BITS Pilani

Work Integrated Learning Programs

Part A: Content Design

Course Title	**ML System Optimization**
Course No(s)	**AIML ZG516**
Credit Units	4
Credit Model	2 +1 + 1 2 unit for class room hours, 1 unit for Reading, 1 unit for Practical Work
Content Authors	Shan Sundar Balasubramaniam
Version	1.0
Date	March 11th, 2023

ML System Optimization

1. Course Objectives:

- Expose learners to the inter-play of ML algorithms and modern-day Computing systems through
  - o Computational Performance and scalability of these algorithms using modern-day systems (such as multi-core CPUs, GPGPUs, clusters, and constrained devices) and/or platforms for ML and Big Data and
  - o The impact of performance improvement techniques on (domain i.e., ML) quality attributes

2. Learning Outcomes:

- Understand and articulate how parallel/distributed ML algorithms leverage standard platforms for ML to obtain performance.
- Implement parallel/distributed ML algorithms on clusters and constrained / Small-Form-Factor devices (such as mobile phones)

- Argue cogently and/or demonstrate the systems-level performance of a broad class of parallel/distributed ML algorithms.

3. Scope and Disambiguation:

- The course is expected to be a broad introduction to <u>systems aspects</u> of ML/DL and expects (as input) a basic understanding of, if not expertise in, Computing Systems in general.
- ML System in general may refer
  1. Computing Systems on which ML algorithms run and/or on which ML applications are implemented
     [***Focus of this course!***]
  2. The overall Computing framework on which ML algorithms and ML applications are trained and deployed.
     [**Should be the focus of MLOps and SE for AIML**]
- This course draws heavily from the knowledge of ML algorithms.
- The focus of the course is on the systems aspects of these algorithms whereas the algorithms themselves may only be briefly exposed as preparation to understanding the systems aspects.

4. Modules

	Module	**Description**
M1	Introduction	Set the context: Contour of ML Solutions, Parallelization/Distribution, Modern Systems
M2	Parallel/Distributed ML algorithms	Introduce how to parallelize/distribute a selection of typical ML algorithms (the training phase)
M3	Scale-out ML	Explain how standard Scale-out platforms (TensorFlow, Spark) obtain performance Explain how large scale neural networks can be distributed
M4	ML under Systems Constraints	Introduce techniques for deploying ML solutions under systems constraints (running time, storage, bandwidth, and energy)

5. Text / References: NONE

Part B: Learning Plan

Academic Term	2nd Sem. 2022-23
Course Title	ML System Optimization
Course No	**AIML CLZG516**
Lead Instructor	Shan Sundar Balasubramaniam

1.Session Plan: (Lectures)

[Note:

- Reading/References will be assigned per session.
- Each session will require reading advanced material and there are no text books.
- Pedagogy:
  - o Some topics require strong grounding in ML/DL including the math
  - o whereas some topics require a broad but sound understanding of systems including Distributed Systems, Small FF Devices/Systems/ Multi-core/GPU architectures.

End of Note.]

Session	Topics	Notes
M1	**Introduction and Context**	
1	ML and DL: 1. Performance: a. Metrics: Time Complexity of Algorithms and Running Time; Memory, Response Time b. Scaling and Tuning of Performance 2. Environments:	● *Broad understanding* required: of **Algorithmic Complexity**, and Performance metrics like **Throughput and Response**

	a. Training vs. Deployment b. Range of Systems: Distributed and Cloud, Embedded and Mobile.	**Time**
2	Parallel and Distributed Algorithms: 1. Systems and Performance; 2. Speedup – Approaches and Issues; 3. Data Parallelism vs. Task Parallelism vs. Request Parallelism. 4. Scale-out Clusters – Cost of communication and impact on Speedup	● *Desired understanding:* **Speedup: Amdahl's Law, Scale-up vs. Scale-out**
3	Modern Systems: 1. Parallel Execution on Multicore processors and GPGPUs 2. Distributed Execution on Clusters: (CPU and GPU clusters) - Data Distribution Strategies	● *Desired understanding:* **Parallel and Multi-core Processing**
M2 Parallel / Distributed ML algorithms - Overview and Techniques		
4-6	Parallel / Distributed ML algorithms - Overview and Techniques: 1. CNN 2. Gradient Descent and Stochastic Gradient Descent 3. SVM 4. k-Means 5. kNN 6. Decision Trees/Random Forests.	● *Prior Knowledge*: **ML algorithms**
M3. Scale-out ML: Systems Aspects		
7-8	1. Large Scale Machine Learning Systems: a. The Parameter Server Model b. Spark Architecture c. TensorFlow Architecture 2. Execution of ML (or Big Data) Algorithms on parallel / distributed systems:	● *Prior Knowledge*: **Client-Server Model, Scale-out Clusters**

	a. Performance Improvement and Trade-offs	
9-12	Distributed Neural Networks 1. Decentralized and Local SGD – System Support (All-reduce, Asynchronous Parallelism) 2. Large Scale Deep NN 3. Systems for Federated Learning	● *Prior Knowledge:* **Deep NNs, SGD**
M4. ML Performance under Systems Constraints		
13	ML Deployment on Constrained Systems I: 1. Model Compression, Compression vs. Inference 2. Quantization and Learning with Limited Numerical Precision	● *Prior Knowledge:* **Deep NNs**
14	Neural Network Pruning 1. Pruning of CNNs 2. Evaluation of Pruning 3. Deep Compression: Leveraging quantization, pruning, and sparsity.	● *Prior Knowledge:* **Deep NNs,**
15	ML Deployment on Constrained Systems II: 1. TinyML and TensorFlow Lite; 2. Energy Constraints – Adapting Algorithms for Constrained Devices; 3. Assessing the tradeoffs - Accuracy of prediction, Model Size, Throughput, Response Time, Energy Consumption	
16	Summary and Conclusion	

2. Assignment / Project [Course credits are distributed **3+1=4**]

[Note on Pedagogy:

- The assignment and project components are intended for learning-by-doing (of appropriate systems and platforms for ML) as opposed to skill development.

- The primary objective is to understand the pragmatics of implementing ML.

End of Note on Intent/Pedagogy]

3. Evaluation

Component	**Weight**	**Duration**	**Schedule**
Assignment	15%	Take-home (3 to 4 weeks)	TBA (before mid-term)
Project	30%	Take-home (about 6 weeks)	TBA (after mid-term)
Mid-Semester Test	25%	120 minutes	Centrally scheduled
Comprehensive Exam	30%	150 minutes	

---END---