EECS E6691: Advanced Deep Learning

Columbia University, Spring 2023

Instructor:	Mehmet Kerem Turkcan	**Time:**	W 10:10AM-12:40PM
Email:	mkt2126@columbia.edu	**Location:**	1024 Mudd Building

Course Pages:

1. Courseworks: https://courseworks2.columbia.edu/courses/168932
2. EdStem Forum: https://edstem.org/us/courses/35588/discussion/
3. Columbia Course Directory: https://doc.sis.columbia.edu/#subj/EECS/E6691-20231-001/

Office Hours: After class, or by appointment, or post your questions in the EdStem forum.

TA/CA Office Hours: Announced on Courseworks.

TA: Chengbo Zang.

Description: This is an advanced-level course in which the students study topics in deep learning. It is required that students have previously taken an introductory course in deep learning. The course consists of (i) lectures on state-of-the-art architectural and modeling concepts, (ii) assignments, (iii) exams, and (iv) a final project. The course will address topics beyond material covered in the first course on Deep Learning (such as ECBM E4040), with applications of interest to students. In 2023, the main subject of the lectures will be object detection.

Students entering the course have to have prior experience with deep learning and neural network architectures including Convolutional Neural Nets (CNNs), Recurrent Neural Networks (RNNs), Long Short Term Memories (LSTMs), and autoencoders. They need to have a working knowledge of coding in Python, Python libraries, Jupyter notebook, TensorFlow both on local machines and on Google Cloud, and GitHub or similar code hosting tools. The framework and associated tools that will be the focus of this course are PyTorch and Google Cloud. Students have to be self-sufficient learners and to take an active role during the classroom activities.

There will be a few (3-4) assignments throughout the semester focusing on coding. In the second half of the course, there will be a midterm exam comprised of multiple-choice questions.

Final projects need to be documented in a conference-style report, with code deposited in a GitHub repository. The code needs to be documented and instrumented such that the instructor can run it after a download from the repository. A Google Slides presentation of the project suitable for a poster presentation is required.

Main References: The main resource for the course is "Deep Learning" by Ian Goodfellow, Yoshua Bengio, and Aaron Courville, MIT Press, `https://www.deeplearningbook.org`, 2016.

Prerequisites:

1. Machine Learning (taken previously, or in parallel with this course).
2. ECBM E4040 Neural Networks and Deep Learning, or an equivalent neural network/DL university course taken for academic credit.
3. The course requires an excellent theoretical background in probability, statistics, and linear algebra.

Students are strongly advised to drop the class if they do not have adequate theoretical background and/or previous experience with programming of deep learning models. It is strongly advised (the instructor's requirement) that students take no more than 12 credits of any coursework (including this course and project courses) during the semester while this course is being taken.

Grading Policy:

- Assignments (30%): The course has 4 assignments (homeworks). All assignments are graded. Each student has to complete their own coding tasks and questions, using Python, TensorFlow, Jupyter notebooks, and code management tools.
- Exam (at Week 11, 30%): Students will take an in-class multiple choice exam covering deep learning theory, architecture, and design.
- Final Projects (40%): Students will work on one project, based on contemporary papers or original ideas, in groups of no more than 3. Projects will have to be documented in code and a report.

Course Outline:

Week 1: Introduction to the Course, History and Fundamentals of Object Detection, Evaluation Criteria for Object Detection
Week 2: Conventional Object Detection Methods, Segmentation
Week 3: Historical Segmentation Approaches, Fully Convolutional Networks (FCNs), U-Nets for Segmentation
Week 4: Region Based Convolutional Neural Network (R-CNN), Fast R-CNN, Faster R-CNN, Generative Pre-Training (GPT) Architectures
Week 5: U-net++, ReInspect, Hungarian Algorithm, Multi-Task Network Cascades, Feature Pyramid Networks (FPN), Mask R-CNN, PointRend, Cascade R-CNN
Week 6: Single Shot MultiBox Detector (SSD), YOLO (You Only Look Once), YOLO9000
Week 7: YOLOv3, YOLOv4, Graph Neural Networks, DeepWalk, Word2Vec
Week 8: RetinaNet, YOLOv5, YOLOv6, YOLOv7, YOLOv8, SORT Object Tracking, ByteTrack, BoT-SORT
Week 9: DeepSORT, SMILEtrack, StrongSORT, Vision Transformers
Week 10: Review and Midterm Preparation
Week 11: Midterm
Week 12: Vision Transformers (Continued), DETR, Deformable DETR, Swin Transformers, Segmentation Transformer, Segmenter, Generative Agents, Open Problems in Research
Week 13: Pix2Seq, CLIP, Diffusion Models, ControlNet
Week 14: OWL-ViT, Video Latent Diffusion Models, SegmentAnything

Important Dates:

Assignment #1	February 5, 2023
Assignment #2	March 12, 2023
Project Proposal	March 21, 2023
Midterm	April 5, 2023
Assignment #3	April 29, 2023
Final Project	May 10, 2023