

Natural Language Processing (CS 7650)

Fall 2026

Instructor: Mark Riedl (riedl@cc.gatech.edu)

General Course Overview

Natural Language Processing (NLP) seeks to endow computers with the ability to intelligently process human language. NLP components are used in conversational agents and other systems that engage in dialogue with humans, automatic translation between human languages, automatic answering of questions using large text collections, the extraction of structured information from text, tools that help human authors, and many, many more.

This course will teach you the fundamental ideas used in key NLP components as well as current state-of-the-art practice in developing NLP algorithms.

Prerequisites

Students are expected to be proficient in the Python programming language. Course requirements include programming assignments written in the Python programming language using [Jupyter notebooks](#) on a personal computer or on [Google Colab](#) (or similar cloud development environment). The first programming assignment is an introduction to programming neural networks in Python and is essentially a warmup exercise that you can use to gauge your competence in Python programming. If you find the assignment very difficult, you may want to develop your Python programming skills a bit more before continuing in this course. The assignments get progressively more challenging.

Students are expected to have a background in data structures and have had some exposure to computational complexity (e.g., analysis of algorithmic complexity, finite automata). Having a basic understand of computational complexity will be very important for some of the later assignments where the auto-grading environment's memory and compute resources are limited (think nested loops vs vector operations and how the two relate to memory use and runtime – If you have no idea what this means, you might want to brush up on these concepts early in the course). Additionally, students are expected to have a background in basic probability, linear algebra, and calculus. Students are encouraged to have taken a course in artificial intelligence or machine learning. Of relevance are familiarity with linear classifiers, perceptrons, Naive Bayes, and logistic regression. While helpful, it is not assumed.

Course Materials and Resources

Online Textbook: [Natural Language Processing](#) (2018) by Jacob Eisenstein.

Computing Environment: Programming assignments in this course will be completed in Python using Jupyter notebooks. Most assignments can be run on a personal computer with a reasonably modern CPU. GPUs **are** not required for completing the coursework.

To ensure a consistent and supported experience for all students, the instructional team officially supports Google Colab. Google Colab is a cloud-based environment that runs Jupyter notebooks. Please note that Colab's free tier should be sufficient to complete all assignments, however please confirm.

Students are welcome to use other environments or services (e.g., local Jupyter Notebook/JupyterLab, other cloud notebook platforms), but these setups vary widely across systems and configurations. As a result, the instructional team will not provide technical support for installation, configuration, or troubleshooting outside of Google Colab. We will, however, help debug course content and code-related issues.

Gradescope requirement: Regardless of where you develop your solution (Colab or another environment), you will submit your work through Gradescope, and your submission must run without errors in the Gradescope environment. Submissions that do not execute successfully in Gradescope will receive no credit. Further details about submission requirements and the grading environment will be provided later in the course. Please allow ample time to test your solutions on the Gradescope environment. There are no limits to the number of times you can submit your assignment.

Canvas: Course materials, assignment downloads, quizzes, and exams.

Ed Discussions. For Official Announcements, and Forums for discussion. All class discussions will be on the Ed Discussion site. Here are some very specific guidelines for these discussions, which must be adhered to:

- All posts must be professional and cordial and about/related to the course material at hand.
- Students WILL NOT post specific answers to any of the assignments to Ed Discussion unless the TAs request it in a PRIVATE post.
- Before asking a question on the Forum, students should search for an answer to their questions. It most probably has been discussed already.
- The instructor team will attempt to answer all questions as soon possible. But please do NOT expect immediate responses. TAs are instructed to let students answer each other's questions too, as that supports more interactive learning. It is always wise to start assignments early, and to help others out when you can. Set the example you hope to benefit from later.
- Students can post anonymously to the class, but their IDENTITIES will be known by the instructor team.
- The instructor team is required to maintain privacy of all students, so please ensure that you communicate with them privately (using the private channels via Ed Discussion.)

- If there is a complaint about the class, please DO NOT post a public note to Ed Discussion. Please communicate directly with the instructor team. We will do our best to address it.

Gradescope. Students will download assignment instructions and materials from Canvas and submit their solutions to Gradescope for grading.

Grading

- **Quizzes:** All quizzes are proctored (10%)
- **Programming Assignments:** There will be 6 programming assignments:
 - 1: Introduction to neural networks (5%)
 - 2: Classification (10%)
 - 3a: Language Modeling, part a (10%)
 - 3b: Language Modeling, part b (10%)
 - 4: Distributional Semantics (10%)
 - 5: Mini-project (15%)
- **Midterm Exam:** A comprehensive **closed notes, closed book, and closed internet** proctored exam (15%)
- **Final Exam:** A comprehensive **closed notes, closed book, and closed internet** proctored exam (15%)

The total sum of quizzes is worth 10% of the final grade. Quizzes are graded by total points earned, not by averaging quiz percentages. This means longer quizzes count more than shorter quizzes. For example, an 8-point quiz has four times the impact of a 2-point quiz within the quiz category.

Programming Assignments

The primary goal of this class is to provide hands-on learning experiences that build natural language processing systems. We have broken these experiences into 6 programming assignments.

This course uses Python in Jupyter notebooks. To ensure a consistent experience for all students, the instructional team officially supports Google Colab, which requires no local installation.

You are welcome to use other environments (local Jupyter, VS Code, etc.), but these setups vary widely across operating systems and configurations. As a result, the instructor team cannot provide technical support for non-Colab environments. We will help debug your code, but not issues related to installation, packages, or local configuration.

The first assignment will familiarize you with the programming environment and Pytorch API (a python library for building and training neural networks).

Assignments 2, 3A, 3B, and 4 will walk you through the construction of increasingly more sophisticated natural language processing models, most of which will be based on neural networks. This will largely involve filling in the code for pre-defined functions.

The final programming assignment, the mini-project, will ask you to work with preparing a pre-determined dataset and building, training, and testing a working end-to-end AI system. The notebook will walk you through the steps but leave much of the logic to you.

All programming assignments except the last will be graded via autograder on Gradescope. The autograder will be provided as part of the notebook so that you can perform self-assessment. Grades will be determined by the same autograder. Any attempt to modify the autograder will result in an automatic zero. The mini-project programming assignment will be manually graded via code inspection. There will be written components in which some analysis and textual description of the solution is given; this will likewise be manually graded.

Submissions to Gradescope must run to completion without error on Gradescope. Please leave time to test your submissions and check the outputs closely for completion. It is possible that there will be differences between your local computing environment and Gradescope's computing environment due to, for example, default dependencies on your local machine that are not present on the Gradescope machine. Also, if GPUs are used, different GPUs on local machines versus Gradescope, which does not have GPUs, can result in different results. You should always check your Gradescope results (there are no submission limits.) Gradescope returns a status as to whether it is able to completely process the notebook or not. A "green light" does not indicate that you are getting the same results as any run on your local computing environment. A "green light" may also occur even if the autograder times out (which is not the same as a crash) or if the assignment code exceeds Gradescope memory limitations.

Midterm Exam

The midterm exam will be delivered on Canvas and proctored by HonorLock. It will be a timed test (1.5 hours) with multiple-choice and short answer questions. It will be **closed notes, closed book, and closed internet** (unless we choose to whitelist websites that you will need to answer a question). **See Remote Proctoring Notice below.**

Final Exam

The final exam will be delivered on Canvas and proctored by HonorLock. It will be a timed test (1.5 hours) with multiple-choice and short answer questions. It will be **closed-notes, closed book, and closed internet** (unless we choose to whitelist websites that you will need to answer a question). **See Remote Proctoring Notice below.**

Quizzes

Quizzes act as attention tests at the end of each module. Quizzes will involve multi-choice questions about the lecture materials and will be delivered via Canvas. Quizzes will be proctored

by HonorLock without the ability to upload or use notes, online resources, or access to the lecture videos. **See Remote Proctoring Notice below.** This is to encourage attention to the lecture materials. The Quizzes will all together total 10% of the grade. The value of each quiz will be proportional to the number of questions on the quiz—not all quizzes will have the same number of questions. We may elect to not have quizzes for particular units.

Remote Proctoring Notice (Canvas + Honorlock)

In this course, quizzes and exams will be delivered through Canvas and proctored using Honorlock (Georgia Tech's approved remote proctoring platform). Honorlock will record video and audio during the exam and may use automated tools to flag behavior that could indicate an academic integrity concern.

Because the quizzes and exams are recorded, you are responsible for choosing a location where you are comfortable being video and audio recorded. Before the quizzes and exams begin, you will be required to complete a scan of your test area using your camera. **This scan is required to include showing your desk/workspace, and the area around your computer.** Please plan ahead and select a space that is well-lit, quiet, and free from distractions or interruptions. You are not expected to take the exam in a dorm room or bedroom if that is not appropriate or comfortable—choose a setting that works for you and allows you to focus.

Quizzes and exams are closed book, closed notes, and closed internet. You may not access any outside resources during the exam unless the instructional team explicitly whitelists specific websites required for a question.

The workspace where you take the quizzes and exams must be clear of all books, papers, and any and all electronic devices except for the approved device you are using to take the proctored quiz or exam. During the exam, the following are strictly prohibited:

- Talking or communicating with anyone (in person, by phone, text, chat, or any other method)
- Using or accessing a cell phone, smartwatch, headphones/earbuds, or any unauthorized device
- Using notes, books, printed materials, second monitors, or unauthorized websites
- Using any AI tools (including ChatGPT or similar systems)
- Leaving the testing area or going out of camera view for any reason (including restroom breaks), unless an approved accommodation applies
- You must remain visible on camera for the entire exam. Looking down briefly to interact with your keyboard is expected; however, repeated or prolonged looking off-screen may be flagged and reviewed.

Important: Honorlock flags do not automatically mean you violated the rules. The instructional team will review all flags and recordings before any academic integrity action is taken.

Students registered with the Office of Disability Services (ODS) will be provided exam settings consistent with their approved accommodations (e.g., extended time, breaks, assistive technology). If you have accommodations, please ensure they are on file and communicated according to course procedures well in advance of the quizzes and exams.

Grading Scale

Grading Scale (for each assignment/unit and for the entire class).

- A: At or above 89.5%
- B: 79.5%-89.49%
- C: 69.5%-79.49%
- D: 59.5%-69.49%
- F: Below 59.5%

The grading scale has been adjusted to automatically include rounding up to the next highest grade if the final grade is within a half point of the grade category threshold. There is no curve for this course.

Late Day Policy

Over the course of the semester, you'll have a total of 5 “free” late days to submit programming assignments (except the final mini project). A late day is used one minute after the due date. A second late day is used 24 hours after that, and so on. Late days are determined by Gradescope submission timestamps. **Late days cannot be used on quizzes or exams or the HW5 mini-project.** Our intention is to give you some flexibility around your work commitments, family obligations, vacations, and the like without having to ask for permission.

Example: *Bob, the student, turns in homework 1 five minutes after the due date because he waited until the last minute and his home internet went out. He has officially used one late day. Bob has a big deliverable at work that coincides with the due date for homework 2. The timestamp on homework 2 shows it is 75 hours late. Bob has used an additional 4 late days, and his assignment is accepted and graded. Bob waits too long to start on homework 4 and with one hour to the deadline realizes that his homework, which runs fine on his souped-up home desktop, is using too much memory for Gradescope and is crashing. He frantically adjusts his model's hyperparameters but cannot get the right configuration and misses the deadline by an hour. His homework is marked as late because he has already used all five late days. Bob emails the instructional team and asks to “swap” the late day he used on homework 1 to be used on homework 4 and take a zero on homework 1 instead because it is worth fewer points. His request is denied.*

Additional rules:

- After five late days are used, homework assignments will be given a 0.

If you have a medical or family emergency, please contact the Dean of Students who may grant an exception to the late policy if your circumstances warrant it. We must receive approval from the Dean to grant a deadline extension. Please seek permission from the Dean of Students as soon as you know or suspect that the situation will impact your ability to complete coursework.

Regrading Policy

Regrade requests can be made for the Midterm exam, Mini-Project, and the Final exam via a private post to the instructor team on Ed Discussion. For midterm and final exam regrades, we will only regrade questions that are not automatically graded (i.e., are not multiple choice).

Please provide clear details as to why you are requesting a regrade. All regrade requests must be made within **48 hours** of the grade release. Exceptions will be considered if accompanied by an accommodation request from the Dean of Students.

To request a regrade, you must submit a clear, written explanation demonstrating why your answer is correct based on course materials. To ensure objectivity, your request will be evaluated by a different TA than the one who originally graded your assessment. If the review identifies a grading error, your score will be updated in Canvas and/or Gradescope; otherwise, the original grade stands.

Note on Escalation: The second TA's decision is final. There is no escalation path beyond this review, and subsequent requests to other members of the instructor team will not be considered.

Important Exception: For the Final Exam and Mini-Project, regrade requests will *only* be processed if the potential point adjustment is mathematically sufficient to change your final course letter grade.

Due Dates

All due dates will be on Canvas. All assignments will be due at 11:59 PM AoE per the Syllabus. Canvas will automatically adjust the date and time based on your local time zone. Please plan accordingly, especially around Daylight Savings changes in the US and in your location.

Honor Code

Georgia Tech aims to cultivate a community based on trust, academic integrity, and honor. Students are expected to act according to the highest ethical standards. The Georgia Tech Academic Honor Code applies to all aspects of this course. Plagiarism is a violation of the Academic Honor code. To plagiarize is defined by Webster's as "to steal and pass off (the ideas or words of another) as one's own; use (another's production) without crediting the source." Any student suspected of cheating or plagiarizing on a quiz, exam or assignment will be reported to the Office of Student Integrity, who will investigate the incident and identify the appropriate penalty for violations. For any questions involving these or any other Academic Honor Code issues, please consult us or <http://catalog.gatech.edu/policies/honor-code/>.

Collaboration Policy

Each individual student **MUST** submit their own work, done solely by themselves. We will be checking for code plagiarism, so **NO** copying code from the Web/Internet or AI – you are not allowed to use any form of AI to assist with the completion of assignments in this course. We understand you may not be familiar with some libraries and APIs presented in this class and you

will likely look up usage examples for individual functions. You may study these examples, but the code used in your assignments must be your own. Document your web/internet sources so that if our plagiarism detector is triggered we can determine if it is a false alarm. Some websites are expressly forbidden: you may not use any information from Chegg, CourseHero, Cliffsnotes, or any other website that is expressly set up to “help” students do their homework by providing solutions or connect you to people who assist. If you are uncertain, please ask. If you find yourself tempted, then you should be talking to the TAs. They are a free and helpful and misconduct-free resource.

To protect yourself and to protect others we recommend the following heuristics when communicating with others about course assignments:

1. Do not copy and paste your own code to share with someone else. If they in turn copy your code into their assignment, this will trigger plagiarism detectors.
2. Do not copy any code that you find on the internet. When more than one person in a class copies the same piece of code from the internet (for example from Stack Overflow), this triggers plagiarism detectors. Instead, use this as a learning episode: close your code, study the example, close down the example, and try to write it yourself. You will find this more valuable from a learning perspective than copying. Document your use of internet sources by adding a comment or markdown cells to your notebooks.
3. Do not share your screen when your Google Colab notebook is visible so that others can see, record, and screen capture.
4. If communicating in person or via video conferencing, use the “whiteboard policy”: write by hand (if possible) on a whiteboard app and erase the whiteboard afterward. Do not take a photo or screen capture. Only write in a human language, never code. For example:

```
Iterate through each point in the training data set:  
  Covert each data point into a vector.  
  Input your vectorized data point into the model.
```

Deviating from these heuristics does not automatically qualify as academic misconduct; however, following these heuristics greatly reduces the probability that your collaboration will not cross the line into misconduct.

As part of this course’s grading process, any **suspicion** of copying WILL be reported to the Office of Student Integrity for further analysis.

All students must also ensure that they DO NOT make any of the code for problem sets publicly available and are required to take steps to prevent future students from having access to it. Consequently, if you're using any version control systems such as git, please make sure that you mark your repositories as private.

You may not collaborate at all on the exams or quizzes. Students are not to discuss any questions or answers from the exams or quizzes with classmates or anyone else **even after deadlines have passed**.

Course Materials are Copyrighted

We do not allow course materials—videos, transcripts—to be publicly disseminated. All materials and assignments are copyrighted by Georgia Tech. You have permission to use them for the purpose of the class. Making videos, screenshots, or transcripts publicly available may trigger Georgia Tech to pursue legal copyright protection actions. Some of the course material has been made publicly available by the instructor in the form of blogs. You may link to these materials but not distribute them further without instructor permission.

With respect to course materials, you are prohibited from sending screenshots, video, or transcripts to a cloud-hosted LLM such as ChatGPT, Claude, Microsoft Copilot, etc. (an exception is made for Georgia Tech sanctioned enterprise versions of these services, if they exist). These services may use any prompts or material transmitted to them as part of future training. The inclusion of copyrighted material in prompts is thus considered an unauthorized copy and dissemination of materials to an unauthorized recipient, and therefore potentially illegal.

You **may** share your notes with other members of the class. A word of warning, however: the reason one takes notes is to engage mental resources with the course material. The process of taking notes is self-reinforcing. One must listen, understand, decide what is important enough to write down, figure out how to summarize and lay out your notes. Note taking ensures that you are engaged in active-listening instead of passive-listening. Relying on other people to take notes for you is not as effective a learning process as taking notes oneself. This is particularly important in this class since all quizzes and exams are closed notes, closed books, and closed internet.

Use of ChatGPT and other Large Language Models

Use of AI as a coding assistant or to assist with the answering of exam questions or quiz questions or to generate code to be turned in as part of an assignment is strictly **prohibited**. The purpose of this class is to help you build cognitive knowledge skills and reinforce coding skills. We are aware that some of our assignments will ask you to implement standard solutions like an LSTM. We are aware that there are packages that make it so one will never have to do this in real life. We also believe that one does not understand a system nearly as well as when one must work out the details by hand. You will get down into the weeds of neural networks and probability computations. You will understand what neural networks are doing in a way that you cannot learn by reading and watching lectures. The coding assignments were carefully chosen to reinforce your understanding. Coding is like lifting weights. Using AI to generate code for a class is like using a forklift to move the weights. Those weights do not need to be moved from one place to another. The change is happening inside you. No one regrets being fit. One might regret never being fit or losing that fitness. To use a crafting metaphor: you learn crafts by

following patterns at first. No one needs another pair of knit socks (I have some, thanks). But you experience all the little things that can go wrong that aren't in the tutorials and learn how to correct for them so that when you want to make something completely new, you will not flounder and fail. The same applies to coding assignments. They may not always be fun. But if AI is doing the coding for you, you may feel like you are learning by reading the generated code, but that is not how learning works. You are not learning in nearly the same way as having to think through the creation of every step of the algorithm. The little differences between lecture and working code are there for you to figure out. Problem solving means cognitive growth.

Your development environment software likely has an AI option. Google Colab has an AI coding assistant built in, for example. Disable coding assistants while working on the assignments.

You may want to ask questions about lecture material. We do recommend that you post to the discussion bulletin board. The instructor team would rather have you ask us than an AI. But if you do, make sure you:

1. Ask questions that do not involve any code. You may receive code anyway, in which case do not copy from the AI output. You might want to ask that code not be given in answers.
2. Never hit "Copy" within your conversation with an AI assistant. If you copy code from a LLM, you are committing academic misconduct since we have expressly forbidden it. If you copy your code into the LLM, it will generate code in return, so you will likely end up using the LLM's code, and now you must resist copying the LLM's code response not to mention we have expressly forbidden you from copying any copyrighted material into an LLM. Instead, use your interaction with the AI assistant as a learning experience, then let your assignment reflect your improved understanding.
3. Do not have your assignment and the AI agent open at the same time. Like above, use your conversation with the AI as a learning experience, then close the interaction down, open your assignment, and let your assignment reflect your revised knowledge. This heuristic includes avoiding using AI directly integrated into your composition environment: just as you should not let a classmate write content or code directly into your submission, so also you should not use tools that directly add content to your submission.
4. In the likely event that your code composition environment integrates language-to-code generation capabilities or code autocompletion, deactivate this functionality. The reason why this is important is because LLMs have a particular style and preference for how to write code. This will trigger our plagiarism detectors because when several people use an LLM they can get nearly identical code, even if they have never met. You may never have to write some pieces of code from scratch, but you will learn more by doing it yourself at least once.

Deviating from these heuristics does not automatically qualify as academic misconduct, and following these heuristics does not automatically make you free of the possibility of being flagged for academic misconduct.

AI usage has been endemic. We will use a variety of techniques in the course to flag the potential use of AI in homework, exams, and quizzes, including plagiarism detectors. People using the same AI for the same assignments will often produce identical code. We now have more than 2,000 reference solutions and flags are very high confidence. Even though there are only a few ways to solve any particular problem and a lot of guidance is given in the instructions, individual coding signatures are still very distinct.

Accommodations for Students with Disabilities

If you have learning needs that require special accommodation, contact the Office of Disability Services at 404-894-2563 or <http://disabilityservices.gatech.edu/> as soon as possible to make an appointment and discuss your special needs and to obtain an accommodations letter. Please also talk with us to discuss your learning needs. Accommodation letters should be shared with the instructor team via a private post on Ed Discussion within the first week of class. This is to ensure that we are applying accommodations consistently throughout the semester.

Student-Faculty Expectations Agreement

At Georgia Tech, it is important to strive for an atmosphere of mutual respect, acknowledgement, and responsibility between faculty member and the student body. Please see <http://catalog.gatech.edu/rules/22/> for an articulation of some basic expectations that you can have of us and that we have of you. These were adopted by both the faculty senate and the student government. In the end, simple respect for knowledge, hard work, and cordial interactions will help build the environment we seek. Therefore, we encourage you to remain committed to the ideals of Georgia Tech while in this class, especially during class discussion.

Inclusion

The Georgia Institute of Technology is committed to creating a campus free of discrimination on the basis of race, color, religion, sex, national origin, age, disability, sexual orientation, gender identity, or veteran status. We further affirm the importance of cultivating an intellectual climate that allows us to better understand the similarities and differences of those who constitute the Georgia Tech community, as well as the necessity of working against inequalities that may also manifest here as they do in the broader society.

If you have questions

We'll be using Ed Discussions in this course. Please use that forum to post questions and comments. The instructor team and other students will be of assistance there. There will be no live office hours during the semester.

If after contacting your TA and the instructor you do not feel your issue has been resolved, you may escalate the issue by emailing oms-advising@cc.gatech.edu.

Schedule

Course Schedule and Due Dates (Subject to Change)

The schedule below is provided to help you plan your work in the course. It is a **projected timeline**, not a fixed agreement, and dates may be updated as the semester progresses (e.g., to improve pacing, respond to class needs, or accommodate unforeseen events).

Official due dates are always the ones posted in Canvas, which serves as the authoritative source for deadlines and assessment timing. Any updates will be communicated via **Canvas** and/or **Ed Discussion**. Students are responsible for monitoring these channels for announcements and changes.

Week	Date	Module	Reading	Released	Due
1	Aug 24	1: Intro to NLP 2: Foundations	Ch 1 Appendix A	Aug 24: - Onboarding quiz - Module 1 quiz - Module 2 quiz Aug 28 - HW1 (neural nets)	Aug 30: - Onboarding quiz - Syllabus quiz
2	Aug 31	3: Classification	Ch 2, 3	Aug 31: Module 3 quiz	Sep 6: - HW1 - Module 1 quiz - Module 2 quiz - Module 3 quiz
3	Sep 7	4: Language models (Part 1)	Ch 6	Sep 7: - Module 4 quiz Sep 11: - HW2 (classification)	
4	Sep 14	4: Language models (Part 2)	Ch 6		Sep 20: Module 4 quiz
5	Sep 21	5: Semantics	Ch 14	Sep 21: - Module 5 quiz Sep 25: - HW 3a (language models)	Oct 4: - HW 2 - Module 5 quiz
6	Sept 28	6: Modern Neural architectures (Part 1)		Sept 28: Module 6 quiz	
7	Oct 5	6: Modern Neural		Oct 5:	Oct 11: HW 3a

Week	Date	Module	Reading	Released	Due
		architectures (Part 2)		HW 3b (language models)	Module 6 quiz
8	Oct 12	7: Information retrieval		Oct 12: - Module 7 quiz - Midterm exam opens	Oct 18: - Module 7 quiz - Midterm exam closes
9	Oct 19	8: Task-oriented dialogue	Ch 19	Oct 19: - Module 8 quiz Oct 23: - HW4 Released	Oct 25: - Module 8 quiz - HW3B
10	Oct 26	9: Summarization 10: Machine Reading	Ch 11	Oct 26: - Module 9 quiz - Module 10 quiz	Nov 1: - Module 9 quiz - Module 10 quiz
11	Nov 2	11: Open-Domain Question-Answering	Ch 18	Nov 2: - Module 11 quiz - HW5	Nov 8: - Module 11 quiz - HW4
12	Nov 9	12: Machine Translation		Nov 9: Module 12 quiz	Nov 15: Module 12 quiz
13	Nov 16	13: Agents and Reasoning		Nov 16: Module 13 quiz	Nov 22: Module 13 quiz
14	Nov 23	14: Privacy-Preserving NLP		Nov 23: Module 14 quiz	Nov 29: Module 14 quiz
15	Nov 30	15: Responsible AI		Nov 30: Module 15 quiz	Dec 6: - Module 15 quiz - HW5
16	Dec 7	End of Semester		Dec 4: Final Exam Opens	Dec 13: Final Exam Closes
17	Dec 14	End of Semester			

All due dates are at midnight AoE. For example, HW1 is due at 11:59pm AoE on Sunday, September 6th which is 7:00 AM Atlanta-time (ET) on Monday September 7th. But don't worry,

Canvas adjusts the due date based on your computer's locale - be sure your computer locale is set correctly.

Note that most deadlines are on Sundays AoE, but some deadlines are mid-week. We recognize that many OMS students work jobs and try to ensure that all assignments span weekends.

Syllabus Changes

We reserve the right to make changes to the syllabus at any time. We will notify the class of changes as early in advance as feasible via announcement on Ed Discussion. It is the student's responsibility to keep up to date with announcements.

Statement of AI Usage by Instructional Team

AI has not been used in the creation of any lecture materials, including assignments, exam questions, quiz questions, or this syllabus. The instructor team will not use AI to answer student questions on Ed or in email communication with students. There are no AI systems installed on Canvas by the instructor team.

We expect students will not use AI in their coursework. We hope (but do not require) that students will extend the same courtesy back to us and not use AI when communicating with the instructional team on Ed Discuss or email.