Components of the Grammar

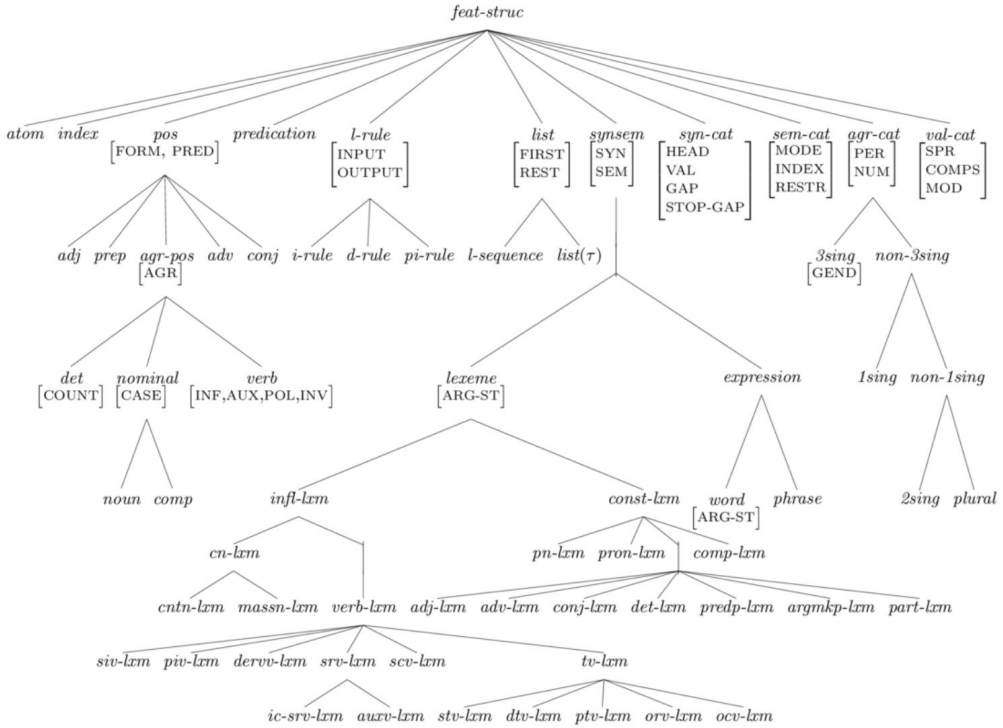
LING566

Components of the Grammar

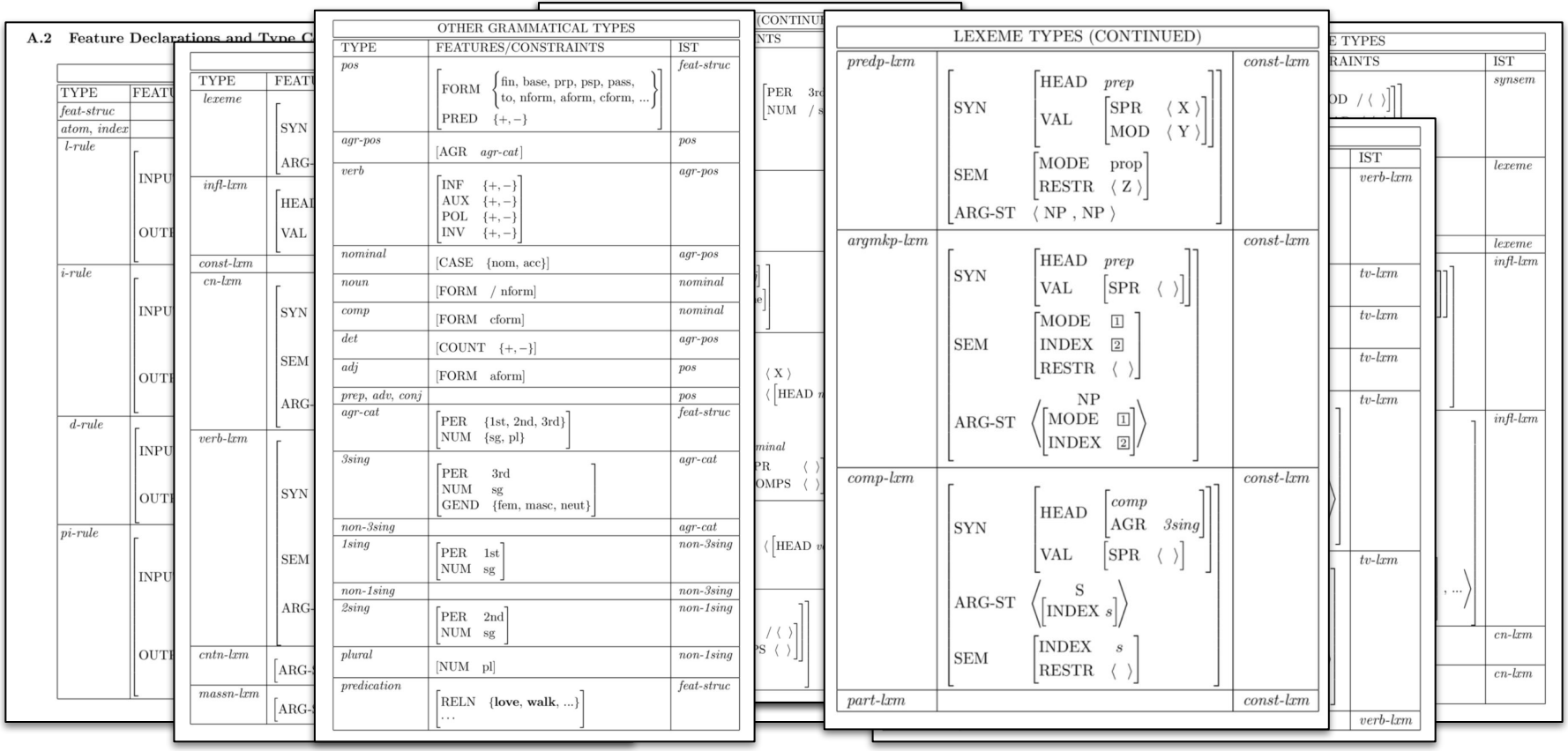
- Type hierarchy
  - this includes lexeme types
- Lexical entries
- Lexical rules
- Grammar rules
- Principles
- Initial symbol

Combinations of constraints from these components determine what trees are licensed by our grammar

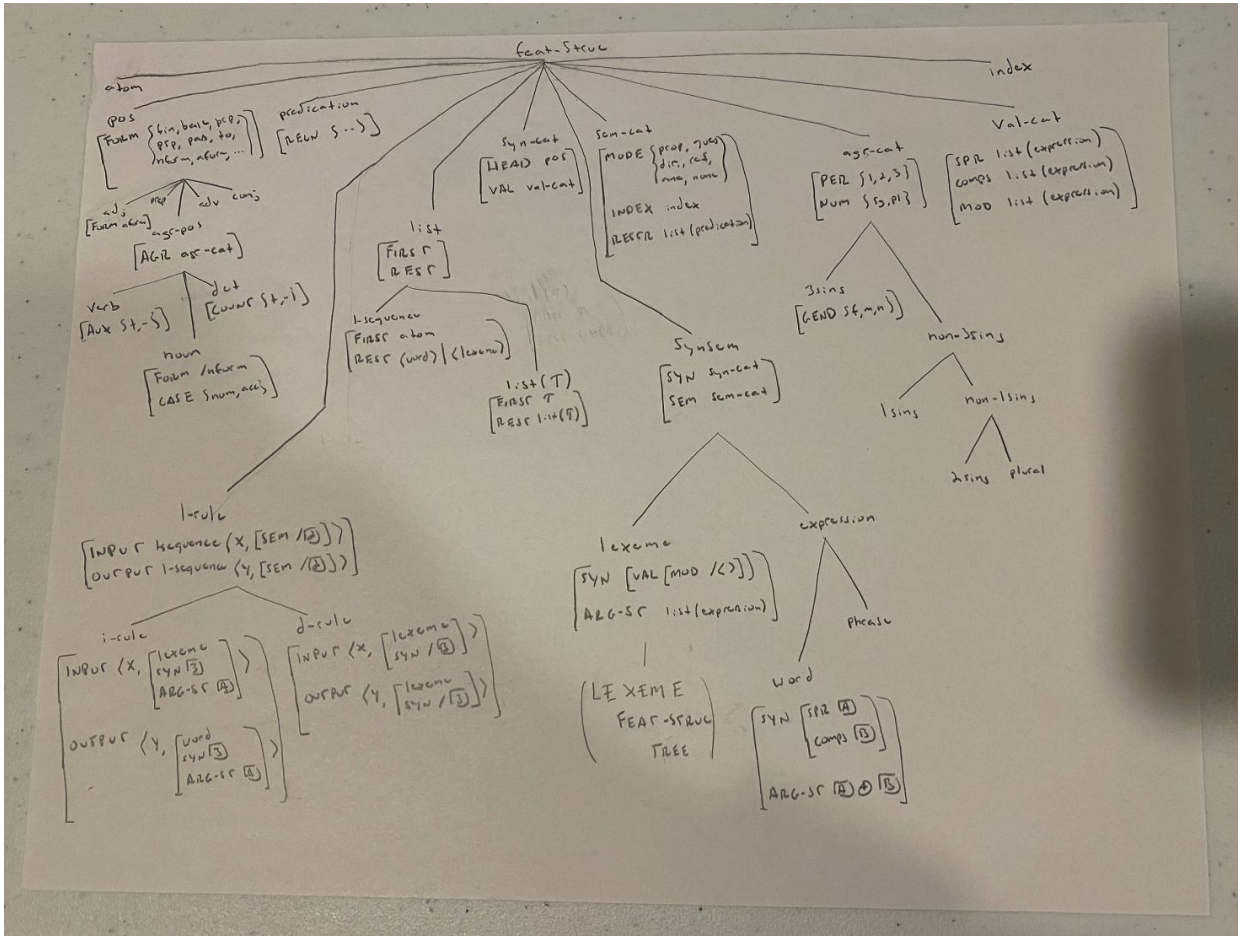
Type Hierarchy



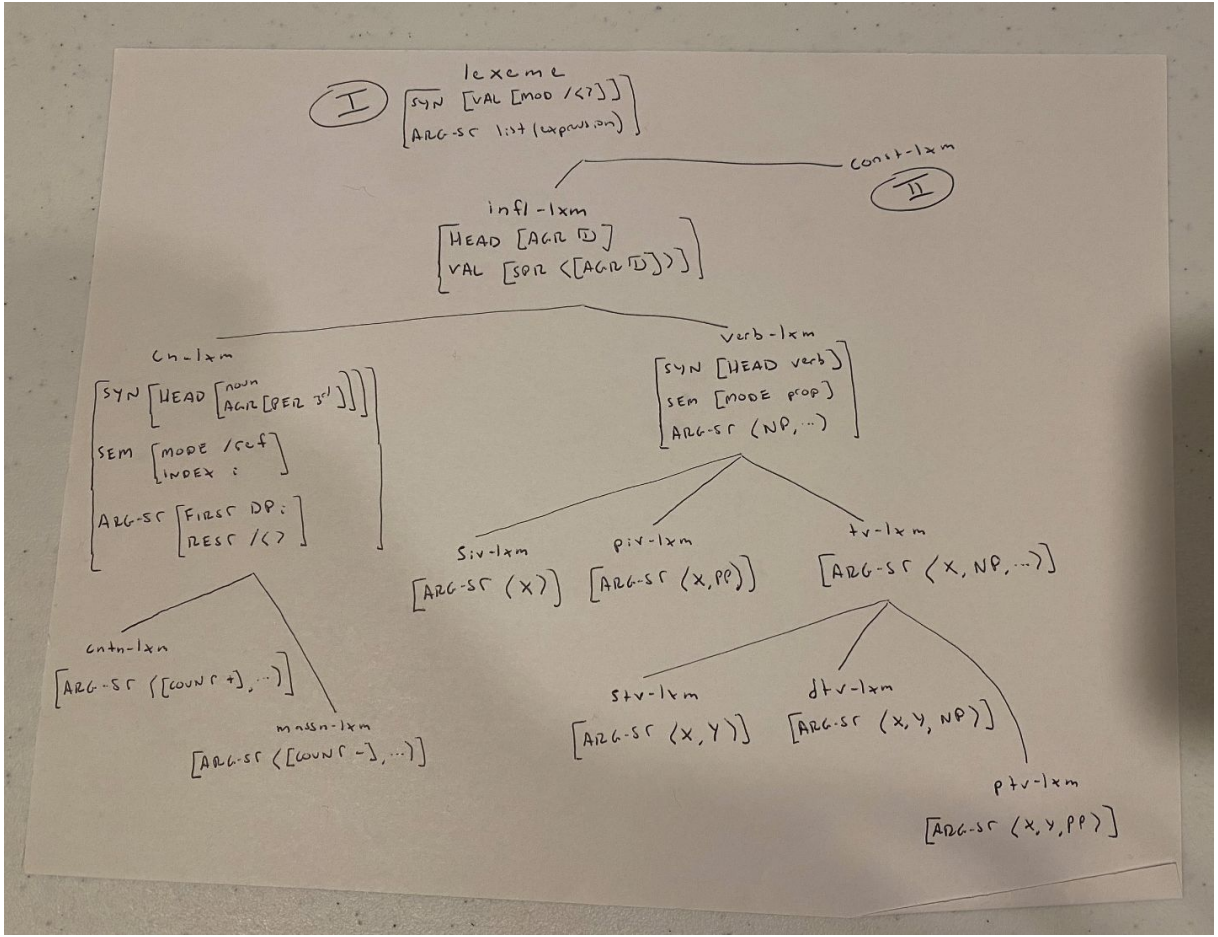
Type Hierarchy



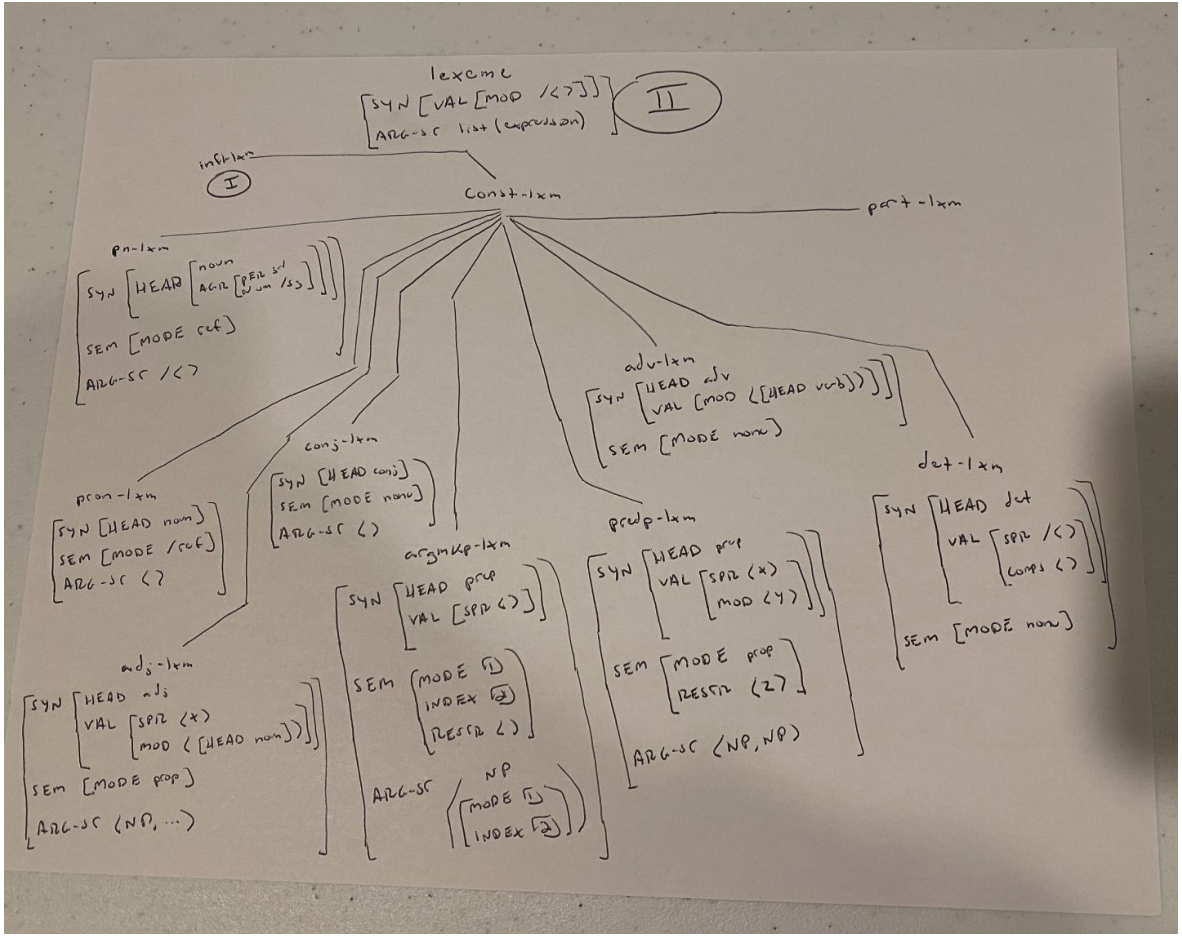
Type Hierarchy by the end of the book (1/3)



Type Hierarchy by the end of the book (2/3)



Type Hierarchy by the end of the book (3/3)



Type Hierarchy

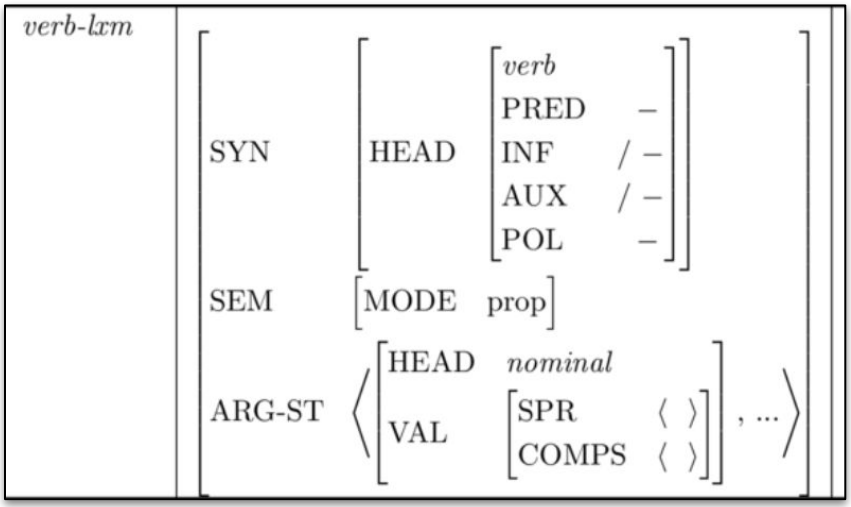
- Lists all of the types and their subtypes
- Specifies what features are appropriate for each type
- Specifies what values are appropriate for each feature

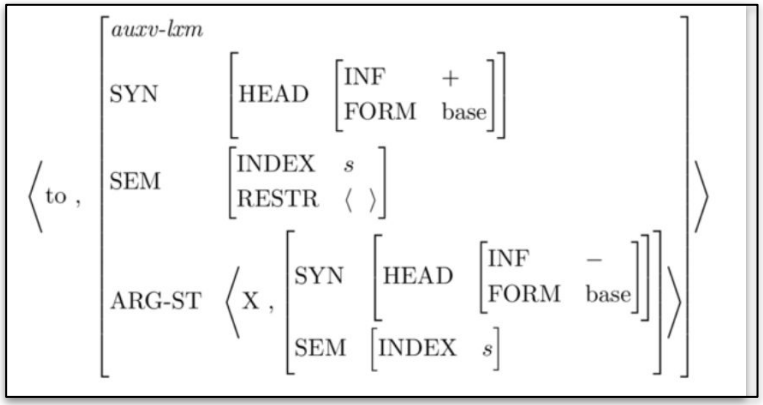
syn-cat	$\begin{bmatrix} \text{HEAD} & \textit{pos} \\ \text{VAL} & \textit{val-cat} \\ \text{GAP} & \textit{list}(\textit{expression}) \\ \text{STOP-GAP} & \textit{list}(\textit{expression}) \end{bmatrix}$	*feat-struc*

pos	$\begin{bmatrix} \text{FORM} & \left\{ \begin{array}{l} \text{fin, base, prp, psp, pass,} \\ \text{to, nform, aform, cform, ...} \end{array} \right\} \\ \text{PRED} & \{+, -\} \end{bmatrix}$	*feat-struc*

Type Hierarchy

- **NOTE: Defeasibility can only be declared in the type hierarchy**
- Features with defeasible values can be overwritten in subtypes, in particular lexical entries, or particular lexical rules





Lexical Terminology

Lexical Terminology

- Lexeme types
- Lexical entries
- Lexical rules

Lexical Terminology

- Lexeme types
- Lexical entries
- Lexical rules

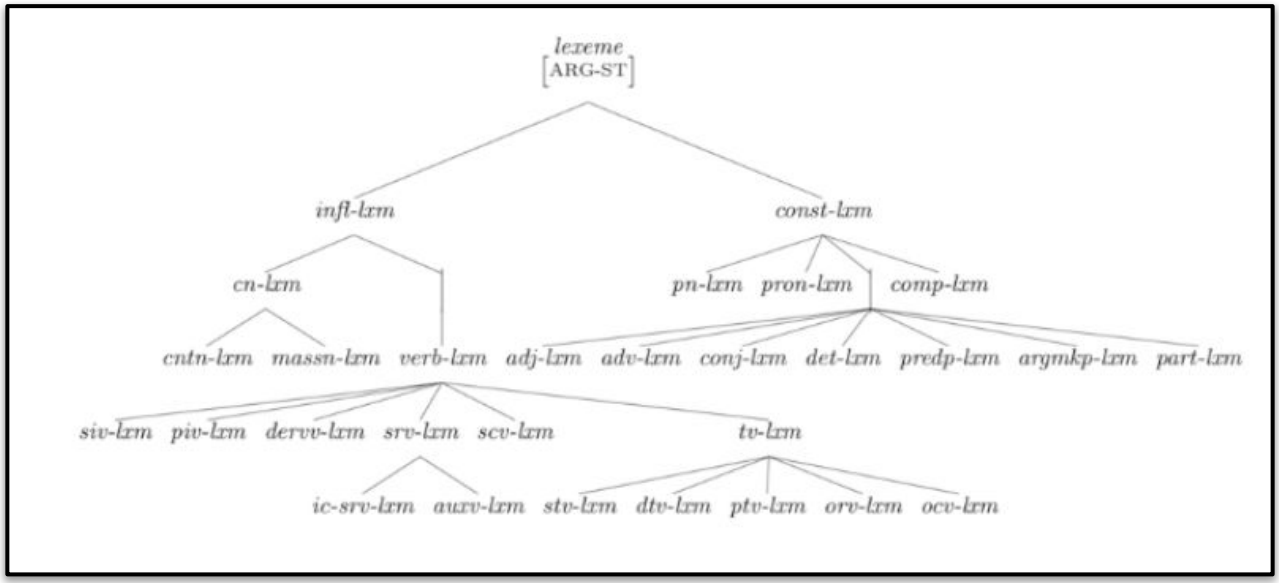
These are elements we posit when writing the grammar

- Lexical sequence
- Family of lexical sequences
- Lexical rule instantiation
- Word structure

These come about as a *consequence* of the way our grammar works

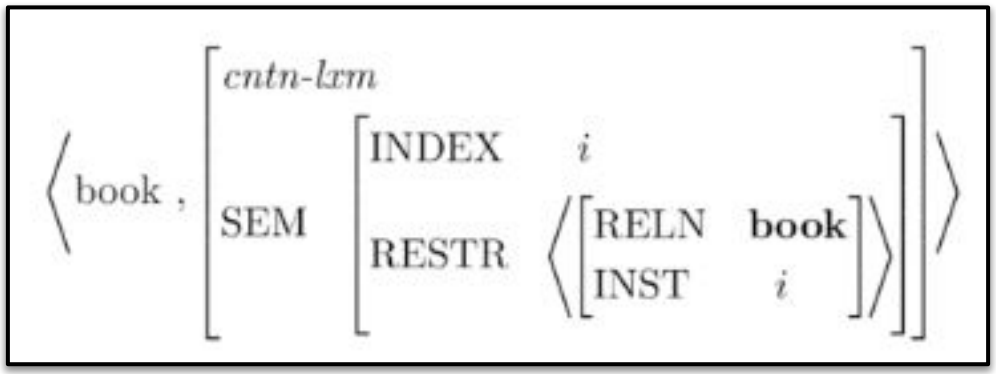
Lexeme

- *lexeme* is a type in our grammar, and it serves as the root of the “subhierarchy” of all lexical types in the type hierarchy



Lexical Entries

- Lexical entries make up the lexicon of the grammar, they are underspecified descriptions of ‘words’
- Each lexical entry is an ordered pair consisting of an orthographic/phonological form and a partial feature structure description



Lexical entry for *book*

cntn-lxm	$\left[\text{ARG-ST} \quad \langle\, [\text{COUNT } +] \,, \ldots \,\rangle\right]$	*cn-lxm*
cn-lxm	$\left[\begin{array}{ll}\text{SYN} & \left[\text{HEAD} \left[\begin{array}{ll}noun & \\ \text{AGR} & [\text{PER 3rd}]\end{array}\right]\right] \\ \text{SEM} & \left[\begin{array}{ll}\text{MODE} & / \text{ ref} \\ \text{INDEX} & i\end{array}\right] \\ \text{ARG-ST} & \left[\begin{array}{ll}\text{FIRST} & \text{DP}_i \\ \text{REST} & / \langle\, \rangle\end{array}\right]\end{array}\right]$	*infl-lxm*
infl-lxm	$\left[\text{HEAD} \left[\text{AGR } \boxed{1}\right] \quad \text{VAL} \left[\text{SPR} \langle \left[\text{AGR } \boxed{1}\right] \rangle\right]\right]$	*lexeme*
lexeme	$\left[\text{SYN} \left[\begin{array}{ll}\text{VAL} & [\text{MOD } / \langle\, \rangle] \\ \text{STOP-GAP} & / \langle\, \rangle\end{array}\right] \quad \text{ARG-ST } \textit{list}(\textit{expression})\right]$	*synsem*

$$\left\langle \text{book} \,, \left[\begin{array}{ll} cntn\text{-}lxm & \\ \text{SEM} & \left[\begin{array}{ll}\text{INDEX} & i \\ \text{RESTR} & \left\langle \left[\begin{array}{ll}\text{RELN} & \textbf{book} \\ \text{INST} & i\end{array}\right] \right\rangle\end{array}\right]\end{array}\right] \right\rangle$$

Lexical entry for *book*

cntn-lxm	$\left[\text{ARG-ST} \quad \langle\, [\text{COUNT } +] \,, \ldots \,\rangle\right]$	*cn-lxm*
cn-lxm	$\left[\begin{array}{ll}\text{SYN} & \left[\text{HEAD} \left[\begin{array}{ll}\textit{noun} & \\ \text{AGR} & \left[\text{PER } 3\text{rd}\right]\end{array}\right]\right] \\ \text{SEM} & \left[\begin{array}{ll}\text{MODE} & / \text{ ref} \\ \text{INDEX} & i\end{array}\right] \\ \text{ARG-ST} & \left[\begin{array}{ll}\text{FIRST} & \text{DP}_i \\ \text{REST} & / \langle\ \rangle\end{array}\right]\end{array}\right]$	*infl-lxm*
infl-lxm	$\left[\begin{array}{ll}\text{HEAD} & \left[\text{AGR } \boxed{1}\right] \\ \text{VAL} & \left[\text{SPR } \langle \left[\text{AGR } \boxed{1}\right] \rangle\right]\end{array}\right]$	*lexeme*
lexeme	$\left[\begin{array}{ll}\text{SYN} & \left[\begin{array}{ll}\text{VAL} & \left[\text{MOD } / \langle\ \rangle\right] \\ & \text{STOP-GAP } / \langle\ \rangle\end{array}\right] \\ \text{ARG-ST} & \textit{list(expression)}\end{array}\right]$	*synsem*

$$\left\langle \text{book}, \left[\begin{array}{ll}\textit{cntn-lxm} & \\ \text{SYN} & \left[\begin{array}{ll}\text{HEAD} & \left[\begin{array}{ll}\textit{noun} & \\ \text{AGR} & \boxed{1}\left[\text{PER } 3\text{rd}\right]\end{array}\right] \\ \text{VAL} & \left[\begin{array}{ll}\text{SPR} & \left\langle \left[\text{AGR } \boxed{1}\right] \right\rangle \\ \text{MOD} & \langle\rangle\end{array}\right]\end{array}\right] \\ \text{SEM} & \left[\begin{array}{ll}\text{INDEX} & i \\ \text{MODE} & \text{ref} \\ \text{RESTR} & \left\langle \left[\begin{array}{ll}\text{RELN} & \textbf{book} \\ \text{INST} & i\end{array}\right] \right\rangle\end{array}\right] \\ \text{ARG-ST} & \left\langle \left[\text{SYN } \left[\text{HEAD } \left[\text{COUNT } +\right]\right]\right]_{\text{DP}_i} \right\rangle\end{array}\right] \right\rangle$$

Lexical entry for *book*

cntn-lxm	$\left[\text{ARG-ST} \quad \langle\, [\text{COUNT } +] \,,\, \ldots \,\rangle\right]$	*cn-lxm*
cn-lxm	$\left[\begin{array}{ll}\text{SYN} & \left[\text{HEAD} \left[\begin{array}{ll}\textit{noun} & \\ \text{AGR} & \left[\text{PER } 3\text{rd}\right]\end{array}\right]\right] \\ \text{SEM} & \left[\begin{array}{ll}\text{MODE} & / \text{ ref} \\ \text{INDEX} & i\end{array}\right] \\ \text{ARG-ST} & \left[\begin{array}{ll}\text{FIRST} & \text{DP}_i \\ \text{REST} & / \langle\ \rangle\end{array}\right]\end{array}\right]$	*infl-lxm*
infl-lxm	$\left[\begin{array}{ll}\text{HEAD} & \left[\text{AGR } \boxed{1}\right] \\ \text{VAL} & \left[\text{SPR } \langle \left[\text{AGR } \boxed{1}\right] \rangle\right]\end{array}\right]$	*lexeme*
lexeme	$\left[\begin{array}{ll}\text{SYN} & \left[\begin{array}{ll}\text{VAL} & \left[\text{MOD } / \langle\ \rangle\right] \\ & \text{STOP-GAP } / \langle\ \rangle\end{array}\right] \\ \text{ARG-ST} & \textit{list(expression)}\end{array}\right]$	*synsem*

$$\left\langle \text{book}, \left[\begin{array}{ll}\textit{cntn-lxm} & \\ \text{SYN} & \left[\begin{array}{ll}\textit{syn-cat} & \\ \text{HEAD} & \left[\begin{array}{ll}\textit{noun} & \\ \text{AGR} & \boxed{1}\left[\begin{array}{ll}\textit{agr-cat} & \\ \text{PER} & 3\text{rd}\end{array}\right]\end{array}\right] \\ \text{VAL} & \left[\begin{array}{ll}\textit{val-cat} & \\ \text{SPR} & \left\langle \left[\text{AGR } \boxed{1}\right] \right\rangle \\ \text{MOD} & \langle\rangle\end{array}\right]\end{array}\right] \\ \text{SEM} & \left[\begin{array}{ll}\textit{sem-cat} & \\ \text{INDEX} & i \\ \text{MODE} & \text{ref} \\ \text{RESTR} & \left\langle \left[\begin{array}{ll}\text{RELN} & \textbf{book} \\ \text{INST} & i\end{array}\right] \right\rangle\end{array}\right] \\ \text{ARG-ST} & \left\langle \left[\text{SYN } \left[\text{HEAD } \left[\text{COUNT } +\right]\right]\right]_{\text{DP}_i} \right\rangle\end{array}\right] \right\rangle$$

Lexical entry for *book*

syn-cat		*feat-struc*
	HEAD *pos* VAL *val-cat* GAP *list(expression)* STOP-GAP *list(expression)*	

noun	[FORM / nform]	*nominal*
nominal	[CASE {nom, acc}]	*agr-pos*
agr-pos	[AGR *agr-cat*]	*pos*
pos	FORM {fin, base, prp, psp, pass, to, nform, aform, cform, ...} PRED {+, −}	*feat-struc*

val-cat		*feat-struc*
	SPR *list(expression)* COMPS *list(expression)* MOD *list(expression)*	

sem-cat		*feat-struc*
	MODE {prop, ques, dir, ref, ana, none} INDEX {*index*, none} RESTR *list(predication)*	

⟨ book, [*cntn-lxm*
SYN [*syn-cat*
HEAD [*noun*
AGR 1 [*agr-cat*
PER 3rd]]
VAL [*val-cat*
SPR ⟨ [AGR 1] ⟩
MOD ⟨⟩]]
SEM [*sem-cat*
INDEX *i*
MODE ref
RESTR ⟨ [RELN **book**
INST *i*] ⟩]
ARG-ST ⟨ [SYN [HEAD [COUNT +]]] ⟩] ⟩
DP$_i$

Lexical entry for *book*

syn-cat		*feat-struc*
	$\begin{bmatrix} \text{HEAD} & \textit{pos} \\ \text{VAL} & \textit{val-cat} \\ \text{GAP} & \textit{list(expression)} \\ \text{STOP-GAP} & \textit{list(expression)} \end{bmatrix}$	

noun	[FORM / nform]	*nominal*
nominal	[CASE {nom, acc}]	*agr-pos*
agr-pos	[AGR *agr-cat*]	*pos*
pos	$\begin{bmatrix} \text{FORM} & \{\text{fin, base, prp, psp, pass, to, nform, aform, cform, ...}\} \\ \text{PRED} & \{+,-\} \end{bmatrix}$	*feat-struc*

val-cat		*feat-struc*
	$\begin{bmatrix} \text{SPR} & \textit{list(expression)} \\ \text{COMPS} & \textit{list(expression)} \\ \text{MOD} & \textit{list(expression)} \end{bmatrix}$	

sem-cat		*feat-struc*
	$\begin{bmatrix} \text{MODE} & \{\text{prop, ques, dir, ref, ana, none}\} \\ \text{INDEX} & \{\textit{index}, \text{none}\} \\ \text{RESTR} & \textit{list(predication)} \end{bmatrix}$	

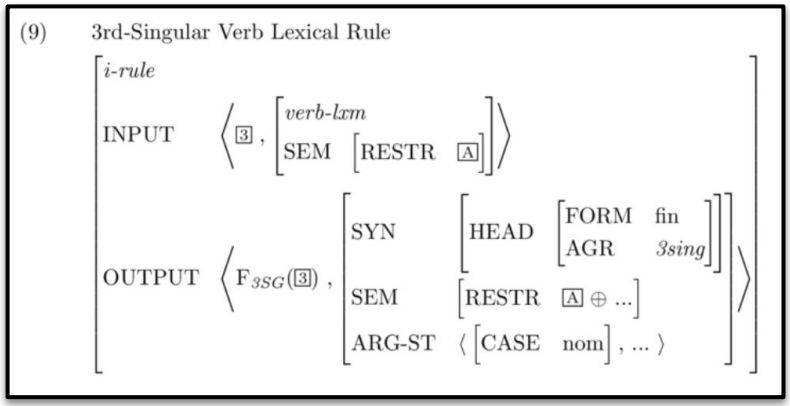
$$\left\langle \text{book}, \begin{bmatrix} \textit{cntn-lxm} \\ \text{SYN} & \begin{bmatrix} \textit{syn-cat} \\ \text{HEAD} & \begin{bmatrix} \textit{noun} \\ \text{FORM} & \text{nform} \\ \text{AGR} & \boxed{1}\begin{bmatrix} \textit{agr-cat} \\ \text{PER} & \text{3rd} \end{bmatrix} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \textit{val-cat} \\ \text{SPR} & \left\langle \begin{bmatrix} \text{AGR} & \boxed{1} \end{bmatrix} \right\rangle \\ \text{MOD} & \langle\rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \textit{sem-cat} \\ \text{INDEX} & i \\ \text{MODE} & \text{ref} \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{book} \\ \text{INST} & i \end{bmatrix} \right\rangle \end{bmatrix} \\ \text{ARG-ST} & \left\langle \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{COUNT} & + \end{bmatrix} \end{bmatrix} \end{bmatrix}_{\text{DP}_i} \right\rangle \end{bmatrix} \right\rangle$$

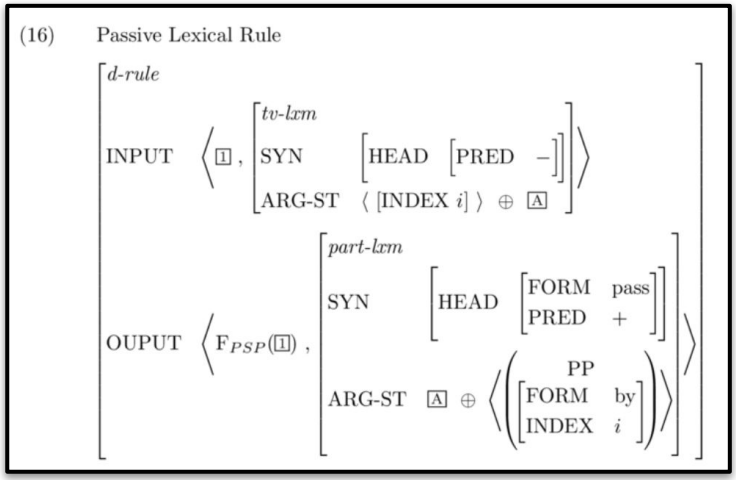
Lexical Terminology

- Lexeme types ✅
- Lexical entries ✅
- Lexical rules
- Lexical sequence
- Family of lexical sequences
- Lexical rule instantiation
- Word structure

Lexical Rules

- Another part of the grammar we write
- Allow us to derive inflected forms (via lexical rules of type *i-rule* or *pi-rule*)
- OR additional lexeme type lexical sequences that can be input to more lexical rules (via lexical rules of type *d-rule*)





Lexical Terminology

- Lexeme types ✅
- Lexical entries ✅
- Lexical rules ✅
- Lexical sequence
- Family of lexical sequences
- Lexical rule instantiation
- Word structure

Lexical Sequence

- Ordered pairs that can serve as the INPUT or OUTPUT values of lexical rules
- Any lexical sequence is a pair that consists of an orthographic/phonological form and a fully resolved feature structure

Family of Lexical Sequences

- If you have an ordered pair that consists of an orthographic/phonological form and an **underspecified** feature structure, we say that it “describes a **family of lexical sequences**”

What lexical sequences does this describe?

syn-cat		*feat-struc*
	[HEAD ? *pos* VAL ? *val-cat* ~~GAP *list(expression)*~~ ~~STOP-GAP *list(expression)*~~]	

noun	[FORM ✅ / nform]	*nominal*
nominal	[CASE ? {nom, acc}]	*agr-pos*
agr-pos	[AGR ? *agr-cat*]	*pos*

agr-cat		*feat-struc*
	[PER ✅ {1st, 2nd, 3rd} NUM ? {sg, pl}]	

pos		*feat-struc*
	[FORM ✅ {fin, base, prp, psp, pass, to, nform, aform, cform, ...} PRED ? {+, −}]	

val-cat		*feat-struc*
	[SPR ✅ *list(expression)* COMPS ? *list(expression)* MOD ✅ *list(expression)*]	

sem-cat		*feat-struc*
	[MODE ✅ {prop, ques, dir, ref, ana, none} INDEX ✅ {*index*, none} RESTR ✅ *list(predication)*]	

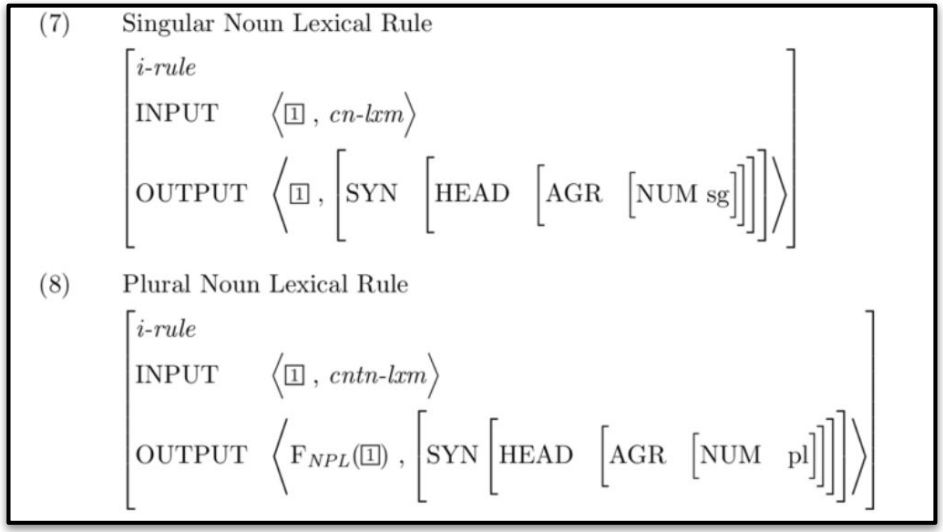
$$\left\langle \text{book}, \begin{bmatrix} \textit{cntn-lxm} \\ \text{SYN} \begin{bmatrix} \textit{syn-cat} \\ \text{HEAD} \begin{bmatrix} \textit{noun} \\ \text{FORM} \quad \text{nform} \\ \text{AGR} \quad \boxed{1} \begin{bmatrix} \textit{agr-cat} \\ \text{PER} \quad 3\text{rd} \end{bmatrix} \end{bmatrix} \\ \text{VAL} \begin{bmatrix} \textit{val-cat} \\ \text{SPR} \quad \left\langle \begin{bmatrix} \text{AGR} \quad \boxed{1} \end{bmatrix} \right\rangle \\ \text{MOD} \quad \langle\rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} \begin{bmatrix} \textit{sem-cat} \\ \text{INDEX} \quad i \\ \text{MODE} \quad \text{ref} \\ \text{RESTR} \quad \left\langle \begin{bmatrix} \text{RELN} & \textbf{book} \\ \text{INST} & i \end{bmatrix} \right\rangle \end{bmatrix} \\ \text{ARG-ST} \quad \left\langle \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{COUNT} & + \end{bmatrix} \end{bmatrix} \end{bmatrix}_{\text{DP}_i} \right\rangle \end{bmatrix} \right\rangle$$

Lexical Terminology

- Lexeme types ✅
- Lexical entries ✅
- Lexical rules ✅
- Lexical sequence ✅
- Family of lexical sequences ✅
- Lexical rule instantiation ✅
- Word structure

Filtering through the “soup” (😕)

- A certain number of the lexical sequences described by that partial description can be “caught” by these lexical rules
- The INPUT and OUTPUT will then each be associated with a particular **fully resolved** lexical sequence
- Infinitely many other lexical sequences will not unify with these constraints and therefore be ignored, left to float in the “soup”



Lexical Rule Instantiation

- A lexical rule instantiation is a **fully resolved** feature structure that is consistent with the specifications of some lexical rule
- We don’t show these in our trees, but when you work through a problem to build a tree, you will step through some number of lexical rule instantiations in order to get the word structure that will sit at each leaf in the tree
- Some implemented HPSG grammars actually represent the lexical rules on the trees
    - ERG demo

Word Structure

- Found at the bottom of the tree, serve as the leaves of the tree
- A feature structure of type *word* over some string
- “Correspond to lexical sequences that correspond to OUTPUT of some lexical rule instantiation that is a fully resolved version of some lexical rule where INPUT is a lexical sequence that is licensed by a lexical entry (or output of some other lexical rule)”

Lexical Terminology

- Lexeme types ✅
- Lexical entries ✅
- Lexical rules ✅
- Lexical sequence ✅
- Family of lexical sequences ✅
- Lexical rule instantiation ✅
- Word structure ✅

The rest of the grammar lol

Components of the Grammar

- Type hierarchy ✅
  - this includes lexeme types
- Lexical entries ✅
- Lexical rules ✅
- Grammar rules
- Principles
- Initial symbol

Combinations of constraints from these components determine what trees are licensed by our grammar

Grammar Rules

- Also called phrase structure rules
- Build feature structures of type (or really subtypes) of *expression* from other feature structures of type *expression*
- This is how we get our tree structure

A.4 The Grammar Rules

(All daughters in our grammar rules are expressions, i.e. of type *word* or *phrase*; never of type *lexeme*).

(1) Head-Specifier Rule

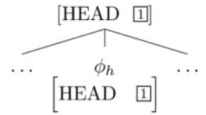
$$\begin{bmatrix} \textit{phrase} \\ \text{SPR} \quad \langle\ \rangle \end{bmatrix} \rightarrow \boxed{1} \quad \mathbf{H} \begin{bmatrix} \text{VAL} \begin{bmatrix} \text{SPR} \quad \langle\ \boxed{1}\ \rangle \\ \text{COMPS} \quad \langle\ \rangle \end{bmatrix} \\ \text{STOP-GAP} \quad \langle\ \rangle \end{bmatrix}$$

A phrase can consist of a (lexical or phrasal) head preceded by its specifier.

Principles

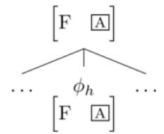
- What principles do we have so far?
  - Head Feature Principle
  - Valence Principle
  - Semantic Inheritance Principle
  - Semantic Compositionality Principle
- Are outlined in the “squiggly bits”

(72) Φ satisfies the Head Feature Principle with respect to a headed rule ρ if and only if Φ satisfies:



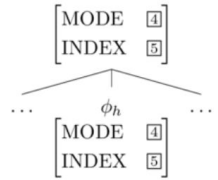
where ϕ_h is the head daughter of Φ.

(73) Φ satisfies the Valence Principle with respect to a headed rule ρ if and only if, for any VAL feature F, Φ satisfies:



where ϕ_h is the head daughter of Φ and ρ does not specify incompatible F values for ϕ_h and ϕ_0.

(74) Φ satisfies the Semantic Inheritance Principle with respect to a headed rule ρ if and only if Φ satisfies:



where ϕ_h is the head daughter of Φ.

Initial Symbol

A.7.6 Structures Defined by the Grammar

(65) **Well-Formed Tree Structure:**

Φ is a Well-Formed Tree Structure according to G if and only if:

1. Φ is a tree structure,
2. the label of Φ's root node satisfies the constraint:

$$\begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{verb} \\ \text{FORM} \quad \text{fin} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} \quad \langle\ \rangle \\ \text{SPR} \quad \langle\ \rangle \end{bmatrix} \\ \text{GAP} & \langle\ \rangle \end{bmatrix} \end{bmatrix}$$

and

3. each local subtree within Φ is either phrasally licensed or lexically licensed.

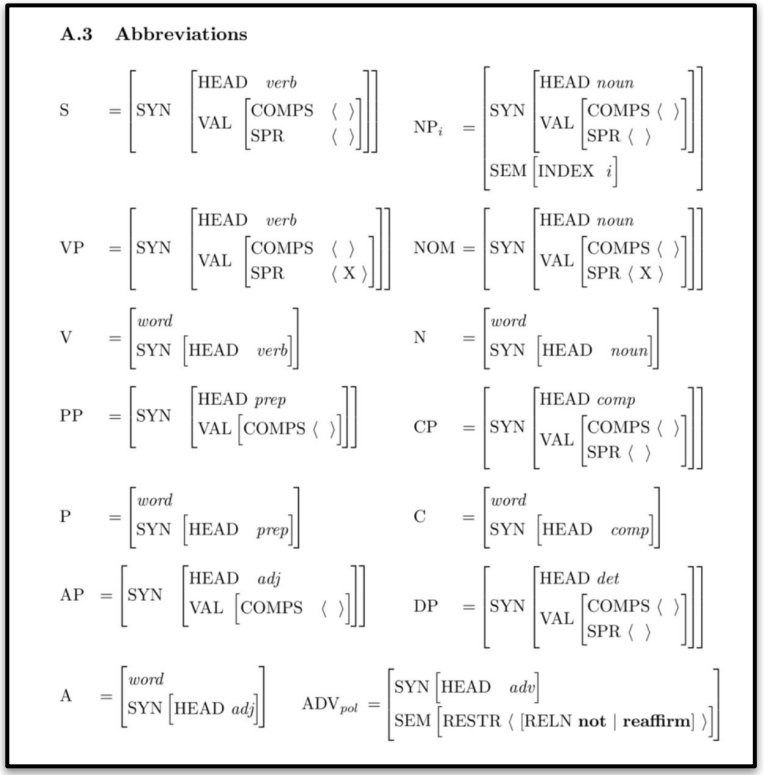
Components of the Grammar

- Type hierarchy ✅
  - this includes lexeme types
- Lexical entries ✅
- Lexical rules ✅
- Grammar rules ✅
- Principles ✅
- Initial symbol ✅

Combinations of constraints from these components determine what trees are licensed by our grammar

Abbreviations??

- Abbreviations are partial descriptions of feature structures
- They serve as a notational convenience because it's easier to write "V" than it is to write the details of what the abbreviation V entails



A.3 Abbreviations

$$\text{S} = \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{verb} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\ \rangle \\ \text{SPR} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix} \qquad \text{NP}_i = \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{noun} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\ \rangle \\ \text{SPR} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{INDEX} & i \end{bmatrix} \end{bmatrix}$$

$$\text{VP} = \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{verb} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\ \rangle \\ \text{SPR} & \langle\ \text{X}\ \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix} \qquad \text{NOM} = \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{noun} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\ \rangle \\ \text{SPR} & \langle\ \text{X}\ \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$\text{V} = \begin{bmatrix} \textit{word} \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{verb} \end{bmatrix} \end{bmatrix} \qquad \text{N} = \begin{bmatrix} \textit{word} \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{noun} \end{bmatrix} \end{bmatrix}$$

$$\text{PP} = \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{prep} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix} \qquad \text{CP} = \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{comp} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\ \rangle \\ \text{SPR} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$\text{P} = \begin{bmatrix} \textit{word} \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{prep} \end{bmatrix} \end{bmatrix} \qquad \text{C} = \begin{bmatrix} \textit{word} \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{comp} \end{bmatrix} \end{bmatrix}$$

$$\text{AP} = \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{adj} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix} \qquad \text{DP} = \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{det} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\ \rangle \\ \text{SPR} & \langle\ \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$\text{A} = \begin{bmatrix} \textit{word} \\ \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{adj} \end{bmatrix} \end{bmatrix} \qquad \text{ADV}_{pol} = \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{adv} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{RESTR} & \langle\ [\text{RELN}\ \textbf{not} \mid \textbf{reaffirm}]\ \rangle \end{bmatrix} \end{bmatrix}$$

yay!