CSCI5527: Deep Learning—Models, Computation, and Applications

Fall 2023

General Information

Over the last few years, deep neural networks (DNNs) have fundamentally transformed the way people think about machine learning and approach practical problems. Successes around DNNs have ranged from traditional AI fields such as computer vision, natural language processing, interactive games, to healthcare and physical sciences—touching each and every corner of theoretical and applied domains. On the other hand, DNNs still largely operate as black boxes, and we only have very limited understanding as for when and why they work. This course introduces the basic ingredients of DNNs, samples important applications, and throws around open problems. Emphasis is put on thinking from first principles and basic building blocks, as the field is still evolving rapidly and there is nothing there that cannot be changed.

- **Prerequisite:** CSCI5521 or CSCI5523 or equivalent. Maturity in linear algebra, calculus, and basic probability is assumed. Familiarity with Python (esp. numpy, scipy) is necessary to complete the homework assignments and final projects.
- **When & Where:** Wed 6:30PM – 9:00PM, Keller 3-210
- **Who:**

  Prof. Ju Sun (Instructor, `jusun@umn.edu`) Office hours: 1–3pm Mon (Keller 6-213)
  Tiancong Chen (TA, `chen6271@umn.edu`) Office hours: 10am–12pm Tue (Keller 2-209)
  Jiandong Chen (TA, `chen8111@umn.edu`) Office hours: 1–2pm Wed (Keller 1-213)

- **Public course website:** `https://sunju.org/teach/DL-Fall-2023/` All course materials, including course schedule, lecture slides, supplementary reading, homework sets, and project description, will be posted on the course website. We will make announcements in Canvas when we post there; enrolled students are encouraged to check the website on a regular basis.
- **Teaching mode: in-person** (except for UNITE-registered students). UNITE will record all lectures and release them immediately to UNITE students, and to on-campus students with a 10**-day delay**. UNITE policy is quoted below.

> Streaming video archives of class meetings are available to students registered in the on-campus section of this course on a TEN-DAY delay for the length of the semester. UNITE will not make media available to students enrolled in on-campus sections for any reason past the final exam, including when assigned an Incomplete by the instructor. See UNITE's Policy for On-Campus Students for details of access for on-campus students.
>
> This ten-day delay is lifted one week prior to scheduled exams and one week prior to finals as long as students are also enrolled in the course through UNITE Distributed Learning. If there are no UNITE enrollments, the ten-day delay may only be lifted the week prior to finals week.
>
> Access these videos through the UNITE Media Portal with your University of Minnesota Internet I.D. and password (this is what you use to access your University of Minnesota email account), [or through the **Media Gallery** on our Canvas page]

UNITE partners with the DRC in implementing accommodations approved by University of Minnesota's Disability Resource Center (DRC) consultants. Students working with the DRC may have their DRC consultant contact UNITE directly.

DO NOT ask the instructor or teaching assistants any questions related to the UNITE Media (such as access, technical or troubleshooting assistance). Instead, use the UNITE Troubleshooting FAQ or "Submit a Trouble Report to UNITE" link found on all pages within the UNITE Media Portal or send an email message to UNITE.

- **Communication:** Our philosophy is to minimize emailing. **Piazza** is the preferred and most efficient way of communication. Please post all questions and discussions related to the course there, and make them public if possible, instead of sending emails. If you have to use emails for non-technical issues, please begin the subject line with "CSCI5527" so that we can prioritize your emails.

Tentative Topics

Lecture sessions The tentative topics are as follows, and they are subject to change later.

- Course overview
- Neural networks: old and new
- Fundamental belief: universal approximation theorem
- Numerical optimization with math: optimization with gradient descent and beyond
- Numerical optimization without math: auto-differentiation and differential programming
- Working with images: convolutional neural networks
- Working with images: recognition, detection, segmentation
- Working with sequences: recurrent neural networks
- Transformers and large language models
- Working with graphs: graph neural networks & applications
- Learning probability distributions: generative models (GANs, VAE, normalization flow, and diffusion models)
- Learning representation without labels: dictionary learning and autoencoders
- Learning representation without labels: self-supervised learning
- Gaming time: deep reinforcement learning

Discussion sessions Lecture sessions are interlaced with several discussion sessions that are designed to help students master the critical computational and practical components and successfully complete their homework assignments and final projects.

- Python, Numpy, PyTorch
- Colab & MSI resources
- Project ideas

Assessment

- Homework 60%: 6 homework sets and 15% each, the top 4 scores will count toward the final grade
- Course project 40%: proposal (5%) + lightning talk (10%) + final report (25%). The project can be survey of a chosen topic not covered in detail in the class, comparison of existing methods, or novel foundational or applied research

Homework

You have approximately 2-week time to complete each homework (the exact due date will be specified in each assignment). **No late submissions will be accepted**, as we will drop the lowest 2 scores as stated above (non-submission leads to 0 automatically). All submissions *must* be electronic and uploaded via the Canvas/Gradescope system. You are encouraged to type your submission using LaTeX or word processing software, although legible scanned handwriting is also acceptable. Computer programs *must* be submitted in the Python notebook format. Only Python 3 will be used and accepted in this course.

Collaboration on homework problems is strongly encouraged, but each student must ensure that the final submission is prepared individually. *Collaborators should be properly acknowledged in the final submission, at the problem level.* The same applies to computer programs. Plagiarism and cheating will not be tolerated and will be subject to disciplinary action. Please consult the student code of conduct for more information: https://regents.umn.edu/sites/regents.umn.edu/files/2019-09/policy_student_conduct_code.pdf

Note on the use of AI resources Generative AI tools such as ChatGPT and GitHub Copilot can boost our productivity and potentially transform industrial and academic jobs. Therefore, the use of AI resources is **strongly encouraged** in this course, with the following guidelines:

- To make the best use of deep learning, understanding its foundation is crucial; to understand the foundation of AI, coding and playing with the basic ideas is the key.
- In this course, we assume that you have the ability to translate mathematical and algorithmic ideas into codes. Therefore, our homework problems focus on understanding, reasoning, and creative thinking, which the current generative AI tools seem weak at. We will scan our problems using these generative AI tools to make sure that the problems are reasonably hard for them. We will also pose more open-ended questions than in previous iterations of the course.
- *Use of AI resources should be properly acknowledged in the final submission, at the problem level.*

Course Project

The course project is to be carried out by teams of 3 or 4 students, and the weight of the project should be proportional to the number of students in the team. All students on the same team will receive the same score for their course project.

The project can be, but not limited to: a survey of literature on a focused topic not covered in class, comparison and improvement of existing methods, novel application of DNN techniques, and novel development of DNN methods and theories. You are encouraged to ask AI resources to generate project ideas for you and even ask them to draft codes for you. Our evaluation will be on the depth you can go: novelty, reasoning, and insights—which current generative AI tools are weak at—that you put into your projects. Use these tools to elevate the level of your project!

Programming and Computing

Our programming environment will be Python 3. We will use Pytorch as the default deep learning framework, although Tensorflow ($\geq$ 2.0) and Jax are also accepted and supported. For small-scale experiments (e.g., typical homework problems), Google Colab (https://colab.research.google.com/; everyone enrolled in the class gets reimbursed for a 3-month subscription to their professional version) will suffice. A local installation of relevant software packages may be a reasonable alternative, although we do not provide technical support for this. For large-scale course projects, we can use the Minnesota Supercomputing Institute (MSI) GPU computing queues based on our class account.

Recommended References

There is no required textbook. Lecture materials and assigned reading will be the primary resources. Recommended reference books are

- **Dive into Deep Learning** by Aston Zhang and Zachary C. Lipton and Mu Li and Alexander J. Smola. Livebook with online URL: https://d2l.ai/ (comprehensive coverage of recent developments and detailed implementations based on NumPy/Tensorflow/Pytorch/MXNet)
- **Deep Learning** by Ian Goodfellow and Yoshua Bengio and Aaron Courville. MIT Press, 2016. Online URL: https://www.deeplearningbook.org/ (comprehensive coverage of developments by 2016)
- **Neural Networks and Deep Learning** by Charu Aggarwal. Springer, 2018. UMN library online access (login required): Click here. (comprehensive coverage of recent developments)
- **The Deep Learning Revolution** by Terrence J. Sejnowski. MIT Press, 2018. UMN library online access (login required): Click here. (account of historic developments and related fields)
- **Deep Learning with Python** by François Chollet. Online URL: https://livebook.manning.com/book/deep-learning-with-python (hands-on deep learning using Keras with the Tensorflow backend)
- **Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems** by Aurélien Géron (2ed). O'Reilly Media, 2019. UMN library online access (login required): click here. (hands-on machine learning, including deep learning, using Scikit-Learn and Keras)
- **Deep Learning for Vision Systems** by Mohamed Elgendy (1ed). Manning Publications, 2020.

Related Courses

Within UMN

- **Topics in Computational Vision: Deep networks** (Prof. Daniel Kersten, Department of Psychology. Focused on connection with computational neuroscience and vision)
- **Analytical Foundations of Deep Learning** (Prof. Jarvis Haupt, Department of Electrical and Computer Engineering. Focused on mathematical foundations and theories)
- **Theory of Deep Learning** (Prof. Yulong Lu, School of Mathematics. Focused on the recent theoretical developments of deep learning)

Global

- **CS230 Deep Learning** (https://cs230.stanford.edu/, Stanford Computer Science)
- **CS231n: Convolutional Neural Networks for Visual Recognition** (http://cs231n.stanford.edu/, Stanford Computer Science, 2019)
- **CS224n: Natural Language Processing with Deep Learning** (http://web.stanford.edu/class/cs224n/, Stanford Computer Science, 2020)
- **Analyses of Deep Learning** (https://stats385.github.io/, Stanford Statistics, 2019)
- **Introduction to Deep Learning** (https://deeplearning.cs.cmu.edu/F20/index.html, CMU, 2020)
- **Advanced deep learning and reinforcement learning** (https://github.com/enggen/DeepMind-Advanced-Deep-Learning-and-Reinforcement-Learning, UCL/Deepmind, 2018)
- **Mathematics of Deep Learning** (https://joanbruna.github.io/MathsDL-spring18/, NYU Courant Institute, 2018)
- **MIT Deep Learning**(https://deeplearning.mit.edu/, MIT courses and lectures on deep learning, deep reinforcement learning, autonomous vehicles, and artificial intelligence)
- **CMSC 35246 Deep Learning** (https://ttic.uchicago.edu/~shubhendu/Pages/CMSC35246.html, U Chicago Computer Science, 2017)
- **Neural Networks for Machine Learning** by Jeof. Hinton (https://www.cs.toronto.edu/~hinton/coursera_lectures.html, U Toronto, 2012)