Stack-Based Parsers

1 Conventions

- A stack is represented as a list of symbols, top of stack at left.
- Input is represented as a list of symbols, next to be read at left.
- Ellipsis indicates a sequence of symbols that is not relevant, e.g. a stack `(VP ...)` indicates that there is a VP on top of the stack and zero or more other symbols below it that we don't care about and we're not going to touch.
- A *configuration* is an ordered pair ⟨ Stack, Input ⟩.
- Greek letters are used to represent sequences of symbols within a (dotted) grammar rule that we don't care about.

2 Top-down (recursive descent)

Start: ⟨ (S), (a b a a ...) ⟩
Operations:

- Expand (applies a rule)

  ⟨ (X ...), (a b a a ...) ⟩ ⇒ ⟨ (Y Z ...), (a b a a ...) ⟩

  where $X \rightarrow YZ$ is a rule of the grammar.
- Match (applies a lexical rule)

  ⟨ (A ...), (a b a a ...) ⟩ ⇒ ⟨ (...), (b a a ...) ⟩

  if word `a` has part of speech `A`.

Successful configuration: ⟨ (), () ⟩

3 Bottom-up (shift-reduce)

Start: ⟨ (), (a b a a ...) ⟩
Operations:

- Shift

  ⟨ (...), (a b a a ...) ⟩ ⇒ ⟨ (A ...), (b a a ...) ⟩

  if word `a` has part of speech `A`.
- Reduce

  ⟨ (Z Y ...), (b a a ...) ⟩ ⇒ ⟨ (X ...), (b a a ...) ⟩

  if $X \rightarrow YZ$ is a rule of the grammar.

Successful configuration: ⟨ (S), () ⟩

4 Left-Corner

Start: $\langle$ (), (a b a a ...) $\rangle$
Operations:

- Shift

  $$\langle\ (\ldots), (\text{a b a a } \ldots)\ \rangle \quad \Rightarrow \quad \langle\ (\text{A} \rightarrow a\bullet, \ldots), (\text{b a a} \ldots)\rangle$$

  if word `a` has part of speech `A`.

- LC-predict

  $$\langle\ (\text{Y} \rightarrow \alpha\bullet, \ldots), (\text{b a a} \ldots)\rangle \quad \Rightarrow \quad \langle\ (\text{X} \rightarrow \text{Y} \bullet\beta, \ldots), (\text{b a a } \ldots)\ \rangle$$

  if $X \rightarrow Y\beta$ is a rule of the grammar.

- Complete

  $$\langle\ (\text{Z} \rightarrow \beta\bullet, \text{X} \rightarrow \text{Y} \bullet \text{Z}\Gamma, \ldots), (\text{a b } \ldots)\ \rangle \quad \Rightarrow \quad \langle\ (\text{X} \rightarrow \text{Y Z} \bullet\Gamma, \ldots), (\text{a b } \ldots)\ \rangle$$

Successful configuration: $\langle\ (\text{S} \rightarrow \Gamma\bullet), ()\rangle$

5 Earley's Algorithm

Note that there are two errors in Jurafsky and Martin's Figure 10.13:

- In SCANNER, the word being scanned is word[j+1], hence the dotted rule you add should be $B \rightarrow$ word[j+1] and it should be added to chart[j+1].
- In COMPLETER, the new state should get added to chart[k].