Silicon Mechanics

Infrastructure Requirements for AI Inference vs Training

Investing in deep learning (DL) is a major decision. You need to be sure you understand the needs for each phase of the process, especially if you're considering AI at the edge. Below are practical tips to help you make a more informed decision about the composition of your AI cluster.

AI Process

Deep learning of an artificial neural network requires teams to curate huge quantities of data into a designated structure then feed that massive training dataset (the bigger, the better for training purposes) into a DL framework.

After the DL framework is trained (it has learned what inputs lead to what logical conclusion), it can leverage this new capability when exposed to novel data and make inferences about the new data that allow action.

For example, after seeing 50,000 images of cats with solid color coats, upon seeing an image of a multicolored cat, it should be able to infer that this image is also of a cat and not something else, like a car or a bicycle. The app or service using the inference model then uses the data in some way.

However, the infrastructure needed to achieve training versus inference is different in some critical ways.

Terminology

Neural Network:

Artificial neural networks are computing systems inspired by the organic neural networks found in human and other animal brains, where nodes (artificial neurons) are connected (artificial synapses) to work together.

Training:

Learning a new capability from existing data

Inference:

Applying this capability to new data (usually via an application or service)

Key Elements to Look for in Training Infrastructure

- Get as much raw compute power and as many nodes as you can afford. Think multi-core processors and GPUs. Why? The most critical issues our clients are facing today is getting accurately trained AI models. And, how very, very long it takes to get there. But, the more nodes and the more mathematical accuracy you can build into your cluster, the faster and more accurate your training will be.
- Training often requires incremental addition of new data sets that remain clean and well-structured. That means these resources cannot be shared with others in the datacenter. Focus on optimization for this workload and you'll have better performance and more accurate training than if you try to make a general compute cluster with the assumption that it can take on other jobs in its free time.
- Huge training datasets require massive networking and storage capabilities to hold and transfer the data, especially if your data is image-based or heterogeneous. Plan ahead for adequate networking and storage capacity, not just for strong computing.

- The greatest challenge in designing hardware for neural network training is scaling. Doubling the amount of training data doesn't mean doubling the number of resources used to process it. It means expanding exponentially.

Key Elements to Look for in Inference Infrastructure

- Inference clusters should be optimized for performance. Think simpler hardware with less power than the training cluster but with the lowest latency possible.
- Throughput is critical to inference. The process requires high I/O bandwidth and enough memory to hold both the required training model(s) and the input data without having to make calls back to the storage components of the cluster.
- Datacenter resource requirements for inference are typically not as great for a single instance compared to training needs. This is because the amount of data or number of users an inference platform can support is limited to the performance of the platform and the application requirements. Think of speech recognition software, which can only operate when there is one, clear input stream. More than one input stream renders the application inoperable. It's the same with inference input streams.

Special Considerations for Inference on the Edge

- Edge-based computers are significantly less powerful than the massive compute power that's located at data centers and the cloud. But that's ok because inference requires much less processing power than training clusters.
- If you have hundreds or thousands of instances of the neural network model to support, though, remember that each of these multiple incoming data sources needs sufficient resources to process the data.
- Normally, you want your storage and memory as close to the processor as possible, to reduce latency. When you have edge devices, though, the memory is sometimes nowhere near the processing and storage components of the system. This means you either need a device that supports GPU or FPGA compute and storage at the edge, and/or access to a high-performance, low-latency network.
- You could also use a hybrid model, where the edge device gathers data but sends it to the cloud, where the inference model is applied to the new data. So long as the inherent latency of moving data to the cloud is acceptable (it is not in some real time applications, such as self-driving cars), this could work for you.