TIRIAS RESEARCH

PLASTER：一个与深度学习性能有关的框架

本白皮书获得 NVIDIA 资助

David A.Teich 与 Paul R.Teich 合著
TIRIAS Research
2018 年 5 月

前言

机器学习 (ML) 是人工智能 (AI) 的一个重要类别。作为 ML 的一种类型，深度学习 (DL) 的相关软硬件技术取得了长足进步，对惊人的 AI 发展趋势的早期阶段起到了催化剂作用。不过，在这个采用阶段存在双重挑战：深度学习解决方案的部署是一个复杂的课题，而且它在迅速变化。业界需要一个框架来应对与深度学习相关的机遇和挑战。

在 2018 年的 NVIDIA GPU 技术大会 (GTC) 上，NVIDIA 总裁兼首席执行官 (CEO) 黄仁勋提出了 PLASTER 框架，引导听众在相关背景下思考提供基于 AI 的服务所面临的重大挑战（图 1）。

图 1：面向 AI 的 PLASTER 框架

来源：NVIDIA

“PLASTER”是一个缩略词，包含了提供基于 AI 的服务所面临的七大挑战。

- **P**rogrammability（可编程性）
- **L**atency（延迟）
- **A**ccuracy（准确性）
- **S**ize of Model（模型大小）
- **T**hroughput（吞吐量）
- **E**nergy Efficiency（能效）
- **R**ate of Learning（学习频率）

本文以 NVIDIA 的 DL 解决方案为背景，逐一探讨这些 AI 挑战。PLASTER 体现了“整体大于部分之和”的观点。有志于开发和部署基于 AI 的服务的人士应通盘考虑 PLASTER 的各个元素，以便全面认识深度学习性能。应对 PLASTER 提及的挑战在任何 DL 解决方案中都很重要，而且特别有助于开发和提供基于 AI 的服务所依赖的推理引擎。本文的每部分都会简要说明每个框架部分的测量方式，还会举例介绍利用 NVIDIA 解决方案来解决机器学习关键问题的客户。

Programmability（可编程性）

机器学习正在经历爆炸式发展，这不仅体现在模型的大小和复杂性上，还体现在迅速涌现的多种神经网络架构上。因此，甚至连专家也难以深入了解模型选项，然后选出合适的模型来解决他们的 AI 业务问题。

完成深度学习模型的编码和训练之后，要针对特定的运行时推理环境优化模型。NVIDIA 开发出两个重要工具，解决了训练和推理难题。开发人员使用 CUDA 为基于 AI 的服务编写代码，它是一个并行计算平台和编程模型，可在 GPU 上进行通用计算。此外，开发人员使用 NVIDIA 的可编程推理加速器 TensorRT 为基于 AI 的服务实现推理功能。

CUDA 简化了在 NVIDIA 平台上实现算法所需执行的步骤，对数据科学家的帮助极大。而 TensorRT 可编程推理加速器工具将经过训练的神经网络作为输入，优化它在运行时部署下的性能。它测试不同的浮点和整数精度水平，让开发人员和操作能在系统所需的准确性和性能之间取得平衡，从而提供优化的解决方案。

开发人员可以直接从 TensorFlow 框架中使用 TensorRT 来优化模型，以便更好地提供基于 AI 的服务。TensorRT 可以从多种框架（包括 Caffe2、MXNet 和 PyTorch）中导入开放神经网络交换 (ONNX) 格式的模型。鉴于深度学习目前仍停留在技术层面的编码阶段，数据科学家可以借助这种导入功能更好地利用宝贵的时间。

测量可编程性

可编程性影响着开发人员的工作效率，进而影响产品上市时间。TensorRT 能加快多个常用框架（包括 Caffe2、Kaldi、MXNet、PyTorch 和 TensorFlow）上的 AI 推理速度。此外，TensorRT 可以接受 CNN、RNN 和 MLP 网络作为输入，并且提供自定义层 API 来处理新颖、独特或专有的层，让开发人员能够实现自己的 CUDA 内核功能。TensorRT 还支持 Python 脚本语言，可让开发人员将基于 TensorRT 的推理引擎集成到 Python 开发环境中。

通过实例了解可编程性

Baker Hughes (BHGE) 是一家领先的油田服务公司。它全方位地帮助油气公司进行勘探、开采、加工和付运。在整个过程的每一步，AI 都能帮助油气公司更好地了解其业务产生的大量数据。每种类型的业务需求可能会依赖不同类型的深度学习模型。这意味着编程人员必须能够高效地实施、测试和实例化多个模型。

BHGE 使用 CUDA 和 TensorRT 创建深度学习模型，以帮助客户对油气资源进行识别和定位。BHGE 还使用了多种 NVIDIA 硬件，包括：使用 DGX-1 服务器来训练模型；在桌边系统或偏远的海上平台使用 DGX 工作站进行模型训练和推理；以及在物联网 (IoT) 的边缘使用 NVIDIA 的 Jetson 平台进行实时、持续的深度学习和推理。

Latency（延迟）

人和机器都需要对象反应才能作出决策和采取行动。延迟是指提出请求与收到反应之间经过的时间。就大多数面向人类的软件系统（不只是 AI）而言，延迟时间通常以毫秒计。

拜 Siri、Alexa 和类似的语音控制界面所赐，语音识别类应用已经广为人知。消费者和客户服务应用对数字助理的需求很广泛。但是，在人尝试与数字助理交互时，即使是短短几秒的延迟也会开始让人感到不自然。

图像和视频管理是需要低延迟、基于实时推理的服务的另一个例子。按照 Google 的说法，对于图像和视频类应用，7 毫秒是最佳延迟目标。

另一个例子是自动翻译。早期的系统采用较为程序化的专家系统设计，并不能足够快速地理解语言的细微差别，因此无法进行实际对话。现在，DL 的表现远胜从前，能产生大为改进的翻译效果。

测量延迟

推理延迟直接影响用户体验 (UX)，它的测量单位是秒或几分之一秒。虽然对反应时间并无严格规定，不过 Jakob Nielsen 的 0.1/1/10 秒限制是很好的准则。如果反应时间介于 2 到 10 秒之间，人们就会开始揣测系统的运行是否仍然正常。用户的活动流程会被打断，从而影响乐趣、性能、时间和金钱。

通过实例了解延迟

作为在线搜索的新风向，视觉搜索方兴未艾。微软的 Bing 服务器工作组一直希望开发出能快速提供搜索结果的视觉搜索平台，为此构建了一个基于神经网络的解决方案。系统的初始延迟约为 2.5 秒，但通过使用 Tesla GPU，微软将延迟降低到仅仅 40 毫秒，也即降低了 62 倍！

Accuracy（准确性）

准确性在各行各业都很重要，但医疗保健业需要特别高的准确性。过去数十年，医学成像技术取得了长足发展，在使用次数越来越多的同时，也要求更多的分析以找出医疗问题。医学成像技术的发展和应用还意味着需要将大量数据从医疗设备传输给医疗专家进行分析。一直以来，无非通过两种方式解决此数据量问题：在高延迟的情况下传输完整的信息，或者数据取样和重建，但相关技术可能导致重建和诊断不准确。

深度学习的一个优点是高精度训练和低精度实施。DL 训练可以在较高的数学精度水平（通常为 FP32）上非常精确地进行。之后，在运行时环境中可以在较低的数学精度水平（通常为 FP16）上进行实施，从而获得更高的吞吐量和效率，甚至还能降低延迟。保持较高的准确性对于确保最佳用户体验至关重要。TensorRT 利用 Tesla V100 Tensor 核心的 FP16 处理功能以及 Tesla P4 的 INT8 功能将推理速度加快了 2 到 3 倍（与 FP32 相比），而且准确性几乎没有下降。

开发人员在开发基于 AI 的服务时可以从效率方面优化深度学习模型，然后以经济实惠的方式现场实施这些模型。

测量准确性

可以通过多种方式来定义准确性的测量。对于 PLASTER，准确性涉及这个问题：在运行时推理中让经过训练的模型保持准确性，同时还要优化推理性能以提高运行时效率（降低延迟等）。在运行时保持准确性的关键在于降低数学精度，以获得卓越的能效、提高吞吐量和利用其他好处，同时不会让准确性降到每个应用场合所需的等级之下。TensorRT 通过允许在多个精度水平上进行的推理比较准确性的变化，帮助作出有关准确性的决策（图 2）。

图 2：TensorRT 的低精度推理性能

	FP32 排名第一	INT8 排名第一	差值
Googlenet	68.87%	68.49%	0.38%
VGG	68.56%	68.45%	0.11%
Resnet-50	73.11%	72.54%	0.57%
Resnet-152	75.18%	74.56%	0.61%

针对 INT8 推理的精度校准：

- 在校准数据集上将 FP32 与 INT8 推理之间的信息损失减到最少
- 全自动

来源：NVIDIA

通过实例了解准确性

麻省总医院的 AA 马蒂诺生物医学成像中心正联合哈佛大学研究一些系统，以加速和改进 MRI 图像的重建。这两家机构利用 DGX-1 开发出 AUTOMAP 深度学习系统，用它直接从传感器数据重建图像。此深度学习系统将噪声和瑕疵过滤掉，使图像重建速度和准确性分别提高了 100 倍和 5 倍，从而带来更准确的诊断结果。

Size of Model（模型大小）

深度学习模型的大小和处理器之间的物理网络容量会影响性能，特别是从 PLASTER 的延迟和吞吐量方面来说。深度学习网络模型的数量正在激增。此类模型的大小和复杂性也在增长，这不仅允许进行更详细的分析，还推动着对功能更强大的训练系统的需求。在深度学习模型中，计算能力和物理网络扩展的推动因素包括：

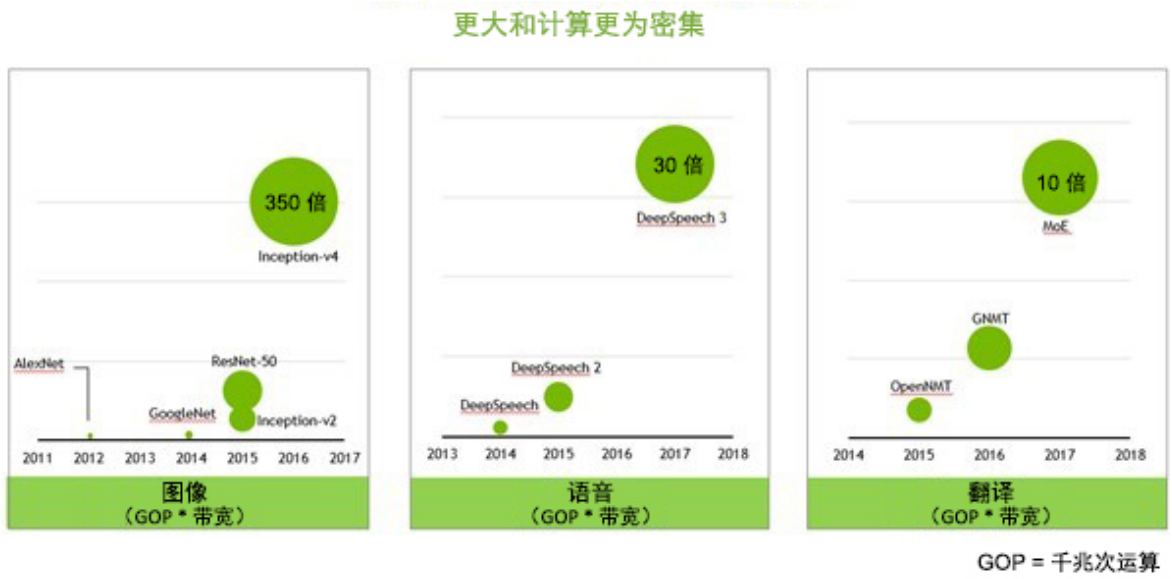
- **层**数
- 每层**节点**数（神经元数）
- 每层的计算**复杂度**
- 某层的某个节点与邻近层的节点之间的**连接**数

当前正处于 DL 市场生命周期的早期阶段。在比较模型大小时，目前的想法可归结为一种实际关系：DL 模型大小与在其他 PLASTER 元素的背景下运行推理所需的计算量和物理网络资源成正比。例如，如果开发人员对训练后的 DL 模型进行优化，使其不会超出规定的推理准确性和延迟范围，那么，该优化可能会降低计算精度和简化每个模型层及其之间的连接。但是，开始时使用较大的训练后模型通常会导致用于推理的优化模型也较大。

测量模型大小

开发人员通常从计算需求和内存延迟两方面来描述 DL 模型大小。对于深度学习模型，图 3 显示模型大小由计算需求和在计算内存空间中移动数据所需的物理网络带宽共同决定。

图 3：深度学习模型大小



来源：NVIDIA

在多个主要应用领域中，深度学习模型的数量已增长了 1-2 个数量级。大小、复杂性和计算需求的这种增长，加上需要低延迟的实时服务的出现，突显了模型大小方面的挑战。因此，必须在硬件层面和通过调整运行时推理准确性（精度）来消除对较大模型的延迟和吞吐量值的影响。

通过实例了解模型大小

百度研究院在 2014 年年底发布了它的 Deep Speech (DS1) 语音识别原始模型。DS1 使用一个 5 层卷积神经网络 (CNN) 模型，以及 1 个递归神经网络 (RNN) 层和大约 810 万个参数。一年后，第二代 Deep Speech 2 (DS2) 模型使用了一个 7 层 RNN，以及 3 个 CNN 层和大约 6770 万个参数（增加到 DS1 的 8.3 倍）。识别词错误率 (WER) 从使用 DS1 时的 24.0% 降低到使用 DS2 时的 13.6%（改善了 43%）。更新颖的 DS2 模型也更大和更复杂，直接导致了语音识别准确性显著改善。NVIDIA 和百度在 2017 年宣布开展合作，以提升数据中心的 AI 训练速度和加速性能。

Throughput（吞吐量）

吞吐量描述所创建或部署的深度学习网络在给定大小的情况下可以提供多少次推理。开发者正在指定的延迟阈值内逐渐优化推理性能。延迟限定可确保良好的客户体验，在该限值内最大化吞吐量对最大程度提高数据中心效率和营收至关重要。

一直以来，业界都倾向于将吞吐量用作唯一的性能指标，原因是每秒计算次数越高，其他方面的性能通常也越好。但是，如果系统未能按照指定的延迟要求、功耗预算或服务器节点数提供足够的吞吐量，最终将无法很好地满足应用场合的推理需求。如果未能在吞吐量和延迟之间取得适当的平衡，可能会导致客户服务水平低下、未达到服务水平协议 (SLA) 的要求和服务遭遇失败。

娱乐行业长期以来使用吞吐量作为关键性能指标，特别是在动态广告投放中。例如，品牌赞助商在电视节目或体育赛事等流视频中动态投放广告。广告商想知道其广告的出现频率以及是否触及目标受众。要让广告商感到满意，关键在于向广告商报告广告投放的准确性和重点。

测量吞吐量

DL 推理吞吐量通常表示为每秒图像数（对于基于图像的网络）和每秒令牌数（对于基于语音的网络）。系统必须在指定的延迟阈值内达到吞吐量。服务运营商可以通过增加 GPU 的数量使每次推理的延迟保持合适水平，以此来管理推理吞吐量。如果增加更多 GPU 不会使延迟比使用第一个 GPU 时更大，则此做法可行。

通过实例了解吞吐量

奥迪赞助了许多体育赛事。SAP 开发了一个用于跟踪广告投放的 DL 解决方案，名叫 SAP Brand Impact，而作为 SAP 的重要客户，奥迪抢先体验了该解决方案（图 4）。

图 4：现场直播节目中的图像识别

来源：NVIDIA

奥迪利用 SAP Brand Impact 开发了自己的深度学习模型。奥迪使用 CUDA 在 NVIDIA DGX-1 服务器上训练它的模型，然后使用 TensorRT 优化模型的推理性能。结果，性能比仅使用 CPU 的解决方案提高了 40 倍，而且每小时成本降低了 32 倍。由于可以实时获得可查证的准确结果，因此 SAP Brand Impact 在节目仍在播出时就能向客户提供结果。

Energy Efficiency（能效）

随着深度学习 (DL) 加速器的性能不断提升，DL 加速器的功耗也越来越高。要想让深度学习解决方案带来投资回报 (ROI)，仅关注系统的推理性能并不足够。功耗可能会迅速增加向客户提供服务的成本，因此，关注设备和系统的能效变得更有必要。

在某些场合下，需要密集地处理数据以便用自然的声音智能地回答问题，而语音处理恰好就是这样一种解决方案。能实时处理语音的数据中心推理功能无疑需要使用许多个机架的计算机，从而影响到公司的总体拥有成本 (TCO)。因此，业界开始使用每瓦特推理次数（越高越好）来衡量运营成效。超大规模数据中心正设法最大程度地提高能效，也即在固定的功耗预算下提供尽可能多的推理次数。

简单看看哪个处理器具有更低的功耗并不能解决问题。例如，如果一个处理器的功耗为 200W，而另一个为 130W，这并不一定表示 130W 的系统更好。如果 200W 的解决方案完成任务的速度快 20 倍，则它的能效更高。

每瓦特推理次数还取决于训练和推理中的延迟因素。能效不仅取决于一段时间内的纯功耗，还取决于同一段时间内的吞吐量。要说明 PLASTER 的各个元素如何相互关联和必须如何综合考虑这些元素以便全面了解推理性能，能效是另一个很好的例子。

测量能效

对于产生式推理，能耗限制了可用的计算资源。电费和冷却费等运营费用 (OPEX) 直接决定了能否在保持质量的同时扩展云服务以便每秒处理更多次推理。扩展端点的推理能力可能会影响端点内部电池的成本和重量、两次充电之间提供的推理次数和质量等等。

通过实例了解能效

科大讯飞是中国领先的语音识别技术提供商，用户人数超过 9.1 亿。科大讯飞基于云的语音识别服务必须在 200 毫秒内应答用户，以便带给用户真实自然的语音应答体验。它的推理服务运行在搭载了 NVIDIA Tesla P4 GPU 显卡的服务器上。与仅使用 CPU 的服务器相比，科大讯飞基于云的语音识别推理即服务现在可以处理多 10 倍的并发请求。系统不仅能处理多 10 倍的请求，而且准确性改善了 20%、运营 TCO 降低了 20%。TCO 降低的很大一部分原因是每瓦特推理次数性能有所改善。

Rate of Learning（学习频率）

近年来，许多企业开始借助功能更强大的系统和更高级别的编程工具来实施“开发与运维”(DevOps) 行动，以便将开发与运维更紧密地联系起来。虽然深度学习目前仍未成熟，但许多正等待利用 DL 的学术界、政界和商界机构并非如此。它们寻求的不是接受不切实际的训练且保持不变的推理引擎。“AI”由两个词组成，其中一个是智能，因此，用户将希望神经网络能在合理的期限内学习和适应。要使复杂的 DL 系统获得商业界的青睐，软件工具开发者必须支持“开发与运维”(DevOps) 行动。

各类组织正不断试验深度学习技术和神经网络，同时学习如何更有效地构建和实施 DL 系统。由于推理服务会收集新的数据，并且会不断发展和变化，因此必须定期重新训练 DL 模型。有鉴于此，IT 组织和软件开发者必须提升模型接收新数据和重新训练的频率。多 GPU 服务器配置已将深度学习训练时间从数天和数周缩短到数分钟和数小时。更快的训练时间意味着开发人员可以更频繁地重新训练网络，以改善准确性或保持高准确性。如今实施的一些深度学习系统已经做到每天重新训练神经网络多次。

可编程性也会影响学习频率。为了简化开发人员的工作流程，Google 和 NVIDIA 最近宣布将 TensorFlow 与 TensorRT 进行集成。开发人员可以从 TensorFlow 框架中调用 TensorRT，以优化训练后的网络，使其在 NVIDIA GPU 上高效运行。更轻松地集成训练和推理的能力使得深度学习能够成为 DevOps 解决方案，从而帮助组织在不断改进其 DL 模型时快速实施更改。

测量学习频率

从以下方面来测量学习频率：

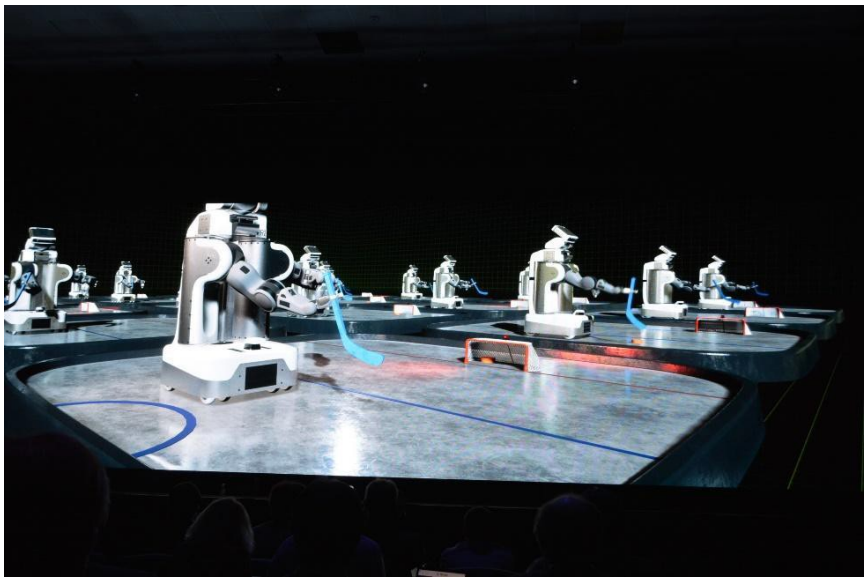
- 对于训练：吞吐量和模型准确性的改善
- 对于生产：吞吐量、模型准确性和延迟的改善
- 对于训练和生产这两者：可编程性、模型大小和能效的改善

借助 DevOps，DL 组织可以吸收和整合深度学习技术的新进展，以在上述方面取得最大改善。

通过实例了解学习频率

NVIDIA 创造了一个名叫 Isaac Sim 的模拟环境，以训练它的 AI 机器人 Isaac（图 5）。NVIDIA 已使用其 Holodeck 项目（一个在物理上精确的协同式虚拟现实 (VR) 环境）演示了此模拟环境。在该演示中，虚拟机器人学习如何打曲棍球。每当一个机器人完成迭代时，它学到的知识会传给其他虚拟机器人，以加快整个学习过程。

图 5：NVIDIA Isaac Sim



在 Isaac Sim 环境中学习的虚拟机器人 来源：NVIDIA

开发人员可以使用 Isaac Sim 在完全集成和高保真度的模拟环境中训练和测试虚拟机器人，而且速度比真实世界要快。工程和测试过程在数分钟而不是数月内就能完成。模拟完成之后，即可将训练后的系统（大脑）转移到物理机器人。

PLASTER：一个与深度学习性能有关的框架

深度学习正从理论进入到早期应用阶段。业界必须评估在 DL 发展成为主流技术的过程中所需要的支持。支持 DL 所需的算法、软件和硬件快速发展，因而需要一个框架来了解面临的挑战和打造可应对这些挑战的环境。

PLASTER 源自 NVIDIA 用来描述深度学习性能的关键要素的评量标准。PLASTER 的元素包括：Programmability（可编程性）、Latency（延迟）、Accuracy（准确性）、Size of model（模型大小）、Throughput（吞吐量）、Energy efficiency（能效）和 Rate of learning（学习频率）。良好的 DL 项目设计应考虑所有这些因素，在它们之间正确地进行必要的取舍，以成功实施 DL 系统。PLASTER 的元素阐明了业界在创造 DL 解决方案时面临的挑战。所有这些元素都不是独立的，而且全都很重要。

PLASTER 框架对于整个 DL 模型开发和部署周期很重要。PLASTER 对于运行时推理尤其重要。生产环境具有必须满足的特定参数 - 从最基本的推动 ROI 的准确性和性能到有关延迟和吞吐量的 SLA，再到涉及监管法例和公司规章的合规性。

将 PLASTER 视为组织原则的组织可以取得以下三项成果，进而推动 DL 走进大众市场。DL 组织可以

- 更好地管理 DL 系统的性能
- 更高效地利用开发人员的时间
- 在 DL 组织中创造 DevOps 环境以支持客户所需的产品和服务

深度学习在其产品生命周期的早期阶段中是一个复杂的问题。将 PLASTER 用作框架的组织可以更好地了解和管理深度学习性能的重要方面。