Syllabus
Linguistics 227/627/Psych 327 - Language and Computation

Course Information

Meeting: Tues/Thurs 4:00-5:15pm, WLH 207

Instructor: Claire Moore-Cantwell, `claire.moore-cantwell@yale.edu`

Office Hours: Thursday, 3-4 pm. 301 Dow

Teaching Assistant: Natalie Schrimpf, `natalie.schrimpf@yale.edu`

Office Hours: Tuesday, 2:30-3:30 pm. 210 Dow

Overview

This course is an introduction to computational linguistics, the study of natural language from a computational perspective. Computational linguistics encompasses both applied (engineering) and theoretical (cognitive) issues, and in this course you will get a taste of both. You will learn how to write programs to automatically process and analyze linguistic structure in language corpora. You will learn how formal language models (grammars) can be implemented computationally and used to represent linguistic structure at various levels. You will use these formal language models to automatically analyze (assign structure to) linguistic data and to model grammaticality, and you will see how these models can be trained using language corpora. A major focus of the course will be on statistical techniques, especially Bayesian inference and maximum likelihood learning, because statistical approaches such as these are the foundation of much current work in computational linguistics, both theoretical and applied.

Note on prior programming experience: Programming is not the focus of this course, but knowing how to program is an essential skill needed to do computational linguistics. The first two weeks will be an intensive introduction to programming in Python, with a particular focus on learning how to process written text. This introduction will provide you with the tools needed to tackle subsequent programming assignments, which will involve implementing and analyzing language models of increasing complexity. Because this programming section will move very quickly and programming assignments will be complex, *prior programming experience is required*. If you don't have programming experience but are still interested in taking the course, please talk to me.

Requirements

Readings There will be a reading assignment corresponding to each topic. The reading should be completed by the date listed. Lectures will complement the reading - there will be a good deal of overlap, but lectures will not cover all the material from the readings. You will be responsible for both lecture and reading material.

Midterm and Final Exams There will be a midterm and final exam. The final will cover material from the second half of the course (it will not be cumulative). While the homework assignments primarily provide you with hands-on experience using computational techniques, the exams are an opportunity to demonstrate you have understood the concepts underlying these techniques.

Homework Assignments There will be seven homework assignments. All the homework assignments involve programming, and some will require you to do some on-paper exercises and/or some math. The assignments cover a range of computational linguistics tasks including corpus processing, (statistical) language modeling and generation, and implementation of complex algorithms.

Your assignments must be turned in electronically through the Classes*v2 server. You will generally have a week and a half to complete assignments, which will generally be due by midnight on Tuesdays or Thursdays.

Term Paper (grad students only) Graduate students will additionally work on a project in some area of computational linguistics, with a short term paper (10-15pp) describing the project due April 30th. The topic is fairly flexible: I'd like you to work on something you find really interesting. I encourage you to start thinking about the project early, but by no later than October 29th, you should submit a 1-2 page prospectus. I encourage you to come talk to me about your project ideas!

Policies

Grading Exams: 150 pts each, Homework assignments: 100 pts each (lowest grade dropped), Participation: 100 pts, Term paper (grads only): 200 pts

Lateness

Late assignments will not be accepted except with a dean's excuse. However, each of you is allotted 10 flex days (24 hours each) to extend the deadline of any homework assignments you choose. 10 days is a lot, but I advise you to use them sparingly in case you have an unexpected illness or absence later in the term or unanticipated trouble with one of the assignments.

Collaboration Programming is a solitary activity. You won't learn how to program unless you do it yourself. Therefore, you are required to write your own code and any other responses to homework assignments. However, I do encourage you to discuss the homework with one another, ask each other for help if you get stuck, and even compare the output of your programs to make sure you have no bugs. For non-coding portions of the homework, I encourage you to discuss the questions, but again, you must write up your own solutions. If you discussed your assignment with another student or students, list their name(s) on your assignment.

Required Texts

Main textbook (be sure to get the second edition!):

> Jurafsky and Martin. 2008. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition.* Second Edition.

Python book (available online through the Yale library):

> Hetland, Magnus Lie. 2008. *Beginning Python: from novice to professional.* Second Edition.

Recommended

Additionally, there are numerous online resources for python, and you may find them a convenient supplement to the Hetland book, especially if you're an experienced programmer.

`https://docs.python.org/2/tutorial/`

`https://www.codecademy.com/en/tracks/python`

If you have a specific problem or question, Stack Overflow is a good place to look for answers: `stackoverflow.com` I recommend searching the site for your question before asking it yourself.

Date	**Topic**	**Assignment**	**Reading**
3 Sep	Introduction, syllabus		J&M: Ch 1
8 Sep	Python 1: Intro	EX assigned	H: Ch 1
10 Sep	Python 2: Data types & Files	**EX due**	H: Ch 2-4, (11)
15 Sep	Python 3: Control & Reg Exp	HW 1 assigned (RegExp Syllables)	H: Ch 5,10 (pp242-258) J&M: 11.4
17 Sep	Python: Functions & More Files, plus Finite State Automata		H: Ch 6, 7, 11 J&M: Ch 2
22 Sep	Morphology and Min Edit Distance		J&M: Ch 3 (Sections)
24 Sep	Basic Probability, Noisy Channel	**HW1 due** HW 2 assigned (Spelling Correction)	Tesar 2007 (provided) J&M: 7.1-7.3
29 Sep	N-grams		J&M: Chapter 4
1 Oct	Smoothing		
6 Oct	Regression & Maximum Entropy	**HW 2 due** HW3 assigned (Ngram Phonotactics)	J&M: Ch 6 (6.6-6.7)
8 Oct	Computational Phonology		J&M: Ch 11
13 Oct	TBA		
15 Oct	Modeling Phonotactics	**HW 3 due** HW 4 assigned (Weighted Constraints)	Albright 2009 (provided)
20 Oct	HMMs & Viterbi		J&M: 5.5 & 6.1-6.5
22 Oct	*Fall Break*		
27 Oct	HMMs: Forward-Backward		
29 Oct		**Midterm**	
3 Nov	CFGs	**Prospectus due** (grads)	J&M: Ch 12-12.6
5 Nov	CFGs for Natural Language & Naïve parsing strategies	**HW 4 due** HW 5 assigned (Writing CFGs)	J&M: Ch 13-13.4
10 Nov	CFG Parsing: CKY & Earley		
12 Nov	PCFG and Probabilistic Parsing		J&M: Ch 14-14.7
17 Nov	Beyond CFG	**HW 5 due** HW 6 assigned (Parsing PCFGs)	J&M Ch 17.1-4, 18.1-2
19 Nov	Semantics		J&M: Ch 19
24-26 Nov	*Thanksgiving*		
1 Dec	Distributional Semantics and Clustering		J&M: Ch 20
3 Dec	Learning without labeled data	**HW 6 due** HW 7 assigned (Clustering)	
8 Dec	TBA		
10 Dec	Last Class: Wrap-up & revew	**HW 7 due**	
20 Dec	**7pm**	**FINAL EXAM**	