Fun With Neural Entanglement

Enrique Fernadez

October 2018

1 Abstract

Senior Design Group 71 At Oregon state university will propose new inference engine practices to develop software functionality that will search network-configuration and enable faster table compressions and lookups. Our client will have us focus on the incidental benefits that is using inference engines. We will be demonstrating key deliverables such as demonstrating a list of projects that use "neural entanglement" to invoke multiple inference engines to work as one unit, and finally using analytical software to describe the effectiveness of our neural network strategy. These deliverables will help assist our client in multiple strategies to provide incidental benefits of using inference engines.

2 Introduction

Neural networks, or more properly termed 'artificial' neural network (ANN) can be best described by a series of interconnected processing systems[1]. ANNs can be either processing algorithms or actual hardware that are modeled by the neural structure of the brain but on a much smaller scale. Neural networks are typically organized in layers that are made up by several interconnected nodes which contain an 'activation function' or better known as a trigger. This network on interconnected nodes make up the neural network and receive input via the input layer. The input layer then communicates with one or more layers where the actual processing happens. The layers that do the actual process are named "hidden layers". The hidden layers then link to an output layer which gives the result. Neural networks differ from conventional computing in a variety of ways. Conventional computer use a central processor that can reference its memory location where programs can be stored. Pre-defined instructions are the foundation of instructions and readings instruction to get them ready for execution. in a conventional computer, the instructions run on this computer are deterministic, sequential, and logical. ANN's are run by a series of triggers and have distributed processing power. The processing power is not complex either. ANNs are also not deterministic. The internal layer responds to the input and produces the proper output. Hence, there is no separate memory from which data is stored from each process. The state of the neural network is the memory of the system. Neural networks are used for approximate solutions. Therefore, is not recommended for critical measurements. The application model in which you plan to implement neural networks must have a high tolerance for error. Some implementation practices is discovering regularities within a set of patters. A good example of regularities in a set of patterns is finding cures for diseases. Neural networks have accelerated cures for diseases and produced medicine.

3 Problem Statement

Lookup tables are a very time-consuming process as well as a high-resource compute process using a control flow process. Neural entanglement, however, is a rising approach to analyze network coefficients to gather the necessary information that is requested. Inference engines are a sub product of neural entanglement practices which are systems that apply logical rules to a specific set of data or knowledge base. Our problem entails the use the inference engines and the knowledge base to optimize characteristics of inference engines to optimize neural networks. Some of those characteristics worth investigating is would be when forward chaining an inference engine. Looking at triggers when data is stored in memory can be a grand opportunity to optimize results. Another situation could be backward chained engines. When these are in place, we must also look at how we might drift away from back-ward chaining so looking at we can steer out of looking at data directly and focus on structure to further optimize results. This fulfills one goal that kicks off the development of our project. Introducing how we might use neural networks

4 Solution

Our Deliverables is to list a as an introduction Inference engines have been practiced by John Huan ko. John designed an energy efficient neural network inference engine that us based on adaptive weight compression using a JPEG image encoding algorithm that achieves 63.4X multilayer perceptron and has reduced the memory requirement which has resulted to higher throughput and lower neural network inference. Projects like these can provide us a plethora of information to full fill the first criteria of our project.

5 Performance Metrics

Our performance metrics are described on the deliverables described on the project description. Our first task is to come up with a list of projects that have implemented neural networks and list some ideas with what we can do with neural entanglement processes. This deliverable implies a deep understanding on neural networks with inference engines. We can demonstrate this deliverable has been met by demonstrating a neural network implementation. Second, once we have demonstrated that we can create a neural network we can create multiple inference engines that can later be interconnected into a global inference engine will solve our second criteria. Finally, our third deliverable is demonstrating how our neural network implementation has shown that we have improved one incidental characteristic when implementing a compression technique.

6 Summary

Through the course of this school year, we plan on implementing a compression algorithm using conventional computing models and compare them to Neural networks to demonstrate incidental benefits of our project. We will present analytical data to provide research and ultimately reach our goal of being published.

7 citations

[1] J. H. Ko, D. Kim, T. Na and S. Mukhopadhyay, "Design and Analysis of a Neural Network Inference Engine based on Adaptive Weight Compression," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. doi: 10.1109/TCAD.2018.2801228 keywords: Image coding;Transform coding;Training;Engines;Entropy;Memory management;Discrete cosine transforms;neural network;weight compression;memoryefficient;JPEG;MLP;CNN., URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=arnumber=8279481isnumber=6917053