EE 382V: Cross-Layer ML Algorithm and Hardware Co-Design

Spring 2020

Instructors: Prof. Mattan Erez and Prof. Michael Orshansky

Lecture Hours: Mon Wed 1:30PM - 3:00PM, ECJ 1.318
Office Hours: TBD
Office: EER 4.814 (Orshansky) EER 5.872 (Erez)
E-Mail: mattan.erez@utexas.edu, orshansky@utexas.edu
Web: https://lph.ece.utexas.edu/merez/MattanErez/Home
http://www.ece.utexas.edu/~michael

This course focuses on co-design of machine learning algorithms and hardware accelerators. Co-design is essential for ML deployment because resources and time per query and for training are constrained. The course provides in-depth discussion of algorithmic, architectural, and circuit-level techniques for trading off accuracy, resources, and speed when designing accelerators for machine learning systems. Students will learn how to design implementations that exploit and trade off algorithmic flexibility using formal design space exploration tools, such as roofline modeling. The course will focus on the methodology used for a cloud-based FPGA deployment on AWS F1 instances.

The course focuses on the compute-heavy deep neural network (DNN) models of machine learning, such as convolutional neural nets, recurrent neural nets, recommender systems, and attention mechanisms. Examples of state-of-the-art tradeoff approaches include compression methods, including sparse pruning and group convolutions, structured weights, dynamic pruning, non-conventional numerical representations and quantization schemes, and data movement optimizations. The course will emphasize cross-layer optimization with circuit-level techniques, including analog/mixed-signal MACs, RRAM/SRAM memory array compute, binary/XOR networks, and approximate computing. Students will read state-of-the-art research papers and complete a design project.

Prerequisites
Required:

- EE460N/EE382N.1 (or equivalent: details of high-performance processor pipelines and memories)
- EE316/460M (or equivalent: logic and digital system design with Verilog/VHDL)
- EE312/422C/360C (or equivalent: experience with programming and a variety of modern programming languages, data structures, and basic algorithms)

Highly recommended:

- EE382M-20 (System-on-Chip Design)
- EE382N-20 (Parallelism and Locality)
- EE460J (Data Science Lab)
- Any machine-learning course
- Any GPU and CUDA course/experience

Tentative Topical Outline

1. Foundational Linear Algebra
   - Intro to ML and relation to computational kernels
   - High performance matrix multiplication
   - Linear algebra fundamentals and accelerating linear algebra, BLAS operations

2. Overview of Common DNN Algorithms
   - CNN architectures (AlexNet, ResNet, Amoeba)
   - RNNs, attention-based methods
   - Recommender Systems: Embeddings, Wide and Deep Models

3. Algorithm/HW Co-Design
   - Dataflows, model/data parallelism, Domain-specific language (Spatial)
   - Design space: roofline model
   - Pruning/Precision/Compression
   - Fast convolution, Winograd transform
   - Numerical errors (floats, rounding, truncation)
   - Group convolution, structured weights

4. Circuit-level techniques for ultra-low-power ML
   - Analog/mixed-signal MACs
   - RRAM/SRAM based techniques
   - Binary networks
   - Approximate computing

Grading

Reading Assignments: 30%

> Reading material will include advanced research papers (from venues such as, ASPLOS, DAC, ISSCC, ISCA, ICML, NeurIPS, and SysML) as well as tutorials and blog posts. Reading assignments will include write-ups in preparation for discussions and preparing a *visual abstract* for two of the papers. For some of the reading assignments, a short in-class quiz may be administered (announced ahead of time).

Programming/design assignments: 30%

> Programming assignments using PyTorch/TensorFlow/Spatial/HLS/Verilog to analyze DNN algorithms and architectures for energy and performance.

Design Project: 40%

> Design project to implement and analyze an accelerator.

Course Website

We will use the web-based course management system “Canvas”. The students are responsible for regularly checking the course web page for announcements and postings at https://utexas.instructure.com/.

Drop Policy: The last day to drop this course without permission from the Dean is the 4th class day. After this day, drops are approved only in the case of health or personal problems. An engineering student should make an appointment with his/her departmental advisor to discuss adding or dropping any course if the change will alter the classes that were originally approved by the departmental advisor. If the add or drop requires the approval of the Dean, then the student will need to schedule an appointment with an Academic Advisor in the Office of Student Affairs, ECJ 2.200 (471-4321) to discuss the request. Additional information can be found at: http://www.engr.utexas.edu/current/policies/pol_add-drop-wdraw.cfm

Academic Dishonesty: Cheating will **not** be tolerated and will be dealt with according to the policy established by the office of the Dean of Students.

Students with disability: The University of Texas at Austin provides, upon request, appropriate academic adjustments for qualified students with disabilities. For more information, contact the Office of the Dean of Students at 471-6259, or the College of Engineering Director of Students with Disabilities at 471-4321.

Religious Holidays: Please contact the instructor with any issues related to religious holidays. Suitable accommodations will be made.