ACELERACIÓN GPU
PARA DEEP LEARNING EN AMD

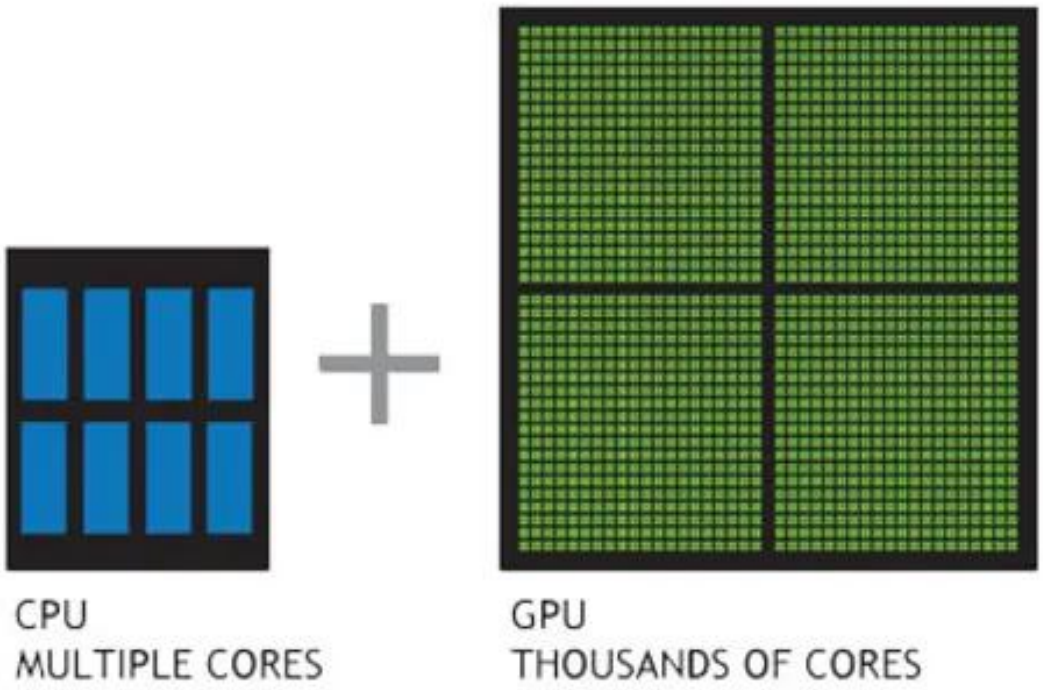
Emilio Barragán Rodríguez

¿QUÉ ES LA ACELERACIÓN GPU?

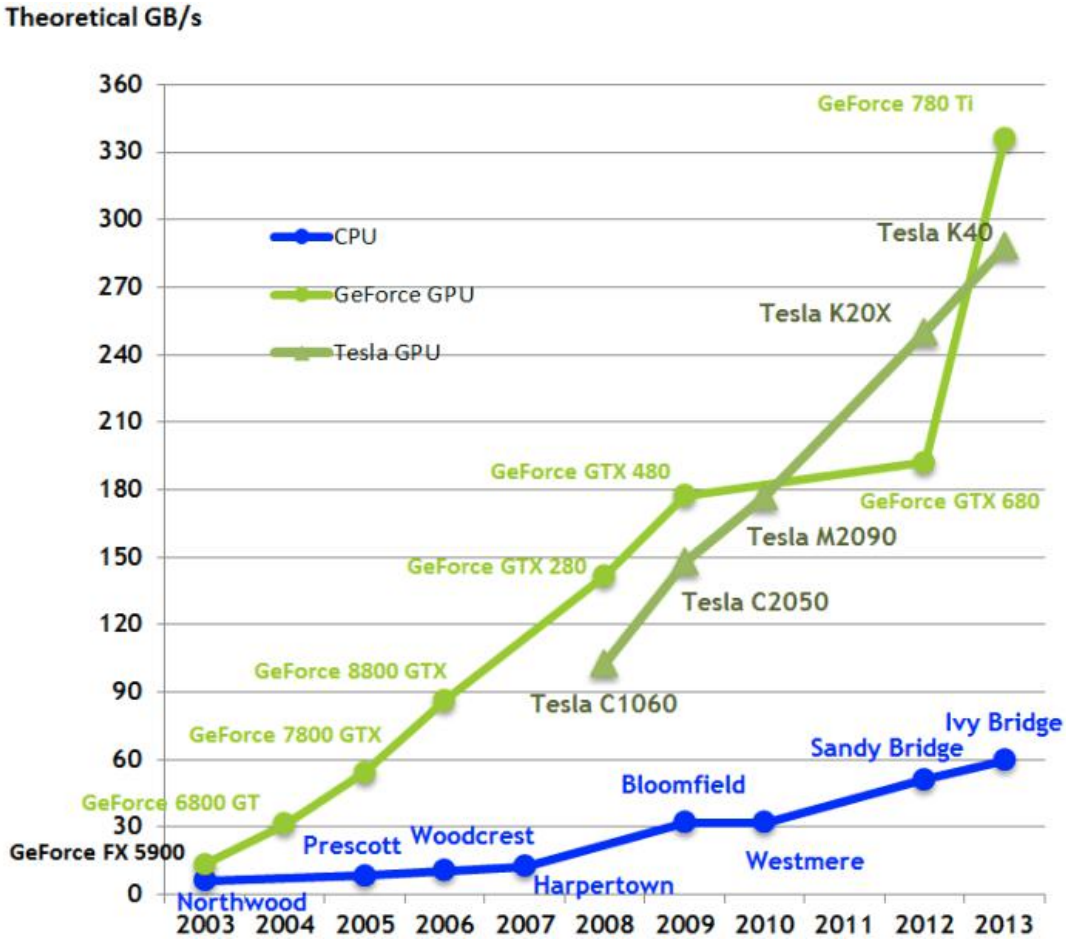
Las GPUs empezaron a fabricarse para soportar la carga de trabajo que daban las imágenes y videos (los tensores de datos).
Las empresas se dieron cuenta que esas GPUs no solo se podrían usar para videojuegos, edición de vídeo y foto, etc. Si no que también se podrían usar para el procesado de mucha cantidad de datos de otra índole.
Así que básicamente la aceleración por GPU es usar la GPU para el procesado de grandes cantidades de datos (para pocos datos no renta usar GPU)

GPU VS. CPU

La CPU está diseñada para el procesamiento en serie: se compone de unos pocos núcleos muy complejos que pueden ejecutar unos pocos programas al mismo tiempo. En cambio, la GPU tiene cientos o miles de núcleos sencillos que pueden ejecutar cientos o miles de programas específicos a la vez. Las tareas de las que se encarga la GPU requieren un alto grado de paralelismo.



Estaríamos comparando un Ferrari (llega rápido a su destino pero no aguanta carga) con un camión (más lento pero soporta mucha más carga de trabajo).



[Comparison of bandwidth for CPUs and GPUs over time.]

SUPREMACÍA DE NVIDIA

NVIDIA llegó antes al mercado de la aceleración por GPU, por lo tanto la mayoría de data centers y equipos de procesado de datos tienen hardware NVIDIA. Esto complica mucho la entrada de AMD en ese mercado ya que supondría un gran problema de integración y sobre todo económico para las empresas y organizaciones.

También NVIDIA está esforzándose mucho en desarrollar hardware específico y bastante bueno para esta tarea por lo que sigue mejorando. Así, también está desarrollando drivers muy avanzados y optimizados específicos para la aceleración por GPU, como es el caso del driver CUDA. AMD por su parte, viendo el panorama actual no invierte tanto en este mercado, pero no sabemos qué hará en un futuro, o si ya está preparando algo.

Debido a todos estos elementos, los frameworks actuales más importantes están pensados y/o optimizados para usarse con NVIDIA, por lo que entramos en un círculo vicioso en el que AMD cada vez tiene menos posibilidades.

ENTONCES... ¿QUÉ PASA SI TENGO AMD?

Debemos tener en cuenta que para deep learning usaremos Keras. Keras funcionará si funciona su backend (tensorflow) por lo que en realidad nuestro objetivo es conseguir que tensorflow funcione.

La primera opción si te sobra el dinero es pasarte a NVIDIA (todo será más fácil y probablemente funcionará mejor). Pero si eres alguien como yo que se compró AMD porque estaba mejor de precio sin saber todo esto, te propongo varias soluciones:

- ROCm (Radeon Open Compute)

- plaidML

- Cloud

ROCm

ROCm es un framework desarrollado por AMD. Ha sido diseñado para ser una plataforma modular de aceleración GPU.Este diseño modular permite a las principales compañías hardware crear drivers que soporten el framework ROCm.

ROCm también está diseñado para soportar cualquier lenguaje de programación, y es muy fácil añadir soporte para otros lenguajes que no estén ya implementados.

Actualmente ROCm soporta variossistemas operativos:

- Ubuntu
- CentOS
- SLES 15 SP1
- RHEL (Red Hat Enterprise)

Algunas de las características más importantes son:

-Multi-GPU
-Concurrencia de procesos
-Grandes asignaciones de memoria
-Dynamics and offline-compilation support
-Operaciones peer-to-peer multi-GPU
-Systems-management API y herramientas

GPUs soportadas:

- GFX8 GPUs

- "Fiji" chips, por ejemplo: AMD Radeon R9 Fury X y Radeon Instinct MI8

- "Polaris 10" chips, por ejemplo: AMD Radeon RX 580 y Radeon Instinct MI6

- GFX9 GPUs

- "Vega 10" chips, por ejemplo: AMD Radeon RX Vega 64 y Radeon Instinct MI25

- "Vega 7nm" chips, por ejemplo: Radeon Instinct MI50, Radeon Instinct MI60 o AMD Radeon VII

GPUs “experimentales”:

- GFX8 GPUs

- "Polaris 11" chips, por ejemplo: AMD Radeon RX 570 y Radeon Pro WX 4100

- "Polaris 12" chips, por ejemplo: AMD Radeon RX 550 y Radeon RX 540

- GFX7 GPUs

- "Hawaii" chips, por ejemplo: AMD Radeon R9 390X y FirePro W9100

plaidML

Creada por una empresa llamada Vertex.AI, que luego compró Intel.

plaidML es un compilador de tensores (al estilo tensorflow) portable, de software libre, que habilita el deep learning en portátiles, dispositivos empotrados o cualquier dispositivo en los que la computación por hardware (en nuestro caso la aceleración por GPU) no está bien soportada, o el software disponible tiene muchas restricciones por culpa licencias que no nos gustan.

Se puede usar como backend de Keras (con lo que cumpliríamos nuestro objetivo). Funciona realmente bien en GPUs y no requiere el uso del driver CUDA ni cDNN en hardware NVIDIA, obteniendo aún así un rendimiento equiparable.

plaidML funciona en la mayoría de sistemas operativos (Linux, macOS y Windows), pero si tu sistema operativo no está soportado, puedes buildear plaidML usando su código fuente.

Usa OpenCL como driver para la aceleración por GPU. Lo bueno de este driver es que es libre.

CLOUD: LA FORMA FÁCIL

Como todos sabemos, el cloud está ahora muy de moda y la verdad que tiene un futuro prometedor (si no contamos con el impacto climático). Pues por supuesto hay muchas empresas que ofrecen máquinas en el cloud sobre las que poder trabajar con aceleración por GPU aún teniendo AMD.

La mayoría de estos servicios (usados profesionalmente) son de pago. Algunos tienen períodos de prueba en los que te dejan probarlo gratis y luego ya pagas. Otros como por ejemplo Google Colab es gratuito pero tiene una versión pro que te da más y mejores funcionalidades.

Algunos ejemplos de servicios en el cloud son:

- Google Cloud

- Google Colab

- Microsoft Azure

- Amazon Web Service

ENLACES DE INTERÉS

Deep Learning en AMD GPUs

- PlaidML

Installation Instructions - PlaidML

ROCm Core Technology

tensorflow-rocm · PyPI

ROCmSoftwarePlatform/tensorflow-upstream: TensorFlow ROCm port

ROCm, a New Era in GPU Computing

Why doesn't AMD compete against NVIDIA in the deep learning GPU market? - Quora

Do we really need GPU for Deep Learning? - CPU vs GPU

On the state of Deep Learning outside of CUDA's walled garden

AWS | Cloud Computing - Servicios de informática en la nube

Servicios de cloud computing | Google Cloud

Google Colab

Microsoft Azure