Evaluating memory structures when performing inference on pruned neural networks

Deliverable 1: Final year Dissertation

Bsc Computer Science: Artificial Intelligence

Sam Fay-Hunt — sf52@hw.ac.uk

Supervisor: Rob Stewart — R.Stewart@hw.ac.uk

October 16, 2020

DECLARATION

I, Sam Fay-Hunt confirm that this work submitted for assessment is my own and is expressed in my own words. Any uses made within it of the works of other authors in any form (e.g., ideas, equations, figures, text, tables, programs) are properly acknowledged at any point of their use. A list of the references employed is included.

Signed: .....

Date: .....

Abstract: a short description of the project and the main work to be carried out – probably between one and two hundred words

Contents

1 Introduction

Summarising the context of the dissertation project, stating the aim and objectives of the project, identifying the problems to be solved to achieve the objectives, and sketching the organisation of the dissertation.

Edge devices have never been cheaper *citation*, stuff about how IoT devices are ubiquitous

Mention how there is an increasing trend to perform computing at the Edge - real time applications + privacy

These devices are often equipped with some form of AI application: Photo enhancment ect.

Online vs offline learning

Edge-side inference

These models can have a huge number of parameters so inference can sometimes be impractical. [1] - "see Table 1"

Issues with limited resource computation [2]

outline the document: We start with ..., then we cover x, y, and z ...

This dissertation is an investigation into the effect of pruning on inference in terms of latency and accuracy using hardware without specific optimisations for processing the resulting sparse matrices from pruning. (Reasoning for statement...).

2 Background

Discussing related work found in the technical literature and its relevance to your project.

This Section will be split into 4 subsections:

Hardware Memory architectures Section 2.1 - brief stuff about this section

Edge Computing Section 2.2 - stuff about edge comp

Deep Learning Section 2.3 - An overview of the basic components of a neural network and the CNN & RNN models.

Compression Types Section 2.4 - ...

2.1 Hardware memory architectures

- Discuss VPU/TPU/APU/GPU/FPGA/ASIC memory arcitecture and how it handles matrix sparsity

- Show ineffectivity of pruning on hardware without optimisations for sparse matrices

The explosion of Deep Neural Network applications in recent years has prompted the production of a wave of specialised hardware architectures to improve the efficiency and compute of these kinds of workloads. The mainstay of this form of processing has been until recently been dominated by GPUs.

2.2 Computing at the edge

Some background on edge computing - maybe a detailed definition

- Challenges of resource bound deep learning

- Online vs offline learning

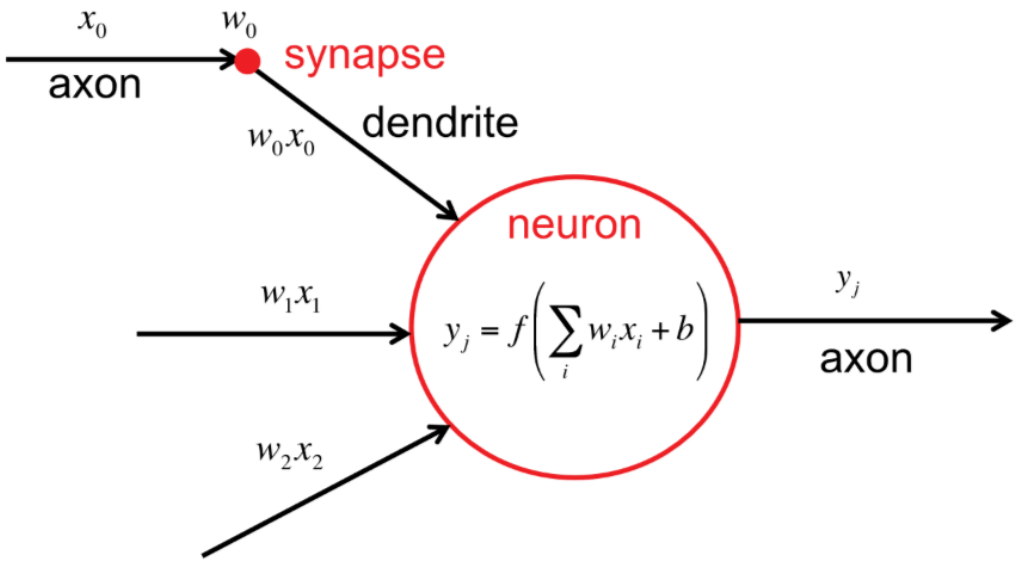


Figure 1: Neuron with corresponding biologically inspired labels.
(Adopted figure from [2])

2.3 Deep Neural Networks

2.3.1 Neural Networks & Deep Learning

- *Summary of NN*

- *Structure of NN*

- *Training & Inference stages*

- *weight update methodologies*

- *Feed Forwards*

- *Feedback Nerual Network*

- *Self-organizing Neural Network*

- *Weight parameters updated using back-propagation*

Deep learning is a category of machine learning techniques where a hierarchy of layers that

perform some manner of information processsing that computes high level abstractions of the data by identifying low level abstractions in the early layers [3].

Neural networks are a subfield within the category of Artifical Intelligence computing, its purpose simply put is to transform an input vector X into an output vector $\hat{Y}$. The output vector $\hat{Y}$ is some form of classification such as a binary classification or a probability distribution over multiple classes [4]. Between the input layer (X) and the output layer ($\hat{Y}$) there exists some number of interior layers that are referred to as hidden layers, the hidden and output layers are composed of neurons that pass signals derived from weights through the network, this model of computing was inspired by connectionism and our understanding of the human brain, see Fig. 1 for labels of the analagous biological components. Weights in a neural network effectively correspond to the synapses in the brain and the output of the neruon is modelled as the axon. All neruons in a Neural network have weights corresponding to their inputs, these weights are are intended to mirror the value scaling effect of a synapse by performing a weighted sum operation [2].

Neural networks and deep neural networks are often reffered to interchangably, they are primarily distinguished by the number of layers, there is no hard rule indicating when a neural network is considered deep but generally a network with more than 3 hidden layers is considered a deep neural network, the rest of this paper will refer to DNNs for consistency. Each neuron in a DNN applies an non-linear activation function to the result of its weighted sum of inputs and weights, without which a DNN would just be a linear algebra operation [2].

DNNs can be categorised as feedforwards, feedback, and self-organising networks depending on their processing method [1].

There are many popular deep neural network architectures, the full scope of these are

beyond the scope of this document. The next sections will focus primarily on the CNN and RNN architectures.

2.3.2 Convolutional Neural Network

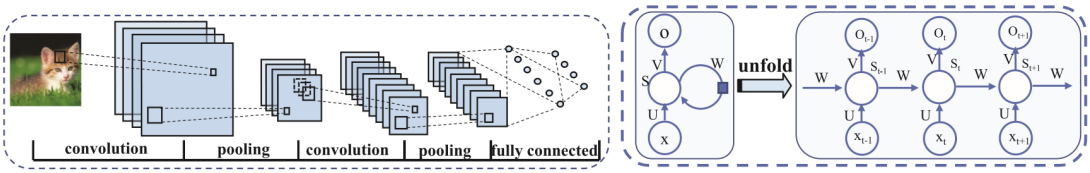
Convolutional Neural Network (CNN)

- A class of DNN

- CNN consist of: Convolutional Layers, Pooling layers & fully connected layers.

- Convolutional Layers contain sets of filters/kernels Much like traditional nerual networks the CNN architecture was inspired by human and animal brains, the concept of processing the input with local receptive fields is conceptually similar some functionality of the cat's visual cortex [5]–[7]. These local receptive fields (or kernels) enable neurons to extract low level features such as edges, corners, and end-points with respect to their orientation. CNNs are robust to input shift or distortion by using receptive fields to identify these low level features across the entire input space, performing local averaging and downsampling in the layers following convolution layers means the absolute location of the features is less important than the position of the features relative to the position of other identified features [6]. Each layer produces higher degrees of abstraction from the input layer, in doing so these abstractions retain important information about the input, these abstractions are referred to as feature maps. The layers performing downsampling are known as pooling layers, they reduce the resolution or dimensions of the feature map which reduces overfitting and speeds up training by reducing the number of parameters in the network [7].

CNNs have been found to be effective in many different AI domains, popular applications include: computer vision, NLP, and speech processing.



Convolutional Neural Network Recurrent Neural Network

Figure 2: A typical example of a CNN (left) and RNN (right)
(Adopted figure from [1])

2.3.3 Recurrent Neural Network

Recurrent Neural Network (RNN)

RNNs are deep learning models that use loops in their layer connections to make predictions with sequential inputs and maintain state over those inputs, this architecture is designed specifically for time series predictions [8].

2.4 Compression types

pruning

distillation

Quantization

Network design strategies

low-rank factorization

3 Research Methodology/ Requirements Analysis

3.1 Research Methodology

This is required for research projects and should be linked back to the project aim and objectives. It should describe the research methods that will be employed in the project and the research questions that will be investigated.

1. build dataset of benchmarks from my systematic benchmark framework from models
2. perform pruning on models
3. run benchmark again with pruning
4. make adjustments to underlaying mechanism of parameter storage
5. verify adjustments do not break the model
5. run benchmarks again
6. draw conclusions

Find baselines/benchmarks

How to perform pruneing

Look at underlaying storage mechanism of parameters in Network

- provide some plots visualising the sparsity of the weights for the pruned matrix

perform some engineering of refactoring/altering these mechanisms

rerun systematic benchmarking framework

3.2 Requirements Analysis

This is required for technical projects and should be linked back to the project aim and objectives. It should provide a detailed use case scenario and suitable use

4 Design

Initial software design/sketch of research Methodology

5 Evaluation Strategy

Details of the evaluation and analysis to be conducted

6 Project Management

6.1 Plan

6.2 Risk Analysis

mention benchmarking NLP/NLG/Audio - text/text - audio models as a risk to the project

6.3 Professional, Legal & Ethical issues

A Back matter

A.1 References

References

[1] Y. Chen, B. Zheng, Z. Zhang, Q. Wang, C. Shen, and Q. Zhang, "Deep Learning on Mobile and Embedded Devices: State-of-the-art, Challenges, and Future Directions," *ACM Computing Surveys*, vol. 53, no. 4, pp. 1–37, Sep. 26, 2020, ISSN: 0360-0300, 1557-7341. DOI: 10.1145/3398209. [Online]. Available: https://dl.acm.org/doi/10.1145/3398209 (visited on 10/01/2020).

[2] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient Processing of Deep Neural Networks: A Tutorial and Survey," *Proceedings of the IEEE*, vol. 105, no. 12, pp. 2295–2329, Dec. 2017, ISSN: 0018-9219, 1558-2256. DOI: 10.1109/JPROC.2017.2761740. [Online]. Available: http://ieeexplore.ieee.org/document/8114708/ (visited on 10/01/2020).

[3] L. Deng, "A tutorial survey of architectures, algorithms, and applications for deep learning," *APSIPA Transactions on Signal and Information Processing*, vol. 3, e2, 2014, ISSN: 2048-7703. DOI: 10.1017/atsip.2013.9. [Online]. Available: https://www.cambridge.org/core/product/identifier/S2048770313000097/type/journal_article (visited on 10/16/2020).

[4] J. Thierry-Mieg, "How the fundamental concepts of mathematics and physics explain deep learning.," p. 16,

[5] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *The Journal of Physiology*, vol. 160, no. 1, pp. 106–154, Jan. 1, 1962, ISSN: 00223751. DOI: 10.1113/jphysiol.1962.sp006837.

[Online]. Available: http://doi.wiley.com/10.1113/jphysiol.1962.sp006837 (visited on 10/15/2020).

[6] Y. LeCun, Y. Bengio, and T. B. Laboratories, "Convolutional Networks for Images, Speech, and Time-Series," p. 15,

[7] S. Pouyanfar, S. Sadiq, Y. Yan, H. Tian, Y. Tao, M. P. Reyes, M.-L. Shyu, S.-C. Chen, and S. S. Iyengar, "A Survey on Deep Learning: Algorithms, Techniques, and Applications," *ACM Computing Surveys*, vol. 51, no. 5, pp. 1–36, Jan. 23, 2019, ISSN: 0360-0300, 1557-7341. DOI: 10.1145/3234150. [Online]. Available: https://dl.acm.org/doi/10.1145/3234150 (visited on 10/15/2020).

[8] J. Chen and X. Ran, "Deep Learning With Edge Computing: A Review," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1655–1674, Aug. 2019, ISSN: 0018-9219, 1558-2256. DOI: 10.1109/JPROC.2019.2921977. [Online]. Available: https://ieeexplore.ieee.org/document/8763885/ (visited on 10/01/2020).

A.2 Appendices

to include additional material, consult with your supervisor.

Acronyms

AI Artifical Intelligence. 4

CNN Convolutional Neural Network. 4, 2, 5, 6

DNN Deep Neural Network. 4, 3, 4

NLP Natural Language Processing. 5

RNN Recurrent Neural Network. 4, 2, 5, 6