MLforCI and CIforML: Future Cyberinfrastructure

Geoffrey Fox, Indiana University and Shantenu Jha, Rutgers University and Brookhaven National Laboratory
February 23 2020

Introduction

Here we review the complementary areas of Machine Learning to enhance Cyberinfrastructure (MLforCI or what we called MLforHPC in previous work [1], [2]) and Cyberinfrastructure to enhance Machine Learning (CIforML or what we called HPCforML). Rather than discuss whether to use the term HPC or CI, we should humbly reference the original contribution [3] "Machine Learning for Systems and. Systems for Machine Learning" by Jeffrey Dean of Google at NeurIPS 2017. So we use CI, HPC or Systems interchangeably. In general terms, both areas MLforCI and CIforML are very promising with CIforML offering a transformative vision for simulations with speedups of 2.10^9 reported in a recent paper [4]; effective performance of zettascale and yottascale should be achieved across many fields. Such increases coming from essentially new algorithms, are much larger than those possible with just novel hardware.

CIforML will drive the Big Data revolution with HPC techniques, often developed for simulations, integrated into deep learning (DL) and machine learning (ML). Nowadays DL is tending to replace classic ML in many large scale data analytics (including those used in CIforML) and so HPC is particularly important in DL and use of HPC for DL is already pervasive in both academic and industry problems. In this merger of Big Data and simulations ideas, one can expect greater use of Big Data software (the Apache Big Data Stack [5]) across the board. In greater detail, we can summarize opportunities and status as follows

MLforCI Remarks

Here we classified [6] approaches into 8 detailed and 3 high-level areas with the latter being:

1. Improving Simulation with Configurations and Integration of Data
2. Learn Structure, Theory, and Model for Simulation
3. Learn Surrogates for Simulation

We have general remarks on MLforCI below but first comment on the styles of use for MLforCI given above. MLforCI is broadly applicable but current use is nonuniform across domains with material science and biomolecular simulations being particularly advanced. A major need is to support the use of MLforCI over a broader range of applications to understand it better, gather requirements and opportunities and use this to improve Cyberinfrastructure support to make MLforCI more effective for more users and make it a frontline supported service at production computing centers.

The category **Improving Simulation with Configurations and Integration of Data** includes ML and DL enhancements of the well-established ideas: Autotuning and data assimilation. The former is an important area but not likely to lead to large improvements as it keeps the simulation largely changed. We found [7] a factor of 3 performance increase using a learning

network to choose the time step dynamically and make a better prediction for a consistency factor. Data assimilation is illustrated by use of neural nets to represent microscopic structure such as that in climate and weather prediction to represent the effects of cloud cover. The current results [8] are promising but more applications are needed.

The category **Learn Structure, Theory, and Model for Simulation** includes smart ensembles and collective coordinates with performance gains of up to 10^8 reported [9]. Here we also see the learning of macroscopic structure such as potentials and coarse-graining. The scaling laws of N^2 to N^7 in many-particle potential makes the learning of potentials as a function of particle position very attractive and successful in many cases.

The category **Learn Surrogates for Simulation** is perhaps the most exciting. A recent paper reports surrogates learned for ten simulation areas covering astrophysics, climate science, biogeochemistry, high energy density physics, fusion energy, and seismology with effective speedups reaching 2.10^9 [4]. This is being used commercially to predict promising drugs [10] (a generalized QSAR process also applied in material science) and by General Electric [11] to allow interactive exploration of aircraft engines. It is interesting that training set sizes vary from hundreds to 14,000; fewer than I would have guessed were necessary. This approach seems particularly promising for agent-based simulations (seen in Sociotechnical simulations [12] and in virtual tissues) as a surrogate for an agent can be used for all agents in the problem leading to significant real speedup. Note surrogates are most effective when they need to be used many times so the time to generate the training set does not weigh down the observed effective performance. Education is a good use case where surrogates will naturally be used many times [13]. These networks are typically convolutional or fully connected. One can also use Recurrent Neural Nets (LSTM) to learn numerical differential operators where the surrogate allows much larger time steps up to 4000 times that of traditional particle dynamics solvers [14].

There is much exciting research needed into surrogates including using them in the most sophisticated approaches such as fast multipoles.

There are some observations that cross all the MLforCI categories.

1. We must of course design and build software systems that support ML and DL dynamically mixed with simulation. This will be particularly important with a new generation of accelerators optimized for ML and DL which will probably not support simulation in contrast to GPUs that support both simulation and DL. This should include a study of the optimized hardware – CPU, Accelerator, Storage, Network for ML+DL dynamically mixed with simulation
2. We need to learn errors as well as values in differential equation surrogates
3. We need to investigate the many different forms of deep learning where, for example, commercial applications are switching to the so-called transformer networks with “attention” mimicking history in a recurrent network. Autoencoders, Reinforcement Learning, and GANs are also important deep learning approaches for MLforCI.
4. DL is making transformational changes in many areas including the geospatial time series illustrated by the differential equation LSTM surrogates mentioned above.

Currently, this is mainly pursued commercially in transportation systems and e-commerce logistics but there is a clear opportunity to understand the many possible applications to scientific data streams.

CIforML Remarks

We have noted that this is already successful and mature as ML and DL tend to have kernels such as linear algebra that we already know how to parallelize with high performance. Graph algorithms are a distinctive class of ML problems where dynamic sparsity can be challenging for good performance but recently good progress has been seen [15]. As well as classic data-center big data problems, we need to support real-time and streaming edge use cases. We suggest stronger involvement with the MLPerf collaboration [16] by the cyberinfrastructure community. We need to extend their datasets to include scientific examples and extend platforms where benchmarks are run. MLPerf's goal of studying Machine learning performance is very important for the scientific research community and we need to design and deploy a high performance deep learning environment meeting the requirements of scientific data analytics. We expect that science like the commercial world will see a growing use of deep learning, especially for the largest and most complex data analysis problems.

A high performance deep learning system offers an amazingly rich parallel computing environment allowing:

1. **Data-parallelism:** Decompose tensor index corresponding to data into blocks i.e. divide the mini-batch into micro-batches run simultaneously on parallel nodes. Needs AllReduce communication [17]
2. **Inter-layer model (pipeline) parallelism**: Decompose model by groups of layers in the model. Just needs a distributed copy [18], [19]
3. **Intra-layer model parallelism**: Decompose one or more other tensor indices. This is classic parallel computing and only case where parallelism requires changes to user code to implement some variant of MPI parallelism. [20], [21]
4. **Hyper-parameter search parallelism** often with a genetic algorithm [22]

Conclusions

This analysis highlights the challenges and opportunities for using and supporting high performance deep learning systems. Challenges include the storage architecture and integration between deep learning and other data management and analysis services. Further, all of this needs to be efficiently integrated into the total shared system which needs to support the major classes of large scale computing applications such as:

1. MLforCI discussed above
2. Classic data-center big data problem
3. Edge to Cloud data-center use case
4. Simulation supercomputing

Most importantly we need to explore and enable the transformative opportunities offered by MLforCI where the largest gains come from changing algorithms and hence the application code. This implies that collaboration is needed between cyberinfrastructure and domain

scientists while the rapidly evolving and very sophisticated deep learning methods require training material aimed at Deep Learning (or AI) for Science.

Acknowledgments

Partial support by NSF CIF21 DIBBS 1443054, NSFnanoBIO 1720625, and NSF BDEC2 1849625 is gratefully acknowledged. We thank the “Learning Everywhere” collaboration James A. Glazier, JCS Kadupitiya, Vikram Jadhao, Minje Kim, Judy Qiu, James P. Sluka, Endre Somogyi, Madhav Marathe, Abhijin Adiga, Jiangzhuo Chen, and Oliver Beckstein for many discussions. SJ is partially supported by DOE ECP “ExaLearn”.

References

[1] Geoffrey Fox, Shantenu Jha, “Understanding ML driven HPC: Applications and Infrastructure,” in *IEEE eScience 2019 Conference*, San Diego, California [Online]. Available: https://escience2019.sdsc.edu/

[2] Geoffrey Fox, James A. Glazier, JCS Kadupitiya, Vikram Jadhao, Minje Kim, Judy Qiu, James P. Sluka, Endre Somogyi, Madhav Marathe, Abhijin Adiga, Jiangzhuo Chen, Oliver Beckstein, and Shantenu Jha, “Learning Everywhere: Pervasive Machine Learning for Effective High-Performance Computation,” in *HPDC Workshop at IPDPS 2019*, Rio de Janeiro, 2019 [Online]. Available: https://arxiv.org/abs/1902.10810, http://dsc.soic.indiana.edu/publications/Learning_Everywhere_Summary.pdf

[3] Jeff Dean, “Machine Learning for Systems and Systems for Machine Learning,” in *Presentation at 2017 Conference on Neural Information Processing Systems*, Long Beach, CA [Online]. Available: http://learningsys.org/nips17/assets/slides/dean-nips17.pdf

[4] M. F. Kasim, D. Watson-Parris, L. Deaconu, S. Oliver, P. Hatfield, D. H. Froula, G. Gregori, M. Jarvis, S. Khatiwala, J. Korenaga, J. Topp-Mugglestone, E. Viezzer, and S. M. Vinko, “Up to two billion times acceleration of scientific simulations with deep neural architecture search,” *arXiv [stat.ML]*, 17-Jan-2020 [Online]. Available: http://arxiv.org/abs/2001.08055

[5] O. Beckstein, G. Fox, J. Qiu, D. Crandall, G. von Laszewski, J. Paden, S. Jha, F. Wang, M. Marathe, A. Vullikanti, and T. Cheatham, “Contributions to High-Performance Big Data Computing,” in *HPCC2018 Cetraro proceedings*, 2019 [Online]. Available: http://dsc.soic.indiana.edu/publications/SPIDALPaperSept2018.pdf

[6] Geoffrey Fox, Shantenu Jha, “Learning Everywhere: A Taxonomy for the Integration of Machine Learning and Simulations,” in *IEEE eScience 2019 Conference*, San Diego, California [Online]. Available: https://escience2019.sdsc.edu/

[7] JCS Kadupitiya, Geoffrey C. Fox, Vikram Jadhao, “Machine Learning for Parameter Auto-tuning in Molecular Dynamics Simulations: Efficient Dynamics of Ions near Polarizable Nanoparticles,” *Int. J. High Perform. Comput. Appl.*, Nov. 2018 [Online]. Available: http://dsc.soic.indiana.edu/publications/Manuscript.IJHPCA.Nov2018.pdf

[8] S. Rasp, M. S. Pritchard, and P. Gentine, “Deep learning to represent subgrid processes in climate models,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 115, no. 39, p. 9684, Sep. 2018 [Online]. Available: http://www.pnas.org/content/115/39/9684.abstract

[9] H. Jung, R. Covino, and G. Hummer, “Artificial Intelligence Assists Discovery of Reaction Coordinates and Mechanisms from Molecular Dynamics Simulations,” *arXiv [physics.chem-ph]*, 14-Jan-2019 [Online]. Available: http://arxiv.org/abs/1901.04595

[10] “A Molecule Designed by AI Exhibits ‘Druglike’ Qualities: Insilico Medicine is among several startups trying to harness artificial intelligence to speed up development of drugs.” [Online]. Available: https://www.wired.com/story/molecule-designed-ai-exhibits-druglike-qualities/. [Accessed: 23-Feb-2020]

[11] J. A. Tallman, M. Osusky, N. Magina, and E. Sewall, “An Assessment of Machine Learning Techniques for Predicting Turbine Airfoil Component Temperatures, Using FEA Simulations for Training Data,” in *ASME Turbo Expo 2019: Turbomachinery Technical Conference and Exposition*, 2019 [Online]. Available: https://asmedigitalcollection.asme.org/GT/proceedings-abstract/GT2019/58646/V05AT20A002/1066873. [Accessed: 23-Feb-2020]

[12] Lijing Wang, J Chen, and Madhav Marathe., “DEFSI : Deep Learning Based Epidemic Forecasting with Synthetic Information,” in *Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)*, Hilton Hawaiian Village, Honolulu, Hawaii, USA, 2019 [Online]. Available: https://www.researchgate.net/publication/328639130_DEFSI_Deep_Learning_Based_Epidemic_Forecasting_with_Synthetic_Information

[13] JCS Kadupitiya , Geoffrey C. Fox , and Vikram Jadhao, “Machine learning for performance enhancement of molecular dynamics simulations,” in *International Conference on Computational Science ICCS2019*, Faro, Algarve, Portugal, 2019 [Online]. Available: http://dsc.soic.indiana.edu/publications/ICCS8.pdf

[14] J. C. S. Kadupitiya, G. Fox, and V. Jadhao, “Recurrent Neural Networks Based Integrators for Molecular Dynamics Simulations,” in *APS March Meeting 2020*, 2020 [Online]. Available: http://meetings.aps.org/Meeting/MAR20/Session/L45.2. [Accessed: 23-Feb-2020]

[15] L. Chen, J. Li, C. Sahinalp, M. Marathe, A. Vullikanti, A. Nikolaev, E. Smirnov, R. Israfilov, and J. Qiu, “Subgraph2vec: Highly-vectorized tree-like subgraph counting,” in *2019 IEEE International Conference on Big Data,* Los Angeles [Online]. Available: http://dsc.soic.indiana.edu/publications/Bigdata_Subgraph2Vec.pdf

[16] “MLPERF benchmark suite for measuring performance of ML software frameworks, ML hardware accelerators, and ML cloud platforms.” [Online]. Available: https://mlperf.org/. [Accessed: 08-Feb-2019]

[17] Uber Engineering, “Horovod: Uber’s Open Source Distributed Deep Learning Framework for TensorFlow.” [Online]. Available: https://eng.uber.com/horovod/. [Accessed: 08-Feb-2019]

[18] Y. Huang, Y. Cheng, A. Bapna, O. Firat, D. Chen, M. Chen, H. Lee, J. Ngiam, Q. V. Le, Y. Wu, and Z. Chen, “GPipe: Efficient Training of Giant Neural Networks using Pipeline Parallelism,” in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d\textquotesingle Alché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 103–112 [Online]. Available: http://papers.nips.cc/paper/8305-gpipe-efficient-training-of-giant-neural-networks-using-pipeline-parallelism.pdf

[19] D. Narayanan, A. Harlap, A. Phanishayee, V. Seshadri, N. R. Devanur, G. R. Ganger, P. B. Gibbons, and M. Zaharia, “PipeDream: generalized pipeline parallelism for DNN training,” in *Proceedings of the 27th ACM Symposium on Operating Systems Principles*, 2019, pp. 1–15 [Online]. Available: https://dl.acm.org/doi/abs/10.1145/3341301.3359646

[20] “Tensorflow MESH Project.” [Online]. Available: https://github.com/tensorflow/mesh. [Accessed: 20-Jan-2020]

[21] N. Maruyama, “Scalable Distributed Training of Large Neural Networks with LBANN,” 2019 [Online]. Available: http://mug.mvapich.cse.ohio-state.edu/static/media/mug/presentations/19/maruyama-mug-19.pdf

[22] “Multi-node Evolutionary Neural Networks for Deep Learning (MENNDL) web page.” [Online]. Available: https://www.ornl.gov/division/csmd/projects/multi-node-evolutionary-neural-networks-deep-learning-menndl. [Accessed: 20-Jan-2020]