



Petuum Poseidon高效的分布式深度学习平台

主讲人：Petuum技术主管 于伟仁
Petuum技术主管 吴竑



Petuum Poseidon: 高效的分布式深度学习平台

主讲人：于伟仁，吴竑
Petuum Inc.

创始人: Eric Xing, 卡内基梅隆大学(CMU)著名教授, 人工智能学界权威

愿景: Petuum Inc. 致力于打造新一代全方位的人工智能操作系统和机器学习软件解决方案, 应用分布式计算和虚拟化, 使得开发企业级AI解决方案更简化更便捷。通过使用标准化的构件和方法, Petuum 产品让各公司团体机构, 拥有创建高效、可扩展的AI/ML解决方案的能力。

2017年1月: 权威机构评选为全球最有前途的100家AI公司之一 (CB Insights AI 100)

2017年9月: 长城会 G-Summit 全球科学创新峰会 AI Startup Top 10.



@Pittsburgh, PA, USA

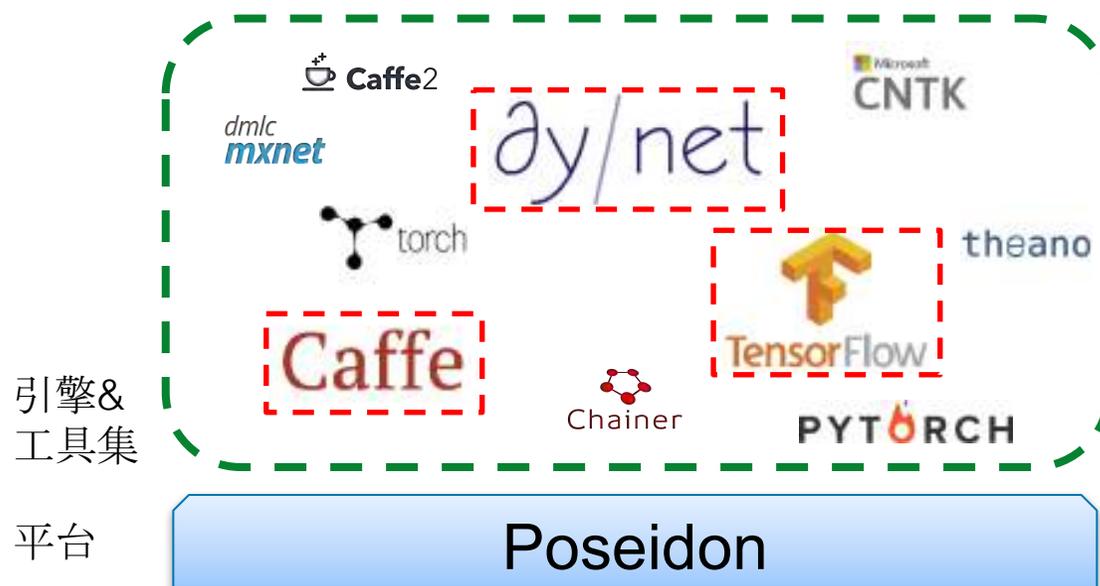


三个产品:

- **Petuum OS** , 企业级分布式人工智能/机器学习操作系统, 能够桥接目前业界所有数据类型、任何编程语言、各种硬件架构。
- **Poseidon** , 高效的分布式深度学习平台, 允许用户使用尖端设计轻松构建新的ML和DL模型, 运行为流行软件 (TensorFlow , Caffe , scikit-learn等) 编写的现有DL程序。
- **PetuumMed** , 医疗行业解决方案, 从海量的EHR数据和医学影像数据挖掘出及时和可行的见解, 帮助医生了解患者的临床状况以及医疗决策支持。

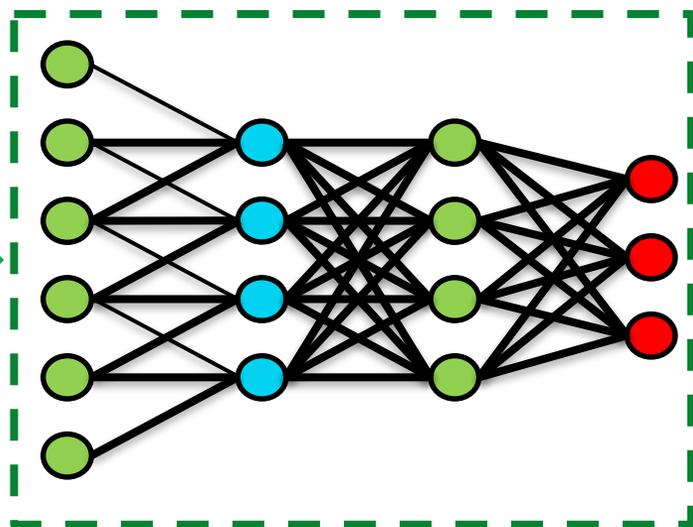


- Poseidon: 一个高效的分布式深度学习平台
 - **增强**现有深度学习引擎
 - **兼容**现有引擎编程接口
 - 现有深度学习模型不需更改任何代码，就可以运行在Poseidon上

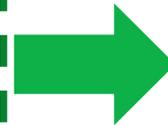




训练数据集：
带标签的图像数据



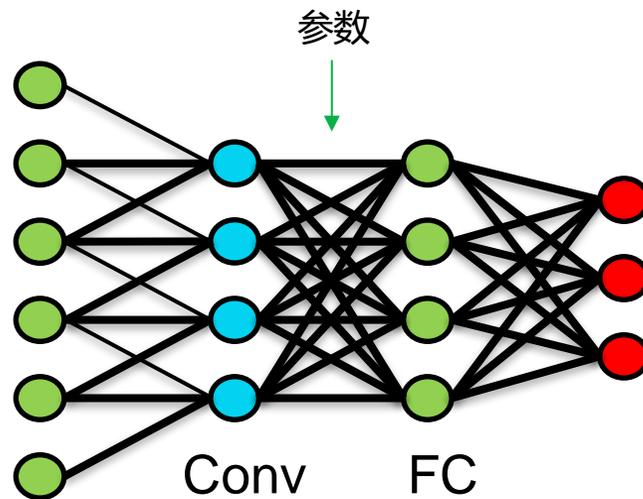
深度学习模型



正确识别图像



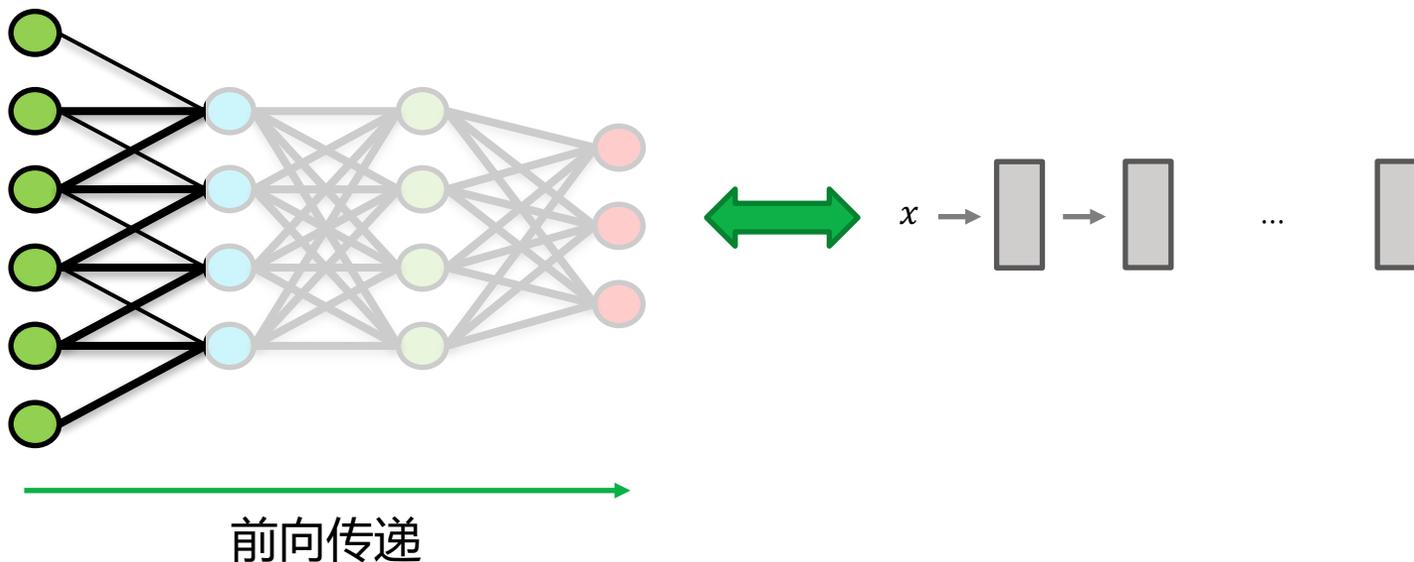
- 本质上，一个神经网络由很多“层”组成
- 每一层对应着复杂的计算：卷积层，全联通层
- 层之间每条链接的权重，是神经网络模型的参数 (Parameter)
- 如何计算模型参数？





训练神经网络：前向传递

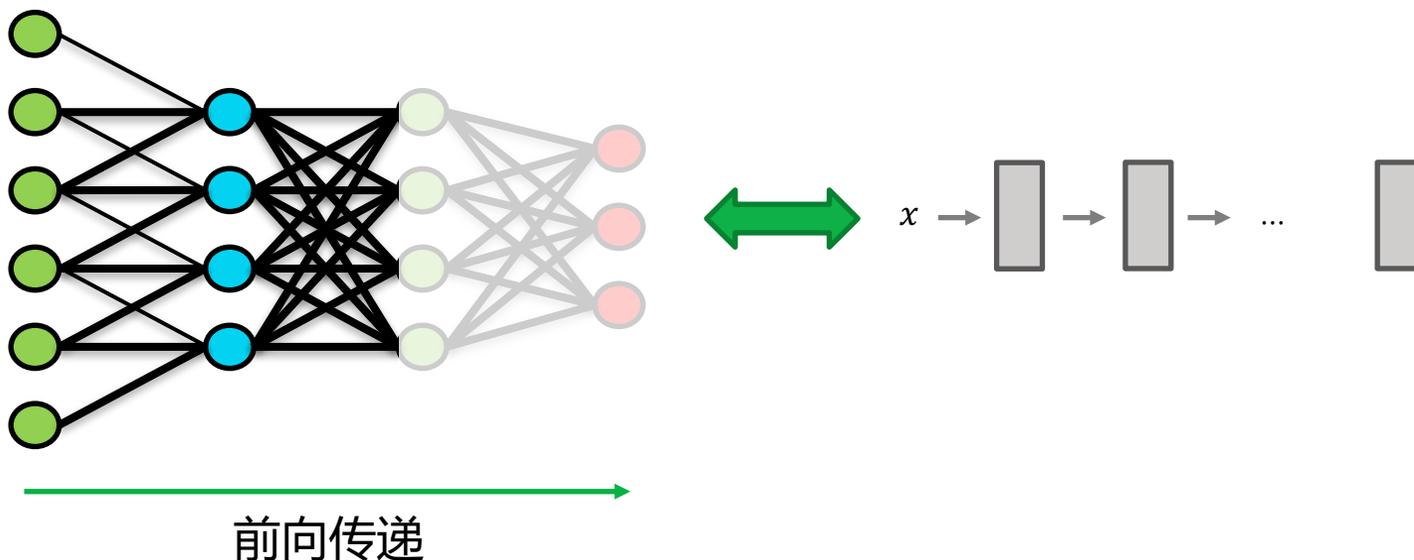
- 反向传播 (backward propagation, BP) 是训练神经网络的主要算法
- 反向传播定义了训练流程，数据会传递过神经网络两遍（前向，后向）
 - 前向传递中，逐层通过神经网络，直到得到Loss \mathcal{L} 函数的值
 - 后向传递中，逐层计算参数的梯度值(gradients)，用以叠加更新参数
 - 如此往复，直到收敛





训练神经网络：前向传递

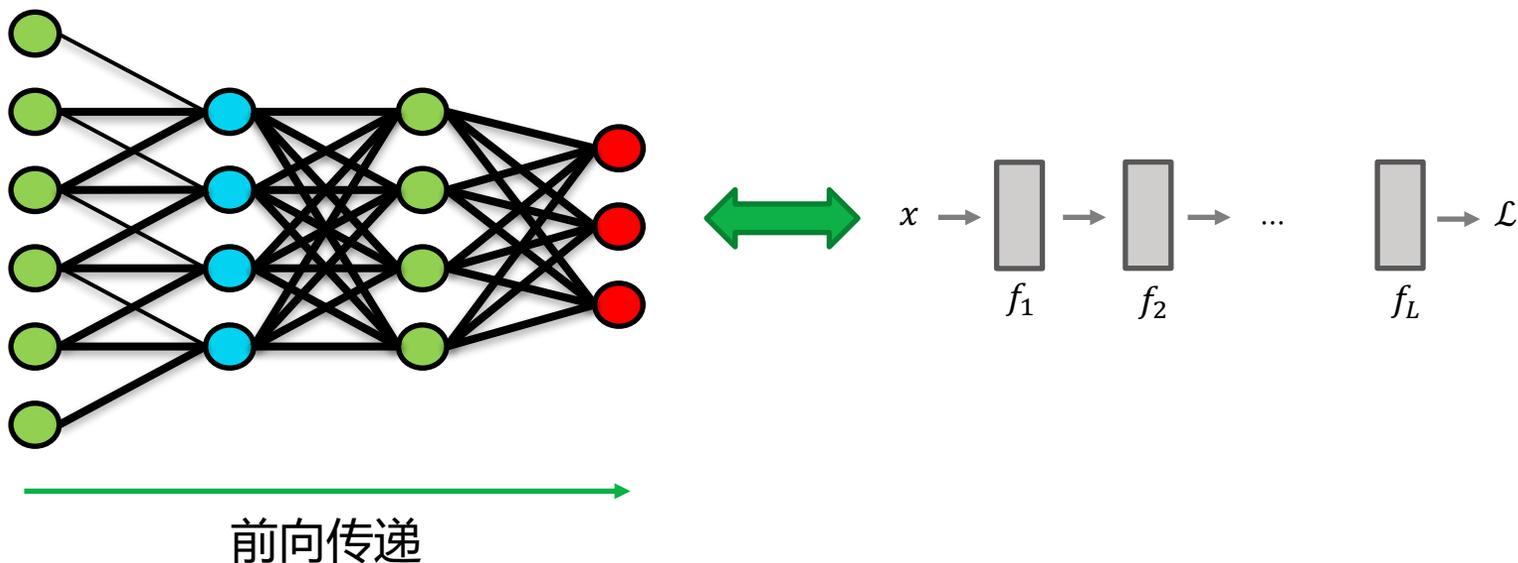
- 反向传播 (backward propagation, BP) 是训练神经网络的主要算法
- 反向传播定义了训练流程，数据会传递过神经网络两遍（前向，后向）
 - 前向传递中，逐层通过神经网络，直到得到Loss \mathcal{L} 函数的值
 - 后向传递中，逐层计算参数的梯度值(gradients)，用以叠加更新参数
 - 如此往复，直到收敛





神经网络中的前向传递

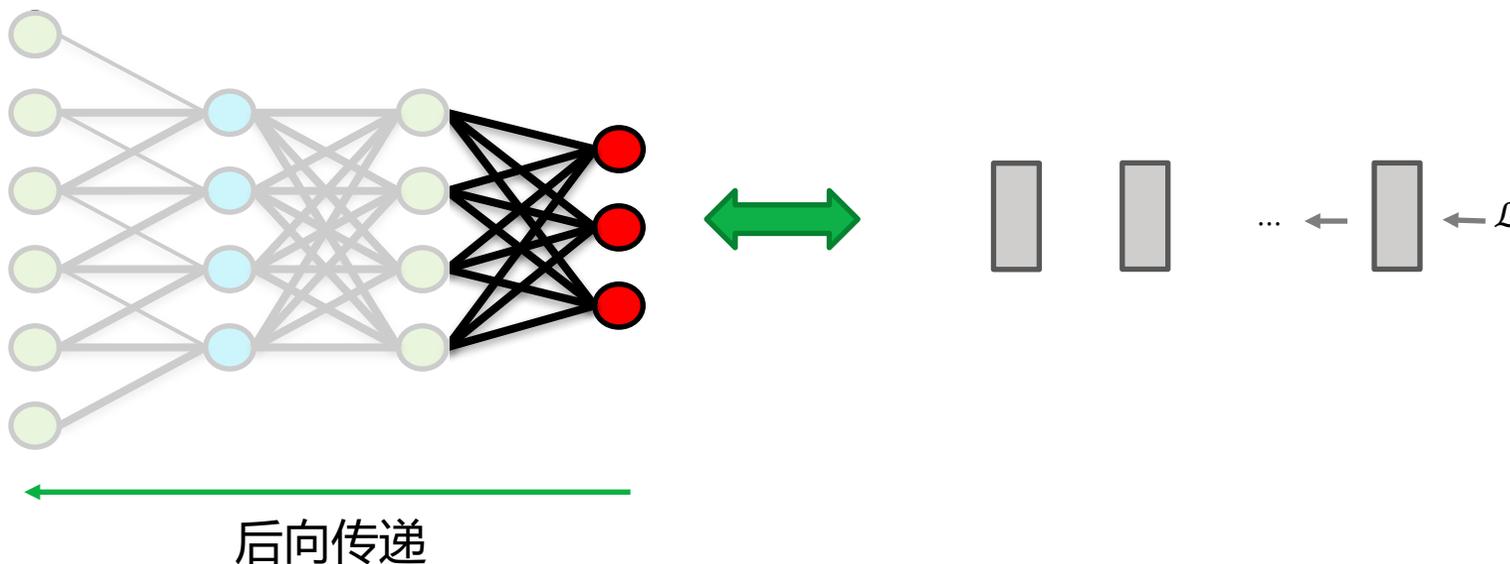
- 反向传播 (backward propagation, BP) 是训练神经网络的主要算法
- 反向传播定义了训练流程，数据会传递过神经网络两遍（前向，后向）
 - 前向传递中，逐层通过神经网络，直到得到Loss \mathcal{L} 函数的值
 - 后向传递中，逐层计算参数的梯度值(gradients)，用以叠加更新参数
 - 如此往复，直到收敛





神经网络中的后向传递

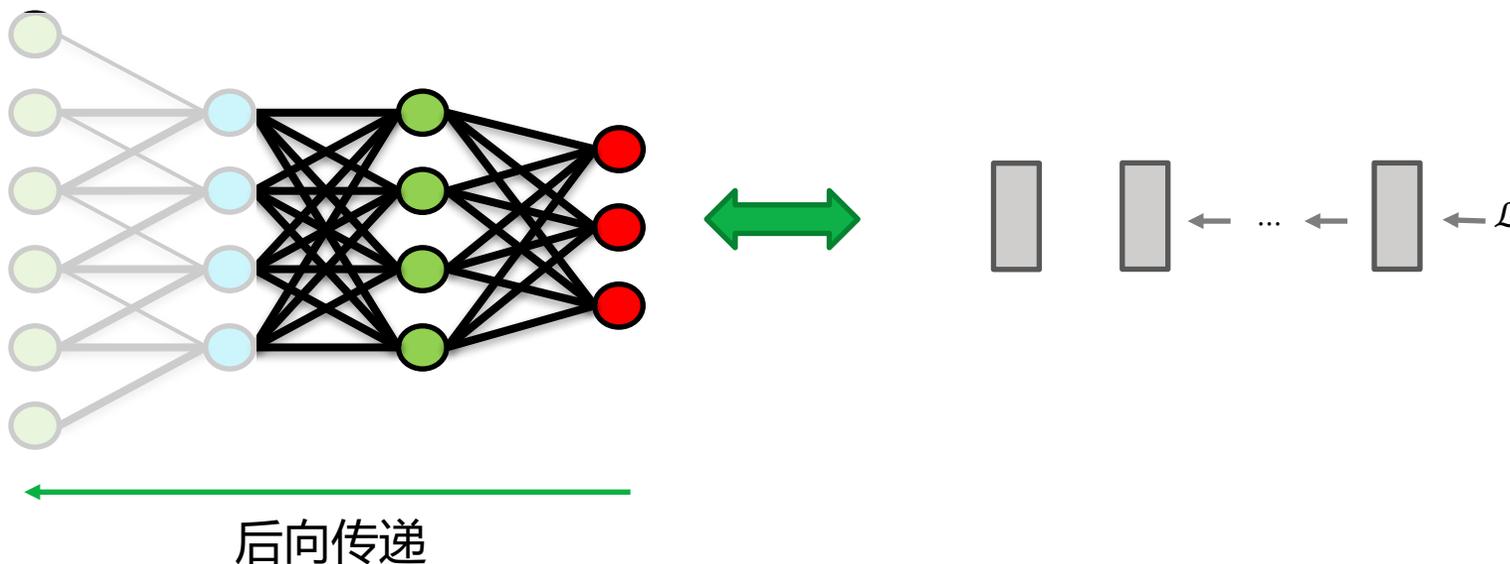
- 反向传播 (backward propagation, BP) 是训练神经网络的主要算法
- 反向传播定义了训练流程，数据会传递过神经网络两遍（前向，后向）
 - 前向传递中，逐层通过神经网络，直到得到Loss \mathcal{L} 函数的值
 - 后向传递中，逐层计算参数的梯度值(gradients)，用以叠加更新参数
 - 如此往复，直到收敛





神经网络中的后向传递

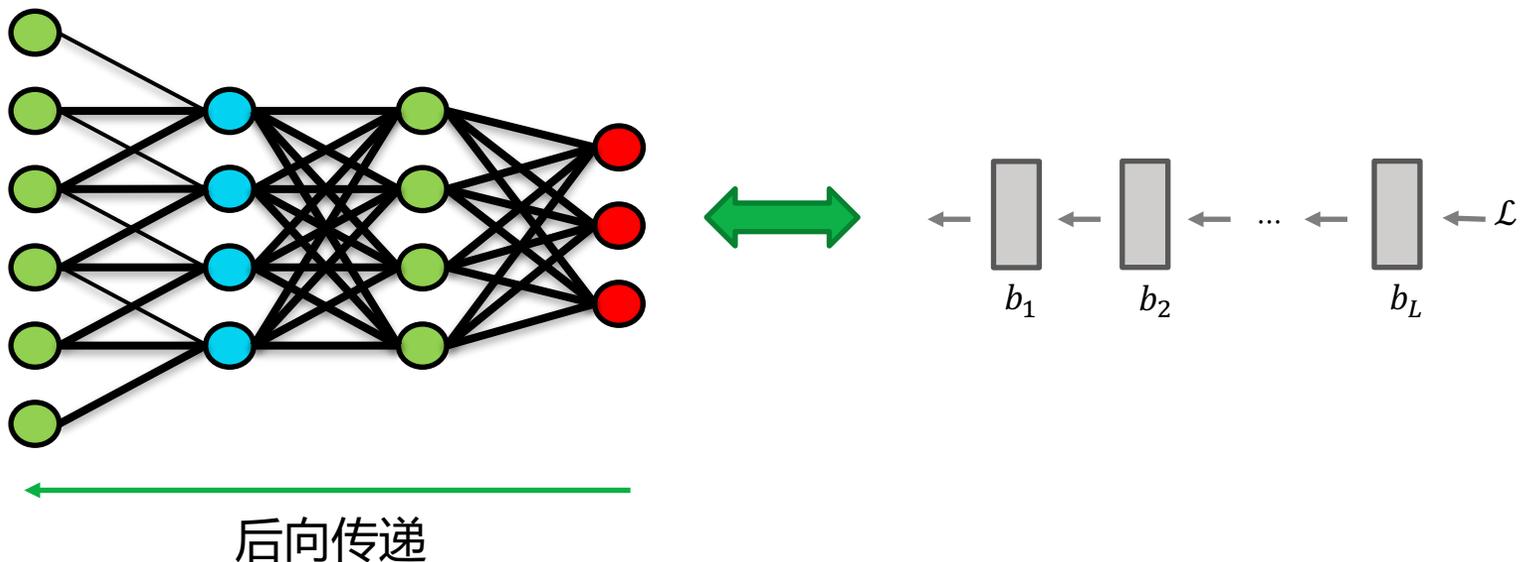
- 反向传播 (backward propagation, BP) 是训练神经网络的主要算法
- 反向传播定义了训练流程，数据会传递过神经网络两遍（前向，后向）
 - 前向传递中，逐层通过神经网络，直到得到Loss \mathcal{L} 函数的值
 - 后向传递中，逐层计算参数的梯度值(gradients)，用以叠加更新参数
 - 如此往复，直到收敛





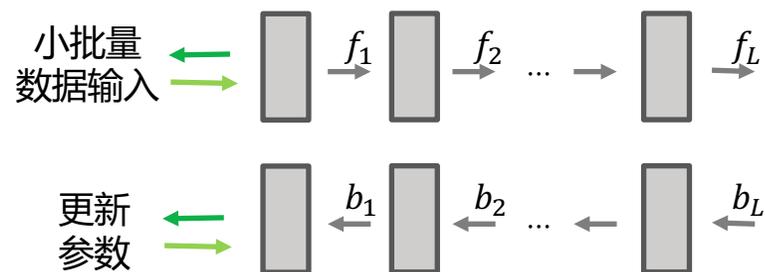
神经网络中的后向传递

- 反向传播 (backward propagation, BP) 是训练神经网络的主要算法
- 反向传播定义了训练流程，数据会传递过神经网络两遍（前向，后向）
 - 前向传递中，逐层通过神经网络，直到得到Loss \mathcal{L} 函数的值
 - 后向传递中，逐层计算参数的梯度值(gradients)，用以叠加更新参数
 - 如此往复，直到收敛

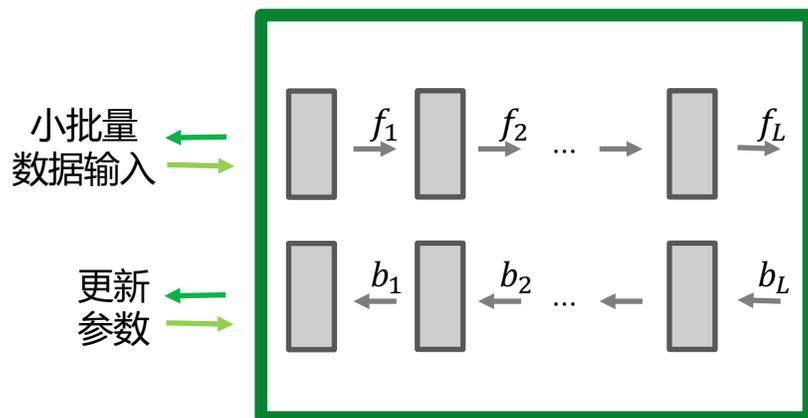




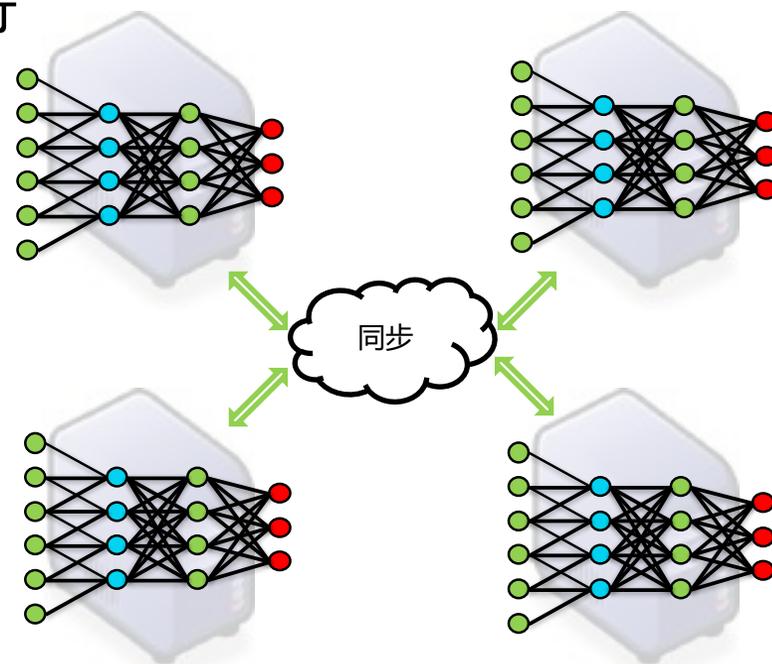
- 将随机梯度下降算法（mini-batch SGD）应用到反向传播（BP）
 - 随机梯度下降算法：定义了梯度值计算方法，以及同步流程。
 - 1. 前向：顺序执行 f_1, f_2, \dots, f_L 操作
 - 2. 后向：顺序执行 b_L, b_{L-1}, \dots, b_1 操作
 - 3. 更新：将梯度值应用到现有的模型参数上去
 - 如此往复



- 分布式深度学习: 用多台计算机并行化深度学习模型训练.
- 即, 将最繁重的任务分发到多台计算机上去并行执行

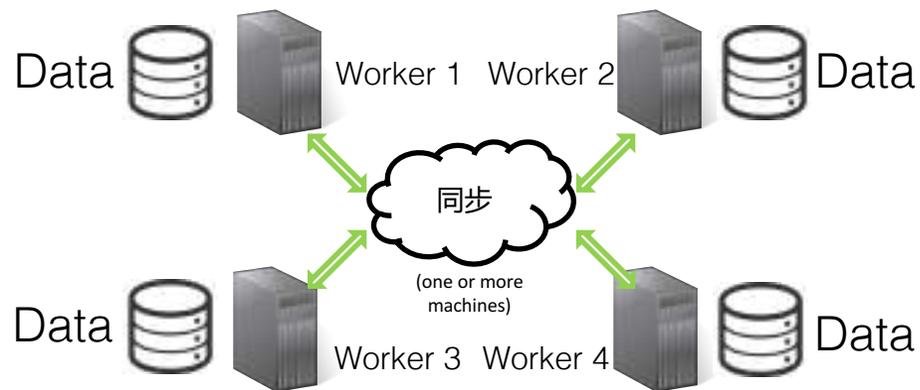


前向传递和后向传递操作
占据训练过程99%的计算量



配备GPU的计算节点集群

- 基于随机梯度下降 (mini-batch SGD) 进行数据并行化
 - 数据处理并行：分割为不同的分区
 - 梯度值计算并行：分布到不同的计算节点
 - 同步并行：利用参数服务器，对梯度值进行同步后叠加.
 - 通信：节点发送梯度值，获取叠加后参数值，**需要大量通信带宽！**

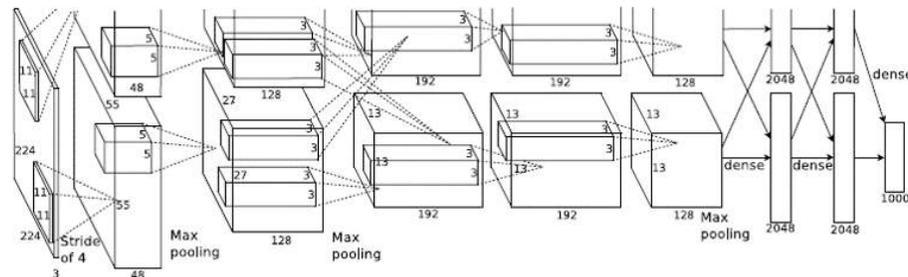




分布式深度学习：案例与瓶颈分析

- 训练AlexNet, 梯度值产生速率 $240\text{M floats}/(\text{s}*\text{GPU})$
 - 61.5M 浮点数, 0.25s/迭代, Geforce Titan X (batchsize = 256)

Figure from
Krizhevsky et al. 2012



- 使用参数服务器，在8个计算节点上并行计算。
 - 为了加入更多计算节点，性能会线性提升，GPU计算不会阻塞
 - 假设: 每个计算节点上存储 1/8 的所有参数
 - 每个节点需要带宽： $240\text{M} * 7/8 * 4 = 840\text{M floats/s} = \mathbf{26\text{Gbps}}$



- 这是我们现在的硬件

实验室中最普遍的网络条件

Ethernet standards

Ethernet	Rate(GBit/s)	Rate (Mb/s)	Rate (# floats/s)
1 GbE	1	125	31.25M
10 GbE	10	1250	312.5M
Infiniband	40	5000	1250M

AWS上最贵的计算资源 (\$18/小时?)

专用硬件! 并非常用硬件, 非常昂贵

- 不幸的是, 这个问题比描述的更加严重
 - 我们仅使用了8个计算节点 (小集群). 32, 128甚至256个节点?
 - 我们也未将其他因素纳入考量, 比如,
 - 网络会与其他应用共享, 造成更低的有效带宽.
 - 突发式的通信流量在GPU上非常常见 (稍后解释).
- 正是通信问题, 使得在分布式环境下, 多机加速比受到了限制!



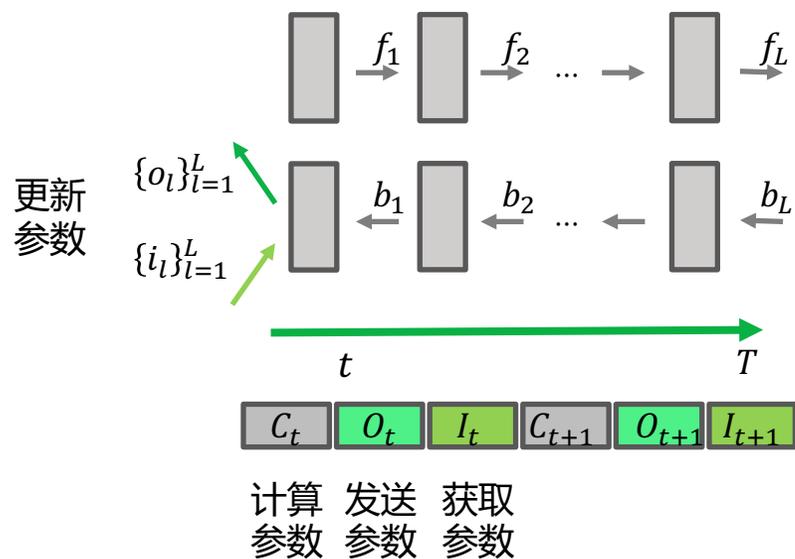
- Poseidon平台
 - 增强现有深度学习引擎，兼容现有平台编程接口
- Poseidon的通信技术
 - 无等待的反向传播 (WFBP): 通过将通信与计算时间重叠来隐藏通信开销
 - 混合通信模型 (HybComm): (无损地)减少网络传输流量



无等待的反向传播 (WFBP)

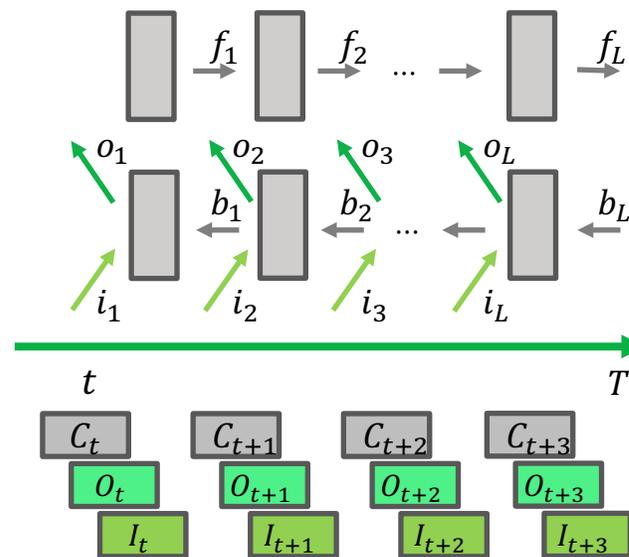
- 利用流水线调度将计算与通信重叠
 - 通信开销隐藏在计算开销之下，结果: 在单位时间内进行更多的计算
 - 解决突发性通信问题

计算与通信顺序发生！



流水线调度

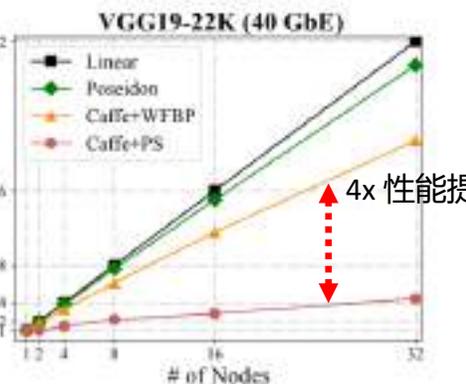
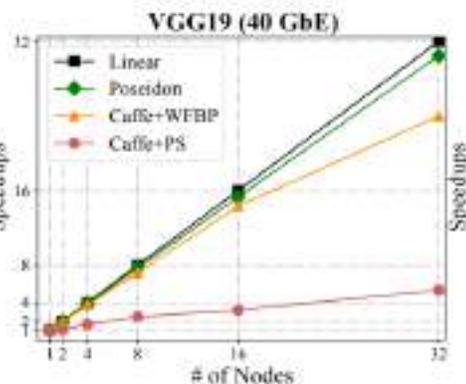
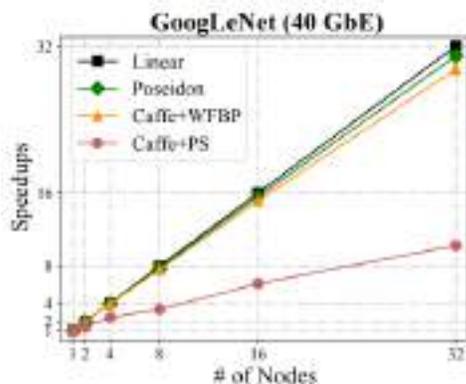
计算与通信并发，隐藏通信开销！



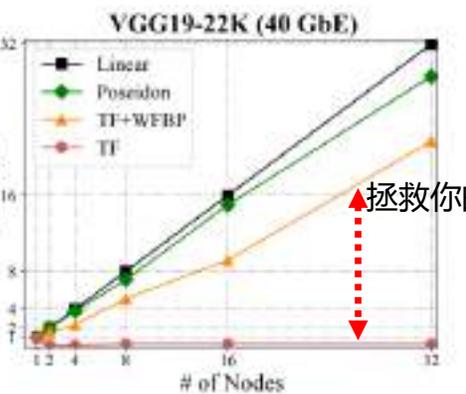
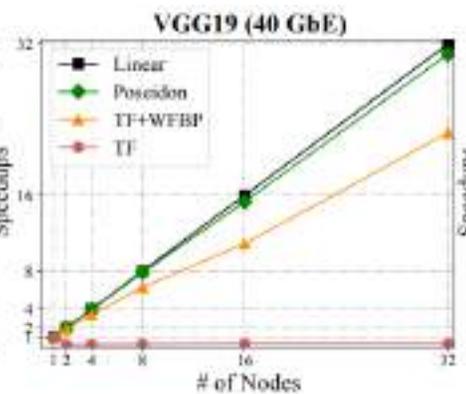
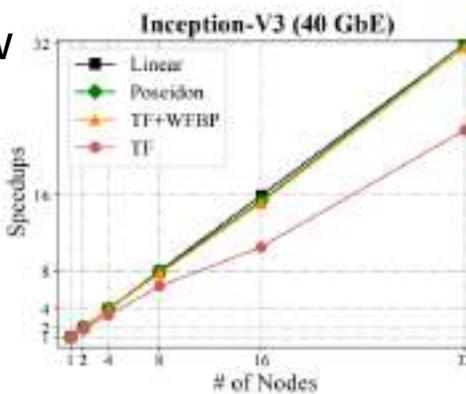


WFBP: 无等待反向传播

- 性能怎么样？
 - On Caffe：



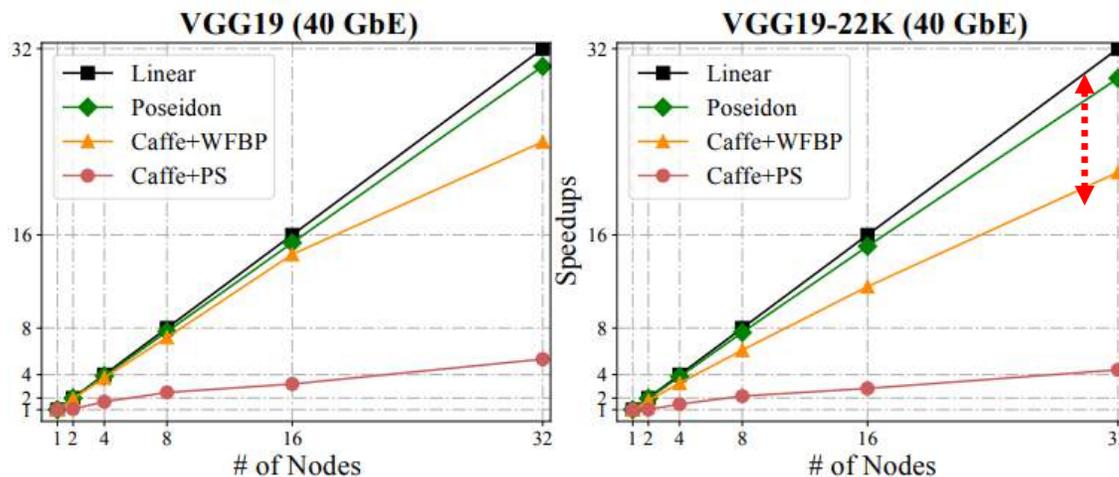
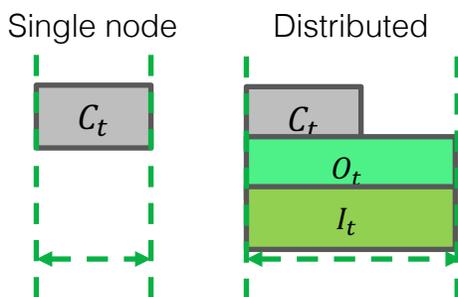
- On TensorFlow





WFBP: 无等待反向传播

- 这能解决所有的问题吗？
 - 不能，当通信时间比计算还要长的时候（如下图）。
 - Gap：理想情况与现实存在的差距



Gap

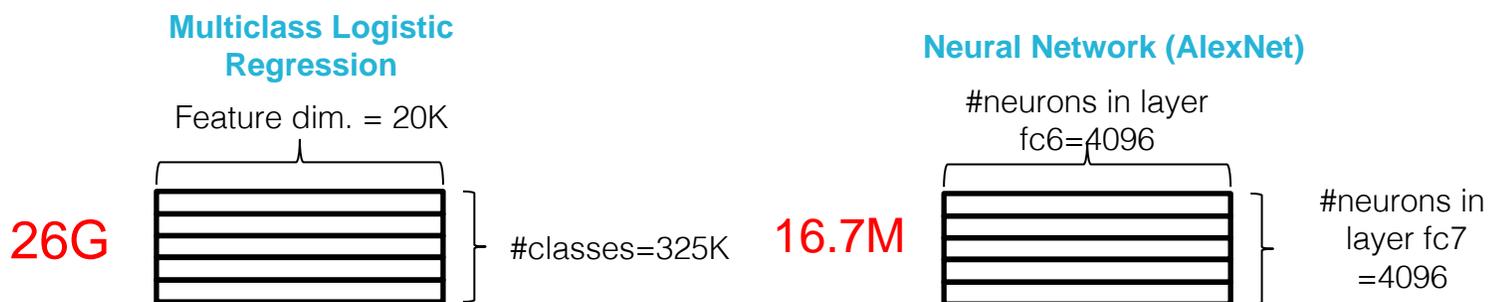


- Poseidon平台
 - **增强**现有深度学习引擎，**兼容**现有平台编程接口
- Poseidon的通信技术
 - 无等待的反向传播 (WFBP): 通过将通信与计算时间重叠来隐藏通信开销
 - **混合通信模型 (HybComm): (无损地)减少网络传输流量**



混合通信模型：充分参数广播(SFB)

- 模型参数矩阵



- 很多参数矩阵的数学性质

- 整个矩阵的参数更新 ΔW 分解为两个向量的外积 uv^T (u, v 叫做充分参数)

$$\min_W \frac{1}{N} \sum_{i=1}^N f_i(Wa_i; b_i) + h(W)$$



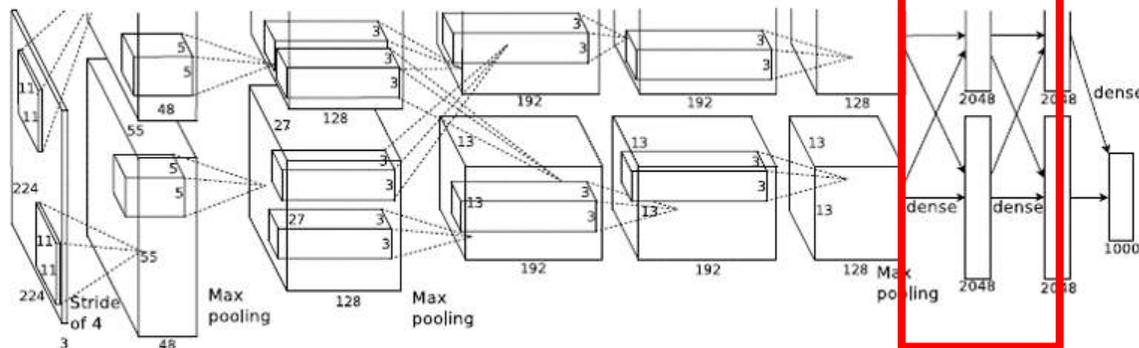
$$\Delta W = uv^T$$



混合通信模型：充分参数广播(SFB)

- 思路: 发送轻量级的充分参数 (u,v) , 而不是昂贵的全矩阵 ΔW !
- 案例: AlexNet CNN model
 - 全连通层参数大小: FC6 = 4096 * 4096 矩阵(16.7M 个参数)
 - 使用SFB通信
 - 分解为两个 4096 向量: u, v
 - 发送两个向量
 - 之后重建梯度值矩阵

Figure from Krizhevsky et al. 2012

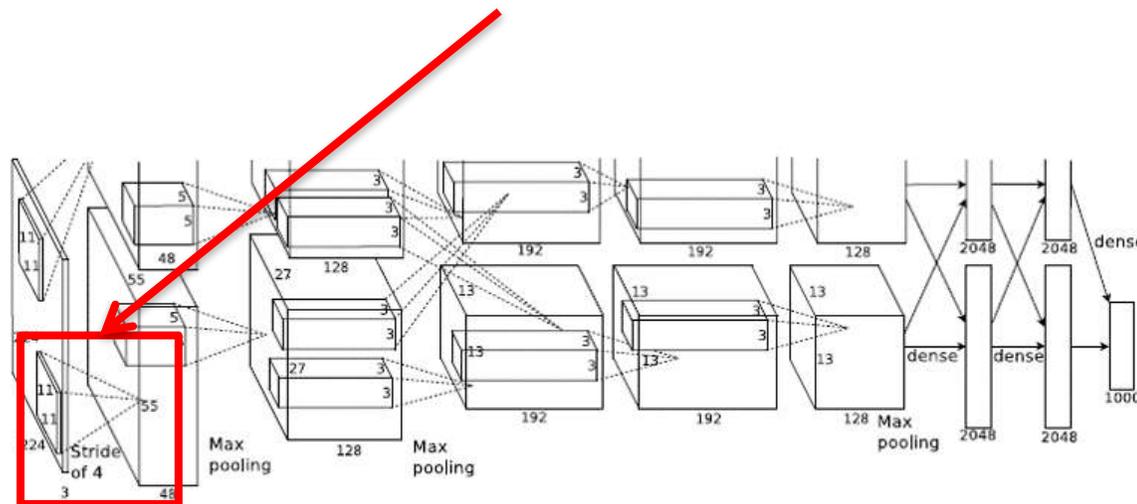




混合通信模型：充分参数广播(SFB)

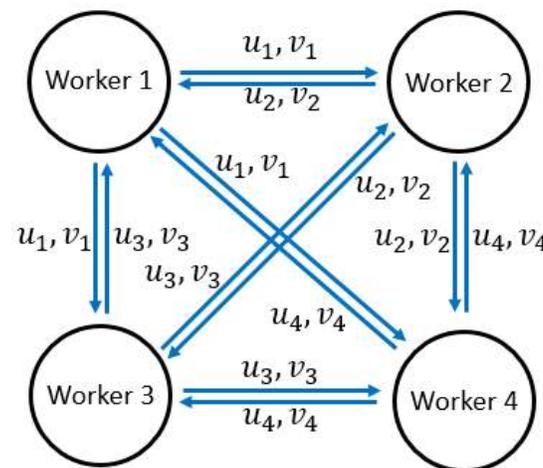
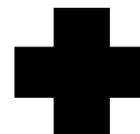
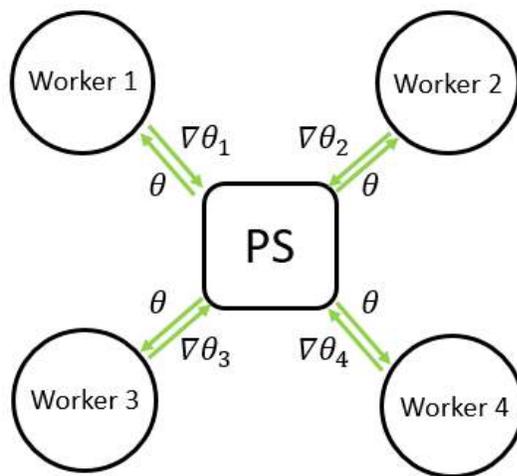
- 有时，充分参数广播并不划算
- 案例: AlexNet CNN model
 - 卷积层参数大小， $11 * 11$ 矩阵(121 parameters)
 - 通信量减小不显著，且增加充分参数计算开销

Figure from
Krizhevsky et al. 2012





- 思路
 - 使用SFB同步全连通层
 - 使用Ps同步卷积层
- 有效性
 - 在很多情况下直接减少通信量
- SFB总是最优吗？
 - 不，它的通信开销随机器数量进行平方式增长
 - 正确的策略：在更高效的情况下使用Ps通信





- 如何选择？我们可以进行定量分析
 - 层的类型: 卷积层CONV 还是 全连通层FC?
 - 层的大小：神经元数量M, N
 - 批量大小：Batch size K
 - 集群中计算节点的数量： P_1, P_2

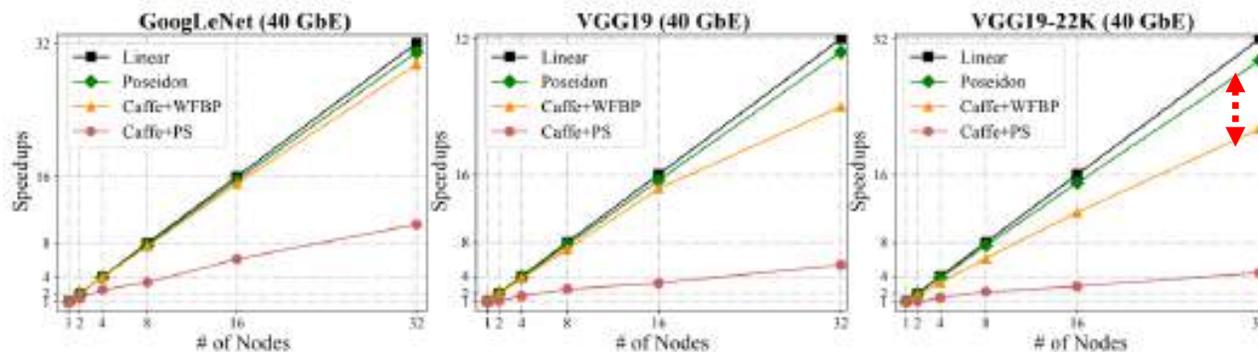
Method	Server	Worker	Server & Worker
PS	$2P_1MN/P_2$	$2MN$	$2MN(P_1 + P_2 - 2)/P_2$
SFB	N/A	$2K(P_1 - 1)(M + N)$	N/A
Adam (max)	$P_1MN + P_1K(M + N)$	$K(M + N) + MN$	$(P_1 - 1)(MN + KM + KN)$

Table 1: Estimated communication cost of PS, SFB and Adam for synchronizing the parameters of a $M \times N$ FC layer on a cluster with P_1 workers and P_2 servers, when batchsize is K .



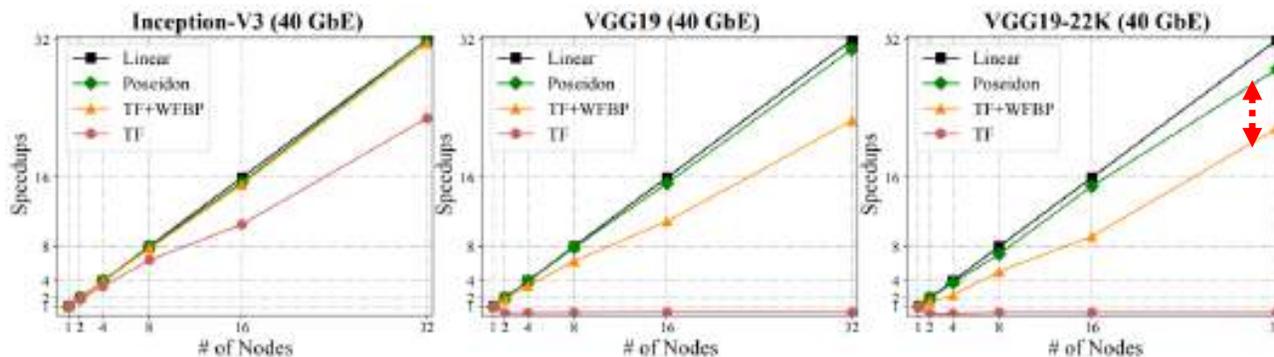
混合通信模式

- 性能怎么样？
 - On Caffe:



性能比WFBP更进一步

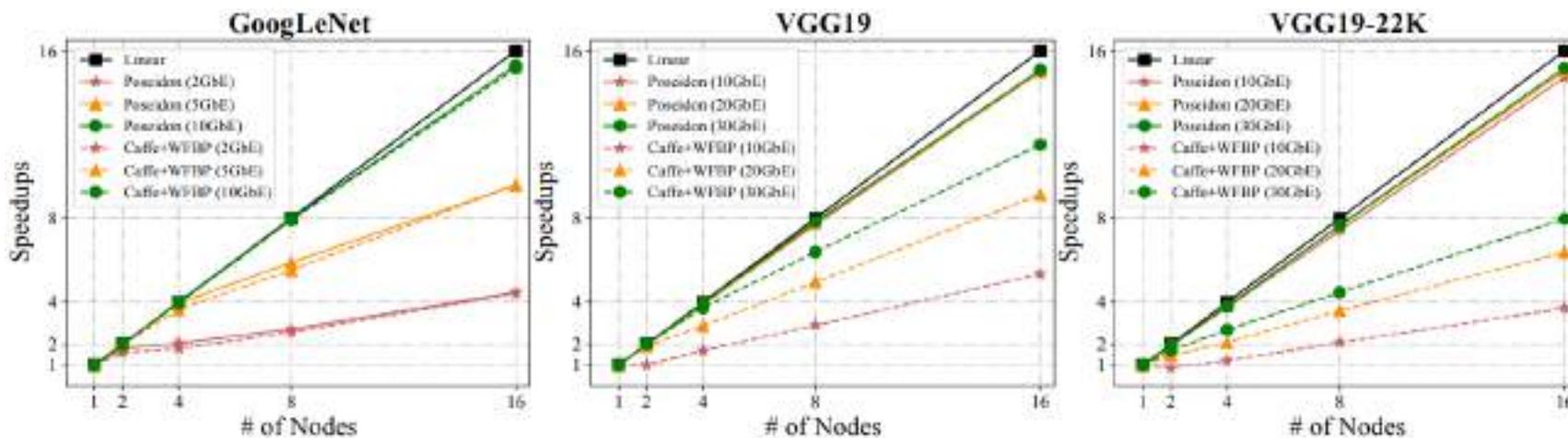
- On TensorFlow:



性能比WFBP更进一步



- 更重要的是, 即使在带宽有限的情况下, 也达到了计算吞吐量的线性扩展!
 - 使用常用硬件也可以进行快速深度学习模型训练
 - 降低分布式机器学习部署成本





总结: 值得记住的几点

- 通信是基于GPU的分布式深度学习的瓶颈
 - GPU很快
 - Ethernet带宽很有限
 - 突发式的通信特征
- Poseidon解决通信瓶颈
 - 无等待反向传播WFBP: 将计算与通信流水线化
 - 混合通信: 动态选择更高效的通信方式
 - 在多个数据集、模型大小、硬件条件下, 达到性能的线性拓展

Zhang, H., Zheng, Z., Xu, S., Dai, W., Ho, Q., Liang, X., Hu, Z., Wei, J., Xie, P. and Xing, E.P., 2017. **Poseidon:** An Efficient Communication Architecture for Distributed Deep Learning on GPU Clusters. USENIX ATC 2017



THANKS



- Petuum Inc.
 - 打造新一代企业级人工智能解决方案
- Petuum Poseidon是一个平台
 - **增强**现有深度学习引擎
 - **兼容**现有引擎编程接口
 - Poseidon更快速可以运行现有模型代码
- Poseidon-DyNet
 - 下一代深度学习引擎，支持动态图
 - 性能比同类平台**快数十倍**
 - <https://github.com/clab/dynet>
 - <http://dynet.io>

引擎&
工具集

平台

