# Introduction to Deep Learning (24-788)

## *Instructor: Amir Barati Farimani*

**Class Time and Location:**
Tuesdays & Thursdays: 4:00 - 5:50 pm
Location: WEH 7500

**Links:**
**Canvas**
**Piazza**
**Gradescope**

**Course Description:**

This course provides an introduction to deep learning. We will learn about the basics of deep neural networks and their applications to different tasks in engineering. Students will be able to apply deep learning to a variety of artificial intelligence tasks pertinent to different engineering problems. Applications of deep learning in Mechanical, chemical, biological, electrical, and material engineering will be discussed.

| **Course Staff:** | **Office hours** |
|---|---|
| Amir Barati Farimani | Tuesday 1:00-2:00 pm, 212 SH |
| Zhefan Xu | Monday 3:00-4:00 pm, TBD |
| Mangalam Sahai | TBD, TBD |
| Jerry Li | Thursday 9:30-10:30 am, TBD |
| Arvind Car | TBD, TBD |
| Selam Gano | TBD, TBD |
| Reid Graves | TBD, TBD |

**Course Discussion Forum:**
This course uses **Piazza**, a forum-style platform where students can post questions and receive answers from both TAs and fellow students. If you have any questions on the course material, please post them on Piazza. When answering questions from fellow students on Piazza, keep in mind the course's academic integrity policy (ie, do not copy and paste your entire solution). You can also send private messages to specific TAs or all instructors that other students do not see. **Emails to TAs and the instructor will be ignored**.

**Topics (tentative):**

- Introduction to Deep Learning and its application
- Neural Networks
- Convolutional Neural Networks (CNN)
- Training and Testing CNN
- Interpretability of Deep Learning
- Graph Convolutional Neural Networks (GCNN)
- Recurrent Neural Networks (RNN)
- Solving Engineering problems using Deep Learning

**Textbooks and Readings:**

The lectures in this course were compiled from different sources and naturally there is no single textbook that covers all of the topics we will discuss in this course. The following textbook is recommended for the course subjects:

- Deep Learning by Ian Goodfellow, 2016: Available online

### Prerequisites:

- (24-787) Machine Learning and Artificial Intelligence for Engineers (or)
- (10-601/701) Introduction to Machine Learning

### Grading:

Course grades will be **based 80% on assignments, 10% on quizzes and 10% on the exam**.

### Grading Scale:

93%-100% = A
90%-93% = A-
87%-90% = B+
83%-87% = B
80%-83% = B-
77%-80% = C+
70%-77% = C
60%-70% = D
0%-60% = R

### Assignments:

All of the assignments in this course involve writing computer programs. In a typical assignment, you will implement a deep learning model from the lecture and train it on a small dataset. You will be graded on how well-written your code is, and the model's performance on the specified learning task. Therefore you should carefully implement, test, and debug each program. Remember, just because there are no error messages, it does not mean the program works as it should. In addition to writing correct code that functions properly as instructed, you also need to annotate and comment the code to be readable and understandable by others. Unorganized and dirty code will not be graded generously.

Some assignments may also include short questions to test your understanding of the concepts or what you learned from the assignment. You are not allowed to copy the answers for these questions from anyone else or any Generative AI tool like ChatGPT. If your response is suspected by a TA to be pasted from an AI tool and is assigned a high likelihood of being AI generated by an AI detector, you will not get the point for that question.

You will be given one week for each assignment. Please start the assignments early, as it is unlikely that you will be able to complete the assignments the night before they are due.

### Python Programming:

In this course, we will use Python for all assignments. You should be familiar with the basics of Python programming (lists, dictionaries, etc), basic object-oriented programming (working with classes), and NumPy. Python is a great general-purpose programming language on its own, and with the help of a few popular libraries (NumPy, matplotlib, scikit-learn, PyTorch, torch_geometric, etc), it becomes a powerful environment for scientific computing, machine learning, and deep learning. In this class, you will learn to use PyTorch to implement and train deep learning models.

### Assignment Submission:

Assignments will be available on Canvas and should be submitted via Gradescope. The instructions and the starter code for the assignments are contained in Jupyter Notebooks which you have to complete for your submission. There may also be a utility python file containing helper functions and dataset files that you will not need to modify or include in your submission. The last cell of the notebook has the code to create the zip file for the autograded part of the assignments, which you will submit to Gradescope.

After the zip file is created, ensure that all the intended files are included as specified at the beginning of the notebook.

♋ For most assignments, you also have to save the completed notebook as a PDF and submit it to Gradescope to be manually graded. To save your completed Jupyter Notebook as a PDF, use the `jupyter nbconvert --to pdf HW.ipynb` command in the terminal. You can run terminal commands in code cells of Jupyter Notebooks by including an exclamation point, like `!jupyter nbconvert --to pdf HW.ipynb`. This ensures that the PDF is formatted properly and no code or text is left out. Creating the PDF by printing the notebook from a browser is also possible, but it has the risk of missing out long lines of code, so do it at your own risk. If a manually-graded section is not visible in your PDF submission, you will lose some of the corresponding points.

♋ Do not change the name of the files inside the zip file, as this will prevent the autograder from functioning as intended.

♋ Failure to comply with the assignment instructions and submission guidelines will result in grade penalties that will be proportional to the severity of your non-compliance.

**Late Submission Policy:**

Assignments are expected to be completed by the due date. Assignments submitted more than 48 hours after the due date will not be accepted. You will have a total of **four free late (calendar) days** to use as you see fit. Once these late days are exhausted, any homework turned in late will be **penalized 40% per late day**. **Every 24 hours or part thereof that a homework is late uses up one full late day**.

If you have a disability and are registered with the Office of Disability Resources, I encourage you to use their online system to notify me of your accommodations and discuss your needs with me as early in the semester as possible. I will work with you to ensure that accommodations are provided as appropriate. If you suspect that you may have a disability and would benefit from accommodations but are not yet registered with the Office of Disability Resources, I encourage you to contact them at `access@andrew.cmu.edu`.

**Statement of Support for Students' Health and Well-being:**

Take care of yourself. Do your best to maintain a healthy lifestyle this semester by eating well, exercising, avoiding drugs and alcohol, getting enough sleep and taking some time to relax. This will help you achieve your goals and cope with stress. If you or anyone you know experiences any academic stress, difficult life events, or feelings like anxiety or depression, we strongly encourage you to seek support. Counseling and Psychological Services (CaPS) is here to help: call 412-268-2922 and visit `http://www.cmu.edu/counseling/`. Consider reaching out to a friend, faculty, or family member you trust for help getting connected to the support that can help.

*Tab. 1:* Detailed Syllabus

| Date | Session | Time | Detail |
|---|---|---|---|
| Tue Jan 14, 2025 | Session 1 | 4:00pm to 5:50pm | Introduction to Deep Learning |
| Introduction to Deep Learning, Applications of DL in vision, natural language processing (NLP), transport phenomena, fluid mechanics, chemical engineering, material science and health. History and cognitive basis of neural computation, Perceptrons and Multi-layer Perceptrons. Course Logistics, Learning Objectives, Grading and Deadlines | | | |
| Thu Jan 16, 2025 | Session 2 | 4:00pm to 5:50pm | Universal Function Approximation |
| Neural Networks as Universal Function Approximators, Empirical Risk Minimization, Divergence function, Inputs and Outputs to Networks, Regression and Classification within NN | | | |
| Fri Jan 17, 2025 | Recitation 1 | 4:00pm to 5:00pm | Tensors and Modules in PyTorch |
| Tue Jan 21, 2025 | Session 3 | 4:00pm to 5:50pm | Neural Nets and Activation Functions |
| Recap on Empirical Risk Minimization, Network Capacity, Different Divergence Function, Inputs and Outputs and modality of Deep Learning data. Simple Feed Forward NN Algorithm implementation in Pytorch | | | |
| Thu Jan 23, 2025 | Session 4 | 4:00pm to 5:50pm | Backpropagation and Optimization |
| Forward Propagation, Partial Derivatives in Backpropagation, Examples of Back-propagation, Optimization of Weights, Introduction to Stochastic Gradient Descent | | | |

| | | | |
|---|---|---|---|
| Fri Jan 24, 2025 | Recitation 2 | 4:00pm to 5:00pm | Training neural networks in PyTorch |
| Tue Jan 28, 2025 | Session 5 | 4:00pm to 5:50pm | Stochastic gradient descent (SGD) and Optimization |
| Overfitting and regularization, Choosing Loss Function, Batch normalization, Adagrad, Adadelta, RMSProp, ADAM optimization, Downsides of SGD, Momentum SGD, Nesterov Accelerated Gradient (NAG), AdaGrad and adaptive learning rate | | | |
| Thu Jan 30, 2025 | Session 6 | 4:00pm to 5:50pm | CNN (part1) |
| Motivations for Convolutional Neural Networks (CNN), Convolution Operation, Convolution Filter, Local Feature Extraction, Volume Convolution, Strides and Filter size, Optimization of CNN | | | |
| Fri Jan 31, 2025 | Recitation 3 | 4:00pm to 5:00pm | Regularization in PyTorch |
| Tue Feb 4, 2025 | Session 7 | 4:00pm to 5:50pm | CNN (part2) |
| Recap on Convolution, Convolution Layers, RGB Channels, Padding, NFSP rule, Pooling (max, average, sum), Why Pooling?, Pooling Layers, Downsides of Pooling, Capsule Network, Architecture of Conv Layers, Conv and FC Layers Connection, Parameter sharing, Parameter (weight) Compression in ConvNet, CNN in vision, Challenges in vision (illumination, deflection, occlusions), Applications of CNNs (image classification, Segmentation, Labeling, Captioning, Counting, Pose Prediction, etc.), What if data is not image like?, Fluid mechanic Example and idea of channels. | | | |
| Thu Feb 6, 2025 | Session 8 | 4:00pm to 5:50pm | Training and Testing CNN |
| Training CNN/NN, Saturated Non-linearity, Internal Covariate Shift, Batch Normalization (BN), Benefits of BN, BN algorithm and shift/scale, Why BN works?, Hessian-Covariance accumulation, Xavier initialization, derivation of Xavier init., cost and accuracy per initialization scheme, Training with shared parameters Coding up CNN, MNIST CNN in PyTorch | | | |
| Fri Feb 7, 2025 | Recitation 4 | 4:00pm to 5:00pm | CNN + Training |
| Tue Feb 11, 2025 | Session 9 | 4:00pm to 5:50pm | Different CNN architectures |
| AlexNet, VGG16, GoogLeNet, ResNet, MobileNet, EfficientNet, ConvNext | | | |
| Thu Feb 13, 2025 | Session 10 | 4:00pm to 5:50pm | Graph Convolutional Neural Networks (GCNN) |
| Message passing on graph, Convolution on graph, GraphSAGE | | | |
| Fri Feb 14, 2025 | Recitation 5 | 4:00pm to 5:00pm | GNN + Training |
| Tue Feb 18, 2025 | Session 11 | 4:00pm to 5:50pm | Recurrent Neural Networks (RNN part1) |
| Motivation for Recurrent Neural Network (RNN), sequential models, simple RNN, hidden states, RNN examples, Cross-entropy loss for RNNs, Training RNNs, Vanishing and exploding gradient problem, Gradient clipping | | | |
| Thu Feb 20, 2025 | Session 12 | 4:00pm to 5:50pm | (RNN Part 2) |
| Recap of RNN, Long Short Term Memory (LSTM): Gated Recurrent Unit (GRU), reset gate, update gate, Long-short term Memory (LSTM), input, forget, new and output gates, Sequence Prediction | | | |
| Fri Feb 21, 2025 | Recitation 6 | 4:00pm to 5:00pm | RNN + LSTM |
| Tue Feb 25, 2025 | Session 13 | 4:00pm to 5:50pm | Dimensionality Reduction in Deep Learning |
| The Simplest Autoencoder, U-Net, Convolutional Auto-encoders, NN used to perform linear or nonlinear PCA | | | |