Field Programmable Gate Arrays

Field programmable gate arrays (FPGAs) are a type of "field programmable" ASIC. Standard FPGAs can often be reconfigured via hardware description languages such as Verilog to implement new ASIC designs dynamically. While FPGAs are generally less efficient than custom ASICs, they can offer significant speed improvements over CPU implementations. Microsoft in particular has used FPGAs to perform deep learning inference and claims to have achieved significant speedups with their deployment. However, the approach has not yet caught on widely outside Microsoft.

Neuromorphic Chips

The "neurons" in deep networks mathematically model the 1940s understanding of neuronal biology. Needless to say, biological understanding of neuronal behavior has progressed dramatically since then. For one, it's now known that the nonlinear activations used in deep networks aren't accurate models of neuronal nonlinearity. The "spike trains" is a better model (see Figure 9-4), where neurons activate in short-lived bursts (spikes) but fall to background most of the time.

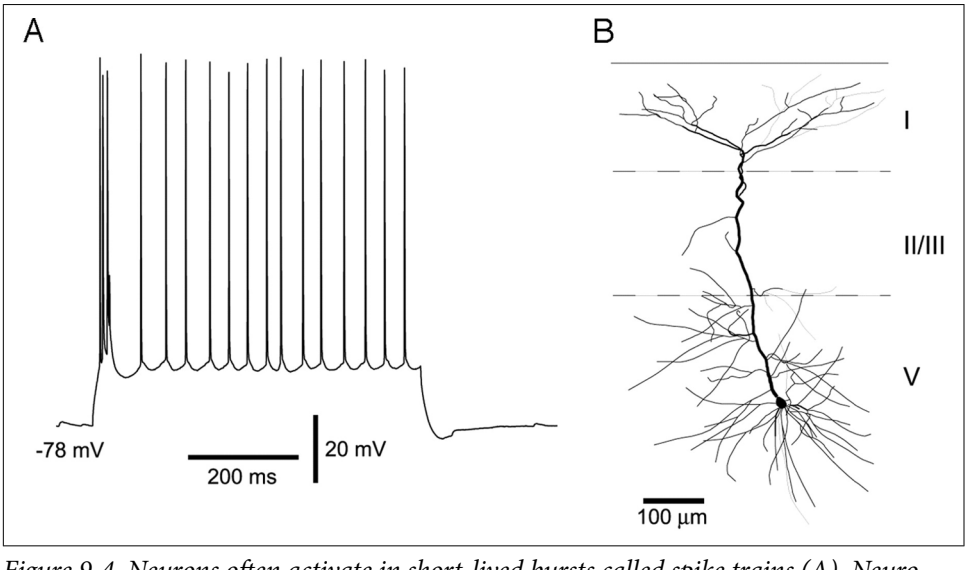

Figure 9-4. Neurons often activate in short-lived bursts called spike trains (A). Neuromorphic chips attempt to model spiking behavior in computing hardware. Biological neurons are complex entities (B), so these models are still only approximate.

Hardware engineers have spent significant effort exploring whether it's possible to create chip designs based on spike trains rather than on existing circuit technologies (CPUs, GPUs, ASICs). These designers argue that today's chip designs suffer from fundamental power limitations; the brain consumes many orders of magnitude less

power than computer chips and smart designs should aim to learn from the brain's architecture.

A number of projects have built large spike train chips attempting to expand upon this core thesis. IBM's TrueNorth project has succeeded in building spike train processors with millions of "neurons" and demonstrated that this hardware can perform basic image recognition with significantly lower power requirements than existing chip designs. However, despite these successes, it is not clear how to translate modern deep architectures onto spike train chips. Without the ability to "compile" TensorFlow models onto spike train hardware, it's unlikely that such projects will see widespread adoption in the near future.

Distributed Deep Network Training

In the previous section, we surveyed a variety of hardware options for training deep networks. However, most organizations will likely only have access to CPUs and perhaps GPUs. Luckily, it's possible to perform *distributed training* of deep networks, where multiple CPUs or GPUs are used to train models faster and more effectively. Figure 9-5 illustrates the two major paradigms for training deep networks with multiple CPUs/GPUs, namely data parallel and model parallel training. You will learn about these methods in more detail in the next two sections.

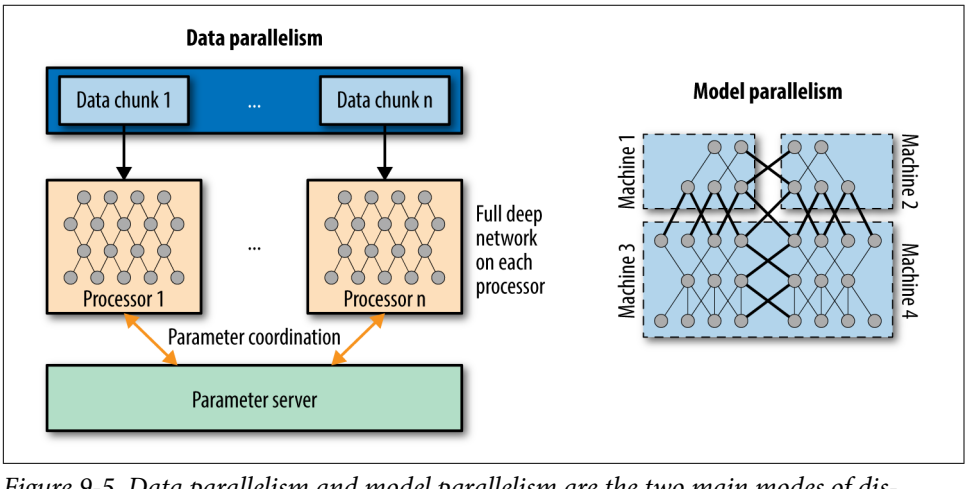

Figure 9-5. Data parallelism and model parallelism are the two main modes of distributed training of deep architectures. Data parallel training splits large datasets across multiple computing nodes, while model parallel training splits large models across multiple nodes. The next two sections will cover these two methods in greater depth.