Accelerators

Compute accelerators are the workhorses of the ML training. At the beginning there were just GPUs. But now there are also TPUs, IPUs, FPGAs, HPUs, QPUs, RDUs and more are being invented.

There exist two main ML workloads - training and inference. There is also the finetuning workload which is usually the same as training, unless a much lighter LORA-style finetuning is performed. The latter requires significantly fewer resources and time than normal finetuning.

In language models during inference the generation is performed in a sequence - one token at a time. So it has to repeat the same `forward` call thousands of times one smallish `matmul` (matrix multiplication or GEMM) at a time. And this can be done on either an accelerator, like GPU, or some of the most recent CPUs, that can handle inference quite efficiently.

During training the whole sequence length is processed in one huge `matmul` operation. So if the sequence length is 4k long, the training of the same model will require a compute unit that can handle 4k times more operations than inference and do it fast. Accelerators excel at this task. In fact the larger the matrices they have to multiply, the more efficient the compute.

The other computational difference is that while both training and inference have to perform the same total amount of `matmul`s in the `forward` pass, in the `backward` pass, which is only done for training, an additional 2x times of `matmul`s is done to calculate the gradients with regards to inputs and weights. And an additional `forward` is performed if activations recomputation is used. Therefore the training process requires at 3-4x more `matmul`s than inference.

Subsections

- Troubleshooting NVIDIA GPUs

1

Bird's eye view on the high end accelerator reality

While this might be changing in the future, unlike the consumer GPU market, as of this writing there aren't that many high end accelerators, and if you rent on the cloud, most providers will have more or less the same few accelerators to offer.

GPUs: - As of today, ML clouds/HPCs started transitioning from NVIDIA A100s to H100s and this is going to take some months due to the usual shortage of NVIDIA GPUs. - AMD's MI250 started popping up here and there, but it's unclear when it'll be easy to access those. MI300X is promised to start being available already in March 2024 at some Tier 2 cloud providers.

HPU: - Intel's Gaudi2 are starting to slowly emerge on Intel's cloud - there is a huge lineup. It's also available on-premise simplementations via Supermicro, WiWynn, and soon others.

IPU: - And there is Graphcore with their IPU offering. You can try these out in Paperspace through their cloud notebooks.

TPU: - Google's TPUs are, of course, available but they aren't the most desirable accelerators because you can only rent them, and the software isn't quite easily convertible between GPUs and TPUs, and so many (most?) developers remain in the GPU land, since they don't want to be locked into a hardware which is a Google monopoly.

On Pods and racks: - Cerebras' WaferScale Engine (WSE) - SambaNova's DataScale - dozens of different pod and rack configs that compose the aforementioned GPUs with super-fast interconnects.

That's about it as Q1-2024.

Glossary

- CPU: Central Processing Unit
- FPGA: Field Programmable Gate Arrays
- GPU: Graphics Processing Unit
- HBM: High Bandwidth Memory
- HPC: High-performance Computing
- HPU: Habana Gaudi AI Processor Unit
- IPU: Intelligence Processing Unit
- MME: Matrix Multiplication Engine
- QPU: Quantum Processing Unit
- RDU: Reconfigurable Dataflow Unit
- TPU: Tensor Processing Unit

The most important thing to understand

I will make the following statement multiple times in this book - and that it's not enough to buy/rent the most expensive accelerators and expect a high return on investment (ROI).

The two metrics for a high ROI for ML training are: 1. the speed at which the training will finish, because if the training takes 2-3x longer than planned, your model could become irrelevant before it was released - time is everything in the current super-competitive ML market. 2. the total $$ spent to train the model, because if the training takes 2-3x longer than planned, you will end up spending 2-3x times more.

Unless the rest of the purchased/rented hardware isn't chosen carefully to match the required workload chances are very high that the accelerators will idle a lot and both time and $$ will be lost. The most critical component is network, then storage, and the least critical ones are (CPU and CPU memory).

If the compute is rented one usually doesn't have the freedom to choose - the hardware is either set in stone or some components might be replaceable but with not too many choices. Thus there are times when the chosen cloud provider doesn't provide a sufficiently well matched hardware, in which case it's best to seek out a different provider.

If you purchase your servers then I recommend to perform a very indepth due diligence before buying.

Besides hardware, you, of course, need software that can efficiently deploy the hardware.

We will discuss both the hardware and the software aspects in various chapters of this book. You may want to start here and here.

What Accelerator characteristics do we care for

Let's use the NVIDIA A100 spec as a reference point in the following sections.

	A100 80GB PCIe	A100 80GB SXM
FP64	9.7 TFLOPS	
FP64 Tensor Core	19.5 TFLOPS	
FP32	19.5 TFLOPS	
Tensor Float 32 (TF32)	156 TFLOPS | 312 TFLOPS*	
BFLOAT16 Tensor Core	312 TFLOPS | 624 TFLOPS*	
FP16 Tensor Core	312 TFLOPS | 624 TFLOPS*	
INT8 Tensor Core	624 TOPS | 1248 TOPS*	
GPU Memory	80GB HBM2e	80GB HBM2e
GPU Memory Bandwidth	1,935 GB/s	2,039 GB/s
Max Thermal Design Power (TDP)	300W	400W ***
Multi-Instance GPU	Up to 7 MIGs @ 10GB	Up to 7 MIGs @ 10GB
Form Factor	PCIe Dual-slot air-cooled or single-slot liquid-cooled	SXM
Interconnect	NVIDIA® NVLink® Bridge for 2 GPUs: 600 GB/s ** PCIe Gen4: 64 GB/s	NVLink: 600 GB/s PCIe Gen4: 64 GB/s
Server Options	Partner and NVIDIA-Certified Systems™ with 1-8 GPUs	NVIDIA HGX™ A100-Partner and NVIDIA-Certified Systems with 4,8, or 16 GPUs NVIDIA DGX™ A100 with 8 GPUs

Figure 1: nvidia-a100-spec

source

TFLOPS

As mentioned earlier most of the work that ML training and inference do is matrix multiplication. If you remember your algebra matrix multiplication is made of many multiplications followed by summation. Each of these computations can be counted and define how many of these operations can be performed by the chip in a single seconds.

This is one of the key characteristics that the accelerators are judged by. The term TFLOPS defines how many trillions of FloatingPointOperations the chip

4

can perform in a second. The more the better. There is a different definition for different data types. For example, here are a few entries for A100:

Data type TFLOPS	w/o Sparsity	w/ Sparsity
FP32	19.5	n/a
Tensor Float 32 (TF32)	156	312
BFLOAT16 Tensor Core	312	624
FP16 Tensor Core	312	624
FP8 Tensor Core	624	1248
INT8 Tensor Core	624	1248

footnote: INT8 is measured in TeraOperations as it's not a floating operation.

footnote: the term FLOPS could mean either the total number of Floating-PointOperations, e.g. when counting how many FLOPS a single Transformer iteration takes, and it could also mean FloatingPointOperations per second - so watch out for the context. When you read an accelerator spec it's almost always a per second definition. When model architectures are discussed it's usually just the total number of FloatingPointOperations.

So you can see that int8 is 2x faster than bf16 which in turn is 2x faster than tf32.

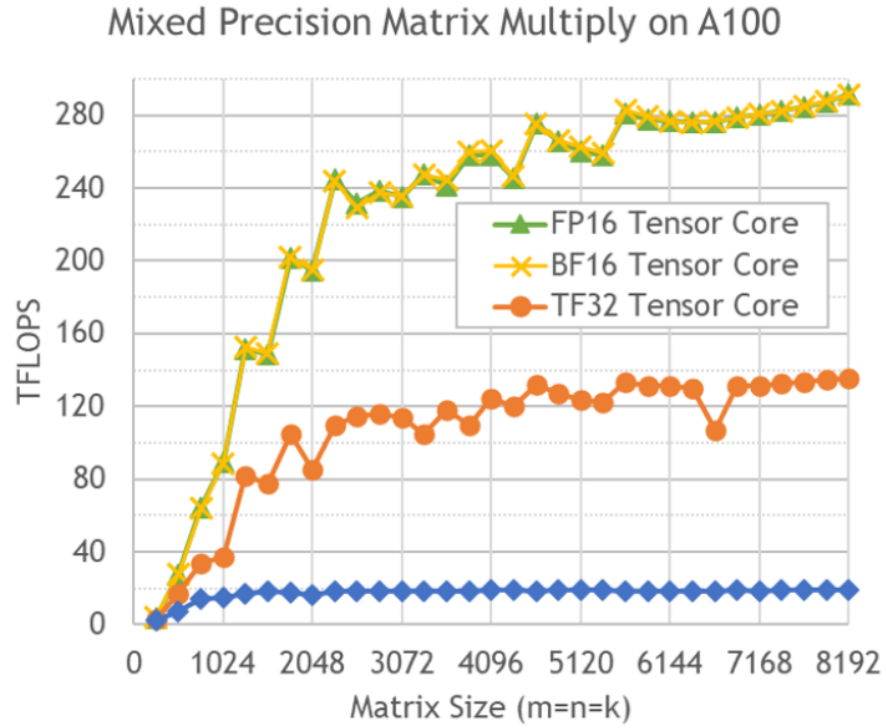Moreover, the TFLOPs depend on the matrices size as can be seen from this table:

Figure 6. Mixed-precision matrix multiply on A100 with cuBLAS.

Figure 2: nvidia-a100-matmul-tflops

source

As you can see the difference in performance is non-linear due to the tile and wave quantization effects.

Let's look at the supported dtypes and the corresponding theoretical peak TFLOPS specs across the high end accelerators (w/o sparsity):

Accelerator TFLOPS	fp32	tf32	fp16/bf16	fp8	int8
NVIDIA A100 SXM	19.5	156.0	312	624	624
AMD MI250	45.3	X	362	X	362
AMD MI250X	47.9	X	383	X	383
NVIDIA H100 SXM	67.0	494.5	989	1979	1979
NVIDIA H200 SXM	67.0	494.5	989	1979	1979
NVIDIA H100 PCIe	51.0	378.0	756	1513	1513

Accelerator TFLOPS	fp32	tf32	fp16/bf16	fp8	int8
NVIDIA H100 dual NVL	134.0	989.5	989	3958	3958
AMD MI300X	163.4	653.7	1300	5220	2600
Intel Gaudi2	V	V	V	V	V

footnote: Intel Gaudi2 doesn't plan to publish TFLOPS specs as of this writing, but it does support FP32, TF32, BF16, FP16 & FP8, INT8 and INT16. This blog posts reports measuring ~400TFLOPS for fp16/bf16 - but, of course, this number can't be compared to theoretical peak so it doesn't belong to this table - guessing, it's probably in the 600-1000TFLOPS range.

footnote: when looking at specs be very careful at which numbers you're reading - many vendors often publish TFLOPS with sparsity, as they are ~2x bigger, but if they even indicate this they often do it in small print. I had to ask NVIDIA to add a note to their H100 spec that those numbers were w/ sparsity as they originally didn't mention this important technical fact. And 99% of the time as of this writing you will be not using sparsity and thus the actual theoretical TFLOPs that you care for most of the time are w/o sparsity (i.e. the table above).

footnote: also beware that if accelerator A publishes a higher TFLOPS than accelerator B, it doesn't mean A is faster. These are theoretical numbers which not only can never be achieved in practice - the actual TFLOPS efficiency (HFU) can vary a lot from vendor to vendor or even for the same vendor's different accelerator architectures.

Maximum Achievable FLOPS

Theoretical peak FLOPS is what gets published on the accelerator's spec. And it's calculated as:

```
Theoretical FLOPS = compute_unit_clock_speed * flops_per_clock_cycle_per_compute_unit
* num_compute_units
```

where: - `compute_unit_clock_speed` - how many times the compute unit clock ticks per second in Hz - `flops_per_clock_cycle_per_compute_unit` - the number of operations the compute unit can execute per clock cycle. - `num_compute_units` - how many units there is in the device

The problem with the advertised theoretical peak FLOPS is that they are **very** theoretical and can't be achieved in practice even if all the perfect conditions have been provided. Each accelerator has its own realistic FLOPS which is not advertised and there are anecdotal community reports that do their best to find the actual best value, but I'm yet to find any official reports.

If you find solid reports (papers?) showing the actual TFLOPS one can expect from one or more of the high end accelerators discussed in this chapter please

kindly submit a PR with this information. The key is to have a reference to a source that the reader can validate the proposed information with.

To provide a numerical sense to what I'm talking about is let's take A100 with its 312 TFLOPS bf16 peak performance in the specs of this card. Until the invent of FlashAttention it was known that 150TFLOPS was close to the highest one could get for fp16/bf16 mixed precision regime. And with FlashAttention it's around 180TFLOPS. This is, of course, measured for training LLMs where the network and IO are involved which create additional overheads. So here the maximum achievable peak performance probably lays somewhere between 200 and 300 TFLOPS.

It should be possible to calculate the actual peak TFLOPS by doing a perfectly aligned max-size matrices `matmul` measured on a single accelerator.

XXX: write a small program to do exactly dynamically figuring out the perfect shapes based on the tile and wave quantization effects and max sizes (how?) so that the benchmark isn't hardcoded to a particular accelerator.

Accelerator memory size and speed

The accelerators use High Bandwidth Memory (HBM) which is a 3D version of SDRAM memory. For example, A100-SXM comes with HBM2 at 1.6TBps, and H100-SXM comes with HBM3 at 3.35TBps.

Here are the specs:

Gen	Data Rate (Gbps)	Bandwidth per Device (GBps)	Stack Height	Max. DRAM Capacity (GB)	Max. Device Capacity (GB)
HBM	1.0	128	8	2	16
HBM2	2.0	256	8	2	16
HBM2e	3.6	461	12	3	36
HBM3	6.4	819	16	4	64
HBM3e	9.6	1229	16	4	64

Since HBM is a stack of multiple DRAM chips, the *Stack Height* specifies how many chips are per device.

Typically the more on-device memory the accelerator has the better. At any given time usually most of the model weights aren't being used as they wait for their turn to be processed and thus large memory allows more of the model to be on the accelerator memory and immediately available for access and update. When there is not enough memory, sometimes the model has to be split across multiple accelerators, or offloaded to CPU and/or disk.

Here are the memory specs for the recent high end accelerators (some aren't GA yet):

Accelerator	Memory (GBs)	Type	Bandwidth (TBps)
NVIDIA A100 SXM	80	HBM2e	2.00
NVIDIA H100 SXM	80	HBM3	3.35
NVIDIA H100 PCIe	80	HBM3	2.00
NVIDIA H100 dual NVL	188	HBM3	7.80
NVIDIA GH200 SXM (1)	96	HBM3	4.00
NVIDIA GH200 SXM (2)	141	HBM3e	4.80
NVIDIA H200 SXM	141	HBM3e	4.80
AMD MI250	128	HBM2e	3.28
AMD MI250X	128	HBM2e	3.28
AMD MI300X	192	HBM3	5.30
Intel Gaudi2	96	HBM2e	2.45

Memory speed (bandwidth) is, of course, very important since if it's not fast enough than the compute ends up idling waiting for the data to be moved to and from the memory.

Heat

This is of interest when you buy your own hardware, when you rent on the cloud the provider hopefully takes care of adequate cooling.

The only important practical understanding for heat is that if the accelerators aren't kept cool they will throttle their compute clock and slow everything down (and could even crash sometimes, albeit throttling is supposed to prevent that).

High end accelerators for LLM/VLM workloads

Cloud and in-house accelerators

Most common accelerators that can be either rented on compute clouds or purchased:

NVIDIA: - A100 - huge availability but already getting outdated. - H100 - 2-3x faster than A100 (half precision), 6x faster for fp8, becoming available on all Tier-1 compute clouds. - GH200 - 2 chips on one card - (1) H100 w/ 96GB HBM3 or 144GB HBM3e + (2) Grace CPU w/ 624GB RAM - first units have been reported to become available.

AMD: - MI250 ~= A100 - very few clouds have them - MI300 ~= H100 - just starting to emerge - and mainly on Tier-2 clouds (lots of new startups).

Intel: - Gaudi2 ~= H100 - spec - Currently there is a very low availability on cloud.google.com with a long waiting list which supposedly should be reduced in Q1-2024. AWS has the older Gaudi1 via DL1 instances.

Graphcore: - IPU - available via Paperspace

SambaNova: - DataScale SN30

In-house accelerator clusters

Cerebras: - clusters - systems based on WaferScale Engine (WSE).

Cloud-only solutions

These can be only used via clouds:

Google - TPUs - lock-in, can't switch to another vendor like NVIDIA -> AMD

Cerebras: - Cloud

How to get the best price

Remember that the advertised prices are almost always open to negotiations as long as you're willing to buy/rent in bulk or if renting for a 1-3 years. What you will discover is that the actual price that you end up paying could be many times less than the advertised "public" price. Some cloud providers already include the discount as you choose a longer commitment on their website, but it's always the best to negotiate directly with their sales team. In addition or instead of a $$-discount you could be offered some useful features/upgrades for free.

If your company has venture capital investors - it could help a lot to mention that, as then the cloud provider knows you are likely to buy more compute down the road and more likely to discount more.

Tier 2 clouds are likely to give better prices than Tier 1. Tier 1 as of this writing is AWS, OCI, Azure and GCP.

For the baseline prices it should be easy to find a few good sites that provide an up-to-date public price comparisons across clouds - just search for something like cloud gpu pricing comparison.

When shopping for a solution please remember that it's not enough to rent the most powerful accelerator. You also need fast intra-node and inter-node connectivity and sufficiently fast storage - without which the expensive accelerators will idle waiting for data to arrive and you could be wasting a lot money and losing time.

Accelerators in detail

NVIDIA

Abbreviations:

- CUDA: Compute Unified Device Architecture (proprietary to NVIDIA)

NVIDIA-specific key GPU characteristics: - CUDA Cores - similar to CPU cores, but unlike CPUs that typically have 10-100 powerful cores, CUDA Cores are weaker and come in thousands and allow to perform massive general purpose computations (parallelization). Like CPU cores CUDA Cores perform a single operation in each clock cycle. - Tensor Cores - special compute units that are designed specifically to perform fast multiplication and addition operations like matrix multiplication. These perform multiple operations in each clock cycle. They can execute extremely fast computations on low or mixed precision data types with some loss (fp16, bf16, tf32, fp8, etc.). These cores are specifically designed for ML workloads. - Streaming Multiprocessors (SM) are clusters of CUDA Cores, Tensor Cores and other components.

For example, A100-80GB has:

- 6912 CUDA Cores
- 432 Tensor Cores (Gen 3)
- 108 Streaming Multiprocessors (SM)

AMD

AMD-specific key GPU characteristics: - Stream Processors - are similar in functionality to CUDA Cores - that is these are the parallel computation units. But they aren't the same, so one can't compare 2 gpus by just comparing the number of CUDA Cores vs the number of Stream Processors. - Compute Units - are clusters of Stream Processors and other components

for example, AMD MI250 has: - 13,312 Stream Processors - 208 Compute Units

Intel Gaudi2

[Architecture](#)

- 24x 100 Gigabit Ethernet (RoCEv2) integrated on chip - 21 of which are used for intra-node and 3 for inter-node (so `21*8=168` cards for intra-node (262.5GBps per GPU), and `3*8=24` cards for inter-node (2.4Tbps between nodes)
- 96GB HBM2E memory on board w/2.45 TBps bandwidth per chip, for a total of 768GB per node

A server/node is built from 8 GPUs, which can then be expanded with racks of those servers.

There are no official TFLOPS information published (and from talking to an Intel representative they have no intention to publish any.) They publish the [following benchmarks](https://developer.habana.ai/resources/habana-models-performance/ but I'm not sure how these can be used to compare this compute to other providers.

Comparison: supposedly Gaudi2 competes with NVIDIA H100

API

Which software is needed to deploy the high end GPUs?

NVIDIA

NVIDIA GPUs run on CUDA

AMD

AMD GPUs run on ROCm - note that PyTorch you can use CUDA-based software on ROCm-based GPUs! So it should be trivial to switch to the recent AMD MI250, MI300X, and other emerging ones.

Intel Gaudi

The API is via Habana SynapseAI® SDK which supports PyTorch and Tensor-Flow.

Useful integrations: - HF Optimum Habana which also includes - DeepSpeed integration.

Apples-to-apples Comparison

It's very difficult to compare specs of different offerings since marketing tricks get deployed pretty much by all competitors so that one can't compare 2 sets of specs and know the actual difference.

- MLPerf via MLCommons publishes various hardware benchmarks that measure training, inference, storage and other tasks' performance. For example, here is the most recent as of this writing training v3.0 and inference v3.1 results.

  Except I have no idea how to make use of it - it's close to impossible to make sense of or control the view. This is a great intention lost in over-engineering and not thinking about how the user will benefit from it, IMHO. For example, I don't care about CV data, I only want to quickly see the LLM rows, but I can't do it. And then the comparisons are still not

apples to apples so how can you possibly make sense of which hardware is better I don't know.

Power and Cooling

It is most likely that you're renting your accelerator nodes and someone else is responsible for ensuring they function properly, but if you own the accelerators you do need to know how to supply a sufficient power and adequate cooling.

Power

Some high end consumer GPU cards have 2 and sometimes 3 PCI-E 8-Pin power sockets. Make sure you have as many independent 12V PCI-E 8-Pin cables plugged into the card as there are sockets. Do not use the 2 splits at one end of the same cable (also known as pigtail cable). That is if you have 2 sockets on the GPU, you want 2 PCI-E 8-Pin cables going from your PSU to the card and not one that has 2 PCI-E 8-Pin connectors at the end! You won't get the full performance out of your card otherwise.

Each PCI-E 8-Pin power cable needs to be plugged into a 12V rail on the PSU side and can supply up to 150W of power.

Some other cards may use a PCI-E 12-Pin connectors, and these can deliver up to 500-600W of power.

Low end cards may use 6-Pin connectors, which supply up to 75W of power.

Additionally you want the high-end PSU that has stable voltage. Some lower quality ones may not give the card the stable voltage it needs to function at its peak.

And of course the PSU needs to have enough unused Watts to power the card.

Cooling

When a GPU gets overheated it will start throttling down and will not deliver full performance and it can even shutdown if it gets too hot.

It's hard to tell the exact best temperature to strive for when a GPU is heavily loaded, but probably anything under +80C is good, but lower is better - perhaps 70-75C is an excellent range to be in. The throttling down is likely to start at around 84-90C. But other than throttling performance a prolonged very high temperature is likely to reduce the lifespan of a GPU.