极客大学

极客大学机器学习训练营
GPU 编程和 TensorRT 基础

王然

众微科技 AI Lab 负责人

二〇二一年五月八日

大纲

大纲

大纲

GPU 和 CPU 架构基本不同

极客大学

- GPU 和 CPU 架构最基本的不同在 SIMT 的引入；
- SIMT=Single Introduction Multiple Thread；
- 对于同样的（底层）的命令，由多个 Cuda Core 执行。

3090 的配置

见官网资料。

GPU 的局限：SIMT

极客大学

- GPU 没有取代 CPU 的原因主要是 GPU 的 CUDA Core 无法独立进行操作；
- CUDA Core 以 Warp 为单位；
- 在一个 Warp 单位中，所有的 Core 必须执行**同样**的操作；
- 在极端情况下，即使是只有一个位置需要进行操作，我们仍然需要调用整个 Warp 的算力；
- Warp 大小一般为 32 或 64。
- 直接结果：任何关于 IF 的语句都可能造成灾难性的后果。

GPU 的局限：内存和显存

极客大学

- GPU 无法直接对内存中的内容进行读取；
- 所有数据必须从内存读取到显存当中才能进行处理；
- 计算结果必须从显存读取到内存当中才能进行应用；
- 内存和显存互相的读取速率可以非常低；
- 一种提升方法是运用 stream 使计算和数据传输可以同时进行；
- 另一种提升方法是使用 pinned memory。

GPU 的局限：Multi-stream Processor

极客大学

- 与 CPU 不同，决定 GPU 整体运算的大脑是 Multi-stream Processor；
- Multi-stream Processor 决定了各种复杂的 GPU 运算操作；
- Multi-stream Processor 的局限之一是它不够智能：这使得 GPU 和 CPU 交流时，会出现各种奇怪的现象；
- Multi-stream Processor 的局限之一是它又太智能了：这使得我们难以预测真实的表现。

大纲

极客大学

基本语法和用例

极客大学

- CUDA 的编译；
- 向量相加；
- 向量求和；
- 细节见代码。

GPU 的其他考虑

极客大学

- Host Memory 考虑；
- 各种不同显存的考虑；
- Stream 和协同；
- CUDA Core 编程；
- 具体资料可以见Wilt (2013)。

大纲

大纲

极客大学


Wilt, Nicholas (2013). *The cuda handbook: A comprehensive guide to gpu programming*. Pearson Education.

极客大学

Thanks!