

在公有云上的分布式可扩展的分布式机器学习

相关介绍

分布式机器学习集中在公有云网络（高节点间带宽，高性能GPU）

没有得到想要的加速比（128块GPU只有40倍的加速）

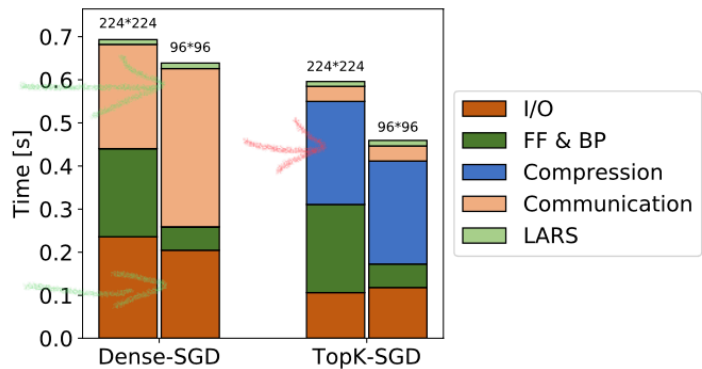
主要工作

- 提出一种对于GPU友好的 TOP_k 计算方法（传统 TOP_k 算法所导致令人诟病的额外计算开销）
- 使用分层次的通讯模式更好的利用带宽

其他工作

- 数据的多级缓存方案
- 并行的后梯度处理方案
- 方法的效率提升验证

存在的问题



TOP_k 随机梯度相比密集型随机梯度在每次epoch上速度有所提高但提升效果不好，关键在于 TOP_k 的计算对于gpu不够友好

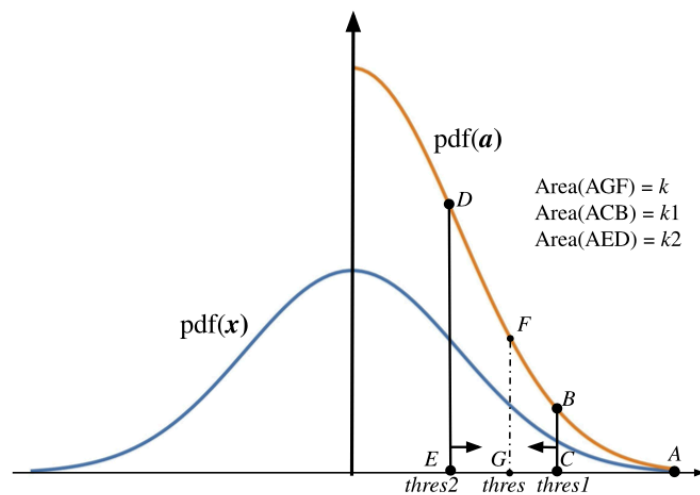
优化算法

- TOP_k 中门限值的查找采用二分法，近似查找符合要求的 TOP_k

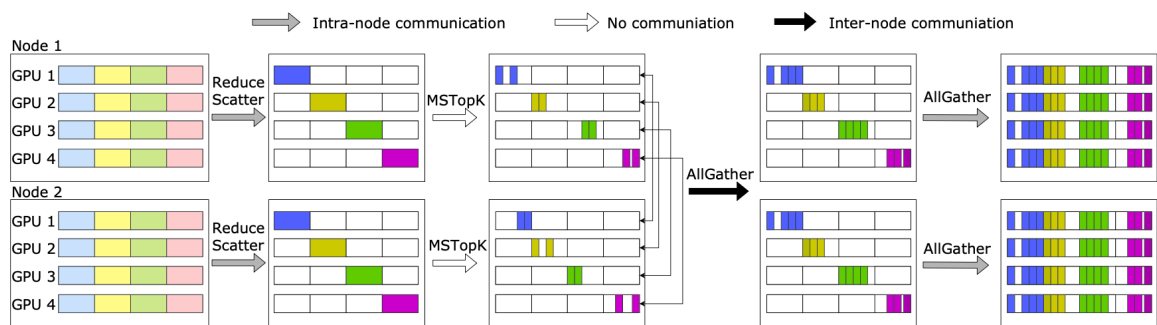
TopK算法的原理：

$$TopK(x, k)^{(i)} = \begin{cases} x^{(i)}, & \text{if } |x^{(i)}| > thres \\ 0, & \text{otherwise} \end{cases}$$

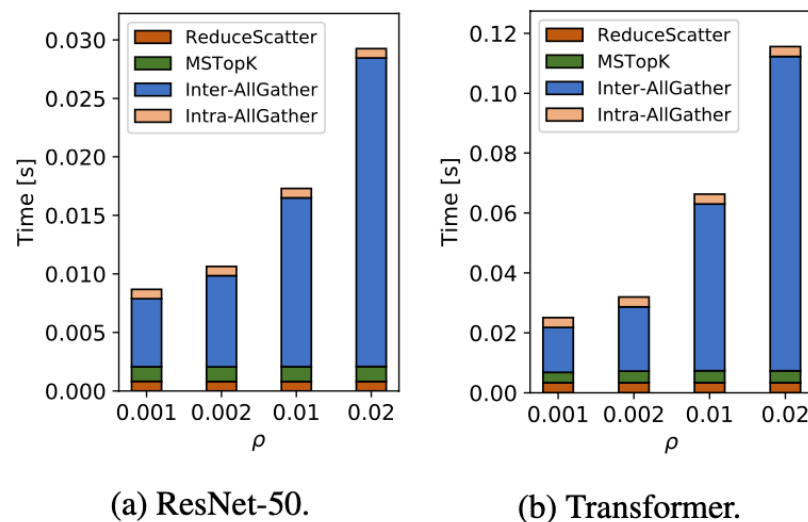
MSTopK采用二分法查找门限值，提高算法对于GPU的友好度



- 通讯方案优化，节点内/外采用分级通讯，充分利用节点内部NVLink提供的高带宽，使得通讯瓶颈细化到节点间通讯

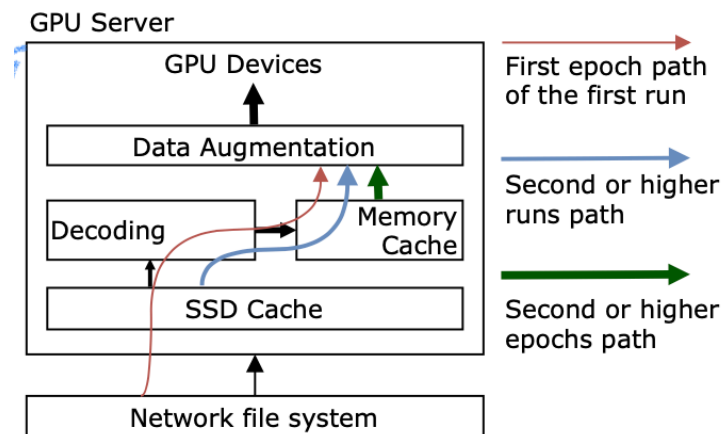


HiTopK将节点内通讯与节点外通讯相互隔离，充分利用不同层级结构的带宽性能

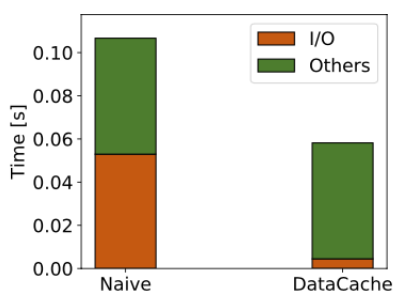


在后续的结果中可以发现，消息通讯的瓶颈还是集中在节点间通讯过程中。

- 在ssd和内存中设置缓存，为数据添加索引，利用缓存和索引以及异步加载的方法加速I/O



数据从数据仓库获取，对所有的数据添加索引，节点内对数据的存储采用多级缓存策略，同时异步pre-process过程可以减少数据使用使用时的时间开销



后续实验证明缓存策略减少了I/O时间消耗

小结

1. 采用优化方案对模型进行优化时也要考虑优化方案的收益（是否引入了额外的开销）
2. 文中提出的MSTopK算法理论上能够加速TopK的查找速度，但具体的加速效果没有明示（猜测和GPU的型号以及运行环境有关，难以量化/递归本身的特点，不清楚是否包含类似剪枝操作的步骤）
3. 文中所有的训练数据均由数据仓库提供，大型数据仓库本质上也是分布式存储系统；而文中提出的缓存模型，在一定程度上是分布式存储系统的雏形；这种缓存系统的思想可以尝试运用在其他方面，起到削峰填谷的效果
4. 模型最终的训练效果和密集形训练网络一致
5. 文中尽管对于模型在训练过程中的各个流程做了优化，但最终瓶颈还是在节点间通讯，在带宽不变的情况下，如何提高单位数据的信息容量是需要关注的目标（开源节流）。一方面可以从数据压缩入手（正如各种梯度压缩算法），一方面可以从通讯协议入手（和实时通讯网络一样，DML也有高带宽/低延时的网络需求，例如WebRTC），提高数据传输量。
6. 论文研究背景还是在公有云网络（高带宽/稳定性好）