Accelerating Neural Recommendation Training with Embedding Scheduling

Chaoliang Zeng, Xudong Liao, Xiaodian Cheng, Han Tian, Xinchen Wan, Hao Wang, and **Kai Chen**

iSING Lab, Hong Kong University of Science and Technology

作者团队背景

Research Interests

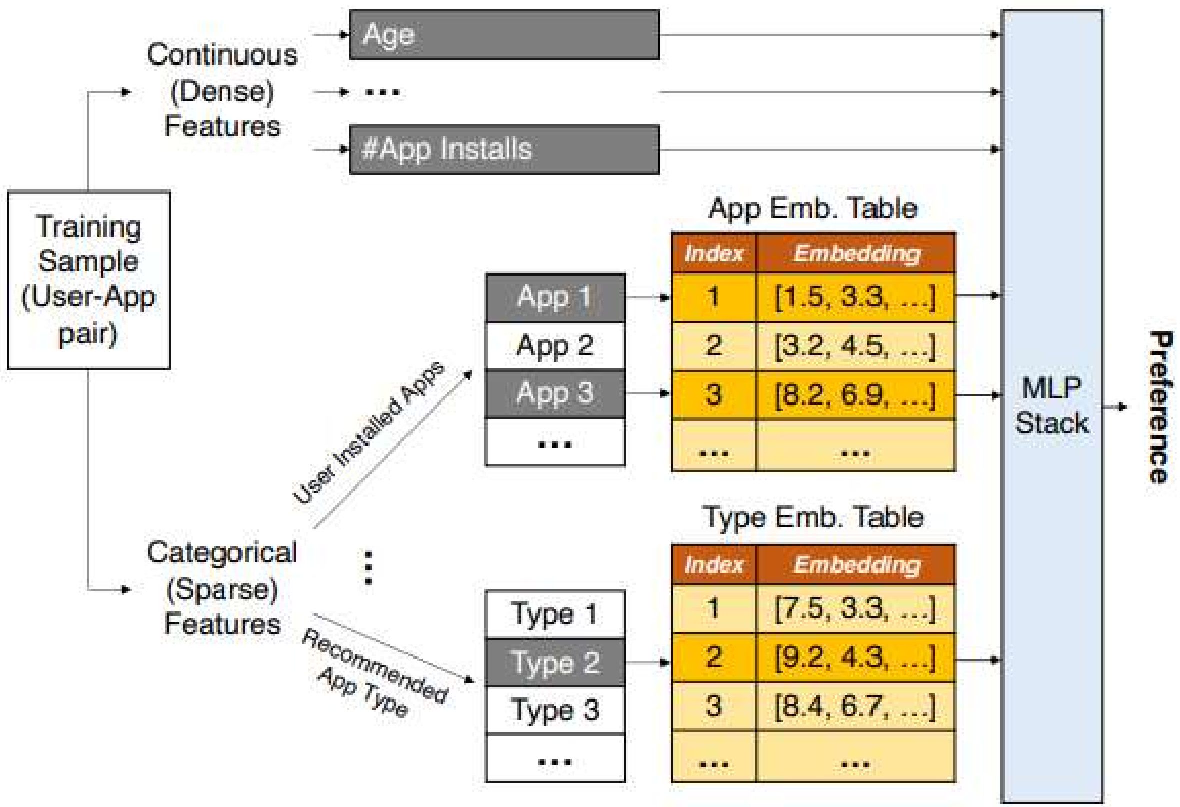
- Data Center Networks
- High-performance Networking
- AI-centric Networking
- Machine Learning Systems
- Hardware Acceleration
- Privacy-preserving Computing

Kai Chen's Homepage (hkust.edu.hk)

目录

基础知识

1. 什么是深度学习推荐模型
(Deep learning recommendation models, DLRM)

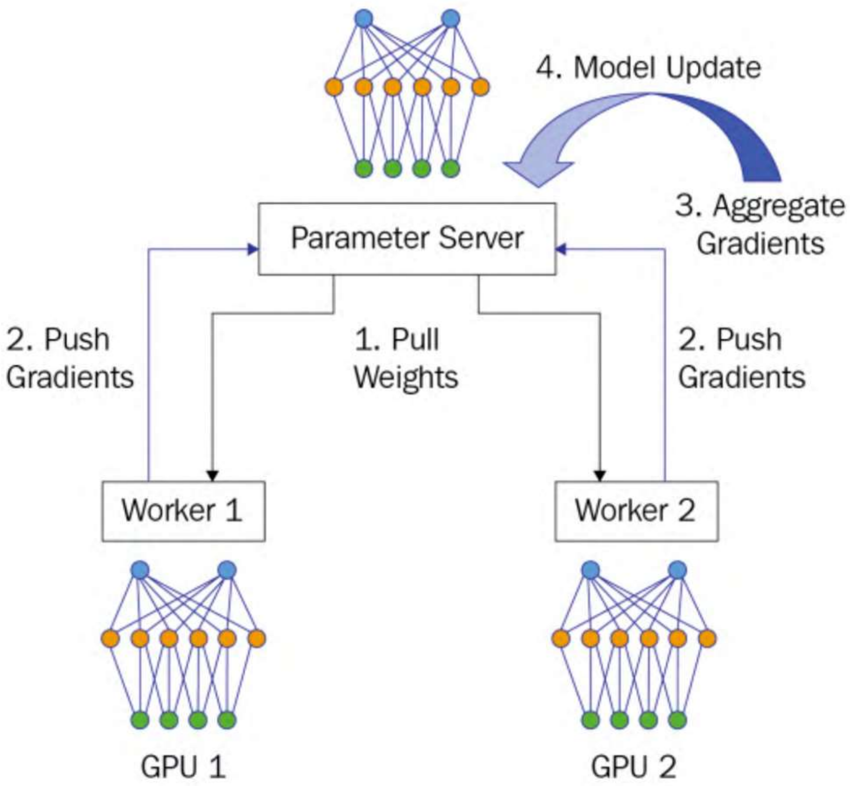


DLRM: 嵌入表+MLP

- 高性能
- 灵活性
- 可扩展性
- 开源

基础知识

2. 什么是参数服务器 (Parameter Server, PS)

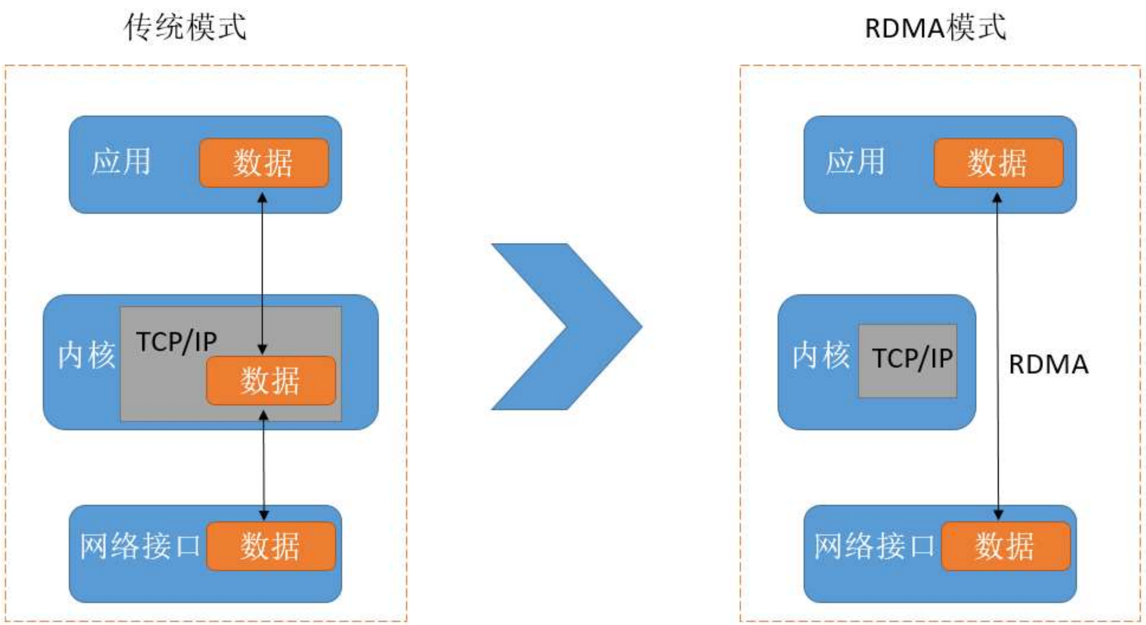


PS架构系统工作流程：

- Pull Weights
- Push Gradients
- Aggregate Gradients
- Model Update

基础知识

3. RDMA 和传统 TCP/IP 的比较

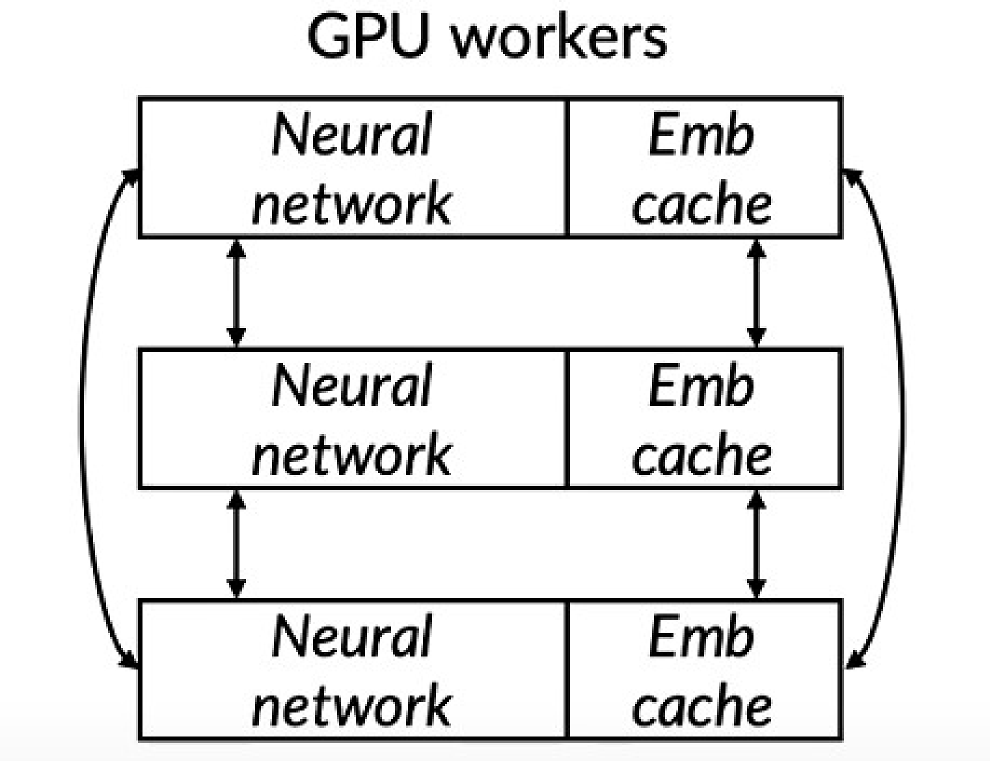


直接通过网络接口访问内存数据

高吞吐、低延迟的网络通信
（大规模并行计算机集群）

基础知识

4. 什么是FAE (Frequently Accessed Embeddings)

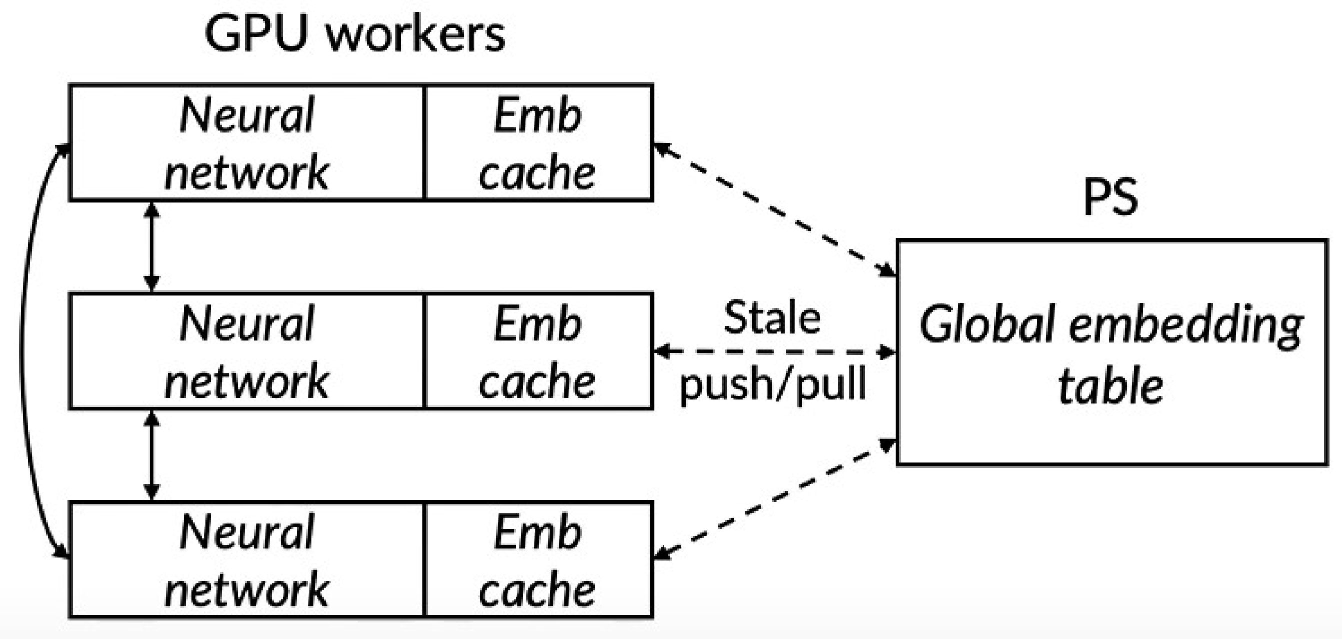


FAE oversample training data containing only hot embeddings

[Accelerating Recommendation System Training by Leveraging Popular Choices]()

基础知识

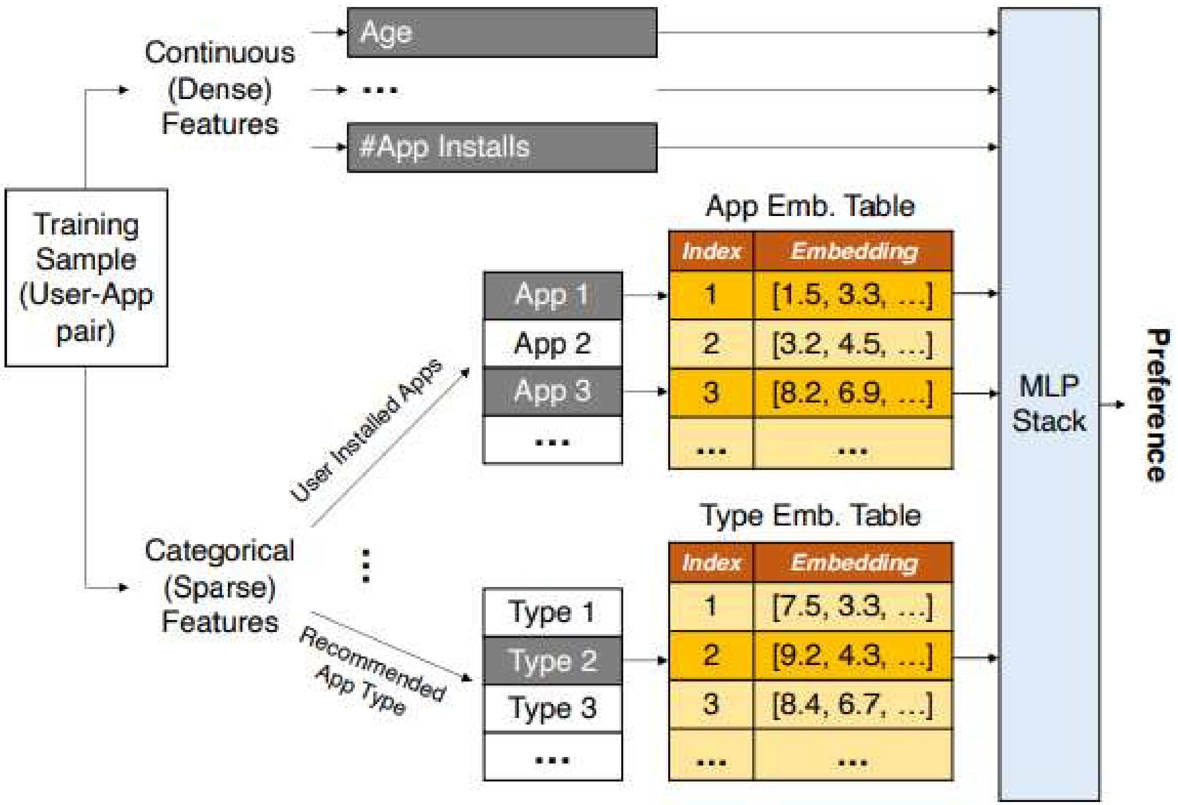
5. 什么是HET (Huge Embedding Model Training via Cache-enabled Distributed Framework)



HET applies a staleness-tolerant embedding update method

[HET: Scaling out Huge Embedding Model Training via Cache-enabled Distributed Framework]()

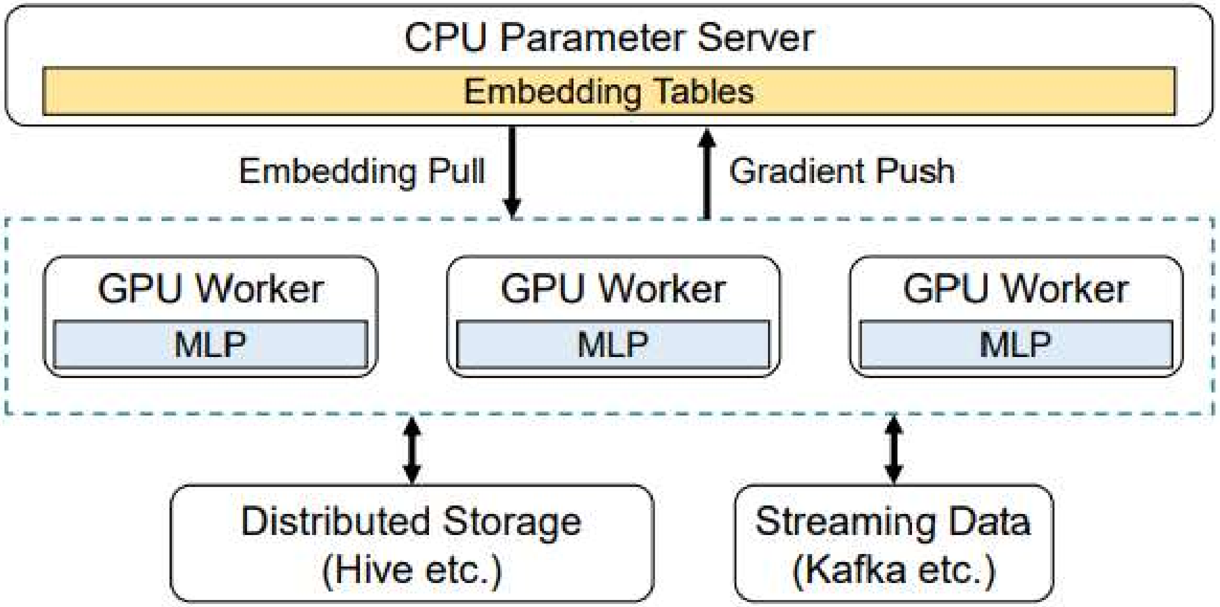
Background & Motivation



DLRM: 嵌入表+MLP

稀疏特征 -> 密集表示 [id]

Background & Motivation



HET架构Embedding cache存在

显著的 Pull / Push 通信开销

Background & Motivation

	Model	Dataset
W1	Wide & Deep [7]	Criteo AD [10]
W2	Neural Collaborative Filtering [19]	MovieLens 25M [17]
W3	DeepFM [16]	Avazu [23]
W4	Deep & Cross [43]	Criteo Sponsored Search [40]

Table 1: Workloads in our case studies.

Larger batch size

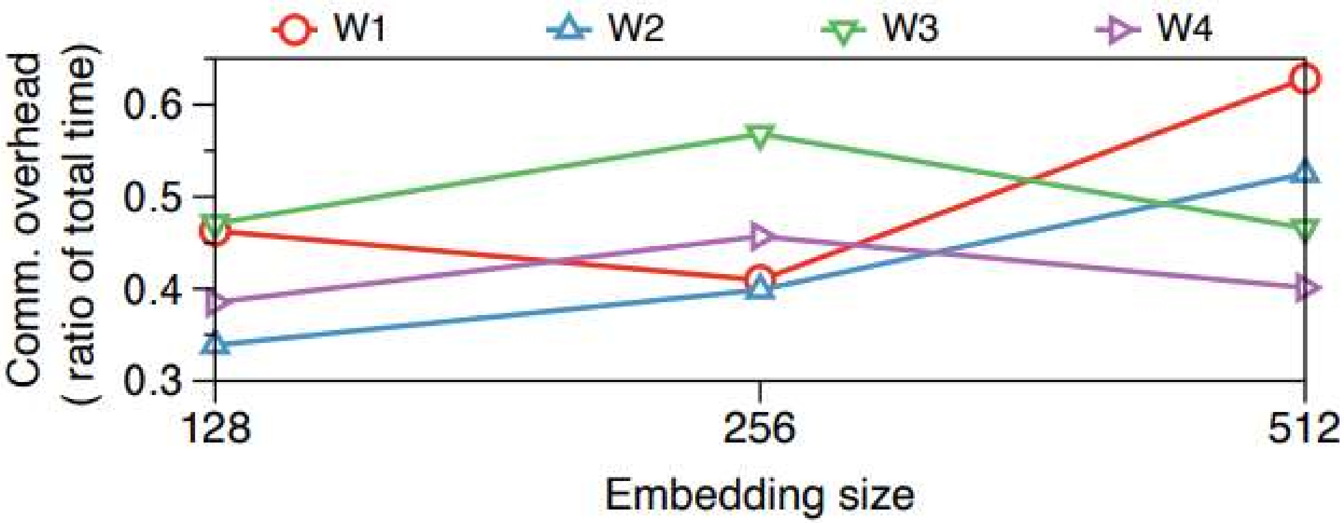
➔

More embedding transmissions

Background & Motivation

	Model	Dataset
W1	Wide & Deep [7]	Criteo AD [10]
W2	Neural Collaborative Filtering [19]	MovieLens 25M [17]
W3	DeepFM [16]	Avazu [23]
W4	Deep & Cross [43]	Criteo Sponsored Search [40]

Table 1: Workloads in our case studies.



Consumes up to 63% of end-to-end DLRM training time.

Related Work

Distributed recommendation systems

- Persia: An Open, Hybrid System Scaling Deep Learning-based Recommenders up to 100 Trillion Parameters 提出使用同步和异步机制分别更新MLP和嵌入表。然而，异步方案是不可拓展的，并且会随着worker数量提升而降低准确性。
- XDL: An Industrial Deep Learning Framework for High-dimensional Sparse Data 提出的优化包括分层样例压缩、工作流管道和零复制。它提供了对DLRM training pipeline的系统优化，并可以从嵌入调度中受益，进一步优化worker/PS之间的通信。

Related Work

Communication acceleration

● SparCML: High-Performance Sparse Communication for Machine Learning 等提出了许多优化稀疏参数同步的集体通信方法。

● Poseidon: An Efficient Communication Architecture for Distributed Deep Learning on GPU Clusters 等利用通信调度，它组织不同层的消息传输顺序，使通信与计算重叠。

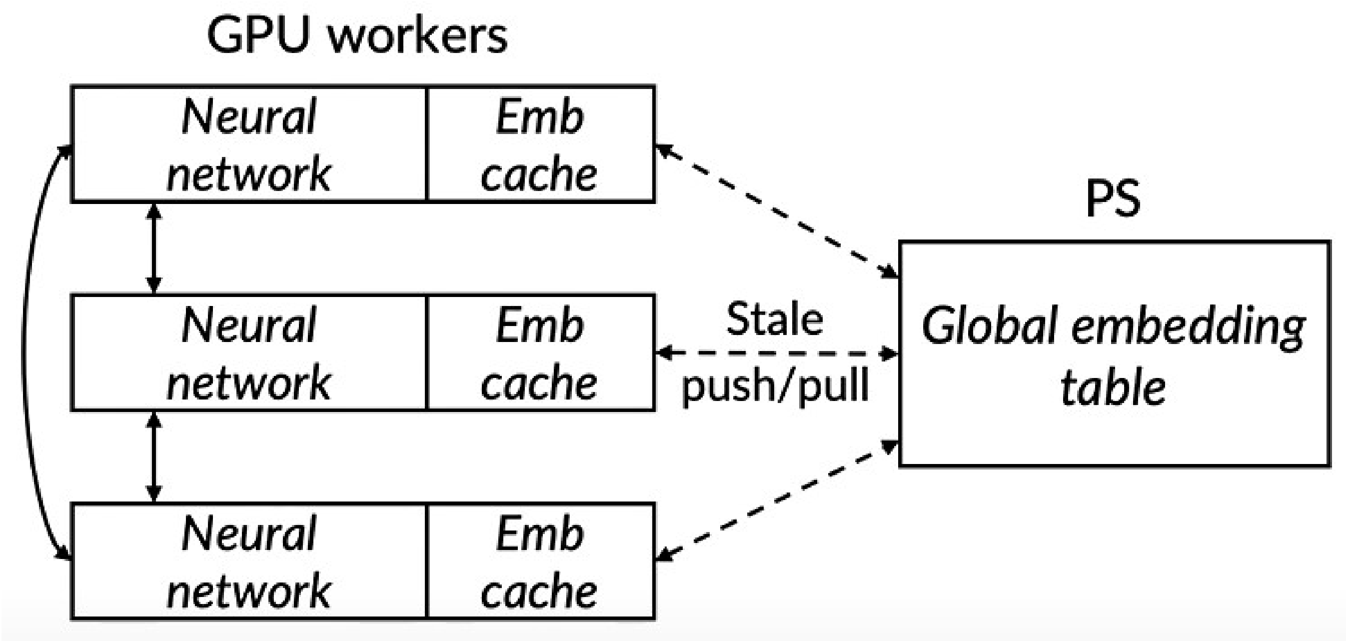
以上所有的通信加速方法都试图回答“如何有效地传递信息”。

Related Work

Serving large embedding tables

- Merlin HugeCTR: GPU-accelerated Recommender System Training and Inference 等直接跨多个GPU worker应用模型并行性，其中每个GPU在其高带宽内存(HBM)上存储一个表分片。
- Distributed Hierarchical GPU Parameter Server for Massive Scale Deep Learning Ads Systems 等利用数据集的偏度特征来加速高人气的嵌入访问。
- Training Personalized Recommendation Systems from (GPU) Scratch: Look Forward not Backwards 等侧重于通过调度嵌入IO和工作线程内的计算来进行缓存预取。

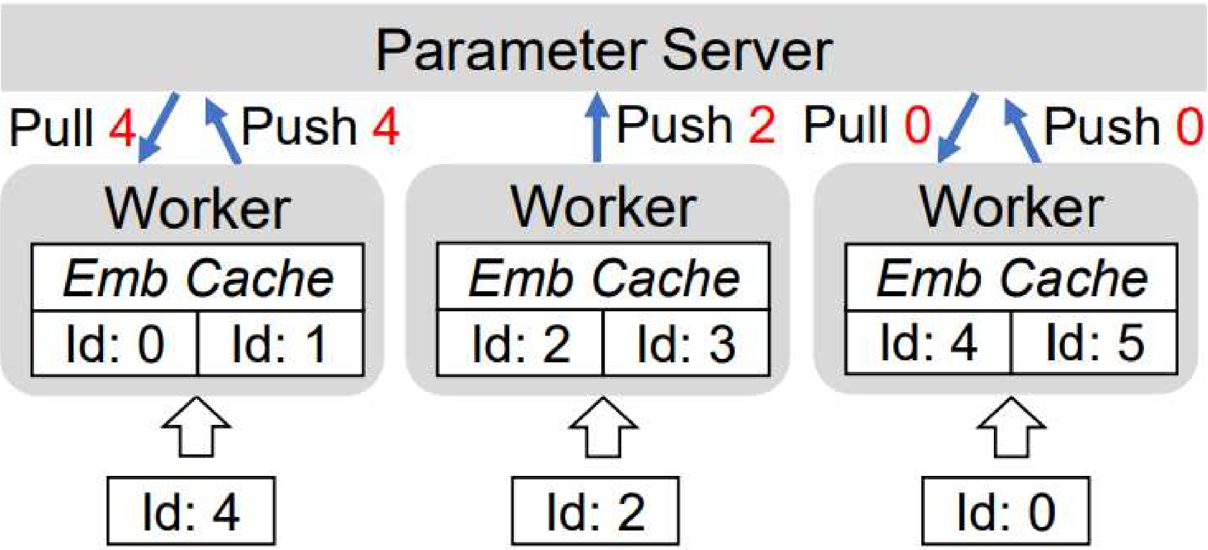
Design



- 应该在哪里训练嵌入
- 哪些嵌入应该同步

Design

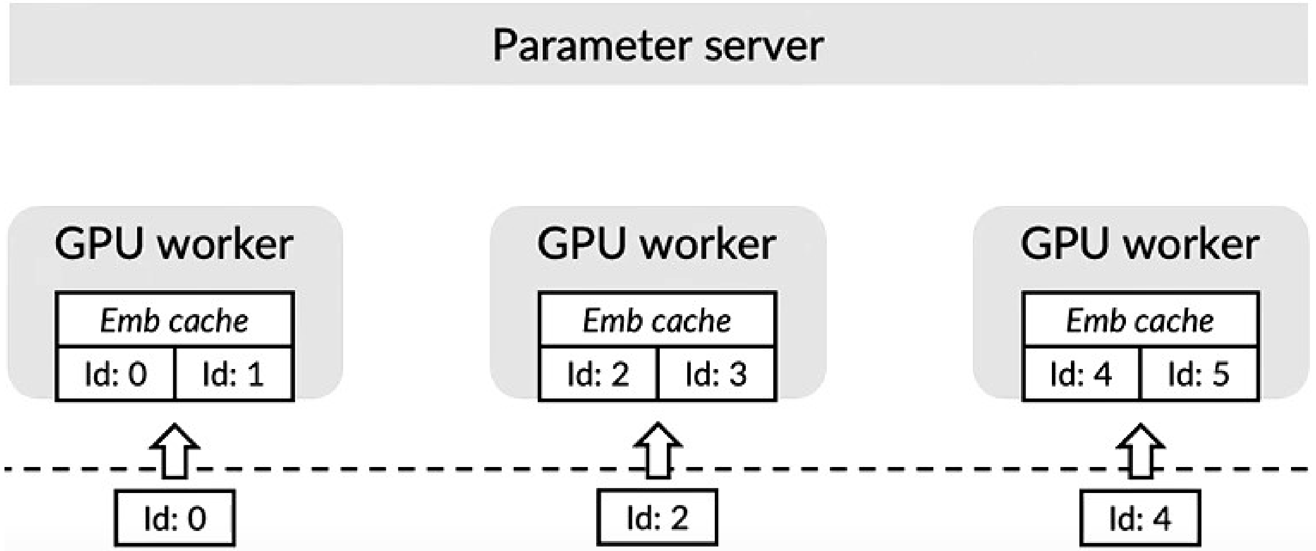
正向传播的缓存命中



- 尽可能更多的在训练中使用
  cache embedding
- 按需同步

Design

优化缓存命中

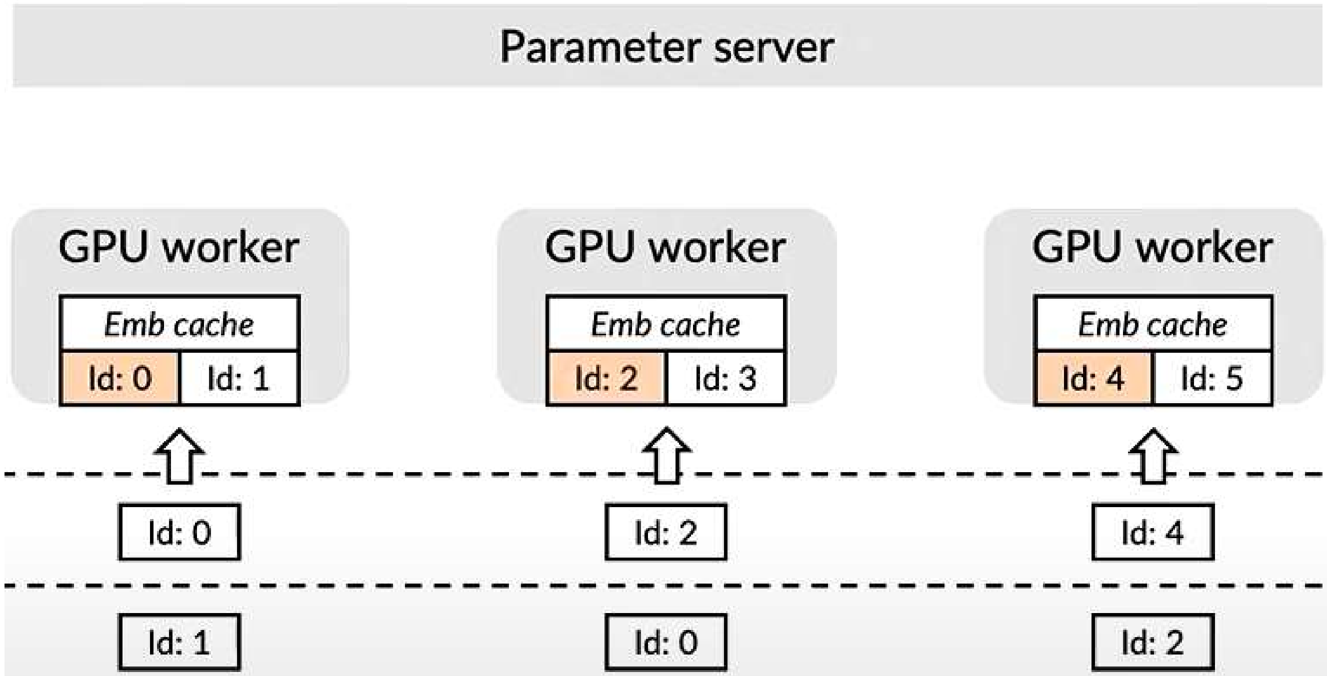


No worker pull

[NSDI '24 - Accelerating Neural Recommendation Training with Embedding Scheduling]()

Design

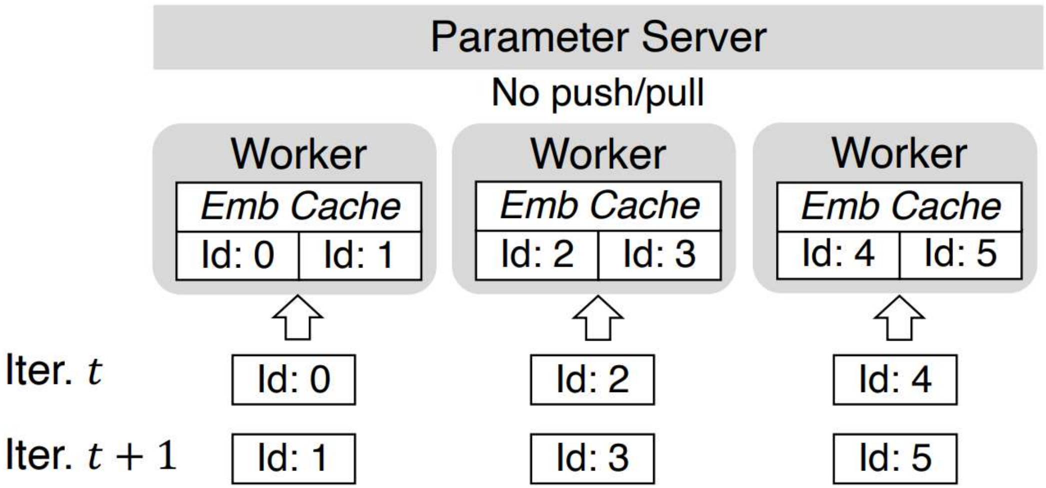
优化嵌入同步 (push)



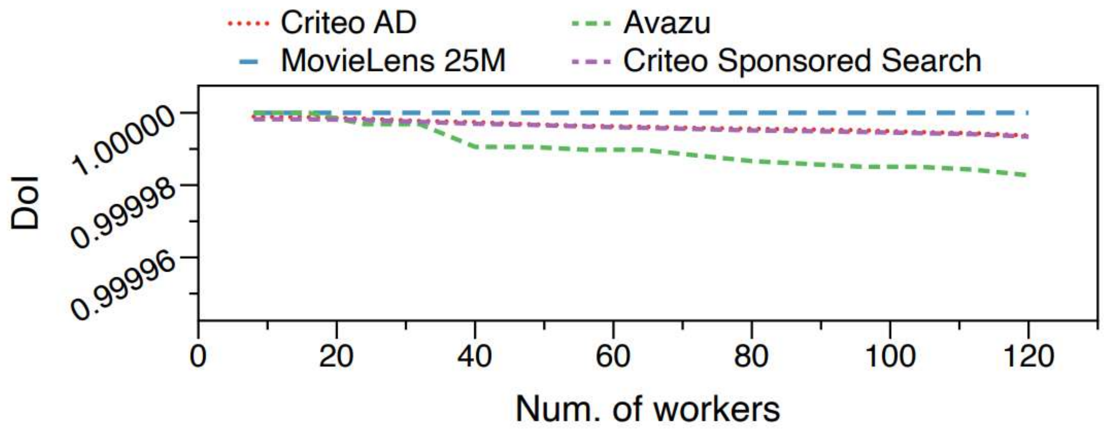
避免不必要的嵌入同步

Design

可行性分析

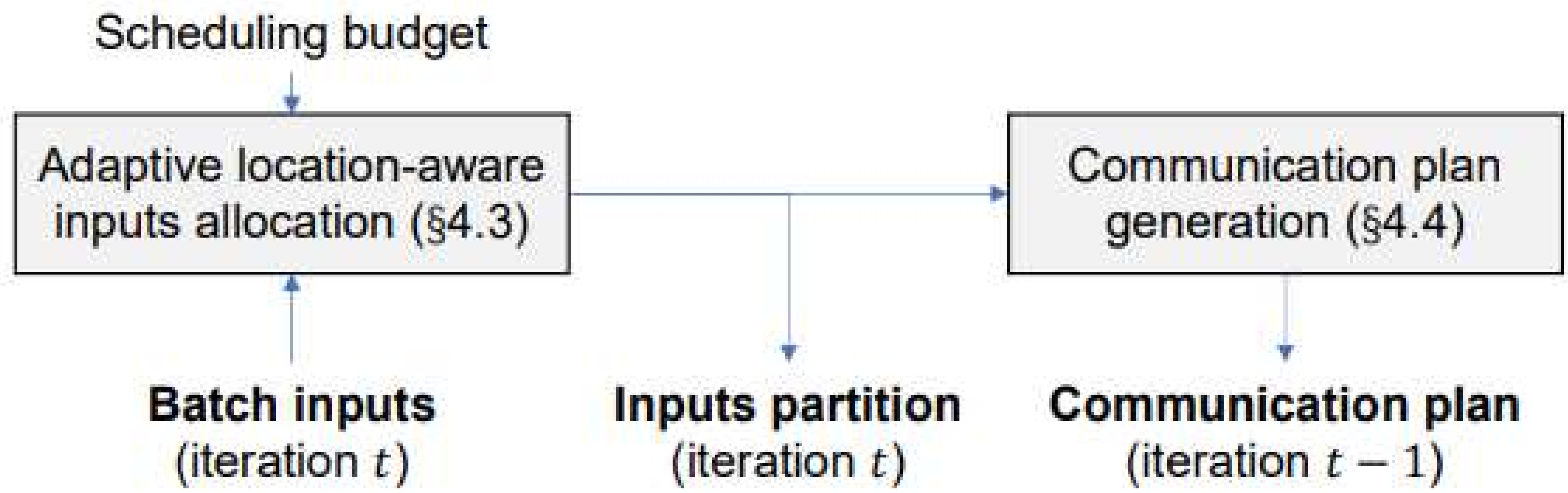


可预测性



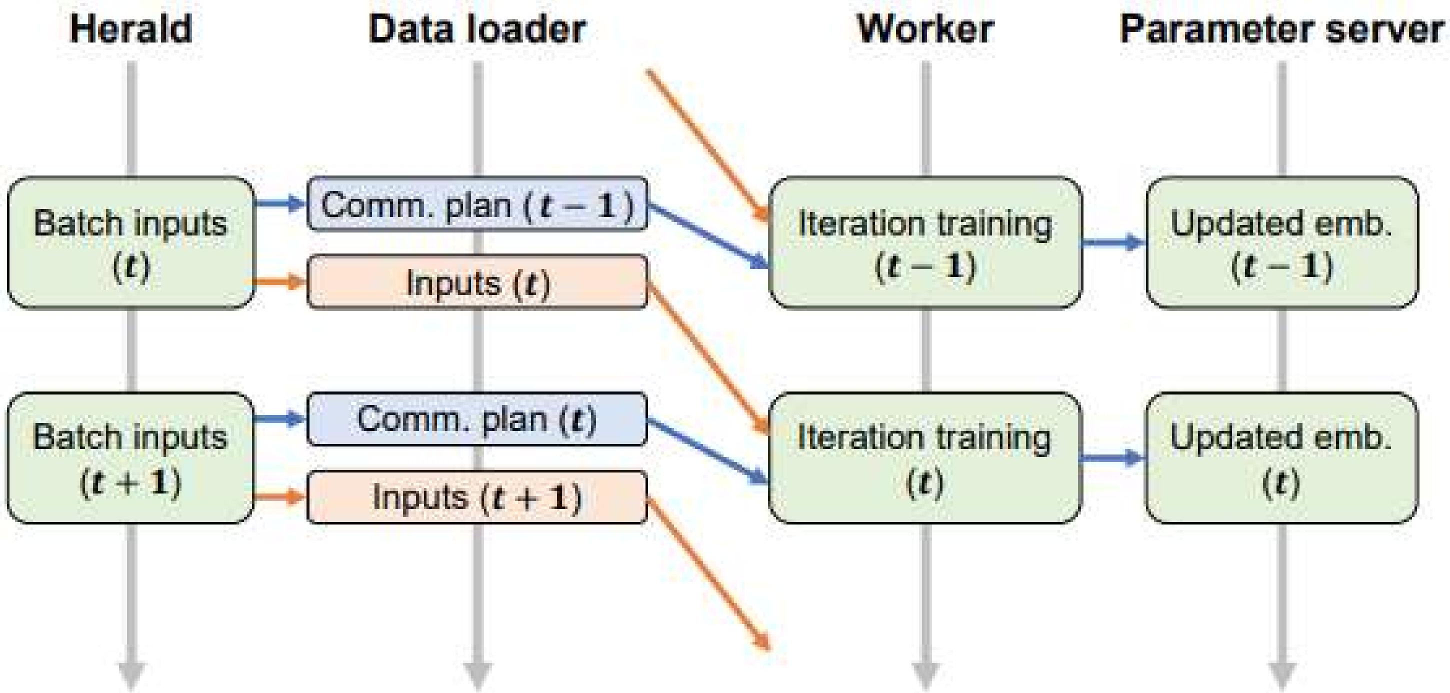
不频繁性

Design



Herald: 解耦目标以支持实时调度

Design



Herald training数据流

Design

Algorithm 1: Static LAIA

input : Batch samples (*Inputs*) and worker list (*Workers*)

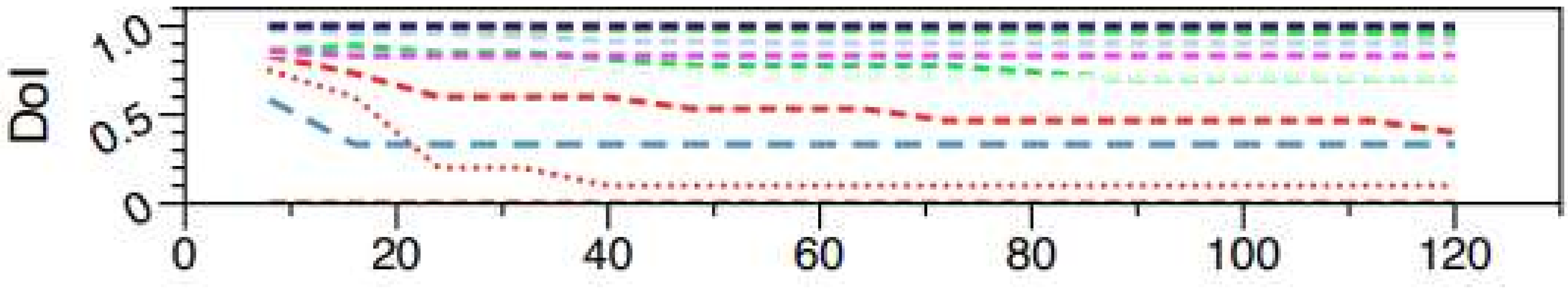
output : Inputs partition (*Alloc*)

1 Init *Alloc*;
2 Init all workers as available;
3 $capacity = size(Inputs)/size(Workers)$;
4 **for** i *in Inputs* **do**
5   **for** w *in Workers* **do**
6         $score_{(i,w)} = |cache(w) \cap embs(i)|$;
7   **end**
8   Find worker w with the largest score among the available workers;
9     $Alloc_{(i,w)} = 1$;
10   **if** $\Sigma_i Alloc_w == capacity$ **then**
11   Mark w as unavailable;
12   **end**
13 **end**
14 **return** *Alloc*;

Location-aware Inputs Allocation (LAIA)

- 计算分数量化相关性 (Line 6)
- 分配任务给得分最高的worker (Line 8-9)
- 保证各worker的任务均匀分布 (Line 10-12)
- 若存在共同最高分：倾向均匀分布地随机选择worker

Design

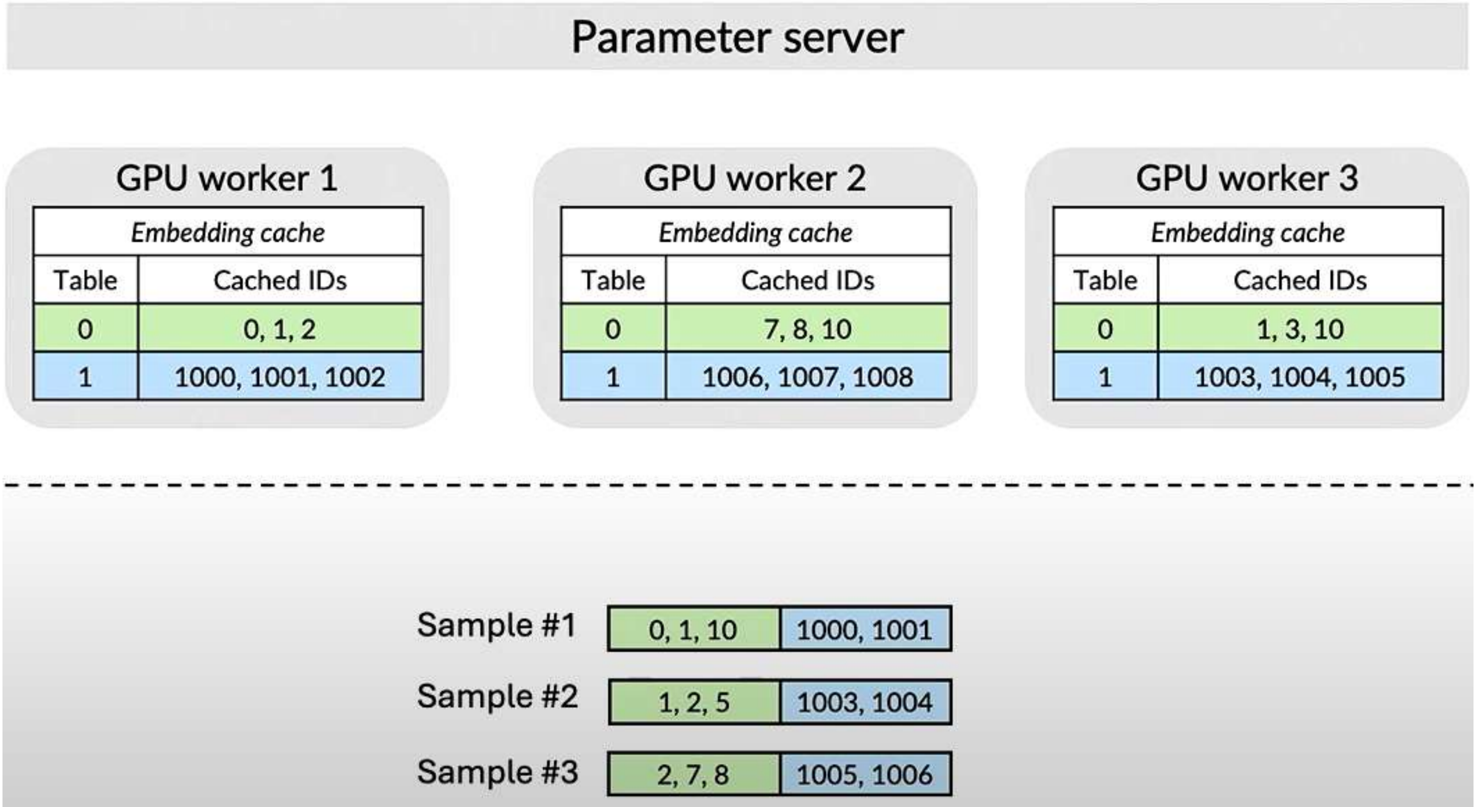


DOI 差异明显：低DOI更新更频繁，调度耗时高于调度优化

表分析：选取DOI最高（值得调度）的k个表进行调度

Design

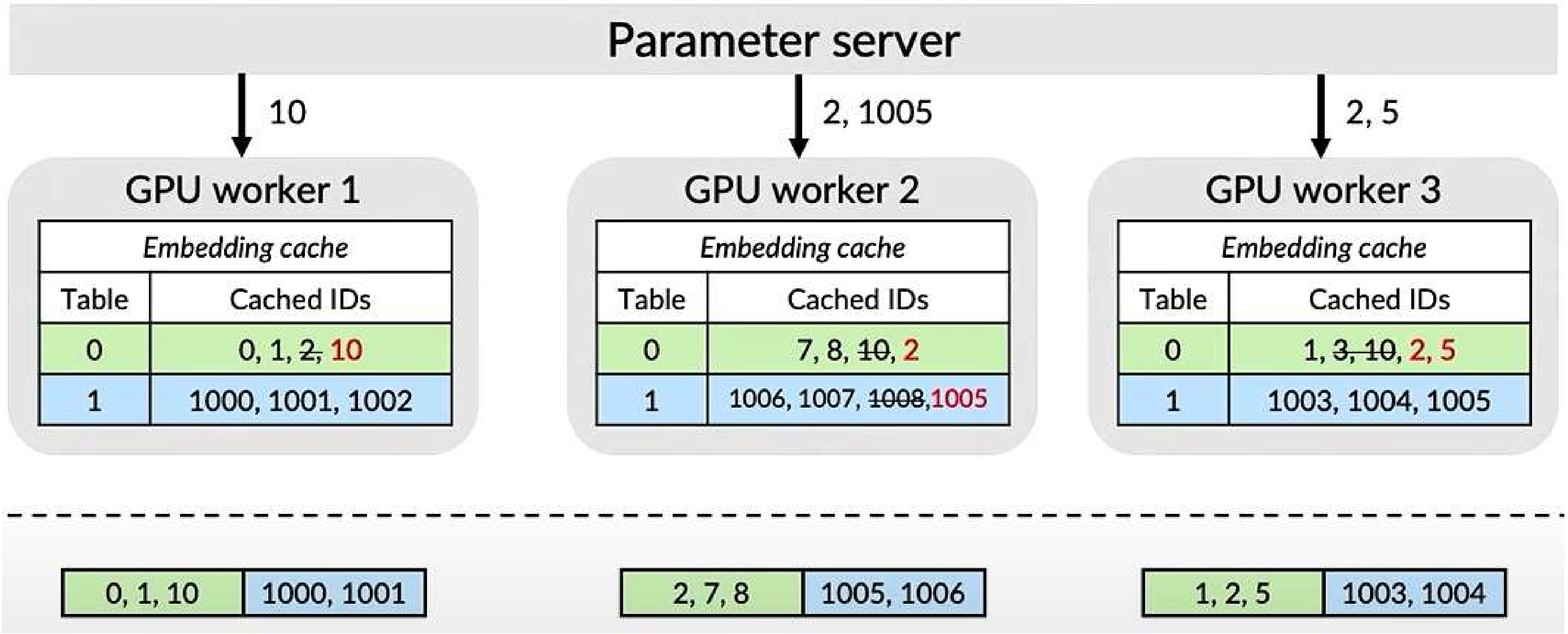
LAIA 案例



最高分任务分配

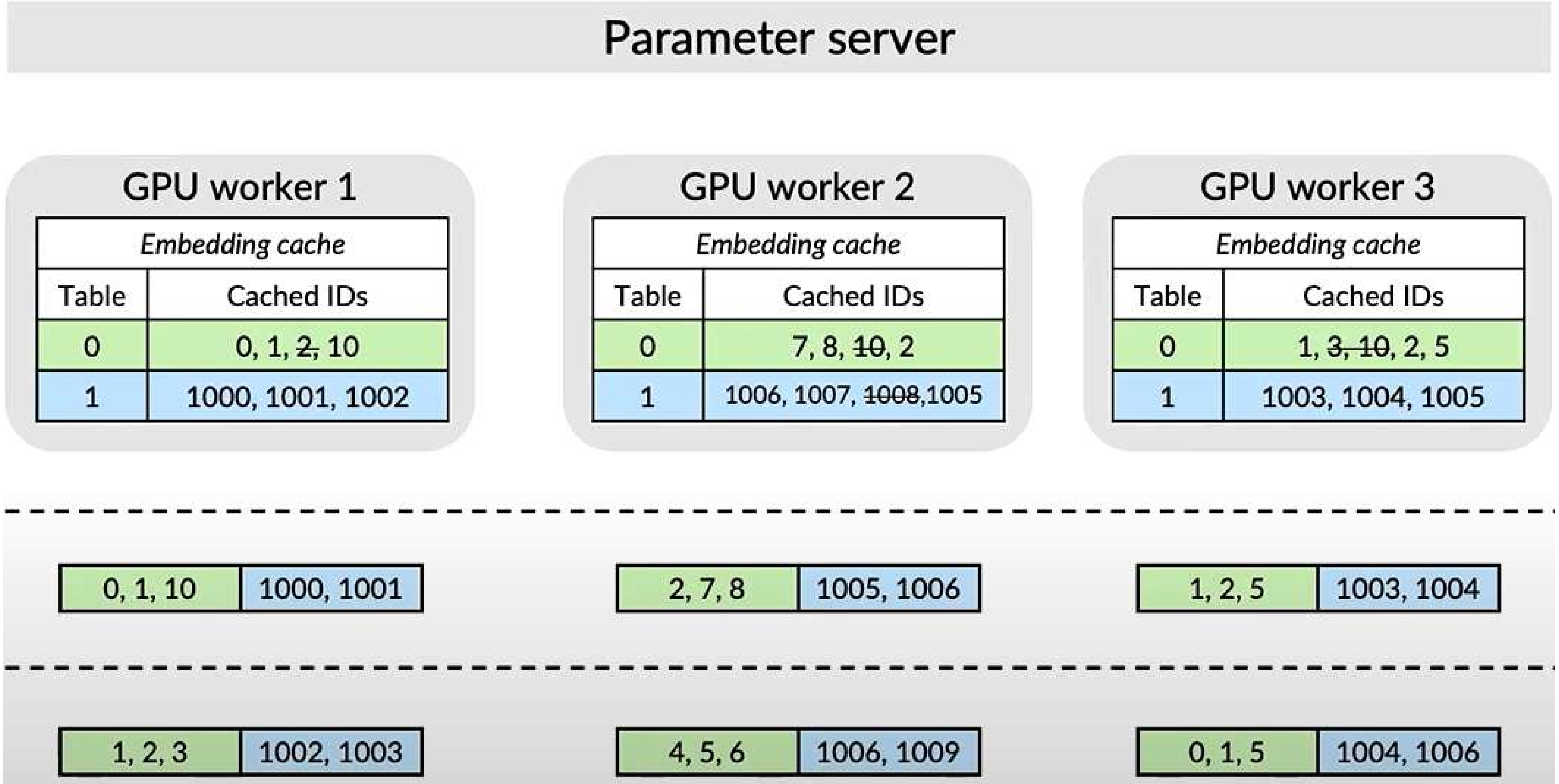
Design

Worker Pull

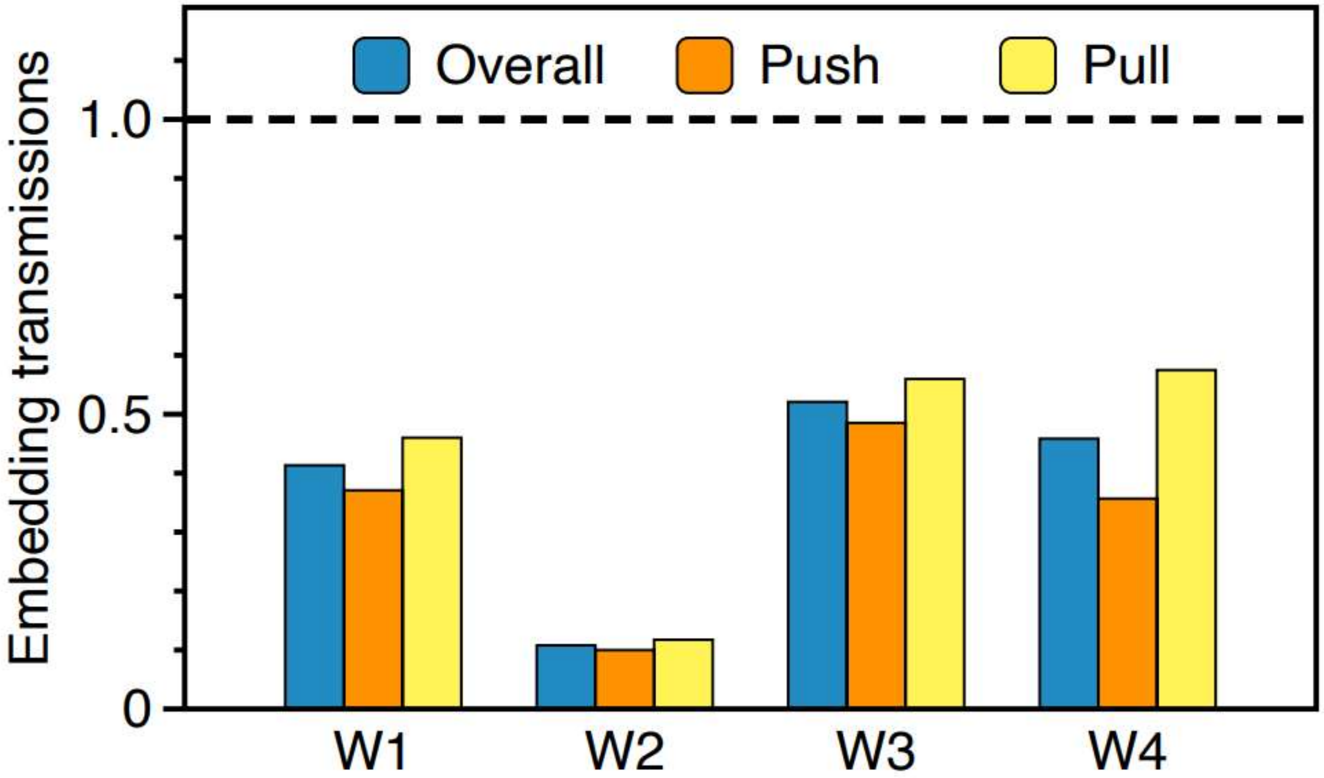


Design

Worker Push

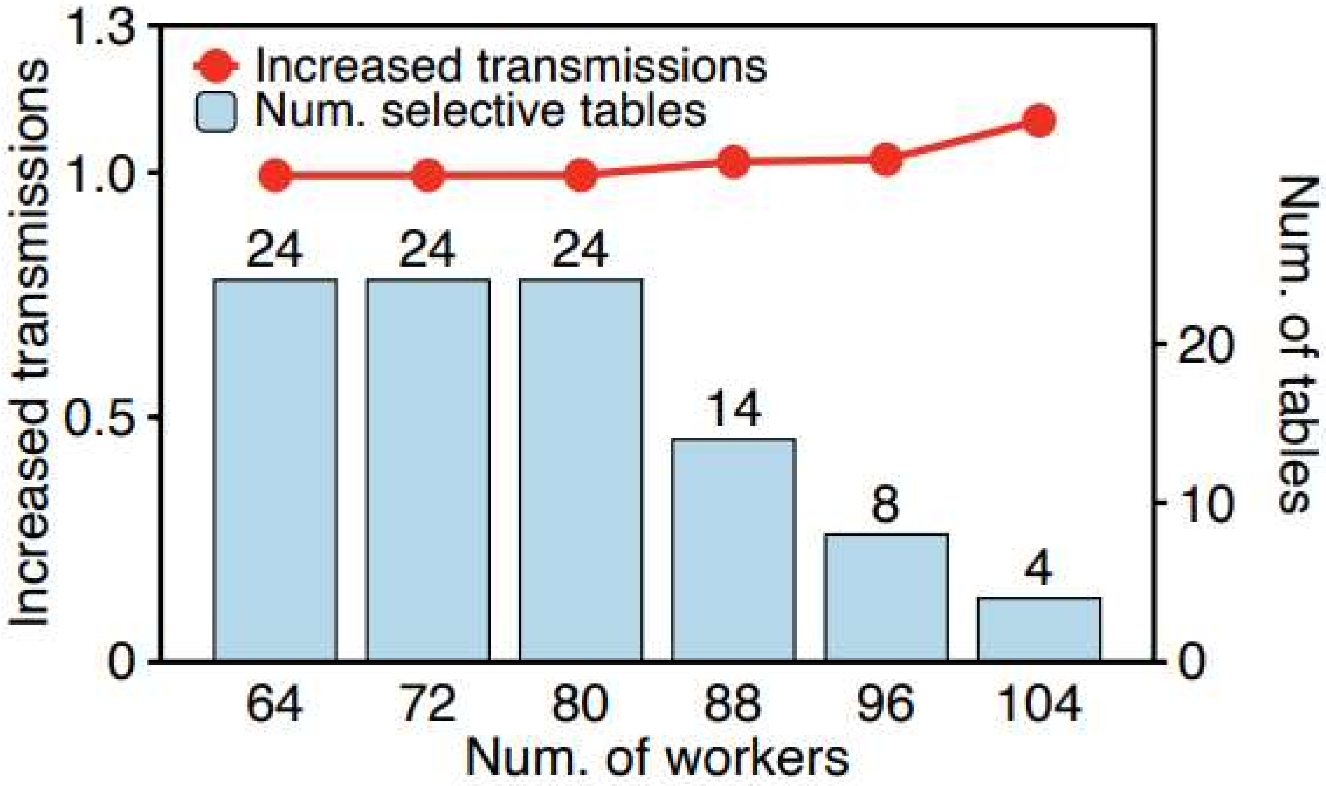


Evaluation



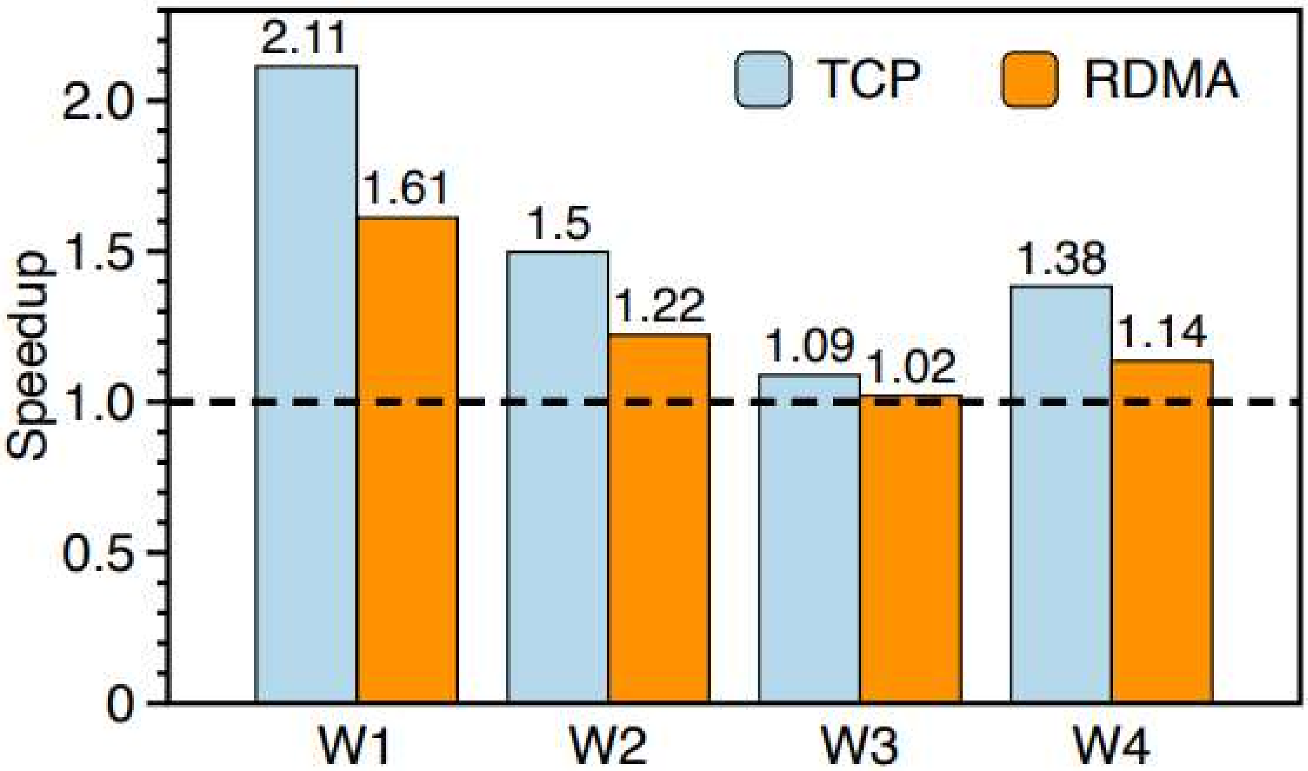
Herald平均减少了48%-89%的嵌入传输次数

Evaluation



自适应Herald与静态Herald相比，传输增量小于1.11倍

Evaluation



100Gbps以太网上，Herald相较于HET，在TCP和RDMA端到端训练中分别达到1.09-2.11倍和1.02-1.61倍的提升

Conclusion

- Herald利用嵌入缓存访问的可预测性和不频繁性（可行性）
- Herald应用自适应位置感知输入分配机制和按需同步策略来减少训练期间worker和PS之间的嵌入传输
- Herald可以显著降低嵌入通信开销，从而提高端到端推荐模型的训练效率

IDEAS

- 能否进一步提高？

✓ 如何找出更适合的k(选择嵌入表数)值，从而降低总体耗时；

✓ 在 LAIA 算法中，能否提出一个新的评分方法减少评分耗时；

✓ Herald 和 XDL 能否进行融合，进一步提升优化效果；

- 能否用到我们的场景？

✓ 有很好的适配性，尤其是对于算力有限的场景中；

Thanks for listening
请老师同学们批评指正

汇报人：黄　　凯
2024 年 7 月 4 日