CS 523 Fall 2025 Deep Learning: Syllabus and Schedule

Course Description:

This course is an introduction to deep learning, a branch of machine learning concerned with the development and application of modern neural networks. Deep learning algorithms extract layered high-level representations of data in a way that maximizes performance on a given task. For example, when asked to recognize faces, a deep neural network may learn to represent image pixels first with edges, followed by larger shapes, then parts of the face like eyes and ears, and, finally, individual face identities. Deep learning is behind many recent advances in AI, including Siri's and Alexa's speech recognition, Facebook's algorithm and self-driving cars. We will cover a range of topics from basic neural networks, convolutional and recurrent network structures, deep unsupervised and reinforcement learning, and applications to problem domains like speech recognition and computer vision. **Prerequisites**: a strong mathematical background in calculus, linear algebra, and probability & statistics, as well as prior coursework in machine learning and programming experience in Python.

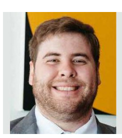
Lecture:

CASCS 523 A1
T/Th 5:00-6:15pm, CAS 522

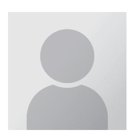
Staff:

Kate Saenko
Instructor
office hours: TBD in CDS 845

Dallin Gordon
Teaching Fellow
Office hours: TBD



Andrew Wood
Instructor
Office hours: TBD in CDS 826



Aditya Bhandari
Teaching Assistant
Office hours: TBD

How to Contact Us: Please use Piazza for all communication; if your question is only directed to the instructors, please make a post to "Individual Student(s) / Instructor(s)" and select "Instructors".
Discussion: Piazza https://piazza.com/class/mf1b5i3hgql1bk/
We will be using piazza for online discussions, questions, and to post assignments.
Gradescope: https://www.gradescope.com/courses/1111626 for submitting and grading assignments. Code: **B3WE7D**

Schedule*

	Topic	Details	Homework
Tue Sep 2	Course overview	What is deep learning? DL successes; syllabus & course logistics.	Pa0 out
Thu Sep 4	Math/ML Review I	Probability, distributions, maximum likelihood, empirical risk minimization.	
Tue Sep 9	Math/ML Review II	Generalization, train/validation/test splits, stability,	Wa0 out

		stochastic gradient descent.	
Thu Sep 11	Neural network basics I: Universality	Classification and regression tasks, perceptron, universal approximation.	Pa0 due Pa1 out
Tue Sep 16	Neural network basics II: Dense layers	MLP, dense layers, activation functions, surrogate loss functions, softmax, and compression.	
Thu Sep 18	Neural network basics III: Learning	Automatic differentiation and backpropagation, matrix derivatives.	Wa0 due Wa1 out
Tue Sep 23	CNNs I	Convolutional neural networks, including AlexNet, VGG, and Inception.	
Thu Sep 25	CNNs II	Modern Conv Nets, ResNet	Pa1 due Pa2 out
Tue Sep 30	Midterm 1	Material Through CNN II	SCC Info
Thu Oct 2	RNNs	Recurrent neural networks; sequence modeling; backpropagation through time; vanishing/exploding gradient problem; gradient clipping, LSTM, GRU, LRU.	Wa1 due Wa2 out
Tue Oct 7	Transformers I: Attention	Embeddings, word vectors, self-attention, transformers.	
Thu Oct 9	Transformers II	Pretraining, masked token modeling task, few-shot learning.	Pa2 due Pa3 out
Tue Oct 14	NO CLASS	MONDAY SCHEDULE FOR INDIGENOUS PEOPLE'S DAY	
Thu Oct 16	Deep Unsupervised Learning I	Autoencoders	Wa2 due Wa3 out
Tue Oct 21	Deep Unsupervised Learning II	Variational Autoencoders	
Thu Oct 23	Deep Unsupervised Learning III	Diffusion Models	Pa3 due Pa4 out
Tue Oct 28	Self-Supervised Learning	self-supervised learning (slides from this tutorial)	
Thu Oct 30	Training Strategies I	Mini-batching, regularization, adversarial examples, dropout, batch norm, layer norm.	Wa3 due Wa4 out
Tue Nov 4	Midterm 2	Material through Training Strategies I	
Thu Nov 6	Training Strategies II	Momentum and acceleration, physical interpretation of accelerated gradient descent, stochastic gradients and variance.	Pa4 due
Tue Nov 11	Training Strategies III	Adaptive gradient methods: adagrad, adam, Lars/Lamb, and large batch sizes. RMSprop	
Thu Nov 13	TBD		Wa4 due
Tue Nov 18	Reinforcement Learning I		
Thu Nov 20	Reinforcement Learning II		Project Status Report Due

			Template
Tue Nov 25	Applications: Vision	Applications to computer vision	
Thu Nov 27	NO CLASS	THANKSGIVING BREAK	
Tue Dec 2	Applications: Audio	Keyword Spotting, Audio Synthesis, Automatic Speech Recognition	
Thu Dec 4	Large Language Models	GPT, BERT	
Tue Dec 9	Project Presentations	Time and location TBD	
Thu Dec 11	Project Presentations	Time and location TBD	Project report due
END OF SEMESTER			

*schedule is subject to change

Syllabus

Course Prerequisites

This is an upper-level undergraduate/graduate course. All students should have the following skills:

- Calculus, Linear Algebra
- Probability & Statistics
- Ability to code in Python
- Background in machine learning (e.g. CS 541, CS 542, CS440, CS365, EC 414, EC 503)

Textbook

There is no required textbook for the course. The recommended textbook is

- Christopher M. Bishop. Deep Learning: Foundations and Concepts, 2024

Other recommended supplemental textbooks:

- Ian Goodfellow, Yoshua Bengio, Aaron Courville. Deep Learning. MIT Press, 2016.
- Aston Zhang, Zack C. Lipton, Mu Li, and Alexander Smola. Dive into Deep Learning, 2020.
- Duda, R.O., Hart, P.E., and Stork, D.G. Pattern Classification. Wiley-Interscience. 2nd Edition. 2001.
- Theodoridis, S. and Koutroumbas, K. Pattern Recognition. Edition 4. Academic Press, 2008.
- Russell, S. and Norvig, N. Artificial Intelligence: A Modern Approach. Prentice Hall Series in Artificial Intelligence. 2003.
- Bishop, C. M. Neural Networks for Pattern Recognition. Oxford University Press. 1995.
- Hastie, T., Tibshirani, R. and Friedman, J. The Elements of Statistical Learning. Springer. 2001.
- Koller, D. and Friedman, N. Probabilistic Graphical Models. MIT Press. 2009.

Recommended online courses

- http://cs231n.stanford.edu/ CS231n: Convolutional Neural Networks for Visual Recognition
- http://web.stanford.edu/class/cs224n/ CS224n: Natural Language Processing with Deep Learning
- http://rll.berkeley.edu/deeprlcourse/ CS 294: Deep Reinforcement Learning
- http://distill.pub/ Very nice explanations of some DL concepts

Deliverables/Graded Work

There will be five written assignments (WA), five programming assignments (PA), two written in-class midterms and a final project. The project will be done in teams and will have several deliverables including a proposal, progress update(s), final report and a final in-class poster presentation. There will be opportunities for extra credit assignments. The course grade consists of the following:

- Homework assignments 35%
- Project (including all components) 35%
- Midterms 30% (15% M1 + 15% M2)

Late Policy

Late work will incur the following penalties

- Project deliverables: 20% off per day up to 2 days
- Homework 20% off per day, up to 3 days, then zero

Project

Projects are done in teams of 2-3 students. Please talk to the instructors if you'd like a different number of teammates.

Homework Assignments

There will be five graded assignments, each consisting of a written assignment (WA) and a programming assignment (PA). We will also provide ungraded tutorials to guide you through practical implementation with Pytorch in preparation for the final project. The following is a tentative plan for assignments and tutorials:

	Topic	Written	Programming	Tutorial (ungraded)
0	Pre-requisites	Probability, MLE LASSO and ridge regression	Single-variable scalar autograd LASSO and ridge regression	
1	MLP	Regression and Priors Universal approximator for RELU	Dense layer autograd Implementing a simple MLP Approximate RELU	Intro to Pytorch
2	CNN	Convolutional layers	Convolutional and deconvolutional layers Put together a CNN with conv and dense layers	SCC cluster intro Training/fine tuning CNNs Neural style transfer learning
3	RNN/Transformer	Architecture comparison	Recurrent layers Attention layers Self attention	Neural Machine Translation
4	Diffusion	Diffusion loss function	Diffusion layer Batch normalization dropout	Transformers for Music Generation Conditional Clothing Generation w/ Diffusion

Software/Hardware

Programming assignments and projects will be developed in the Python programming language. We will also use the PyTorch deep learning library for some homeworks and for the project. Students are expected to use the [Shared Computing Cluster (SCC)] and/or their own machines to complete work that does not require a GPU. For the projects, we will provide GPU resources.

Academic Honesty Policy

The instructors take academic honesty very seriously. Cheating, plagiarism and other misconduct may be subject to grading penalties up to failing the course. Students enrolled in the course are responsible for familiarizing themselves with the detailed BU policy, available here. In particular, plagiarism is defined as follows and applies to all written materials and software, including material found online. Collaboration on homework is allowed, but should be acknowledged and you should always come up with your own solution rather than copying (which is defined as plagiarism):

> **Plagiarism**: Representing the work or ideas of another* as one's own and/or using another's work or ideas without appropriately crediting the source. Plagiarism includes, but is not limited to, the following: copying the answers of another student on an examination; copying or restating the work or ideas of another person/persons or **artificial intelligence software** in any oral or written work (printed or electronic) without appropriately citing the source; using visuals, audio, or video footage that comes from another source (including work done by another student) without permission and/or acknowledgement of that source; and collaborating with someone else in an academic endeavor without acknowledging their contribution. Plagiarism can consist of acts of commission (appropriating the words or ideas of another as one's own), or omission (failing to acknowledge/document/credit the source or creator of words or ideas).
>
> *"Another" may refer to anything that can be a source of information or work product, including (but not limited to) individuals, books, online sources, academic journals, and software/programs (e.g., artificial intelligence software/programs).