CPT_S 434/534: Neural Network Design And Application

Instructor: Yan Yan

Spring 2025

Class Hours: Tue/Thu 7:45AM - 9:00AM

Classroom: CLEV 30

Instructor Office Hours: Tue 11:30AM - 12:30PM
Sign up on Calendly

Office: EME 123

TA: TBA
TA Office Hours: TBA

TA office: TBA

Contact:
Send a Message on Canvas to ALL Instructor and TAs

Course schedule: See this Google Sheet or Section 7

Materials/slides are on Instructor's homepage

Gest TA: Yuanjie Shi (Technical Consultation for Course Projects)
Office Hour Time: 11:30AM-12:30PM
Office Hour Date: every other Tuesday
(from 1/14, except 2/25 and 3/11)
Office Hour Location: DANA 114
Sign up on Calendly

---

1 Course Description

This course provides an introduction to deep neural networks (DNNs), a key component in modern machine learning. It includes both *theoretical* foundations of basic machine learning, and hands-on *implementation* of deep neural networks for some applications.

This course starts with basic concepts and components in statistical machine learning to shed light on the answers to these questions: (i) *what is learning*, (ii) *what can be learned*, (iii) *how to learn it*, (iv) *how well we can learn it*, and (v) *what machine learning can do*.

Then we will cover the motivation of DNNs, as modern machine learning methods, over traditional machine learning algorithms in the viewpoint of *representation learning*, an important task of deep learning. As a key type of machine learning method, neural networks themselves have unique properties. We will discuss (i) *general frameworks and typical components* in DNNs, (ii) the *computational challenges* and (iii) *key principles* to learn DNNs.

After that we will detail several common types of neural networks that are commonly used for different modalities of data or learning targets, such as (i) convolutional neural networks (CNNs)

for image data, (ii) recurrent neural networks (RNNs) for sequence data, (iii) graph neural networks (GNNs) for graph data and (iv) generative adversarial networks (GANs). As introducing these well-established networks, we will also cover some recent open research problems in deep learning such as diffusion models and representation learning.

2 Course Sections

- Machine learning [ML]
  - Basic concepts: general ideas, terminologies, examples.
  - PAC learning: formal definition of what is learning, what can be learned, how well we can learn.
  - ERM and MLE: common ways to build/interpret a machine learning model.
- Neural networks [NN]
  - Basic concepts: general ideas, terminologies, examples.
  - Composition of functions: what is neural network structure
  - Units in NN: commonly used functions for building a neural network
  - BP: how to optimize the learning problem in NN.
  - MLP Softmax: a real example of learning a neural network Softmax classifier with BP.
- Convolutional neural networks [CNN]
  - Convolutional operation on image: how the convolutional operation works and what is the output.
  - CNN design: typical Convolutional layers and properties.
  - CNN architecture: the history of CNN architecture design and properties (advantages and disadvantages).
  - Practical tricks for building CNNs: normalization, batch normaltization, data augmentation, dropout, weight decay, pre-trained model, stagewise training.
- Recurrent neural networks [RNN]
  - Sequence data: properties, motivation for recurrent networks.
  - Design RNNs: typical way to build RNNs, character-level language model.
  - Design LSTM: from conventional RNN cell to LSTM cell.
  - RNN architecture: more complicated RNN architectures: bi-directional RNN, Seq2seq RNN.
  - Attention mechanism: architecture design.
  - Transformers (a neural network architecture, not a movie).
- Graph neural networks [GNN]

  - Graph data in ML: from i.i.d. data to connected (non-i.i.d.) data.
  - Graph representation: how to describe a graph.
  - GNN: how to aggregate neighbors in a neural networks.
  - Oversmoothing in GNN: a key challenge.

- Generative adversarial networks [GAN]

  - Adversarial ML: definition and properties.
  - Generative model: definition and properties.
  - GAN: how to generate synthetic data.
  - Diffusion models.

3 Course Objectives

- Understand machine learning basics: the pipeline of machine learning, what can be learned, how to learn, and statistical/optimization/computational challenges
- Understand the benefits of DNNs for learning representations from raw data over traditional machine learning algorithms, and the computational/statistical challenges of learning DNNs.
- Understand the key principles/ideas for efficiently learning parameters of DNNs from data.
- Understand the structures of different types of DNNs, e.g., feedforward neural networks, CNNs, RNNs and GNNs, and how they can handle target data types.
- Able to apply DNNs to solve supervised learning problems over images, text, and graphs.
- Able to read research papers in deep learning, understand the issues raised by them and catch their ideas of proposed solutions.

4 Textbooks

- Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. "Foundations of Machine Learning". MIT Press, Second Edition, 2018. Online: https://cs.nyu.edu/~mohri/mlbook/.
- Ian Goodfellow, Yoshua Bengio, and Aron Courville. "Deep Learning". MIT Press. Online: https://www.deeplearningbook.org/.
- Aston Zhang, Zachary C. Lipton, Mu Li, Alexander J. Smola. "Dive into Deep Learning". Online book with code: https://d2l.ai/.

5 Course Materials

Lecture slides can be found on Instructor's homepage. Assignments can be found on Canvas.

6 Grading Policy

6.1 Assignments (60% grade)

- HW1: ML, code + written, 10% of final grade
- HW2: NN, code + written, 10% of final grade
- HW3: CNN, code + written, 10% of final grade
- HW4: RNN, code, 10% of final grade
- HW5: GNN, code, 10% of final grade
- HW6: GAN, code, 10% of final grade

All coding assignments will be based on Python, Pytorch and Jupyter notebook. This course focuses on neural networks, and does not cover the particular topics about how to program in Python and how to use Pytorch and jupyter notebook. However, there are many good resources as guidelines for learning them. For example, a tutorial for Python and numpy provided by CS231n from Stanford University, Recitations and Bootcamps provided by 11-785 from CMU, etc.

When submitting code assignments:

For coding assignments, it is extremely important to save your running records and make sure they can be displayed in your submissions. Each time when submitting, double check you have uploaded (i) a .ipynb file for Jupyter notebook that can be re-produced by TA, and (ii) a .pdf or .html file converted from the .ipynb file that tracks all your running records to show that you have completely finished this assignment. Note that uploading these 2 files are sufficient, and zipping the 2 files is NOT needed.

6.2 Mid-Term Quiz (10% grade)

- Format: Canvas Quiz, online, *2-hour* time limit. Open book: textbooks and slides can be used, but cannot search questions/answers on internet.
- Types of questions: multiple choice (choose one answer), multiple answers (choose at least one answer), ordering, True/False, etc.
- Number of questions: 24.
- Sections covered: ML, NN, CNN.
- Available time window: 8:00AM - 7:59PM, Feb 20th, 2025, but you can only submit once.
- Grading method: Canvas auto grading.

6.3 Course Project (25% grade)

Course project is a team work. There can be *one, two or three* (up to three) students in each team. For CPT_S 434, there are 3 milestones for each course project (project proposal, project midterm report and project final report), while for CPT_S 534, there are 5 milestones for each course

project (project proposal, project midterm report, midterm presentation, project final report and final presentation). The grading systems for CPT_S 434 and CPT_S 534 is in Section 6.3.7.

The key information and required component of each milestone is elaborated in the following sections.

6.3.1 Project Proposal (Required for CPT_S 434 and CPT_S 534)

A project proposal (6% grade for CPT_S 434, 4% grade for CPT_S 534) is used to: (i) determine team members, (ii) decide what problem you are going to investigate, and (iii) submit a written proposal to describe your idea and plan.

Below is a list of required information for the project proposal submission:

- Due date: Feb 20th, 2025.
- Length limit: 2 pages (title, abstract, main text) + unlimited reference page + unlimited appendix page.
- Format: PDF (exported from Microsoft Word or compiled from LaTeX, e.g., a LaTeX template).
- What should be included:
  - Title: give a name to the project to highlight the significance, what contributions this project can have. The title should be short and relevant to your project.
  - Team members and student IDs. You still need to ensure that all your team members join the correct group on Canvas (see how to self sign up and join a group). Otherwise, any team member that is not included in the group will NOT receive credit.
  - Abstract: a paragraph that introduce key points, e.g., (i) why you choose this problem and this project, (ii) what unique advantage your project will have, and (iii) a plan of the key steps for finishing this project, or anything else.
  - Introduction: a detailed explanation/elaboration of Abstract on the key points, e.g., one paragraph to elaborate one key point.
  - Literature review: any references that are important and relevant to the project, e.g., some techniques in the project that are inspired by other people's work including tutorials, Git repository, research publication, blogs, etc. Introduce more about what inspired the project. For CPT_S 434, you will need to cite at least 2 references or papers. For CPT_S 534, you will need to cite at least 4 references or papers.
  - Technical plan: a detailed elaboration of techniques used in this project (based on the techniques in Section "Literature review", if any). This plan can be described by a flowchart or any other figures.
  - Reference: if anything inspired your project or is relevant to your project, a citation is required. As required in the section of "Literature review", for CPT_S 434, you will need to cite at least 2 references or papers, while for CPT_S 534, you will need to cite at least 4 references or papers.

6.3.2 Midterm Report (Required for CPT_S 434 and CPT_S 534)

A midterm report (9% grade for CPT_S 434, 6% grade for CPT_S 534), after deciding the project by the course project proposal, is used to start and summarize the project of each team. A midterm report is required to elaborate how the project is going.

Below is a list of required information for the project midterm report submission:

- Due date: Mar 20th, 2025.
- Length limit: 6 pages (title, abstract, main text) + unlimited reference pages + unlimited appendix pages. The midterm report should be built on the proposal submission. If necessary, some contents and texts originally written in the proposal should be revised/updated in the midterm report as the project progresses.
- Format: PDF (exported from Microsoft Word or compiled from LaTeX, e.g., a LaTeX template).
- What should be included:
  - Title: give a name to the project to highlight the significance, what contributions this project can have. The title should be short and relevant to your project.
  - Team members and student IDs. You still need to ensure that all your team members join the correct group. Otherwise, any team member that is not included in the group will NOT receive credit.
  - Abstract (**UPDATED**): a paragraph that introduce key points, e.g., (i) why you choose this problem and this project, (ii) what unique advantage your project will have, and (iii) a plan of the key steps for finishing this project, or anything else. If there is anything that should be changed, you may update it in Abstract.
  - Introduction (**UPDATED**): a detailed explanation/elaboration of Abstract on the key points, e.g., one paragraph to elaborate one key point. If there is anything that should be changed, you may update it in Introduction.
  - Literature review (**UPDATED**): any references that are important and relevant to the project, e.g., some techniques in the project that are inspired by other people's work including tutorials, Git repository, research publication, blogs, etc. Introduce more about what inspired the project. For CPT_S 434, you will need to cite at least 2 more references or papers. For CPT_S 534, you will need to cite at least 4 more references or papers.
  - Technical plan (**UPDATED**): a detailed elaboration of techniques used in this project (based on the techniques in Section "Literature review", if any). This plan can be described by a flowchart or any other figures. *As you start your project, you may find some part in the plan should be revised/updated. You are highly encouraged to update and improve your plan towards the final milestone.*
  - Intermediate results (**NEW**): following the plan, what has been done in the project? Elaborate what the team tried for finishing the plan, and what observation and knowledge the team acquired so far. Both theoretical analysis and experimental result can be regarded as the intermediate results.

- Future work (**NEW**): to ensure that the proposed plan can be done, what else work remaining for this project? Elaborate what the team plans to try. This section may modify the plan originally proposed in proposal, if any new understanding is established and some adjustment is necessary.
- Reference (**UPDATED**): if anything inspired your project or is relevant to your project, a citation is required. As required in the section of "Literature review", for CPT_S 434, you will need to cite at least 2 more references or papers, while for CPT_S 534, you will need to cite at least 4 more references or papers.

Note that the midterm report can re-use materials from the project proposal. New sections, i.e., intermediate results and future work must be included in the midterm report.

6.3.3 Midterm Presentation (CPT_S 534 Required, CPT_S 434 Optional)

Project midterm presentations (3% grade for CPT_S 534) take place during week 12, i.e., Mar 25th and Mar 27th, 2025. Please present your current project and summarize your latest report in 5 minutes, including the motivation, problem setup, challenges, the existing methods, your new method and new results, etc.

Check the "Midterm presentation list" tab about the detailed arrangement on the Google Sheet for the course schedule.

6.3.4 Final Report (Required for CPT_S 434 and CPT_S 534)

A final report (10% grade for CPT_S 434, 8% grade for CPT_S 534) is a report summarizing all the work a team has done, including theoretical or empirical results, which serves as a conclusion for your teamwork.

Below is a list of required information for the project final report submission:

- Due date: Apr 15th, 2025.
- Length limit: 8 pages (title, abstract, main text) + unlimited reference page + unlimited appendix page.

  Format: PDF (exported from Microsoft Word or compiled from LaTeX, e.g., a LaTeX template).

  What should be included:

  - Title: give a name to the project to highlight the significance, what contributions this project can have. The title should be short and relevant to your project.
  - Team members and student IDs. You still need to ensure that all your team members join the correct group. Otherwise, any team member that is not included in the group will NOT receive credit.
  - Abstract (**UPDATED**): a paragraph that introduce key points, e.g., (i) why you choose this problem and this project, (ii) what unique advantage your project will have, and (iii) a plan of the key steps for finishing this project, or anything else. If there is anything that should be changed, you may update it in Abstract.

- Introduction (**UPDATED**): a detailed explanation/elaboration of Abstract on the key points, e.g., one paragraph to elaborate one key point. If there is anything that should be changed, you may update it in Introduction.
- Literature review: any references that are important and relevant to the project, e.g., some techniques in the project that are inspired by other people's work including tutorials, Git repository, research publication, blogs, etc. Introduce more about what inspired the project. For CPT_S 434, you will need to cite at least 4 references or papers totally. For CPT_S 534, you will need to cite at least 8 references or papers totally.
- Technical plan (**UPDATED**): a detailed elaboration of techniques used in this project (based on the techniques in Section "Literature review", if any). This plan can be described by a flowchart or any other figures. *In the final milestone, this plan should be finalized and matched with your code implementations.*
- Complete results (**NEW**): following the plan (or the modified plan after the progress report), what has been done in the project? Elaborate what the team tried for finishing the plan, and what observation and knowledge the team acquired so far. Both theoretical analysis and experimental result can be regarded as the results. *You are highly encouraged to include as many figures and tables as possible to elaborate your findings in this course project.* For example, in this paper, there are 16 figures and 3 tables in the main paper (first 11 pages). Moreover, you can find additional 25 figures and 21 tables in the Appendix.
- Future work after this course (**NEW**): given what the team tried and learned from the project, is there any idea in the future that can be explored further?
- Future work (**NEW**): to ensure that the proposed plan can be done, what else work remaining for this project? Elaborate what the team plans to try. This section may modify the plan originally proposed in proposal, if any new understanding is established and some adjustment is necessary.
- References: if anything inspired your project or is relevant to your project, a citation is required. For CPT_S 434, you will need to cite at least 4 references or papers totally, while for CPT_S 534, you will need to cite at least 8 references or papers totally.

Note that the final project report can re-use materials from the project proposal and midterm report. New sections, i.e., complete results and future work after this course must be included in the final report.

What can be *optionally* included:

- Github repository (a URL link) that includes a complete copy of your implementation code (in this way, your code will be open source under a selected license).
- Jupyter-notebook that is used to demo how your model can work and display experimental results.

6.3.5 Final Presentation (CPT_S 534 Required, CPT_S 434 Optional)

Project final presentations (4% grade for CPT_S 534) take place during week 15 and 16, i.e., Apr 17th, Apr 22nd and Apr 24th, 2025. Please present your entire project and summarize your final report in 6 minutes.

Check the "Final presentation list" tab about the detailed arrangement on the Google Sheet for the course schedule.

6.3.6 Tips: How to Select A Project?

Below are some examples of the idea for a course project.

- Example reports Previous reports from other courses
  - CS229, Stanford: Fall 2017, Fall 2018, Fall 2019, Spring 2020, Spring 2021 (report + poster), suggested ideas
  - CS221, Stanford: Spring 2017 (titles only)
  - CS231n, Stanford: Spring 2017, Spring 2022 (report + poster)
- Examples from open competition on Kaggle competitions: deep learning applications
  - X-ray image classification/ranking: CheXpert competition from Stanford ML group
  - Landmark Retrieval: Google landmark retrieval competition 2021
  - Find extraterrestrial signals in data from deep space: SETI Breakthrough Listen - E.T. Signal Search
  - Identify the category of foliar diseases in apple trees: Plant Pathology 2021 - FGVC8
- Or any other ideas that are related to applying neural networks to a real use case

6.3.7 Course Project Grading

There are two grading systems for CPT_S 434 and CPT_S 534, respectively:

- CPT_S 434:
  - Proposal: 6%
  - Midterm report: 9%
  - Final report: 10%
- CPT_S 534:
  - Proposal: 4%
  - Midterm report: 6%
  - Midterm presentation: 3%
  - Final report: 8%
  - Final presentation: 4%

The grading rubrics for course project milestones (for reports and presentations) are based on:

- the clarification of statements,

- the elaborated motivations for the project,
- the details about why existing methods are not sufficient or not working,
- the design specifications of the proposed methods,
- the clear diagrams/tables/figures that describe the proposed model,
- the data source/description,
- the training techniques/tricks used in the project,
- the loss function to train the model,
- the analysis about how experiments relate to the problem statement,
- the comparison between the proposed method and existing ones,
- the visualization (figures, tables, etc.) of the comparison results,
- the ablation study and verification experiments,
- the explanation of each key finding,
- the reasonable future work based on the current exploration,
- the conclusion to summarize the high-level finding from the project.

6.4 Attendance (5% grade)

In-class quizzes (randomly), for attendance check purposes. For absence, please see the university policy for absences. Contact instructor to address your absence under this policy (see Section D).

6.5 Late Work Grading

Late submissions can still be graded under some circumstances.

6.5.1 Written Assignments And Course Project

For written assignments or written part of assignments in the format of Canvas Quiz, and course project (proposal + midterm report + final report), there is NO late submission policy, and the submission will be closed after due date. In other words, the due date for written assignments is firm.

6.5.2 Coding Assignments

For coding assignments or coding part of assignments, TA grader will use the following late work policy.

- After the scheduled due date, you will have a 2-calendar-day non-penalty period, allowing any variations of everyone's situation.

- After the 2-calendar-day non-penalty period, the maximum possible grade is decreased by 20% per day.
- Maximum 5-calendar-day late submission (i.e., 2-calendar-day non-penalty and 3-calendar-day penalty) is allowed. After that, assignment submission will be closed.
- Example 1: the scheduled due date is Feb 13, 23:59, PT. I submit my assignment at Feb 13, 23:58, PT (1 minute before the scheduled due date). The maximum possible grade I can get is 100 (full points). No action is taken.
- Example 2: the scheduled due date is Feb 13, 23:59, PT. I submit my assignment at Feb 15, 23:58, PT (1 minute before the end of two-day non-penalty period). The maximum possible grade I can get is 100 (full points). No action is taken. However, the late submission may influence my schedule for following assignments, which may bring me time constraint.
- Example 3: the scheduled due date is Feb 13, 23:59, PT. I submit my assignment at Feb 16, 00:03AM, PT (4 minutes after the end of three-day non-penalty period). The maximum possible grade I can get is 80 (= 100 - 20). If my submission originally gets 81 points out of 100, it will be capped at 80. If my submission originally gets 79 points out of 100, it will not be changed.
- Example 4: the scheduled due date is Feb 13, 23:59, PT. I submit my assignment at Feb 18, 23:59, PT (last minute of the six-day late submission). The maximum possible grade I can get is 40 (= 100 – 3*20). If my submission originally gets 81 points out of 100, it will be capped at 40.
- Example 5: the scheduled due date is Feb 13, 23:59, PT. I would like to submit my assignment at Feb 19, 00:01AM, PT. I found the submission has been closed, and my submission will not be graded.

6.6 Extension of Assignments

If more time for finishing an assignment is necessary, it is possible to get extension. For example, you have difficulty for finishing assignments on time due to health condition or other emergency condition. In this case, you can request extension with any supporting document such as doctor's appointment records, doctor's note, medical prescription, etc. Request without reasonable supporting document will not considered.

How to request extension: (i) contact TA if your request is to extend less than 7 calendar days (including 7 calendar days), or (ii) contact both TA and instructor if your request is to extend more than 8 calendar days (including 8 calendar days).

6.7 Grading Scale to Letter Grade

Table 1 shows the grading scale from percentage to letter grade. Curving grade is possibly considered only if the final grade is very biased and skewed that cannot reflect the performance in this course.

If there is any issue for TA to grade your coding assignment, TA will temporarily place 0% credit for your assignment, leave a comment to your submission on Canvas and send a Canvas

94-100.00%	A
90-93.99%	A-
86-89.99%	B+
82-85.99%	B
78-81.99%	B-
74-77.99%	C+
70-73.99%	C
66-69.99%	C-
62-65.99%	D+
58-61.99%	D
0-57.99%	F

Table 1: Grading scale to letter grade

Message to reach out to you, so it is important to *check your Canvas Inbox* and respond to TA if anything is required for TA to finish grading. For example, if TA cannot re-produce your code (things can be complicated on different systems), an in-person demo is required, and TA will leave a comment on Canvas to your submission and send you a Canvas Message. This in-person demo can then be done during a TA office hour.

7 Course Schedule

The more detailed course schedule can be found at this Google Sheet, Sheet 1. On Sheet 2 and 3 of this file, you can find the arrangement of presentation sessions (for CPT_S 534 only).

7.1 Class Meeting

Week 01, 01/06 - 01/10:

- Lecture 1: Syllabus
- TA session 1: Coding tools

Week 02, 01/13 - 01/17:

- Lecture 2: [ML] ML basics
- Lecture 3: [ML] PAC learning

Week 03, 01/20 - 01/24:

- Lecture 4: [ML] ERM and MLE
- TA session 2: HW1

Week 04, 01/27 - 01/31:

- Lecture 5: [NN] NN basics
- Lecture 6: [NN] Compositing units in neural networks

Week 05, 02/03 - 02/07:

- Lecture 7: [NN] Train a MLP Softmax classifier
- TA session 3: HW2

Week 06, 02/10 - 02/14:

- Lecture 8: [CNN] Convolutional layer and CNN models
- Lecture 9: [CNN] CNN architectures

Week 07, 02/17 - 02/21:

- Lecture 10: [CNN] Practical training for CNN
- Midterm exam (Feb 20th)

Week 08, 02/24 - 02/28:

- Lecture 11: [RNN] Sequence data and RNN
- TA session 4: HW3, HW4

Week 09, 03/03 - 03/07:

- Lecture 12: [RNN] LSTM and RNN architectures
- Lecture 13: [RNN] Attention and transformer

Week 10, 03/10 - 03/14: Spring vacation – no class

Week 11, 03/17 - 03/21:

- Lecture 14: [GNN] Graph data in ML and graph representation
- Lecture 15: [GNN] GNN

Week 12, 03/24 - 03/28: Midterm project presentations

- Midterm project presentation session 1 (CPT_S 534 required, CPT_S 434 optional)
- Midterm project presentation session 2 (CPT_S 534 required, CPT_S 434 optional)

Week 13, 03/31 - 04/04:

- Lecture 16: [GNN] Over-smoothing in GNN
- Lecture 17: [GAN] Adversarial ML

Week 14, 04/07 - 04/11:

- TA session 5: HW5, HW6
- Lecture 18: [GAN] Generative model and generative adversarial networks

Week 15, 04/14 - 04/18:

- Lecture 19: [GAN] Practical GAN training
- Final project presentation session 1 (CPT_S 534 required, CPT_S 434 optional)

Week 16, 04/21 - 04/25:

- Final project presentation session 2 (CPT_S 534 required, CPT_S 434 optional)
- Final project presentation session 3 (CPT_S 534 required, CPT_S 434 optional)

7.2 Reading Tasks

During the lectures that covers some corresponding sections, i.e., [ML], [NN], [CNN], [RNN], students are required to read some materials listed as follows.

ML
- Book "Foundations of machine learning":
  - Chapter 1 Introduction
  - Chapter 2 The PAC learning framework: (Definitions are required to read. Theoretical results, examples and proofs are optional to read)
  - Chapter 3 Rademacher complexity and VC-dimension is optional to read
  - Chapter 4 Model selection: Read Section 4.1 and 4.2. Proof is optional to read
- Book "Deep learning": Section 5.5 Maximum Likelihood Estimation

NN
- Book "Deep learning": Section 6 Deep forward networks

CNN
- Book "Deep learning":
  - Section 9 Convolutional networks
  - 7.4 data augmentation
  - 7.1 parameter norm penalties
  - 7.2 norm penalties as constrained optimization
  - 7.3 regularization and under-constrained problem
  - 7.12 dropout
- Book "Dive into deep learning":
  - 13.1 data augmentation
  - 4.5 weight decay
  - 4.6 dropout
- Reference papers mentioned by slides (note that only some sections of each paper are required to read. See below):

* [Alexnet] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems25 (2012): 1097-1105. Conference proceeding version at NeurIPS 2012 Abstract page or NeurIPS 2012 paper page (Section 3.5 required, other sections are optional)
* [pyramid] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Spatial pyramid pooling in deep convolutional networks for visual recognition." IEEE transactions on pattern analysis and machine intelligence37, no. 9 (2015): 1904-1916. ArXiv page (Section 2.2 required, other sections are optional)
* [NIN] Lin, Min, Qiang Chen, and Shuicheng Yan. "Network in network." arXiv preprint arXiv:1312.4400(2013). ArXiv page (Section 3.2 required, other sections are optional)
* [Inception] Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going deeper with convolutions." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1-9. 2015. ArXiv page (Section 4 and 5 required. Other sections optional)
* [ResNet] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778. 2016. ArXiv page (Section 3.1, 3.2 and 3.3 required. Other sections optional)

RNN

* Book "Deep learning"
  - Chapter 10: Section 10.1 - 10.5, 10.7, 10.10
* Reference paper “Transformer”. Vaswani, A. "Attention is all you need." Advances in Neural Information Processing Systems (2017). ArXiv paper.
* Optional external reading material:
  - Blog: understanding LSTM
  - Self-attention lecture slides from Shusen Wang
  - CS231n lecture slides for RNN

A Canvas Tutorials

- Send/reply Canvas Message: this tutorial.

B Tips for Coding Assignments

Python, Pytorch, Jupyter notebook are very commonly used tools for building deep neural networks. This course covers principles/rationales of deep neural networks, and does not cover how to use these tools to implement neural networks. 6 assignments will provide this chance to practice hands-on skills using these tools. If you have any specific issues regarding coding, **Google** and check out some online tutorials showcasing how these tools can work, such as Python Numpy Tutorial (with Jupyter and Colab) from Stanford CS231n and Recitations and Bootcamps provided by 11-785 from CMU, etc.

C Academic Integrity Policy

We must ensure that academic honesty is upheld. Learn more about Plagiarism, Unauthorized Assistance, Fabrication by visiting WSU's Academic Integrity Policy.

C.1 Can I Use Generative AI?

Generative AI, such as ChatGPT, should NOT be used to help directly and entirely write the code for any graded assignments/exams. However, generative AI can be used to search any relevant questions that you have when you attempt complete your assignments, analogous to using Google for searching your general questions, such as "what is Big-O notation?" or "is convolution operation a linear operation?", rather than a specific question like "help me write the code to build a ResNet model". Such general questions should not be directly related to the specific assignment questions.

Generative AI must NOT be used during any exams.

C.2 Can I Use Code from Internet?

You may use others' code in parts of your assignments, if you (i) give a complete and appropriate acknowledgement/citation in your submission to the source where you used the code, including the URL links, etc., and (ii) highlight which part of your submission is your original work that is different from the referred source.

However, TA will grade your submissions depending on how much major task you complete by your own. Therefore, using too much external code in your submission can still result in bad grades.

D Access Center

Let instructor know if you need special accommodations as early as possible. You may also visit WSU's Access Center.