Notes

November 6, 2019

0.1 Feed Forward Neural networks

The activation function serves the purpose of forcing the successive layers of the neural network to be linearly independent (the successive weight matrices do not combine into one matrix). From a computational perspective, these operations are heavy with multiplications.

0.2 Complexity

- The complexity of a square $[n \times n]$ matrix multiplied by a $[n \times 1]$ matrix is given by $O(n^2)$.

Where a multiplication operation is assumed to have a complexity of $O(1)$.
Each hidden layer has at least an matrix operation of this kind.

- The activation function's complexity is at least proportional to $O(\log(n))$.
- Hence, for every layer of the network l (except the input layer), there is at least a matrix multiplication and at least an activation.
- If a given layer has 10 neurons, then the matrix operation has complexity of at least $O(100)$, while a layer of size 100 has complexity of at least $O(10\,000)$.

Given that a typical CPU can run as many floating point operations in parallel as there are cores, a multi-cored system can reduce the effective run time as a factor of the number of cores. For instance, the 8^{th} generation Intel i7 Processor has six cores. Six cores means a maximum of six multiplication operations for any given cycle which implies a reduced run-time by a factor of 6 at most. Use of all the cores would also leave the processor unavailable for other tasks. From this it can be seen that large ANNs can very quickly require large amounts of computational power; since any given forward pass of the NN requires each computation to be made again.

0.3 FPGAs

0.3.1 Configurable Logic Block

FPGAs allow for hardware level implementation of computational structures. FPGAs are generally made up of configurable logic blocks (CLBs); whose connections are configurable. The LE elements frequently consist of one or more look-up tables (LUTs), storage elements, carry logic, and multiplexers. These components allow the CLBs to simulate various logical operations from simple Boolean logic, to addition, to multiplication.

0.3.2 Congifurable Logic Block Matrix

The logic elements are also networked together by means of configurable buses in such away that logic elements across the length of the FPGA can be connected to carry out logical operations. These configurable elements allow for complex computational architectures to be implemented on the FPGA; from arithmetic logic units (ALUs) to digital signal processing (DSP) functions like digital filters and fast Fourier transforms (FFTs). Implementation of these architectures can serve any number of purposes, from rapid prototyping of computational architectures, to offloading necessary but repetitive computations from a microprocessor or CPU.

The modularity of FPGAs CLBs, coupled with the configurable data bus system allows for repeated parallel structures. This lends itself well to implementing processors that are required to execute many similar operations in parallel rapidly, such as FFTs or digital filters. To that end FPGA manufacturers have been loading FPGAs with less configurable more dedicated hardware such as pre-optimised multiply-accumulators (MACs) known as DSP slices. DSP slices are in high-demand in these multiplication heavy computations.

Given how well FPGAs seem to lend themselves to parallelism and modularity, it seems reasonable to assume that NNs could take advantage of this without the need for dedicated resources that could otherwise be used for other DSP applications.

0.4 The binary spiking neural network

0.4.1 Synapse

Synapses are implemented as cyclic shift registers, where the output of the shift register feeds the input. The output of the shift register acts as an input to a binary two-input AND operator. The second input is the output of the presynaptic neuron, and the output of the AND operator is an input to a postsynaptic neuron. The function of the synapse is to regulate the firing rate of the presynaptic neuron, and the accumulation rate -and hence the firing rate- of the post-synaptic neuron.

The guiding principle and intuition behind the synaptic implementation is that the synapse regulates the firing rate of the presynaptic neuron by modulating the probability of each spike reaching the post synaptic neuron. The result is that if the presynaptic neuron fires with frequency f_{pre} and the synapse is seeded with some probability p_{synapse} then the apparent firing rate at the postsynaptic neuron

0.4.2 Neurons

Input Neuron The input neuron is fundamentally different from the remaining neurons in the network. The premise of the binary spiking neural network is that it reduces the complexity of the basic operations of a standard feed-forward neural network. The matrix multiplications of the standard FFNN are reduced to simultaneous binary AND operations across the network that execute in a single clock cycle. However, this construction requires a conversion of real world input parameters (typically some decimal value) into binary. The chosen method of conversion requires normalising the input value to some p where