A Computational Introduction to Deep Learning

Instructor:	Brandon Franzke	Office:	EEB 504B	
Email:	franzke@usc.edu	Hours:	Monday:	17:10 – 19:00
			Thursday:	14:10 – 16:00

Machine learning using large datasets is the most transformative technology of the 21st century. Advances in generative ML promise solution to almost any problem imaginable. This course provides in-depth knowledge of deep learning systems theory and practice. It builds on prerequisite software skills from EE541: A Computational Introduction to Deep Learning and analytical skills from EE 559: Machine Learning I: Supervised Methods.

Lecture   Wednesday (section: 30404)   15:00 – 16:50

Enrollment is in-person ONLY. Attendance is mandatory to all lectures. Taping or recording lectures or discussions is strictly forbidden without the instructor's explicit written permission.

Teaching assistants

TA:	Karkala Hegde	Grader:	TBA
Email:	khegde@usc.edu	Email:	
Office:	(see BrightSpace)	Office:	(by appointment)

Course materials

[1] *Neural Networks and Deep Learning*, Aggarwal, C., Springer International, 2018. online, via SpringerLink.

[2] *Deep Learning*, Goodfellow, I., Bengio, Y., Courville, A., The MIT Press, 2016. online, http://www.deeplearningbook.org.

[3] *Deep Learning Architectures, A Mathematical Approach*, Calin, O., Springer International, 2020. online, via SpringerLink.

NOTE: Texts are secondary to in-class lecture material and homework sets.

"AI" policy. You may use AI-powered tools in this course to enhance your learning and productivity. Use AI as a collaborative tool for understanding concepts, generating ideas, and troubleshooting. Approach AI-generated content critically and use it responsibly. Engage with AI as you would with a knowledgeable peer or tutor, using iterative conversations to deepen your understanding. You must attribute all AI-generated content in your work, including the prompts you used. You are fully accountable for the accuracy and appropriateness of any AI-assisted work. AI should supplement, not substitute, your own critical thinking and problem-solving. For assignments, you may use AI to clarify concepts or resolve issues, but the core work must be your own. Submitting AI-generated work as your own without proper understanding or attribution is academic misconduct and will be treated as such.

You must develop complete mastery of all course material independently of AI assistance. Your knowledge and skills will be evaluated in contexts where AI tools are not accessible, mirroring real-world scenarios

where you must rely solely on your own expertise. This ensures you can perform effectively in any situation, with or without AI support. Violations of this policy will result in severe academic penalties. The goal is to prepare you to use AI effectively in your future work while ensuring you develop a strong, self-reliant foundation in the course material.

Course Outline

	Topics	Homework
Week 1 28 Aug	Introduction, PyTorch, and deep learning review.	**HW 0 assigned**.
Week 2 04 Sep	Advanced CNN architectures and applications.	HW 0 due. **HW 1 assigned**.
Week 3 11 Sep	Generative Adversarial Networks (GANs).	
Week 4 18 Sep	Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM).	HW 1 due. **HW 2 assigned**.
Week 5 25 Sep	Sequence-to-sequence models and attention mechanisms.	
Week 6 02 Oct	Transformers and self-attention.	HW 2 due. **HW 3 assigned**.
Week 7 09 Oct	**Exam #1** (weeks 1–6). AutoML and hyperparameter optimization.	
Week 8 16 Oct	Reinforcement learning principles.	HW 3 due.
Week 9 23 Oct	Advanced reinforcement learning applications.	**HW 4 assigned**.
Week 10 30 Oct	ML-Ops, pipelines, and production deployments. Project overview.	
Week 11 06 Nov	Project meetings.	**Draft project proposal due (03 Nov)**.
Week 12 13 Nov	Diffusion and modern generative models.	HW 4 due. **Revised project proposal due**.
Week 13 20 Nov	**Exam #2** (weeks 7–12). Explainable AI and ethical considerations.	
(28 Nov)	**No class, Thanksgiving Break.**	
Week 14 04 Dec	Project meetings.	**Status report due (02 Dec)**.
Friday 13 Dec	**Project presentations, 14:00 - 17:00 (mandatory).**	
Monday 16 Dec	**Project deliverables, due 12:00.**	

Grading Procedure

Homework (35%). Homework is assigned every 2-3 weeks. Assignments include analytic and programming problems and encourage experimentation and curiosity. You may discuss homework problems with classmates but each student must submit their own original work. Cheating warrants an "F" on

the assignment. Turning in substantively identical homework solutions counts as cheating.

Late homework is accepted with a 0.5% deduction per hour, up to 48-hours – **no exceptions**. Technical issues while submitting are not grounds for extension. No submissions will be accepted 48-hours after the due date. Graders score what is submitted and will not follow up if the file is incorrect, incomplete, or corrupt. It is your responsibility to ensure you submit the correct files and that they are accessible.

Exams (30%). Exams are non-cumulative and cover the most recent material (approximately 6-weeks). They test your ability to apply major principles, demonstrate conceptual understanding, and may require writing Python code. They occur during weeks 7 and 13 (tentative). You are expected to bring a scientific (non-graphing) calculator. You may use a single 8.5"x11" reference sheet (front and back OK). You may not use any additional resources. Any cheating may result in an "F" in the course and will be referred to Student Affairs for other penalties.

Exams include multiple-choice and short answer questions. They may also include free-response or open-ended questions to demonstrate conceptual understanding. You are expected to write reasonably correct code as well as determine expected behavior of novel computer code. Grading primarily follows correct reasoning but may include deductions for major syntax errors, algorithmic inefficiency, or poor implementation.

Final Project (35%). This course culminates with a final project in lieu of a final exam. Teams of three students (teams of two with instructor approval) design and implement a deep-learning system to a self-identified problem. Students should treat the project as a platform to demonstrate mastery of problem specification, model selection, data analysis, testing, debugging, and results validation and analysis. The instructor will guide teams with difficulty identifying a suitable problem. Teams will prepare and present their approved project and show how it applies course concepts and deep learning best-practices. Attendance and participation during the project presentation session is mandatory.

Course Grade
A if 90 - 100 points, **B** if 80 - 89 points, **C** if 70 - 79 points, **D** if 60 - 69 points, **F** if 0 - 59 points.
("+" and "−" at $\approx$ 1.5% of grade boundary).

Cheating
Cheating is not tolerated on homework or exams. Penalty ranges from F on exam to F in course to recommended expulsion.

Final Project

Requirements. Groups are encouraged to devise solutions to novel problems of personal interest to their background or research. But teams may select a problem with prior solutions provided their effort demonstrates novelty in addition to mastery of the course material. Groups may abstract problems from original context to fit within the project timeline and simplify constraints and scale. Projects must include sufficient mathematical and hypothetical complexity and include or extend substantive material from the course. <u>All projects must obtain the instructor's written approval</u>. All projects must use PyTorch as the primary deep learning framework unless approved explicitly in writing by the instructor. But projects may use additional languages for tooling and support. The instructor may provide additional requirements when introducing the final project assignment.

Scoring and Milestones.

Topic Proposal (initial and revised)	week 10	4% + 10%
Status Report - data, training, integration	week 14	8%
Presentation and demo	final	20%
Project report		20%
Model Card		4%
Model and source code		30%
Video		4%

Project Deliverables.

Topic proposal: describe the problem, prior related work, candidate data sets/sources, proposed technical approach, and expected outcomes. It should communicate that your topic is adequately prepared and it should outline immediate next steps. But the proposal is merely a guidepost and reasonable deviations in method, approach, and scope are expected.

Written report: summarize the topic, provide relevant background (theoretical or applied), timeline and contributions, and document challenges and extensions. It should provide discussion sufficient that an uninformed expert can understand the models, analytic decisions, outcomes, and implementation. Teams should provide quantifiable metrics to justify engineering tradeoffs.

Presentation: approximately 20 minute presentation to describe the project problem, your approach, and results. It should provide only what is necessary to understand the *what* and *why* and include minimal theoretical background.

Source code: submitted as a GitHub repository archive file (zip). It must include README file(s) that describe the repository structure, execution instructions, and special technical requirements. It should not include any training data or model files.

Video: a 4-minute video that describes the topic, your implementation, and your results. You may choose to upload this to a video sharing site such as YouTube but that is not required.

Academic Accommodations

Any student requesting academic accommodations based on a disability is required to register with Disability Services and Programs (DSP) each semester. A letter of verification for approved accommodations can be obtained from DSP. Please be sure the letter is delivered to me as early in the semester as possible. DSP is located in STU 301 and is open 08:30 - 17:00, Monday through Friday. The phone number for DSP is (213) 740-0776.

Support Systems

A number of USC's schools provide support for students who need help with scholarly writing. Check with your advisor or program staff to find out more. Students whose primary language is not English should check with the *American Language Institute* http://dornsife.usc.edu/ali, which sponsors courses and workshops specifically for international graduate students. *The Office of Disability Services and Programs* http://sait.usc.edu/academicsupport/centerprograms/dsp/home_index.html provides certification for students with disabilities and helps arrange the relevant accommodations. If an officially declared emergency makes travel to campus infeasible, *USC Emergency Information* http://emergency.usc.edu will provide safety and other updates, including ways in which instruction will be continued by means of brightspace, teleconferencing, and other technology.

Discrimination, sexual assault, and harassment are not tolerated by the university. You are encouraged to report any incidents to the *Office of Equity and Diversity* http://equity.usc.edu or to the *Department of Public Safety* http://capsnet.usc.edu/department/department-public-safety/online-forms/contactus. This is important for the safety of the whole USC community. Another member of the university community - such as a friend, classmate, advisor, or faculty member - can help initiate the report, or can initiate the report on behalf of another person. *The Center for Women and Men* http://www.usc.edu/studentaffairs/cwm/ provides 24/7 confidential support, and the sexual assault resource center webpage http://sarc.usc.edu describes reporting options and other resources.

Academic Conduct

The University of Southern California is foremost a learning community committed to fostering successful scholars and researchers dedicated to the pursuit of knowledge and the transmission of ideas. Academic misconduct is in contrast to the university's mission to educate students through a broad array of first-rank academic, professional, and extracurricular programs and includes any act of dishonesty in the submission of academic work (either in draft or final form).

This course will follow the expectations for academic integrity as stated in the *USC Student Handbook*. All students are expected to submit assignments that are original work and prepared specifically for the course/section in this academic term. You may not submit work written by others or "recycle" work prepared for other courses without obtaining written permission from the instructor(s). Students suspected of engaging in academic misconduct will be reported to the *Office of Academic Integrity*.

Other violations of academic misconduct include, but are not limited to, cheating, plagiarism, fabrication (*e.g.*, falsifying data), knowingly assisting others in acts of academic dishonesty, and any act that gains or is intended to gain an unfair academic advantage.

Academic dishonesty has a far-reaching impact and is considered a serious offense against the university. Violations will result in a grade penalty, such as a failing grade on the assignment or in the course, and disciplinary action from the university itself, such as suspension or even expulsion.

For more information about academic integrity see the student handbook https://policy.usc.edu/studenthandbook/, or the Office of Academic Integrity's website https://academicintegrity.usc.edu/, and university policies on Research and Scholarship Misconduct https://policy.usc.edu/research-and-scholarship-misconduct/.