SYLLABUS (18 Aug 2022)

Deep Learning Systems

EE 641: Fall 2022 (2 units)

Machine learning using large datasets stands as one of the most transformative technologies of the 21st century. It enables reliable face and speech recognition, internet search and monetization, computer vision, and fully autonomous vehicles. The goal of this class is to provide in-depth knowledge in deep learning systems theory and practice by building on deep learning software skills from EE541: A Computational Introduction to Deep Learning and analytical skills from EE 559: Machine Learning I: Supervised Methods.

Instructor:	Brandon Franzke	
Email:	franzke@usc.edu	
Office:	EEB 504B	
Zoom:	meet:	998 5176 5591
	code:	574987
Hours:	Monday:	17:00 - 18:00
	Wednesday:	12:00 - 14:00

Lecture

Monday (section: 30415)
12:00 - 13:50

Wednesday (section: 30404)
15:00 - 16:50

Attendance and Participation

This class is offered with **in-person enrollment ONLY.** Attendance is mandatory to all lectures. You are responsible for missed announcements or changes to the course schedule or assignments. Taping or recording lectures or discussions is strictly forbidden.

Piazza

https://piazza.com/usc/fall2022/ee641

Piazza enables fast and efficient help from classmates and instructors. Use Piazza to post questions about course material, homeworks, and policies instead of emailing questions to the teaching staff.

Canvas

https://canvas.usc-ece.com

Use Canvas to electronically submit your homework and view course grades. You will receive an email to register during the first week of classes. Contact Dr. Franzke with any technical issues.

Autolab

https://autolab.usc-ece.com

Use Autolab to electronically submit programming portions of homework for “auto-grading”. You will receive an email to register during the first weeks of the course. Contact Dr. Franzke with technical issues.

TAs and staff

TA:	Karkala Hegde (Shashank)	TA:	Ganning Zhao
Email:	khegde@usc.edu	Email:	ganningz@usc.edu
Office:	PHE 320	Office:	PHE 320
Zoom:	meet: 724 621 7299	Zoom:	meet: 938 9006 3612
Hours:	Thursday: 11:00 - 12:30		code: 977169
	Friday: 15:00 - 16:30	Hours:	Friday: 10:00 - 12:00
CP:	Manikanta Chunduru Balaji	Grader:	Tejas Suresh Khanolkar
E-mail:	mchundur@usc.edu	E-mail:	tkhanolk@usc.edu
Hours:	by appointment	Hours:	by appointment

Learning objectives

Upon completion of this course a student will be able to:

- Understand the relationship between inference methods based on statistical models and data-driven methods.
- Describe relationships among common loss and regularization functions to maximum likelihood parameter estimation, maximum a posteriori probability parameter estimation, Kullback–Leibler divergence, and cross-entropy.
- Understand stochastic gradient descent learning in neural networks and its relation to other methods such as least-mean squares (LMS) adaptive linear filtering, and the EM algorithm.
- Derive back-propagation equations for dense, convolutional, and recurrent layers (back-propagation in time).
- Apply commonly used deep-learning techniques: activations, optimizers, loss functions, regularization, drop-out, batch normalization, and down/up sampling layers. Apply this knowledge to design appropriate networks for a given inference problem.
- Employ good data systems engineering practices including pipeline development for data acquisition, cleaning, augmentation, training, testing, and closed-loop refinement.
- Understand the role of deep learning networks in modern speech, computer vision, and signal processing systems.

Course materials

- "*Neural Networks and Deep Learning*", Charu Aggarwal, Springer International, 2018. (online via SpringerLink).
- "*Deep Learning*", Ian Goodfellow, Yoshua Bengio, Aaron Courville, The MIT Press, 2016. (online: http://www.deeplearningbook.org).
- "*Deep Learning Architectures, A Mathematical Approach*", Ovidiu Calin, Springer International, 2020. (optional, online via SpringerLink).

Course Outline (tentative)

	Topics/Daily Activities	Reading & Homework	Deliverables
Week 1 (22 Aug)	Introduction, CNN/PyTorch review		
Week 2 (29 Aug)	CNN architectures, image segmentation	HW 1 assigned.	
Week 3 (05 Sep)	Generative Adversarial Networks (GNN)		
Week 4 (12 Sep)	Recurrent Neural Networks (RNN)	HW 2 assigned.	HW 1 due.
Week 5 (19 Sep)	RNN, Attention mechanisms		HW 2 due (23 Sep).
Week 6 (26 Sep)	**Exam #1 (weeks 1-5).**		
Week 7 (03 Oct)	Reinforcement Learning	HW 3 assigned.	
Week 8 (10 Oct)	Transformer architectures	HW 4 assigned.	HW 3 due.
Week 9 (17 Oct)	AutoML, Hyperparameter tuning, and training optimization		Preliminary proposal due (20 Oct). Proposal return: 26 Oct.
Week 10 (24 Oct)	Diffusion and modern generative models		HW 4 due.
Week 11 (31 Oct)	ML-Ops, developing pipelines, and production deployments		Revised proposal due (04 Nov).
Week 12 (07 Nov)	Explainability, Bias and ethical AI		Phase 1 status due.
Week 13 (14 Nov)	**Exam #2 (weeks 8-13).**		
(21 Nov)	**No class, Thanksgiving break, University holiday.**		
Week 14 (28 Nov)	Optional topic: Natural language processing (NLP), Graph neural networks		Phase 2 status due.

Grading Procedure

Homework

Homework is assigned every 2-3 weeks. Assignments include analytic and computational programming problems and will encourage experimentation and curiosity. <u>No late submissions for credit</u>. Your total homework score sums your best homework scores (as percentages). You may discuss homework problems with classmates but each student must do their own original work. Cheating warrants an F in the course. Turning in identical homework establishes a rebuttable presumption of cheating.

Exams

Exams are non-cumulative assessments that cover the most recent material (approximately 6-weeks). Exams highlight important concepts and methods. They test ability to apply major principles, demonstrate conceptual understanding, and may require writing snippets of Python code. They occur during weeks 6 and 13 (tentative). You may use a single 8.5"x11" reference sheet (front and back OK). You may not use any additional resources. You are expected to bring a scientific (non-graphing) calculator. Any cheating may result in an "F" in the course and will be referred to Student Affairs for other penalties. Alternate arrangements will be considered only for valid medical or family emergency excuses (proof required).

Final Project

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 "mock" industry or research 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 material, concepts, and best-practices. Attendance and participation during the project presentation session(s) are mandatory.

Course Grade

Homework	25%
Exams	30%
Final Project	45%

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 "–" within approx. 3% of grade boundary).

Cheating

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

Final Project

Requirements

Groups are encouraged to devise solutions to novel problems of particular interest to their backgrounds, interest, or research. But teams may select a problem with prior solutions provided their efforts demonstrate novelty in addition to mastery of the course material. Groups may substantially abstract problems from original context to fit within the project timeline and simplify both constraints and scale. Projects must include sufficient mathematical and hypothetical complexity and include or extend substantive material from the course. All projects must obtain the instructor's written approval. All projects must use PyTorch as the primary deep learning framework unless approved explicitly in writing by the instructor. But teams may use or integrate 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 11	14%
Phase 1 – Data collection, training methodology	week 13	8%
Phase 2 – Integration and deployment	week 15	8%
Presentation and demo	final	20%
Project report		20%
Model and source code		25%
Video		5%

Final Deliverables

Written project 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 could understand the models, analytic decisions, outcomes, and implementation. Teams should provide quantifiable metrics to justify engineering tradeoffs.

Presentation: approximately 15 minute (depends on class size) presentation to describe the project problem and their solution. It should provide only what is necessary to understand the "what" and "why" and include minimal theoretical background.

Source code: submitted to instructor by providing link to pull from github.

Video: a 3-minute video that describes the problem, your design, and implementation. You may choose to upload this to a video sharing site such as YouTube but that is not required. All team members must participate equally.

Academic Conduct

Plagiarism

Presenting someone else's ideas as your own, either verbatim or recast in your own words – is a serious academic offense with serious consequences. Please familiarize yourself with the discussion of plagiarism in SCampus in Section 11, Behavior Violating University Standards https://scampus.usc.edu/1100-behavior-violating-university-standards-andappropriate-sanctions. Other forms of academic dishonesty are equally unacceptable. See additional information in SCampus and university policies on scientific misconduct, http://policy.usc.edu/scientific-misconduct. 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 Integrity

Academic integrity is critical the assessment and evaluation we perform which leads to your grade. In general, all work should be your own and any sources used should be cited. Gray-areas occur when working in groups. Telling someone how to do the problem or showing your solution is a VIOLATION. Reviewing examples from class or other sources to help a fellow classmate understand a principle is fine and encouraged. All students are expected to understand and abide by these principles. SCampus, the Student Guidebook, contains the University Student Conduct Code in Section 10, while the recommended sanctions are located in Appendix A. Students will be referred to the Office of Student Judicial Affairs and Community Standards for further review, should there be any suspicion of academic dishonesty.

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 blackboard, teleconferencing, and other technology.

Academic Accommodations

Any student requiring 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 GFS 120 and is open 08:30 – 17:00, Monday through Friday. The phone number for DSP is (213) 740-0776.