First half of the course:
- Introduction (Aug 23–25)
- Regular expressions (Aug 28–30)
- Edit distance (Sept 1)
- Information theory (Sept 6–8)
- Language models (Sept 11–18)
- Parts of speech and HMMs (Sept 18–27)
- Parsing (Sept 29–Oct 4)
- Review (Oct 6–9)
- Midterm (Oct 11)

Parsing unit:
- Constituents, parsing, and context free grammars (last week Friday)
- Recursive descent parsing (Monday, in lab)
- CKY parsing (Today)

Today:
- Limitations of top-down parsing
- Conceptual differences between top-down and bottom-up
- CKY parsing
  - Constraints: Chomsky Normal Form
  - Sample grammar
  - Practice by hand
  - CKY algorithm details
Four HMM problems:

Problem 0. Given $\tilde{O}$ together with $\tilde{S}$, compute $\lambda = (A, B, \pi)$ most likely to have produced those sequences.
[Solution: MLE, possibly with smoothing.]

Problem 1. Given $\lambda = (A, B, \pi)$ and $\tilde{O}$, compute the probability that $\lambda$ assigns to $\tilde{O}$.
[Solution: The forward algorithm.]

Problem 2. Given $\lambda = (A, B, \pi)$ and $\tilde{O}$, find $\tilde{S}$ that maximizes the probability that $\lambda$ assigns to $\tilde{O}$.
[Solution: The Viterbi algorithm.]

Problem 3. Given $\tilde{O}$, $M$ (or $V$), and $N$, find $\lambda = (A, B, \pi)$ that maximizes the likelihood of $\tilde{O}$.
[Solution: The Baum-Welch algorithm, a version of EM.]

The corpus isn’t tagged, but we tag the training text using NLTK’s tagger. (For our purposes in this project, we treat NLTK’s tagger as “correct,” although of course it’s not infallible.) The HMMTagger class is instantiated using this tagged version of the training text.

Once trained, we call `pos_tag()` on the HMMTagger object with the test text. Finally we tag the test text with nltk’s tagger also, and compare the results.

1. Finish the constructor for HMMTagger to compute the transition probabilities and emission probabilities (this is what I refer to as HMM “Problem 0”).

2. Implement the method `pos_tag()`, which computes the most likely sequence of POS tags for a given text, The corresponds to what is commonly called HMM “Problem 2,” and it is solved by the Viterbi algorithm.
These two approaches give rise to the two search strategies underlying most parsers: top-down or goal-directed search, and bottom-up or data-directed search. These constraints are more than just search strategies. They reflect two important insights in the western philosophical tradition: the rationalist tradition, which emphasizes the use of prior knowledge, and the empiricist tradition, which emphasizes the data in front of us.

The weakness in top-down parsers arises from the fact that they generate trees before ever examining the input. Bottom-up parsers, on the other hand, never suggest trees that are not at least locally grounded in the input.

Jurafsky and Martin, 2e, pg 429 & 432
Sentence → NounPhrase VerbPhrase

NounPhrase → AbsNounPhrase | ConcNounPhrase

AbsNounPhrase → That Sentence

ConcNounPhrase → CNPA RelativeClause | CNPA PrepositionalPhrase | CNPA

CNPA → PersonalPronoun | Article Nominal

Nominal → Adjective Nominal | Noun

RelativeClause → RelativePronoun VerbPhrase

PrepositionalPhrase → Preposition NounPhrase

VerbPhrase → VPA Adverb | VPA

VPA → VPB PrepositionalPhrase | VPB

VPB → Verb Adjective | Verb NounPhrase | Verb
the dog who barked won the prize
Coming up:

- Do HMMs & POS programming assignment (Wed, Oct 4)
- (Read J&M 17.(0-6). (Mon, Oct 2))
- Take CKY parsing quiz (Thurs, Oct 5)
- Do CKY parsing programming assignment (Mon, Oct 9)
- Take midterm (Wed, Oct 11)