Syllabus: Natural Language Processing

Instructor: Greg Durrett, gdurrett@cs.utexas.edu

1 Course Description

Natural language processing (NLP) is a subfield of AI focused on solving problems that involve dealing with human language in a sophisticated way: these include information extraction, machine translation, automatic summarization, conversational dialogue, syntactic analysis, and many others. Much of the progress on these problems over the last 25 years has been driven by statistical machine learning and, more recently, deep learning. Recently, the emergence of the Transformer architecture has enabled improvements in systems by a dramatic scaling up of the compute invested in them and the amount of data they are trained on. Large Transformer language models such as ChatGPT have significantly advanced the state-of-the-art in the field and opened up a broad set of potential applications.

This class is intended to be a survey of modern NLP in two respects. First, it covers the main applications of NLP techniques today, both in academia and in industry, as well as enough linguistics to put these problems in context and understand their challenges. Second, it covers a range of modeling techniques including classification, pre-trained encoders, language models, sequence-to-sequence models, and statistical parsers. We study the models themselves, examples of problems they are applied to, inference methods, parameter estimation, and optimization. Programming assignments involve understanding how to build neural classifiers and language models from the ground up, culminating in a Transformer language model, as well as understanding how the data these systems are trained on impacts their functionality in practice.

2 Prerequisites

- One course in machine learning is recommended: the presentation of classification will go very quickly in the first few weeks
- Familiarity with Python (for programming assignments). These will assume you have an understanding of object definition and creation, field access, loops and list comprehensions, imports, and how to run scripts from the command line.
- Additional prior exposure to probability, linear algebra, optimization, linguistics, and NLP useful

3 Course Content

Course Material: All of the course videos are available on edX, and also mirrored at
https://www.cs.utexas.edu/~gdurrett/courses/online-course/materials.html

You will need to visit the edX site for: (1) this syllabus; (2) edX exercises; (3) walkthrough and debugging videos for each assignment.

You will need to visit the external site (the URL above) for: (1) readings accompanying the videos; (2) assignment PDFs and code releases.

You will also need to make use of **Canvas** and **Gradescope**. See below under "Grading", as these platforms are primarily used to grade assignments and track grades.

Office Hours: Office hours are scheduled as Zoom meetings on Canvas under the "Zoom" tab.

Ed Discussion: Ed Discussion is the primary way to ask questions and discuss the material with the course staff and other students. You can also post privately to instructors if you are including information you do not wish to share publicly. This platform operates nearly identically to Piazza. **See Canvas for a link.**

4 Assignments

The course primarily features four programming assignments and a final project designed to teach you to implement various models important for NLP. Grading of the assignments is primarily based on autograded system performance, with partial credit awarded based on the implementation if your system doesn't pass the autograder. The final project involves producing a written report that will be assessed by the course staff and your peers.

There is also a midterm exam, taken during a window of a few days, and a set of online exercises on edX, to be completed before the end of the semester.

(11%) Assignment 1: Sentiment Classification

(11%) Assignment 2: Feedforward Neural Networks

(11%) Assignment 3: Transformer Language Modeling

(25%) Midterm Exam (pen-and-paper, no coding)

(11%) Assignment 4: Factuality of ChatGPT

(25%) Final Project: Dataset Artifacts

(6%) Online Exercises

4.1 Grading

Assignments for this course will be submitted using a few different platforms:

1. **Gradescope:** Autograders for the assignments will be run through Gradescope. You will upload your code on this platform and get rapid feedback about whether or not it works and what performance it achieves.
2. **Canvas:** We will use Canvas as the main "hub" for your grades. We will also use this for any peer grading that we do, such as for the final project. **Assignment due dates and times in Canvas are the "official" due dates.**
3. **edX:** The edX platform itself will be used for online exercises. **You do not need to submit any code through edX, and the edX grades will not be updated throughout the semester.** If you see something assignment-related in edX that seems to disagree with Canvas (e.g., due dates), Canvas is always correct.

Midterm The midterm will be a timed exam uploaded on Gradescope, to be completed sometime in a 3-day window.

Late Assignments Assignments turned in late will lose 5% of the credit for that assignment per day that they are submitted late. For example, if you would've gotten a score of 100% on an assignment and you turn it in three days late, you will instead receive 85% of the credit.

Partial Credit Please note that although the assignments are autograded, we may still award you partial credit if you upload a partial solution. Uploading any code is better than none as it allows us to evaluate whether you have at least partially completed the assignment. However, it is up to you to decide whether to stop and submit an assignment or continue working past the due date and take the late penalty.

Final Grades Your final grade is computed based on the total points earned across all assignments. The final grade is mapped to a letter as follows, with grades on the boundary receiving the higher grade:

A	$100 - 93.3$
A−	$93.3 - 90$
B+	$90 - 86.7$
B	$86.7 - 83.3$
B−	$83.3 - 80$
C+	$80 - 76.7$
C	$76.7 - 73.3$
C−	$73 - 70$
D+	$70 - 65$
D	$65 - 60$
F	< 60

4.2 Compute Resources

Assignments are all designed to be doable on local hardware; they have all been developed, run, and debugged on a Macbook Pro. Use of GPUs can accelerate things for the later assignments. For the final project, you will very likely want to use Google Colab and may benefit from signing up for Colab Pro, but should not need to run lots of expensive experiments on cloud platforms. If you have access to your own GPUs or more substantial cloud resources, you can choose to use them in your project.

5 Course Schedule

See `https://www.cs.utexas.edu/~gdurrett/courses/online-course/materials.html` or edX for the detailed course schedule. The following is a week-by-week breakdown with assignments. Assignments are given out every two weeks; **see Canvas for assignment dates**.

The first six week cover basics of classification, neural networks, and Transformers necessary to understand the modern NLP stack. We then take a detour and cover methods for structured prediction and syntactic analysis. Finally, we discuss cutting-edge topics, including large language models, explanation methods, and a range of applications.

- **Week 1 (8/18/23):** Introduction and linear classification basics: perceptron, logistic regression, optimization, and sentiment analysis (Assignment 1 released)
- **Week 2 (8/25/23):** Multiclass classification and feedforward neural networks
- **Week 3 (9/1/23):** Word embeddings (Assignment 2 released)

- **Week 4 (9/8/23):** Language Modeling and Self-Attention
- **Week 5 (9/15/23):** Transformers and Decoding (Assignment 3 released)
- **Week 6 (9/22/23):** Pre-trained Transformers and seq2seq models
- **Week 7-8 (9/29/23):** Structured Prediction: Part-of-speech, Syntactic Parsing **(Midterm: October 12-15)** [note: two weeks of content are merged to allow for time for midterm review]
- **Week 9 (10/13/23):** Modern Large Language Models (Assignment 4)
- **Week 10 (10/20/23):** Explanations (FP released)
- **Week 11 (10/27/23):** Question Answering, Dialogue Systems (Final Project)
- **Week 12 (11/3/23):** Machine Translation, Summarization
- **Week 13 (11/10/23):** Multilinguality, Language Grounding, Ethical Issues in NLP

See Canvas for precise assignment due dates.

Readings Readings will be posted on the course site. They are primarily taken from open-access academic papers and websites, as well as Jacob Eisenstein's textbook *Natural Language Processing*, which is publicly available on GitHub.

6 Miscellaneous

6.1 Extensions

For all questions of extensions and accommodations, please email `onlinenaturallanguageprocessing@austin.utexas.edu`. Do not email the instructors directly.

Religious Holy Days: A student who cannot meet an assignment deadline due to the observance of a religious holy day may submit the assignment up to 24 hours late without penalty, if proper notice of the planned absence has been given. Notice must be given at least 14 days prior to the due date. For religious holy days that fall within the first 2 weeks of the semester, notice should be given on the first day of the semester. Notice should be emailed to the instructor and course staff.

Illness and Medical Extensions: Extensions may be granted in cases of illness (including COVID-19), medical emergency, or other circumstances. In all cases, the student should inform the course staff as soon as is practical, and the extension must be negotiated before the assignment's original due date.

6.2 Academic Honesty

Please read the department's academic honesty[1] policies. For this course, students are encouraged to discuss lecture material and coding assignments with others. However, **your final written source code or written work must be your own**. You are not allowed to collaborate with other students directly on code and submit

[1] https://www.cs.utexas.edu/academics/conduct

shared code as part of two more more students' submissions, unless this is explicitly allowed as in the case of the final project.

Finally, note that you may consult external resources such as blog posts, YouTube videos, academic papers, GitHub repositories, and more. However, your use of such resources, particularly GitHub repositories, must be limited in the same way as discussions with other students: you can look at these to get an idea of how to solve a problem, but you should not take external code and submit it as part of your assignment.

Be sure you respect these policies when posting on the discussion board. Asking clarifying questions, addressing possible bugs in the provided code, etc. are fair game, but you should discuss solutions in a substantive way that might spoil them for others. When in doubt, do not post large amounts of source code publicly to the class.

Students who violate these policies may receive a failing grade on the assignment in question or for the course overall, depending on the instructors' judgment and the severity of the infraction.

6.3 Policy on ChatGPT, Copilot, and other AI assistants

We encourage you to use ChatGPT and other related tools to understand concepts and as an assistant with the programming assignments in this class. Understanding the capabilities of these systems and their boundaries is a major focus of this class, and there's no better way to do that than by using them!

You are allowed to use these tools for programming assignments. However, **usage of ChatGPT must be limited in the same way as usage of other resources discussed above.** You should come up with the high-level skeleton of the solution yourself and use these tools primarily as coding assistants.

An example of a good question is, *"Write a line of Python code to reshape a Pytorch tensor x of [batch size, seqlen, hidden dimension] to be a 2-dimensional tensor with the first two dimensions collapsed."* Similar invocation of Copilot will probably be useful as well.

An example of a bad question would be to try to feed in a large chunk of the assignment code and copy-paste the problem specification from the assignment PDF. This is also much less likely to be useful, as it might be hard to spot subtle bugs.

As a heuristic, it should be possible for you to explain what each line of your code is doing. If you have code in your solution that is only included because ChatGPT told you to put it there, then it is no longer your own work in the same way.

Specific policies for usage of these tools on the midterm will be communicated closer to the exam date.

6.4 Disabilities

Students with disabilities may request appropriate academic accommodations from the Division of Diversity and Community Engagement, Services for Students with Disabilities[2] at 471-6259.

6.5 Program Inquiries

Please send any inquiries about the MCSO program to `MCSOGradCoordinator@austin.utexas.edu`. If you are in the MSDS program, you should send email to `MSDSGradCoordinator@utexas.edu`.

[2] On the web at `https://diversity.utexas.edu/disability/`