Deep Learning on SpiNNaker: Report

Jonas Fassbender
jonas@fassbender.dev

Abstract

Contents

1 Introduction

According to the SpiNNaker project's website:

> SpiNNaker is a novel massively-parallel computer architecture, inspired by the fundamental structure and function of the human brain, which itself is composed of billions of simple computing elements, communicating using unreliable spikes (SpiNNaker Team, 2020).

SpiNNaker is targeted at three main areas of research: (i) neuroscience: understanding the human brain, (ii) robotics: low power hardware and (iii) computer science: new approach to supercomputing and massive parallelism. The dissertation project, which this paper reports on, will concern itself with (iii) and one of the research areas of computer science, which has emerged as a driving force behind advancements in many fields and for many tasks like speech and image recognition, drug discovery and genomics: deep learning (LeCun et al., 2015).

While deep learning is a promising field and deep neural networks at the center of important advancements, like described above, they face a major problem: the sheer amount of computation needed for training them. Researchers at OpenAI have estimated, that the amount of computation needed for training state of the art deep neural networks increases exponentially, doubling every 3.4 months (Amodei et al., 2019). In order to cope with such unprecedented amounts of computation and energy needed, the search for specialized hardware is well underway. Current state of the art hardware for accelerating the training of deep neural networks are ASICs like TPUs and general purpose GPUs (Jouppi et al., 2017; Mittal and Vaishay, 2019).

The goal of the dissertation project that is introduced in this paper, will be to analyze if SpiNNaker could be an energy efficient, scalable and fast alternative to the above mentioned hardware. Since deep neural networks are derived from the human brain and nerve system (Goodfellow et al., 2016) and SpiNNaker was designed to model the human brain, it seems rather probable, that SpiNNaker will be a good target for accelerating the training of deep neural networks.

This paper concerns itself with the dissertation project: "Deep Learning on SpiNNaker," which will be conducted in the period from May 2019 to August 2019 as the final work of the author to achieve his Master of Science in High Performance Computing with Data Science from the University of Edinburgh. The report is mostly a summary of the preliminary work conducted in the months before the actual work on the disseration will be conducted.

The finindings of the preliminary work and the changes made to the original project scope are the focal point of this report. The original title of the disseration project was "A Tensorflow Backend to SpiNNaker," but the preliminary work conducted to this point show, that the scope will be redirected from implementing a backend for tensorflow—a library for running fast linear algebra operations on distributed, heterogenous systems, mainly designed for implementing computationally demanding machine learning algorithms like deep learning in a fast manner (Abadi et al., 2015)—to an approach focused on implementing deep learning directly on SpiNNaker. Because of SpiNNaker's specialized design, which works rather contrary to that of tensorflow and current hardware trends for building accelerators for deep learning, interfacing between SpiNNaker's runtime and tensorflow was deemed too difficult and not beneficial. Instead, this dissertation aims at implementing deep learning directly on SpiNNaker, providing an interface to the well known deep learning library Keras (Chollet et al., 2015). Mario Antonioletti and Alan Stokes (2019) shows the original disseration project's scope.

This paper starts with presenting a brief outline of the background of the dissertation, providing a discription of the technologies important in this report: SpiNNaker, deep learning and tensorflow in Section 2. Also, the updated goal for the dissertation is given. Afterwards, in Section 3, some papers and other literature crucial for the further doings for this project are presented. The paper continues by giving a work plan in Section 4, before presenting a risk analysis in Section 5. Afterwards the preliminary findings outlined above are disscussed in more depth in Section 6. At last, Section 7 gives the final project proposal and Section 8 will contain a review of a related dissertation project done in 2018: “Deep Learning Performance on Different Architectures” by Spyro Nita (Nita, 2018).

2 Background

3 Related Work

4 Work Plan

5 Risk Analysis

6 Preliminary Findings

7 Final Project Proposal

8 Deep Learning Performance on Different Architectures: Review

This section will concern itself with a review of the dissertation of Spyro Nita, which he did for earning his Master of Science in High Performance Computing with Data Science at the University of Edinburgh: “Deep Learning Performance on Different Architectures” (Nita, 2018). First, a summary of his dissertation will be given, before the review of his work is presented. The review will start by looking at how well the context of the dissertation is explained and how well the scope of the dissertation's problem is defined. The last part of the review will be the consideration of the dissertation's layout and formatting. After the review, the importance of the dissertation for “Deep Learning on SpiNNaker” will be discussed and evaluated.

Nita (2018) concerns itself with measuring the computational performance of two different approaches for training deep neural networks: (i) distributed training using GPUs and (ii) distributed training using CPUs. The dissertation is a work derived of the efforts of Team EPCC at the Student Cluster Competition at the International Supercomputing Conference 2018, held in Frankfurt, where Team EPCC competed against other teams by building a small supercomputing cluster and running various benchmarks on it, to see which team built the best supercomputing cluster. One of these benchmarks concerned itself with measuring the performance of training a deep neural network on the clusters, which is picked up in Nita (2018) and represents the benchmark for approach (i).

The benchmark is based on the famous ImageNet dataset, which consists of more than 14 million images, organized according to the WordNet hierarchy into over 21 thousand so called synsets (synonym sets) (Russakovsky et al., 2015; Miller, 1995). Contrary to the annual Large Scale Visual Recognition Challenge

(ILSVRC)—a benchmark also based on the ImageNet dataset; the de-facto standard for benchmarking image recognition models—the benchmark for the Student Cluster Competition only concerns itself with the throughput of images during training and not with the accuracy of the trained models. The througput is measured in images per second. Only if two teams should have the same throughput during training is the training accuracy taken into account as the secondary criterion for tie-breaking. The teams participating in the Student Cluster Competition had to train the VGG16 deep neural network—a famous model architecture introduced in the ILSVRC 2014 by researchers from Oxford University (Simonyan and Zisserman, 2014)—on 1000 synsets containing 1.2 million images. The main limitation for the clusters were their power budget of 3kW and—for the ImageNet benchmark—a maximum runtime of three hours (Nita, 2018).

Nita (2018) compares the throughput of the supercomputing cluster of Team EPCC for the performance of a distributed GPU system against Cirrus—a supercomputer hosted by the EPCC—for the performance of a distributed CPU system (EPCC, 2020).

Concerning the results of Nita (2018), two major problems were encountered: (i) cluster damage and failure of the Team EPCC cluster shortly before the competition and (ii) the fact that the CPU benchmark on Cirrus could not be distributed over multiple backend nodes. Fortunately benchmarks were done on the Team EPCC cluster, before its failure and Nita (2018) presents the results of using the first node of the cluster with six NVIDIA V100 GPUs. Concerning (ii), Nita (2018) presents the results for a single backend node and assumes that Cirrus would be able to linearly scale to multiple backend nodes, when comparing the two clusters. The single node of the Team EPCC cluster with its six NVIDIA V100 GPUs generates a throughput of 2,052 images per second, while a single backend node of Cirrus (two Intel Xeon E5-2695 processors, each with 18 physical cores) can generate a throughput of just 18 images per second. Assuming linear scaling over multiple backend nodes of Cirrus, 21 nodes would be needed for the throughput of a single NVIDIA V100 GPU. Nita (2018) concludes that GPUs offer a significant advantage compared to CPUs, when it comes to training deep neural networks.

The dissertation starts with an introduction, which includes a description of the dissertation's layout. The second chapter concerns itself with a description of convolutional neural networks for image classification, describing ImageNet and VGG16 as a special convolutional neural network. Chapter three describes everything concerning the Student Cluster Competition, including the benchmarks, rules and various features of the Team EPCC cluster, including the used hardware, operating system and other important software libraries. It also describes the difficulties and hardware damage to the cluster and—concerning the ImageNet benchmark—how the images were preprocessed and reformatted to the TFRecord format (Abadi et al., 2015). Chapter four concerns itself with the optimization of the VGG16 neural network, mainly through different parameters which are passed to tensorflow, which is used for implementing VGG16. It thoroughly describes how one can distribute training across multiple GPUs. The benchmark for the Student Cluster Competition is described and analyzed, as is the benchmark for Cirrus. The chapter ends by comparing both. Nita (2018) concludes by summarizing the results presented in chapter four and gives an outline for future work, which could shed more light on the performance of GPU and CPU clusters for training deep neural networks. The two main points of emphasis for future work presented are: (i) scaling training to hundreds of CPU nodes (see e.g. You et al., 2017, for distributing training onto multiple CPUs) and (ii) comparing power consumption of CPU nodes against GPU nodes.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL `https://www.tensorflow.org/`. Software available from tensorflow.org.

Dario Amodei, Danny Hernandez, Girish Sastry, Jack Clark, Greg Brockman, and Ilya Sutskever. AI and Compute. `https://openai.com/blog/ai-and-compute/`, 2019.

François Chollet et al. Keras. `https://keras.io`, 2015.

EPCC. Cirrus, 2020. URL `https://www.cirrus.ac.uk`.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

Norman P. Jouppi, Cliff Young, Nishant Patil, David A. Patterson, Gaurav Agrawal, Raminder Bajwa, Sarah Bates, Suresh Bhatia, Nan Boden, Al Borchers, Rick Boyle, Pierre-luc Cantin, Clifford Chao, Chris Clark, Jeremy Coriell, Mike Daley, Matt Dau, Jeffrey Dean, Ben Gelb, Tara Vazir Ghaemmaghami, Rajendra Gottipati, William Gulland, Robert Hagmann, Richard C. Ho, Doug Hogberg, John Hu, Robert Hundt, Dan Hurt, Julian Ibarz, Aaron Jaffey, Alek Jaworski, Alexander Kaplan, Harshit Khaitan, Andy Koch, Naveen Kumar, Steve Lacy, James Laudon, James Law, Diemthu Le, Chris Leary, Zhuyuan Liu, Kyle Lucke, Alan Lundin, Gordon MacKean, Adriana Maggiore, Maire Mahony, Kieran Miller, Rahul Nagarajan, Ravi Narayanaswami, Ray Ni, Kathy Nix, Thomas Norrie, Mark Omernick, Narayana Penukonda, Andy Phelps, Jonathan Ross, Amir Salek, Emad Samadiani, Chris Severn, Gregory Sizikov, Matthew Snelham, Jed Souter, Dan Steinberg, Andy Swing, Mercedes Tan, Gregory Thorson, Bo Tian, Horia Toma, Erick Tuttle, Vijay Vasudevan, Richard Walter, Walter Wang, Eric Wilcox, and Doe Hyun Yoon. In-datacenter performance analysis of a tensor processing unit. *CoRR*, abs/1704.04760, 2017. URL `http://arxiv.org/abs/1704.04760`.

Yann LeCun, Y. Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521:436–44, 05 2015. doi: 10.1038/nature14539.

Mario Antonioletti and Alan Stokes. A Tensorflow Backend to SpiNNaker, 2019. URL `https://www.wiki.ed.ac.uk/display/hpcdis/A+TensorFlow+BackEnd+to+SpiNNaker`.

George A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):3941, November 1995. ISSN 0001-0782. doi: 10.1145/219717.219748. URL `https://doi.org/10.1145/219717.219748`.

Sparsh Mittal and Shraiysh Vaishay. A survey of techniques for optimizing deep learning on gpus. *Journal of Systems Architecture*, 08 2019. doi: 10.1016/j.sysarc.2019.101635.

Spyro Nita. Deep learning performance on different architectures. 2018. URL `https://static.epcc.ed.ac.uk/dissertations/hpc-msc/2017-2018/Spyro_Nita-dissertation-spyro-nita.pdf`.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. 09 2014.

SpiNNaker Team. SpiNNaker Project, 2020. URL `http://apt.cs.manchester.ac.uk/projects/SpiNNaker/project/`.

Yang You, Zhao Zhang, Cho-Jui Hsieh, James Demmel, and Kurt Keutzer. Imagenet training in minutes, 2017.