Parallel Machine Learning & AI – CSYE7374

Spring 2020 38227

Time: Monday 2:00 – 5:30 PM
Location: Behrakis Health Science Center 204
Instructor: **Prof. Handan Liu**
Email: h.liu@northeastern.edu

COURSE GOALS

This course is designed to explore the parallelism of machine learning and deep learning to achieve high speedup and high performance on heterogeneous cluster architectures, as well as the applications to a variety of domains, including image classification, speech recognition, and natural language processing. The homeworks explore parallel algorithms and simple applications, and the final project allows an in-depth exploration of a particular application area.

PREREQUISITES

You should be comfortable with programming in Python. It's even better if you understand the fundamentals of machine learning and the Linux operating system.

GRADING

- Homework: 50% (HW1 10%, HW2 20%, HW3 20%)
- Project: 40%
    - Group Grading: Proposal 5%, Final Report 10%, Coding 10%, Presentation Slides 5%
    - Individual Grading: Oral Speech 10%
- Attendance: 10%

Final grades will be assigned based on the following scale:

A	92 - 100	**C+**	77 - <80
A-	90 - <92	**C**	72 - <77
B+	87 - <90	**C-**	70 - <72
B	82 - <87	**F**	<70
B-	80 - <82		

ASSIGNMENTS

- Lectures are complemented by homework (programming assignments) to bridge the theory with the practice. The homework assignments are associated with the three main parts of the course that mostly consist of programming assignments to exercise a technology or programming model.
- Submissions will be made via **Blackboard** as a single file or single ZIP file. Any written work (e.g. math problems, reports) is to be included as a Word or PDF. Code is finished in python files. Code is expected to be professional and properly documented; any required data files/libraries must be included. Code can be run correctly.
- All assignments have a specific due date and time. Submissions will be accepted up to one day after the deadline with a 50% penalty. For example, an on-time submission might receive a grade of 90 points. The same assignment submitted after the deadline would receive 45 points (90 x 0.5).
- Assignment Review or Code Review is scheduled by TAs. All students should follow the schedule of TAs. TAs will grade the assignment for the instructor's reference.
- The 3 assignments are scheduled as below:

	Assigned	Due date
HW1	January 28	February 7
HW2	February 25	March 6
HW3	March 31	April 10

PROJECT

- The goal of the project is to gain hands-on experience with a real-life dataset.
- The project will be completed by groups. Each group consists of about 3 students, and is freely combined.

	Date
Start the team-up	February 24
Finish the team-up	March 9
Submit proposals	March 19
Instructor reviews and comments proposals	March 30
Submit all materials of your project	April 3

MAKE-UP POLICY

Students who miss the presentation will not, as a matter of course, be able to make up it. If you have a legitimate reason, you should contact the instructor beforehand and get the approval.

If there is a legitimate reason why a student will not be able to complete an assignment on time, they should contact the instructor beforehand. Under extreme circumstances, as decided on a case-by-case basis by the instructor, students may be allowed to make up assignments or other requirements without first informing the instructor.

COURSE SCHEDULE

Note: This schedule **is subject to change** and will be adjusted as needed throughout the semester.

Week	Topics	Reading	Assignments
Jan. 6	• Syllabus details • Introduction to High Performance Parallel Computing • Introduction to the NEU cluster and Linux fundamentals Hands-on Lab: login Discovery cluster; data transfer, load modules, using Slurm, running jobs, interactive mode and batch mode; scripting for job submission; Linux commands; Learn how to compile and run OpenMP and MPI programs (in C), etc.		
Jan. 13			
Jan. 27			HW1
Feb. 3	<u>Parallel Machine Learning: 4 lectures</u> • Parallel methods for hyperparameter and cross-validation in sklearn by multiprocessing, joblib etc. Hands-on Lab: practice examples by using multiprocessing and joblib on the cluster		
Feb. 10			
Feb. 24	• PySpark Machine Learning: MLlib Hands-on Lab: practice examples by PySpark MLlib on the cluster		HW2
Mar. 9	• XGBoost parallel in OpenMP, MPI and CUDA on CPU and GPU on HPC cluster Hands-on Lab: practice examples by XGBoost in CPU and GPU on the cluster		
Mar. 16	<u>Parallel Deep Learning/AI: 4 lectures</u> • TensorFlow parallel training on CPU and GPU, with Keras Hands-on Lab: practice examples in the areas of image classification, speech recognition, natural language processing etc. by TensorFlow and Keras on CPU and GPU on the cluster		
Mar. 23			

Mar. 30	• Microsoft Cognitive Toolkit - CNTK with CUDA-Aware MPI and NCCL on GPU parallel computing • ChainerMN: Distributed Deep Learning with Chainer, with CUDA-Aware MPI and NCCL on GPU parallel computing Hands-on Lab: practice examples in various areas image classification, speech recognition, natural language processing, etc. by CNTK and ChainerMN on GPU parallel computing		HW3
Apr. 6			
Apr. 13	Presentation I by groups		
Apr. 20	Presentation II by groups		

RESOURCES

Students are expected to read the materials before class.

- Slack (for discussion)
- Textbook: **No books in the markets**
- MPI
    - https://www.open-mpi.org/
    - https://www.mpich.org/
    - https://www.mpi-forum.org/
- OpenMP https://www.openmp.org/
- NVidia: High Performance Supercomputing and CUDA
    - https://developer.nvidia.com/cuda-zone
    - https://www.nvidia.com/en-us/data-center
- Multiprocessing
    - https://docs.python.org/3.4/library/multiprocessing.html?highlight=process
- IPython Parallel Computing
    - http://ipython.org/ipython-doc/dev/parallel/
- PySpark Machine Learning Library:
    - https://spark.apache.org/docs/2.3.0/ml-tuning.html
- NVidia Deep Learning:
    - https://developer.nvidia.com/deep-learning
    - https://www.nvidia.com/en-us/deep-learning-ai/education/
- TensorFlow https://www.tensorflow.org/

- Microsoft CNTK
    - https://docs.microsoft.com/en-us/cognitive-toolkit/
    - https://github.com/microsoft/CNTK
- Chainer/ChainerMN
    - https://chainer.org/
    - https://github.com/chainer/chainermn
    - https://docs.chainer.org/en/stable/chainermn/

Other reference materials will be provided according to different needs during class.