一、通信时间优化

【论文】Campos V, Sastre F, Yagües M, et al. Scaling a Convolutional Neural Network for Classification of Adjective Noun Pairs with TensorFlow on GPU Clusters[C]// Ieee/acm International Symposium on Cluster, Cloud and Grid Computing. IEEE, 2017:677-682.

【面临问题】

针对GPU集群，优化节点与参数服务器的通信效率。

【解决方案】

提出了节点内4个GPU同步更新，节点间异步更新的思想，利用TensorFlow进行了实验验证。

【论文】Du X, Tang J, Li Z, et al. Wheel: Accelerating CNNs with Distributed GPUs via Hybrid Parallelism and Alternate Strategy[C]// ACM, 2017:393-401.

【面临问题】

分布式深度学习中，Worker和PS之间需要传递大量的数据，从而导致极长的通信传输时间以及GPU闲置时间，本文旨在减少通信时间、提高GPU利用率。

【解决方案】

* 提出一种数据并行与模型并行的并行模式

1. 根据观察可知，神经网络的参数很大一部分为全连接层参数
2. 本文在数据并行下，将卷积层部署在计算节点的GPU，全连接层部署在参数服务器端的GPU，重定义为参数中心，因而只有卷积层部分的参数需要在网络中传输，大大减少通信量

* 提出GPU内双Worker运行的调度算法

1. 工作节点内每个GPU建立两个训练worker，当一个worker等待接收消息被阻塞时，另外一个worker立即开始计算
2. 由于全连接层的前向和后向计算时间与参数更新时间之后远小于卷积层的前向和后向计算时间，所以参数中心可以兼顾

【论文】Li M, Andersen D G, Smola A, et al. Communication efficient distributed machine learning with the parameter server[J]. Advances in Neural Information Processing Systems, 2014, 1:19-27.

【面临问题】

三代之前的PS框架面临着以下两大问题：

1. 工作节点频繁读取或写入全局共享的参数信息，庞大的数据交互量带来了巨大的通信瓶颈
2. 许多学习算法是有时序性的，要求在众多计算节点间采用同步更新的模式，因而不可避免地会存在掉队节点拖慢集群速度，成为了主要瓶颈之一

【解决方案】

* 设计了一个三代PS框架，比现有框架更好地解决了异步更新的inconsistency问题

1. 将整个训练任务拆分为多个tasks，引入task之间的依赖机制，来丰富算法逻辑
2. 根据存在的依赖关系对tasks构造计算图，最终图的类型会因为依赖的不同呈现三种情形，Sequential（完全顺序执行）、Eventual（无依赖）和Bounded delay（部分有依赖）
3. 各种用户自定义filter，用于对计算得到的参数信息进行筛选、转换、引入随机性、cache缓存以及压缩等操作

【论文】P. Sun, Y. Wen, T. N. B. Duong and S. Yan, "Timed Dataflow: Reducing Communication Overhead for Distributed Machine Learning Systems," 2016 IEEE 22nd International Conference on Parallel and Distributed Systems (ICPADS), Wuhan, 2016, pp. 1110-1117.

【面临问题】

许多分布式机器学习系统在处理大数据集时表现出很高的通信开销。现在的主流的分布式ML系统在通信上花费的时间比计算花费的时间多很多。

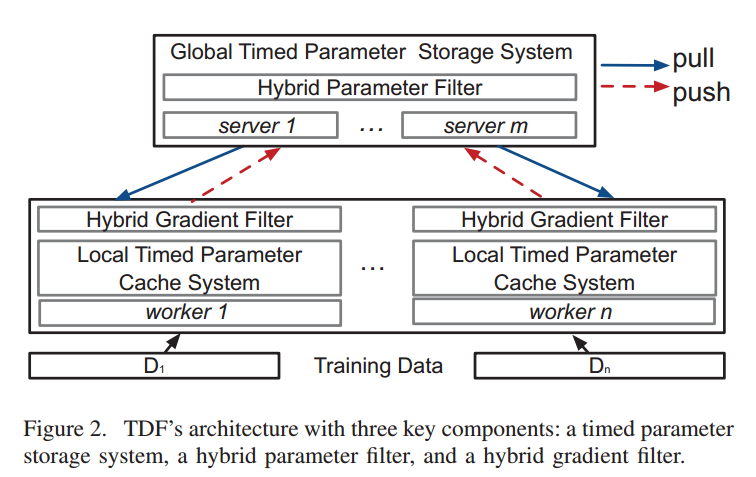
【解决方案】

这么高的通信开销主要是由于拉参数和推梯度两种操作引起的。在论文中，提出了一种定时数据流（TDF）的方法。在TDF方法中使用三种技术来减少通信开销：

1. 定时参数存储系统，使用两个子系统来管理带有一个版本号的参数:一个是服务器节点上的全局参数存储系统，另一个是工作节点上的一个本地参数缓存系统。
2. 混合参数过滤器，用于过滤服务器节点上的参数
3. 混合梯度过滤器，用于过滤工作节点上的梯度

总的来说，定时参数存储系统和混合参数过滤器使服务器能够在拉操作期间丢弃未改变的参数，混合梯度过滤器允许服务器在推操作期间选择性地降低梯度。

相应的构架如下：



【论文】F. N. Iandola, M. W. Moskewicz, K. Ashraf and K. Keutzer, "FireCaffe: Near-Linear Acceleration of Deep Neural Network Training on Compute Clusters," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 2592-2600.

【面临问题】

高精度的深度神经网络（DNN）的训练时间长，阻碍了对新的DNN体系结构的研究，并且使得高精度的DNN的开发速度减慢。在改进和加速DNN训练上，分布式算法的速度和可扩展性总是受到服务器之间通信开销的限制，那么怎样尽可能减少通信开销，同时不降低DNN模型的精度就是需要考虑的问题了。

【解决方案】

针对上面提出的问题，作者提出了几点解决方法：

1. 在GPU服务器之间实现高带宽的网络硬件，比如Infiniband或Cray连接器产品。
2. 考虑更好的通信算法，文中主要介绍了reduction trees算法，是一种和PS不同的策略，并且随着PS通信开销随着工作节点数线性增长时，reduction trees的通信效率更好。
3. 将并行化策略运用到高精度的DNN框架中，增加batch-size的大小，来减少DNN训练期间的总通信量，并且确定超参数，让我们在训练大的batch-size的同时保证了小批量的精度。

在对GoogLe Net和ImageNet在128个GPU集群上训练时，分别实现了47倍和39倍的加速。

【论文】Cong, Guojing, O. Bhardwaj, and M. Feng. "An Efficient, Distributed Stochastic Gradient Descent Algorithm for Deep-Learning Applications."International Conference on Parallel Processing IEEE Computer Society, 2017:11-20.

【面临问题】

ASGD（asynchronous stochastic gradient descent）是由SGD推导出来的，因为它能容忍陈旧的参数更新来减少同步和通信开销，所以应用很广。但是在理论分析中对ASGD性能至关重要的通信开销和实际学习速率会被掩盖，所以作者提出了俩个问题：

1.怎样能用实例来分析ASGD的通信性能和收敛性能

2.是否可以提出其他更好的算法比ASGD有更好的通信和收敛性能

【解决方案】

1. 在论文中作者用Downpour算法为例来评估ASGD的实际效率。用两个不同的深度学习模型（CIFAR-10和NLC-F），但是都是用同样的训练算法。然后通过一系列的实验来分析ASGD的实际通信和收敛性能。
2. 作者提出了一种批量同步分布式SGD算法（SASGD），SASGD在ASGD的基础上，减少了陈旧程度受到的约束，在SASGD中只受聚合间隔（T）约束，降低了每个梯度聚合传输的数据量的大小。实验结果显示SASGD的效果相比ASGD更好。

【论文】Zhang S, Choromanska A, Lecun Y. Deep learning with Elastic Averaging SGD[J]. 2015:685-693.

【面临问题】

在通信约束条件下实现深度学习并行计算的随机优化问题。

【解决方案】

提出了一种算法，使本地worker能够进行更多的探索，即允许变量可以发散性的从中心值进行探索，并同时减少本地worker与master之间的交互。该方法与常见的ADMM比较，得到更快的加速效果，通信效率也很快。算法中还提及了基于梯度动量的版本。

【论文】Iandola F N, Moskewicz M W, Ashraf K, et al. FireCaffe: Near-Linear Acceleration of Deep Neural Network Training on Compute Clusters[J]. 2015, 37:2592-2600.

【面临问题】

分布式算法的速度和可扩展性总是受到服务器之间通信开销的限制，如何减少通信开销并保证训练的DNN模型的精度不降低。

【解决方案】

提出了新的方法，其中有三个关键的地方。首先，选择在GPU服务器之间实现高宽带的网络硬件。其次，考虑其他的通信算法，发现reduction tree比参数服务器方法更加高效并且可扩展性也更好。第三，选择增加批量大小以减少DNN训练期间的通信总量，并且确定超参数，使我们能够在大批量训练的同时实现小批量的准确性。

【论文】Wang J, Kolar M, Srebro N, et al. Efficient Distributed Learning with Sparsity[J]. 2016.

【面临问题】

提高大规模并行计算中由于通信频繁带来的冗余开销。

【解决方案】

提出了一种新的，高效的分布式稀疏学习方法，该方法利用机器间的随机分块观测进行分布式稀疏学习。在每一轮迭代中，worker节点计算数据loss值的梯度，master采用L1正则化进行loss值最小化的计算。

【论文】Cho M, Brand D. MEC: Memory-efficient Convolution for Deep Neural Network[J]. 2017.

【面临问题】

卷积已是当下深度神经网络中的关键组件，但无论何种卷积方法，都带来了巨大的内存开销，从而导致性能下降并且通常的做法是在性能和内存损耗之间提供较差的折中。

【解决方案】

提出了一种内存高效的卷积（MEC），能够减少内存的开销，实质上加速了卷积的过程，MEC以一种有效的方法减小了输入的矩阵（降低内存开销），然后并行地执行多个小矩阵乘法以此完成卷积操作。MEC可以显著降低内存消耗，同时提高加速性能。

二、梯度评估与版本控制

【论文】Jiang J, Cui B, Zhang C, et al. Heterogeneity-aware Distributed Parameter Servers[C]// ACM International Conference. ACM, 2017:463-478.

【面临问题】

在分布式深度学习中，由于现实生活中集群内的各个节点难免性能不一，存在掉队节点拖慢整体计算速度，甚至会造成梯度退化的现象。

【解决方案】

提出了2种算法：

1. ConSGD：在原方案的基础上，更新参数值时，将各个计算节点上传的梯度值乘以系数λ，可以证明取0.5时最好，即w→w+λu
2. DynSGD：引入参数版本控制的概念以及陈旧度（staleness）的评判标准，在进行参数汇总时，陈旧度较高的梯度权值会成反比例减小，以此来减少旧版本参数对训练收敛速度的影响

【论文】Asynchronous Parallel Greedy Coordinate Descent

【面临问题】

现有的异步梯度下降方法由于其机制的问题，导致模型准确率收敛比较慢。

【解决方案】

提出了一种对有界约束的光滑函数最小化的异步并行贪心坐标下降算法。最后实验证明该算法具有良好的加速性能，并且优于现有的多核支持向量机的解决方案。

【论文】Chen C Y, Choi J, Brand D, et al. AdaComp : Adaptive Residual Gradient Compression for Data-Parallel Distributed Training[J]. 2017.

【面临问题】

高性能计算平台上的大规模分布式深度神经网络训练受到严重的通信限制。为了克服这个限制，需要新的梯度压缩技术。

【解决方案】

介绍一种新技术——自适应残差梯度压缩方法，AdaComp可以基于本地的梯度残差，并根据本地的活跃性调整压缩率，从而自适应地进行梯度压缩。并且证明该方法在全连接层和复发层的压缩率达到200倍，卷积层的压缩率达到40倍，伴随着梯度的压缩，模型的准确率没有出现下降。

【论文】Lian X, Zhang H, Hsieh C J, et al. A Comprehensive Linear Speedup Analysis for Asynchronous Stochastic Parallel Optimization from Zeroth-Order to First-Order[J]. 2016.

【面临问题】

异步并行优化可以给我们带来多大的加速收益。

【解决方案】

提供一个全面而通用的分析，以研究从零阶到一阶方法的各种异步并行随机算法的加速性能，其结果改善了对特殊情况的现有分析，为理解异步并行行为提供了更多见解，并首次提出了一种新颖的异步并行零阶方法。

【论文】Han J, Liu Q. Bootstrap Model Aggregation for Distributed Statistical Learning[J]. 2016.

【面临问题】

在分布式或隐私保护学习中，需要利用一组概率模型估计来自不同的本地存储库，并要求将它们合并成一个模型，以提供有效的统计估计。一种简单的方法是对局部模型的参数进行线性平均，然而，对于非凸模型或具有不同参数维度的模型，这些参数趋向于退化或不适用。另一个更实用的策略是从局部模型生成自举样本，然后基于组合自举集合学习联合模型。 不幸的是，自举程序引入了额外的噪音，并可能显著恶化性能。

【解决方案】

提出了两种方法来降低自举噪声，其中包括一个加权的M估计器，其在统计上是有效的并且效果十分明显。

【论文】Reddi S J, Hefny A, Sra S, et al. On Variance Reduction in Stochastic Gradient Descent and its Asynchronous Variants[J]. Evidence Report/technology Assessment, 2016, 31(183):2647-2655.

【面临问题】

研究基于方差的随机梯度下降算法，通过SAG,SVRG,SAGA等算法的发展，这些算法已经被证明在理论上和实践上都优于SGD。然而这些算法的异步版本还没有得到广泛研究。缺少统一的实验框架是主要问题。

【解决方案】

提出了一种基于框架的异步算法，并证明了它的快速收敛性。

【论文】Jiang J, Cui B, Zhang C, et al. Heterogeneity-aware Distributed Parameter Servers[C]// ACM International Conference. ACM, 2017:463-478.

【面临问题】

在异构环境下的分布式机器学习，现有的分布式随机梯度下降系统在同构环境中工作得很好，但在异构环境中，他们的同步协议不适合异构设置，所以掉队者很常见，因此它们的性能可能会下降，有时甚至达到10%。

【解决方案】

提出一种异构感知算法，它在进行全局更新前不间断地进行训练，抑制陈旧节点对收敛性的影响。以及考虑到每次更新的陈旧信息采用更为复杂的学习速率自适应机制。理论上证明该两种方法的有效融合。比原型系统快了2-12倍。

三、CPU与GPU协作

【论文】Cui H, Zhang H, Ganger G R, et al. GeePS: scalable deep learning on distributed GPUs with a GPU-specialized parameter server[C]// Eleventh European Conference on Computer Systems. ACM, 2016:4.

【面临问题】

众所周知GPU比CPU更加适合深度学习训练任务，然而只有当参数和数据存储在GPU内存时才会有加速效果，然而GPU内存较小，无法存储大规模深度网络的参数即全部的训练数据。

现在的参数服务器框架大多是针对CPU环境来设计，扩展性较强，但是不适合用作GPU集群。

【解决方案】

* 利用CPU与GPU间的协作，保持参数信息存储在GPU内存中
* 利用CPU与GPU的协作，在后台动态加载训练数据至GPU内存，并且在节点内部进行参数传输时，只传输参数的指针而不是整个参数信息，合理管理GPU内存

最终详细介绍了GeePS整个系统的实现逻辑以及设计的API。

四、PS配置优化

【论文】Yan F, Ruwase O, He Y, et al. Performance Modeling and Scalability Optimization of Distributed Deep Learning Systems[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2015:1355-1364.

【面临问题】

在分布式深度学习中，针对特定的神经网络结构，很难决定最佳的分布式集群的配置参数，这些参数的每种组合可能都会带来很大的性能差异。

现在主要的解决方法为网格搜索法，即将参数进行排列组合并且一一做实验，十分耗时。

【解决方案】

* 开发了一个Performance Model，量化configuration不同对最终时间的影响

1. 详细讲述了在模型并行的情况下，估算各部分运行时间（前向、后向和权重同步，包括其中通信）的方法和公式

2. 在数据并行下，阐述了三种数据并行模式下epoch计算时间的估算方法

* 创建Optimizer搜索最佳System Configuration

1. Configuration包括决定有多少资源的参数（PS数、Replica数、Worker数）、决定DNN每层的分区数线程数、资源到任务标识（某一层某一分区）的映射函数

2. 4个限制：节点的总数、每个节点可以同时运行的线程数、每层两个不同的Segment不会被映射到同一个Worker、确保利用了芯片级别的并行

3. 搜索算法：贪心算法决定每个Worker对应哪个Segment、利用动态变成构建优化问题减少layer异地放置的花销

【论文】Johnston T, Young S R, Hughes D, et al. Optimizing Convolutional Neural Networks for Cloud Detection[C]// the Machine Learning. 2017:1-9.

【面临问题】

随着深度学习的日益流行以及其带来的强大的经济效益，最大的挑战就是如何针对应用场景合理地设计神经网络，包括网络的嵌套结构以及超参的设置，一种简单的方法是随机搜索，但是比较费时。

【解决方案】

提出两阶段搜索法减少耗时：

1. 第一阶段是在可行的网络配置下进行一个大规模并行化随机搜索
2. 第二阶段用第一阶段的结果，训练一个SVM模型，并根据模型的输出结果知道第二阶段的大规模并行化搜索，最终得到最佳超参配置

五、PS稳定运行优化

【论文】Li M, Tanimura Y, Nakada H. A quantitative analysis of fault tolerance mechanisms for parallel machine learning systems with parameter servers[C]// International Conference on Ubiquitous Information Management and Communication. ACM, 2017:69.

【面临问题】

在大型深度学习集群中，大规模的数据并行模式对集群的容错模式是一个很大的挑战。然而，虽然已经有了许多关于一般化集群的容错机制研究，但基于深度学习并行化环境，关注的人较少，暂时没有太多行之有效的办法。

【解决方案】

利用参数服务器框架，为分布式深度学习系统提供额外的冗余，使得参与检查点记录的节点变为以前的两倍，并且针对集中容错方法在分布式环境中进行实验验证，评估各个方法的有效性。

六、去中心化架构

【论文】Wang M, Zhou H, Guo M, et al. A scalable and topology configurable protocol for distributed parameter synchronization[M]. 2014.

【面临问题】

现有分布式深度学习系统中，普遍存在可扩展性较差以及由参数版本不一致导致的收敛震荡的问题，单节点通信瓶颈问题的存在也十分不利于系统的进一步扩展。

【解决方案】

* 提出一种去中心化异步参数共享协议，包括多种一致性层次

1. 每一个Worker需要根据别的Worker传输的梯度信息和自己的梯度信息，来更新自己的参数版本
2. 一致性层次包括强同步和弱同步（最慢和最快节点间差距在给定的迭代数之内）
3. 证明了工作节点间通过多次参数交换可以保证各节点间参数一致

【论文】Johnston T, Young S R, Hughes D, et al. Optimizing Convolutional Neural Networks for Cloud Detection[C]// the Machine Learning. 2017:1-9.

【面临问题】

PS框架中，为了确保模型收敛，需要很好地分配Worker和PS之间的工作量，保证不出现瓶颈，在实际场景下，需要手工对模型配置甚至是硬件设备进行调整。

【解决方案】

提出基于数据流的去中心化Ako系统，所有的节点都作为Worker全力利用CPU资源进行计算。Worker之间尽可能频繁地相互交换梯度信息，经过几轮交换后，最终达到同步的效果，不影响收敛。

Ako不需要任何分配资源的操作，最终的收敛速度甚至比PS框架下的收敛速度更快。

七、分布式学习系统和框架的开发

【论文】Qi Y A, Qi Y A, Qi Y A, et al. KunPeng: Parameter Server based Distributed Learning Systems and Its Applications in Alibaba and Ant Financial[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2017:1693-1702.

【面临问题】

现有的分布式集群系统，均不能很好地满足公司内部关于需求。如经典的Hadoop和Spark，只支持强同步模式而且Map和Reduce封装较好难以进行修改，不容易适应多变的业务场景。MPI框架，灵活性较强，可以实现各种需求，但是编码难度较大耗时较长且容易造成很多bug，不利于工作效率的提升。

【解决方案】

实现了鲲鹏系统，将分布式集群管理与深度学习算法集合在一起，支持数据并行与模型并行、负载均衡、异步更新机制、稀疏数据处理、容错机制等，先已完美应用于各个业务场景中。

本文创新性地在PS framework下提出了Coordinator这一角色，负责存储计算过程，负责构造计算图并将任务分配给各个Worker和Server。其余部分主要是工程性讲解。

【论文】Xing E P, Ho Q, Dai W, et al. Petuum: A New Platform for Distributed Machine Learning on Big Data[J]. IEEE Transactions on Big Data, 2015, 1(2):49-67.

【面临问题】

无论是各种机器学习的库还是Spark MLib，都无法很好地适应现在分布式机器学习任务的需求，急需一个包容性较好的普适性平台改善现有局面。

【解决方案】

设计了Petuum系统，支持模型并行、数据并行等主流模式，有丰富的集群管理机制、容灾机制，实现了大部分公认效果较好的机器学习算法。

【论文】Zhang Y, Li J, Sun C, et al. HotML: A DSM-based Machine Learning System for Social Networks[J]. Journal of Computational Science, 2017.

【面临问题】

在社交网络分析方面，很多的机器学习算法被采用。但是数据集很大，在一个机器学习系统上训练一个模型需要很长的时间。作者提出应该考虑性能问题来促进训练的过程。

【解决方案】

1. 设计了一个机器学习系统：HotML。HotML是在参数服务器构架（PS）下设计的。
2. 在HotML中，作者在DSM（Distributed shared memory）的基础上结合参数服务器框架（PS）设计了一个新的框架（DPS），在DPS中的PS组件将PS中的服务器和工作节点分离。
3. 在HotML中，作者设计了改良的SSP的版本：SSPPush和SSPDrop，利用空闲的网络带宽来push全局参数给工作节点来减少SSP的等待时间。
4. 设计了一种灵活的工作端和一致的服务器端检查点机制，来提高容错性

【论文】S. Gupta, W. Zhang and F. Wang, "Model Accuracy and Runtime Tradeoff in Distributed Deep Learning: A Systematic Study," 2016 IEEE 16th International Conference on Data Mining (ICDM), Barcelona, 2016, pp. 171-180.

【面临问题】

大量参数的深度学习要分布式的进行训练，其中模型精度和运行时间是需要考虑的重要因素，但是在模型训练的过程中，并没有研究过这两个因素之间的权衡。

【解决方案】

本文提出了一个基于参数服务器的分布式计算框架，Rudra，引入了一种新的学习速率调制策略来克服过时梯度的影响，提出了一种新的同步协议，可以有效地限制梯度的过时性，提高运行时性能，达到较好的模型精度。在Rubra中主要做了下面工作：

1. 一种新的学习速率调整策略，可以降低过时参数更新的影响。
2. 一个新的同步协议，以减少网络带宽开销，同时实现良好的分类精度和运行时性能。
3. 一种系统技术（vector clock）来量化梯度下降参数更新过时，提到了时间戳来进行标记更新。

研究了训练算法的超参数（最小批量，学习率（梯度下降步长））与分布式实现参数（梯度陈旧度，学习者数量）之间的相互依赖关系对神经网络分类精度和训练时间的影响。

【论文】Y. Li, H. Wan, B. Jiang and X. Long, "More Effective Synchronization Scheme in ML Using Stale Parameters," 2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), Sydney, NSW, 2016, pp. 757-764.

【面临问题】

在机器学习中，我们使用的模型变得越来越重要。我们通过迭代处理训练数据来调整模型参数直到收敛。尽管在同步模型参数以实现并行性时，数据并行机器学习系统通常会有一个很好的容错，但是模型参数的同步可能会有延迟，这在大规模上会变得更糟糕。那么怎样减少参数同步的延迟时作者思考的问题。

【解决方案】

提出了一种有界异步并行模型（BAP）。在BAP中，允许分布式的工作节点使用存储在本地缓存中的陈旧参数来进行更新，而不是必须等到PS产生新的版本，这就明显减少了工作节点的等待时间，减少了同步开销，同时BAP模型还通过限制陈旧参数的最大的间隔来保证机器学习算法的收敛。

【论文】 Yue Yu; Jinrong Jiang; Xuebin Chi Using Supercomputer to Speed up Neural Network Training 2016 IEEE 22nd International Conference on Parallel and Distributed Systems (ICPADS) Year: 2016 Pages: 942 - 947

【面临问题】

深度学习方面的研究表明，大型模型可以显著提高性能。对于大型模型怎么提高性能？

【解决方案】

在文中提出了使用多个GPU的深度网络训练，提出了基于Caffe的框架（Caffe-HPC）。

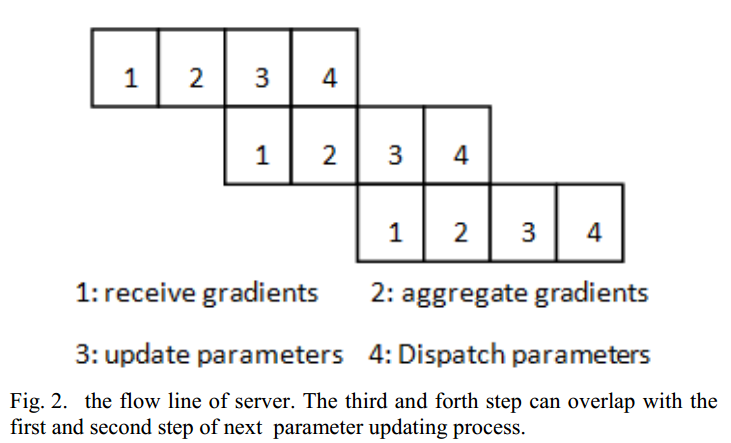
Caffe-HPC侧重于数据并行性，并且使用模型的多个副本来优化单个目标。 每个模型副本都使用GPU或多个CPU核心进行训练，并且使用参数服务器来同步许多分布式渐变更新的整体效果。

Caffe-HPC是在MVAPICH2 MPI实现之上构建的。

在论文中，作者提到了几种方法来减少参数通信开销：

1. 在硬件方面，使用56Gb FDR Infiniband通信，它比典型的以太网速度快得多。而使用GPUDirect RDMA 技术来减少GPU和CPU之间的内存拷贝。
2. 在算法方面，设计PS的算法，过程如下;
3. 接受工作节点的梯度，保存在缓冲区;
4. 聚合到渐变缓冲区;
5. 使用梯度下降算法更新参数;
6. 向工作节点发送参数;

服务器的流水线如下：



【论文】P. Sun, Y. Wen, N. B. D. Ta and S. Yan, "Towards Distributed Machine Learning in Shared Clusters: A Dynamically-Partitioned Approach," 2017 IEEE International Conference on Smart Computing (SMARTCOMP), Hong Kong, 2017, pp. 1-6.

【面临问题】

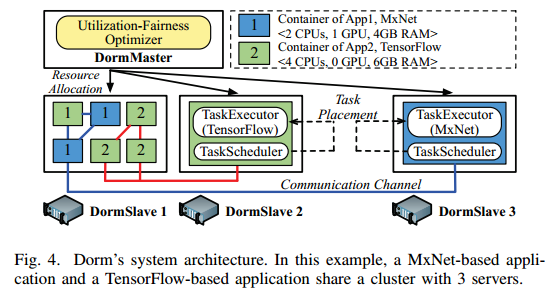
集群管理系统建议多种分布式计算系统共享一个集群。但是根据高资源利用率，资源公平分配，低共享开销的标准，现在就是分布式机器学习的工作负载问题。

【解决方案】

1. 一个新的集群管理系统Dorm，处理在共享集群中多种分布式机器学习的工作负载问题。Dorm结构图如下：

DormMaster：管理所有的集群资源，使用容器来分片集群，每一个应用一个分片。后面的The utilization-fairness optimizer 是在DormMaster中决定资源分配的。

DormSlave：管理一个集群服务器的本地资源，报告一个集群服务器中可利用资源的数目给DormMaster，同时使用容器来给多种应用程序共享一个集群服务器。



1. 两种技术，一是 Dynamically-Partitioned Resource Management，主要思想是调整现有资源来实施新的资源分配，具体做法是在个别的服务器上创建和销毁容器。
2. 另一个是The utilization-fairness optimizer。一种公平利用的优化

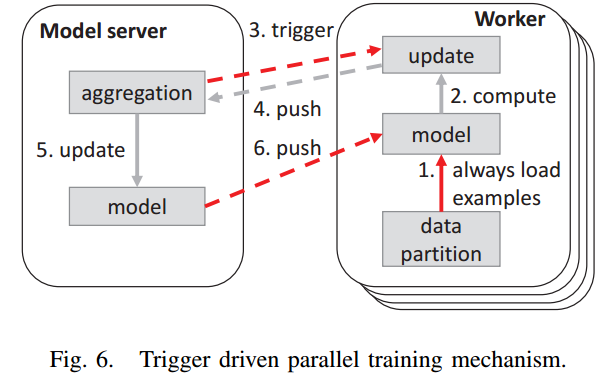
【论文】Li, Shenglong, et al. "Efficient Distributed Machine Learning with Trigger Driven Parallel Training." Global Communications Conference IEEE, 2017:1-6.

【面临问题】

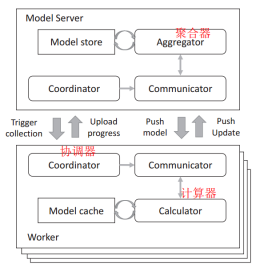
为了减轻掉队节点的干扰，现在分布式机器学习系统允许worker使用本地陈旧的参数来进行更新，而不需要等待最新的参数，并且在一定范围内限制异步步骤以保证算法的正确性。但是有限制的异步计算不能忍受一直掉队的节点。论文中的作者解释这个问题的根本原因是基于worker驱动的并行训练模式。

【解决方案】

1. 为了解决上面提出的问题，作者设计出了一种新的并行训练的模式——基于触发器驱动的并行训练模式。在基于触发器驱动的并行训练模式中，模型服务器主动触发，从worker节点那里收集更新的参数，而不是被动的接收。如图：



1. 提出了一种动态负载平衡策略，让每个数据的采样频率相等，来避免模型的偏移。
2. 将上面两个创新点整合到一个分布式机器学习系统中——Squirrel。
3. 在Squirrel中，作者在worker和Model server中都加了一个Coordinator.，用来控制触发器信息，具体如图：



【论文】Scardapane S, Dilorenzo P. A Framework for parallel and distributed training of neural networks[J]. Neural Networks, 2017, 91:42–54.

【面临问题】

现今对神经网络在分布式环境中训练的框架有很多，但是没有一个通用的框架。

【解决方案】

提出了一个在全分布式情况下训练一般神经网络模型的算法框架，是一种基于网络的非凸优化设计框架，它依赖于一种(原始的)凸化分解技术来处理非凸性，以及动态一致的过程。实验表明，与代理的通信拓扑结构、算法参数的选择以及optim的结构相比，所生成的算法是收敛于非凸问题的静态解决方案的。

【论文】Wang Y, Ding C, Li Z, et al. Towards Ultra-High Performance and Energy Efficiency of Deep Learning Systems: An Algorithm-Hardware Co-Optimization Framework[J]. 2018.

【面临问题】

实现深度神经网络硬件的超高效率和性能。

【解决方案】

开发了一种基于算法的硬件协同优化框架，适用于不同的DNN类型、大小和应用场景。

算法部分采用一般的块循环矩阵，实现精度和压缩比的细粒度权衡。当然，这个算法适用于全连接层和卷积层，并且对该方法有效性进行了严格的数学证明。这个算法将每层的计算复杂度和存储复杂度都降低，用于训练和推理。

硬件部分采用高效的现场可编程门阵列（FPGA）实现，采用有效的重构、批量处理、深度流水线、资源重用和层次控制。

【论文】Wang X, Dunson D, Leng C. DECOrrelated feature space partitioning for distributed sparse regression[J]. 2016.

【面临问题】

当样本容量或者数据集的尺寸非常大时，拟合统计模型在计算上是具有挑战性的。

【解决方案】

降低问题大小的一个方法：将数据集划分为子集，然后使用分布式算法进行匹配。数据集可以水平或者垂直地分区。现有的划分地方法容易受到高相关性的影响，或者在减少模型维度方面效率低下。

在本文中，提出了一个并行框架DECO来解决上面的问题，该框架用于分布式变量选择和参数估计。

在DECO中，变量首先被划分并分配给m个分布式工作节点。然后通过为高维问题设计的任何算法来拟合各个工作节点中的子集数据。通过合并步骤，DECO可以在几乎没有假设的情况下，在每个子集上实现一致的变量选择和参数估计。

八、策略和原则

【论文】Xing E P, Ho Q, Xie P, et al. Strategies and Principles of Distributed Machine Learning on Big Data[J]. Engineering, 2016, 2(2):179-195.

【面临问题】

在一个很庞大的分布式集群中，通常有很多的工程性工作，那么这样的工程是真正的属于ML研究领域吗？ 作者讨论了从最近的工业级ML解决方案上所做的一系列的原则和策略，并且通过剖析成功案例来介绍应该如何利用这些原则和策略来设计通用的ML框架。

【解决方案】

在论文中，作者主要提出了4种原则：

1. 如何分配：调度和平衡工作量
2. 如何连接计算和通信：桥接模型和有界异步
3. 如何通信：管理通信和拓扑
4. 通信的内容是什么？

作者还在论文中提到了很多的ML解决方案的策略和原理，比如：在ML程序中的负载平衡，计算的优先次序，并行化的结构，陈旧的同步并行等等。

最后，作者还提到了一个实例：Petuum系统。

九、针对具体业务场景或算法的优化

【论文】Gong T, Fan T, Guo J, et al. GPU-based parallel optimization of immune convolutional neural network and embedded system[J]. Engineering Applications of Artificial Intelligence, 2016.

【面临问题】

当代一些便携嵌入式系统中，许多应用都需要用到深度神经网络进行训练，然而嵌入式设备的计算性能相对PC而言差距较大，可能会造成训练时间过长、过拟合和准确率较低的影响。如何在嵌入式系统中提升深度学习算法的性能是一个值得研究的问题。

【解决方案】

创新性地提出了一种免疫机制来改善现有的卷积神经网络结构，叫做免疫卷积神经网络，并且借助CUDA编程，在嵌入式系统上通过多GPU并行实现了这一网络结构，取得了更高的准确率、更稳定的表现和更快的计算速度。

【论文】Cong G, Kingsbury B, Gosh S, et al. Accelerating deep neural network learning for speech recognition on a cluster of GPUs[C]// the Machine Learning. 2017:1-8.

【面临问题】

面对语音、文字、股票信息等大容量时序化数据，虽然RNN等深度学习网络可以取得较好的结果，但会遭遇运算时间过长等效率问题，而且这些模型很难通过有效的并行方式缩短计算时间，存在异步更新收敛度不高等瓶颈问题。

【解决方案】

针对大容量时序化数据的特点，设计了一个P-learner K-step的算法，对传统的异步梯度下降算法进行了改进，在4台GPU中取得了业界领先的效果。

【论文】Hu X, Huang J, Qiu M. A Communication Efficient Parallel DBSCAN Algorithm based on Parameter Server[C]// ACM, 2017:2107-2110.

【面临问题】

机器学习算法DBSCAN，经过MPI改造成分布式训练后，其性能远超其他机器学习算法。然而随着节点数的增加，通信的开销会以几何式增长，可扩展性较差。

【解决方案】

将非连通集数据结构与PS框架结合起来，最大程度上减小通信开销，利用图论的思想对树的分裂、剪枝等操作进行了优化。

【论文】Ordentlich E, Yang L, Feng A, et al. Network-Efficient Distributed Word2vec Training System for Large Vocabularies[C]// ACM International on Conference on Information and Knowledge Management. ACM, 2016:1139-1148.

【面临问题】

Word2vec是文本挖掘时较常用到的深度学习算法，与传统的文本分析算法相比，Word2vec的准确度大大提升，并且因为其属于无监督学习，所以省去了大量的人工标注所用的时间。然而当在很长的文本中进行主题分析时，运用Word2vec模型会导致计算任务较重，无法在单机环境中完成。

【解决方案】

针对Word2vec模型的特点，提出一种特定的并行算法，取得较好效果。

【论文】Jiang J, Jiang J, Cui B, et al. TencentBoost: A Gradient Boosting Tree System with Parameter Server[C]// IEEE, International Conference on Data Engineering. IEEE, 2017:281-284.

【面临问题】

GBT（Gradient boosting tree）作为一种业界领先的机器学习算法，在学术界和工业界都取得了很好的效果。然而现有框架对GBT的实现仍不太理想，当训练数据的维度过高时很容易出现梯度退化的现象。

【解决方案】

结合PS架构，也出一种分布式并行的方法，优化模型聚合的能力，具体步骤如下：

1、训练数据分配到各个Worker

2、树分裂节点形成下一代时，每个节点计算本地数据的梯度信息，对于每个特征，都需要建立一个独立的直方图

3、一个协调器将所有Worker的直方图汇总在一起，寻找最佳分裂特征和用于分类的值

4、协调器广播分裂结果，每个Worker根据广播结果进行更新，然后继续进行分裂

在参数服务器端，对树的分裂、剪枝等操作做了特别的优化。

【论文】Hu H, Ren Y, Xu X, et al. Reducing view inconsistency by predicting avatars' motion in multi-server distributed virtual environments[J]. Journal of Network & Computer Applications, 2014, 40:21-30.

【面临问题】

多服务分布式虚拟环境（DVE）系统包含一组服务器，来支持大量地理分布的用户。它提供了一个共享的虚拟环境，用户通过网络参与交互。有两个技术难题：由于网络延迟造成的不一致和由于用户分布不均匀导致的服务器过载。

【解决方案】

在论文中，作者提出了一个新的分区框架来解决这两个问题。通过概率模型预测用户的运动来观察一致性和工作负载平衡。

通过区域四叉树来对虚拟环境进行索引，提出了一种QD\_InitialPartition的算法来平均分配服务器之间相邻位置的用户，以平均每个服务器的计算工作量。

提出了一个名为Parallel\_UMCluster的基于密度的方法来帮助每个服务器通过计算每个角色的未来运动。

提出了一种称为LB\_Repartition的算法，在服务器之间重新分配未分配的头像，以减少DVE中的视图不一致性和工作负载平衡。

【论文】Wang Y, Dou Y, Liu X, et al. PR-ELM: Parallel regularized extreme learning machine based on cluster[J]. Neurocomputing, 2016, 173:1073-1081.

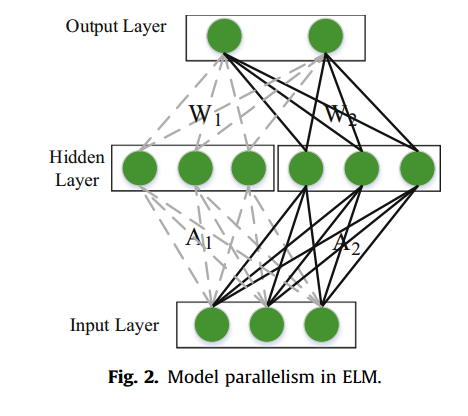
【面临问题】

极限学习机（ELM）由于其高效和易于实施，在过去的几年中得到了深入的研究。而且已经提出了许多的变体，例如PELM和OS-ELM，以提高其时序性能并使其能够增量学习。那么如何进一步提高ELM在处理大规模学习任务中的计算效率就值得思考了。

【解决方案】

提出了两种并行变体，统称并行正则化ELM（PR-ELM）。

1. 数据并行正则化ELM（DPR-ELM）：将整个数据集分成N个数据块，通过分别在每个数据块中训练的N个子模型来构建整个模型。
2. 模型并行正则化ELM（MPR-ELM）：用来处理样本数量小于隐藏节点数量的情况。模型并行的图如下：



最后通过PR-ELM，整个数据集或整个模型被分割成小块或子模型，因此它们大大减少了大量运行时间的需求，而且独立地训练每个数据块或每个子模型，所以它们具有更好的并行性。

【论文】Wang C, Liu C, Zhang Z H, et al. Minimizing the total completion time for parallel machine scheduling with job splitting and learning[J]. Computers & Industrial Engineering, 2016, 97(C):170-182.

【面临问题】

受到劳动密集型行业的实际情况的驱使，管理者需要决定怎样拆分和分配给并行工作的团队。这就涉及到了并行机器调度的问题，考虑到作业的拆分和学习的话，怎样能最大限度的缩短完成时间？

【解决方案】

由于要尽量减少工作的完成时间，就要尽可能多的拆分项目，然而拆分会对未来工作的完成时间产生负面影响，所以作者引入了一个贪婪算法来最好的平衡当前工作和未来工作的完成时间。

其中贪婪功能分为两个部分：第一部分是工作的完成时间，由工作分配带来的直接成本。第二部分是工作拆分带来的，是工作的间接成本。那么贪婪搜索就是使成本最小。

【论文】Khalil-Hani M, Shan S L, Bakhteri R. An Optimized Second Order Stochastic Learning Algorithm for Neural Network Training[J]. Neurocomputing, 2016, 186(C):74-89.

【面临问题】

论文主要就是介绍改进的B-SDLM算法。

【解决方案】

1. 论文中提出了一种改进的随机二阶学习算法用于有监督神经网络训练，称为 bounded stochastic diagonal Levenberg–Marquardt（B-SDLM）
2. B-SDLM利用梯度和曲率信息来实现快速收敛，而只需要比随机梯度下降（SGD）方法更少的计算开销。
3. B-SDLM只有一个超参数，而不像其他的大多数学习算法由于超参数需要调整而遭受超参数过拟合问题。

【论文】Q. Chao, G. Xiao-Guang and C. Da-Qing, "On Distributed Deep Network for Processing Large-Scale Sets of Complex Data," 2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), Hangzhou, 2016, pp. 395-399.

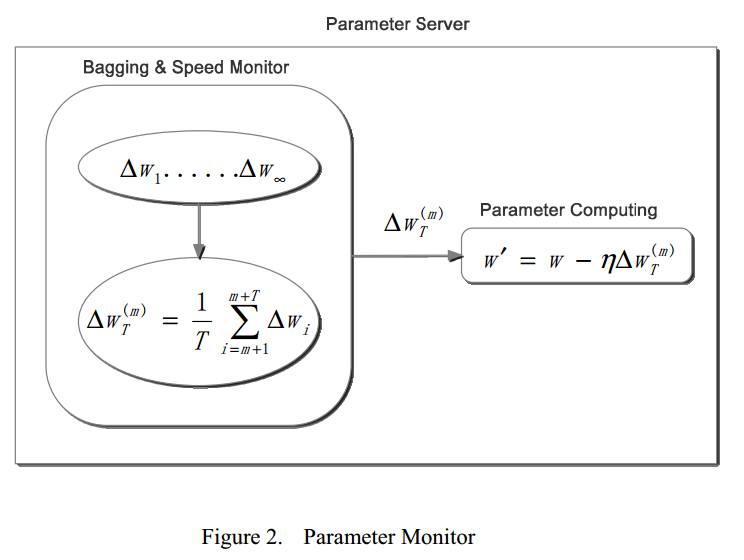
【面临问题】

在深度学习和无监督学习中，可以看出，训练大型模型可以显著提高性能。然后，作者就提出了使用分布式CPU核来训练具有数百万个参数的深度网络的问题。

【解决方案】

提出了一种有效的方法：Bagging-Down SGD，利用学习速率并支持大量模型副本的异步随机梯度下降算法。

1. 对每个参数使用单独自适应的学习率，而不是PS上使用单一的固定的学习率。
2. 对单个模型，采用多个副本来进行异步的训练，解决了同步等待的问题。
3. 在PS中增加了Bagging and Speed Monitor，用来收集参数并且创建参数序列，然后每次利用T个参数来更新权重，如图;



【论文】Gong T, Fan T, Guo J, et al. GPU-based parallel optimization of immune convolutional neural network and embedded system[J]. Engineering Applications of Artificial Intelligence, 2016.

【面临问题】

在设计图像识别系统的时候，普通的CNN网络会有以下的缺点，很长的训练时间、容易过拟合、高的误分类率等等。作者研究的动机是提出一种新型的基于GPU的免疫计算和在识别的准确率和速度方面来改进普通的CNN。

【解决方案】

1. 通过分析人工免疫网络和CNN之间的关系，提出了在普通的CNN中加入改进的免疫网络和基于GPU的并行计算的想法。
2. 设计了嵌入式图像识别系统和基于GPU的服务器，来处理实时需求和被限制的硬件资源的问题。
3. 设计免疫改进的卷积神经网络算法，来改进普通CNN的缺点。主要是将免疫分类算法结合到CNN的输出层中。

【论文】Li Y, Cai W. Update schedules for improving consistency in multi-server distributed virtual environments[J]. Journal of Network & Computer Applications, 2014, 41(1):263-273.

【面临问题】

分布式虚拟环境（DVE）被广泛应用于在线游戏，军事训练和协作工程应用中。在DVE中，主要任务是保持所有用户对虚拟世界的一致。多服务器体系结构已经被证明可以支持DVE中大量的用户，但是由于一些原因，服务器可能会饱和网络带宽，那么状态更新就不能及时传播，虚拟世界的一致性就不能得到保证。

【解决方案】

在论文中，作者分析了一个理想情况下的DVE（假设DVE的配置保持不变），然后寻找出最有更新时间表，以便将总TSI最小化为一个不等式约束问题（ICP），再用拉格朗日乘子用于帮助推导出一个可用于实际系统的启发式更新算法。

算法的开发是下面三个步骤：

1.对于理想情况，使用拉格朗日乘子来导出ICP最小化的条件。根据分析结果，对于实际系统，根据减少不一致性的长期收益，为每个副本定义全局更新优先级。

2.由于以集中方式测量的更新优先级可能导致计算和通信开销过大，因此提出局部调节算法以分布式方式估计每个副本的更新优先级。

3.根据每个副本的更新优先级，开发一个分布式更新算法，用于最小化总TSI

【论文】SplitNet: Learning to Semantically Split Deep Networks for Parameter Reduction and Model Parallelization

【面临问题】

一种新颖的深度神经网络。

【解决方案】

提出一种新颖的深度神经网络，可以轻便又有效的用于模型并行化。

这个网络称为SplitNet，通过学习了类对组和特征对组赋值矩阵以及网络权值，自动学会将网络权值分为多个组的集合或者多个组的层次结构，这会产生一个树状结构网络，所以SplitNet大大减少了参数的数量和所需的计算，并且在测试时还可以建模为并行，对每个子网络的计算是完全独立的。

【论文】Adaptive Consensus ADMM for Distributed Optimization

【面临问题】

乘法器的交替方向法（ADMM）通常用于分布式模型拟合问题，但其性能和可靠性强烈依赖于用户定义的惩罚参数。理论上，ADMM收敛于任何常数惩罚参数；然而，在实践中，ADMM的效率对这个参数选择非常敏感。

【解决方案】

提出了O（1 / k）自适应ADMM方法的收敛速度节点特定的参数，并提出自适应共识ADMM（ACADMM）方法，它可以通过曲率信息来自动调整参数。

【注释】

交替方向乘法器（ADMM）是一种通过将凸分解成更小的且更易处理的部分来解决凸优化问题的算法。

【论文】A Communication-Efficient Parallel Algorithm for Decision Tree

【面临问题】

随着大数据的出现，越来越需要并行化决策树的培训过程。然而，大多数沿着这条线路的现有尝试都遭受高昂的通信费用。

【解决方案】

提出了一种称为并行投票决策树（PV-Tree）的新算法，在每次迭代中执行本地投票和全局投票。本地投票根据本地数据从每台机器中选择top-k属性，全局投票通过本地投票的结果再次投票选出top-2k属性。最后，从本地机器收集全局top-2k属性的全粒度直方图，以便确定最佳属性及其分割点。PV-Tree可以实现非常低的通信成本（独立于总数属性数量），因此可以非常好地扩展。

【论文】Communication-Optimal Distributed Clustering

【面临问题】

聚类大型数据集是机器学习中的一些应用程序的基本问题。数据通常收集在不同的站点上，我们希望分布式设置中集群的效果与集中式设置中（所有数据驻留在单个站点上）的效果相匹配。

【解决方案】

研究了两个分布式模型中的聚类问题：（1）点对点模型;（2）具有广播频道的模型，并给出了两种模型中的协议，证明显示它们几乎是最优的。对于分布在s个服务器中的图谱中的n个点或n个顶点，在最坏情况下划分点对点模型中的通信复杂度为n·s，而在广播模型中为n +s秒。

【论文】Feature-distributed sparse regression: a screen-and-clean approach

【面临问题】

大多数现有的分布式稀疏回归方法假定数据按样本进行分区。 但是，对于高维数据，通过特征划分数据更自然。

【解决方案】

提出了一种分布式稀疏回归算法，当数据通过特征而不是样本进行分割时，允许用户通过折算通讯网络和通讯次数发送的总数据量来调整。 我们表明，我们的方法的实现能够在几分钟内解决数百万个功能的l1l2正则化回归问题。

【论文】GPU-based parallel optimization of immune convolutional neural network and embedded system

【面临问题】

在图像识别系统的设计中，传统的卷积神经网络存在一些缺陷如训练时间长，容易过拟合，分类错误率高等。

【解决方案】

运用免疫机制对卷积神经网络进行改进，并提出了一种新的免疫卷积神经网络算法，分析了卷积神经网络的网络结构和参数。算法不但集成了网络节点的位置数据和可调参数，还动态调整了基函数的平滑因子。

【注释】

免疫算法是受生物免疫系统的启发而推出的一种新型的智能搜索算法，是一种确定性和随机性选择相结合并具有“生成+检测”能力的启发式随机搜索算法。

【论文】Accelerating deep neural network learning for speech recognition on a cluster of GPUs

【面临问题】

我们采用分布式GPU集群的方式训练深度神经网络来解决大词汇量连续语音识别的声学建模问题。并证明ASGD并行化方法对于这个应用程序来说效率不高。即使使用4个GPU，也会导致大量开销而且达不到足够的精确度。

【解决方案】

采用P-learner K-step模型平均算法（KAVG）来明确管理并行实现中的梯度单调性(gradient staleness)。为了扩展超过4个GPU，我们引入了自适应批量大小方案，用以处理非常大的（不适合GPU内存的）批量大小。

【论文】A Communication Eicient Parallel DBSCAN Algorithm Based on Parameter Server【面临问题】

最近的研究表明，基于MPI的DBSCAN分布式实现（例如PDSDBSCAN）优于其他实现，如apache Spark等。但是，MPI DBSCAN的通信成本随着处理器数量的增加而急剧增加，这使得它不适用于大规模问题。

【解决方案】

提出PS-DBSCAN，一种结合了不相交集数据结构和参数服务器框架的并行DBSCAN算法，以最大限度地降低通信成本。由于同一集群内的数据点可能分布在不同的工作人员之间，导致出现多个不相交集合，因此将它们合并会导致巨大的通信成本。在我们的算法中，我们采用快速全局联合方法来结合不相交集以减轻通信负担。

【注释】

DBSCAN是一种基于密度的聚类算法，这类密度聚类算法一般假定类别可以通过样本分布的紧密程度决定。在某一类别任意样本周围不远处一定有同类别的样本存在。

通过将紧密相连的样本划为一类，这样就得到了一个聚类类别。通过将所有各组紧密相连的样本划为各个不同的类别，则我们就得到了最终的所有聚类类别结果。

【论文】Network–Efficient Distributed Word2vec Training System for Large Vocabularies

【面临问题】

Word2vec是一种流行的算法，用于无监督地训练大型文本语料库上的单词的密集向量表示。无论是将所有词汇词的向量存储在单个服务器的存储器中，还是由于大规模网络数据传输而遭受不可接受的训练等待时间，都使得现有的word2vec训练系统对于训练超大规模词汇量的新应用不切实际。

【解决方案】

提出一种新颖的分布式并行训练系统，该系统能够在共享的商品服务器集群上为数亿个词汇的词汇表提供前所未有的矢量实用培训，使用的网络流量远远低于现有解决方案。

【论文】TencentBoost: A Gradient Boosting Tree System with Parameter Server

【面临问题】

梯度提升树（GBT）是一种广泛使用的机器学习算法，尽管实现GBT的现有可伸缩系统（如XGBoost和MLlib）对于具有中等维度特征的数据集性能良好，但对于训练数据集包含高维特征的许多工业应用来说，它们可能会遭受性能下降。

【解决方案】

提出了一个使用可扩展的参数服务器体系结构来促进模型聚合，同时引入稀疏拉伸方法和高效的索引结构来提高处理速度。

【论文】Reducing view inconsistency by predicting avatars' motion in multi-server distributed virtual environments

【面临问题】

多服务器分布式虚拟环境（DVE）系统包含一组服务器以支持大量地理分布的用户。它提供了一个共享的虚拟场景，其中用户通过替身进行网络交流。然而，在设计具有这种架构的高效DVE系统时存在两个技术挑战：查看由于用户分布不均导致的网络延迟和服务器过载造成的不一致。

【解决方案】

提出了一个新的分区框架来同时解决这两个问题，其中DVE方案被分成一组不相交的网格，称为虚拟单元。然后通过区域四叉树对虚拟单元进行索引，并且提出了一种算法来搜索区域四叉树以便在服务器之间平均分配替身。通过使用概率模型预测每个替身的未来运动和其感兴趣区域将覆盖的虚拟区域。通过估算替身未来位置到每个集群的几何中心的可能距离，提出了一种算法来重新分配服务器之间的未分配替身，以减少视图不一致并保持服务器之间的工作负载平衡。

【论文】PR-ELM: Parallel regularized extreme learning machine based on cluster

【面临问题】

极限学习机（ELM）由于其高效和易于实施的特点而在过去的十年中得到了深入的研究。近来，已经提出许多变体，例如并行ELM（PELM），增量ELM和在线顺序ELM（OS-ELM），以改进其时序性能并使其能够增量学习。

【解决方案】

我们提出两个并行变体，称为数据并行正则化ELM（DPR-ELM）和模型并行正则化ELM（MPRELM），以进一步提高ELM在处理大规模学习任务时的计算效率。所提出的PR-ELM算法优于现有变体的优点如下所示：

（1）它们独立地训练每个数据块或每个子模型，因此它们具有更好的并行性。

（2）整个数据集或整个模型被分割成小块或子模型，因此它们大大减少了大量运行时内存的需求。

（3）它们可以在具有更多计算节点的群集上进行配置，因此DPR-ELM和MPR-ELM都具有更好的可扩展性。

【论文】Update schedules for improving consistency in multi-server distributed virtual environments

【面临问题】

分布式虚拟环境（DVE）已广泛应用于在线游戏，军事训练和协作工程应用等。但是，由于几个原因，服务器仍然可能会饱和网络带宽。在这种情况下，状态更新不能及时传播，虚拟世界的一致性无法得到保证。

【解决方案】

首先，在理想情况下分析DVE，假设DVE的配置保持不变。制定了寻找最佳更新时间表的问题，以便将总TSI最小化为一个不等式约束问题（ICP）。其次，使用拉格朗日乘子来帮助推导可用于实际系统的启发式更新算法。

十、算法分析

【论文】Li Q, Zhou Y, Liang Y. Convergence Analysis of Proximal Gradient with Momentum for Nonconvex Optimization[C]// ICML. 2017.

【面临问题】

研究非凸规划的加速近似梯度法。

【解决方案】

将通常的近端梯度步与线性外推步进行比较，并接受函数值较低的一种，实现单调递减。并且进一步提出了一个随机方差减少APGnc(SVRG-APGnc)算法，并建立其线性收敛的一种特殊情况下的KL属性。

【论文】Fast ADMM Algorithm for Distributed Optimization with Adaptive Penalty

【解决方案】

提出了一种新的方法来加速交替方向乘法器（ADMM）的交汇，通过在每次迭代中自动决定参数一致所需的约束惩罚来加速收敛速度。另外，我们还提出了自适应确定更新惩罚的最大迭代次数的方法的扩展，并证明这种方法有效地导致了分布式优化的自适应动态网络拓扑结构。

【论文】Failures of Gradient-Based Deep Learning

【解决方案】

描述了四种深度学习中的简单问题，在这四种问题中，基于梯度的深度学习会失败或者遇到极大阻碍。

【论文】An optimized second order stochastic learning algorithm for neural network training

【解决方案】

提出了一种改进的随机二阶学习算法用于监督神经网络训练，称为有界随机对角线Levenberg-Marquardt（BSDLM），它利用梯度和曲率信息来实现快速收敛，而只需要最小的计算开销而不是随机梯度下降（SGD）方法。 B-SDLM只有一个超参数，因此不会受到超参数过拟合问题的困扰。使用多层感知器（MLP）和卷积神经网络（CNN）模型的实验表明，B-SDLM在分类准确性和计算效率方面优于其他学习算法。

【论文】Asynchronous Stochastic Gradient Descent with Delay Compensation

【面临问题】

异步随机梯度下降（Asynchronous Stochastic Gradient Descent，ASGD）被广泛用于使用大量训练数据来学习大型神经网络，但是，它仍然受到梯度延迟问题的困扰。也就是说，当本地worker将梯度添加到全局模型时，全局模型可能已被其他worker更新，从而使得梯度产生延迟

【解决方案】

提出一种通过利用梯度函数的Taylor展开和Hessian的有效逼近来实现的损失函数矩阵的算法，称之为延迟补偿ASGD（DCASGD）。对CIFAR-10和ImageNet数据集上的算法进行的评估表示， DC-ASGD优于同步SGD和异步SGD，且几乎接近顺序SGD的性能。

【论文】On Distributed Deep Network for Processing Large-Scale Sets of Complex Data

【面临问题】

随机梯度下降（SGD）可能是训练深度神经网络最常用的优化程序。然而SGD的传统表述本质上是连续的，使得将它应用于非常大的数据集上不切实际。 如果一台机器出现故障，则整个训练过程的时长将取决于最慢的机器。

【解决方案】

提出了一种异步随机梯度下降过程中有效的方法：Bagging-Down SGD，提高学习速率并支持大量模型副本，以确保在处理大规模数据时的学习速率和准确性。

【论文】Asynchronous Distributed Semi-Stochastic Gradient Optimization

【面临问题】

现有的随机梯度下降（SGD）算法由于随机梯度的固有方差，很多收敛缓慢，一些收敛比较快的方法也是以牺牲准确率为代价的。

【解决方案】

提出了一种快速分布的基于SGD的异步减少算法。 可以使用恒定的学习速率，并且还可以保证线性地收敛到最佳解决方案。

【论文】Weight Normalization: A Simple Reparameterization to Accelerate Training of Deep Neural Networks

【面临问题】

基于一阶梯度的优化高度依赖于被优化的目标曲率。能有多种参数化同一模型的等价方法，寻找神经网络参数化的好方法是深度学习中的一个重要问题。

【解决方案】

提出一种权重归一化的方法，用于提高神经网络模型权重的优化性，加速随机梯度下降优化的收敛。

【论文】A Comprehensive Linear Speedup Analysis for Asynchronous Stochastic Parallel Optimization from Zeroth-Order to First-Order

【解决方案】

本文通过全面而通用的分析，研究了从零阶到一阶法的各种异步并行随机算法的加速性能，并首次提出了一种新颖的异步并行零阶方法。

【论文】Asynchronous Coordinate Descent under More Realistic Assumption

【解决方案】

证明了有界和无界延迟下的异步BCD收敛结果的选择，以及随机和确定性块选择。 到目前为止，这些结果不需要绝大多数的独立性假设。 因此，异步BCD可以更好地模拟真正的异步求解器的行为。

1. 加速器

【论文】Nair A, Srinivasan P, Blackwell S, et al. Massively Parallel Methods for Deep Reinforcement Learning[J]. Computer Science, 2015.

【面临问题】

提出了针对深度强化学习的大规模分布式架构

【解决方案】

该架构利用四个组件来实现——产生新行为的并行节点，利用存储的内容进行训练的并行节点，函数值表示或行为策略的分布式神经网络，分布式的内容存储。利用新架构实现Deep Q-Network。

【论文】Chen T, Zhang S, Liu S, et al. A Small-Footprint Accelerator for Large-Scale Neural Networks[J]. Acm Transactions on Computer Systems, 2015, 33(2):1-27.

【面临问题】

针对大型CNN和DNN的机器学习加速器性能受限

【解决方案】

设计了一个用于大型CNN和DNN的高通量加速器，特别强调记忆对加速器的设计，性能和能量的影响。

【论文】Tallada M G. Coarse grain parallelization of deep neural networks[J]. 2016, 51(8):1-12.

【面临问题】

描述并实现以及分析DNN训练算法的网络不可知且收敛不变的粗粒度并行化。

【解决方案】

通过批次层级的并行机制实现粗粒度的并行化，并利用MNIST和CIFAR-10实验来进行验证粗粒度并行化的加速效果。

【论文】Braun G, Pokutta S, Zink D. Lazifying Conditional Gradient Algorithms[J]. 2016.

【面临问题】

有条件的梯度算法在实际的许多情况下，其实现的代价太高。

【解决方案】

提出了一种通用的松弛条件梯度算法，该算法在实际计算中会带来几个数量级的Wall-Clock的加速效果。

（注：若有“文献整理”压缩包中的论文未在此提及的，说明其相关性较差，无借鉴价值）