Course Missive

Spring 2023

Class Time and Location: MWF 12:00-12:50 PM, Salomon Center 001
Lectures will be given live and recorded for asynchronous viewing.
In-person attendance highly encouraged.
Website: `www.cs.brown.edu/courses/csci1470/`
Canvas Site: `www.canvas.brown.edu/courses/1091047`
Instructor's office hours: Thursdays 2:30-4:30 PM or by appointment
Office hours location: CIT 203 (Zoom option available)

Course Staff

What	Who
Professor	Ritambhara Singh (`rsingh47`)
Head TAs	Dylan Hu (`dhu24`)
	Nitya Thakkar (`nthakka3`)
	Raymond Dai (`rdai4`)
	Robert Scheidegger (`rscheide`)
	Vadim Kudlay (`vkudlay`)
TAs	Preeti Nagalamadaka (`pnagala1`)
	Shirley Loayza Sanchez (`sloayzas`)
	Logan Bauman (`lbauman`)
	Jun Ha (`jha38`)
	Will Guo (`wguo25`)
	Bumjin Joo (`bjoo2`)
	Henry Sowerby (`hsowerby`)
	Ray Del Vecchio (`rdelvecc`)
	Iris Cheng (`icheng3`)
	Nange Li (`nli32`)
	Eric Han (`ehan31`)
	Karan Kashyap (`kkashyap`)
	Joe Dodson (`jdodson4`)
	Dave Lubawski (`dlubawsk`)
	Taishi Nishizawa (`tnishiza`)
	Ray Wang (`xwang356`)
	Earth Mokkamakkul (`jmokkama`)
	Michael Lu (`mlu54`)
	Evan Lu (`elu14`)
	Xianghao Xu (`xxu43`)
STAs	Brendan Ho (`bho15`)
	Faizaan Vidhani (`fvidhani`)

Introduction

Welcome to CSCI 1470/2470! Over the past few years, Deep Learning has become a popular area, with deep neural network methods obtaining state-of-the-art results on applications in computer vision (Self-Driving Cars), natural language processing (Google Translate), and reinforcement

learning (AlphaGo). These technologies are having transformative effects on our society, including some undesirable ones (e.g. deep fakes). This course intends to give students a practical understanding of how Deep Learning works, how to implement deep neural networks, and how to apply them ethically. We introduce students to the core concepts of deep neural networks, including the backpropagation algorithm for training neural networks, as well as specific operations such as convolution (in the context of computer vision), word embeddings, and recurrent neural networks (in the context of natural language processing). Throughout the lectures, labs, and assignments, we emphasize and require students to think critically about potential ethical pitfalls that can result from mis-application of these powerful models. The course is taught using the Tensorflow deep learning framework.

Course objectives

By the end of this course, you will be able to:

- Learn about the fundamental algorithms that underly all modern deep learning models.
- Implement different types of deep learning models in Tensorflow.
- Think critically about using a deep learning model for a task and its potential societal impact.
- Collaborate with classmates on a team project to apply deep learning models to task of your choice.
- Communicate your findings (both positive and negative results are encouraged) through presentations.

Prerequisites

- A basic programming course: (CSCI 0150, 0170 or 0190)
- A linear algebra course: (CSCI 0530, MATH 0520 or 0540)
- A stats / probability course: (CSCI 0220, 1450, 0450, MATH 1610, APMA 1650 or 1655)

Exceptions may be possible for those missing one of these prerequisites if (a) the student has taken another course which covers similar material, or if (b) the student will be concurrently taking the prerequisite. If either of these situations applies to you, use the "Request Override" feature in Courses@Brown to request an override code (and explain why you believe your situation merits one).

Textbook

None required. Students are encouraged to refer to the following textbook, which is available online:

- Deep Learning, by Ian Goodfellow, Yoshua Bengio, and Aaron Courville.

Assessment of learning

There are six assignments, weekly labs, and a final project. The grade breakdown for this course is as follows:

- 60% – Assignments (programming + written components)
- 15% – Weekly Labs + 2 Socially Responsible Computing (SRC) sessions
- 20% – Final Project
- 5% – Class participation

Assignments Assignments consist of a programming component (e.g. implementing and training some neural network model) as well as a written component. The written component contains both algorithmic conceptual questions as well as an open-ended ethics question that requires students to reflect on the societal ramifications of the machine learning models they are building in the programming portion of the assignment. Answers to the written questions are due **before** the programming component deadline, i.e., there are two separate deadlines: one for the written component, followed by one for the programming component.

Late Policy **Each of the assignments is due by 6 PM EST** on the given due date — thus, you have until then for the day listed as the due date to complete the assignment. **You will have four (4) free late days** to use towards all but the last project. These late days can be used on code or conceptual assignments. After your late days are expended you will lose 10% of your project grade for each extra day your handin is delayed. At the end of the term, we will calculate how to best divvy out your late days to best help your final grade. There is no need for you to do anything special for this other than keep track of submission dates for your records. Note that because we do this, **use of late days will not be reflected in the initial grade report for your assignments**.

Exceptions to the Late Policy Sometimes there are special circumstances during the semester that result in exceptions to this late policy. All such circumstances require an official note from the Deans for undergraduate students. In general, they only provide support notes on behalf of students who are experiencing disruptive medical or personal circumstances, including those related to Title IX situations, that affect their ability to do academic work in a timely way.

Undergraduate students: If you believe an extension is warranted for you, please have a note from the Dean sent to the instructor. The Dean will send the note directly to the instructor, and you do not have to follow up on it. Once the instructor receives the Dean's note, she will notify you and the TAs of the granted extension with the date of submission in 2-3 days. **What you should not do:**

- Do not send email to the instructor requesting an extension—**you will not receive a response**.
- Do not send a dean's note to a UTA/HTA. These often contain sensitive personal information, the receipt of which by anyone other than the instructor is a FERPA violation.

Graduate students: If you believe an extension is warranted for you, please email the instructor with the request for an extension and the reason. For health-related issues, please attach a note from the medical staff. **What you should not do:**

- Do not send the request to the UTA/HTA. These often contain sensitive personal information, the receipt of which by anyone other than the instructor is a FERPA violation.

You should manage other special circumstances such as interviews, personal travel, or extra-curricular factors using the late-day policy above.

Submission Mistakes Every year, a few students suffer from submission mistakes: accidentally submitting the wrong version of their code, re-submitting a broken version which overwrites a previous correct solution, etc. Since we are using Github Classroom, all student assignment work will be stored it Github repositories. If you have a submission mistake that cannot be resolved before the deadline, inform the HTAs, and include a link to a commit in your assignment repository that represents the code you wish you had submitted. Note that this commit must have a timestamp from before the assignment submission deadline. Handling such exception cases adds overhead to an already work-intensive grading process on our part. Thus, while your first mistake carries no grade penalty, subsequent mistakes will cost you 10% of your assignment grade. In other words, it's in your best interest to be thorough with your code submissions.

Labs

Students will attend one two-hours-long weekly lab slot assigned to them. It is expected that students should be able to complete the exercises for a given lab and get them checked off by the lab TA during this assigned slot. Lab materials for each week will be released online at the beginning of the week; students are welcome to look at and beginning working on the lab in advance of their lab slot, if they wish. If the student does not complete the lab and/or get it checked off during their slot, they can go to another lab section that week to get it checked off (without penalty).

If a student misses their assigned lab slot, they can also go to another lab section to get the lab checked off. However, this will incur a lab grade penalty (e.g. a 3/4 instead of a 4/4 for that lab). Students will be permitted to miss their assigned lab slot three (3) times without incurring this penalty.

In all cases, a lab must be checked off at latest by the student's assigned lab slot the following week.

If a student knows well in advance that they will not be able to make their assigned lab slot but will be able to make another slot that week, the student may message their lab TA **at least 48 hours** in advance to be temporarily switched into a different lab slot for that week. This will not incur a late lab penalty.

Lab collaboration Students may complete labs in groups of two if they wish. **In fact, teamwork is highly encouraged for lab work**. In this case, both students must be present for the TA to check them off, and the TA will ask check-off questions of both students.

Socially Responsible Computing (SRC) discussion sessions

It is impossible to learn about deep learning and not discuss the societal impact of the models and their applications. With every passing year, SRC questions and issues related to deep learning are becoming more prevalent, and in this course, we would like to keep the dialogue going. Therefore, we have included SRC questions for students to work on in their assignments. Additionally, this semester, we will be holding SRC discussion sessions. Each student will sign up to attend two discussion sessions during the course. The STAs will facilitate discussions related to the SRC topics covered in the homework assignments, and the students will participate in groups. More details will be provided during the course.

Final Project

All students will be required to complete a final project in groups of 2-4. At minimum, this should entail re-implementing the methods described by a recent deep learning research paper. Expectations are higher for 2470 students and 1470 capstone students; see the relevant sections below for more information. More details will be provided during the course.

Deep Learning Day: The class's collective final project efforts culminate in "Deep Learning Day," an end-of-semester celebration consisting of poster sessions and oral presentations organized as a day-long symposium. Details of the day will be shared during the course.

Class participation

Students are highly encouraged to attend the classes in person. However, we will record lectures for students to view if they miss a class or need to review class material. Since keeping track of in-class participation is difficult in large classes, **after Wednesday's lecture, the instructor will post a short quiz on Canvas.** Students are expected to complete the short quiz by noon the next day (Thursday) in order to obtain full participation credit for the week. If you attend the class (or watch the lectures) regularly, you should be able to finish the quiz quickly and with little effort. The quiz will also help you review the course material regularly and make sure you are keeping up with the course topics. The students will be given full points for correct answers, partial points for incorrect answers, and no points for incomplete or no submission. **Please note that no extensions will be provided for completing the quizzes.** They contribute a relatively small percentage to your final grade, so missing a quiz or two will not affect your grade drastically.

How can you do well?

This class has a high-level course load and you can ensure your success in it by doing the following:

- Regularly attending classes and lab sections. Participation in class discussions is highly encouraged.
- Starting the work on assignments and projects early.

- Completing and turning in all assignments and quizzes on time.
- Equally contributing to the final project and clearly presenting your project.

Differences Between 1470 and 2470

Both sections of this course share the same course staff, lectures, labs, and assignments. 1470 is open to everyone who meets the prerequisites. 2470 is a graduate-level course and is intended for graduate students; registration is gated accordingly. The instructor reserves the right to admit undergraduates into 2470 at her discretion.

In terms of required coursework, the two sections have the following differences:

Assignments (Programming): Every programming assignment has a quantitative metric threshold that all students' code is expected to meet (e.g. test-set accuracy). The threshold for 2470 students is higher. Reaching this higher threshold may require more advanced model architectures, the implementation of which may require reading additional reference material.

Assignments (Written): 2470 students will be required to answer additional written questions. These may be mathematical (e.g. proving/deriving some property of a model/algorithm) or conceptual in nature (e.g. requiring reading a related paper and connecting its findings to what we have discussed in class).

Final Project: 2470 students will be required to go above-and-beyond re-implementing an existing research paper: rather, they will be required to attempt a novel research project. This could involve investigating a new model architecture to solve an existing problem, adapting methodology from a different problem to solve an existing problem, or defining and solving a new problem of interest.

Programming

All programming in this course will be done in Python, primarily using numpy and the Python Tensorflow API. Labs will be provided as Google Colab notebooks.

Since this is not a software engineering course, we won't be enforcing stringent style guidelines, but you should write so that someone who isn't a Python wizard will be able to understand what your program is doing (add plenty of comments, break up code into smaller functions, i.e. apply basic common sense). If you turn in a partially-functional assignment and we can't tell what you were trying to do, we'll probably be very grumpy about giving partial credit.

As that translates to an official policy, so long as your code produces the expected output(s) and adheres to any specific project restrictions (runtime, etc.) then you will not lose points for poor design or coding practices. *However*, as this is not a software design course, it is not the responsibility of the TAs to attempt to understand the intentions underlying confusing code. If it is not fully clear what you were trying to do in the implementation of a partially-functional

assignment (i.e. not all of the output is as expected) then partial-credit will be given sparingly and at our discretion.

Policy on the use of AI-powered tools for course assignments

All work that students submit during the course must be their own original work and represent their own thoughts and ideas. As such, the use of AI-powered tools (such as OpenAI's ChatGPT or GitHub's CoPilot) for completing course assignments is discouraged. The use of AI-powered tools without citation will be considered academic misconduct. For programming assignments, generating code from a comment or "todo" statement provided in the stencil will also be considered academic misconduct.

If a student chooses to use these tools for course assignments, they must acknowledge and thoroughly document their use of the tool. The student must: 1) cite the tool used, 2) include an explanation of how the tool was used for the assignment, and 3) fully document the student's own contribution versus the contribution of the tool (e.g., including full ChatGPT transcripts as an appendix to your assignment). All assignments will be graded based on the student's original ideas – students risk losing credit if the documentation provided is insufficient to determine the student's original contributions.

Time Requirements

In addition to 3 hours per week in class, you will probably need 1 hour of help from a UTA, 2 hours for lab, 2 hours of reading and review to solidify your grasp of the material (including submitting the weekly quiz) and 6 hours for the assignments. Additionally, you will need 2 hours (during the semester) for attending the 2 SRC discussion sessions. (184 hours/semester)

Capstone

This course may be used as a capstone course for an Sc.B. degree. Bring a copy of the capstone form to the instructor after class or during office hours. All 1470 students who use the course as a capstone will be expected to do fulfill the same final project requirements as students taking 2470.

Students with Special Needs:

Brown University is committed to full inclusion of all students. Please inform me early in the term if you have a disability or other conditions that might require accommodations or modification of any of these course procedures. You may speak with me after class or during office hours. For more information, please contact Student and Employee Accessibility Services at 401-863-9588 or SEAS@brown.edu. Students in need of short-term academic advice or support can contact one of the deans in the Dean of the College office.

Diversity Statement:

This course is designed to support an inclusive learning environment where diverse perspectives are recognized, respected and seen as a source of strength. It is our intent to provide materials and activities that are respectful of various levels of diversity: mathematical background, previous computing skills, gender, sexuality, disability, age, socioeconomic status, ethnicity, race, and culture.

Academic Integrity and Collaboration Policy

Academic dishonesty will not be tolerated. This includes cheating, lying about course matters, plagiarism, or helping others commit a violation. Plagiarism includes reproducing the words of others without both the use of quotation marks and citation. Students are reminded of the obligations and expectations associated with the Brown Academic and Student Conduct Codes.

Discussion of course material with your classmates is both permitted and encouraged.However, **showing, copying, or other sharing of actual code or verbatim answers to written questions is forbidden**. This includes publishing projects on Github or any other public platform. This policy **will be** enforced.

One of the ways that we enforce the collaboration policy is by running MOSS on all code submissions. For those who are new to the department or otherwise unfamiliar with it, MOSS (short for "measure of software similarity") is a software tool which detects similarities between pieces of code. The course staff manually examines the output of MOSS to look for cases where two students' submitted code is similar in such a way that it is sufficiently unlikely for them to have independently produced it (i.e., an instance of cheating or code plagiarism).

Regret clause: We recognize that college is stressful, and sometimes that stress leads to panicked, bad decision making near deadlines. We are also aware that this problem is likely to be exacerbated by the unusually stressful period of history through which we are all living right now. In the spirit of this understanding, we are implementing a "regret clause" for the course this year. If a student submits an assignment that they know to contain plagiarized material (either code or answers to written questions), that student may invoke the regret clause by informing the instructor of the plagiarized material within 48 hours of the assignment submission deadline. The instructor will ask to meet with the student, and the outcome of this meeting can vary. On a first invocation of the regret clause, the instructor will exercise her discretion, applying a penalty up to receiving zero credit on the plagiarized portion of the assignment. On subsequent invocations, all plagiarized portions of the assignment will automatically receive zero credit. In all cases, the occurrence of plagiarism will be kept confidential by the instructor.

If the course staff detects a plagiarized submission, and the student has not invoked the regret clause, then we will write up a formal plagiarism case to be heard by the Academic Code Committee. These hearings can result in a variety of outcomes, including immediately failing the course and receiving a "Directed NC" (i.e. a form of NC that does show up on your transcript, along with a note that the NC was given for an academic code violation).

Accommodations

Brown University is committed to full inclusion of all students. Please inform the instructor if you have a disability or other condition that might require accommodations or modification of any of these course procedures. You may email the instructor, come to office hours, or speak with her after class, and your confidentiality is respected. We will do whatever we can to support accommodations recommended by SEAS. For more information contact Student and Employee Accessibility Services (SEAS) at 401-863-9588 or `SEAS@brown.edu`. Students in need of short-term academic advice or support can contact one of the deans in the Dean of the College office.

Mental Health

Being a student can be very stressful. If you feel you are under too much pressure or there are psychological issues that are keeping you from performing well at Brown, we encourage you to contact Brown's Counseling and Psychological Services (CAPS). They provide confidential counseling and can provide notes supporting extensions on assignments for health reasons.

Incomplete Policy

We expect everyone to complete the course on time. However, we certainly understand that there may be factors beyond your control, such as health problems and family crises, that prevent you from finishing the course on time.

CSCI 1470/2470 is now offered in the Fall and Spring semesters with similar course content. Therefore, starting this academic year, we are introducing a strict incomplete policy for the course. The instructor will allow only students with roughly 30% of the course requirements remaining to opt for the INC grade at the end of the semester. The exact 30% requirement can be determined by discussing it with the instructor. For example, if there are 6 assignments and the student completed 4 full assignments. Similarly, if there are 8 labs, the student has 2-3 required labs that are not checked off. If a student chooses the INC option, the remaining work must be completed within 1 month of the end of the course.

For students with $> 30\%$ incomplete course requirements, we encourage them to drop the course and retake it next semester. We will not be able to provide the student with an INC grade, and they will automatically get a low grade or NC if they remain in the course. However, by retaking the course, students have the option to transfer their grades on the completed assignments to the next semester. Please let the instructor know that you plan to use this option.

Finally, not completing the course project will count as $> 30\%$ incomplete course requirement, and you will not have the option to use an INC. This is because a course project is a whole semester team effort and cannot be completed within the 1-month frame by an individual student for INC completion.

For students planning to graduate in the subsequent semester and have $> 30\%$ incomplete course requirements, please discuss this situation with an academic dean and the instructor.

Acknowledgments

Thanks to Tom Doeppner and Laura Dobler for the text on accommodation, mental health, and incomplete policy. Thanks to Karianne Bergen for the text on the policy for AI-generation tools usage. This document was adapted from earlier versions written by Daniel Ritchie and Chen Sun.