Benchmarking Deep Learning for Time Series: Challenges and Directions

Xinyuan Huang*, Geoffrey C. Fox†, Sergey Serebryakov‡, Ankur Mohan§, Pawel Morkisz¶‖, Debojyoti Dutta*
*Cisco Systems, San Jose, California, USA {xinyuahu,dedutta}@cisco.com
†Indiana University, Bloomington, Indiana, USA gcf@indiana.edu
‡Hewlett Packard Enterprise, San Jose, California, USA sergey.serebryakov@hpe.com
§In-Q-Tel, Arlington, Virginia, USA amohan@iqt.org
¶Nvidia, Warsaw, Poland pmorkisz@nvidia.com
‖AGH University of Science and Technology, Krakow, Poland

***Abstract*—Deep learning for time series is an emerging area with close ties to industry, yet under represented in performance benchmarks for machine learning systems. In this paper, we present a landscape of deep learning applications applied to time series, and discuss the challenges and directions towards building a robust performance benchmark of deep learning workloads for time series data.**

***Index Terms*—machine learning, deep learning, time series, performance, benchmark**

I. Introduction

With the rapid growth of hardware and software (HW/SW) innovation in machine learning (ML), there is a need for representative benchmarks to enable fair and reproducible performance benchmarks that can accelerate the development of new algorithms and ML systems. Deep learning (DL) is a popular branch of ML that poses unique challenges to ML systems due to its high demand and complex workload in a combination of computation, network and storage. In the recent years, a variety of works in DL benchmark have evolved [1]–[4], among which MLPerf [4] is becoming a leading standard with vast adoption in both industry and academia. Designed as an end-to-end system performance benchmark suite, MLPerf and its predecessors provide a select set of reference workloads that aims to represent real-world ML/DL use cases covering image classification, object detection, translation, speech recognition, recommendation and reinforcement learning. While those are undoubtedly of high commercial values, lots of blanks are yet to be filled in many industry focusing application domains, where time series is one of the representing problems.

Time series analysis plays an important role in various industrial areas including manufacturing, transportation, security, health, finance, and scientific computing, etc. (Table I) While statistical and econometric models have been well studied for decades in classical tasks such like forecasting and anomaly detection [5], [6], DL based approaches have only recently demonstrated the high potential of success as they bring new solutions to complex time series problems [7], [8]. Currently, most of the work has been concentrated on the measurements of model accuracy. System-wide performance, however, plays an equally important role. ML/DL based applications for time series problems have various performance issues in both training and inference stages of its life cycle, which justifies the need for proper methodologies to evaluate performance in such systems.

Inspired by the state-of-the-art works in ML benchmarks, and driven by the needs for performance evaluation in time series systems, we look into the intersection between the two with a specific focus on DL based use cases. The goal of this paper is to establish a starting point for future research of DL performance benchmarks for time series. The paper is organized in the following way, Section II provides an overview of the variety of application domains where DL is applied to time series problems. Section III discusses about the challenges and propose potential directions in the design of performance benchmarks time series DL systems. We conclude in Section IV with some future directions.

II. Applications, Models and Datasets

We base our work on a study of application areas that have high impacts in industry and potential value for system performance evaluation, and that have not been well represented in existing benchmarks.

As Table I shows, DL has been applied to a diverse set of time series use cases. Not only is it competent in classical tasks such as forecasting [62], classification [7], and anomaly detection [63], it also proved to be capable of addressing new complex tasks that were hard for traditional algorithms such like spatiotemporal data mining [15], representation learning [25], [29], etc. It is interesting to notice that, although being relatively costly in computation, there have been exploratory works applying DL under a real-time, low-latency streaming context such like high-frequency trading [43], [45].

Model structures for time series diverge by use cases. While CNN and RNN/LSTM are popular choices for many applications, there are a variety of works employing hybrid or other models. Hybrid models combining traditional statistical algorithms with deep neural networks proved to be superior

TABLE I
LANDSCAPE OF DEEP LEARNING APPLIED TO TIME SERIES

Areas	Applications	Model types	Data sets	Papers
Transportation	Cars, Taxis, Freeway Detectors	RNN[1], GCN[2], BNN[3]	Caltrans highway traffic[9], Taxi/Uber trips[10]–[12]	[13]–[16]
Health	Medical sensors: EEG, ECG, ERP, Patient data analysis	CNN, RNN[1], Markov	OPPORTUNITY[17], [18], EEG[19]–[22], MIMIC[23]	[24]–[30]
Human activities	wearable devices, motion detection, gesture recognition	CNN, RNN[1]	CMU-MoCap[31], Soli[32]	[29], [33], [34]
Cybersecurity	Intrusion, traffic classification, anomaly detection	CNN, RNN[1]	IDS2018[35], SherLock[36]	[37]–[39]
Finance	high frequency trading, stock prices, cryptocurrency anomaly detection	CNN, RNN[1], GCN[2]	FI2010[40], Elliptic Dataset [41]	[42]–[45]
Industrial operation	Software operation, industry process control, anomaly	RNN[1]	Enterprise software system[46], GPL-loop[47]	[47], [48]
Science	Climate, Tokomak, Earthquake	RNN[1], Markov	USHCN[49], Earthquake[50], [51]	[30], [52]–[56]
General social statistics	Economic, Finance, Demographics, Industry, etc.	CNN, RNN[1], AR[4]	M4[57], electricity[58]	[59]–[61]

[1] includes variants e.g. LSTM etc. [2] graph convolutional network [3] Bayesian neural network [4] variants of auto-regressive models

than either approach alone in certain general forecasting areas [64]. Models combining CNN and RNN/LSTM are popular among problems with both spatial and temporal features [15]. Graph based model is an emerging new method often applied to problems with large scale graph representations (e.g. financial transactions [44], transportation networks [65], etc.).

Time series data being one of the most natural and widespread type of data is found in many domains. PhysioNet [20], [23] is a large collection of synthetic, clinical and waveform data from health domain. Transportation [9]–[12] and science [49]–[51] domains provide large, mostly spatio-temporal, datasets. Industrial datasets are considered highly valuable and sensitive data and are not generally available publicly. Existing datasets [35], [36], [47], [48] are datasets specifically collected/simulated for research purposes. Multiple approaches are used to construct datasets when they either do not exist or do not satisfy certain conditions such as quality, size etc. The first approach is to collect data in controlled or semi-controlled environments [17], [18], [31], [35], [36], [48]. The second approach is to model on a computer the process or processes that generate corresponding time series data [47].

III. CHALLENGES AND DIRECTIONS

DL for time series problems poses unique challenges to the design of performance benchmarks. We discuss four of the aspects that we consider most important.

A. Diverse application domains and models

Unlike other domains such as computer vision (CV) or natural language processing (NLP), where the majority of applications share common focuses on certain set of learning tasks and models, the field of time series has a broad range of applications with various types of learning tasks and models. In order to be representative of the field, the selection of reference workloads in a time series benchmark should be conducted with careful consideration of trade-offs between number of reference workloads and coverage of the following aspects: (1) use cases such as forecasting (e.g. traffic prediction), classification (activity recognition) and anomaly detection (intrusion detection) etc.; (2) properties of data such as single- and multi-variate time series, sequence length, number of time series and variability in sampling rates etc; (3) feature engineering including features from time and frequency domains; and (4) design choices for output variables, for instance, in case of forecasting it is common to use multiple horizons, e.g. forecast 5, 10 and 20 minutes forward etc.

B. Diverse systems and performance requirements

The training-inference life cycle of time series applications can involve multiple kinds of hardware systems ranging from low power embedded devices (e.g. IoT sensors, wearable devices, edge computing, etc.), compute intensive accelerators (e.g. GPU, TPU, FPGA, etc.) to cloud scale high performance clusters. Real-world time series systems may also involve special software stacks. For example, time series databases (TSDB) and extract-transform-load (ETL) stacks are often used in production systems for storage, querying and online transformation of continuously ingested time series samples. In cases of large scale distributed training and real-time, low-latency inference, the combination of these HW/SW stacks can leave spaces for performance optimizations that other ML benchmarks may not reveal.

C. Performance measurement and workload characterization

We consider two aspects of performance analysis for a ML benchmark. Measurement of end-to-end system performance is one aspect that enables fair and reproducible evaluations under real-world usage scenarios. The end-to-end performance can be measured using satisficing and optimizing metrics [66]. For example, the training benchmarks can set a target quality for a model and measure time-to-train as the performance metric [4], and the inference benchmarks can set a pre-determined QoS target and measure latency, number-of-stream, throughput, or throughput distribution depending on deployment scenario [67]. Workload characterization is an equally important aspect which helps understanding performance bottlenecks and drives optimizations. Prior works provide a diverse set of measurement for both training and inference workloads, which includes accelerator resource utilization [2], [68] and multi-layer profiling from kernel operations, ML frameworks to models and applications [1], [69].

We expect that the directions of existing performance analysis approaches can be adopted by new benchmarks for time series DL workloads. However, several specific characteristics need to be considered in detailed design. For example, in end-to-end evaluation, specific rules for target quality selection need to be determined so as to minimize the impact of randomness in model's convergence curves, where the target quality may involve multiple accuracy metrics constrained by different time windows. In workload characterization, the special designs of model architectures, such like hybrid models and graph-based models, may introduce new computation patterns that are less studied than those of well understood workloads in CV and NLP.

D. Open and standard datasets

Dataset is an essential part of the workloads in a ML benchmark. The success of ImageNet [70] and the likes have proved that well adopted, public accessible datasets can strongly impact the progress of a ML field. We notice that, in spite of the wide existence of public datasets in the time series domain, often the scale and dimension of data in real production can be orders of magnitude larger than those available to the public (e.g. [13], [60]). A public, well designed dataset representing various real-world workloads for time series is a huge gap.

IV. Conclusion

DL for time series is an important field with high impact in the industry, yet it lacks representation in today's performance benchmarks for ML systems. In this paper, we reviewed various application domains, discussed the unique challenges as well as potential directions in the performance evaluation of HW/SW systems in this field. We hope that this work can serve as a starting point for future work towards a representative performance benchmark of DL for time series.

References

[1] S. Narang and G. Diamos, "Baidu deepbench," https://github.com/baidu-research/DeepBench, 2017.

[2] H. Zhu, M. Akrout, B. Zheng, A. Pelegris, A. Jayarajan, A. Phanishayee, B. Schroeder, and G. Pekhimenko, "Benchmarking and analyzing deep neural network training," in *2018 IEEE International Symposium on Workload Characterization (IISWC)*. IEEE, 2018, pp. 88–100.

[3] C. Coleman, D. Narayanan, D. Kang, T. Zhao, J. Zhang, L. Nardi, P. Bailis, K. Olukotun, C. Ré, and M. Zaharia, "Dawnbench: An end-to-end deep learning benchmark and competition," *Training*, vol. 100, no. 101, p. 102, 2017.

[4] P. Mattson, C. Cheng, C. Coleman, G. Diamos, P. Micikevicius, D. Patterson, H. Tang, G.-Y. Wei, P. Bailis, V. Bittorf *et al.*, "Mlperf training benchmark," *arXiv preprint arXiv:1910.01500*, 2019. [Online]. Available: https://mlperf.org/

[5] J. D. Hamilton, *Time series analysis*. Princeton university press Princeton, NJ, 1994, vol. 2.

[6] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah, "Time-series clustering–a decade review," *Information Systems*, vol. 53, pp. 16–38, 2015.

[7] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Deep learning for time series classification: a review," *Data Mining and Knowledge Discovery*, vol. 33, no. 4, pp. 917–963, Jul 2019. [Online]. Available: https://doi.org/10.1007/s10618-019-00619-1

[8] J. C. B. Gamboa, "Deep learning for time-series analysis," *arXiv preprint arXiv:1701.01887*, 2017.

[9] "Traffic data collected from california department of transportation freeway detectors," http://pems.dot.ca.gov .

[10] "Tlc trip record data," https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page.

[11] "Uber trip data from a freedom of information request to NYC's taxi & limousine commission," https://github.com/fivethirtyeight/uber-tlc-foil-response, accessed: 2019-6-5. [Online]. Available: https://github.com/fivethirtyeight/uber-tlc-foil-response

[12] "Uber movement information," https://movement.uber.com/cities?lang=en-US, accessed: 2019-6-5. [Online]. Available: https://movement.uber.com/cities?lang=en-US

[13] L. Zhu and N. Laptev, "Deep and confident prediction for time series at uber," in *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2017, pp. 103–110.

[14] N. Laptev, J. Yosinski, L. E. Li, and S. Smyl, "Time-series extreme event forecasting with neural networks at uber," in *International Conference on Machine Learning*, no. 34, 2017, pp. 1–5.

[15] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=SJiHXGWAZ

[16] B. Yu, H. Yin, and Z. Zhu, "St-unet: A spatio-temporal u-network for graph-structured time series modeling," *arXiv preprint arXiv:1903.05631*, 2019.

[17] D. Roggen, A. Calatroni, M. Rossi, T. Holleczek, K. Förster, G. Tröster, P. Lukowicz, D. Bannach, G. Pirkl, A. Ferscha *et al.*, "Collecting complex activity datasets in highly rich networked sensor environments," in *2010 Seventh international conference on networked sensing systems (INSS)*. IEEE, 2010, pp. 233–240.

[18] R. Chavarriaga, H. Sagha, A. Calatroni, S. T. Digumarti, G. Tröster, J. d. R. Millán, and D. Roggen, "The opportunity challenge: A benchmark database for on-body sensor-based activity recognition," *Pattern Recognition Letters*, vol. 34, no. 15, pp. 2033–2042, 2013.

[19] G. Schalk, D. J. McFarland, T. Hinterberger, N. Birbaumer, and J. R. Wolpaw, "Bci2000: a general-purpose brain-computer interface (bci) system," *IEEE Transactions on biomedical engineering*, vol. 51, no. 6, pp. 1034–1043, 2004.

[20] "EEG motor Movement/Imagery dataset," https://www.physionet.org/pn4/eegmmidb/, accessed: 2019-6-4. [Online]. Available: https://www.physionet.org/pn4/eegmmidb/

[21] "Temple university hospital EEG (TUH EEG) resources," https://www.isip.piconepress.com/projects/tuh_eeg/, accessed: 2019-6-5. [Online]. Available: https://www.isip.piconepress.com/projects/tuh_eeg/

[22] "EEG / ERP data available for free public download," https://sccn.ucsd.edu/ arno/fam2data/publicly_available_EEG_data.html, accessed: 2019-6-5. [Online]. Available: https://sccn.ucsd.edu/ arno/fam2data/publicly_available_EEG_data.html

[23] "Mimic data," https://mimic.physionet.org/.

[24] F. Ordóñez and D. Roggen, "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, p. 115, 2016.

[25] P. Bashivan, I. Rish, M. Yeasin, and N. Codella, "Learning representations from eeg with deep recurrent-convolutional neural networks," *International conference on learning representations*, 2016.

[26] A. Y. Hannun, P. Rajpurkar, M. Haghpanahi, G. H. Tison, C. Bourn, M. P. Turakhia, and A. Y. Ng, "Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network," *Nature medicine*, vol. 25, no. 1, p. 65, 2019.

[27] S. Roy, I. Kiral-Kornek, and S. Harrer, "ChronoNet: A deep recurrent neural network for abnormal EEG identification," Jan. 2018. [Online]. Available: http://arxiv.org/abs/1802.00308

[28] Y. Roy, H. Banville, I. Albuquerque, A. Gramfort, T. H. Falk, and J. Faubert, "Deep learning-based electroencephalography analysis: a systematic review," *Journal of neural engineering*, 2019.

[29] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie, "Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[30] Z. Che, S. Purushotham, G. Li, B. Jiang, and Y. Liu, "Hierarchical deep generative models for multi-rate multivariate time series," in *International Conference on Machine Learning*, 2018, pp. 783–792.

[31] C. G. Lab, "Cmu graphics lab motion capture database," http://mocap.cs.cmu.edu/. [Online]. Available: http://mocap.cs.cmu.edu/

[32] J. Lien, N. Gillian, M. E. Karagozler, P. Amihood, C. Schwesig, E. Olson, H. Raja, and I. Poupyrev, "Soli: Ubiquitous gesture sensing with millimeter wave radar," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, p. 142, 2016.

[33] S. Wang, J. Song, J. Lien, I. Poupyrev, and O. Hilliges, "Interacting with soli: Exploring fine-grained dynamic gesture recognition in the radio-frequency spectrum," in *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, 2016, pp. 851–860.

[34] X. Li, Y. He, and X. Jing, "A survey of deep learning-based human activity recognition in radar," *Remote Sensing*, vol. 11, no. 9, p. 1068, 2019.

[35] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization." in *ICISSP*, 2018, pp. 108–116.

[36] Y. Mirsky, A. Shabtai, L. Rokach, B. Shapira, and Y. Elovici, "Sherlock vs moriarty: A smartphone dataset for cybersecurity research," in *Proceedings of the 2016 ACM workshop on Artificial intelligence and security*. ACM, 2016, pp. 1–12.

[37] S. Rezaei and X. Liu, "Deep learning for encrypted traffic classification: An overview," *IEEE Communications Magazine*, vol. 57, no. 5, pp. 76–81, 2019.

[38] D. S. Berman, A. L. Buczak, J. S. Chavis, and C. L. Corbett, "A survey of deep learning methods for cyber security," *Information*, vol. 10, no. 4, p. 122, 2019.

[39] P. Wang, X. Chen, F. Ye, and Z. Sun, "A survey of techniques for mobile service encrypted traffic classification using deep learning," *IEEE Access*, 2019.

[40] A. Ntakaris, M. Magris, J. Kanniainen, M. Gabbouj, and A. Iosifidis, "Benchmark dataset for mid-price forecasting of limit order book data with machine learning methods," *Journal of Forecasting*, vol. 37, no. 8, pp. 852–866, 2018.

[41] Kaggle, "Elliptic data set," https://www.kaggle.com/ellipticco/elliptic-data-set. [Online]. Available: https://www.kaggle.com/ellipticco/elliptic-data-set

[42] Q. Zhang, R. Luo, Y. Yang, and Y. Liu, "Benchmarking deep sequential models on volatility predictions for financial time series," *arXiv preprint arXiv:1811.03711*, 2018.

[43] Z. Zhang, S. Zohren, and S. Roberts, "Deeplob: Deep convolutional neural networks for limit order books," *IEEE Transactions on Signal Processing*, vol. 67, no. 11, pp. 3001–3012, 2019.

[44] M. Weber, G. Domeniconi, J. Chen, D. K. I. Weidele, C. Bellei, T. Robinson, and C. E. Leiserson, "Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics," *KDD Workshop on Anomaly Detection in Finance*, 2019.

[45] T. Shintate and L. Pichl, "Trend prediction classification for high frequency bitcoin time series with deep learning," *Journal of Risk and Financial Management*, vol. 12, no. 1, p. 17, 2019.

[46] F. Huch, "Kaggle software anomaly competition," https://www.kaggle.com/anomalydetectionml/datasets, accessed: 2019-6-4. [Online]. Available: https://www.kaggle.com/anomalydetectionml/datasets

[47] P. Filonov, A. Lavrentyev, and A. Vorontsov, "Multivariate industrial time series with cyber-attack simulation: Fault detection using an lstm-based predictive data model," *arXiv preprint arXiv:1612.06676*, 2016.

[48] F. Huch, M. Golagha, A. Petrovska, and A. Krauss, "Machine learning-based run-time anomaly detection in software systems: An industrial evaluation," in *2018 IEEE Workshop on Machine Learning Techniques for Software Quality Evaluation (MaLTeSQuE)*. IEEE, 2018, pp. 13–18.

[49] "United states historical climatology network (ushcn) daily dataset," https://cdiac.ess-dive.lbl.gov/ftp/ushcn_daily/.

[50] Kaggle, "LANL earthquake prediction can you predict upcoming laboratory earthquakes?" https://www.kaggle.com/c/LANL-Earthquake-Prediction, accessed: 2019-10-27. [Online]. Available: https://www.kaggle.com/c/LANL-Earthquake-Prediction

[51] USGS, "Earthquake hazards program: Search earthquake catalog," https://earthquake.usgs.gov/earthquakes/search/, accessed: 2019-11-4. [Online]. Available: https://earthquake.usgs.gov/earthquakes/search/

[52] Bill Tang, "New models for integrated inquiry: Fusion energy exemplar," in *Online Resource for Big Data and Extreme-Scale Computing Workshop*, G. F. Terry Moore, Ed., Nov. 2018. [Online]. Available: https://www.exascale.org/bdec/sites/www.exascale.org.bdec/files/whitepapers/Tang_BDEC2_WP.pdf

[53] J. Kates-Harbeck, A. Svyatkovskiy, and W. Tang, "Predicting disruptive instabilities in controlled fusion plasmas through deep learning," *Nature*, vol. 568, no. 7753, pp. 526–531, Apr. 2019.

[54] Ashley Smart, "Artificial intelligence takes on earthquake prediction," https://www.quantamagazine.org/artificial-intelligence-takes-on-earthquake-prediction-20190919/, accessed: 2019-10-27. [Online]. Available: https://www.quantamagazine.org/artificial-intelligence-takes-on-earthquake-prediction-20190919/

[55] P. M. R. DeVries, F. Viégas, M. Wattenberg, and B. J. Meade, "Deep learning of aftershock patterns following large earthquakes," *Nature*, vol. 560, no. 7720, pp. 632–634, Aug. 2018. [Online]. Available: http://dx.doi.org/10.1038/s41586-018-0438-y

[56] J. B. Rundle, M. Luginbuhl, A. Giguere, and D. L. Turcotte, "Natural time, nowcasting and the physics of earthquakes: Estimation of seismic risk to global megacities," in *Earthquakes and Multi-hazards Around the Pacific Rim, Vol. II*, C. A. Williams, Z. Peng, Y. Zhang, E. Fukuyama, T. Goebel, and M. R. Yoder, Eds. Cham: Springer International Publishing, 2019, pp. 123–136. [Online]. Available: https://doi.org/10.1007/978-3-319-92297-3_10

[57] "M4 competition combined multi-area time series," https://www.mcompetitions.unic.ac.cy/the-dataset/.

[58] "Individual household electric power consumption data set," http://archive.ics.uci.edu/ml/datasets/Individual+household+electric+power+consumption.

[59] M. Bińkowski, G. Marti, and P. Donnat, "Autoregressive convolutional neural networks for asynchronous time series," *arXiv preprint arXiv:1703.04122*, 2017.

[60] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, "Deepar: Probabilistic forecasting with autoregressive recurrent networks," *International Journal of Forecasting*, 2019.

[61] A. Redd, K. Khin, and A. Marini, "Fast es-rnn: A gpu implementation of the es-rnn algorithm," *arXiv preprint arXiv:1907.03329*, 2019.

[62] H. Hewamalage, C. Bergmeir, and K. Bandara, "Recurrent neural networks for time series forecasting: Current status and future directions," *arXiv preprint arXiv:1909.00590*, 2019.

[63] T. Wen and R. Keyes, "Time series anomaly detection using convolutional neural networks and transfer learning," *CoRR*, vol. abs/1905.13628, 2019. [Online]. Available: http://arxiv.org/abs/1905.13628

[64] M. Khairalla, Xu-Ning, and N. T. AL-Jallad, "Hybrid forecasting scheme for financial time-series data using neural network and statistical methods," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 9, 2017. [Online]. Available: http://dx.doi.org/10.14569/IJACSA.2017.080945

[65] Y. Li, R. Yu, C. Shahabi, and Yanecasting, "Graph convolutional recurrent neural network: Data-driven traffic forecasting," *CoRR*, vol. abs/1707.01926, 2017. [Online]. Available: http://arxiv.org/abs/1707.01926

[66] A. Ng, *Machine learning yearning: Technical strategy for AI engineers in the era of deep learning*, 2019. [Online]. Available: https://www.deeplearning.ai/machine-learning-yearning/

[67] V. J. Reddi, C. Cheng, D. Kanter, P. Mattson, G. Schmuelling, C.-J. Wu, B. Anderson, M. Breughe, M. Charlebois, W. Chou *et al.*, "Mlperf inference benchmark," *arXiv preprint arXiv:1911.02549*, 2019. [Online]. Available: https://mlperf.org/

[68] G.-Y. Wei, D. Brooks *et al.*, "Benchmarking tpu, gpu, and cpu platforms for deep learning," *arXiv preprint arXiv:1907.10701*, 2019.

[69] C. Li, A. Dakkak, J. Xiong, W. Wei, L. Xu, and W.-m. Hwu, "Across-stack profiling and characterization of machine learning models on gpus," *arXiv preprint arXiv:1908.06869*, 2019.

[70] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.