

查论文 & 读论文

推荐系统训练阶段容错

Efficient Fault Tolerance for Recommendation Model Training via Erasure Coding

现状：检查点机制的高开销问题；减少检查点机制开销的技术，如近似检查点、异步检查点和日志记录等，存在各自的问题

解决：综合运用纠删码和复制技术，结合多种策略提升容错能力和训练效率

CPR: UNDERSTANDING AND IMPROVING FAILURE TOLERANT TRAINING FOR DEEP LEARNING RECOMMENDATION WITH PARTIAL RECOVERY（这篇文章是上一篇文章的引用）

现状：完全恢复高开销；部分恢复精度问题

解决：通过部分恢复提高推荐模型训练的效率和可靠性，在降低训练时间的同时，保持模型精度在可接受范围内

推荐系统训练阶段性能问题

Accelerating Communication in Deep Learning Recommendation Model Training with Dual-Level Adaptive Lossy Compression

现状：大规模训练时，其**嵌入表规模庞大**，导致多设备间的**全对全通信成为训练瓶颈**

解决：提出一种专为 DLRM 数据设计的优化混合误差有界有损压缩算法，包括量化编码器和无损编码器

RecShard: Statistical Feature-Based Memory Optimization for Industry-Scale Neural Recommendation

现状：DLRMs 的特点是使用嵌入层（embedding layers）将稀疏特征（如用户行为、内容类别等）转换为密集向量，这些嵌入层的内存需求极高（TB 级别），且计算密集度较低；随着模型规模的不断扩大，DLRMs 对**内存容量和带宽的需求呈超线性增长**，这给现有的硬件架构带来了巨大挑战

解决：通过分析嵌入表的访问模式和内存特性，提出了一种基于训练数据分布和模型特征的最优 EMB 分片策略，显著提高了 EMB 的训练吞吐量和内存利用率

Heterogeneous Acceleration Pipeline for Recommendation System Training

现状：推荐系统训练中的**内存和带宽限制**

解决：通过动态识别频繁访问的**嵌入表**条目（流行嵌入），并将这些流行嵌入存储在 GPU 的 HBM 中，而非频繁访问的嵌入存储在 CPU 的主内存中，从而优化了推荐系统训练的效率

推荐系统推理阶段性能问题

PIFS-Rec: Process-In-Fabric-Switch for Large-Scale Recommendation System Inferences

现状：DLRMs 的特点：与计算密集型模型不同，DLRMs**主要受限于带宽**，因为它们需要处理大量的**嵌入表（embedding tables）**

解决：PIFS-Rec 架构：该架构通过在 CXL 的 Fabric Switch 中实现近数据处理，利用下游端口的可扩展性和靠近内存的优势来加速嵌入表操作

Pushing the Performance Envelope of DNN-based Recommendation Systems Inference on GPUs

现状：DLRM 推理的性能瓶颈：DLRM 主要由嵌入、底部多层感知器（MLP）、特征交互和顶部 MLP 四个阶段组成。**其中，嵌入阶段由于频繁且不规则的内存访问，成为整个推理过程的主要瓶颈。**

Embedding Bag 操作通过 CUDA 线程并行处理，每个线程处理部分嵌入向量，但该操作中的 gather - reduce 操作导致不规则内存访问，影响嵌入阶段性能。

解决：提出软件预取和 L2 缓存固定等优化技术

EVStore: Storage and Caching Capabilities for Scaling Embedding Tables in Deep Recommendation Systems

现状：深度学习驱动的推荐系统依赖快速的模型推理，其中**嵌入向量（EV）表**用于将稀疏的分类数据转换为密集向量，是推荐系统的关键组件。然而，**随着推荐模型规模扩大，EV 表尺寸呈指数增长**，给内存管理带来巨大挑战。

解决：提出了 EVStore，通过创新的三层缓存设计优化深度学习推荐系统中嵌入表的存储和查找性能，在减少延迟、提高吞吐量和降低内存使用方面成效显著

Accelerating Personalized Recommendation with Cross-level Near-Memory Processing

现状：深度学习推荐系统中，**嵌入层因存储需求大、计算强度低和内存访问模式不规则，成为系统性能瓶颈**，制约了推荐系统的发展

解决：提出了一种跨级近内存处理（NMP）架构 ReCross，有效加速个性化推荐系统中的嵌入操作

思考

可以看到上面的论文不论是训练阶段的容错，还是训练/推理阶段的性能均面临**嵌入表规模庞大导致的内存需求高**的问题

训练容错聚焦内存高开销；训练优化聚焦内存访问和通信；推理优化聚焦内存管理和内存访问

感觉可以探索的点有：

1. 容错机制中，部分恢复是否可以和第一篇提到的纠删码和复制技术结合
2. 训练阶段性能中以降低通讯开销而提出的混合误差有界有损压缩算法是否能运用到推理中去
3. 训练阶段和推理阶段关于解决嵌入层内存访问问题的方法能否相互借鉴