

Call for applications – PhD candidate

Lyon Institute of Nanotechnology
Ecole Centrale de Lyon, 36 av. Guy de Collongue,
F-69134 Ecully, FRANCE
<http://inl.cnrs.fr>



Custom Precision Learning Hardware Architectures

Deep Neural Networks (DNNs) [1] are currently one of the most intensively and widely used predictive models in the field of machine learning. DNNs have proven to give very good results for many complex tasks and applications, such as object recognition in images/videos, natural language processing, satellite image recognition, robotics, aerospace, smart healthcare, and autonomous driving.

However, the cost of inference and in particular of the **training** became so complex that it now requires specialized hardware such as GPUs or TPUs. Other than the complexity of the hardware, the energy cost and the related environmental impact of training is now comparable to other human activities. For example, in [2], it is shown that the CO₂ emissions due to training is equivalent to the life time CO₂ emission of a single car. Furthermore, there is an incentive to migrate AI from the cloud into the edge devices, i.e., Internet-of-Things (IoT) devices, in order to address data confidentiality issues and bandwidth limitations, given the ever-increasing internet-connected IoTs, and also to alleviate the communication latency, especially for real-time safety-critical decisions, e.g., in autonomous driving.

Accelerating the training, reduce its cost and make it available at the edge has thus become a hot research topic, but early work in this direction did not necessarily translate to a wide adoption and availability of low/mixed-precision training hardware. The most widespread approach to increase performance and efficiency of DNN training at the arithmetic level is through the use of mixed precision. For example, NVIDIA has offered the possibility to do low precision training¹ since the Pascal architecture in 2016 and mixed precision training (combining float16 and float32 arithmetic). The Google Tensor Processing Units (TPUs) offer similar support for mixed precision training with the introduction of bfloat16, a 16-bit floating-point format that, when compared to float16, trades in mantissa bits for exponent bits (a 5-bit exponent and 10-bit mantissa for float16 versus an 8-bit exponent and 7-bit mantissa for bfloat16). Intel and ARM are also adopting bfloat16 in their push to offer AI-enhanced hardware, while AMD has introduced software support for bfloat16 in recent versions of their ROC platform. As from 2020, the Ampere architecture from NVIDIA also introduces bfloat16 [3].

The goal of this Ph.D. thesis is to go beyond the state-of-the-art and analyze the impact of **custom precision floating point** in which both the bit width of the exponent and the mantissa can change without respecting any standard as well as **hardware implementation** targeting **energy efficiency**.

In the framework of the French research project AdaptING, the Electronic group at INL will work in collaboration with CEA-List. In this context we are currently looking for a (m/f) **PhD student** for a **3-years** contract to be supervised by Alberto Bosio (INL), Bastien Deveautour (INL) and Andrea Bocco (CEA).

Job description

The Ph.D. thesis is structured in the following 5 main tasks.

1. Bibliography (M0 to M6)

- Review the existing literature on Neural Network training algorithms and their implementations using Python and TensorFlow framework.
- Review the mixed precision training approaches and implementations.
- Review the custom floating point, the arithmetic and associated circuit implementations.

2. DNN custom floating precision (M6 to M12)

- The goal of this task is to become familiar with an in-house tool developed at INL lab called Inspect-NN [4]. This tool is based on TensorFlow and allow to replace standard multipliers with custom multipliers. Inspect-NN has to be extended in order to be able to use custom floating points formats

¹ <https://docs.nvidia.com/deeplearning/performance/mixed-precision-training/index.html>

(such as modified versions of the IEEE-754 formats, unum, posit, and others), and the associated arithmetic circuits models implemented in Python.

3. DNN exploration (M12-M22)

- In this task a set of representative DNNs models have to be selected. We plan to target DNNs designed for embedded systems and classification tasks such as EfficientNet, MobileNet, Squeezenet, etc..
- For the selected model, the exploration of training based on custom floating point will be done. Results will be compared with classical training approach and state of the art mixed precision training. Comparison is done by extracting the following metrics: Accuracy, Elapsed Time, Energy consumption. Energy consumption for custom floating-point training will be based on models since at this time of the work a hardware implementation is not yet available.
- This information will be used to select the training model to be accelerated in hardware. Before the hardware implementation, the student will have to profile the applications to better estimate what part of the training algorithm takes the major benefits to be accelerated in hardware.

4. Hardware Implementation (M22 – M30)

- This task aims to model the hardware accelerator needed to accelerate DNNs training, deduced from the circuit models from previous tasks. This task is divided in two main steps. The first one consists to implement an arithmetic functional model in Python to explore the optimal hardware implementation to implement the custom floating-point training. The second one consists to model, and then emulate, the optimal hardware accelerator, implemented as a hybrid solution integrating a dedicated hardware accelerator in to a microcontroller (e.g., RISC-V). The hardware accelerator will be based on open-source solutions, such as the GEMMINI [5], written in Chisel² (a HDL language made to prototype hardware units).

5. Manuscript preparation and dissemination (M30-M36)

- Scientific papers and thesis manuscript preparation.

Profile

You have or are about to obtain an MSc in Computer Engineer / Computer Science with strong experience in at least one of the following areas: computer architectures, digital circuit design (VHDL, system verilog), optimization algorithms. Good programming skills (python, C and C++) are required. Previous experience in Neural Networks is a plus (e.g., knowledge of major NN frameworks such as Pytorch and Tensorflow). Excellent written and verbal communication skills in English. Fluency in French is also a plus but not mandatory.

About INL

INL is a 250-strong research institute based in Lyon, France, carrying out fundamental and applied research in electronics, semiconductor materials, photonics and biotechnologies. The Electronic group is a leader in the area of advanced nanoelectronic design, with research projects and collaborations at both national and European level. Recent highlights include the development of genetic algorithms based multi objective design space exploration [6, 7].

Dates:

Ph.D. will start in 2024.

Environment:

The Ph.D. thesis will be done in collaboration with CEA-List. The Ph.D. candidate will be supervised by the INL team in Lyon (Ecole Centrale Campus). The Ph.D. salary will follow standard French rates.

References:

- [1] Y. LeCun, et al., "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.
- [2] E. Strubell, A. Ganesh, and A. McCallum, "Energy and Policy Considerations for Deep Learning in NLP." arXiv, 2019. doi: 10.48550/ARXIV.1906.02243. Available: <https://arxiv.org/abs/1906.02243>
- [3] P. Micikevicius *et al.*, "Mixed precision training," 2018, arXiv:1710.03740.
- [4] M. H. Ahmadilivani *et al.*, "Special Session: Approximation and Fault Resiliency of DNN Accelerators," 2023 IEEE 41st VLSI Test Symposium (VTS), San Diego, CA, USA, 2023, pp. 1-10, doi: 10.1109/VTS56346.2023.10140043.

² <https://www.chisel-lang.org>

[5] H. Genc et al., "Gemmini: Enabling Systematic Deep-Learning Architecture Evaluation via Full-Stack Integration," 2021 58th ACM/IEEE Design Automation Conference (DAC). IEEE, Dec. 05, 2021. doi: 10.1109/dac18074.2021.9586216. Available: <http://dx.doi.org/10.1109/DAC18074.2021.9586216>

Send CV and statement of purpose (in English or French) to

Alberto Bosio / INL - Lyon Institute of Nanotechnology - Ecole Centrale Lyon – email: alberto.bosio@ec-lyon.fr

Bastien Deveautour / INL - Lyon Institute of Nanotechnology - CPE – email: bastien.deveautour@cpe.fr

Andrea Bocco / CEA – email: Andrea.BOCCO@cea.fr