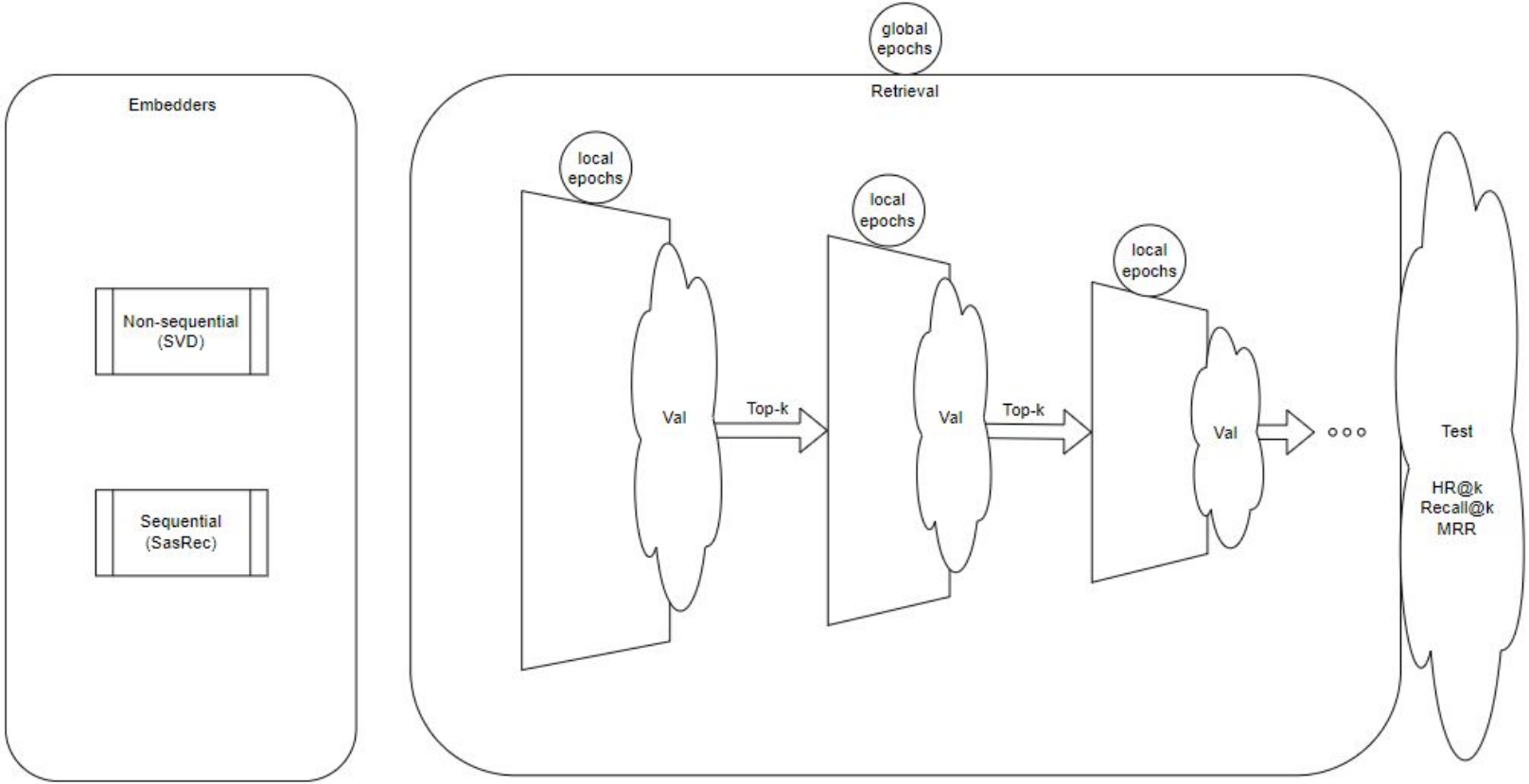
Train : validation : test — 0.7 : 0.2 : 01

Из validation и test убираем юзеров и айтемы, у которых меньше pcore=5 взаимодействий. Обрезаем последовательности айтемов до 200

127 негативных примеров на один позитивный

Dataset	# of users	# of items	avg. ratings per item	avg. ratings per user
ML-1M	6,040	3,416	193.45	109.41
ML-20M	138,493	18,231	707.18	93.09
Beauty	22,363	12,101	16.09	8.71
Games	24,303	10,672	20.79	9.13
Books	27,352	13,151	23.85	11.47

Пайплайн


Embedders
Non-sequential (SVD)
Sequential (SasRec)
global epochs
Retrieval
local epochs
Val
Top-k
local epochs
Val
Top-k
local epochs
Val
ooo
Test
HR@k
Recall@k
MRR

Эксперименты (ML-1M)

5 глобальных эпох, 1 локальная на модель

embedder+model	HR@1	HR@10	HR@50	HR@500	MRR ▾
sequential (SasRec), MLP + BCE	0,0644	0,3467	0,7044	0,9378	0,1588
sequential (SasRec), MoL + BCE	0,0644	0,3489	0,6533	0,9378	0,1521
non-sequential (SVD), MLP + BCE	0,0600	0,3244	0,6844	0,9356	0,1497
sequential (SasRec), MoL + SS	0,0600	0,3200	0,6178	0,9267	0,1454
sequential (SasRec), Dot product	0,0466	0,3355	0,6355	0,9311	0,1355
sequential (SasRec), FM + BCE	0,0578	0,3044	0,5667	0,9222	0,1334
sequential (SasRec), NeuralFM + BCE	0,0378	0,2956	0,6156	0,9200	0,1257
non-sequential (SVD), FM + BCE	0,0511	0,2200	0,4844	0,8644	0,1142
non-sequential (SVD), NeuralFM + BCE	0,0378	0,2667	0,6356	0,9133	0,1113
non-sequential (SVD), Dot product	0,0044	0,1933	0,5400	0,9088	0,0633

Как можно улучшить? Оптимизация применения модели

Квантизация (повышение пропускной способности на 16%)
Оптимизация скалярного произведения (ускорение x1.5)
Низкоуровневая оптимизация ядра (ускорение x2)

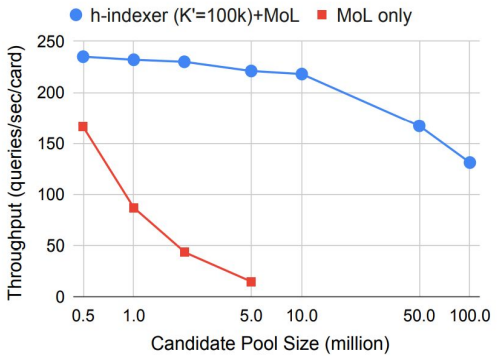


Figure 3: Recall and serving cost of the two-stage h-indexer/MoL model: (a) relative recall ratio vs MoL-only model under varying k' over 10M items, and (b) throughput of two-stage vs one-stage over different pool sizes.

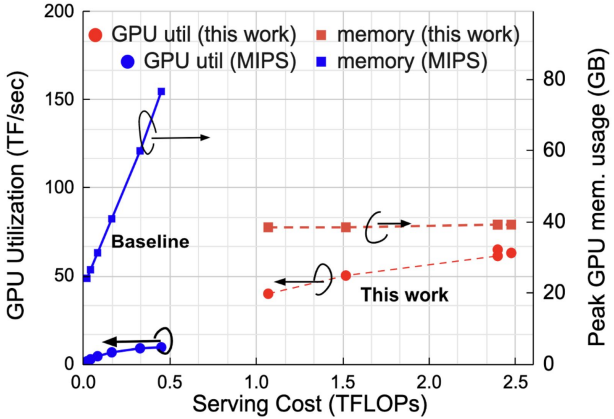


Figure 2: Infra efficiency in production: GPU utilization and peak memory scaling with serving FLOPs.

Ссылки на результаты

GitHub

DataLens (метрики)

Слайд с мемами (спасибо за внимание, можно задавать вопросы по презентации)