HARDWARE ACCELERATORS FOR NEURAL NETWORKS

M.Tech Seminar Report

Submitted in partial fulfillment of the requirements
for the degree of

Master of Technology

by

Yashwanth Kumar Resham
(Roll No. 21307R018)

Under the guidance of

Prof. Virendra Singh

Department of Electrical Engineering
Indian Institute of Technology Bombay
October 2022

Acknowledgement

I express my gratitude to my guide Prof. Virendra Singh for providing me the opportunity to work on this topic.

Yashwanth Kumar Resham
Electrical Engineering
IIT Bombay

Abstract

Deep Neural Networks(DNN) are currently widely used for many Artificial Intelligence applications including supervision,speech processing and so on. DNNs provide high accuracy at the cost of computational complexity. This computational complexity makes them energy inefficient and throughput. To efficiently process the DNNs accelerators for DNNs can be implemented with various architectures and hardware platforms.

Contents

List of Figures

Chapter 1

Introduction

Human brain is the most complex machine in the universe. The main computational element of human brain is the biological neuron.It is the inspiration for neural networks.In the average human brain there are 86 billion neurons, these are connected to each other through dendrites and element leaving them is called axon. This is illustrated in Figure 3.1

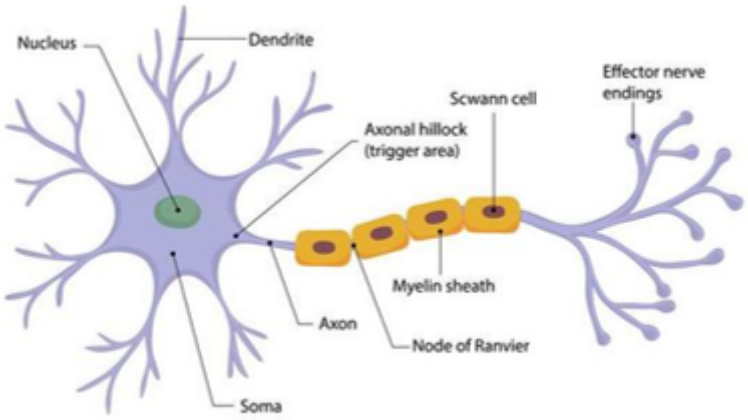

Figure 1.1: Biological Neuron

Neurons can be modelled mathematically as shown in Figure 3.2.The function of synapse is to scale the input x_i , scaling factor is referred to as weight. These weights are changed according to the output required and input given. Figure 1.3 shows a diagrammatic picture of a computational neural network. The neurons in the input layer receive some values and propagate them to the neurons in the middle layer of the network, which is also frequently called a "hidden layer." The weighted sums from one or more hidden layers are ultimately propagated to the output layer, which presents the final outputs of the network to the user. To align brain-inspired terminology with neural networks, the outputs of the neurons are often referred to as activations, and the synapses are often referred to as weights. Within the domain of neural networks there is an area called deep learning in which , neural networks have more than one hidden layer referred to as Deep Neural Networks(DNN). Deep neural networks (DNNs) are currently the foundation for many modern artificial intelligence (AI) applications. Since the breakthrough application of DNNs to speech recognition and image recognition , the number of applications that use DNNs has exploded. These DNNs are employed in a myriad of applications from selfdriving cars , to detecting cancer to playing complex games. In many of these domains, DNNs are now able to exceed human accuracy. The superior performance of DNNs comes

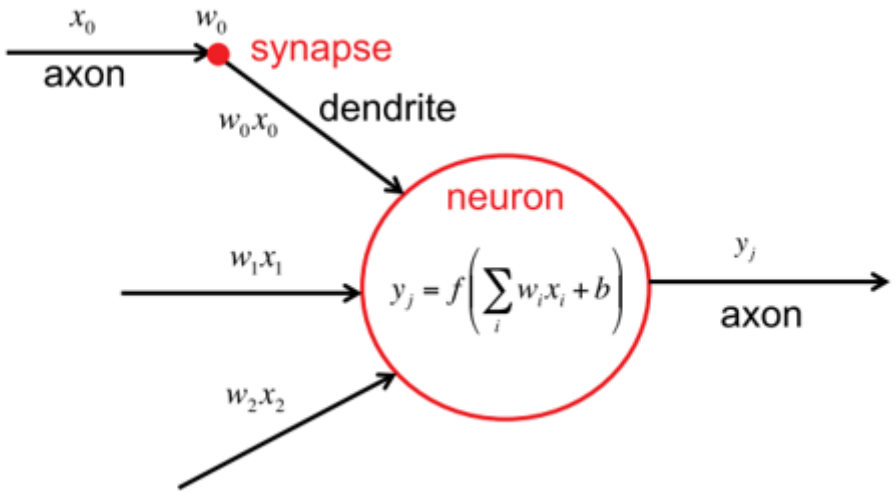

Figure 1.2: Mathematical Modelling of a neuron

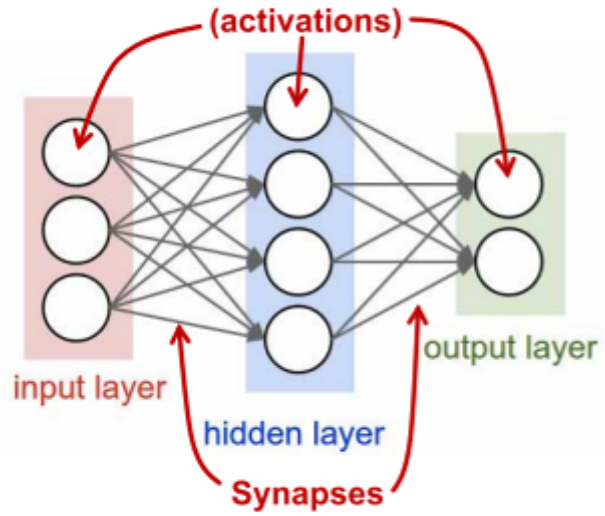

Figure 1.3: Simple Neural Network

from its ability to extract high-level features from raw sensory data after using statistical learning over a large amount of data to obtain an effective representation of an input space. This is different from earlier approaches that use hand-crafted features or rules designed by experts.

Chapter 2

Literature Survey

Deep Neural Networks has been a major area of research now which is aimed at decreasing the intervention of human in all tasks. Neural networks has been first proposed by [1].Later several architectures were developed with increasing complexity. [2] gives an overview of the work carried out in the area of Neural networks. It also provides the bottlenecks in processing of neural networks.

Chapter 3

Review

3.1 Efficient Processing of Deep Neural Networks: A Tutorial and Survey

3.1.1 Summary

DNNs have wide variety of sizes and shapes depending on the application.In all the cases input to the DNN is information that to be analyzed by the system , for instance pixels of an image in a vision processing system. Neural networks are of two forms (a) feed forward (b) recurrent neural networks. Feed forward networks all of the computation is performed on the output of a previous layer in a sequence manner. In contrast, Recurrent Neural Networks(RNNs) some intermediate operations generate values that are stored in the network and used in conjunction with later input. DNNs can be composed of Fully

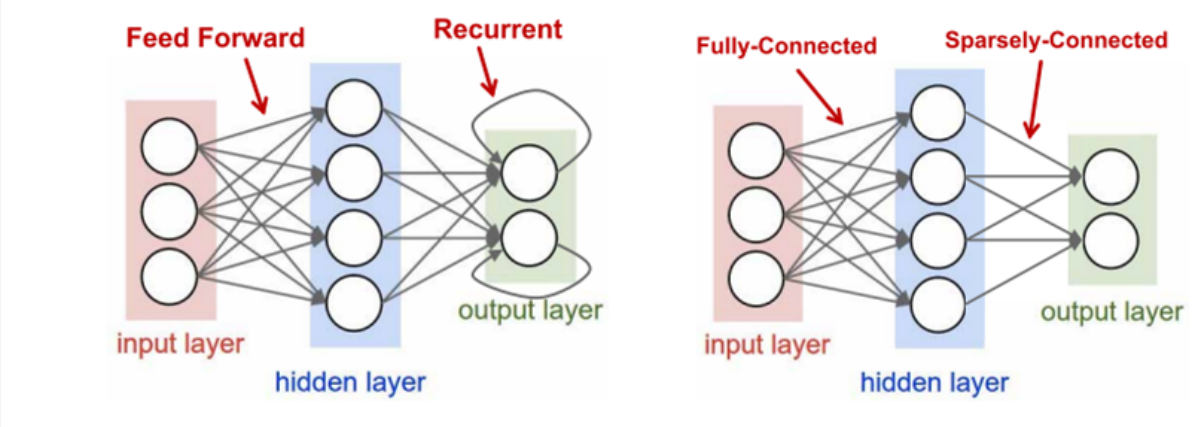

Figure 3.1: Different types of Neural Networks

connected layers in which all output activations are composed of all input activations. Whereas in Sparsely connected layers some weights can ignored by compromising accuracy. Each DNN contains convolutional(CONV) layer. Each CONV layer performs a set of operations as described in Figure 3.2 . Convolution operation is performed on input feature map and the filter weights, in the next stage a non-linear activation function is used to introduce non-linearity into DNN. A pooling operation is performed to increase the robustness of the network to small distortions by reducing dimensions. Due to popularity of DNNs , many hardware platforms have been introduced with special features for DNN processing. Efficient design of hardware has been done with two major architectures (a) Temporal (b) Spatial. In temporal architecture the computational transforms on data

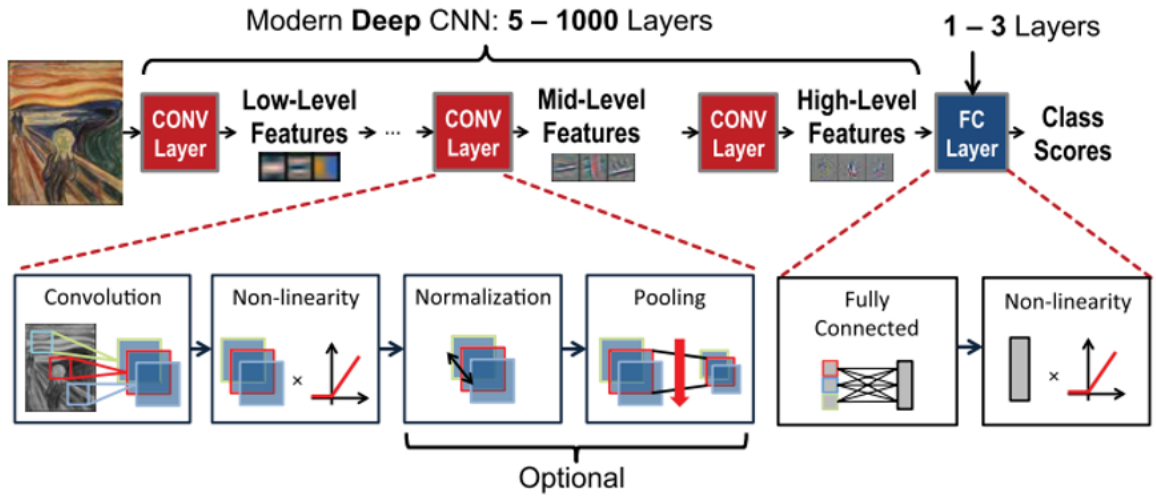

Figure 3.2: Convolutional Neural Network

like Fast Fourier Transform(FFT), improving MAC operation are key considerations. In the later type data reuse after computations by storing in local buffer is considered. It is done with the help of introducing a local buffer in each Processing Element(PE). Also each PE can communicate with each other and use output of the other PE. This type of architecture is energy efficient as it decreases the read/write operations to the DRAM.

Data reuse can be of three types viz. weight stationary , input stationary , row stationary. In weight stationary , filters weights are read from DRAM to Register File(RF) in PE and later processing is carried out till computations requiring weights in RF are completed. Similarly with other data reuse.

Also co-design of DNN hardware can be used by reducing the bit precision of operands and operations in the MAC unit.

3.1.2 Strengths

Application specific DNN hardware can decrease the computational complexity and is energy efficient. DNNs with data reuse decreases the DRAM accesses and the storage required for computations.

3.1.3 Weakness

Approaches of co-design of DNN hardware and their energy efficiency should be exploited more. Standardized benchmarks for evaluation of hardware accelerators can be introduced.

3.1.4 Opportunities

While creating a accelerator , better decisions can be made by exploring the design space. How effectively an accelerator works can be known with the help of simulations. Various simulation frameworks that give an insight of accelerator with benchmarks can be studied.

3.2 Conclusion

The explosive growth of DNNs in the past few years has made a path for special hardware. Because DNNs can achieve high accuracy at the cost of high high computation. This needs to be effectively reduced. The DNN domain also affords an excellent opportunity for joint hardware/software codesign. For example, various efforts have noted that efficiency can be improved by increasing sparsity (increasing the number of zero values) or optimizing the representation of data by reducing the precision of values or using more complex mappings of the stored value to the actual value used for computation. However, to avoid losing accuracy it is often useful to modify the network or fine tune the network's weights to accommodate these changes.

References

[1] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological Cybernetics*, vol. 36, pp. 193–202, 1980.

[2] V. Sze, Y.-H. Chen, T.-J. Yang, and J. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proceedings of the IEEE*, 2017.