Frontiers in Deep Learning (CSC 266/466) - Syllabus
Spring 2023

Instructor & Lecturer: Prof. Christopher Kanan
4 Credit Hours

Catalog Description: Deep learning has revolutionized many areas of artificial intelligence, including computer vision, natural language understanding, and more. This course reviews some of the latest advancements in deep learning, with a focus on methods that overcome the need for vast amounts of labeled data. Topics covered may include, but are not limited to, self-supervised learning in vision and language, self-training methods (e.g., student-teacher networks), continual learning, foundation models (e.g., Stable Diffusion, GPT-3, CLIP, etc), learning from biased and long-tailed datasets, and theory topics related to generalization in deep learning (e.g., The Lottery Ticket Hypothesis, Deep Double Descent, etc.).

Course Frequency: Offered in Spring

Prerequisites: There are no formal requirements, but students are expected to know Python, linear algebra, calculus, and have basic knowledge of deep learning, e.g, loss functions, optimization, regularization, multi-layer perceptron neural network architectures, etc. Additionally, students should have familiarity with a deep learning toolbox, e.g., PyTorch, Keras, TensorFlow, etc.

Class Location & Time: Monday/Wednesday, 10:25am - 11:40am, Meliora Room 205

No textbook is needed for this course, but you may wish to refer to these resources::
- "Deep Learning" (2016) by Ian Goodfellow, Yoshua Bengio, and Aaron Courville. A free online version is available here: http://www.deeplearningbook.org
- "Deep Learning with PyTorch" (2020) by Eli Stevens, Luca Antiga, and Thomas Viehmann. Available here: https://pytorch.org/assets/deep-learning/Deep-Learning-with-PyTorch.pdf
- There will be readings from other sources.

Slack Link: https://join.slack.com/t/frontiersinde-knx6658/shared_invite/zt-1migl1uzm-IOXDLA5SefwQ31yeQ66vZw

Student Outcomes:
1. Demonstrate understanding of the latest advancements in deep learning.
2. Demonstrate the ability to characterize deep neural network performance.
3. Gain experience deploying deep learning models or in conducting deep learning R&D.

Instructor Contact:

Name: Prof. Christopher Kanan
Office Hours: Mondays 2:30-4:00pm Wegmans 3017 (No office hours on 2/20, 3/6, 3/20, 4/3)
Email Address: ckanan@cs.rochester.edu

Illness Policy: All lecture slides will be posted online. If you are feeling unwell, you are encouraged to not attend class or at the very least wear a well fitted mask.

Teaching Assistants & Graders:

Role	Name	Office Hours	Email
Grad TA	Hakki Motorcu	Tue: 12:20pm - 2:30pm, Wegmans Hall 3209	hmotorcu@ur.rochester.edu
UG TA	Yanchen Wang	Mon: 3pm - 5pm, Wegmans Hall 2nd Floor CSUG Lab	ywang330@u.rochester.edu
UG TA	Sangwu Lee	Thu: 1-3pm, Wegmans Hall 2nd Floor Study Area	slee232@u.rochester.edu

Evaluation and Grading: The final course grade will be weighted as follows:

Homework:	40%
Midterm:	20%
Project:	40%

We will follow standard grading guidelines to assign the percentage into a letter grade. The professor may choose to “curve” the class by giving all students the same number of additional points.

Homework: Your homework submissions must cite any references used (including articles, books, code, websites, and personal communications). All solutions must be written in your own words, and you must program the algorithms yourself. You are responsible for starting them early to ensure that you complete them by the deadline. If you start the day before, you will probably do poorly on the assignment. It takes time to implement, train, and evaluate neural networks. Depending on the hardware and implementation, some problems may take hours or more to train the network. All assignments will be done in Python with PyTorch. Your homework solutions must be typed and output to PDF format. Use LaTeX to write up your answers. Your solutions should include all diagrams, written explanations, code, and program outputs.

Project: You are required to complete a project. For CSC 466, projects should be done individually, but for CSC 266 you may have up to 3 team members. Exceptions to team group limits may be given with strong justification. Your project should be at the frontier of deep learning, but it does not necessarily need to move the frontier forward. You may use the programming language and toolboxes of your choice. Replicating results from a recent paper and comparing it to other works would be a good project. An alternative is to build and rigorously evaluate a real-world application of deep learning. Run your early ideas by Prof. Kanan and other staff via email or in person. Your project should endeavor to go beyond merely supervised learning. The schedule for the project is as follows:

1. **Project Proposal**: The project proposal should clearly state what you plan to do. It should be four pages long (not including references). It should contain a list of three to six milestones and deadlines. You should list the questions the project will address and that will be discussed in the report. You should list what software you will be using or will build upon. Describe the datasets you will use and how will you know if the project is successful. Describe the hypotheses you will test and the related work. The proposal should be a well organized document in continuous English, and it should not be merely an outline. You should be able to reuse much of the text for the final report. It should be submitted as a PDF (under 10MB).

2. **Revised Project Proposal (optional)**: The revised proposal is an opportunity to improve your grade if you fail to do the project proposal effectively. You may submit a revised proposal that takes into account the comments received by the instructor and TA. The new grade will replace the original score, but the maximum score for the revised proposal is 80%.
3. **Project Report**: The project report will describe the project, i.e., what you did and the result. It should be about eight pages long (not including references) and formatted in NeurIPS format. It should be submitted as a PDF (under 10MB).

Policy on Late Work: No credit will be given for the project report if it is turned in late. For the project proposal, a late submission will be treated as submitting a revised project proposal capping the score at 80%. For homework assignments, full points will be awarded only if the assignment is turned in at most one day late. Late homework assignments will be accepted up to 7 days late with a 20% penalty imposed, meaning the highest possible score will be 80% for any assignment that is 2-7 days late. No credit will be given for assignments turned more than 7 days late. An exception to this policy is that no assignments will be accepted after the project report deadline.

Programming Environment: For homework assignments, this course uses Python and PyTorch (Lightning). For the class project, you may use the programming language and framework of your choice, but most of our expertise is in PyTorch.

Academic Honesty and Integrity: All assignments and activities associated with this course must be performed in accordance with the University of Rochester's Academic Honesty Policy. You are expected to read, understand, and follow the policy. Additionally:

- In general, homework is to be completed independently. However, you are encouraged to study together and to discuss information and concepts covered in lecture and the sections with other students. You can give "consulting" help to or receive "consulting" help from such students. However, this permissible cooperation should never involve one student having possession of a copy of all or part of the work done by someone else. Should copying occur, both the student who copied work from another student and the student who gave material to be copied will both automatically receive a zero for the assignment. Penalty for violation of this Code can also be extended to include failure of the course and University disciplinary action.
- Posting homework and project solutions to public repositories on sites like GitHub is a violation of the College's Academic Honesty Policy, Section V.B.2 "Giving Unauthorized Aid."
- During examinations, you must do your own work. Talking or discussion is not permitted during the examinations, nor may you compare papers, copy from others, or collaborate in any way. Any collaborative behavior during the examinations will result in failure of the exam, and may lead to failure of the course and University disciplinary action.

Prior Course Materials: Unauthorized use of course materials from previous semesters (e.g., material you have received from others), is strictly prohibited.

New Course Materials: Course materials (slides, lectures, assignments, etc.) may not be re-distributed or posted elsewhere online. Redistribution of copyright protected material outside this course may be prohibited by law.

Notes on Plagiarism: Plagiarism is a serious offense and is in violation of university policy.

- If you are unsure of what constitutes plagiarism in written documents, a good description can be found here: https://rochester.edu/college/gradstudies/assets/pdf/Plagiarism_Misconduct.pdf

- Plagiarism does not just occur in written documents; it also occurs in code. Many of the algorithms we will code and problems we will solve have been solved by others who have posted code (in various programming languages) online. It is unacceptable (and it is considered plagiarism) to copy code developed by others and submit it as your own. (This includes code that is written by your fellow students!) Even making minor changes, such as changing variable names, function names, formatting, etc., is not enough to allow you to claim your submission as your own because the underlying structure of the code remains unchanged. You may also be in violation if you excessively rely on AI “Co-Pilot” systems to assist you with writing your code.
- If you do consult any online sources of code, you must properly attribute the corresponding sections in your code to their original source, as you would add quotations, footnotes, or references in a written document. The consequences of plagiarism, whether in code or in written documents, are at the discretion of the instructor, and can be as severe as automatic failure of the course.

Academic Accommodations: We are committed to providing reasonable accommodations to students with disabilities. Please see the professor about your required accommodations as early as possible in the term. The University of Rochester respects and welcomes students of all backgrounds and abilities. In the event you encounter any barrier(s) to full participation in this course due to the impact of a disability, please contact the Office of Disability Resources. The access coordinators in the Office of Disability Resources can meet with you to discuss the barriers you are experiencing and explain the eligibility process for establishing academic accommodations. You can reach the Office of Disability Resources at: disability@rochester.edu; (585) 276-5075; Taylor Hall; www.rochester.edu/college/disability.

Course Schedule: The following schedule lists dates for class topics. *The content in this schedule is tentative and subject to change.* It is your responsibility to attend class and to remain informed of any changes that may be announced.

Week	Date	Assignments	Class / Discussion Topics	Presenter
1	1/09		No Class - Semester not started yet	N/A
	1/11	Homework 0 Out	Introduction	Kanan
2	1/16		No Class - MLK Day	N/A
	1/18		Review: Linear Algebra, Feedforward Networks, & Backpropagation	Kanan
3	1/23		Neural Network Zoo	Kanan
	1/25	Homework 0 Due	Lottery Ticket Hypothesis & Deep Double Descent	Kanan
4	1/30	Homework 1 Out	Convolutional Neural Networks	Kanan
	2/01	Homework 2 Out	Autoencoders, RNNs, Seq2Seq	Kanan
5	2/06		Overview of Foundation Models Transformers	Kanan
	2/08		Project Teaming [Prof. Kanan in DC for AAAI]	TBD
6	2/13		Transformers for Language Prompting and Prompt Engineering	Kanan
	2/15	Homework 1 Due (2/17)	Vision Transformers	Kanan
7	2/20	Homework 2 Due (2/21)	Semi-Supervised Learning and Teacher-Student Models	Kanan
	2/22		Self-Supervised Learning Part 1	Kanan
8	2/27		Self-Supervised Learning Part 2	Kanan
	3/01	Proj. Proposal Due 3/3	Midterm	N/A
9	3/06		No Class - Spring Break	N/A
	3/08		No Class - Spring Break	N/A
10	3/13		Self-Supervised Learning Part 3	Kanan
	3/15		Self-Supervised Learning Part 4	Kanan
11	3/20		No Class [Prof. Kanan in Germany for Dagstuhl Seminar on Continual Learning]	TBD
	3/22		No Class [Prof. Kanan in Germany for Dagstuhl Seminar on Continual Learning]	TBD
12	3/27	Homework 3 Out	Writing AI papers	Kanan
	3/29	Revised Project Proposal Due (optional)	Cross-Modal Embeddings	Kanan
13	4/03		Continual Learning	Kanan
	4/05		Multi-Modal Foundation Models	Kanan
14	4/10		Generative Methods Part 1: GANs	Kanan
	4/12		Generative Methods Part 2: Muse and VAEs	
15	4/17		Generative Methods Part 3: Diffusion, Stable Diffusion, DALL-E v2, DreamBooth	
	4/19	Project Report Due (4/21)	Generative Flow Networks – https://milayb.notion.site/GFlowNet-Tutorial-919dcf0a0f0c4e978916a2f509938b00 Memory Augmented Networks Last Lecture & Course Wrap-up	Kanan
16	4/24		Project Presentations (extended class possible)	Students
	4/26		Project Presentations (extended class possible)	Students
	5/1	Homework 3 Due		