Goals of this class

Goal:
▶ Investigate the effect on parsing by modifying the input and output of syntactic parsers
▶ Do not necessarily deal with internal models
▶ Effect could/should be positive, but could also reveal deeper insights about parsing and/or language

Focus on the linguistic representations of data given to & read from a parser

Specific topics

The specific topics will thus include:
- various linguistic modifications on the input side
- adding (de)projectivization
- optimal representations of linguistic phenomena
- interactions with semantic role labelers

Part of that will depend upon student interest

Generality

Focus on: linguistic structures of parser input/output
1. the types of techniques we will look at need to be very general, i.e., not parser-specific
2. techniques will have recourse to full parses

I am personally most interested in working with parser output

Theme #1

The impact of parse representations

Parse accuracy is not simply about how smart the parser is at making decisions
- Annotation schemes were developed to balance both internal and external criteria
- Some questions we’ll explore:
  ▶ What role does the representation of syntactic information play on parse accuracy?
  ▶ How can representations be converted from one form to another?

If representations affect parser accuracy, then we should consider changing them automatically:
- Are there latent properties or rule-based properties we can add to rebank annotation?
- What role does representation continuity play in improving parsing?

Theme #2

Manipulating parse input

1. Various linguistic modifications on the input side
   - adding (de)projectivization
   - optimal representations of linguistic phenomena
   - interactions with semantic role labelers

2. Techniques will have recourse to full parses

I am personally most interested in working with parser output
Parser models tend to be restricted in what information they can use because they don’t yet have a parse tree to lean on.

With parser output, we now have a linguistic structure to manipulate:

- What can we learn from parse output?
- How does access to full trees provide information about the quality of the parses? (e.g., is it non-well-formed in some way?)
- How do different parsers make different mistakes?
- How does the output interact with semantics?