COMS E 6998 Practical Deep Learning Systems Performance

Overview

This course serves as a graduate-level practical introduction to Deep Learning, with an emphasis on practical system performance issues and related research. The course will cover several topics related to Deep Learning (DL) systems and their performance. Both algorithmic and system related building blocks of DL systems will be covered including DL training algorithms, network architectures, and best practices for performance optimization. We will study hyperparameter choices, scalable distributed DL training, Kubernetes based DL system stack on cloud, tools and benchmarks for performance evaluation of DL systems, transfer learning, federated learning etc. Emphasis will be on getting working knowledge of tools and techniques for performance evaluation of DL systems. The students will gain practical experience working on different stages of DL life cycle, including model development, testing, and deployment. The assignments will be mostly hands-on involving standard DL frameworks (Tensorflow, Pytorch) and open source technologies.

Prerequisites

Basic knowledge of machine learning concepts; Knowledge of python and experience using Jupyter Notebook is preferred.

Syllabus

Module 1: Introduction to Machine Learning (ML) and Deep Learning (DL)
ML revolution and cloud; Overview of ML algorithms, Supervised and Unsupervised Learning; ML performance concepts/techniques: bias, variance, generalization, regularization; Performance metrics: algorithmic and system level; DL training: backpropagation, gradient descent, activation functions, data preprocessing, batch normalization, exploding and vanishing gradients, weight initialization, learning rate policies; Regularization techniques in DL Training: dropout, early stopping, data augmentation

Module 2: DL Training Architectures, Frameworks, Hyperparameters
Stochastic and mini-batch gradient descent; Gradient descent strategies: momentum-based, AdaGrad, AdaDelta, RMSProp, Adam; DL training architectures: model and data parallelism, single node training, distributed training, parameter server, all reduce; DL training hyperparameters: batch size, learning rate, momentum, weight decay, convergence and runtime issues; Hardware acceleration: GPUs, Tensor cores, NCCL, Intra and Inter node performance; Specialized DL architectures: CNNs, RNNs, LSTMs, GANs

Module 3: Cloud Technologies and ML Platforms
ML system stack on cloud; Micro-services architecture: docker, kubernetes, kubeflow, katib; Cloud storage: file, block, object storage, performance and flexibility; Network support on cloud platforms; Cloud based ML platforms from AWS, Microsoft, Google, and IBM; System stack,

capabilities and tools support in different platforms; Monitoring, performance, availability, and observability

Module 4: DL Performance Evaluation Tools and Techniques
Monitoring GPU resources (nvprof, nvidiasmi), network monitoring; Time series analysis of resource usage data; Predictive performance modeling techniques: black-box vs white-box modeling, regression modeling, analytical modeling; Predictive performance models for DL convergence and runtime

Module 5: ML/DL Benchmarking
DAWNBench, MLperf,, TensorflowHPM, Kaggle, OpenML; Datasets: MNIST, CIFAR10/100, ImageNet; Performance metrics for DL jobs; Runtime, cost, response time, accuracy, time to accuracy (TTA); Study of published numbers by different cloud service providers/vendors at benchmark forums; Open Neural Network Exchange (ONNX)

Module 6: DL Systems Performance Evaluation
Training-logs: framework specific support, instrumentation, analysis ; Checkpointing: framework specific support, restarting from checkpoint; Job scheduling policies like FIFO, gang, earliest deadline first, dominant resource fairness; Job schedulers for DL clusters: Kubernetes, Gandiva, Optimus; Job Elasticity: scaling GPUs during runtime, platform support; Scalability: learners, batch size, single node, distributed

Module 7: Operational Machine Learning
Devops principles in machine learning, Model lifecycle management, MLOps; DL system testing and quality: Full life cycle testing; DL deployment and production cycle; Drift detection and re-training; Robustness and adversarial training; Automated Machine Learning; MLOps tool-chain

Module 8: Special topics in DL Systems
Learning with limited data: transfer learning and pseudo-labeling techniques; Deep reinforcement learning systems; Neural network synthesis and architecture search including hyperparameter optimization techniques; Federated learning and ML on edge devices

Homework

There will be **five homework assignments**..All programming assignments should be done using Jupyter notebook. Both the notebook and its pdf should be submitted.

Course Information

- **Instructor**: Parijat Dube

- **Grading:** Homework (50%) + Final Project (25%) + Midterm Seminar (15%) + Quizzes (10%)

- **Homework**
  - Assignments will use Python, Tensorflow, PyTorch
  - Assignments will involve running Deep Learning training jobs on GPU enabled public cloud platforms and use of open source code/technologies

- **Course project**
  - Project proposals are due by midterm
  - Maximum team size is 2.
  - Final presentations of all projects towards the end of the course.