An internship proposal within Labex Cominlabs LeanAI project

Mixed Precision DNN Training with Multilevel Strategies

at ENS Lyon or Inria Rennes

Advisors: Elisa Riccietti (elisa.riccietti@ens-lyon.fr), Silviu Filip (silviu.filip@inria.fr)

Despite their massive success in training state-of-the-art deep learning models, gradient-based deep neural network (DNN) training methods such as stochastic gradient descent (SGD) still possess many weaknesses. Among them we have a low convergence rate that is heavily dependent on the tuning of the learning parameters, the well-known problem of vanishing and exploding gradients, and the fact that the backpropagation algorithm does not allow simultaneous weight updates across layers, somewhat limiting parallel execution. There has thus been significant interest in the development of alternative methods to tackle these difficulties and accelerate DNN training.

In the context of this internship we wish to focus on multilevel strategies to accelerate the training of DNNs. The basic idea of multilevel strategies is to exploit the fact that many problems can be represented at different scales, for instance an infinite dimensional problem can be discretized choosing different grid parameters [1]. These strategies are well-known in optimization [5,9] and have recently been used for the training of DNNs [2,3,7], showing that the use of hierarchical neural network models and multilevel training approaches allows for faster training, while still achieving good accuracy.

On the other hand, a recent research direction to speed up the training of DNNs is the use of low precision arithmetic. Traditionally, DNNs training has been mostly done using 32-bit floating-point [6] arithmetic, but recent efforts have shown that in many cases it is possible to use lower precision computations (even going down to sub 8-bit floating-point formats) for SGD-based methods and still converge to an acceptable result [4, 8, 10].

The **aim of the internship will be to combine these two ideas to design multilevel mixed precision training strategies**. To achieve this, we will explore two complementary research directions.

The first research direction will concern the study of neural network models, in which the parameters in the different layers can be coded with different numerical precisions. The goal will be to design dynamical strategies to choose the best numerical format for each layer, in order to accelerate training, while maintaining a good accuracy.

The second research direction focuses on training methods exploiting a hierarchy of models. In the existing hierarchical approaches for DNN training, each level of the hierarchy corresponds to a network model with a given number of layers. Training cost can be reduced by not performing expensive computations (such as the step computation in the optimization method) at higher levels (the ones with the largest number of parameters) and instead transferring cheap information computed at lower levels to the higher ones. In the context of the internship, we will extend these ideas by the use of multiple numeric formats. We will consider a hierarchy where at each level the parameters are coded with a given precision and the most expensive computations will be performed at the lowest levels (in which the parameters are coded in low precision). The key challenge is to devise efficient techniques to transfer information between levels.

The work will build upon a custom precision training simulation framework constructed atop PyTorch, called `mptorch`, developed by the internship coordinators.

The ideal candidate must be familiar with continuous optimization methods and how stochastic versions of these algorithms are used in supervised deep learning applications. Experience with Python programming is also required. Knowledge of deep learning frameworks such as Pytorch is also desirable.

References

[1] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A multigrid tutorial.* SIAM, 2000.

[2] H. Calandra, S. Gratton, E. Riccietti, and X. Vasseur. On a multilevel Levenberg–Marquardt method for the training of artificial neural networks and its application to the solution of partial differential equations. *Optimization Methods and Software*, pages 1–26, 2020.

[3] L. Gaedke-Merzhäuser, A. Kopaničáková, and R. Krause. Multilevel minimization for deep residual networks. *arXiv preprint arXiv:2004.06196*, 2020.

[4] N. Mellempudi, S. Srinivasan, D. Das, and B. Kaul. Mixed precision training with 8-bit floating point. *arXiv preprint arXiv:1905.12334*, 2019.

[5] A. Migdalas, P. M. Pardalos, and P. Värbrand. *Multilevel optimization: algorithms and applications*, volume 20. Springer Science & Business Media, 2013.

[6] J.-M. Muller, N. Brunie, F. de Dinechin, C.-P. Jeannerod, M. Joldes, V. Lefèvre, G. Melquiond, N. Revol, and S. Torres. *Handbook of Floating-Point Arithmetic*. Birkhäuser Boston, 2nd edition, 2018.

[7] C. Scott and E. Mjolsness. Multilevel artificial neural network training for spatially correlated learning. *SIAM Journal on Scientific Computing*, 41(5):S297–S320, 2019.

[8] N. Wang, J. Choi, D. Brand, C.-Y. Chen, and K. Gopalakrishnan. Training deep neural networks with 8-bit floating point numbers. In *Advances in Neural Information Processing Systems*, pages 7675–7684, 2018.

[9] S. Yang, M. Wang, and E. X. Fang. Multilevel stochastic gradient methods for nested composition optimization. *SIAM Journal on Optimization*, 29(1):616–659, 2019.

[10] P. Zamirai, J. Zhang, C. R. Aberger, and C. De Sa. Revisiting BFloat16 Training. *arXiv preprint arXiv:2010.06192*, 2020.