# HPC AI500: A Benchmark Suite for HPC AI Systems

Zihan Jiang[1,2], Wanling Gao[1,2,3], Lei Wang[1,3], Xingwang Xiong[1,2], Yuchen Zhang[5], Xu Wen[1,2], Chunjie Luo[1], Hainan Ye[3,4], Xiaoyi Lu[6], Yunquan Zhang[9], Shengzhong Feng[7], Kenli Li[8], Weijia Xu[10], and Jianfeng Zhan[1,2,3] ⋆

[1] State Key Laboratory of Computer Architecture, Institute of Computing Technology, Chinese Academy of Sciences
[2] University of Chinese Academy of Sciences, China
[3] BenchCouncil (International Open Benchmarking Council)
[4] Beijing Academy of Frontier Sciences and Technology
[5] State University of New York at Buffalo
[6] Department of Computer Science and Engineering, The Ohio State University
[7] National Supercomputing Center in Shenzhen, China
[8] National Supercomputing Center in Changsha, China
[9] National Supercomputing Center in Jinan, China
[10] Texas Advanced Computing Center, The Texas University at Austin
{jiangzihan,gaowanling,
zhanjianfeng,wanglei_2011,xiongxingwang,wenxu,luochunjie,
gaowanling,zyq}@ict.ac.cn
{zhang232}@buffalo.edu
{fengsz}@nsccsz.cn
{lkl}@hnu.edu.cn
{xwj}@tacc.utexas.edu
{yehainan}@mail.bafst.com
{luxi}@cse.ohio-state.edu

**Abstract.** In recent years, with the trend of applying deep learning (DL) in high performance scientific computing, the unique characteristics of emerging DL workloads in HPC raise great challenges in designing, implementing HPC AI systems. The community needs a new yard stick for evaluating the future HPC systems. In this paper, we propose HPC AI500 — a benchmark suite for evaluating HPC systems that running scientific DL workloads. Covering the most representative scientific fields, each workload from HPC AI500 is based on real-world scientific DL applications. Currently, we choose 14 scientific DL benchmarks from perspectives of application scenarios, data sets, and software stack. We propose a set of metrics for comprehensively evaluating the HPC AI systems, considering both accuracy, performance as well as power and cost. We provide a scalable reference implementation of HPC AI500. The specification and source code are publicly available from `http://www.benchcouncil.org/HPCAI500/index.html`. Meanwhile, the AI benchmark suites for datacenter, IoT, Edge are also released on the BenchCouncil web site.

**Keywords:** HPC · Deep Learning · Benchmarking

⋆ Jianfeng Zhan is the corresponding author.

## 1 Introduction

The huge success of AlexNet [1] in the ImageNet [2] competition marks that deep learning(DL) is leading the renaissance of Artificial Intelligence (AI). Since then, a wide range of application areas have started using DL and achieved unprecedented results, such as image recognition, natural language processing, and even autonomous driving. In the commercial fields, many DL-based novel applications have emerged, creating huge economic benefits. In the fields of high performance scientific computing, similar classes of problems are faced, i.e., predicting extreme weather [21], finding signals of new particles [22], and estimating cosmological parameters [23]. These scientific fields are essentially solving the common class of problems that exist in commercial fields such as classifying images, predicting classes labels, or regressing a numerical quantity. In several scientific computing fields, DL has replaced traditional scientific computing methods and becomes a promising tool [24].

As an emerging workload in high performance scientific computing, DL has many unique features compared to traditional high performance computing. First, training a DL model depends on massive data that are represented by high-dimensional matrices. Second, leveraging deep learning frameworks such as Tensorflow [3] and caffe [4] aggravates the difficulty of the software and hardware co-design. Last but not least, the heterogeneous computing platform of DL is far more complicated than traditional scientific workloads, including CPU, GPU, and various domain-specific processor (e.g. Cambricon Diannao [5] or Google TPU [6]). Consequently, the community requires a new yardstick for evaluating future HPC AI systems. However, the diversity of scientific DL workloads raise great challenges in HPC AI benchmarking.

1. Dataset: Scientific data is often more complex than MINST or ImageNet data sets. First, the shape of scientific data can be 2D images or higher-dimension structures. Second, there are hundreds of channels in a scientific image, while the popular image data often consists of only RGB. Third, Scientific datasets are always terabytes or even petabytes in size.
2. Workloads: Modern scientific DL doesn't adopt off-the-shelf models, instead builds more complex model with domain scientific principles (e.g. energy conservation) [21].
3. Metrics: Due to the importance of accuracy, using a single performance metric such as FLOPS leads to insufficient evaluation. For a comprehensively evaluation, the selected metrics should not only consider the performance of the system, but also consider the accuracy of the DL model [8].
4. Scalability: Since the scientific DL workloads always run on the supercomputers, which are equipped with tens of thousands nodes, the benchmark program must be highly scalable.

Most of the existing AI benchmarks [8, 29, 9, 7, 28, 10] are based on commercial scenarios. Deep500 [30] is a benchmarking framework aiming to evaluate high-performance

deep learning. However, its reference implementation uses commercial open source data sets and simple DL models, hence cannot reflect real-world HPC AI workloads. We summary these major benchmarking efforts for AI and compare them with HPC AI500 as shown in the table below.

Table 1: Comparison of AI Benchmarking Efforts.

| Benchmark Efforts | Datasets | Problem domains | | | | Implementation | |
|---|---|---|---|---|---|---|---|
| | | Scientific | | | Commercial | Standalone | Distributed |
| | | EWA[1] | Cos[2] | HEP[3] | | | |
| HPC AI500 | Scientific data | ✓ | ✓ | ✓ | × | ✓ | ✓ |
| TBD | Commercial data | × | × | × | ✓ | ✓ | × |
| MLPerf | Commercial data | × | × | × | ✓ | ✓ | × |
| DAWNBench | Commercial data | × | × | × | ✓ | ✓ | × |
| Fathom | Commercial data | × | × | × | ✓ | ✓ | × |
| Deep500 | Commercial data | Framework, undefined | | | | ✓ | ✓ |

[1] Extreme Weather Analysis
[2] Cosmology
[3] High Energy Physics

Consequently, targeting above challenges, we propose HPC AI500—a benchmark suite for HPC AI systems. Our major contributions are as follows:

1. We create a new benchmark suite that covers the major areas of high performance scientific computing. The benchmark suite consists of micro benchmarks and component benchmarks. The workloads from component benchmarks use the state-of-the-art models and representative scientific data sets to reflect the real-world performance results. In addition, we select several DL kernels as the micro benchmarks for evaluating the upper bound performance of the systems.
2. We propose a set of metrics for comprehensively evaluating the HPC AI systems. Our metrics for component benchmarks include both accuracy and performance. For micro benchmarks, we provide metrics such as FLOPS to reflect the upper bound performance of the system.

Coordinated by BenchCouncil ( `http://www.benchcouncil.org`), we also release the datacenter AI benchmarks [17, 16], the IoT AI benchmarks [15], edge AI benchmarks [14], and big data benchmarks [12, 13], which are publicly available from `http://www.benchcouncil.org/HPCAI500/index.html`.

## 2 Deep Learning in Scientific Computing

In order to benchmark HPC AI systems, the first step is to figure out how DL works in scientific fields. Although it is an emerging field, several scientific fields have applied DL to solve many important problems, such as extreme weather analysis [40–42, 21], high energy physics [36–39, 22], and cosmology [23, 26, 33–35].

### 2.1 Extreme Weather Analysis

Extreme Weather Analysis (EWA) poses a great challenge to human society. It brings severe damage to people health and economy every single year. For instance, the heatwaves in 2018 caused over 1600 deaths according to the UN report [44]. And the landfall of hurricane Florence and Michael caused about 40 billion dollars worth of damage to US economy [45]. In this context, understanding extreme weather life cycle and even predicting its future trend become a significant scientific goal. Achieving this goal always requires accurately identifying the weather patterns to acquire the insight of climate change based on massive climate data analysis. Traditional climate data analysis methods are built upon human expertise in defining multi-variate thresholds of extreme weather events. However, it has a major drawback: there is no commonly held set of criteria that can define a weather event due to the man-made subjectivism, which leads to inaccurate pattern extraction. Therefore, DL has become another option for climate scientists. Liu et al. (2016) [40] develop a relatively simple CNN model with two convolutional layers to classify three typical extreme weather events and achieve up to 99% accuracy. Racah et al. (2017) [42] implement a multichannel spatiotemporal CNN architecture for semi-supervised prediction and exploratory extreme weather data analysis. GlobeNet [41] is a CNN model with inception units for typhoon eye tracking. Kurth et al. (2018) [21] use variants of Tiramisu and DeepLabv3+ neural networks which are both built on Residual Network (ResNet) [20]. They deployed these two networks on Summit and firstly achieved exascale deep learning for climate analysis.

### 2.2 High Energy Physics

Particle collision is the most important experiment approach in High Energy Physics (HEP). Detecting the signal of new particle is the major goal in experimental HEP. Today's HEP experimental facility such as LHC creates particle signals with hundreds of millions channels with a high data rate. The signal data from different channels in every collision usually are represented as a sparse 2d image, so called a jet-image. In fact, accurately classifying these jet-images is the key to find signals of new particles. In recent years, due to the excellent performance in pattern recognition, DL has become the focus of the data scientists in HEP community and has a tendency to go mainstream. Oliveira et al. (2016) [38] use a CNN model with 3 convolutional layers to tag jet-images. They firstly demonstrated that using DL not only improve the discrimination power, but also gain new insights compared to designing physics-inspired features. Komiske et al. (2017) [39] adopt a CNN model to discriminate quark and gluon jet-image. Kurth et al.(2017) [22] successfully deploy CNN to analyze massive HEP data on the HPC system and achieve petaflops performance. Their work is the first attempt at scaling DL on large-scale HPC systems.

### 2.3 Cosmology

Cosmology is a branch of astronomy concerned with the studies of the origin and evolution of the universe, from the Big Bang to today and on into the future [49]. In 21st

century, the most fundamental problem in cosmology is the nature of dark energy. However, this mysterious energy greatly affects the distribution of matter in the universe that is described by cosmological parameters. Thus, accurately estimating these parameters is the key to understand the insight of the dark energy. For solving this problem, Ravanbakhsh et al. (2017) [26] firstly propose a 3D CNN model with 6 convolutional layers and 3 fully-connected layers and opens the way to estimating the parameters with high accuracy. Mathuriya et al. (2018) propose CosmoFlow [23], which is a project aiming to process large 3D cosmology dataset on HPC systems. They extend the CNN model designed by Ravanbakhsh et al. (2017) [26]. Meanwhile, in order to guarantee the high fidelity numerical simulations and avoid the use of expensive instruments, generating high quality cosmological data is also important. Ravanbakhsh et al. (2017) [33] propose a deep generative model for acquiring high quality galaxy images. Their results show a reliable alternative for generating the calibration data of cosmological surveys.

### 2.4 Summary

After investigating the above representative scientific fields, we have identified the representative DL applications and abstracted these DL applications into classical AI tasks. As shown in Table 2, almost all the applications are essentially using CNN to extract the patterns of various scientific image data. From this perspective, *image recognition*, *image generation*, and *object detection* are the most important tasks in modern scientific DL. In our benchmark methodology (Section 3.1), we use these three classic AI tasks as the component workloads of the HPC AI500 Benchmark.

Table 2: Modern Scientific Deep Learning.

| Scientific Fields | DL Applications | Classical DL Tasks | Model Type |
|---|---|---|---|
| Extreme Weather Analysis | Identify weather patterns | Object Detection | CNN |
| High Energy Physics | Jet-images discrimination | Image Recognition | CNN |
| Cosmology | Estimate parameters<br>Galaxy image generation | Image Recognition<br>Image Generation | CNN |

## 3 Benchmarking Methodology and Decisions

### 3.1 Methodology

Our benchmarking methodology is shown in Figure 1, similar to that [12]. As HPC AI is an emerging and evolving domain, we take an incremental and iterative approach. First of all, we investigate the scientific fields that use DL widely. As mentioned in Section 2, *extreme weather analysis, high energy physics*, and *cosmology* are the most representative fields. Then, we pay attention to the typical DL workloads and data sets in these three application fields.

In order to cover the diversity of workloads, we focus on the critical tasks that DL has performed in the aforementioned fields. Based on our analysis in Section 2, we extracts three important component benchmarks that can represent modern scientific DL, namely *image recognition*, *image generation*, and *object detection*. This shows that CNN models play an important role. In each component, we choose the state-of-the-art model and software stack from the applications. We also select the hotspot DL operators as the micro benchmark for evaluating upper bound performance of the system.

We chose three real-world scientific data sets from aforementioned scientific fields and consider their diversity from the perspective of data formats. In modern DL, the raw data is always transformed into matrix for downstream processing. Therefore, we classify these matrices into three kinds of formats: 2D sparse matrix, 2D dense matrix, and 3 dimensional matrix. In each matrix format, we also consider the unique characteristics (e.g., multichannel that more than RGB, high resolution) in the scientific data.

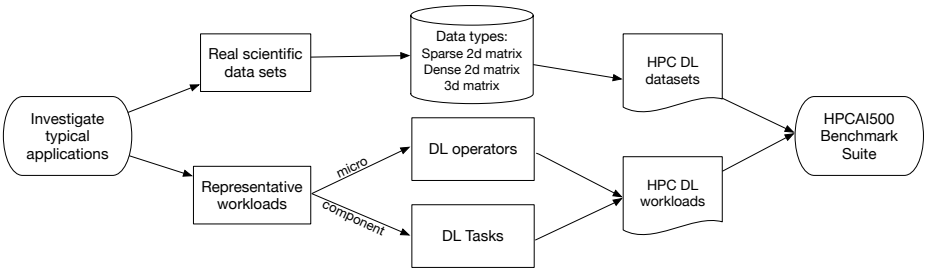


Fig. 1: HPCAI500 Methodology

### 3.2 The Selected Datasets

We investigate the representative data sets in our selected scientific fields and collect three data sets as shown in Table 3. Our selection guidelines follow the aforementioned benchmarking methodology.

**The Extreme Weather Data set** [46] is made up of 26-year of climate data. The data of every year is available as one HDF5 file. Each HDF5 file contains two data sets: images and boxes. *Images data set has 1460 example dense images* (4 per day, 365 days per year) with 16 channels. Each channel is 768 * 1152 corresponding to one measurement per 25 square km on earth. Boxes dataset records the coordinates of the four extreme weather events in the corresponding images: tropical depression, tropical cyclone, extratropical cyclone and the atmospheric river.

**The HEP Data set** [51] is divided into two classes: the RPV-Susy signal and the most prevalent background. The training data set is composed of around 400 k jet-images. Each jet-image is represented as a 64*64 sparse matrix and has 3 channels. It also provides validation and test data. All the data are generated by using the Pythia event generator [52] interfaced to *the Delphes fast detector simulation* [38].

**The Cosmology Data set** [23] aims to predict the parameters of cosmology. It is based on dark matter N-body simulations produced using the MUSIC [53] and pycola [54] packages. Each simulation covers the volumes of $512h^{-1}Mpc^3$ and contains $512^3$ dark matter particles.

Table 3: The Chosen Datasets

| Dataset | Data Format | Scientific Features |
|---|---|---|
| Extreme Weather Dataset | 2D dense matrix | high resolution, multichannel |
| HEP Dataeset | 2D sparse matrix | multichannel |
| Cosmology Dataset | 3D matrix | multidimensional |

### 3.3 The Selected Workloads

**Component Benchmarks** Since object detection, image recognition, and image generation are the most representative DL tasks in modern scientific DL. We choose the following state-of-the-art models as the HPC AI500 component benchmarks.

*Faster-RCNN* [61] targets real-time object detection. Unlike the previous object detection model [62, 63], it replaces the selective search by a region proposal network that achieves nearly cost-free region proposals. Further more, Faster-RCNN combines the advanced CNN model as their base network for extracting features and is the foundation of the 1st-place winning entries in ILSVRC'15 (ImageNet Large Scale Visual Recognition Competition).

*ResNet* [27] is a milestone in Image Recognition, marking the ability of AI to identify images beyond humans. It solves the degradation problem, which means in the very deep neural network the gradient will gradually disappear in the process of propagation, leading to poor performance. Due to the idea of ResNet, researchers successfully build a 152-layer deep CNN. This ultra deep model won all the awards in ILSVRC'15.

*DCGAN* [64] is one of the popular and successful neural network for GAN [65]. Its fundamental idea is replacing fully connected layers with convolutions and using transposed convolution for upsampling. The proposal of DCGAN helps bride the gap between CNNs for supervised learning and unsupervised learning.

**Micro Benchmarks** We choose the following primary operators in CNN as our micro benchmarks.

*Convolution* In mathematics, convolution is a mathematical operation on two functions to produce a third function that expresses how the shape of one is modified by the other [55]. In a CNN, convolution is the operation occupying the largest proportion,

which is the multiply accumulate of the input matrix and the convolution kernel, and then produces feature maps. There are many convolution kernels distributed in different layers responsible for learning different level features.

*Full-connected* The full-connected layer can be seen as the classifier of a CNN, which is essentially matrix multiplication. It is also the cause of the explosion of CNN parameters. For example, in AlexNet [1], the number of training parameters of fully-connected layers reaches about 59 million and accounts for 94 percent of the total.

*Pooling* Pooling is a sample-based discretization process. In a CNN, the objective of pooling is to down-sample the inputs (e.g., feature maps), which leads to the reduction of dimensionality and training parameters. In addition, it enhances the robustness of the whole network. The commonly used pooling operations including max-pooling and average-pooling.

Table 4: The Summary of HPC AI500 Benchmark.

| App Scenarios | Workloads | Fields | Datasets | Data Format | Software Stack |
|---|---|---|---|---|---|
| Micro Benchmarks | Convolution<br>Pooling<br>Fully-Connected | HEP[1]<br>EWA[2]<br>Cos[3] | Matrix | Sparse 2D Matrix<br>Dense 2D Matrix<br>3D Matrix | CUDA<br>MKL |
| Image Recognition | ResNet | HEP<br>Cos | HEP Dataset<br>Cos Dataset | Sparse 2D matrix<br>3D matrix | TensorFlow<br>Pytorch |
| Object Detection | Faster-RCNN | EWA | EWA Dataset | Dense 2D Matrix | TensorFlow<br>Pytorch |
| Image Generation | DCGAN | Cos | Cos Dataset | 3D Matrix | TensorFlow<br>Pytorch |

[1] High Energy Physics
[2] Extreme Weather Analysis
[3] Cosmology

### 3.4 Metrics

**Metrics for Component Benchmarks** At present, time-to-accuracy is the most well-received solution [8, 29]. For comprehensive evaluate, the training accuracy and validation accuracy are both provided. The former is used to measure the training effect of the model, and the latter is used to measure the generalization ability of the model. The threshold of target accuracy is defined as a value according to the requirement of corresponding application domains. Each application domain needs to define its own target accuracy. In addition, cost-to-accuracy and power-to-accuracy are provided to measure the money and power spending of training the model to the target accuracy.

**Metrics for Micro Benchmarks** The metrics of the micro benchmarks is simple since we only measure the performance without considering accuracy. we adopt FLOPS and images per second(images/s) as two main metrics. We also consider power and cost related metrics.

## 4 Reference Implementation

### 4.1 Component Benchmarks

According to the survey [60] of NERSC (National Energy Research Scientific Computing Center, the most representative DL framework is TensorFlow, and the proportion of which is increasing year by year. Consequently, we adopt TensorFlow for preferred framework.

In order to evaluate large-scale HPC systems running scientific DL, scalability is the fundamental requirement. In modern distributed DL, synchronized training through data parallelism is the mainstream. In this training scheme, each training process gets a different portion of the full dataset but has a complete copy of the neural network model. At the end of each batch computation, all processes will synchronize the model parameters by *all_reduce* operation to ensure they are training a consistent model. TensorFlow implements *all_reduce* through a parameter server [32] and use the GRPC protocol for communication by default. The master-slave architecture and socket-based communication can not extend to large-scale clusters [56]. Horovod [57] iirrespective a library originally designed for scalable distributed deep learning using TensorFlow. It implements *all_reduce* operation using ring-based algorithm [58] and MPI (Message Passing Interface) for communication. Due to the decentralized design and high effective protocol, the combination of TensorFlow and Horovod has successfully scaled to 27360 GPUs on Summit [21]. Therefore, we leverage Horovod to improve the scalability.

### 4.2 Micro Benchmarks

The goal of micro benchmarks is to determine the upper bound performance of the system. To do so, we implement it with succinct software stack. Every DL operator is written in C++ or call the low-level neural networks library (e.g. CuDNN) without any other dependencies.

## 5 Conclusion

In this paper, we propose HPC AI500—a benchmark suite for evaluating HPC system that running scientific deep learning workloads. Our benchmarks model real-world scientific deep learning applications, including extreme weather analysis, high energy physics, and cosmology. We propose a set of metrics for comprehensively evaluating the HPC AI systems, considering both accuracy, performance as well as power and cost. We provide a scalable reference implementation of HPC AI500. The specification and source code of HPC AI500 are publicly available from `http://www.benchcouncil.org/HPCAI500/index.html`.

## Acknowledgment

This work is supported by the Standardization Research Project of Chinese Academy of Sciences No.BZ201800001.

## References

1. Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems. 2012.
2. http://www.image-net.org/
3. Abadi, Martín, et al. "Tensorflow: a system for large-scale machine learning." OSDI. Vol. 16. 2016.
4. Jia, Yangqing, et al. "Caffe: Convolutional architecture for fast feature embedding." Proceedings of the 22nd ACM international conference on Multimedia. ACM, 2014.
5. Chen, Yunji, et al. "DianNao family: energy-efficient hardware accelerators for machine learning." Communications of the ACM 59.11 (2016): 105-112.
6. Jouppi, Norman P., et al. "In-datacenter performance analysis of a tensor processing unit." Computer Architecture (ISCA), 2017 ACM/IEEE 44th Annual International Symposium on. IEEE, 2017.
7. Adolf, Robert, et al. "Fathom: Reference workloads for modern deep learning methods." Workload Characterization (IISWC), 2016 IEEE International Symposium on. IEEE, 2016.
8. Coleman, Cody, et al. "DAWNBench: An End-to-End Deep Learning Benchmark and Competition." Training 100.101 (2017): 102.
9. Zhu, Hongyu, et al. "TBD: Benchmarking and Analyzing Deep Neural Network Training." arXiv preprint arXiv:1803.06905 (2018).
10. Shi, Shaohuai, et al. "Benchmarking state-of-the-art deep learning software tools." Cloud Computing and Big Data (CCBD), 2016 7th International Conference on. IEEE, 2016.
11. Hennessy, John L., and David A. Patterson. Computer architecture: a quantitative approach. Elsevier, 2011.
12. Wang, Lei, et al. "Bigdatabench: A big data benchmark suite from internet services." 2014 IEEE 20th International Symposium on High Performance Computer Architecture (HPCA). IEEE, 2014.
13. Jia Z, Wang L, Zhan J, et al. Characterizing data analysis workloads in data centers[C]//2013 IEEE International Symposium on Workload Characterization (IISWC). IEEE, 2013: 66-76.
14. Hao T, Huang Y, Wen X, et al. "Edge AIBench: Towards comprehensive end-to-end edge computing benchmarking." 2018 BenchCouncil International Symposium on Benchmarking, Measuring and Optimizing (Bench18), 2018.
15. Luo C, Zhang F, Huang C, Xiong X, J. Chen, et al. "AIoT Bench: Towards comprehensive benchmarking mobile and embedded device intelligence." 2018 BenchCouncil International Symposium on Benchmarking, Measuring and Optimizing (Bench18), 2018.
16. Gao W, Tang F, Wang L, Zhan J, et al. "AIBench: An Industry Standard Internet Service AI Benchmark Suite." Technical Report 2019.
17. Gao W, Luo C, Wang L, Xiong X, et al. "AIBench: Towards Scalable and Comprehensive Datacenter AI Benchmarking." 2018 BenchCouncil International Symposium on Benchmarking, Measuring and Optimizing (Bench18), 2018.
18. J. Dean. Keynote: Large Scale Deep Learning.
19. Collobert, Ronan, Samy Bengio, and Johnny Mariéthoz. Torch: a modular machine learning software library. No. EPFL-REPORT-82802. Idiap, 2002.

20. He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
21. Kurth T, Treichler S, Romero J, et al. Exascale deep learning for climate analytics[C]//Proceedings of the International Conference for High Performance Computing, Networking, Storage, and Analysis. IEEE Press, 2018: 51.
22. Kurth T, Zhang J, Satish N, et al. Deep learning at 15pf: supervised and semi-supervised classification for scientific data[C]//Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. ACM, 2017: 7.
23. Mathuriya A, Bard D, Mendygral P, et al. CosmoFlow: using deep learning to learn the universe at scale[C]//Proceedings of the International Conference for High Performance Computing, Networking, Storage, and Analysis. IEEE Press, 2018: 65.
24. https://www.oreilly.com/ideas/a-look-at-deep-learning-for-science
25. Bhimji W, Farrell S A, Kurth T, et al. Deep Neural Networks for Physics Analysis on low-level whole-detector data at the LHC[C]//Journal of Physics: Conference Series. IOP Publishing, 2018, 1085(4): 042034.
26. Ravanbakhsh S, Oliva J B, Fromenteau S, et al. Estimating Cosmological Parameters from the Dark Matter Distribution[C]//ICML. 2016: 2407-2416.
27. He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
28. Chen T, Chen Y, Duranton M, et al. BenchNN: On the broad potential application scope of hardware neural network accelerators[C]//2012 IEEE International Symposium on Workload Characterization (IISWC). IEEE, 2012: 36-45.
29. https://mlperf.org/
30. Ben-Nun T, Besta M, Huber S, et al. A Modular Benchmarking Infrastructure for High-Performance and Reproducible Deep Learning[J]. arXiv preprint arXiv:1901.10183, 2019.
31. Patton R M, Johnston J T, Young S R, et al. 167-PFlops deep learning for electron microscopy: from learning physics to atomic manipulation[C]//Proceedings of the International Conference for High Performance Computing, Networking, Storage, and Analysis. IEEE Press, 2018: 50.
32. Li M, Andersen D G, Park J W, et al. Scaling distributed machine learning with the parameter server[C]//11th USENIX Symposium on Operating Systems Design and Implementation (OSDI 14). 2014: 583-598.
33. Ravanbakhsh S, Lanusse F, Mandelbaum R, et al. Enabling dark energy with deep generative models of galaxy images[C]//Thirty-First AAAI Conference on Artificial Intelligence. 2017.
34. Mustafa M, Bard D, Bhimji W, et al. Creating virtual universes using generative adversarial networks[J]. arXiv preprint arXiv:1706.02390, 2017.
35. Schmelzle J, Lucchi A, Kacprzak T, et al. Cosmological model discrimination with Deep Learning[J]. arXiv preprint arXiv:1707.05167, 2017.
36. Peterson C. Track finding with neural networks[J]. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 1989, 279(3): 537-545.
37. Denby B. Neural networks and cellular automata in experimental high energy physics[J]. Computer Physics Communications, 1988, 49(3): 429-448.
38. de Oliveira L, Kagan M, Mackey L, et al. Jet-images-deep learning edition[J]. Journal of High Energy Physics, 2016, 2016(7): 69.
39. Komiske P T, Metodiev E M, Schwartz M D. Deep learning in color: towards automated quark/gluon jet discrimination[J]. Journal of High Energy Physics, 2017, 2017(1): 110.
40. Liu Y, Racah E, Correa J, et al. Application of deep convolutional neural networks for detecting extreme weather in climate datasets[J]. arXiv preprint arXiv:1605.01156, 2016.
41. Hong S, Kim S, Joh M, et al. Globenet: Convolutional neural networks for typhoon eye tracking from remote sensing imagery[J]. arXiv preprint arXiv:1708.03417, 2017.

42. Racah E, Beckham C, Maharaj T, et al. ExtremeWeather: A large-scale climate dataset for semi-supervised detection, localization, and understanding of extreme weather events[C]//Advances in Neural Information Processing Systems. 2017: 3402-3413.
43. Gómez-Bombarelli R, Wei J N, Duvenaud D, et al. Automatic chemical design using a data-driven continuous representation of molecules[J]. ACS central science, 2018, 4(2): 268-276.
44. https://www.ecowatch.com/un-extreme-weather-climate-change-2633131018.html
45. https://www.cbsnews.com/news/extreme-weather-events-2018-top-3-most-expensive-climate-driven-events-took-place-in-us/
46. https://extremeweatherdataset.github.io/
47. http://stanford.edu/group/stanford_atlas/
48. Spira M, Djouadi A, Graudenz D, et al. Higgs boson production at the LHC[J]. Nuclear Physics B, 1995, 453(1-2): 17-82.
49. https://en.wikipedia.org/wiki/Cosmology
50. Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets[C]//Advances in neural information processing systems. 2014: 2672-2680.
51. Bhimji W, Farrell S A, Kurth T, et al. Deep Neural Networks for Physics Analysis on low-level whole-detector data at the LHC[C]//Journal of Physics: Conference Series. IOP Publishing, 2018, 1085(4): 042034.
52. Sjöstrand T, Mrenna S, Skands P. PYTHIA 6.4 physics and manual[J]. Journal of High Energy Physics, 2006, 2006(05): 026.
53. https://www-n.oca.eu/ohahn/MUSIC/
54. https://bitbucket.org/tassev/pycola/
55. https://en.wikipedia.org/wiki/Convolution
56. Mathuriya A, Kurth T, Rane V, et al. Scaling grpc tensorflow on 512 nodes of cori supercomputer[J]. arXiv preprint arXiv:1712.09388, 2017.
57. Sergeev A, Del Balso M. Horovod: fast and easy distributed deep learning in TensorFlow[J]. arXiv preprint arXiv:1802.05799, 2018.
58. Andrew Gibiansky. Bringing HPC techniques to deep learning. http://research.baidu.com/bringing-hpc-techniques-deep-learning, 2017. [Online; accessed 6-December2017].
59. https://www.open-mpi.org/
60. https://www.jlab.org/indico/event/247/session/8/contribution/30/material/slides/0.pdf
61. Ren, Shaoqing, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." Advances in neural information processing systems. 2015.
62. Girshick, Ross, et al. "Rich feature hierarchies for accurate object detection and semantic segmentation." Proceedings of the IEEE conference on computer vision and pattern recognition. 2014.
63. Girshick, Ross. "Fast r-cnn." Proceedings of the IEEE international conference on computer vision. 2015.
64. Radford, Alec, Luke Metz, and Soumith Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks." arXiv preprint arXiv:1511.06434 (2015).
65. Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial nets." In Advances in neural information processing systems, pp. 2672-2680. 2014.