专题出品人：周经森（Kingsum Chow）博士

计算机软硬件优化首席科学家、高级首席工程师

周经森（Kingsum Chow），计算机软硬件优化首席科学家、高级首席工程师。曾就职于美国英特尔公司和中国阿里巴巴集团，2023 年加入浙江大学软件学院（宁波）。二十年来与十余家世界 500 强高科技企业合作，共同开发了世界级硬件性能优化技术的发展。

曾作为项目总监主持备受瞩目的云计算蓝图项目（IntelCloudBlueprint），该项目由英特尔和甲骨文的首席执行官于 2015 年共同宣布，吸引了超过 4 万名开发者的参与，为云计算行业绘制了全新的技术蓝图，对行业发展产生了深远影响。

自 2016 年加入阿里巴巴，为中国的性能优化技术发展做出了巨大贡献。2018 年，其作为唯一一名加入 Java 全球管理组织 JavaCommunityProcess（JCP）最高执行委员会 JCP-EC 的中国企业（阿里巴巴）代表，参与制定了 Java 的全球标准。

周博士在 CPU 利用率报告不准确（数据普遍误解）方面发表的研究，引起了业界和学术界的广泛关注。周博士拥有超过 30 年的软硬件协同优化的工业实践经验，培养了大批优秀的系统性能优化人才。至今已获授权中国专利 11 项，美国专利 24 项，发表学术论文 127 篇，在过去 6 年的 QCon 中国大会上发表 2 场主题演讲，出品 2 场软件系统性能优化主题讲座。

专题：LLM 时代的性能优化

地点：中优大宴会厅 1（三层）

LLM 时代的性能分析正在 CPU 和 GPU 平台上如火如荼地进行，这是为了更好地理解 LLM 在不同计算环境下的性能表现，从而找到最佳的应用策略和优化方法，为 LLM 的应用和发展提供更多的可能性。

LLM 训练推理加速在阿里巴巴的实践

by 杨斯然
阿里巴巴
高级技术专家

by 刘侃
阿里巴巴
高级技术专家

在大规模语言模型 (LLM) 的训练和推理实践中，工程和算法需求间存在许多需要细心权衡的问题。这些问题涉及到从软硬件协同优化，到分布式处理，以及至算法工程 Co-design 等多个领域。为了解决这些挑战，我们深入研究了不同的应用场景和流量特性，并因此对我们的系统进行了全面优化。

尽管从 HuggingFace 上可以拿到 LLaMA 等模型的代码，但用自己的数据训练一个 LLaMA 模型对个人用户或中小型组织并不是一件低成本且简单的工作。Megatron-LLaMA 框架基于有着成熟社区的 Megatron-LM 项目，充分降低了 LLaMA 等开源模型的训练成本和门槛。

Megatron-LLaMA 中包含了：

1. 基于社区 HuggingFace LLaMA 模型的续训能力；

2. 在不同规模、硬件上大模型训练的最佳实现；

3. 针对训练成本的性能优化。

在推理方面，基于 TensorRT LLM，我们进一步研发了 Maga Transformer 框架和 LLM 推理平台产品。这个系统通过应用一套统一的框架和多种不同的策略，成功地优化了不同推理场景下的成本和用户体验。我们全面支持各种开源和内部 LLM 模型，以 LLaMA 7B 和 Falcon 180B 为例，我们的系统达到了xxx的性能水平，这一结果充分证明了我们的系统在处理大规模数据时的卓越性能。

此外，我们的系统还广泛支持各种量化和剪枝方法，以及 KVCache Reuse、Speculative Decoding、Medusa 等辅助优化方法，这可以进一步提高推理效率并降低存储和计算需求。这一系列的特性使得我们的系统在处理大规模语言模型推理时表现出强大的竞争力。

解析云原生数仓 ByteHouse 如何构建高性能向量检索技术

by 田昕晖 博士
字节跳动
技术专家

向量检索被广泛使用于以图搜图、内容推荐以及大模型推理等场景。随着业务升级与 AI 技术的广泛使用，用户期望处理的向量数据规模越来越大，对向量数据库产品的稳定性、易用性与性能需求也越来越高。为此火山引擎 ByteHouse 团队基于社区 ClickHouse 进行技术演进，提出了全新的向量检索功能设计思路，满足业务对向量检索稳定性与性能方面的需求。

演讲提纲：

1. 向量检索概念以及在 LLM 场景的应用
2. 当前业界向量数据库发展情况
3. ClickHouse 结合向量检索的优势，以及社区当前向量检索局限性与性能问题分析
4. ByteHouse 向量检索功能设计思路介绍
5. 性能比较

听众收益点：

- 向量搜索的使用场景
- 向量搜索与 OLAP 结合的优势
- 如何在 OLAP 系统中实现高效向量搜索

基于时间序列数据预测模型的智能量化交易系统性能优化实践

by 黄益聪
榷间科技有限公司
CEO

金融市场的行情数据，如股票价格、成交量、交易队列等是典型的时间序列数据，具有很强的时间性和顺序依赖性。智能量化交易系统需要对市场上高频产生的时间序列数据进行处理计算，输入深度学习模型进行预测，执行交易策略，生成交易行为进行交易。整个过程需要覆盖全市场一万以上的品种，并且需要在很小的时间窗口，比如秒级完成。进一步的，我们使用了多语言进行系统开发。其中数据采集模块使用了 C++ 以达到高性能，交易策略引擎使用了 Java Spring Boot 搭建服务，AI 模型使用了 Python 基于 TensorFlow 和 Torch 框架。

业务需求的系统低延迟计算和多语言系统模块的交互，给我们的性能优化带来了挑战。这次分享，将带来我们对系统全链路从数据采集-数据计算-模型预测-交易下单，全流程进行优化的实践分享，包括怎样高效的在 Java 处理计算 C++ 高频产生的时间序列数据，怎么降低高频产生、长生命周期数据对 Java GC 的影响，怎么高效部署调用低延迟、多模型、多版本的 AI 模型预测服务，系统故障的数据断点快速恢复等。

演讲提纲：

1. 背景与项目概况
   - 量化交易系统介绍
   - 项目技术架构
   - C++/ Java / Python多语言交互
2. 全链路数据流优化
   - 实时行情数据收集与处理
   - 低延迟、高吞吐性能挑战
   - 尝试的优化手段：GC Tuning，Direct Buffer
   - 我们的解决方案
   - 性能优化效果
3. 服务化 AI 模型预测
   - Tensorflow 模型性能优化实践
   - Torch 模型转换
4. 总结与展望
   - 复杂模型和低延迟预测性能的权衡
   - 目标展望：更快、更准

听众收益点：

- 构建高性能、低延迟的智能量化交易系统
- 多语言开发的复杂系统的全链路性能分析和优化
- AI 模型在智能量化交易系统的实践

大模型时代：最大化CPU价值的优化策略

by 何普江
英特尔
数据中心与人工智能事业部AI软件架构师

本次演讲将探讨在大语言模型时代充分利用 CPU 资源的关键策略。具体介绍一些结合硬件特性的优化方法，例如利用 CPU 的多核特性、采用并行计算和 AMX 指令集扩展技术来提高处理速度。

此外还将介绍一种结合 CPU 和 GPU 的投机采样方法，通过在 CPU 上运行部分计算任务，充分利用 CPU 资源并减少对 GPU 的依赖。最后，我将分享一些最新的性能情况，让您了解这些优化策略的实际效果。通过这些方法，您将能够更好地利用 CPU 资源，提高模型推理速度，以更快速高效的实现生成式模型部署落地。

演讲提纲：

1. 大语言模型时代为什么需要最大化 CPU 价值
2. CPU 上的大模型优化策略
   - 大语言模型计算特点
   - CPU 硬件特性概览
   - 优化方法
   - 从向量化到张量化
   - 从并行执行到分布式推理
   - 低精度优化
   - 深入 CPU 微架构的软件优化
   - 各优化策略的实际性能数据对比及效果展示
3. 结合 CPU 和 GPU 的投机采样方法
   - CPU 和 GPU 协同工作的背景
   - 投机采样技术的介绍
   - 利用 CPU 进行部分计算任务的优势
   - 优化方法：选择合适的投机采样策略、任务调度等
4. 总结与展望
   - 各优化方法的核心优势与局限性总结
   - 对未来大语言模型时代的展望与挑战

听众收益点：

- 理解并结合硬件特性进行优化，提高模型推理速度和处理能力
- 了解 CPU 上的最新性能情况，为实际业务的大模型线上部署提供更多选择
- 掌握结合 CPU 和 GPU 协同工作的优化策略，减少对 GPU 的依赖，提高资源利用率