12. LA- and PS-hierarchies in comparison

12.1 Language classes of LA- and PS-grammar

12.1.1 Complexity degrees of the LA- and PS-hierarchy

	LA-grammar	*PS-grammar*
undecidable	—	recursively enumerable languages
exponential	A-languages B-languages C3-languages	context-sensitive languages
polynomial	C2-languages	context-free languages
linear	C1-languages	regular languages

12.2 Subset relations in the two hierarchies

12.2.1 Subset relations in the PS-hierarchy

regular lang. $\subset$ context-free lang. $\subset$ context-sensitive lang. $\subset$ rec. enum. languages

12.2.2 Subset relations in the LA-hierarchy

C1-languages $\subseteq$ C2-languages $\subseteq$ C3-languages $\subseteq$ B-languages $\subset$ A-languages

12.3 Non-equivalence of the LA- and PS-hierarchy

12.3.1 Languages which are in the same class in PS-grammar, but in different classes in LA-grammar

$a^k b^k$ and WW^R are in the same class in PS-grammar (i.e. context-free), but in different classes in LA-grammar: $a^k b^k$ is a C1-LAG parsing in linear time, while WW^R is a C2-LAG parsing in n^2.

12.3.2 Languages which are in the same class in LA-grammar, but in different classes in PS-grammar

$a^k b^k$ and $a^k b^k c^k$ are in the same class in LA-grammar (i.e. C1-LAGs), but in different classes in PS-grammar: $a^k b^k$ is context-free, while $a^k b^k c^k$ is context-sensitive.

12.3.3 Inherent complexity

The inherent complexity of a language is based on the number of operations required in the worst case on an abstract machine (e.g. a Turing or register machine). This form of analysis occurs on a very low level corresponding to machine or assembler code.

12.3.4 Class assigned complexity

The complexity of artificial and natural languages is usually analyzed at the abstraction level of grammar formalisms, whereby complexity is determined for the grammar type and its language class as a whole.

12.3.5 Difference between the two types of complexity

Languages which are inherently of high complexity (e.g. 3SAT and SUBSET SUM) are necessarily in a high complexity class (here exponential) in any possible grammar formalism.

Languages which are inherently of low complexity (e.g. $a^k b^k c^k$) may be assigned high or low class complexity, depending on the formalism.

12.3.6 PS-Grammar of L_{no}

$S \rightarrow 1S1$ $\quad S \rightarrow 1S$ $\quad S \rightarrow \#$

$S \rightarrow 0S0$ $\quad S \rightarrow 0S$

12.3.7 PS-grammar derivation of 10010#101 in L_{no}

derivation tree:	generated chains:	states:	
S			
		1.S1	1S1.
1 S 1	1S1	1.S	
		0.S0	0S0.
0 S 0	10S01	0.S	
		0.S0	
0 S	100S01	0.S	0S.
		1.S1	1S1.
1 S 1	1001S101	1.S	
		0.S0	
0 S	10010S101	0.S	0S.
#	10010#101	#.	

12.3.8 C3-LAG for L_{no}

LX $=_{def}$ {[0 (0)], [1 (1)], [# (#)]}
$ST_S =_{def}$ {[(seg_c) {r_1, r_2, r_3, r_4, r_5}] }, where seg_c, seg_d ϵ {0, 1}.

r_1: (seg_c)(seg_d)	$\Rightarrow \varepsilon$	{r_1, r_2, r_3, r_4, r_5}
r_2: (seg_c)(seg_d)	$\Rightarrow$ (seg_d)	{r_1, r_2, r_3, r_4, r_5}
r_3: (X)(seg_c)	$\Rightarrow$ (X)	{r_1 r_2, r_3, r_4, r_5}
r_4: (X)(seg_c)	$\Rightarrow$ (seg_c X)	{r_1 r_2, r_3, r_4, r_5}
r_5: (X)(#)	$\Rightarrow$ (X)	{r_6}
r_6: (seg_c X)(seg_c)	$\Rightarrow$ (X)	{r_6}

$ST_F =_{def}$ {[ε rp_6]}

12.4 Comparing the lower LA- and PS-classes

Context-free PS-grammar has been widely used because it provides the greatest amount of generative capacity within the PS-grammar hierarchy while being computationally tractable.

12.4.1 How suitable is context-free grammar for describing natural and programming languages?

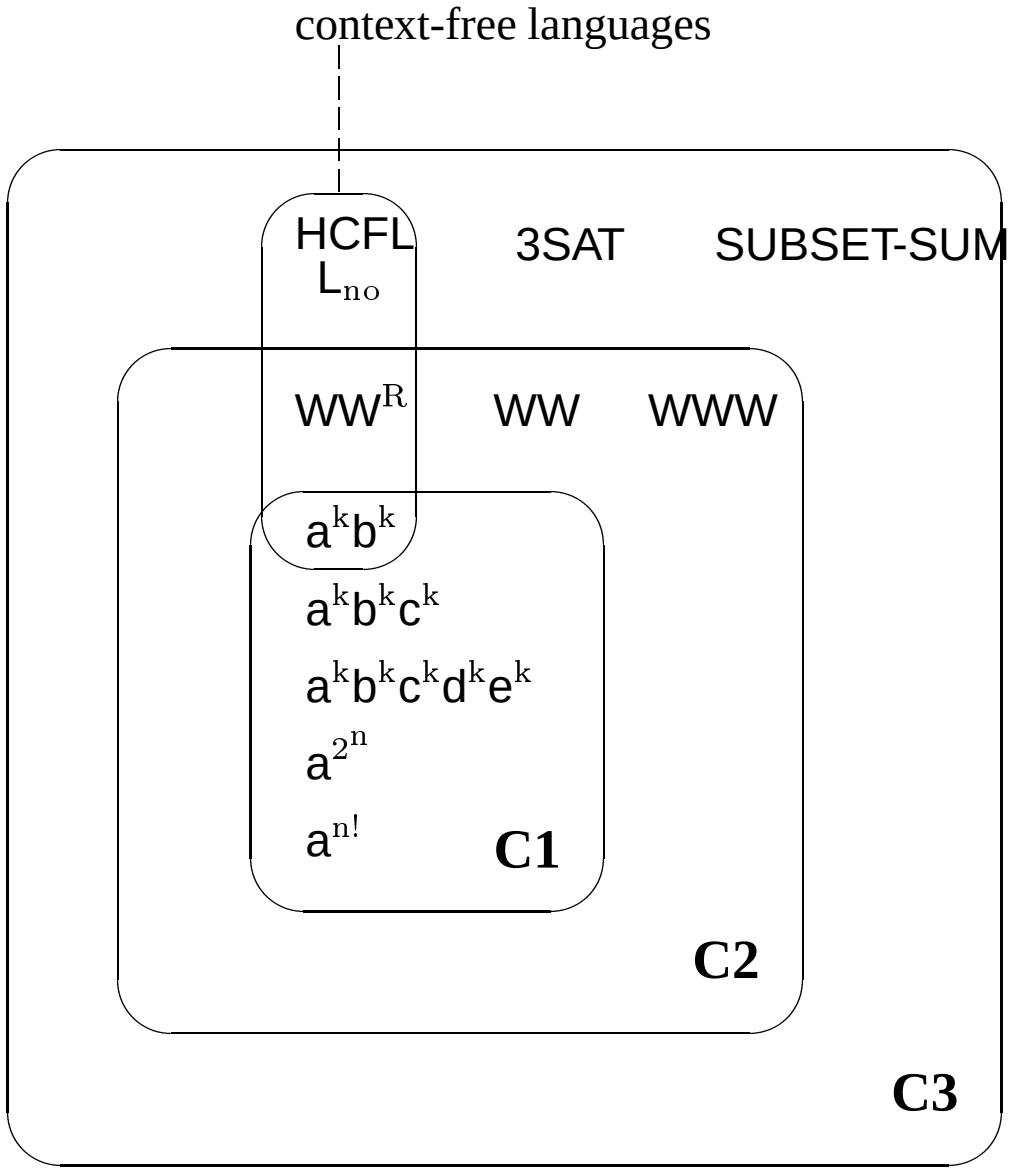
There is general agreement in linguistics that context-free PS-grammar does not properly fit the structures characteristic of natural language. The same holds for computer science:

> It is no secret that context-free grammars are only a first order approximation to the various mechanisms used for specifying the syntax of modern programming languages.
>
> S. Ginsberg 1980, p.7

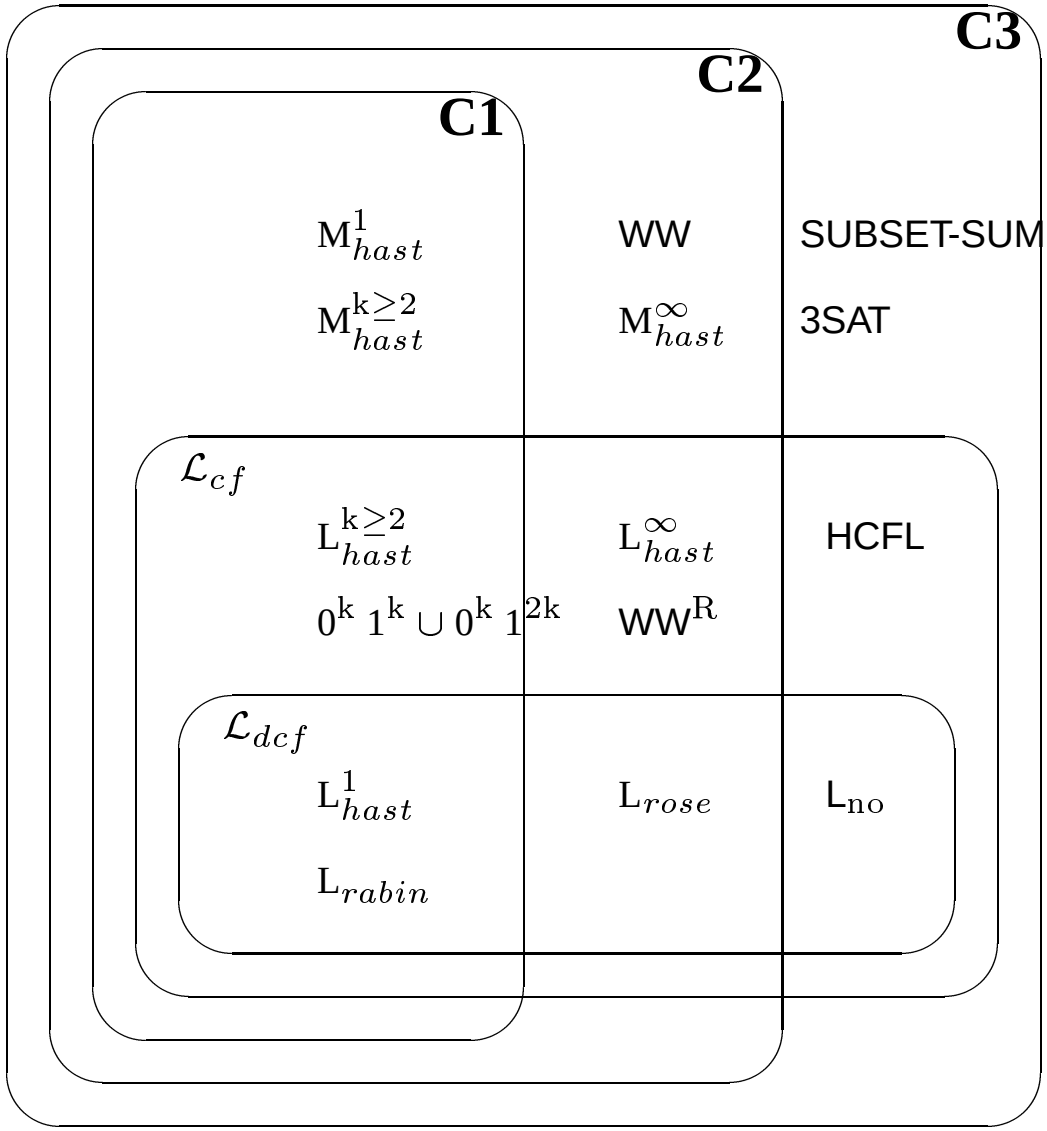
12.4.2 Conservative extensions of the PS-grammar hierarchy

regular lang. $\subset$ context-free lang. $\subset$ TAL $\subset$ index lang. $\subset$ context-sensitive lang. $\subset$ r.e. languages

12.4.3 Orthogonal relation between C- and cf-languages



12.4.4 Orthogonal $\mathcal{L}_{dcf}$, $\mathcal{L}_{cf}$, C_1, C_2, and C_3 classifications



12.5 Linear complexity of natural language

12.5.1 Why the natural languages are likely to be C-languages

In a context-sensitive language which is not a C-language, the category length would have to grow just within the LBA-definition of context-sensitive languages, but grow faster than the pattern-based categorial operations of the C-LAGs would permit.

That this type of language should be characteristic for the structure of natural language is highly improbable.

12.5.2 If the natural languages are C-LAGs

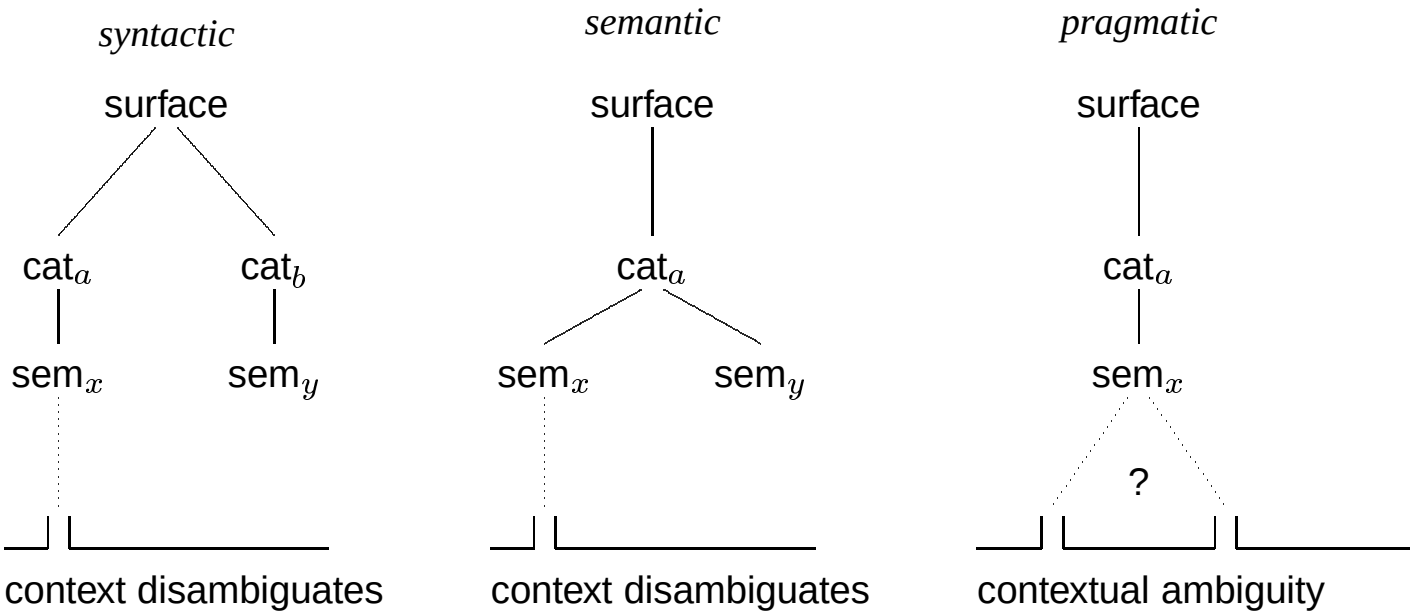
then the following two questions are equivalent:

(i) *How complex are the natural languages?*

(ii) *How ambiguous are the natural languages?*

This is because the C-LAG subclasses differ solely in their degrees of ambiguity.

12.5.3 SLIM-theoretic analysis of ambiguity

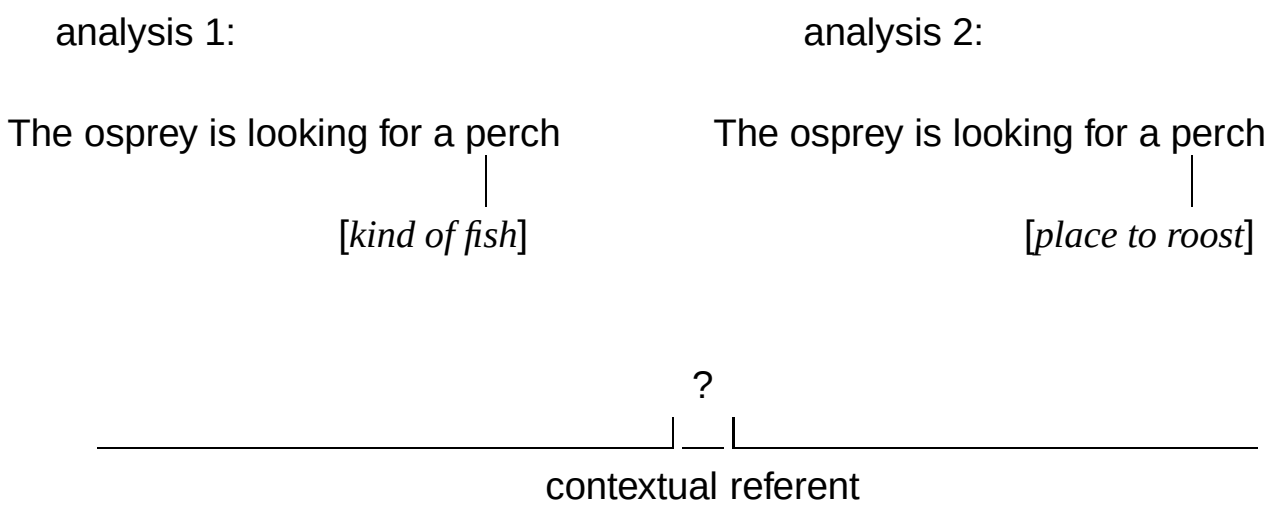


12.5.4 Multiple interpretations of prepositional phrases: a syntactic or a semantic ambiguity?

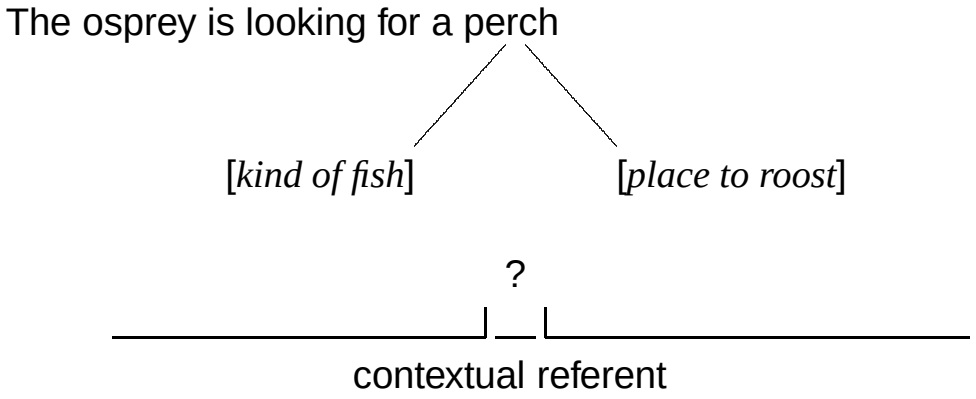
The man saw the girl with the telescope.

Julia ate the apple on the table behind the tree in the garden.

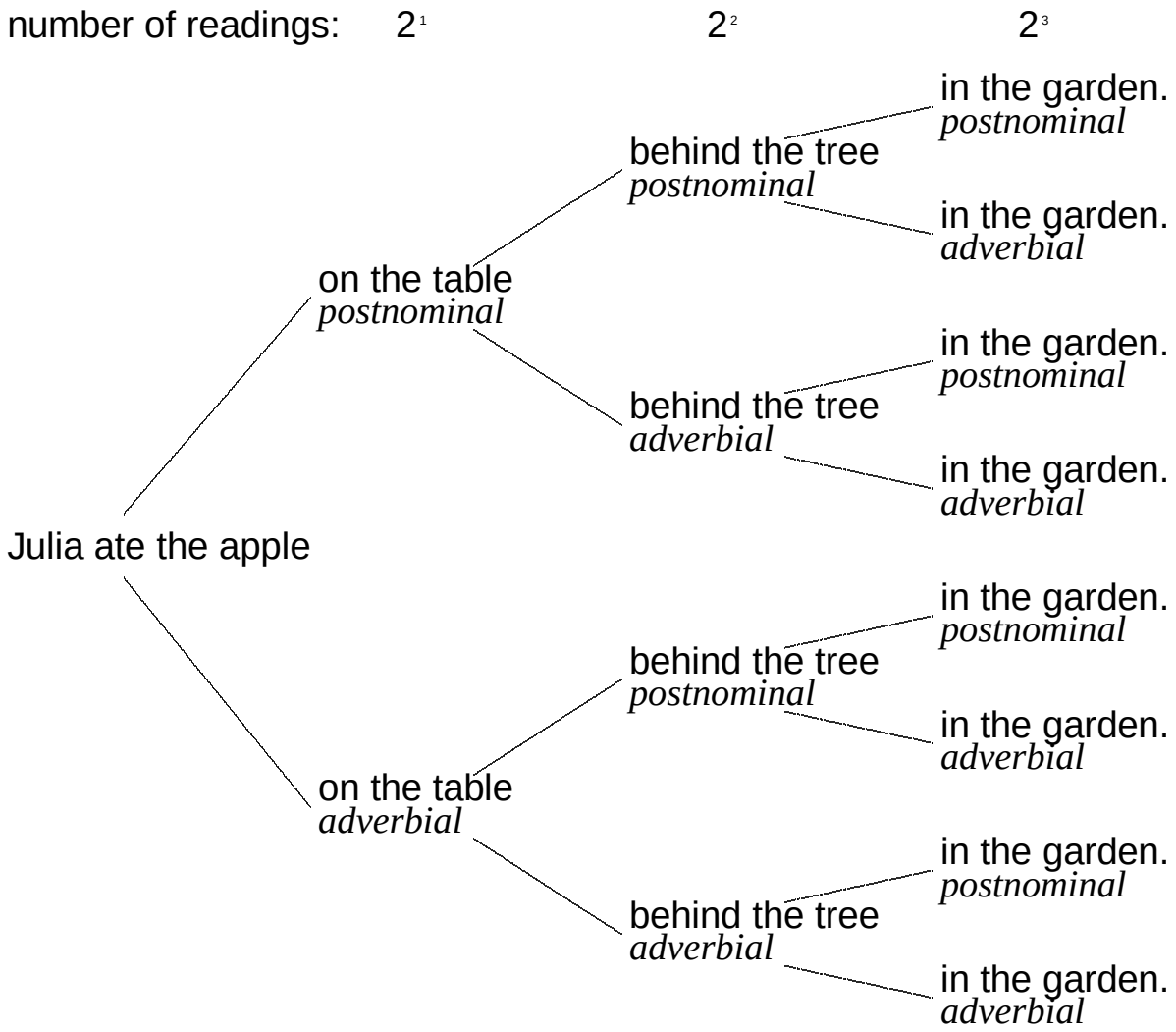
12.5.5 Incorrect analysis of a semantic ambiguity



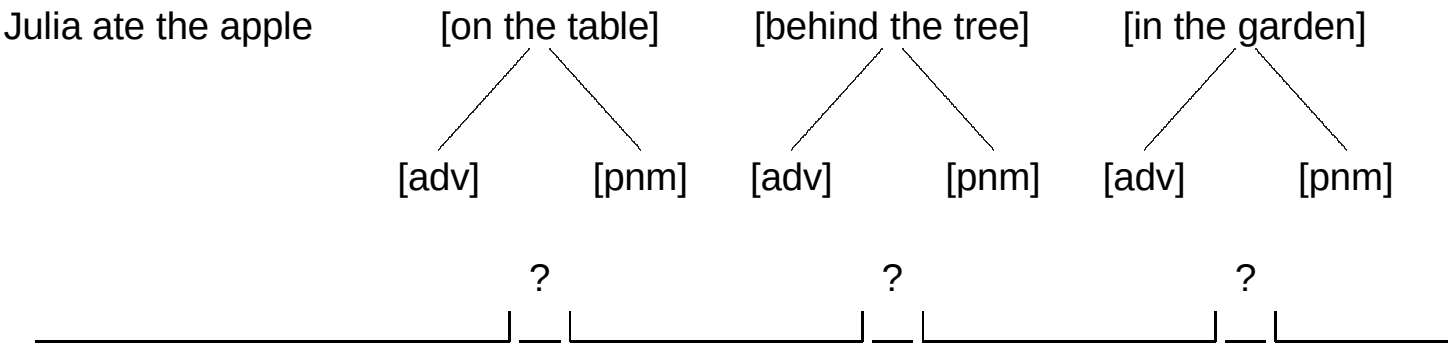
12.5.6 Correct analysis of a semantic ambiguity



12.5.7 Incorrect analysis: a recursive pseudo-ambiguity



12.5.8 Correct analysis with *semantic doubling*



12.5.9 CoNSyx hypothesis (Complexity of natural language syntax)

The natural languages are contained in the class of C1-languages and parse in linear time.

This hypothesis holds as long as no recursive ambiguities are found in natural language.