

---

# Advanced Natural Language Processing

## Lecture 10

### Grammar Formalisms (I)

Frank Keller (slides by Mark Steedman and Bonnie Webber)

12 October 2011



## Three Ways to Look at Syntactic Structures

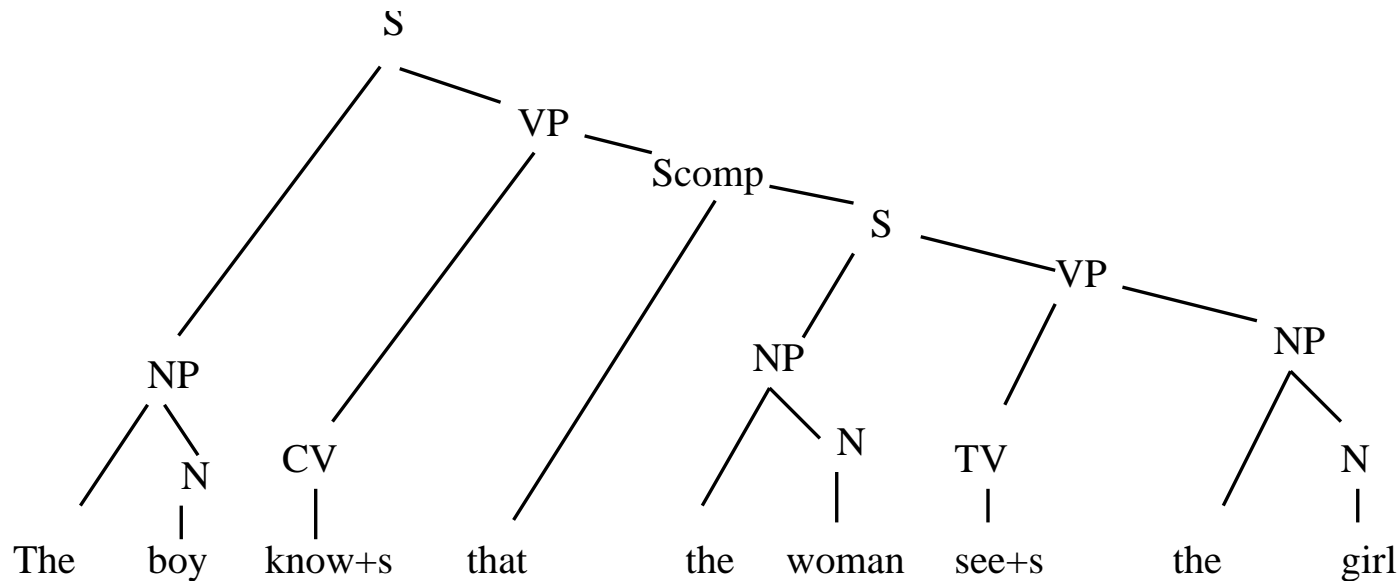
- Phrase Structure Grammar (PSG)
- Categorical Grammar (CG)
- Dependency Grammar

These differ in whether or not they have syntactic categories, and if so, what they are.

## Toy Context-free PSG for a Fragment of English

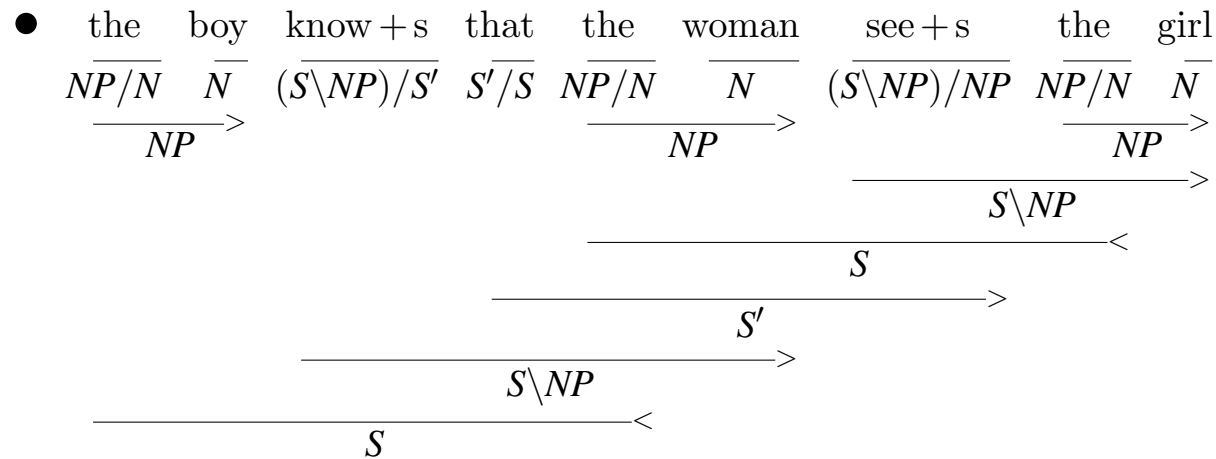
$$\begin{aligned}
 S &\rightarrow NP \quad VP \\
 NP &\rightarrow \left\{ \begin{array}{l} \text{the} \quad N \\ \{John, \quad Fred, \quad Sally, \\ \quad Mary, \dots\} \end{array} \right. \\
 VP &\rightarrow \left\{ \begin{array}{l} TV \quad NP \\ CV \quad Scomp \\ VP \quad PP \end{array} \right. \\
 Scomp &\rightarrow \text{that} \quad S \\
 N &\rightarrow \{man, woman, boy, girl, \dots\} \\
 TV &\rightarrow \{see, like, loath, \dots\} + s \\
 CV &\rightarrow \{see, know, dream, \dots\} + s
 \end{aligned}$$

## A Context-free PSG Derivation



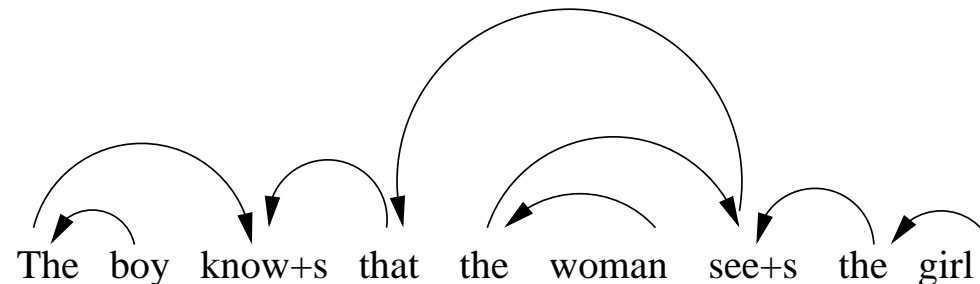
- The structure of the tree reflects the structure of the interpretation (logical form).

## A CG Derivation



- This is just a context-free derivation tree, except it is the right way up.
- It too corresponds to the structure of semantic interpretation.
- **The Grammar is entirely defined by the lexicon.**

## A Dependency Graph



- This dependency graph uses a convention where dependency corresponds to CG function application, and the arrow points from argument to functor.
- ◈ There are other types of dependency and other directionality conventions—in particular:
  - **Head-dependencies** (e.g. from determiners to nouns and nouns to verbs);
  - **Semantic Dependencies** (e.g. from subject of auxiliary to subject of main verb).

## Natural Grammars: Bounded Dependency

- Natural Language Grammar is concerned with two kinds of constructions involving **dependency** between elements of syntactic structure:
- Those like Binding and Control involve **bounded dependencies** between elements of the **same tensed domain**:
  - *Gilbert likes himself/showed himself a movie/\*thinks Hermione likes himself.*
  - *Gilbert persuaded George to take a bath/\*that Hermione advised to take a bath*
- Those like relative clauses involve **unbounded dependencies** between nouns and verbs **across tensed clause boundaries**:
  - *A man who Gilbert thinks he likes*

## Unbounded Dependency

### Topicalized sentences:

- You should *sell that bike* soon.
- *That bike* you should *sell* soon.
- *That bike* I think you should *sell* soon.
- *That bike* I heard they think you should *sell* soon

### Wh-Questions:

- *Which bike* should you *sell* soon?
- *Which bike* do they think you should *sell* soon?

## More Unbounded Dependency

### Object Relative Clauses:

- Fred *likes that man*.
- John likes the *man* [that Fred *likes*].
- John likes the *man* [that Sally thinks [that Fred *likes*]].
- John likes the *man* [that Sally thinks [that Fred sees [that Mary *likes*]]].

## Context Free Base-Generation?

Does it make sense to think that **unbounded dependency** is generated in a context-free PSG?

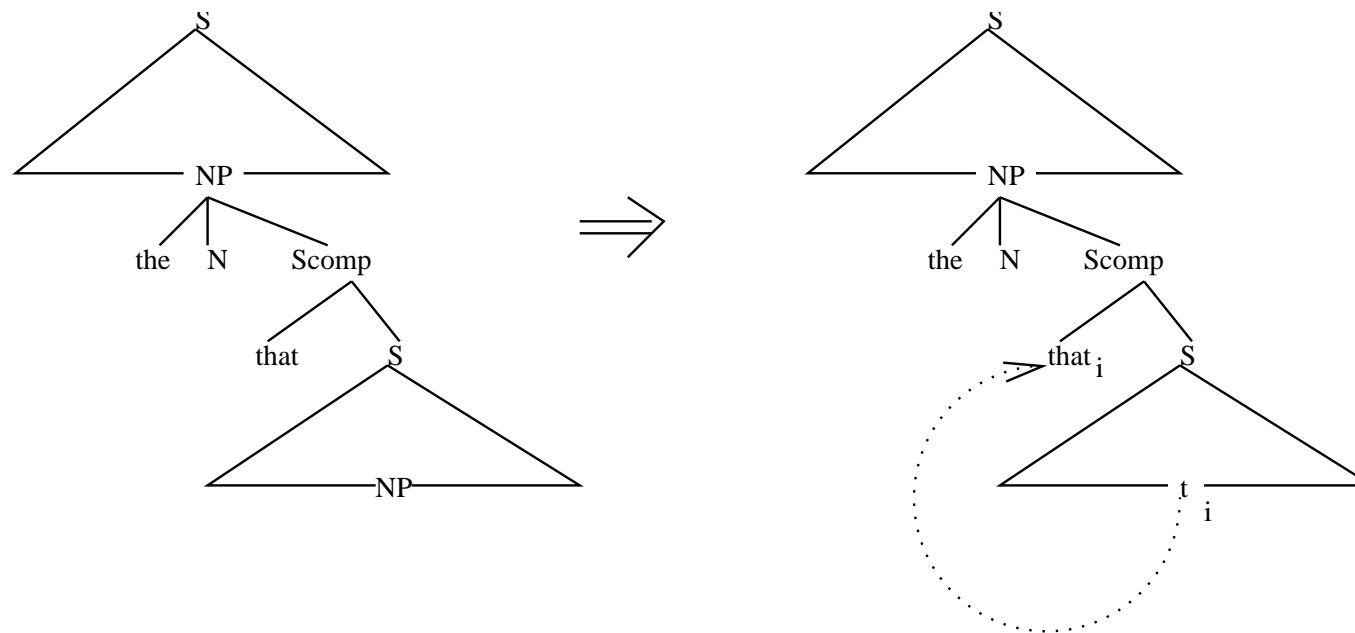
$$\begin{aligned} S &\rightarrow NP \quad VP \\ NP &\rightarrow \begin{cases} the \quad N \quad (Scomp) \\ t \\ \{John, Fred, Sally, Mary, \dots\} \end{cases} \\ VP &\rightarrow \begin{cases} TV \quad NP \\ CV \quad Scomp \end{cases} \\ Scomp &\rightarrow that \quad S \\ N &\rightarrow \{man, woman, boy, girl, \dots\} \\ TV &\rightarrow \{see, like, loath, \dots\} + s \\ CV &\rightarrow \{see, know, dream, \dots\} + s \end{aligned}$$

## Apparent Overgeneralization

- – Fred met the man that Sally saw on Tuesday.
  - \*Fred thinks that Sally saw on Tuesday.
- – \*Fred met the man that Sally saw John on Tuesday.
  - I think that Sally saw John on Tuesday.
- Answers to the problem of unbounded dependency:
  - Transformational Grammar (Chomsky 1957);
  - Generalized Phrase-structure Grammar (Gazdar 1981);
  - Linear Indexed Grammar (Gazdar, 1988).

# I: Generic Transformational Generative Grammar

(Chomsky 1957; 1965; 1975; 1981; 1995 *passim*)



- As expressive as a Turing Machine (Peters and Ritchie 1973)

## II: Generalized Phrase-Structure Grammar (GPSG)

### Base generation again:

*We might attempt to extend the notions of phrase structure to account for discontinuities.* [Chomsky 1957:41, note 6]

- Information about a “moved” or “missing” constituent can be passed up and down the tree by introducing
  - Additional category symbols (“slashed” categories) that must dominate a “gap” at some arbitrary depth beneath them
  - A rule introducing a “gap” that must be licensed by a slashed category

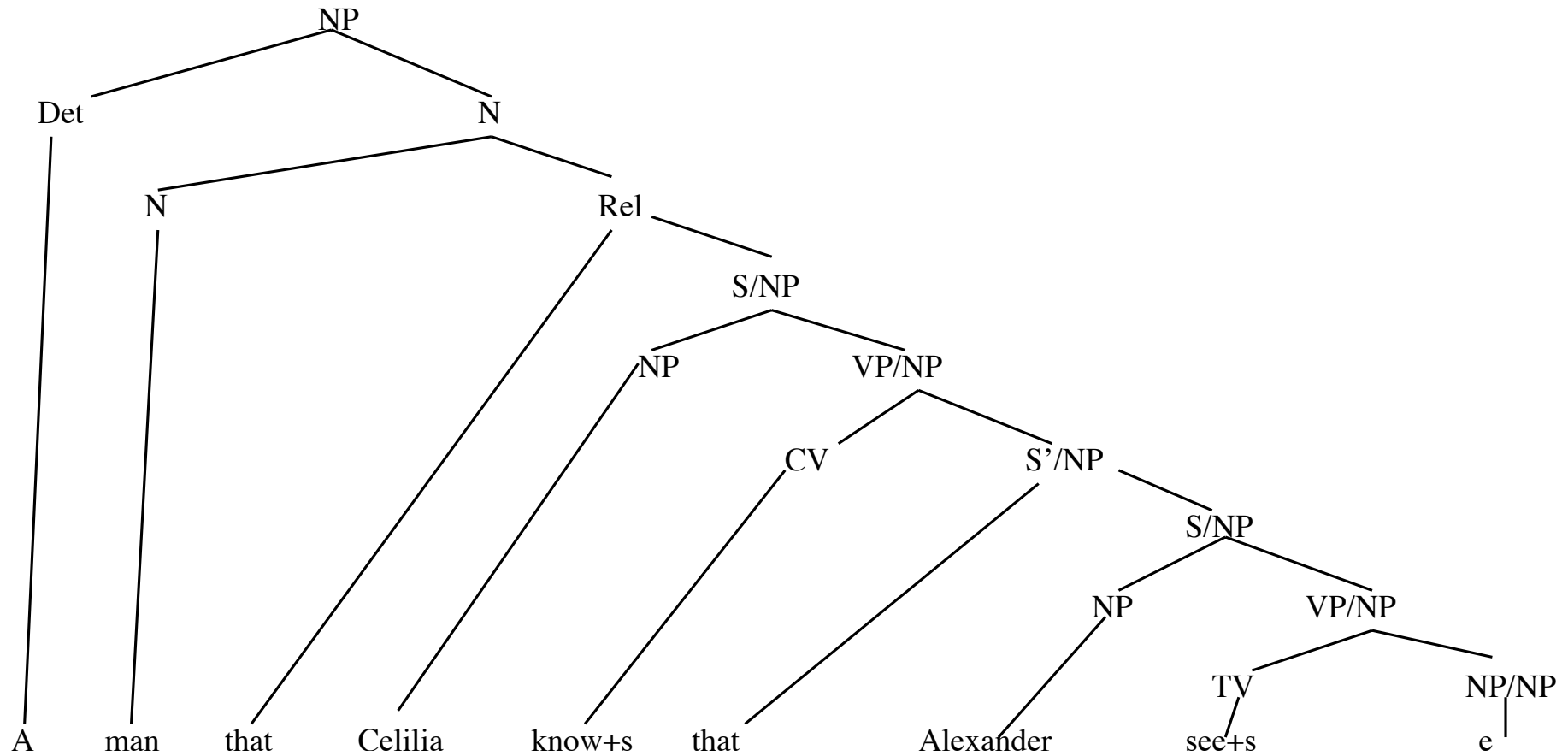
## The Context-free PSG Fragment of English Again

S	→	NP	VP	NP	→	Det	N
				NP	→	PN	
VP	→	TV	NP				
VP	→	DTV	NP	PP	N	→	N Rel
					S'	→	that S
VP	→	CV	S'		Rel	→	that VP

## The GPSG Version

S	→	NP VP	NP	→	Det N
S/NP	→	NP VP/NP	NP	→	PN
VP	→	TV NP	NP/NP	→	$\epsilon$
VP/NP	→	TV NP/NP	N	→	N Rel
VP	→	DTV NP PP	S'	→	that S
VP/NP	→	DTV NP/NP PP	S'/NP	→	that S/NP
VP/NP	→	DTV NP PP/NP	Rel	→	that VP
VP	→	CV S'	Rel	→	that S/NP
VP/NP	→	CV S'/NP			

# A GPSG Derivation



## Coordinate Structure Constraint

- General assumption: only “like” categories can be coordinated...
  - (1) a. [<sub>NP</sub> NP and NP]  
b. [<sub>VP</sub> VP and VP]  
c. \* [<sub>?P</sub> NP and VP]
- ... but a “slashed” category is different from an unslashed one
  - (2) \* [<sub>?P</sub> VP/NP and VP]
- So the “Coordinate Structure Constraint” follows directly
  - (3) \*a man that<sub>i</sub> we [<sub>?P</sub> [<sub>VP/NP</sub> met  $t_i$  yesterday] and [<sub>VP</sub> saw him today]]

## The Across-The-Board Exception to CSC

- Two unslashed categories can be identical:

(4) I [<sub>VP</sub> [<sub>VP</sub> met him yesterday] and [<sub>VP</sub> saw him again today]].

- A slashed category cannot be identical to an unslashed one:

(5) \*a man that I [<sub>?P</sub> [<sub>VP/NP</sub> met *t* yesterday] and [<sub>VP</sub> saw him again today]].

- But two slashed categories can also be identical so ATB is allowed:

(6) a man that I [<sub>VP/NP</sub> [<sub>VP/NP</sub> met *t* yesterday] and [<sub>VP/NP</sub> saw *t* again today]].

## More than Context-Free Power

- Recall the Dutch examples for crossing  $a^n b^n$  structures:
  - a. dat ik<sub>1</sub> Henk<sub>2</sub> zag<sub>1</sub> zwemmen<sub>2</sub>.  
that I Henk saw swim  
“that I saw Henry swim”
  - b. dat ik<sub>1</sub> Jan<sub>2</sub> Henk<sub>3</sub> zag<sub>1</sub> zien<sub>2</sub> zwemmen<sub>3</sub>.  
that I Jan Henk saw see swim  
“that I saw John see Henry swim”
  - c. dat ik<sub>1</sub> Anna<sub>2</sub> Jan<sub>3</sub> Henk<sub>4</sub> zag<sub>1</sub> helpen<sub>2</sub> zien<sub>3</sub> zwemmen<sub>4</sub>  
that I Anna Jan Henk saw help see swim  
“that I saw Anna help John see Henry swim”
- This cannot be captured by a context-free grammar, not even if we assume slash passing. We need a **stack**.

## III: Linear Indexed Grammars (LIG)

- Linear Indexed Grammar (LIG, [Gazdar, 1988]) is a way of integrating the slash feature with a PDA stack.
- LIG is a natural family of languages in the Chomsky hierarchy, in the sense that it is associated with a characteristic automaton, the Embedded PDA (EPDA [Joshi, 1991]).
- LIG falls just above the context-free class in the hierarchy.
- It is therefore “nearly context free”.

## A LIG for $a^n b^n$ Crossed

- LIG can capture crossing dependencies.

$$(7) A_{[\dots]} \rightarrow \alpha B_{[\dots]} \beta$$

$$(8) \text{“pushing”}: A_{[\dots]} \rightarrow \alpha B_{[i,\dots]} \beta$$

$$\text{“popping”}: A_{[i,\dots]} \rightarrow \alpha B_{[\dots]} \beta$$

$$(9) S_{[\dots]} \rightarrow n_i S_{[i,\dots]}$$

$$S_{[\dots]} \rightarrow S'_{[\dots]}$$

$$S'_{[i,\dots]} \rightarrow S'_{[\dots]} v_i$$

$$S'_{[]} \rightarrow \epsilon$$

## Example: A LIG Derivation for $n^3v^3$ Crossed

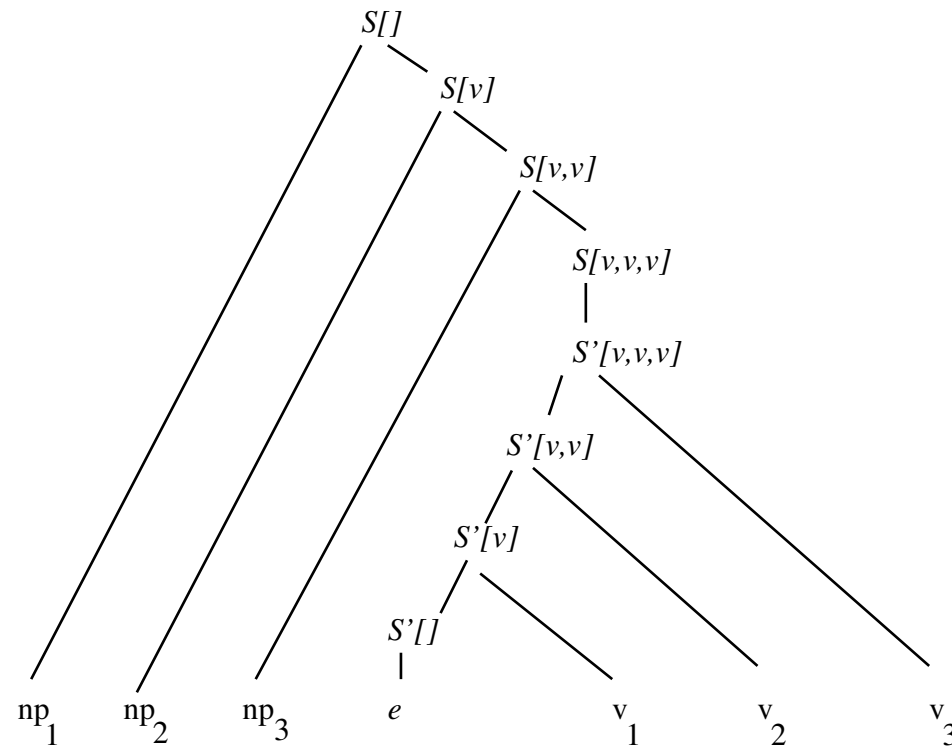


Figure 1: (Weakly adequate) LIG derivation for  $n^3v^3$

## An Answer to Chomsky's Implicit Question?

- LIG offers a convincing candidate for a upper bound on the complexity of Natural Grammars.
- However, the above LIG is only weakly adequate.
- In the next lecture, we will see **Tree Adjoining Grammar (TAG)**, which licenses strongly adequate structures for crossing dependencies.
- TAG has the same complexity as LIG.

## References

Chomsky, Noam. 1957. *Syntactic Structures*. The Hague: Mouton.

Gazdar, Gerald. 1981. Unbounded dependencies and coordinate structure. *Linguistic Inquiry* 12: 155–184.

Peters, Stanley, and Robert Ritchie. 1973. On the generative power of transformational grammars. *Information Science* 6: 49–83.