Applied Machine Learning at Facebook: A Datacenter Infrastructure Perspective

Facebook的应用机器学习:数据中心基础设施的视角

（HPCA 2018，Facebook公司）

I.Introdution

Facebook使用机器学习：新闻推荐、翻译、图片分类

本文描述了支持Facebook机器学习的数据中心基础设施的几个重要方面。基础设施包括内部的“ML-as-a- Service”流、开源机器学习框架和分布式训练算法。

从硬件的角度来看，Facebook利用了大量的CPU和GPU平台来训练模型，以便在所需的服务延迟时支持必要的训练频率。对于机器学习推测，Facebook主要依赖于所有主要服务的cpu，而新闻Feed等神经网络排名服务占据了全部计算负载。

机器学习服务所需的海量数据对Facebook数据中心的全球规模提出了挑战。有几种技术可以有效地将数据提供给模型，包括数据提供和训练的解耦、数据/计算的协同定位和网络优化。与此同时，Facebook的规模提供了独特的机遇。在非高峰时段，日负载周期为分布式训练算法留下了大量可用的cpu。Facebook的计算团队分布在10个数据中心位置，scale还提供了灾难恢复功能。灾难恢复计划至关重要，因为及时交付新的机器学习模型对Facebook的运营非常重要。

展望未来，Facebook预计机器学习将在现有和新服务[4]中快速增长。这种增长将导致为这些服务部署基础设施的团队面临越来越大的可伸缩性挑战。虽然在现有平台上存在优化基础设施的重大机会，但我们仍在积极评估和原型新的硬件解决方案，同时仍然认识到改变游戏规则的算法创新。

本文的主要贡献包括以下关于Facebook机器学习的主要见解:

•机器学习被广泛应用于几乎所有的服务，而计算机视觉只代表了资源需求的一小部分。

•Facebook依赖极其多样化的机器学习方法，包括但不限于神经网络。

•大量数据通过我们的机器学习管道输送，这对工程和效率的挑战远远超出了计算节点。

•Facebook目前严重依赖cpu进行推理，以及cpu和gpu进行培训，但不断地从每瓦性能的角度构建和评估新的硬件解决方案。

•Facebook用户的全球规模和相应的日常活动模式导致了大量的机器可以用于机器学习任务，比如大规模的分布式培训。

II. Machine Learning At Facebook

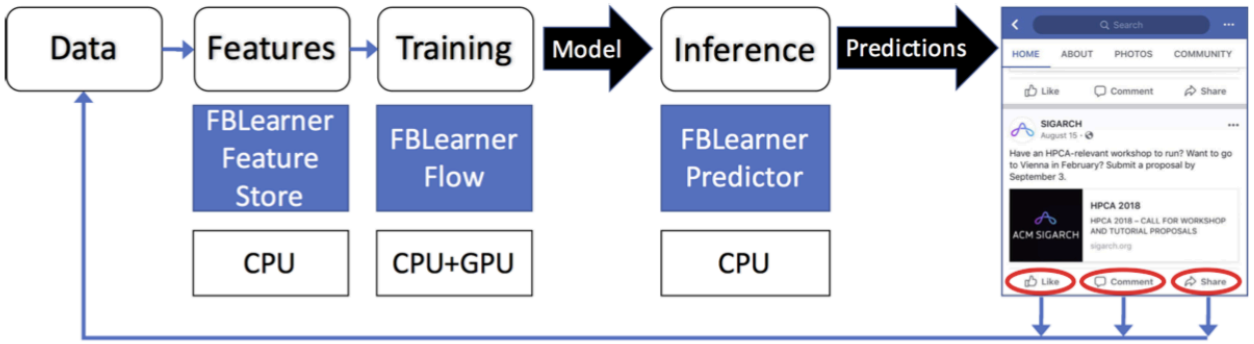

Fig. 1. Example of Facebook's Machine Learning Flow and Infrastructure.

A. Major Services Leveraging Machine Learning

News Feed、Ads、Search、
Sigma（是通用分类和异常检测框架，用于各种内部应用程序，包括网站完整性、垃圾邮件检测、支付、注册、未经授权的员工访问和事件建议。Sigma包括每天在生产中运行的数百个不同的模型，每个模型都经过训练，以检测异常或更一般地对内容进行分类。）
Lumos（从图像及其内容中提取高级属性和嵌入，使算法能够自动理解它。这些数据可以用作其他产品和服务的输入，例如，就像文本一样。）
Facer（是Facebook的人脸检测和识别框架。给定一个图像，它首先找到该图像中的所有面孔。然后，它运行一个特定于用户的人脸识别算法，以确定该人脸属于已启用人脸识别的前n位好友之一的可能性。这使得Facebook能够建议你的哪些朋友可能想要在你上传的照片中添加标签。）
Language Translation、
Speech Recognition（是将音频流转换为文本的服务。这为视频提供了自动字幕。）

B. Machine Learning Models

Models	Services
Support Vector Machines (SVM)	Facer (User Matching)
Gradient Boosted Decision Trees (GBDT)	Sigma
Multi-Layer Perceptron (MLP)	Ads, News Feed, Search, Sigma
Convolutional Neural Networks (CNN)	Lumos, Facer (Feature Extraction)
Recurrent Neural Networks (RNN)	Text Understanding, Translation, Speech Recognition

TABLE I
MACHINE LEARNING ALGORITHMS LEVERAGED BY PRODUCT/SERVICE.

C. ML-as-a-Service Inside Facebook
有几个内部平台和工具包旨在简化在Facebook产品中利用机器学习的任务。主要的例子包括FBLearner、Caffe2 PyTorch。FBLearner是一套由三种工具组成的套件，每一种工具都侧重于机器学习管道的不同部分。FBLearner利用内部作业调度程序在共享的gpu和cpu池上分配资源和调度作业，如图1所示。Facebook的大部分ML培训都是通过FBLearner平台进行的。通过共同工作，这些工具和平台旨在使ML工程师更高效，并帮助他们专注于算法创新。

FBLearner Feature Store.
（特性库本质上是几个特性生成器的目录，可以用于培训和实时预测，并且它作为一个市场，多个团队可以使用它来共享和发现特性。）

FBLearner Flow
（FBLearner Flow是Facebook的机器学习平台，用于模型训练[8]。Flow是一个管道管理系统，它执行一个工作流，描述培训和/或评估一个模型的步骤以及所需的资源。工作流由离散单元或操作符构建，每个操作符都有输入和输出。通过跟踪从一个操作符到下一个操作符的数据流自动推断操作符之间的连接，流处理调度和资源管理以执行工作流。Flow还拥有用于实验管理的工具和一个简单的用户界面，该界面跟踪由每个工作流执行或实验生成的所有工件和度量。用户界面使比较和管理这些实验变得简单。）

FBLearner predict
（是Facebook的内部推理引擎，它使用在Flow中训练的模型来实时提供预测。预测器可以用作多租户服务，也可以用作可集成到特定于产品的后端服务中的库。Facebook的多个产品团队都使用预测器，其中许多团队都需要低延迟解决方案。）

流和预测器之间的直接集成还有助于运行在线实验和管理产品中模型的多个版本。

D. Deep Learning Frameworks
在Facebook，我们利用了两种截然不同但具有协同作用的框架来进行深度学习:PyTorch(针对研究进行了优化)和咖啡因2(针对生产进行了优化)。

Caffe2
是Facebook的内部生产框架，用于培训和部署大规模机器学习模型。特别地，Caffe2关注产品所需的几个关键特性:性能、跨平台支持，以及基本机器学习算法(如卷积神经网络(CNNs)、递归网络(RNNs)和多层感知器(MLPs)的覆盖范围(稀疏或密集的连接，并且具有高达数百亿个参数)。该设计涉及一种模块化方法，其中统一的图形表示在所有后端实现(cpu、gpu和加速器)之间共享。单独的执行引擎提供不同的图执行需求，而Caffe2抽象引入第三方库(例如cuDNN、MKL和Metal)，以便在不同平台上实现最佳运行时。

PyTorch
是Facebook人工智能研究的首选框架。它的前端关注灵活性、调试和动态神经网络，从而支持快速实验。由于其执行依赖于Python，因此不适合生产和移动部署。当研究项目产生有价值的结果时，需要将模型转移到生产中。传统上，这是通过在产品环境中使用其他框架重写培训流水线来实现的。最近，我们开始构建ONNX工具链来简化传输过程。例如，动态神经网络被用于前沿的人工智能研究，但是模型需要更长的时间来成熟到足以用于生产。通过解耦框架，我们避免了设计性能所需的更复杂的执行引擎(如Caffe2中的引擎)。此外，研究人员可能会优先考虑灵活性而不是速度。例如，在探索阶段，性能下降30%可能是可以容忍的，特别是当它带来模型可检查性和可视化的好处时。但是，同样的降解不适用于生产。这种二分法出现在各自的框架中，PyTorch提供了良好的默认值和合理的性能，而Caffe2可以使用异步图执行、量化权重和多个专用后端等特性来实现最大的性能。

FBLearner平台不知道正在使用的框架是什么，无论是咖啡因2、TensorFlow、PyTorch还是其他替代品，但是AI软件平台团队提供了特定的功能来允许FBLearner很好地与Caffe2集成。总的来说，解耦研究和生产框架(分别为PyTorch和咖啡因2)使我们能够在每一边快速移动，在添加新特性的同时减少约束的数量。

ONNX。
深度学习工具生态系统在整个行业中仍处于早期阶段。不同的工具更适合不同的问题子集，并且在灵活性、性能和支持平台上具有不同的权衡—类似于前面描述的PyTorch和Caffe2的权衡。因此，人们非常希望在不同框架或平台之间交换经过培训的模型。为了填补这一空白，在2017年底，我们与几个利益相关者合作，引入了开放神经网络交换(ONNX)[9]，这是一种以标准方式表示深度学习模型的格式，以支持跨不同框架和供应商优化的库的互操作性。ONNX被设计为一个开放规范，允许框架作者和硬件供应商参与设计，并拥有框架和库之间的各种转换器。我们正在与这些合作伙伴合作，使ONNX成为所有这些工具之间的活生生的协作，而不是作为一个官方标准。

在Facebook中，我们使用ONNX作为将研究模型从PyTorch环境转移到Caffe2中的高性能生产环境的主要方法。ONNX提供了自动捕获和转换模型静态部分的能力。我们有一个额外的工具链，它通过

将动态图部件映射到Caffe2中的控制流原语，或者将它们作为自定义操作符在c++中重新实现，从而促进了动态图部件从Python的传输。

III. Resource Implications Of Machine Learning

A. Summary of Hardware Resources at Facebook

Facebook基础设施在为主要软件服务生产高效平台方面有着悠久的历史，包括为每个主要工作负载[10]的资源需求定制设计的服务器、存储和网络支持。我们目前支持大约八种主要的计算和存储机架类型，它们映射到相同数量的主要服务。新服务倾向于映射到现有的机架类型，直到它们上升到保证自己的设计的级别。这些主要机架类型的设计是为了满足主要服务的资源需求。例如，图2显示了一个2U机箱，它容纳三个计算sleds，支持两种备选服务器类型。一个sled选项是一个单socket CPU服务器 (1 xcpu)支持web层,这是一个面向输出量的无状态的工作负载,因此可用更低功耗CPU (Broadwell-D处理器)与相对少量的DRAM (32 gb)和最小的片上硬盘或闪存[11]。另一个sled选项是一个更大的双插槽CPU服务器(2x 高功率BroadwellEP或Skylake SP CPU)，带有大量DRAM，用于计算和内存密集型服务。

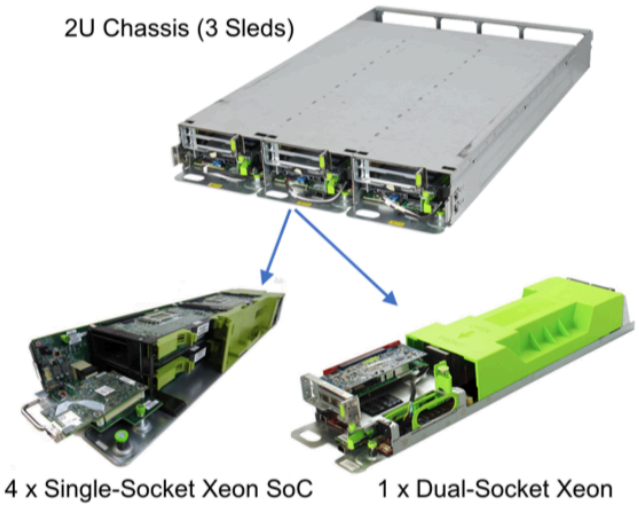

Fig. 2. CPU-based compute servers. The single-socket server sled contains 4 Monolake server cards, resulting in 12 total servers in a 2U form factor. The dual-socket server sled contains one dual-socket server, resulting in three dual-socket servers in a 2U chassis.

为了在训练更大更深入的神经网络的同时加快我们的进程，我们还在2017年创建了最新一代GPU服务器Big Basin，如图3所示。最初的大盆设计包括八个NVIDIA Tesla P100 GPU加速器，使用NVIDIA NVLink连接，形成一个8 GPU混合立方体网格[12]。该设计已经升级到支持V100 gpu以及。

Big Basin是我们之前的Big Sur GPU服务器的继任者，后者是我们数据中心中第一个广泛部署的高性能人工智能计算平台，旨在支持NVIDIA M40 GPU，后者是2015年开发的，通过开放计算项目发布。与大苏尔相比，新的V100大盆地平台在每瓦特性能上有更好的提高，得益于每GPU的单精度浮

点运算从7万亿次浮点运算增加到15.7万亿次浮点运算，以及提供900 GB/s带宽(大苏尔3.1倍)的高带宽内存(HBM2)。为了进一步提高吞吐量，这种新架构的半精度也提高了一倍。Big Basin可以训练出比原来大30%的模型，因为它具有更高的算术吞吐量，内存也从12gb增加到了16gb。分布式训练也通过gpu间的高带宽NVLink通信得到了增强。在使用ResNet-50图像分类模型的测试中，与Big Sur相比，我们的吞吐量提高了近300%，使我们能够比以前更快地进行实验，并处理更复杂的模型。

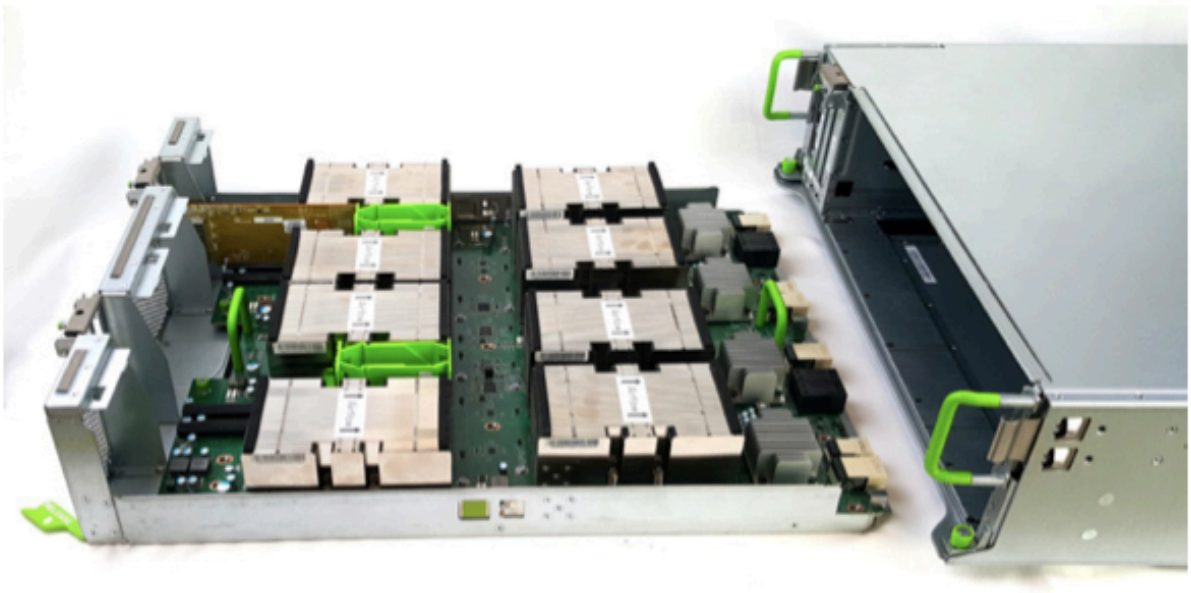

Fig. 3. The Big Basin GPU server design includes 8 GPUs in a 3U chassis.

这些计算服务器的每个设计，以及一些存储平台，都通过Open compute项目公开发布。同时，在内部，我们始终更新我们的硬件设计，并彻底评估所有有前途的替代方案和新技术。

B. Resource Implications of Offline Training

今天，不同的产品利用不同的计算资源来执行它们的离线培训步骤。一些产品，如Lumos，可以完成所有的gpu培训。其他产品，如Sigma，在dualsocket高内存CPU计算服务器上进行所有的培训。最后，像Facer这样的产品有一个两阶段的培训过程，在这个过程中，他们不经常(许多个月)在gpu上培训一个通用的人脸检测和识别模型，然后更有规律地在数千台1xCPU服务器上培训特定于用户的模型。

在本节中，我们将提供关于机器学习培训平台、频率和持续时间的各种服务的高级详细信息，如表二所示。我们还讨论了数据集的趋势和对我们的计算、内存、存储和网络基础设施的影响。

Service	Resource	Training Frequency	Training Duration
News Feed	Dual-Socket CPUs	Daily	Many Hours
Facer	GPUs + Single-Socket CPUs	Every N Photos	Few Seconds
Lumos	GPUs	Multi-Monthly	Many Hours
Search	Vertical Dependent	Hourly	Few Hours
Language Translation	GPUs	Weekly	Days
Sigma	Dual-Socket CPUs	Sub-Daily	Few Hours
Speech Recognition	GPUs	Weekly	Many Hours

TABLE II
FREQUENCY, DURATION, AND RESOURCES USED BY OFFLINE TRAINING FOR VARIOUS WORKLOADS.

训练类型：
•GPU培训:Lumos，语音识别，语言翻译

•CPU培训:新闻推送，Sigma

•两者培训:Facer(由于模型稳定，每隔几年对GPU进行一次通用模型培训;特定于用户的模型在1xCPU上训练，以响应新图像数据的阈值)，搜索(利用多个独立的垂直搜索，并应用预测分类器启动最合适的垂直搜索)。

目前GPU机器的主要用例是离线训练，而不是为用户提供实时数据。这在逻辑上是说得通的，因为大多数GPU架构都是针对吞吐量超过延迟进行优化的。与此同时，培训过程确实大量利用了来自大型生产存储的数据，因此出于性能和带宽方面的原因，gpu需要位于所访问数据附近的生产环境中。每个模型所使用的数据都在快速增长，因此随着时间的推移，数据源的本地性(其中许多是区域性的)变得越来越重要。

内存、存储和网络：
从内存容量的角度来看，CPU平台和GPU平台都提供了足够的训练能力。这甚至适用于像Facer这样的应用程序，它在我们的1xCPU服务器上用32gb RAM训练特定于用户的SVM模型。在可能的情况下，利用高效的平台和闲置的产能可以获得显著的整体效率。
需要大量的本地/附近存储，允许从远程区域离线批量数据传输，以避免训练管道等待额外的示例数据而停滞。

规模考虑和分布式培训
数据并行性包括生成模型副本(并行实例)来并行处理多个批。
由于培训所需的数据随着时间的推移而增加，硬件限制可能导致不可接受的总体培训延迟和收敛时间的增加。分布式培训是克服这些硬件限制和减少延迟的一种解决方案。
一个常见的假设是，对于跨机器的数据并行性，需要一个专门的互连。然而，在我们的分布式培训工作中，我们发现基于以太网的网络已经足够了，提供了近乎线性的扩展能力。线性扩展的能力与模型大小和网络带宽密切相关。如果网络带宽太低，以至于执行参数同步比执行梯度计算花费更多时间，那么跨机器的数据并行性的好处就会减少。凭借其50G以太网网卡，我们的大型流域服务器允许我们扩展视觉模型的训练，而不需要机器间同步成为瓶颈。
在所有情况下，需要使用在同步(每个副本都看到相同的状态)、一致性(每个副本生成正确的更新)和性能(按次线性伸缩)上提供权衡的技术与其他副本共享更新，这些技术可能会影响培训质量。使用某些超参数设置，我们可以将图像分类模型训练成非常大的小批，扩展到256+ gpu[13]。对于我们的一个更大的工作负载,数据并行性已被证实能提供4x吞吐量（使用5x机器数）。

如果模型变得异常大，可以使用模型并行性训练，其中模型层被分组和分布，以优化机器之间的激活管道吞吐量。优化可能与网络带宽或延迟有关，或者与平衡内部机器限制有关。这增加了模型的端到端延迟，因此步骤时间内的原始性能增长常常与步骤质量的下降相关联。这可能会进一步降低每一步的模型精度。步长精度的综合下降可能导致最优的并行处理量。

在许多情况下，在推理过程中，DNN模型本身被设计为在一台机器上运行，因为在机器之间划分模型图会导致大量的通信。但主要服务公司一直在权衡扩大其模式的成本/效益。这些考虑可能决定网络容量需求的变化。

C. Resource Implications of Online Inference

Services	Relative Capacity	Compute	Memory
News Feed	100X	Dual-Socket CPU	High
Facer	10X	Single-Socket CPU	Low
Lumos	10X	Single-Socket CPU	Low
Search	10X	Dual-Socket CPU	High
Language Translation	1X	Dual-Socket CPU	High
Sigma	1X	Dual-Socket CPU	High
Speech Recognition	1X	Dual-Socket CPU	High

TABLE III
RESOURCE REQUIREMENTS OF ONLINE INFERENCE WORKLOADS.

离线训练之后，在线推理步骤包括将模型加载到机器上，并使用实时输入运行该模型，从而为web流量生成实时结果。表三总结了几种服务的相对计算能力和计算类型。

为了提供一个在线推理模型运行的例子，我们将遍历广告排名模型。广告排名模型会向下显示成千上万的广告，以显示新闻提要中排名前1到5位的广告。这是通过对依次较小的广告子集执行逐步复杂的排序计算来实现的。每一次传递都由一个类似mlp的模型组成，该模型包含稀疏的嵌入层，每一次传递都缩小候选广告的数量。稀疏嵌入层是内存密集型的，因此对于“模型拥有更多参数”的后续传递，它在与MLP传递分开的服务器上运行。

从计算的角度来看，绝大多数在线推理运行在大量的1xCPU(单套接字)或2xCPU(双套接字)生产机器上。因为对于Facebook来说，1xCPU机器的功耗和成本效率都要高得多，所以只要有可能，就会强调将模型从2xCPU服务器迁移到1xCPU服务器。随着高性能移动硬件的兴起，甚至可以在用户的移动设备上直接运行一些模型，以提高延迟和降低通信成本。然而，一些计算和内存密集型服务仍然需要2xCPU服务器才能获得最佳性能。

最后，不同的产品对在线推理的结果有不同的延迟要求。在某些情况下，结果数据可以被认为“很好”，或者可以在向用户返回初始快速估计后返回。例如，在某些情况下，最初将内容分类为可接受的可能是可以接受的，而运行更复杂的模型，稍后可以覆盖最初的分类。与此同时，广告和新闻Feed等模式都有严格的sla来为用户提供合适的内容。这些sla驱动了模型的复杂性和依赖性，因此更高级的计算能力可以产生更高级的模型。

IV. Machine Learning At Datacenter Scale

除了资源需求外，在数据中心级别部署机器学习时还需要考虑一些重要因素，包括重大数据需求以及面对自然灾害时的可靠性。

A. Getting Data to the Models

对于Facebook的许多机器学习模型来说，成功取决于大量高质量数据的可用性。快速处理这些数据并将其输入训练机器的能力对于确保我们进行快速有效的离线训练非常重要。

对于复杂的ML应用程序，如Ads和Feed排名，每个培训任务需要摄取的数据量超过数百tb。此外，采用复杂的预处理逻辑，确保数据被清理和标准化，从而实现高效的传输和易于学习。这对资源的要求非常高，特别是对存储、网络和CPU。

作为一个通用解决方案，我们希望将数据工作负载与培训工作负载解耦。这两个工作负载具有非常不同的特性。数据工作负载非常复杂、特别、依赖于业务，并且变化很快。另一方面，培训工作负载通常是常规的(例如GEMM)、稳定的(核心操作相对较少)、高度优化的，并且更喜欢“干净”的环境(例如独占缓存使用和最小的线程争用)。为了对两者进行优化，我们将不同的工作负载物理地隔离到不同的机器上。数据处理机器，又称“读取器”，从存储器中读取数据，对其进行处理和压缩，然后发送到训练机器(又称“训练器”)。另一方面，培训人员只专注于快速有效地执行培训方案。可以分布式阅读器和培训器，以提供极大的灵活性和可伸缩性。我们还针对不同的工作负载优化了机器配置。

另一个重要的优化指标是网络使用率。培训产生的数据流量可能非常大，有时甚至会很突兀。如果处理不当，这很容易使网络设备饱和，甚至破坏其他服务。为了解决这些问题，我们在压缩、调度算法、数据/计算布局等方面进行了优化。

B. Leveraging Scale

作为一家为全球用户提供服务的公司，Facebook必须维护大量的服务器，以处理任何给定时间的峰值负载。如图4所示，由于用户活动的变化，由于白天的负载和特定事件(例如区域假日)期间的峰值，所以在特定时间内，一个大的服务器池常常处于空闲状态。这有效地在非高峰时间提供了一个巨大的计算资源池。正在进行的一项重大工作探索了利用这些异构资源的机会，这些资源可以以灵活的方式分配给各种任务。对于机器学习应用程序，这提供了一个很好的机会来利用分布式培训机制，这种机制可以扩展到大量异构资源(例如，不同的CPU和GPU平台具有不同的RAM分配)。在这些低利用率期间可用的计算资源的规模导致了根本不同的分布式培训方法，带来了一些挑战。调度程序必须首先在异构硬件之间适当地平衡负载，这样主机就不必彼此等待同步。当训练跨越多个主机时，调度程序还必须考虑网络拓扑结构和同步成本。如果处理不当，沉重的机架内或机架间同步流量会严重影响训练速度和质量。例如，在1xCPU设计中，四个1xCPU主机共享一个50G NIC[11]。如果所有4台主机都试图同时与其他主机同步它们的梯度，那么共享NIC将很快成为瓶颈，导致数据包丢失和超时。因此，需要在网络拓扑结构和调度程序之间进行协同设计，以便在非高峰时间有效地利用空闲服务器。此外，这种算法还必须能够提供检查点，以便在负载发生变化时停止和重新启动培训。

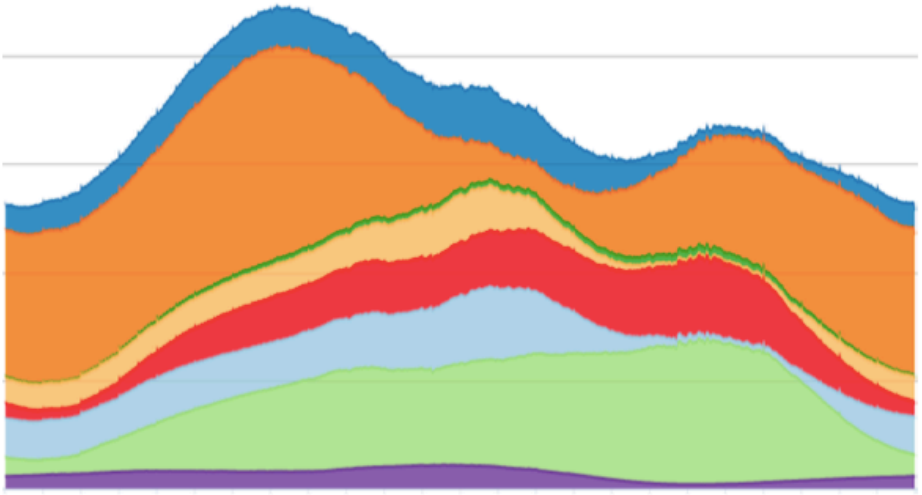

Fig. 4. Diurnal load across Facebook's fleet over a 24-hour period on 19 September 2017.

C. Disaster Recovery

无缝处理部分Facebook全球计算、存储和网络占用空间损失的能力一直是Facebook基础设施[14]的长期目标。在内部，我们的灾难恢复团队定期进行演练，以确定和纠正全球基础设施和软件堆栈中最薄弱的环节。破坏性操作包括在几乎没有通知的情况下让整个数据中心脱机，以确认丢失任何一个全球数据中心都会对业务造成最小的破坏。

对于机器学习的训练和推理部分，灾难准备的重要性不可低估。虽然推断驱动几个关键项目的重要性并不令人意外，但在注意到几个关键产品的可测量性能下降之前，对频繁培训的依赖可能令人意外。

我们讨论了三个关键产品的频繁ML培训的重要性，并讨论了适应这种频繁培训所需的基础设施支持，以及这一切与灾难恢复遵从性之间的关系。

What Happens If We Don't Train Our Models?

我们分析了利用ML培训的三个关键服务，以确定无法通过培训频繁更新模型的影响，包括广告、新闻提要和社区完整性。我们的目标是理解失去一周、一个月和六个月训练他们的模型的能力所带来的影响。

第一个明显的影响是工程师的效率，因为机器学习的进展常常与频繁的实验周期相关联。虽然许多模型都可以在cpu上进行培训，但是在某些用例中，对gpu的培训通常能够显著提高cpu的性能。这些加速提供了更快的迭代时间，以及探索更多想法的能力。因此，gpu的损失将导致这些工程师的净生产力损失。

此外，我们还确定了Facebook产品的重大影响，尤其是那些严重依赖频繁更新其模型的产品。我们总结了这些产品使用陈旧模型时出现的问题。

Community Integrity

创建一个安全的地方让人们分享和联系是Facebook的核心使命;快速准确地发现攻击性内容是这项任务的核心。我们的社区整合团队充分利用机器学习技术来检测文本、图像和视频中的攻击性内容。攻击性内容检测是垃圾邮件检测的一种特殊形式。为了向用户显示令人反感的内容，对手们一直在寻找新的和创新的方法来绕过我们的标识符。为了抵御这些努力，我们经常训练模型来学习这些新模式。每次训练迭代都要花费数天的时间来生成一个用于不良图像检测的精细模型。我们正在继续推进使用分布式培训技术更快地培训模型的边界，但是不能完全培训将导致内容的退化。

News Feed
不那么令人惊讶的是，我们发现像News Feed这样的产品严重依赖于机器学习和频繁的模型培训。在每次访问我们的站点时，为每个用户确定最相关的内容，这在很大程度上依赖于最先进的机器学习技术来正确地查找和排列这些内容。与其他一些产品不同，Feed排名的学习部分分为两个步骤:一个离线步骤来培训最佳模型，该模型运行在两个cpu / gpu上，然后是当前运行在cpu上的连续在线培训。
陈旧的新闻提要模型对质量有可衡量的影响。News Feed团队不断地在他们的排名模型上进行创新，而模型本身也要接受几个小时的培训。即使损失一周的训练计算，也会阻碍团队探索新模型和新参数的能力。

Ads
最不令人惊讶的是对广告排名模型进行频繁培训的重要性。寻找和展示最好的广告涉及到对机器学习的极大依赖和创新。为了强调这种依赖的重要性，我们了解到利用陈旧的ML模型的影响是以小时为单位度量的。换句话说，使用一个旧模型比使用一个小时的旧模型要糟糕得多。

总的来说，我们的调查强调了机器学习培训对于许多Facebook产品和服务的重要性。不应低估这一巨大和不断增加的工作量的灾害准备情况

Infrastructure Support for Disaster Recovery
图5显示了Facebook数据中心基础设施的全球分布。如果我们关注训练和推理期间使用的CPU资源的可用性，那么我们几乎在每个区域都有足够的计算服务器，以适应最大区域的潜在损失。然而，为GPU资源提供同等冗余的重要性最初被低估了。

最初利用gpu进行培训的工作负载主要是计算机视觉应用程序，培训这些模型所需的数据在全球范围内复制。当gpu还是Facebook基础设施的新成员时，将它们部署到一个单独的区域似乎是一个明智的可管理性选择，直到设计成熟，我们可以根据它们的服务和维护需求构建内部专家。这两个因素导致了将所有生产gpu物理隔离到一个数据中心区域的决定。

然而，在那之后发生了一些关键的变化。由于越来越多的产品采用深度学习，包括排名、推荐和内容理解，GPU计算和大数据之间的局部性变得越来越重要。而使计算数据托管需求复杂化的一个战略重点是存储的超大区域方法。巨型区域的概念意味着一小部分数据中心区域将容纳Facebook的大部分数据。顺便说一句，整个GPU机群所在的区域并不位于存储大区域。

因此，除了将计算与数据放在一起的重要性之外，考虑一下如果我们完全失去容纳gpu的区域会发生什么也变得非常重要。这种考虑的结果促使有必要使ML训练所用的gpu的物理位置多样化。

V. Future Directions In Co-design : Hardware ,Software , And Algorithms

随着模型复杂度和数据集大小的增加，ML的计算需求也随之增加。ML工作负载具有许多影响硬件选择的算法和数值特性。众所周知，卷积和中等规模矩阵乘法是深度学习正反向遍历的关键计算核心。对于较大的批处理大小，每个参数权重都可以更频繁地重用，因此这些内核在算术强度(每个访问内存字节的计算操作数)方面会有所改进。增加算术强度通常会提高底层硬件的效率，因此在延迟的限制下，使用更大的批运行是可取的。计算界ML工作负载将受益于更广泛的SIMD单元、专门的卷积或矩阵乘法引擎和专门的协处理器。

在某些情况下，每个节点的小批处理大小是必要的，无论是在实时推理中，当并发查询很少时，还是在培训期间，当扩展到大量节点时。较小的批处理大小常常导致较低的算术强度(例如，在完全连

接的层上的矩阵-向量乘法操作，这是固有的带宽限制)。这可能会降低几个常见用例的性能，在这些用例中，完整的模型不适合on-die SRAM或last-level缓存。这可以通过模型压缩、量化和高带宽内存来缓解。模型压缩可以通过稀疏化和/或量化[15]来实现。稀疏化在训练过程中删除连接，导致模型变小。量化压缩模型使用定点整数或更窄的浮点格式，而不是FP32(单精度浮点)的权重和激活。一些使用8位或16位的流行网络也证明了类似的精度。也有正在进行的工作，使用1或2位的权重[16]，[17]。除了减少内存占用，修剪和量化还可以通过减少带宽和允许硬件体系结构在使用定点编号时具有更高的计算速率来加快底层硬件的速度，这比在FP32值上运行效率高得多。

减少培训时间和加快模型交付需要分布式培训。正如IV-B节所讨论的，分布式培训需要对网络拓扑结构和调度进行仔细的协同设计，才能有效地利用硬件，达到良好的培训速度和质量。在分布式培训中，最广泛使用的并行形式是数据并行，如III-B节所述，它要求同步或异步地跨所有节点进行梯度下降。同步SGD需要全缩减操作。当使用递归加倍(和减半)执行all-reduce时，一个有趣的特性是带宽需求随递归级别呈指数级下降。这鼓励分层系统设计，在分层结构的底部节点形成具有高连接性的超级节点(例如，通过高带宽点对点连接或高基数开关连接);在层次结构的顶部，超级节点通过较慢的网络(例如以太网)连接。另外，异步SGD(处理批而不等待其他节点)比较困难，通常通过共享参数服务器完成;节点将其更新发送到参数服务器，参数服务器聚合并分发更新到节点。为了减少更新的滞后性和参数服务器的压力，混合设计可能是有益的。在这种设计中，异步更新发生在具有高带宽和本地节点之间低延迟连接的超级节点中，而同步更新发生在超级节点之间。进一步提高可伸缩性需要在不牺牲收敛性的前提下增加批处理大小。这是Facebook内外算法研究的一个活跃领域。

在Facebook，我们的任务是为机器学习构建高性能、节能的系统，以满足我们丰富的基于ml的应用程序的需求，如第二节所述。我们不断地评估和原型新的硬件解决方案，同时关注即将到来的、近期的和长期的算法更改，以及它们对系统级设计的潜在影响。

VI. CONCLUSION

基于机器学习的工作负载的重要性日益增加，其含义涵盖了系统堆栈的所有部分。作为回应，计算机体系结构社区对如何最好地应对由此产生的挑战越来越感兴趣。虽然之前的工作主要围绕有效地处理ML训练和推理所需的计算，但是当考虑到大规模考虑解决方案时出现的额外挑战时，情况就会发生变化。

在Facebook,我们发现,出现在规模和几个关键因素驱动的决定在我们的数据中心基础设施的设计:的重要性与计算的数据进行集中管理,处理各种各样的ML工作负载的重要性,不仅仅是计算机视觉,机会出现闲置产能从日计算周期。在设计端到端解决方案时，我们考虑了所有这些因素，这些解决方案包括定制的、现成的、开源硬件，以及平衡性能和可用性的开源软件生态系统。这些解决方案为今天服务于21亿多人的大规模机器学习工作负载提供了动力，并反映了机器学习算法和系统设计领域专家的跨学科努力。