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 $\bar{\mathcal{O}}$ together with $\bar{S}$, 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 $\bar{\mathcal{O}}$, compute the probability that λ assigns to $\bar{\mathcal{O}}$.
[Solution: The forward algorithm.]

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

Problem 3. Given $\bar{\mathcal{O}}$, M (or V), and N, find $\lambda = (A, B, \pi)$ that maximizes the likelihood of $\bar{\mathcal{O}}$.
[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.

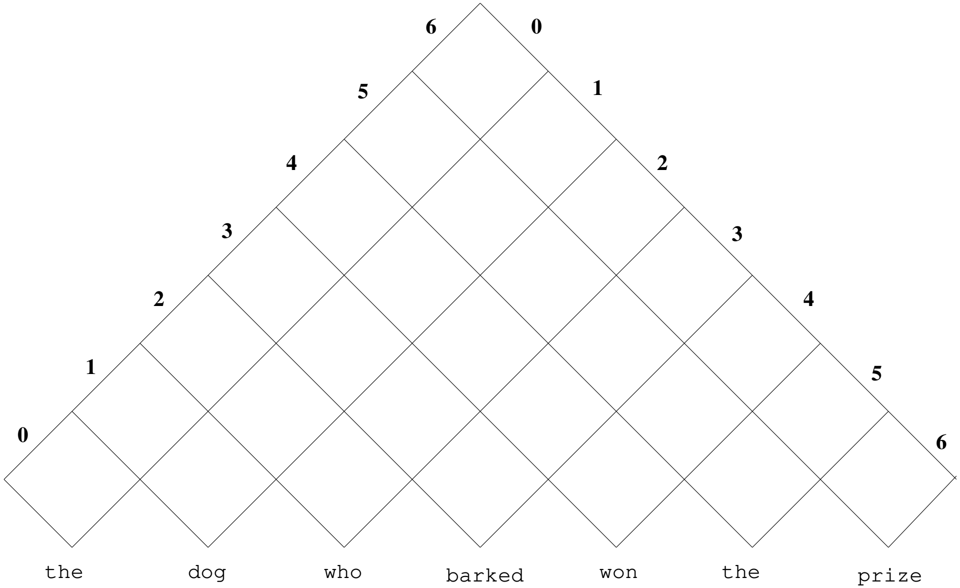
$$\begin{aligned}
\textit{Sentence} &\rightarrow \textit{NounPhrase Predicate} \\
\textit{NounPhrase} &\rightarrow \textit{ConcNounPhrase} \mid \textit{AbsNounPhrase} \\
\textit{ConcNounPhrase} &\rightarrow \textit{Article AdjectiveList Noun} \\
\textit{AdjectiveList} &\rightarrow \textit{Adjective AdjectiveList} \mid \varepsilon \\
\textit{AbNounPhrase} &\rightarrow \textit{That Sentence} \\
\textit{Predicate} &\rightarrow \textit{VerbPhrase VerbModifier} \\
\textit{VerbPhrase} &\rightarrow \textit{LinkingVerbPhrase} \mid \textit{TransitiveVerbPhrase} \mid \\
& \textit{IntransitiveVerbPhrase} \\
\textit{LinkingVerbPhrase} &\rightarrow \textit{LinkingVerb Adjective} \\
\textit{TransitiveVerbPhrase} &\rightarrow \textit{TransitiveVerb NounPhrase} \\
\textit{IntransitiveVerbPhrase} &\rightarrow \textit{IntransitiveVerb} \\
\textit{VerbModifier} &\rightarrow \textit{PrepositionalPhrase} \mid \textit{Adverb} \mid \varepsilon \\
\textit{PrepositionalPhrase} &\rightarrow \textit{Preposition NounPhrase}
\end{aligned}$$

[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 tress that are not at least locally grounded in the input.

Jurafsky and Martin, 2e, pg 429 & 432

$$
\begin{array}{rcl}
\mathit{Sentence} & \rightarrow & \mathit{NounPhrase\ VerbPhrase} \\
\mathit{NounPhrase} & \rightarrow & \mathit{AbsNounPhrase} \mid \mathit{ConcNounPhrase} \\
\mathit{AbsNounPhrase} & \rightarrow & \mathit{That\ Sentence} \\
\mathit{ConcNounPhrase} & \rightarrow & \mathit{CNPA\ RelativeClause} \mid \mathit{CNPA\ PrepositionalPhrase} \mid \mathit{CNPA} \\
\mathit{CNPA} & \rightarrow & \mathit{PersonalPronoun} \mid \mathit{Article\ Nominal} \\
\mathit{Nominal} & \rightarrow & \mathit{Adjective\ Nominal} \mid \mathit{Noun} \\
\mathit{RelativeClause} & \rightarrow & \mathit{RelativePronoun\ VerbPhrase} \\
\mathit{PrepositionalPhrase} & \rightarrow & \mathit{Preposition\ NounPhrase} \\
\mathit{VerbPhrase} & \rightarrow & \mathit{VPA\ Adverb} \mid \mathit{VPA} \\
\mathit{VPA} & \rightarrow & \mathit{VPB\ PrepositionalPhrase} \mid \mathit{VPB} \\
\mathit{VPB} & \rightarrow & \mathit{Verb\ Adjective} \mid \mathit{Verb\ NounPhrase} \mid \mathit{Verb}
\end{array}
$$


6
0
5
1
4
2
3
3
2
4
1
5
0
6
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)