Building a Large Neural Network on a Single Machine

Project proposal

Supervisor: Luís Alexandre (luis.alexandre@ubi.pt)

1 Goals

One of the most important techniques in Artificial Intelligence are Neural Networks (NNs). While there has been recently an enormous improvement in the capacity that these networks have, common networks are still very far away from the number of parameters the human brain is supposed to store (10^{15}).

There have been some advances in the training of large networks, such as the work in [1], where 512 CPU cores were used to train a model with 1.7×10^9 parameters, or the use of parallelization platforms, for instance, [3] used MapReduce with 32 PCs and [4] used MapReduce (Hadoop) and Giraph (although, in this case, the work was for spiking neurons) to train 100000 neurons, with over 10^9 parameters using over 24GB of RAM. Our work [2] also showed how to take advantage of distributed learning, by using a smart problem splitting technique, in this case applied to convolutional neural networks.

One of the most difficult problems is, of course, the communication between distributed models: in [5], they propose Linear Pipelining to reduce the communication bottleneck. If the model stays in a single machine, these problems can be considerably alleviated.

With this project we want to study the implementation limits of NNs using a single machine if we consider the use of disk space as a way to store information in run-time for the network when only part of it fits the RAM.

The work will be done in linux. We will use PyThorch for the implementation.

2 Work plan

The project has the following tasks:

T1 Introduction to neural network learning (2 weeks).

T2 Study the distributed and large scale network learning approaches (3 weeks).

T3 Implement new code to train a large network on a single machine taking advantage of hard disk space (7 weeks).

T4 Write the project's report (3 weeks).

3 Technical and Academic Requirements

Be able to program using Python on Linux, use a source code repository and produce documentation (using doxygen, sphinx or other similar tool).

It is desirable that the student has grades above 13 on the following courses: Estruturas de Dados, Probabilidades e Estatística, Inteligência Artificial.

4 Expected Results

- Source code and documentation of all code developed
- Project report

5 References

[1] Jeffrey Dean, Greg S. Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Quoc V. Le, Mark Z. Mao, Marc'Aurelio Ranzato, Andrew Senior, Paul Tucker, Ke Yang, and Andrew Y. Ng. Large scale distributed deep networks. In *NIPS*, 2012.

[2] Jose Marques, Gabriel Falcao, and Luís A. Alexandre. Distributed learning of CNNs on heterogeneous CPU/GPU architectures. *Applied Artificial Intelligence*, 32(9-10):822–844, 2018.

[3] Kairan Sun, Xu Wei, Gengtao Jia, Risheng Wang, and Ruizhi Li. Large-scale artificial neural network: Mapreduce-based deep learning. *CoRR*, abs/1510.02709, 2015.

[4] S. Yang, N. D. Spielman, J. C. Jackson, and B. S. Rubin. Large-scale neural modeling in mapreduce and giraph. In *IEEE International Conference on Electro/Information Technology*, pages 556–561, June 2014.

[5] Yiyang Zhao, Linnan Wang, Wei Wu, George Bosilca, Richard W. Vuduc, Jinmian Ye, Wenqi Tang, and Zenglin Xu. Efficient communications in training large scale neural networks. In *Proceedings of the on Thematic Workshops of ACM Multimedia 2017, Mountain View, CA, USA, October 23 - 27, 2017*, pages 110–116, 2017.