CSCI5527: Deep Learning—Models, Computation, and Applications
Fall 2022

General Information

Over the last few years, deep neural networks (DNNs) have fundamentally transformed the way people think of 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, 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 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:** Tue 6:30PM – 9:00PM, Keller 3-210
- **Who:** Prof. Ju Sun (Instructor) Email: jusun@umn.edu Office hours: 2–4pm Mon
  Hengkang Wang (TA) Email: wang9881@umn.edu Office hours: 1–3pm Thur
  Yash Travadi (TA) Email: trava029@umn.edu Office hours: 2–4pm Wed
- **Course Website:** https://sunju.org/teach/DL-Fall-2022/ All course materials, including course schedule, lecture slides, supplementary reading, homework assignments, 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). Note that as we return to this pre-COVID mode, releasing of UNITE recorded lectures for **on-campus students** is subject to a 10**-day delay**, different than the no-delay policy during COVID. 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).

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 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

We have 15 weeks and hence 30 75-min sessions.

26 lecture sessions The tentative topics and time allocation are as follows.

- Course overview (1)
- Neural networks: old and new (1)
- Fundamental belief: universal approximation theorem (2)
- Numerical optimization with math: optimization with gradient descent and beyond (2)
- Numerical optimization without math: auto-differentiation and differential programming (2)
- Working with images: convolutional neural networks (2)
- Working with images: recognition, detection, segmentation (2)
- Working with sequences: recurrent neural networks (2)
- Working with graphs: graph neural networks & applications (3)
- Learning probability distributions: generative models (GANs, VAE, Normalization flow, and diffusion models) (3)
- Learning representation without labels: dictionary learning and autoencoders (2)
- Learning representation without labels: self-supervised learning (2)
- Gaming time: deep reinforcement learning (2)

4 discussion sessions Lecture sessions are interlaced with 4 discussion sessions, that are designed to help the students master the critical computational and practical components and successfully complete their homework assignments and final projects.

- Python, Numpy, Jupyter Notebook, Colab
- Project ideas
- Pytorch for deep learning
- Research ideas

Assessment

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

Homework

You have approximately 2-week time to complete each homework (exact due date will be specified in each assignment). **No late submissions will be accepted**—we will drop the bottom 2 scores as stated above. All submissions *must* be electronic and uploaded via the Canvas/Gradescope system. You're encouraged to typeset 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 are 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

Course Project

The course project is to be performed 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 from the same team will get the same score for their course 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/) and UMN MSI notebook service will suffice. Local installation of the relevant software packages may be a reasonable alternative. 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)

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)