11.1 Signs

So far, we’ve seen a formal representation of rules that can be used in planning. To obtain these from natural language sentences, we’ll need to define signs, which have:

- **signified** structures (edges labeled sig) – these are our planning rules;
- **syntactic categories** (edges labeled 0) – we’ve seen these already (N-aD, V-aN, etc.);
- **syntactic arguments** (labeled 1’, 2’, etc., from signified), connecting semantic arguments;
- **apex/base** associations (labeled A, and B), connecting derivation fragments on the store.

For example, here’s a lexical sign for the word *give*, defined to mean *cause to have*:

And here’s a store of signs after the word *Most* in the sentence *Most large pumps work*: 
11.2 Processing

Comprehension proceeds as follows, using simplified pushdown automaton operations:

1. a **lexical** decision is made about whether to **match** store elements at the next word, and
   a **lexical inference rule** is applied (choose a meaning for the next word).

   (a) no-match ($L\downarrow$):

   

   (b) yes-match ($L\uparrow$):

   

2. a **grammatical** decision is made about whether to **match** store elements at the next rule, and
   a **grammatical inference rule** is applied (choose a rule to compose the next subtree).

   (c) no-match ($G\downarrow$):

   

   (d) yes-match ($G\uparrow$):

   

2
11.3 Lexical inference rules

Lexical inference rules add lexical signs.

(Quantified noun ‘everything’ highlights how constraints are applied in modifiers and arguments.)

a) Every BeingAThing

\[\begin{align*}
\text{N} & \quad \text{sig} \\
\text{x} & \quad \text{rin} \\
\text{y} & \quad 0 \\
\text{z} & \quad 0
\end{align*}\]

b) Working

\[\begin{align*}
\text{V-aN} & \quad \text{sig} \\
\text{x} & \quad \text{rin} \\
\text{1'} & \quad 0 \\
\text{y} & \quad \text{sig}
\end{align*}\]

c) BeingHere

\[\begin{align*}
\text{A-aN} & \quad \text{sig} \\
\text{x} & \quad \text{rin} \\
\text{1'} & \quad 0 \\
\text{y} & \quad \text{sig}
\end{align*}\]

11.4 Grammatical inference rules

First we need rules to attach arguments:

Aa:

\[\begin{align*}
\text{τφ_{1..n-1}} & \quad \text{sig} \\
\text{x} & \quad \text{rin} \\
\text{y} & \quad n' \\
\text{z} & \quad 0
\end{align*}\]

Ab:

\[\begin{align*}
\text{τφ_{1..n-1}} & \quad \text{sig} \\
\text{x} & \quad \text{rin} \\
\text{y} & \quad n' \\
\text{z} & \quad 0
\end{align*}\]

These rules attach constraints to the ‘nuclear scopes’ of the quantified noun phrase:

\[\begin{align*}
\text{S} & \quad \text{V-aN} \\
\text{N} & \quad \text{sig} \\
\text{x} & \quad \text{rin} \\
\text{y} & \quad 0 \\
\text{z} & \quad 0
\end{align*}\]
Then we need rules to attach modifiers:

Ma:

These rules attach constraints to the ‘restrictors’ of quantified noun phrases:

There are several other rules as well, covered in more detail here:

https://linguistics.osu.edu/sites/linguistics.osu.edu/files/tech-rept-casp_0.pdf

References