Ling 5701: Lecture Notes 9
Context-Free Grammars

9.1 We can use CFGs to model natural languages like English

Do natural languages have natural category types that we can use in CFG rules? Yes!
First, observe that natural languages use different argument structures for different verbs:

- They sleep. – one argument ahead (intransitive)
- They find pets. – one argument ahead and one argument behind (transitive)
- They give people pets. – one argument ahead, two arguments behind (ditransitive)

Next, observe natural languages *coordinate* conjunctions (combine like types): $\langle\alpha\rangle \rightarrow \langle\alpha\rangle$ and $\langle\alpha\rangle$.

- $[_\beta$ $[_\beta$ They find people] and $[_\beta$ they find pets]]. – sounds ok (β is sentence)
- They find $[_\gamma$ $[_\gamma$ people] and $[_\gamma$ pets]]. – sounds ok (γ is noun phrase)
- *They find $[_\gamma$ $[_\beta$ they find people] and $[_\gamma$ pets]]. – sounds wrong; conjuncts must match

Now, allowable conjunctions give us insight into the category structure of language:

- They $[_\delta$ $[_\delta$ sleep] and $[_\delta$ find pets]]. – sounds ok (δ is verb phrase)
- They $[_\eta$ $[_\eta$ find] and $[_\eta$ give people]] pets. – sounds ok (but what's η?)

Transitive verbs (find) match type with ditransitive verb + indirect object (give people)!
Both lack argument ahead and behind – it seems types are defined by missing arguments!

Formalize set of categories C as follows – clauses with various unmet requirements:

1. every U is in C, for some set U of primitive categories;
2. every $C \times O \times C$ is in C, for some set O of type-combining operators;
3. nothing else is in C

Define primitive categories $U = \{\mathbf{N}, \mathbf{V}\}$:

- **N**: noun-headed category with no missing arguments (noun phrase)
- **V**: verb-headed category with no missing arguments (sentence)

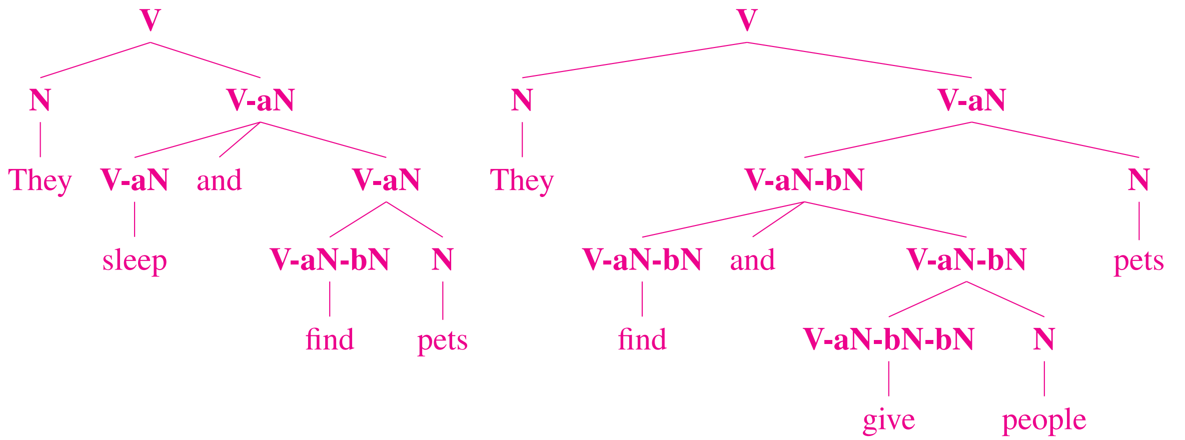
Define type-combining operators $O = \{\textbf{-a}, \textbf{-b}\}$:

- $\langle\alpha\textbf{-a}\beta\rangle$: α lacking β argument ahead (e.g. **V-aN** for intransitive δ above)
- $\langle\alpha\textbf{-b}\beta\rangle$: α lacking β argument behind (e.g. **V-aN-bN** for transitive η above)

Now we can define CFG rules R over these categories:

- $\langle\alpha\rangle \rightarrow \langle\beta\rangle\ \langle\alpha\textbf{-a}\beta\rangle$: argument attachment ahead
- $\langle\alpha\rangle \rightarrow \langle\alpha\textbf{-b}\beta\rangle\ \langle\beta\rangle$: argument attachment behind
- $\langle\alpha\rangle \rightarrow \langle\alpha\rangle$ and $\langle\alpha\rangle$: conjunction

These three rules model all of the above sentences:



Also note that the parents in these rules all have simpler types than the children.
This means for any lexicon (constraining types at tree leaves), the set of categories C is finite.

9.2 Limits of CFGs

Natural languages may also use non-local dependencies.
In English, these show up in topicalization, which seem to use a gap '_' at one argument:

- These pets, you say they found _.

These coordinate as well, but our test shows categories with gaps differ from those without:

- These pets, you [$_\delta$ [$_\delta$ say they found _] and [$_\delta$ think _ gave people joy]]. – sounds ok
- *These pets, you [$_\delta$ [$_{\textbf{V-aN}}$ say they found pets] and [$_\delta$ think _ gave people joy]]. – wrong

We can model this by adding a new type-combining operator for non-local dependencies:

- $\langle\alpha\textbf{-g}\beta\rangle$: α lacking non-local β argument (e.g. **V-aN-gN** for intransitive δ above)

and adding rules to introduce non-local dependencies:

- $\langle\alpha\textbf{-g}\beta\rangle \rightarrow \langle\alpha\textbf{-a}\beta\rangle$: introduce non-local dependency to argument ahead
- $\langle\alpha\textbf{-g}\beta\rangle \rightarrow \langle\alpha\textbf{-b}\beta\rangle$: introduce non-local dependency to argument behind

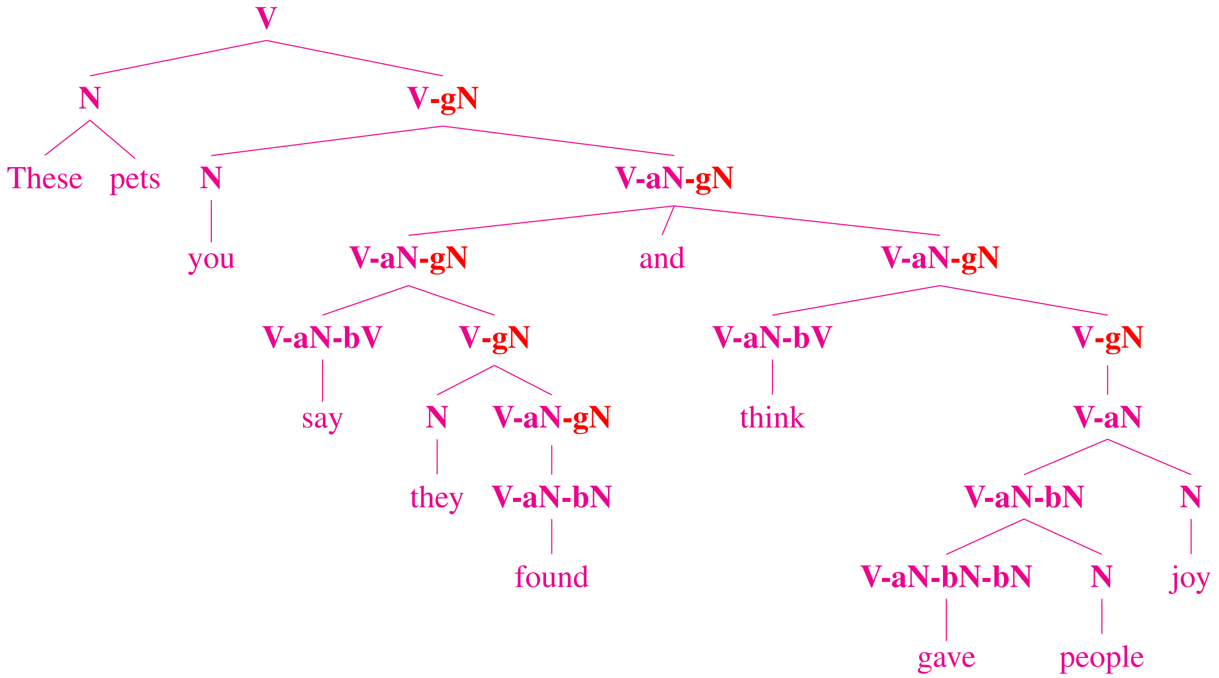
and adding rules to attach non-local dependencies:

- $\langle\alpha\rangle \rightarrow \langle\beta\rangle\ \langle\alpha\textbf{-g}\beta\rangle$: non-local dependency attachment

and modifying existing rules to propagate non-local dependencies $\psi_m \in \{\textbf{-g}\} \times C$:

- $\langle \alpha \psi_{1..M} \rangle \rightarrow \langle \beta \psi_{1..m} \rangle \; \langle \alpha\text{-}\mathbf{a}\beta \psi_{m+1..M} \rangle$: argument attachment ahead, with propagation
- $\langle \alpha \psi_{1..M} \rangle \rightarrow \langle \alpha\text{-}\mathbf{b}\beta \psi_{1..m} \rangle \; \langle \beta \psi_{m+1..M} \rangle$: argument attachment behind, with propagation

Here's the analysis:



Note that M above is unbounded, so our rules no longer guarantee a finite set of categories. (Any number of arguments may be extracted and propagated up from children.)

Some use evidence like this to argue language isn't context-free but mildly context-sensitive [Shieber, 1985, Joshi, 1985, Steedman, 2000].

In practice, though, we can just constrain category sets to combinations seen in training data.

References

[Joshi, 1985] Joshi, A. K. (1985). How much context sensitivity is necessary for characterizing structural descriptions: Tree adjoining grammars. In D. Dowty, L. K. and Zwicky, A., editors, *Natural language parsing: Psychological, computational and theoretical perspectives*, pages 206–250. Cambridge University Press, Cambridge, U.K.

[Shieber, 1985] Shieber, S. (1985). Evidence against the context-freeness of natural language. *Linguistics and Philosophy*, 8:333–343.

[Steedman, 2000] Steedman, M. (2000). *The syntactic process*. MIT Press/Bradford Books, Cambridge, MA.