Machine Learning-Based GPU Energy Prediction for Workload Management in Datacenters

Authors:
Brandon Ismalej
Matthew Smith
Dr. Xunfei Jiang

Department of Computer Science
California State University, Northridge

Motivation

- Data Centers (DCs) consume **~1%** of global energy (2020) [1]
- Rising demand for AI, ML and Cloud Computing will significantly increase this consumption

- GPUs are key for these workloads but inefficient GPU management leads to:
  - Suboptimal resource utilization
  - Higher energy consumption
  - Increased costs
- Efficient, GPU-aware workload management in DCs can help:
  - Maximize resource efficiency
  - Minimize energy consumption
  - Contribute to sustainable and scalable DC operations

Problem & Objective

- **Problem:**
  - Limited research exists on predicting GPU power consumption using real-world workload data
  - Publicly available workload traces are scarce, hindering accurate modeling

- **Objective:**
  - Develop GPU power prediction models using synthesize data to emulate real-world workloads
  - Integrate model into GPUCloudSim Plus to further study of energy-aware workload simulations
  - Support energy-efficient GPU workload management in DC’s

Design

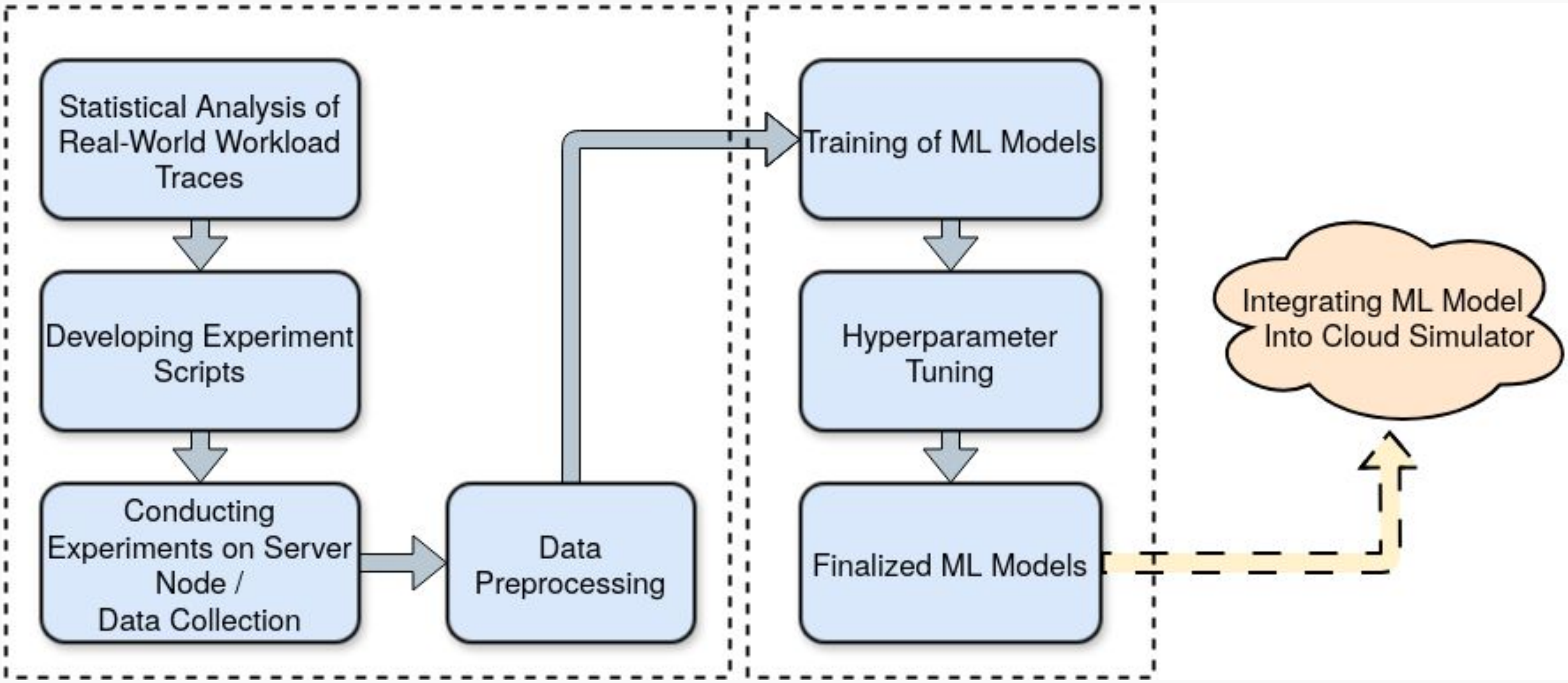


Figure 1: Research Design Framework

Statistical Analyses of Workload Traces

- **Conducted on:**
  - Alibaba v2020 Cluster Trace [2]
    - 6,500 GPUs
    - Spanning ~1,800 machines
  - SenseTime Helios GPU Cluster Trace [3]
    - ~6,000 GPUs
    - Spanning 4 GPU clusters
- **In this analysis:**
  - Overlapping tasks excluded
  - Focused on 95th percentile
    - To gain a broader view of non-zero inter-task delay times

- Insights gained from this analysis shaped design of experiments.

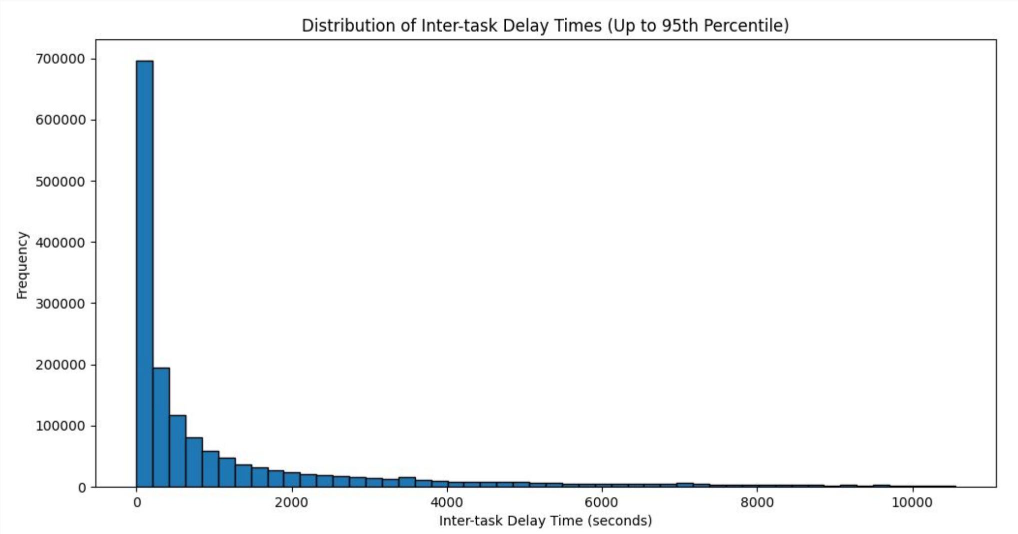


Figure 2: Distribution of Alibaba Trace

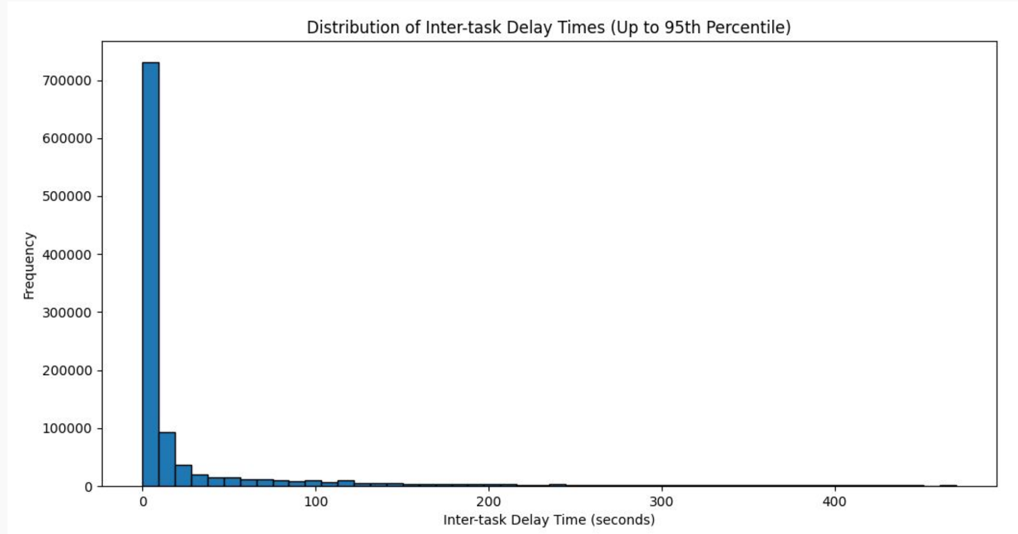


Figure 3: Distribution of Helios Trace

	Alibaba Trace	Helios Trace
Count	1,669,790	1,112,131
Q1	67	2
Median	348	4
Q3	1,624	21
Max	2,786,690	4,196,357

Table 1: Summary Statistics of Inter-Task Delay Times

Experiments & Data Collection

- A set of 14 benchmarks with varying GPU usage:

Benchmark Application	Domain	Avg. Power [W]	Avg. GPU Util [%]	GRAM Avg. [GiB]
2D CONVOLUTION	Image Processing	59.05	98.10	0.60
BERT_QA_PREDICTION	Natural Language Processing	60.73	95.05	1.29
BLACKSCHOLES	Computational Finance	22.86	8.83	0.16
DISTILBERT_TRAINING	Natural Language Processing	57.87	98.62	6.47
EUCLIDEAN_DISTANCE	Mathematical Computation	51.08	99.33	1.64
FFT	Signal Processing	55.19	99.45	4.61
GAN_MNIST_DIGITS	Generative Adversarial Network	60.77	97.96	0.79
IMAGE CLASSIFIER	Image Classification	23.09	6.74	0.29
K MEANS	Unsupervised Learning	22.86	17.46	0.41
MATMUL	Matrix Multiplication	27.85	13.06	1.36
MONTE CARLO PRICING	Financial Simulation	43.01	98.31	3.48
SEPIA_TRANSFORMATION	Image Processing	23.10	5.94	0.26
SPECTROGRAM_TRANSFORMATION	Audio Processing	23.65	5.18	0.18
SIMPLE CNN	Image Recognition	42.30	97.21	2.02

Table 2: Domain and Metrics of GPU Benchmark Applications

Experiments & Data Collection

- Designed five distinct experiments to cover a wide range of inter-task delay times, using our benchmarks

CPU	2 * Intel(R) Xeon(R) CPU E5-2690 v2 @ 3.00GHz (10 cores)
Memory	98,304 MB
Disk	2 * 480 GB
GPU	NVIDIA Tesla P4 8GB GDDR5
OS	Ubuntu 22.04.2 LTS

Table 3: Cluster System Configuration

- "No delays" simulates peak usage
- Short delays simulate high-frequency task switching
- Longer delays reflect idle periods, like downtime

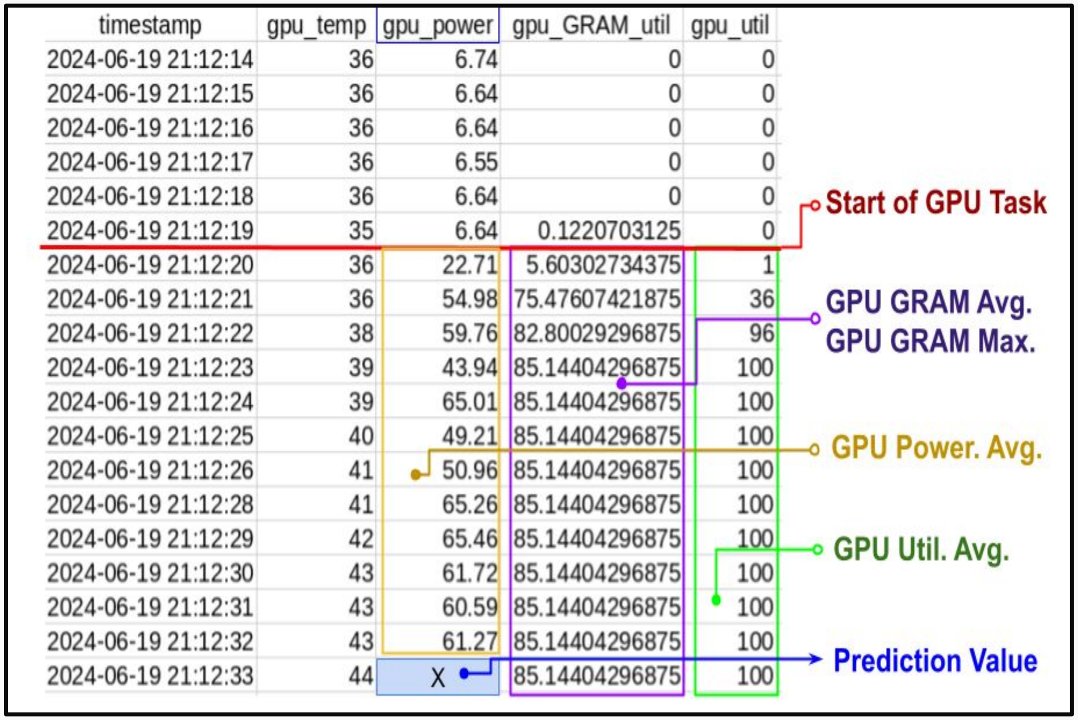
	Delay Times (seconds)	**Task Order**
Exp 1	No delays	Sequential, reverse, shuffled
Exp 2	0 - 10	shuffled
Exp 3	1 - 20	Sequential, reverse, shuffled
Exp 4	1 - 30	shuffled
Exp 5	300 - 1000	shuffled

Table 4: Experiment Configurations

- Experiments yielded ~40 hours of data
- Data collected second-by-second

Experiments & Data Collection

- Features extracted for duration of every task-active and idle period
  - To align with available features of real-world workloads that would be implemented in cloud simulator



timestamp	gpu_temp	gpu_power	gpu_GRAM_util	gpu_util
2024-06-19 21:12:14	36	6.74	0	0
2024-06-19 21:12:15	36	6.64	0	0
2024-06-19 21:12:16	36	6.64	0	0
2024-06-19 21:12:17	36	6.55	0	0
2024-06-19 21:12:18	36	6.64	0	0
2024-06-19 21:12:19	35	6.64	0.1220703125	0
2024-06-19 21:12:20	36	22.71	5.60302734375	1
2024-06-19 21:12:21	36	54.98	75.47607421875	36
2024-06-19 21:12:22	38	59.76	82.80029296875	96
2024-06-19 21:12:23	39	43.94	85.14404296875	100
2024-06-19 21:12:24	39	65.01	85.14404296875	100
2024-06-19 21:12:25	40	49.21	85.14404296875	100
2024-06-19 21:12:26	41	50.96	85.14404296875	100
2024-06-19 21:12:28	41	65.26	85.14404296875	100
2024-06-19 21:12:29	42	65.46	85.14404296875	100
2024-06-19 21:12:30	43	61.72	85.14404296875	100
2024-06-19 21:12:31	43	60.59	85.14404296875	100
2024-06-19 21:12:32	43	61.27	85.14404296875	100
2024-06-19 21:12:33	44	X	85.14404296875	100

Figure 4: Illustration of Data Preprocessing

- **GRAM Avg. & Max. [GiB] ; Util. Avg. [%] ; Power Avg. [W]**

Modelling GPU Power

- Four machine learning algorithms utilized:
  - XGBoost (eXtreme Gradient Boosting)
  - CatBoost (Categorical Boosting)
  - LightGBM
  - LSTM (Long Short-Term Memory)

- Utilized a 60/20/20 split of data for:
  - Training/Testing/Validation

Modelling GPU Power

- To predict GPU power consumption:
  - Training Input:
    - GPU GRAM Avg. [GiB]
    - GPU GRAM Max. [GiB]
    - GPU Utilization Avg. [%]
  - Training Output:
    - GPU Power Avg. [W]

- Initial ML model training results:

	CatBoost	**LightGBM**	**XGBoost**	**LSTM**
RMSE Value	1.218	1.224	1.284	1.520

Table 5: RMSE Values of Initial Model Training

- Grid-search hyperparameter tuning performed to improve RMSE

Modelling GPU Power

- Best performing model - **XGBoost** RMSE = 1.217

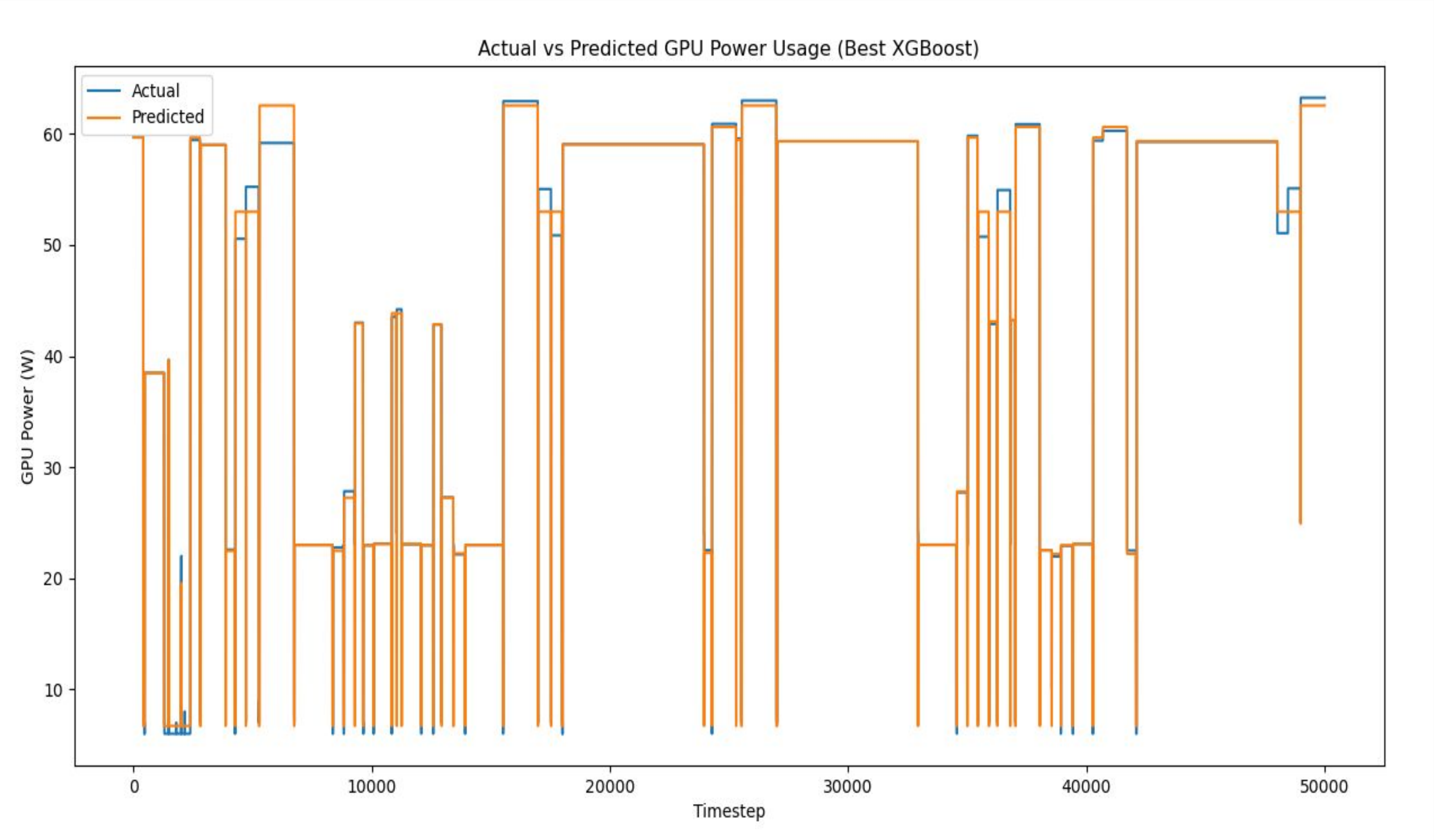


Figure 5: Best XGBoost Model Predicted vs. Actual GPU Consumption

Model Integration to Simulator

- Processing the Alibaba 2020 workload trace:
  - Using 4 files of: machine spec, task, instance, and sensor

- Filter out tasks missing:
  - Planned VM resources
  - Sensor, server info.
  - Task duration

Future Work

- Integrate new GPU power model into modified GPUCloudSimPlus
- Model energy for servers using multi-CPUs and multi-GPUs
- Rewrite load balancing algorithms
- Run full duration simulations

Acknowledgement

- Supported by the SfS[2] Program and funded by the United States Department of Education FY 2023 Title V, Part A, Developing Hispanic-Serving Institutions Program five-year grant, Award Number P31S0230232, CFDA Number 84.031S.

- However, the contents of this presentation do not necessarily represent the policy of the US Department of Education, and you should not assume endorsement by the Federal Government.

SECURE for Student Success

References

- [1] E. Masanet, A. Shehabi, N. Lei, S. Smith, and J. Koomey, “Recalibrating global data center energy-use estimates,” Science, vol. 367, no. 6481, p. 984–986, Feb. 2020.
- [2] Q. Weng, W. Xiao, Y. Yu, W. Wang, C. Wang, J. He, Y. Li, L. Zhang, W. Lin, and Y. Ding, “MLaaS in the wild: Workload analysis and scheduling in large-scale heterogeneous GPU clusters,” in 19th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 22), 2022.
- [3] Q. Hu, P. Sun, S. Yan, Y. Wen, and T. Zhang, “Characterization and prediction of deep learning workloads in large-scale gpu datacenters,” in Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, ser. SC ’21. New York, NY, USA: Association for Computing Machinery, Nov. 2021, p. 1–15. [Online]. Available: https://dl.acm.org/doi/10.1145/3458817.3476223

Questions?

brandon.ismalej.671@my.csun.edu

CSUN® | COMPUTER SCIENCE