Ling 4400/2040: Computational Language Processing

Course Number: LING-4400 (graduate), LING-2040 (undergraduate)
Lecture: Tuesday and Thursday, 11:00 – 12:15
Location: St. Mary's 120

Instructor: Ethan Gotlieb Wilcox
Office: Poulton 248 (Office Hours: Monday, 11:00 am – 12:00 pm)
Email: ethan.wilcox@georgetown.edu
Website: https://wilcoxeg.github.io/

Teaching Assistant: Xiulin Yang
Office Hours: Friday, 11:00 am – 12:00 pm
Email: xy236@georgetown.edu

Summary: This course will introduce students to the basics of Natural Language Processing (NLP), a field that combines linguistics and computer science to produce applications, such as generative AI, that are profoundly impacting our society. We will cover a range of topics that form the basis of these exciting technological advances and will provide students with a platform for future study and research in this area. We will learn to implement symbolic representations such as finite-state techniques, *n*-gram models, and basic parsing in the Python programming language. Previous knowledge of Python is not required, but students should be prepared to invest the necessary time and effort to become proficient over the course of the semester. Students who take this course will gain a thorough understanding of the fundamental methods used in natural language processing, along with an ability to assess the strengths and weaknesses of natural language technologies based on these methods.

This course fulfills the college's *Quantitative Reasoning and Data Literacy* (QRDL) requirement as part of the Core Curriculum. This course counts for the interdisciplinary *Tech, Ethics and Society* minor, as well as for the college *Certificate in Artificial Intelligence*, in the "Science of AI" area.

Learning Outcomes:

- Students will learn how to write basic code in Python
- Students will learn how to debug Python code using a debugger
- Students will become familiar with the theoretical fundamentals of programming
- Students will become familiar with the fundamental techniques used in natural language processing, including string manipulation, finite state techniques, language modeling, and vector space models
- Students will gain practice thinking critically about the behavior, abilities, and limitations of current NLP tools, including large language models

Course Schedule

Week	**Module**	**Assignments Out & Due** Dates are *approximate*; refer to dates posted in Canvas assignments	**Readings** (all sections are inclusive)
Aug 31, 2026 (1 session, August 27th)	Introduction	8/27 Assignment 1 (installing Python)	
Aug 31, 2026	Intro to Python	9/1 Assignment 1 due 9/3 Assignment 2 (palindrome checker)	
Sep 7, 2026 (1 session, Sept 10th)	Processing text with Python		
Sep 14, 2026	Tokenization, Regular Expressions	9/15 Assignment 2 due 9/17 Assignment 3 (phone scraper)	Jurafsky & Martin, Chapter 2.0 - 2.2; 2.5 - 2.7
Sep 21, 2026	More Regular Expressions, Finite State Automata		Jurafsky & Martin (2007 edition) 2.2 - 2.2.4 (PDF available in Canvas)
Sep 28, 2026	Finite State Automata, Finite State Transducers	9/29 Assignment 3 due 9/29 Assignment 4 (regular expressions)	Jurafsky & Martin (2007 edition) 3.0 - 3.5 (PDF available in Canvas)
Oct 5, 2026	Introduction to Probability, Language Modeling	10/8 Assignment 4 due	Jurafsky & Martin, Chapter 3.0 - 3.1
Oct 12, 2026	Midterm review **Thursday, October 15th: Midterm Exam (in class)**		
Oct 19, 2026	Advanced Language Modeling, Intro to Neural Network Language Models	10/20 Assignment 5 (language modeling)	Jurafsky & Martin, Chapter 3.2 - 3.6 (3.7 is a fun bonus if you're interested!) Jurafsky & Martin, Chapter 6.0 - 6.5
Oct 26, 2026	Large Language Models for NLP	10/29 Assignment 5 due 10/29 Assignment 6 (part of speech tagging)	Jurafsky & Martin, Chapter 7 (optional)
Nov 2, 2026	Introduction to Markov Models, Sequence Tagging	11/5Assignment 6 due 11/5 Assignment 7 (named entity recognition)	Jurafsky & Martin, Chapter 18.0 - 18.4

Nov 9, 2026	Context-Free Grammars and Parsing	11/12 Assignment 7 due	Jurafsky & Martin, Chapter 19.0 - 19.5 (You can skip 19.4)
Nov 16, 2026	Vector Space Models	11/17 Assignment 8 (vector space models)	Jurafsky & Martin, Chapter 5.0 - 5.4
Nov 23, 2026 (1 session, Nov. 24th)	Vector Space Models	11/24 Assignment 8 due 11/24 Assignment 9 (practice final; not graded for correctness)	Jurafsky & Martin, Chapter 5.4 - 5.9
Nov 30, 2026	Naive Bayes Classifiers, Final Review	12/3 Assignment 9 due	Jurafsky & Martin, Appendix B.0 - B.8
Dec 17, 2026	**Final Exam: (12:30 - 2:30 pm)**		

Prerequisites: There are no prerequisites for this course. In particular, we will assume that students have no prior experience with programming. If you *do* have some limited prior experience with coding but not with NLP, this course is still likely appropriate for you. However, please speak with the course instructors if you have any questions or concerns.

Grade Breakdown

Midterm Exam	25%
Final Exam	35%
Homework Assignments	30%
● Coding Assignments	25%
● Readings	5%
Participation and Attendance	10%

Graduate vs. Undergraduate Section: Requirements for LING 4400 and LING 4020 are largely the same, with the following differences:

- Students enrolled in 4400 will be asked to complete additional questions on the homework assignments. Students in 4020 can complete these for extra credit, with a max homework grade of 100% (that is, additional points earned on homework won't "spill over" to boost midterm or final grades).
- Midterm and final exams will differ slightly between the sections, with slightly more challenging material for the 4400 exam. The number of questions and total number of points will remain the same between the exams.

Attendance: The TA will take attendance each class. You are permitted three absences, no questions asked; after that, your attendance grade will be lowered. You do not need to tell us in advance if you are

going to use one of your allowed absences. If you need to miss course for an approved event (medical appointment, medical emergency, family emergency, or religious observation), please email Ethan and the TA well in advance of the absence. For more information regarding the university policy on attendance, please see the academic standards section of the Undergraduate Student Bulletin (https://bulletin.georgetown.edu/regulations/standards/)

- Class time will involve personal and group coding sessions, so **please bring your laptop to class!**

Readings: Most of the course readings will be derived from Jurafsky and Martin, 3rd edition, which can be accessed online here: https://web.stanford.edu/jurafsky/slp3/. There are several readings from an older version of this text, which will be posted online in Canvas. Readings *not* from this book will also be posted online.

Assignments: Over the semester, students will complete

- *Programming Assignments:* Students will complete eight programming assignments. Assignments are due before the start of class on their specified due date. Submissions will be graded for correctness by the TA. One week after the assignment is due, we will review the solution for the assignment in class. Due dates for the original submission are given in the above course schedule; however, they may be adjusted throughout the semester.
- *Reading Responses:* Please submit a short response to the required reading (if there is one) each week. Reading responses can be very brief and should include two parts: (a) *Two questions* from the reading, or parts of the reading that you did not understand or would like further clarification about. You can also ask questions based on lecture material covered in class. (b) *One connection* between this week's material and material covered previously in class. How does this material build on or relate to a previous topic? Responses should be about 100 words and are due in Canvas by Friday at 5:00 pm of their associated week.
- *Practice Final:* The practice final is graded for completion only, not correctness.

- **Late Policy:** Late homework can receive credit, but the grade you can achieve is capped based on how late it is, with the maximum grade you can achieve lowered by 10% per 24-hour period after the due time. For homework, once we review the assignment in class, submissions are no longer permitted.

- **Group Work Policy:** The code that you submit as part of homework assignments *must be written by you*. However, working in groups on assignments can be extremely helpful. If you work with a group to come up with *pseudocode* (a high-level plan for how your program will run), or if you work with a group to understand and debug a piece of non-functioning code, that's OK. But you need to implement the pseudocode, or fix the bug yourself!

For more information on the University's honor code, please see https://honorcouncil.georgetown.edu/.

Midterm and Final: This course includes two pen-and-paper exams. The midterm and final exams will not require you to write code. Rather, they will involve commenting on or correcting code, explaining why and how code does (or doesn't) work, as well as explaining concepts and working with formulae we've covered in class. The TA will lead practice sessions in advance of each exam. The mid-term will take place during a class session.

AI Policy: The use of AI assistants, such as ChatGPT or Claude, as well as coding plug-ins like Copilot, are not explicitly forbidden. These tools are increasingly part of our world and are based on the technologies we will learn about in this class. Indeed, one of the learning outcomes is to encourage critical thinking about the abilities and limitations of such systems.

With that in mind, a warning: While LLM-based tools can be extremely effective, they suffer from two major drawbacks. First, they make mistakes. Second, they reduce the amount *you* have to think about and understand the code you run (that's the point!). If you are an advanced coder or an expert debugger, then you will be able to spot the mistakes that these systems make and fix them. However, for beginners, it's quite possible that correcting a buggy LLM-generated algorithm will take more time than writing the algorithm yourself. Furthermore, every time you rely on an LLM to generate something for you, that's one less opportunity you have to solidify the concepts we learn in lectures and to practice coding as a skill. Importantly, you won't be able to use LLMs during the midterm or the final. If you use these systems extensively in the beginning, you'll have a much harder time preparing for the exams.

Accommodations: If you have a recognized accommodation through ARC, please contact Ethan. More information about accommodations and support, including student-athlete support, can be found on the ARC website (https://academicsupport.georgetown.edu/)