Artificial Intelligence for Scientific Discovery at High-Performance Computing Scales

Kin Long Kelvin Lee and Nalini Kumar, Intel Corporation

The proliferation of artificial intelligence (AI) and machine learning techniques provides natural applications for solving scientific problems across several domains. In this article, we provide a selective overview of recent scientific AI workloads at the largest computing scales.

Digital Object Identifier 10.1109/MC.2023.3241692
Date of current version: 5 April 2023

Scientific discovery has been a major driving force for advances in computation and calculation: every domain and every application comprises its own unique set of data and compute characteristics, and the amount of global resources that have been dedicated to development, maintenance, and execution are likely nigh unquantifiable. As of late, however, artificial intelligence (AI) and machine learning (ML) have become an attractive means to unify a significant portion of workloads under the same modeling framework by using neural networks as universal function approximators.[1]

As AI and ML began to make significant advances in algorithms (in particular, fast automatic differentiation, which forms the backbone of all AI/ML training), expressiveness and capabilities, and new hardware/platforms, AI/ML modeling started permeating throughout scientific applications. Today,

0018-9162/23©2023IEEE

EDITORS
NORITA AHMAD American University of Sharjah; nahmad@aus.edu
PREETI CHAUHAN IEEE Reliability Society; preeti.chauhan@ieee.org

it is common for most large technology organizations to devote significant resources toward "AI for Science," research into how AI can be used to accelerate scientific discovery, and to tackle some of the biggest open challenges for humanity.

For many of these applications, AI/ML models developed for industry problems directly map onto scientific ones: the transformer for NLP, for instance, sees use in protein folding and weather forecasting. Although this benefits modeling capabilities, one distinguishing trait and major bottleneck in scientific AI is the sparsity of available data: experimental data are often difficult and expensive to acquire, and simulation with a modest computing budget can only afford so many data points and only on specific systems of interest. In contrast, AI applications in industry have seen significant investment in data collection since before the advent of big data; language models can be trained with text scraped from the Internet, while the same is impossible for genomics or cosmology. There are recent concerted efforts to overcome this data availability and coverage challenge. The workloads we review in this article are unique not solely because they have high-performance computing (HPC)-scale requirements, but also because they represent concerted efforts to curate, generate, and standardize large amounts of scientific data and novel neural network architectures, and open sourced for communities to build on. It is our opinion that these grand efforts will hopefully make collective progress toward common goals in areas such as novel materials discovery, treatment of disease, and weather forecasting.

AI/ML AT HPC SCALES

Given that "HPC-scale AI/ML" is a highly nebulous and transient description, for the context of this article, we classify workloads as such based loosely (note that not all criteria need to be fulfilled) on four computational requirements, namely,

- dataset size in the realm of hundreds of gigabytes to terabytes; "HPC-scale" datasets necessitate a high training throughput obtained through distributed data parallelism and/or accelerator offloading.
- the number of learnable parameters, on the order of billions; "HPC-scale" neural network models involve intensive compute and memory requirements, both in terms of capacity and bandwidth.
- extremely low latency; applications that require neural network computations to finish on the order of microseconds or even lower.
- an AI/ML model incorporated inside a conventional HPC workload/simulation.

Whereas the other categories are more obvious in their HPC requirements, dataset size warrants some discussion. Conventional or "classical" AI datasets like ImageNet[2] (~150 GB) we would classify as being on the outskirts of our definition, although a few years ago it likely would have fit comfortably given its size. Now, when paired with models such as ResNet50, workstations with consumer-grade hardware are able to train to convergence within a day or two; thus, training on a distributed scale is not strictly necessary to obtain a result within a reasonable time frame even if it can naturally benefit from doing so.[3] Although not strictly a scientific nor holdfast definition, an intuitive and practical one would see HPC-scale AI workloads being those that would take a few hours to close to a day to complete even a single epoch without data parallelism, assuming of course vectorized code running on recent accelerators and associated deep learning framework and libraries.

> "HPC-scale" neural network models involve intensive compute and memory requirements, both in terms of capacity and bandwidth.

DOMAINS

In the next sections, we review a sampling of current scientific AI/ML workloads that operate at HPC scale, per our narrow definition discussed in the preceding section. Although we do not provide an exhaustive set of workloads, we believe that they are representative of the current state of the art, sufficient to see how and where AI can be applied to scientific problems at scale.

Chemistry

In applied chemistry, AI/ML surrogates are now routinely used for high throughput screening of large databases of molecules and materials for desirable properties such as toxicity for drug discovery and bandgaps for semiconductors. Traditionally, these properties were either determined experimentally, leading to exorbitant research costs, or calculated and/or simulated, although the computational resources required for this restricted the breadth and complexity of systems that were able to be considered. In these screening applications, surrogate models learn to parameterize functions that map molecular/material features onto

target properties using existing experimental and theoretical data, at a mere fraction of the computational cost.

Because the space of potential molecules and systems is extremely large and the enumeration of even small molecules is a daunting task,[4] HPC-scale datasets are necessary to obtain AI/ML models that can generalize to unknown systems reliably. An excellent example of this is the Open Catalyst Project,[5] a collaborative effort by Meta (formerly known as *Facebook*) AI research and Carnegie Mellon University that resulted in an HPC-scale dataset and open source repository for catalytic material discovery, which utilizes graph neural networks as surrogate models for gas-surface interactions and structure optimization. In its latest incarnation, referred to as *OC22*,[6] the dataset comprises approximately 2 TB of data pertaining to different ML tasks, catalytic materials, adsorbates, and configurations. The extremely wide range of systems, particularly in diversity of rare-earth atoms typically employed in batteries and renewable energy applications, presents a highly promising platform for discovering new materials. Similarly, the complexity of the data has required as well as spawned a suite of novel neural network architectures with hundreds of thousands of parameters up to the billion scale.[7]

Another avenue for AI in chemistry that has seen a significant amount of interest is representation learning. Conveniently, molecules can be compactly represented as linear strings with only some loss of information (see Krenn et al.[8] and the references therein). In contrast to OC20/22, molecular strings can be significantly easier to generate and obtain and are accessible in a number of sources such as PubChem[9] and GDB,[10] which enumerate hundreds of billions of molecules, making them a part of the frontier for scale in scientific data. The other advantage of string-based representations is their conversion of modeling problems into natural language tasks, thereby taking advantage of the recent advancements made with large language models such as Generative Pre-trained Transformers (GPT). Notably, ChemGPT by Frey et al.[11] repurposes the open source GPT-NeoX[12] to operate on two common molecular string formats, elegantly demonstrating that the model- and dataset- scaling laws seen in conventional NLP[13] hold true for chemical representations. The rich embeddings obtained from these large language models are then typically used for downstream tasks, capturing enough information for even simple models to reproduce complex properties, for example, drug discovery.[14]

Another avenue for AI in chemistry that has seen a significant amount of interest is representation learning.

Biology

Similar to our discussion in the previous section, biology has seen significant advancements through the application of large language models for drug discovery and protein interactions. The latter have received significant public attention from numerous organizations ranging from DeepMind/AlphaFold[15] and Meta AI/ESMFold[16] as well as broad community efforts like OpenFold,[17] all of which use large language models for protein-sequence modeling and inference. Notably, ESMFold does not require multiple sequence alignment, a key differentiation from how protein-sequence modeling has been done in the past, and provides a prime example as to how AI can fundamentally change how scientific problems can be approached, and not just simply surrogate models.

In regard to medical applications, vision models and image datasets have been a notably intuitive application of deep learning. Although there are countless individual research papers, notably those captured in the grand challenges, on applying various neural network architectures to imaging data, there have only recently been efforts to try and consolidate medical AI applications in a data-centric way. DeepLeison[18] is an example of a concerted effort to curate a large-scale dataset for radiology imaging, comprising 33,688 labeled images (~220 GB compressed) for multiclass leison segmentation. More recently, AMOS[19] represents a more data-centric[20] effort toward organ segmentation; in the same way OpenCatalyst provides multiple tasks, AMOS provides labeled and unlabeled data for (semi)-supervised learning in different contexts and problems. This shift in attitudes surrounding data and modeling enables researchers to devise workflows that actively contribute toward general solutions of problems, as opposed to an emphasis on models that perform well on datasets with an extremely narrow scope. For this reason, the CANcer Distributed Learning Environment (CANDLE) project[21] demonstrates another holistic approach to multiple areas of cancer research and precision medicine, providing an open source pipeline for large-scale distributed model training and hyperparameter optimization.

Unlike simulated materials data, however, medical images are considerably more difficult to collect, calibrate, and label due to privacy and availability constraints. One way of alleviating this issue is with federated learning, whereby data storage is decentralized, and the only information transferred among workers in this massively distributed setting is model parameters. As a recent and perhaps largest example of federated learning for biological applications, Pati et al.[22] demonstrated the need for highly generalizable AI models applied to large, diverse datasets. In this case, an aggregate, "consensus model"

pooled training results for tumor detection from 71 sites across six continents, generating a model that is robust to different data distributions.

Physics and Astronomy

Out of the domains covered in this article, physics and astronomy have perhaps the highest dynamic range of AI computational requirements: ranging from extremely low latency applications to processing large historical datasets over time. For this reason, physics and astronomy HPC AI applications see specialized developments in model and hardware development not typically seen in other areas, such as application-specified integrated circuits and field-programmable gate arrays (FPGAs). Here we chose to highlight some more extreme cases to emphasize their novelty.

In the case of particle physics, there has been a diverse range of domain-specific developments, particularly due to the high data volumes borne from operation of the Large Hadron Collider, which operates at 40 MHz, or latency on the order of microseconds. These extremely low latencies have required consideration and development of highly specialized accelerators, in particular FPGAs, not just for the initial data pipeline but for AI/ML inference for real-time processing.[23] In these cases, neural network models are trained offline in the usual way, and subsequently compressed and adapted for computations on the FPGA board. The typical challenges associated with these ventures are sacrificing abstraction for performance, highly customized and tuned kernels are required for function, and the pipelining process may not have a straightforward mapping from its CPU/GPU counterpart. Progress has been made, for example, by Khoda et al.,[24] to port architectures like recurrent neural networks that are well suited for sequences/time series to FPGAs, and demonstrate latencies spanning tens to several hundred microseconds, depending on the size of the model. Increasingly, these hardware developments make AI processing a viable option in real-time particle data collection and processing. In the related field of fusion energy research, the authors from DeepMind[25] showed incredible success in applying deep reinforcement learning for real-time reactor control; however, they noted that their modeling choice was constrained specifically to fit within the L2 cache of the CPU control system to ensure low latencies. Given that some of these constraints would be lifted with FPGAs, there is a natural synergy between these two developments. This will be a highly anticipated and exciting area to see.

On the other end of the spectrum, astronomical observations at various wavelengths have produced extremely large troves of well-curated and calibrated data that could be analyzed in "time-domain" astronomy.[26] At the same time, the advent of next-generation telescopes such as the square-kilometer array (SKA) is expected to accelerate data generation, with the SKA projected to produce approximately a terabyte per second, or an estimated 8.5 EB of data over a 15-year life span.[27] For these reasons, astronomy has seen a steady increase in AI use for drawing inferences from large archives of data built on top of a long historical partnership.[28] Object detection is perhaps one of the most intuitive applications, and in the past has relied on crowd-sourced, citizen-science efforts like galaxy zoo, which has produced hundreds of thousands of labeled images. These images can subsequently be used to train computer vision models for object detection, classification, and segmentation[29] that can be used as downstream data products.

In addition to galaxies, high-throughput AI models can also be applied toward analyzing and extracting physical insight from "rare" events, for example, in the work by Zhang et al.,[30] in applying neural density estimation to studying gravitational microlensing. Another avenue is the generation of synthetic images, which could be used to produce data for rare events in sufficient quantities to train and test new algorithms. Smith et al.,[31] for instance, applied diffusion image modeling to generate realistic galaxy images while preserving physical quantities such as galaxy size and flux, thereby showcasing a direction that would strongly benefit areas of astronomy that have traditionally been data sparse.

Weather and Climate

Naturally, weather and climate have traditionally been HPC-heavy areas, with extreme computational requirements given the large spatial and temporal grids involved. For the same reason, AI/ML surrogates are an attractive option to attempt to minimize time to solution in addition to introducing modeling flexibility and uncertainty quantification. Furthermore, the years of historical data, combined with well-optimized weather/climate modeling codes, has resulted in large datasets for training and evaluating AI/ML surrogates.

An early and recent approach toward standardizing training data for weather/climate applications, in particular radiation transfer, was developed by the Rolnick Group at the University of Toronto and coined ClimART[32] based on the Canadian Earth System model. The authors' efforts include best practices for data and model curation: for the former, organization into "tasks" or scenarios similar to OC20/22 to provide benchmark targets, and a number of baseline model implementations for the latter. This way, ClimART facilitates open source development of fast and accurate radiative transfer surrogates, which can potentially be used to substitute modules within weather codes.

An even more recent approach was presented by researchers at DeepMind and Google, who developed a framework termed *GraphCast*,[33] which combines an autoregressive graph neural network approach with 39 years of historical weather data,[34] resulting in a new state of the art in medium-range weather forecasting. Of particular interest to the HPC community is the

computational requirements of scientific AI at such a scale: the authors note that each time step comprises almost a gigabyte of single-precision data, even before considering the memory requirements for model weights and backpropagation. Society has much to gain in the face of such a computationally demanding workload, however, given the ability to generate accurate 10-day forecasts in under a minute with a single tensor processing unit. Similar approaches to GraphCast like FourCastNet[35] have considered model ensembling to obtain uncertainty estimates, which would be instrumental toward adopting AI in real-world weather forecasting.

FUTURE OUTLOOK

Although we have chosen to focus on HPC-scale scientific AI for this article, this niche is by no means the only path taken. Broadly speaking, industry applications in computer vision and NLP have adopted extremely large models and datasets, taking advantage of the un/self-supervised nature of deep neural networks that scale with both dimensions; in turn, they present a special, interesting case for HPC. The opposite end of the spectrum, however, deals with data scarcity and highly parameter-efficient models, those that capture physical biases well, such as symmetry in nature[36] and those mimicking mathematical operators. Although they may not be parameter intensive, they may demonstrate complex, compute-intensive use patterns that require highly specialized tuning and accelerators. Graph neural networks are a recent example of this; naive implementations of graph structures are extremely prone to cache misses, and a significant amount of effort has been devoted to proposing algorithms and architectures[38] that improve graph processing.

Another area that is currently lacking in HPC-scale scientific AI is interpretability and explainability. Thus far, the surrogates highlighted in this article are "black-box" regressors, and although they are effective at doing so, they do not offer insight into why a prediction was obtained. Although some early interpretive techniques could be used with the trained model, such as input gradients, there are well-known shortcomings to such posthoc analyses,[39] and there is a strong case to be made for model architectures with built-in uncertainty quantification and explainability. In particular, frameworks that include causal learning (see Kaddour et al.[40] for a comprehensive and recent review on the topic) where simple modifications to the architecture and training approach can provide robust and interpretable predictions.[41] Uncertainty quantification, on the other hand, is critical for decision making: the authors of GraphCast, too, note that prediction ensembling is missing from their approach, which would be needed to describe how different initial conditions map onto final weather states, a ubiquitous problem for modeling/simulating physical systems. In this light, although scientific AI has certainly reached new scales that enable new scientific discoveries, there are still improvements to be made to unlock its full potential.

Finally, and perhaps to end on a positive, forward-looking note, we discuss how the HPC-scale scientific AI landscape may evolve thanks to algorithmic and hardware improvements: models and data that cannot be considered today due to data, compute, and memory constraints. For these, we rely on fundamental shifts in AI/ML abstraction and frameworks. We give two examples of these efforts: generative modeling and new approaches to model training.

In the former, a mere two or three years ago, generative models remained a research niche comprising generative adversarial networks,[42] neural flows,[43] and diffusion models,[44] among other architectures. In the last year, stable diffusion and DALL-E have propelled generative models into the public zeitgeist; although there were existing efforts to use generative models to generate scientific data (for a recent example, see Hoogeboom et al.[45]), they have yet to become widespread in the scientific AI tool kit despite being a viable and scalable way to mitigate sparsity in scientific datasets. By generating useful data with these approaches, we are hopeful that new scientific discoveries and inferences can be drawn from models trained on HPC-scale datasets in more domains.

In the case of model training, backpropagation has been at the core of deep learning, enabling extremely large models to be trained. Typically, backpropagation records all operations in the forward pass of the data to facilitate automatic differentiation, leading to high memory costs. Advancements in this field circumvent these requirements by avoiding backpropagation altogether, usually by means of approximate gradients such as "forward gradients"[46] and the "forward-forward" algorithm.[47] By alleviating memory constraints, AI training will require less memory movement, and thus enable even larger and complex models, perhaps those needed to solve some of the biggest open questions in science and humanity.

REFERENCES

1. K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, Jan. 1989, doi: 10.1016/0893-6080(89)90020-8.
2. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255, doi: 10.1109/CVPR.2009.5206848.
3. P. Goyal et al., "Accurate, large minibatch SGD: Training ImageNet in 1 hour," Apr. 2018, *arXiv:1706.02677*.
4. J.-L. Reymond, "The chemical space project," *Accounts Chem. Res.*, vol. 48, no. 3, pp. 722–730, Mar. 2015, doi: 10.1021/ar500432k.

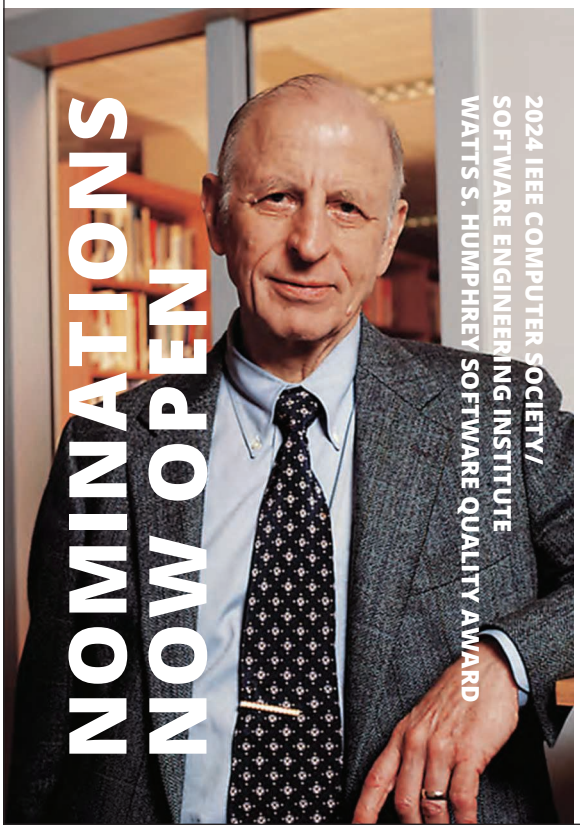
5. L. Chanussot et al., "The Open Catalyst 2020 (OC20) dataset and community challenges," *ACS Catalysis*, vol. 11, no. 10, pp. 6059–6072, May 2021, doi: 10.1021/acscatal.0c04525.
6. R. Tran et al., "The Open Catalyst 2022 (OC22) dataset and challenges for oxide electrocatalysts," Nov. 2022, *arXiv:2206. 08917*.
7. A. Sriram, A. Das, B. M. Wood, S. Goyal, and C. L. Zitnick, "Towards training billion parameter graph neural networks for atomic simulations," Mar. 2022, *arXiv:2203.09697*.
8. M. Krenn et al., "SELFIES and the future of molecular string representations," Mar. 2022, *arXiv:2204.00056*.
9. S. Kim et al., "PubChem 2023 update," *Nucleic Acids Res.*, vol. 51, no. D1, pp. D1373–D1380, Jan. 2023, doi: 10.1093/nar/gkac956.
10. T. Fink, L. C. Blum, L. Ruddigkeit, R. Van Deursen, and J.-L. Reymond, Aug. 2021, "GDB Databases," Zenodo, doi: 10.5281/ZENODO.5172018.
11. N. Frey et al., "Neural scaling of deep chemical models," *ChemRxiv*, May 2022, doi: 10.26434/chemrxiv2022-3s512.
12. S. Black et al., "GPT-NeoX-20B: An open-source autoregressive language model," Apr. 2022, *arXiv:2204.06745*.
13. J. Kaplan et al., "Scaling laws for neural language models," Jan. 2020, *arXiv:2001.08361*.
14. H. Öztürk, A. Ozgur, P. Schwaller, T. Laino, and E. Ozkirimli, "Exploring chemical space using natural language processing methodologies for drug discovery," *Drug Discovery Today*, vol. 25, no. 4, pp. 689–705, Apr. 2020, doi: 10.1016/j.drudis.2020.01.020.
15. J. Jumper et al., "Highly accurate protein structure prediction with AlphaFold," *Nature*, vol. 596, no. 7873, pp. 583–589, Aug. 2021, doi: 10.1038/s41586-021-03819-2.
16. Z. Lin et al., "Evolutionary scale prediction of atomic level protein structure with a language model," *bioRxiv*, Dec. 2022, doi: 10.1101/2022.07.20.500902.
17. G. Ahdritz et al., "OpenFold: Retraining AlphaFold2 yields new insights into its learning mechanisms and capacity for generalization," *bioRxiv*, Nov. 2022, doi: 10.1101/2022.11.20.517210.
18. K. Yan, X. Wang, L. Lu, and R. M. Summers, "DeepLesion: Automated deep mining, categorization and detection of significant radiology image findings using large-scale clinical lesion annotations," Oct. 2017, *arXiv:1710.01766*.
19. Y. Ji et al., "AMOS: A largescale abdominal multi-organ benchmark for versatile medical image segmentation," Sep. 2022, *arXiv:2206.08023*.
20. M. Mazumder et al., "DataPerf: Benchmarks for data-centric AI development," Jul. 2022, *arXiv:2207.10062*.
21. J. M. Wozniak et al., "CANDLE/Supervisor: A workflow framework for machine learning applied to cancer research," *BMC Bioinf.*, vol. 19, no. S18, p. 491, Dec. 2018, doi: 10.1186/s12859-018-2508-4.
22. S. Pati et al., "Federated learning enables big data for rare cancer boundary detection," *Nature Commun.*, vol. 13, no. 1, p. 7346, Dec. 2022, doi: 10.1038/s41467-022-33407-5.
23. J. Duarte et al., "Fast inference of deep neural networks in FPGAs for particle physics," *J. Instrum.*, vol. 13, no. 7, Jul. 2018, Art. no. P07027, doi: 10.1088/1748-0221/13/07/P07027.
24. E. E. Khoda et al., "Ultra-low latency recurrent neural network inference on FPGAs for physics applications with hls4ml," Jul. 2022, *arXiv:2207.00559*.
25. J. Degrave et al., "Magnetic control of tokamak plasmas through deep reinforcement learning," *Nature*, vol. 602, no. 7897, pp. 414–419, Feb. 2022, doi: 10.1038/s41586-021-04301-9.
26. M. Poudel, R. P. Sarode, Y. Watanobe, M. Mozgovoy, and S. Bhalla, "A survey of big data archives in time-domain astronomy," *Appl. Sci.*, vol. 12, no. 12, Jan. 2022, Art. no. 6202, doi: 10.3390/app12126202.
27. A. M. M. Scaife, "Big telescope, big data: Towards exascale with the square Kilometre array," *Philos. Trans. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 378, no. 2166, Jan. 2020, Art. no. 20190060, doi: 10.1098/rsta.2019.0060.
28. A. S. Miller, "A review of neural network applications in Astronomy," *Vistas Astron.*, vol. 36, pp. 141–161, Jan. 1993, doi: 10.1016/0083-6656(93)90118-4.
29. R. Hausen and B. E. Robertson, "Morpheus: A deep learning framework for the pixel-level analysis of astronomical image data," *Astrophys. J. Suppl. Series*, vol. 248, no. 1, p. 20, May 2020, doi: 10.3847/1538-4365/ab8868.
30. K. Zhang, B. S. Gaudi, and J. S. Bloom, "A ubiquitous unifying degeneracy in two-body microlensing systems," *Nature Astron.*, vol. 6, no. 7, pp. 782–787, Jul. 2022, doi: 10.1038/s41550-022-01671-6.
31. M. J. Smith, J. E. Geach, R. A. Jackson, N. Arora, C. Stone, and S. Courteau, "Realistic galaxy image simulation via score-based generative models," *Monthly Notices Roy. Astronomical Soc.*, vol. 511, no. 2, pp. 1808–1818, Feb. 2022, doi: 10.1093/mnras/stac130.
32. S. R. Cachay, V. Ramesh, J. N. S. Cole, H. Barker, and D. Rolnick, "ClimART: A benchmark dataset for emulating atmospheric radiative transfer in weather and climate models," Nov. 2021, *arXiv:2111.14671*.
33. R. Lam et al., "GraphCast: Learning skillful medium-range global weather forecasting," Dec. 2022, *arXiv:2212.12794*.
34. H. Hersbach et al., "The ERA5 global reanalysis," *Quart. J. Roy. Meteorolog. Soc.*, vol. 146, no. 730, pp. 1999–2049, Jul. 2020, doi: 10.1002/qj.3803.
35. J. Pathak et al., "FourCastNet: A global data-driven high-resolution weather model using adaptive Fourier neural operators," Feb. 2022, *arXiv:2202.11214*.
36. H. Stark, O.-E. Ganea, L. Pattanaik, R. Barzilay, and T. Jaakkola, "EquiBind: Geometric deep learning for drug

binding structure prediction," Jun. 2022, *arXiv:2202.05146*.
37. N. Kovachki et al., "Neural operator: Learning maps between function spaces," Oct. 2022, *arXiv:2108.08481*.
38. C.-Y. Gui et al., "A survey on graph processing accelerators: Challenges and opportunities," *J. Comput. Sci. Technol.*, vol. 34, no. 2, pp. 339–371, Mar. 2019, doi: 10.1007/s11390-019-1914-z.
39. H. Shah, P. Jain, and P. Netrapalli, "Do input gradients highlight discriminative features?" Oct. 2021, *arXiv:2102.12781*.
40. J. Kaddour, A. Lynch, Q. Liu, M. J. Kusner, and R. Silva, "Causal machine learning: A survey and open problems," Jul. 2022, *arXiv:2206.15475*.
41. S. Wang, S. Sankaran, and P. Perdikaris, "Respecting causality is all you need for training physics-informed neural networks," Mar. 2022, *arXiv:2203.07404*.
42. I. J. Goodfellow et al., "Generative adversarial networks," Jun. 2014, *arXiv:1406.2661*.
43. W. Grathwohl, R. T. Q. Chen, J. Bettencourt, I. Sutskever, and D. Duvenaud, "FFJORD: Freeform continuous dynamics for scalable reversible generative models," Oct. 2018, *arXiv:1810.01367*.
44. J. Sohl-Dickstein, E. A. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," Nov. 2015, *arXiv:1503.03585*.
45. E. Hoogeboom, V. G. Satorras, C. Vignac, and M. Welling, "Equivariant diffusion for molecule generation in 3D," Mar. 2022, *arXiv:2203.17003*.
46. A. G. Baydin, B. A. Pearlmutter, D. Syme, F. Wood, and P. Torr, "Gradients without backpropagation," Feb. 2022, *arXiv:2202.08587*.
47. G. Hinton, "The forward-forward algorithm: Some preliminary investigations," Dec. 2022, *arXiv:2212.13345*.

KIN LONG KELVIN LEE is a high-performance computing artificial intelligence/machine learning research engineer within the AXG group at Intel Corporation, Hillsboro, OR 97124 USA. Contact him at kin.long.kelvin.lee@intel.com.

NALINI KUMAR is an applications engineer focused on high-performance computing/artificial intelligence within the AXG group at Intel Corporation, Santa Clara, CA 95054 USA. Contact her at nalini.kumar@intel.com.

Carnegie Mellon University
Software Engineering Institute



Since 1994, the SEI and the Institute of Electrical and Electronics Engineers (IEEE) Computer Society have cosponsored the Watts S. Humphrey Software Quality Award, which recognizes outstanding achievements in improving an organization's ability to create and evolve high-quality software-dependent systems.

Humphrey Award nominees must have demonstrated an exceptional degree of **significant**, **measured**, **sustained**, and **shared** productivity improvement.

TO NOMINATE YOURSELF OR A COLLEAGUE, GO TO computer.org/volunteering/awards/humphrey-software-quality

Nominations due by September 1, 2023.

FOR MORE INFORMATION
resources.sei.cmu.edu/news-events/events/watts

Digital Object Identifier 10.1109/MC.2023.3254044