# HG4041 Theories of Grammar

# Semantics

Francis Bond
**Division of Linguistics and Multilingual Studies**
http://www3.ntu.edu.sg/home/fcbond/
bond@ieee.org

Lecture 4
Location: HSS SR3

# Schedule

| Lec. | Topic | | Reading | Problems |
|---|---|---|---|---|
| 1 | Introduction (HPSG) | | SWB 1–2 | 1:1 |
| 2 | Feature Structures | | SWB 3 | 3:1, 3 |
| 3 | Complex Feature Values | | SWB 4 | 4:1, 5, 6 |
| 4 | Semantics | | SWB 5–6 | 5:1; 6:1, 3, 4, 5 |
| 5 | Binding | | SWB 7 | 7:1, 2 |
| 6 | The Structure of the Lexicon | **Mid-term** | SWB 8 | 8:1, 2, 6 |
| 7 | Realistic Grammar | | SWB 9 | 9:1 |
| 8 | Passive | | SWB 10 | 10: 1, 3 |
| 9 | Dummies and Idioms | | SWB 11 | 11:1, 3, 4 |
| 10 | Raising and Control | | SWB 12 | 12:1, 2, 4, 6 |
| 11 | Long Distance Dependencies | **Final** | SWB 14 | 14: 1, 2, 3 |
| 12 | Wrap-up | **Project Presentations** | SWB 16 | |
| | **Research Paper** | | | |
| | due two weeks after presentations | | | |

# Overview

- Last week: HC, HS, VP, SHAC
- Some notes on the linguist’s stance
- Which aspects of semantics well tackle
- Our formalization; Semantics Principles
- Building semantics of phrases
- Modification, coordination
- Structural ambiguity

# Overview

- Complex Feature Structures allow us to
  - Write more general rules
  - Constrain them lexically

# Head-Complement Rule

$$\begin{bmatrix} \textit{phrase} & \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\rangle \end{bmatrix} \end{bmatrix} \rightarrow \boldsymbol{H} \begin{bmatrix} \textit{word} & \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \left\langle \boxed{1}, \ldots, \boxed{n} \right\rangle \end{bmatrix} \end{bmatrix} \boxed{1}, \ldots, \boxed{n}$$

- The possible complements are specified lexically

$$\left\langle \text{devour}, \begin{bmatrix} \textit{word} & \\ \text{HEAD} & \textit{verb} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle \text{NP} \rangle \end{bmatrix} \end{bmatrix} \right\rangle \quad \left\langle \text{put}, \begin{bmatrix} \textit{word} & \\ \text{HEAD} & \textit{verb} \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle \text{NP PP} \rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

# Head-Specifier Rule

$$\begin{bmatrix} \textit{phrase} & \\ \textit{VAL} & \begin{bmatrix} \textit{COMPS} & \langle\rangle \\ \textit{SPR} & \langle\rangle \end{bmatrix} \end{bmatrix} \rightarrow \boxed{2} \quad \textbf{\textit{H}} \begin{bmatrix} \textit{VAL} & \begin{bmatrix} \textit{COMPS} & \langle\rangle \\ \textit{SPR} & \langle \boxed{2} \rangle \end{bmatrix} \end{bmatrix}$$

- Combines the rules expanding S and NP (and other, . . . ).
- Again, restrictions on specifiers come from the lexicon.

$$\left\langle \text{dog}, \begin{bmatrix} \textit{word} & \\ \text{HEAD} & \textit{noun} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \text{D} \rangle \\ \text{COMPS} & \langle\rangle \end{bmatrix} \end{bmatrix} \right\rangle \quad \left\langle \text{eat}, \begin{bmatrix} \textit{word} & \\ \text{HEAD} & \textit{verb} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \text{NP} \rangle \\ \text{COMPS} & \langle (\text{NP}) \rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

# Two Principles

- **The Valence Principle**

  Unless the rule says otherwise, the mothers values for the VAL features (SPR and COMPS) are identical to those of the head daughter.

- **The Specifier-Head Agreement Constraint (SHAC)**
  Verbs and nouns must be specified as:

$$\begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{AGR } \boxed{1} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR } \left\langle \text{AGR } \boxed{1} \right\rangle \end{bmatrix} \end{bmatrix}$$

  Actually inherited from a lexical super-type

# Semantics

# Overview

- Some notes on the linguist's stance
- Which aspects of semantics well tackle
- Our formalization; Semantics Principles
- Building semantics of phrases
- Modification, coordination
- Structural ambiguity

# The Linguist's Stance: Building a precise model

- Some of our statements are statements about how the model works:

  [*prep*] and [AGR *3sing*] cant be combined because AGR is not a feature of the type prep.

- Some of our statements are statements about how (we think) English or language in general works.

  The determiners *a* and *many* only occur with count nouns, the determiner *much* only occurs with mass nouns, and the determiner *the* occurs with either.

- Some are statements about how we code a particular linguistic fact within the model.

  All count nouns are [SPR $<$ [COUNT +]$>$].

# The Linguist's Stance:

A Vista on the Set of Possible English Sentences

- . . . as a background against which linguistic elements (words, phrases) have a distribution
- . . . as an arena in which linguistic elements behave in certain ways

# Semantics: Where's the Beef?

So far, our grammar has no semantic representations. We have, however, been relying on semantic intuitions in our argumentation, and discussing semantic contrasts where they line up (or don't) with syntactic ones.

- structural ambiguity
- S/NP parallelism
- count/mass distinction
- complements vs. modifiers

# Our Slice of a World of Meanings

Aspects of meaning we won't account for (in this course)

- Pragmatics
- Fine-grained lexical semantics
  The meaning of *life* is
  - `life` (or life′)
  - $\begin{bmatrix} \text{RELN} & \textit{life} \\ \text{INST} & i \end{bmatrix}$
  - Not like wordnet: $\texttt{life}_1 \subset \texttt{being}_1 \subset \texttt{state}_1 \ldots$

# Our Slice of a World of Meanings

$$
\begin{bmatrix}
\text{MODE} & \textit{prop} \\
\text{INDEX} & s \\
\text{RESTR} & \left\langle
\begin{bmatrix}
\text{RELN} & \textbf{save} \\
\text{SIT} & s \\
\text{SAVER} & i \\
\text{SAVED} & j
\end{bmatrix},
\begin{bmatrix}
\text{RELN} & \textbf{name} \\
\text{NAME} & \textit{Chris} \\
\text{NAMED} & i
\end{bmatrix},
\begin{bmatrix}
\text{RELN} & \textbf{name} \\
\text{NAME} & \textit{Pat} \\
\text{NAMED} & j
\end{bmatrix}
\right\rangle
\end{bmatrix}
$$

. . . the linguistic meaning of *Chris saved Pat* is a proposition that will be true just in case there is an actual situation that involves the saving of someone named Pat by someone named Chris.

(Sag *et al*, 2003, p. 140)

# Our Slice of a World of Meanings

What we are accounting for is the **compositionality** of sentence meaning.

- How the pieces fit together

  **Semantic arguments** and **indices**

- How the meanings of the parts add up to the meaning of the whole.

  Appending RESTR lists up the tree

The value of **RESTRiction** is the set of conditions that must hold (in some possible world) for the expression to be applicable.

# Semantics in Constraint-Based Grammar

- Constraints as (generalized) truth conditions
  - **proposition**: what must be the case for a proposition to be true
  - **directive**: what must happen for a directive to be fulfilled
  - **question**: the kind of situation the asker is asking about
  - **reference**: the kind of entity the speaker is referring to
- **Syntax/semantics interface**:

  Constraints on how syntactic arguments are related to semantic ones, and on how semantic information is compiled from different parts of the sentence.

# Feature Geometry

$$
\begin{bmatrix}
\textit{expression} & \\
\text{SYN} & \begin{bmatrix}
\textit{syn-cat} & \\
\text{HEAD} & \begin{bmatrix} \textit{pos} \\ \cdots \end{bmatrix} \\
\text{VAL} & \begin{bmatrix} \text{SPR} & \langle \ldots \rangle \\ \text{COMPS} & \langle \ldots \rangle \end{bmatrix}
\end{bmatrix} \\
\text{SEM} & \begin{bmatrix}
\textit{sem-cat} & \\
\text{MODE} & \{\, \textit{prop, ques, dir, ref, none} \,\} \\
\text{INDEX} & \{\, i, j, k, \ldots, s_1, s_2, \ldots \} \\
\text{RESTR} & \langle \ldots \rangle
\end{bmatrix}
\end{bmatrix}
$$

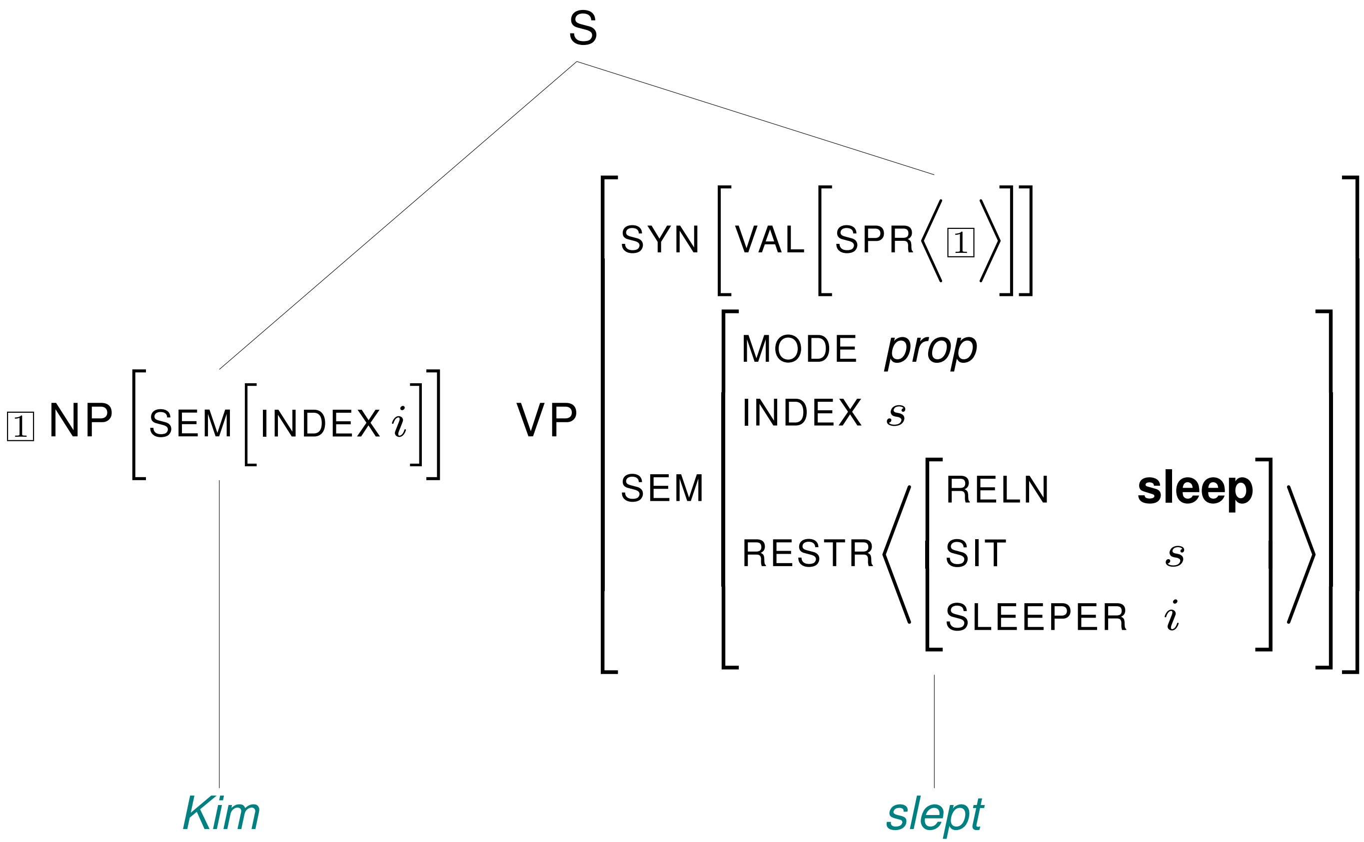
# How the Pieces Fit Together

$$\left\langle \textit{Kim}, \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \textit{noun} & \\ \text{AGR} & \textit{3sing} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\rangle \\ \text{COMPS} & \langle\rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \textit{ref} \\ \text{INDEX} & i \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{name} \\ \text{NAME} & \textit{Kim} \\ \text{NAMED} & i \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

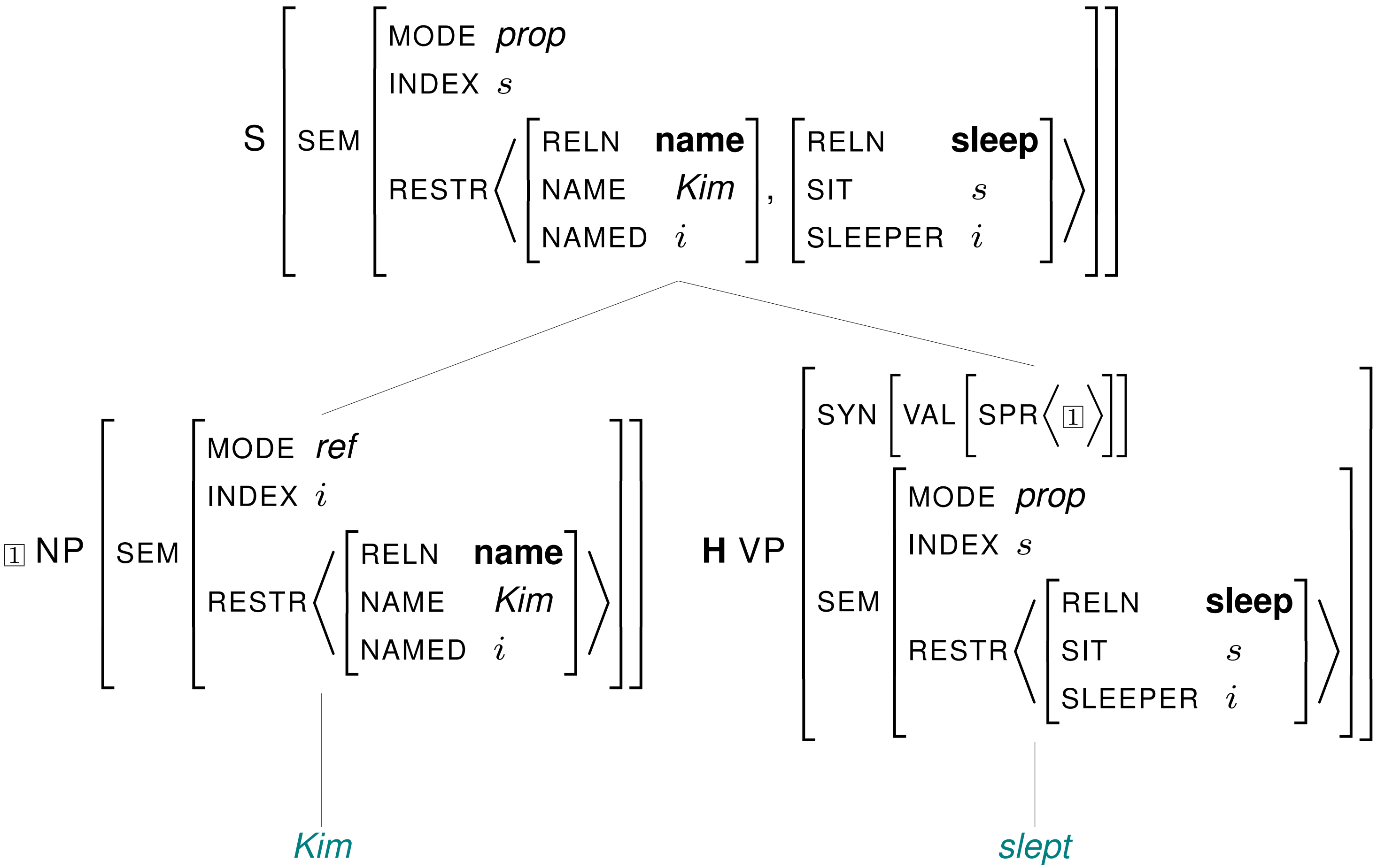
# How the Pieces Fit Together

$$\left\langle \textit{sleep}\ ,\ \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{verb} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\, NP_i \,\rangle \\ \text{COMPS} & \langle\,\rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \textit{prop} \\ \text{INDEX} & s \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{sleep} \\ \text{SIT} & s \\ \text{SLEEPER} & i \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

# The Pieces Together



# Another View of the Same Tree



# How to Share Semantic Information

We need the Semantics Principles

- **The Semantic Inheritance Principle**

  In any headed phrase, the mother's MODE and INDEX are identical to those of the head daughter.

- **The Semantic Compositionality Principle**

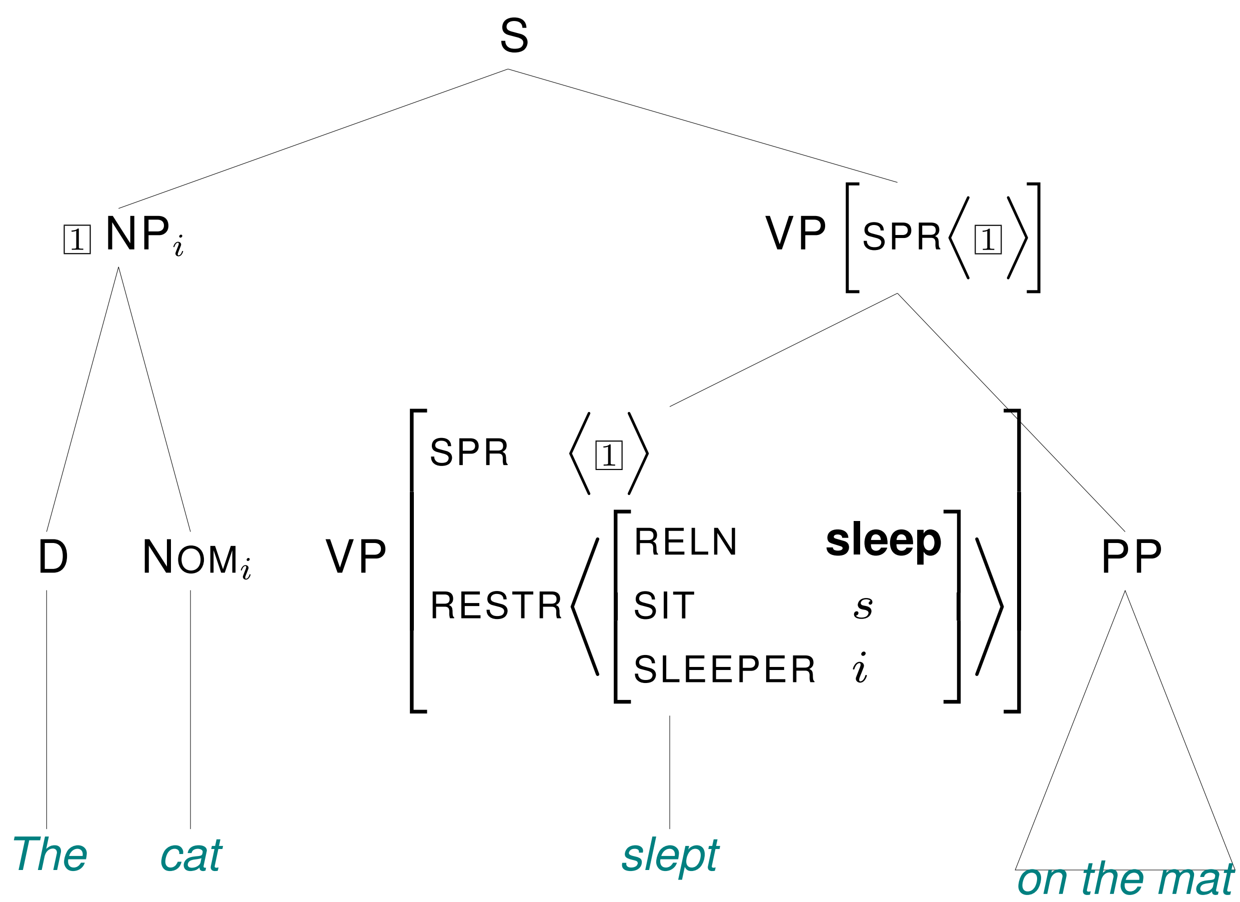
  In any well-formed phrase structure, the mother's RESTR value is the sum of the RESTR values of the daughter.

List summation: $\oplus$ (technically concatenation) $\langle A \rangle \oplus \langle B \rangle \neq \langle B \rangle \oplus \langle A \rangle$

$$\langle a_1, a_2, \ldots, a_n \rangle \oplus \langle b_1, b_2, \ldots, b_m \rangle = \langle a_1, a_2, \ldots, a_n, b_1, b_2, \ldots, b_m \rangle$$

$$\langle b_1, b_2, \ldots, b_m \rangle \oplus \langle a_1, a_2, \ldots, a_n \rangle = \langle b_1, b_2, \ldots, b_m, a_1, a_2, \ldots, a_n \rangle$$

# What Identifies Indices?



# Summary: Words . . .

- contribute predications
- expose one index in those predications, for use by words or phrases
- relate syntactic arguments to semantic arguments

$$\left\langle \textit{sleep}\ , \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{verb} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\, NP_i \,\rangle \\ \text{COMPS} & \langle\,\rangle \end{bmatrix} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{MODE} & \textit{prop} \\ \text{INDEX} & s \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{sleep} \\ \text{SIT} & s \\ \text{SLEEPER} & i \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

# Summary: Grammar Rules . . .

- Identify feature structures (including the INDEX value) across daughters
  - Head Specifier Rule

$$\begin{bmatrix} \textit{phrase} & \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\rangle \end{bmatrix} \end{bmatrix} \rightarrow \boxed{1} \quad \textbf{\textit{H}} \begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{SPR} & \langle \boxed{1} \rangle \\ \text{COMPS} & \langle\rangle \end{bmatrix} \end{bmatrix}$$

  - Head Complement Rule

$$\begin{bmatrix} \textit{phrase} & \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\rangle \end{bmatrix} \end{bmatrix} \rightarrow \textbf{\textit{H}} \begin{bmatrix} \textit{word} & \\ \text{VAL} & \begin{bmatrix} \text{COMPS} & \left\langle \boxed{1}, \ldots, \boxed{n} \right\rangle \end{bmatrix} \end{bmatrix} \boxed{1}, \ldots, \boxed{n}$$

  - Head Modifier Rule

$$\begin{bmatrix} \textit{phrase} \end{bmatrix} \rightarrow \textbf{\textit{H}}\,\boxed{1} \begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\rangle \end{bmatrix} \end{bmatrix} \begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\rangle \\ \text{MOD} & \left\langle \boxed{1} \right\rangle \end{bmatrix} \end{bmatrix}$$

# Summary: Grammar Rules . . .

- Identify feature structures (including the INDEX value) across daughters
- License trees which are subject to the semantic principles
  - SIP: passes up MODE and INDEX from head daughter
  - SCP: gathers up predications (RESTR list) from all daughters
- The semantics is strictly compositional — all of the meaning comes from the words, rules and principles.
  - We then enrich this with pragmatic inference — but we need a base to infer from

# Other Aspects of Semantics

- Tense, Quantification (only touched on here)
- Modification
- Coordination
- Structural Ambiguity

# Evolution of a Phrase Structure Rule

C2 NOM $\rightarrow$ NOM PP; VP $\rightarrow$ VP PP

C3 $$\begin{bmatrix} \textit{phrase} & \\ \text{VAL} & \begin{bmatrix} \text{SPR} & - \\ \text{COMPS} & \textit{itr} \end{bmatrix} \end{bmatrix} \rightarrow \boldsymbol{H}\begin{bmatrix} \textit{phrase} & \\ \text{VAL} & \begin{bmatrix} \text{SPR} & - \end{bmatrix} \end{bmatrix} \; \textit{PP}$$

C4 $$\begin{bmatrix} \textit{phrase} \end{bmatrix} \rightarrow \boldsymbol{H}\begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\rangle \end{bmatrix} \end{bmatrix} \; \textit{PP}$$

C5 $$\begin{bmatrix} \textit{phrase} \end{bmatrix} \rightarrow \boldsymbol{H}\boxed{1}\begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\rangle \end{bmatrix} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{COMPS} & \langle\rangle \\ \text{MOD} & \left\langle \boxed{1} \right\rangle \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

$$= \begin{bmatrix} \textit{phrase} \end{bmatrix} \rightarrow \boldsymbol{H}\boxed{1}\begin{bmatrix} \text{COMPS} & \langle\rangle \end{bmatrix} \begin{bmatrix} \text{COMPS} & \langle\rangle \\ \text{MOD} & \left\langle \boxed{1} \right\rangle \end{bmatrix}$$

# Evolution of Another Phrase Structure Rule

C2 $X \rightarrow X+$ CONJ $X$;

C3
$$\boxed{1} \rightarrow \boxed{1}+ \begin{bmatrix} \textit{word} & \\ \text{HEAD} & \textit{conj} \end{bmatrix} \boxed{1}$$

C4
$$\begin{bmatrix} \text{VAL} & \boxed{1} \end{bmatrix} \rightarrow \begin{bmatrix} \text{VAL} & \boxed{1} \end{bmatrix}+ \begin{bmatrix} \textit{word} & \\ \text{HEAD} & \textit{conj} \end{bmatrix} \begin{bmatrix} \text{VAL} & \boxed{1} \end{bmatrix}$$

C5
$$\begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{VAL} & \boxed{0} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{IND} & s_0 \end{bmatrix} \end{bmatrix} \rightarrow$$

$$\begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{VAL} & \boxed{0} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{IND} & s_1 \end{bmatrix} \end{bmatrix} \ldots \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{VAL} & \boxed{0} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{IND} & s_{n-1} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{HEAD} & \textit{conj} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{IND} & s_0 \\ \text{RESTR} & \langle \begin{bmatrix} \text{ARGS} & \langle s_1, \ldots, s_{n-1}, s_n \rangle \end{bmatrix} \rangle \end{bmatrix} \end{bmatrix} \begin{bmatrix} \text{SYN} & \begin{bmatrix} \text{VAL} & \boxed{0} \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \text{IND} & s_n \end{bmatrix} \end{bmatrix}$$

$$= \begin{bmatrix} \text{VAL} & \boxed{0} \\ \text{IND} & s_0 \end{bmatrix} \rightarrow \begin{bmatrix} \text{VAL} & \boxed{0} \\ \text{IND} & s_1 \end{bmatrix} \ldots \begin{bmatrix} \text{VAL} & \boxed{0} \\ \text{IND} & s_{n-1} \end{bmatrix} \begin{bmatrix} \text{HEAD} & \textit{conj} \\ \text{IND} & s_0 \\ \text{RESTR} & \langle \begin{bmatrix} \text{ARGS} & \langle s_1, \ldots, s_{n-1}, s_n \rangle \end{bmatrix} \rangle \end{bmatrix} \begin{bmatrix} \text{VAL} & \boxed{0} \\ \text{IND} & s_n \end{bmatrix}$$

# Combining Constraints and Coordination

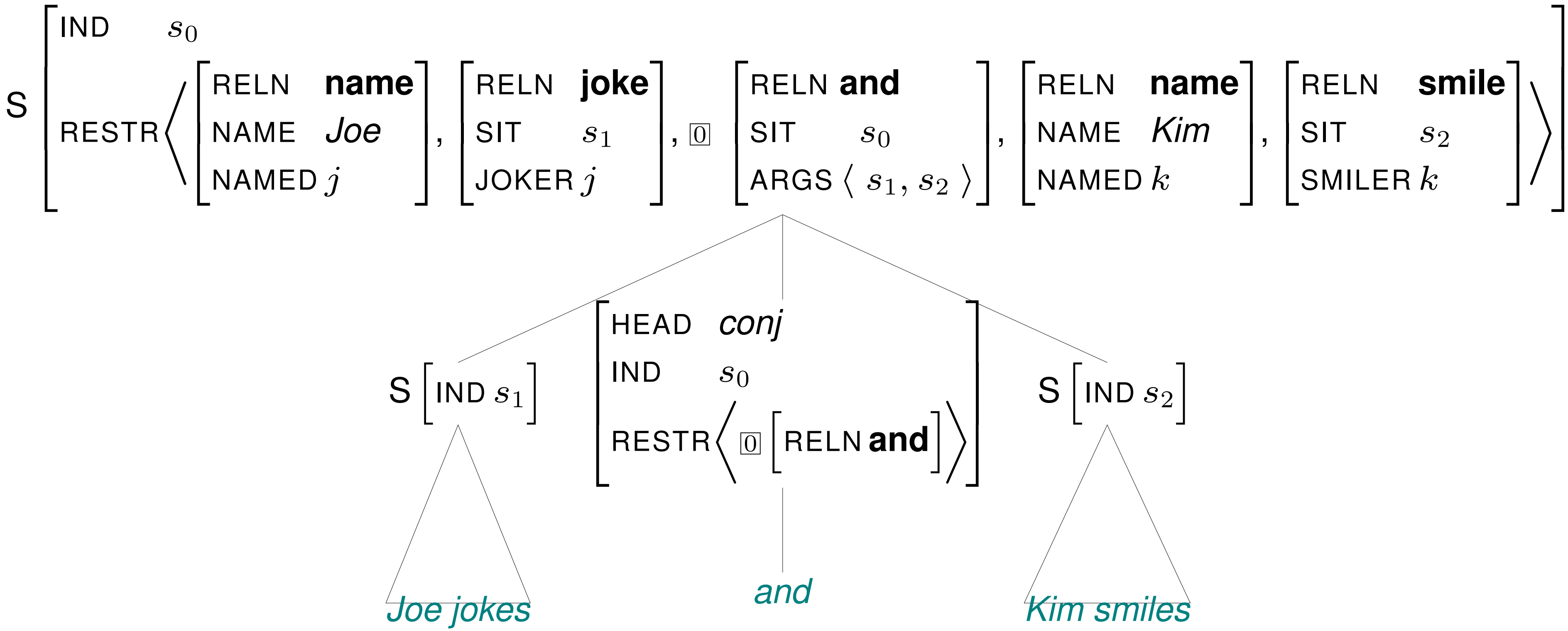
- Coordination Rule

$$\begin{bmatrix}\text{VAL} & \boxed{0}\\ \text{IND} & s_0\end{bmatrix} \rightarrow \begin{bmatrix}\text{VAL} & \boxed{0}\\ \text{IND} & s_1\end{bmatrix} \dots \begin{bmatrix}\text{VAL} & \boxed{0}\\ \text{IND} & s_{n-1}\end{bmatrix} \begin{bmatrix}\text{HEAD} & \textit{conj}\\ \text{IND} & s_0 \\ \text{RESTR} & \left\langle \begin{bmatrix}\text{ARGS} & \langle\, s_1,\dots,s_{n-1},s_n \,\rangle\end{bmatrix}\right\rangle\end{bmatrix} \begin{bmatrix}\text{VAL} & \boxed{0}\\ \text{IND} & s_n\end{bmatrix}$$

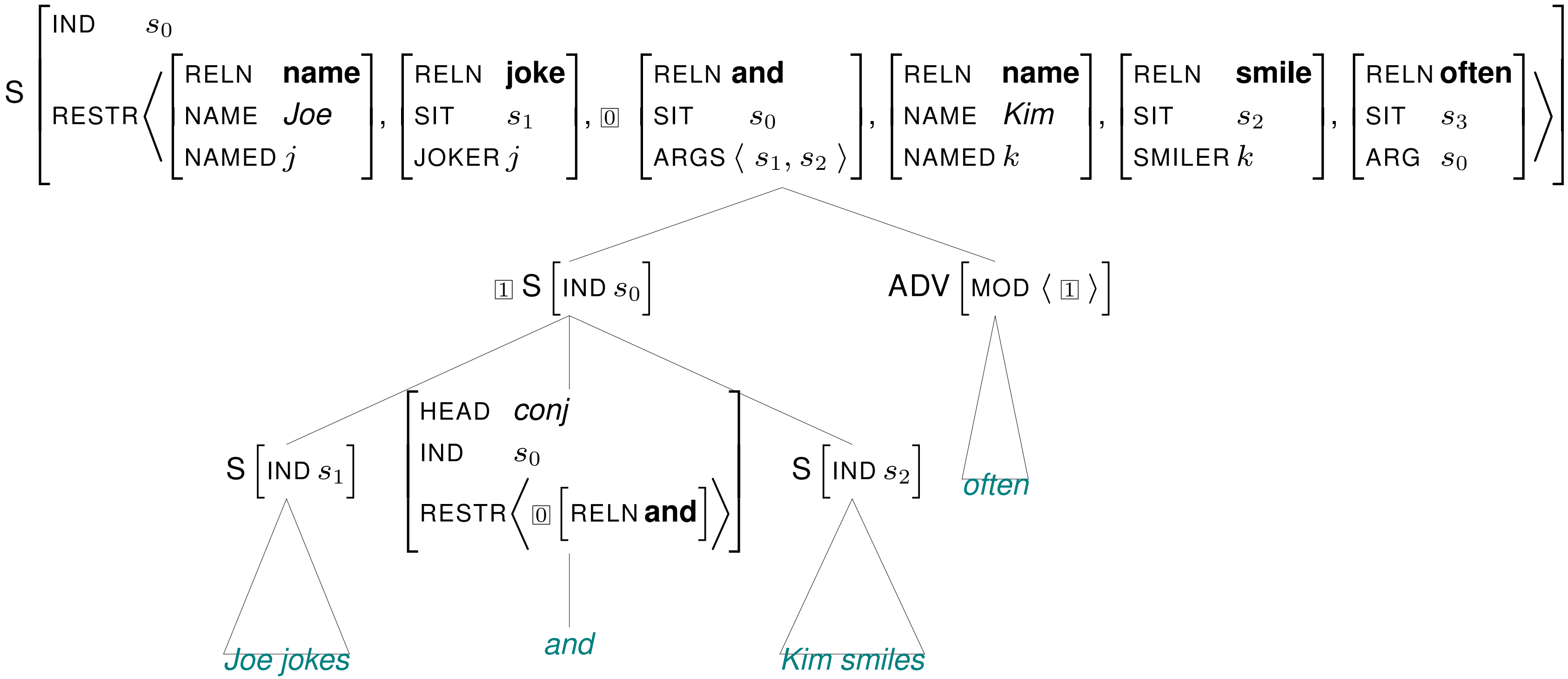
- Lexical Entry for ***and***

$$\left\langle \textit{and}\ ,\ \begin{bmatrix}\text{SYN} & \begin{bmatrix}\text{HEAD} & \textit{conj}\end{bmatrix}\\ \text{SEM} & \begin{bmatrix}\text{MODE} & \textit{none}\\ \text{IND} & s\\ \text{RESTR} & \left\langle \begin{bmatrix}\text{RELN} & \textbf{and}\\ \text{SIT} & s\end{bmatrix}\right\rangle\end{bmatrix}\end{bmatrix}\right\rangle$$

# Combining Constraints and Coordination

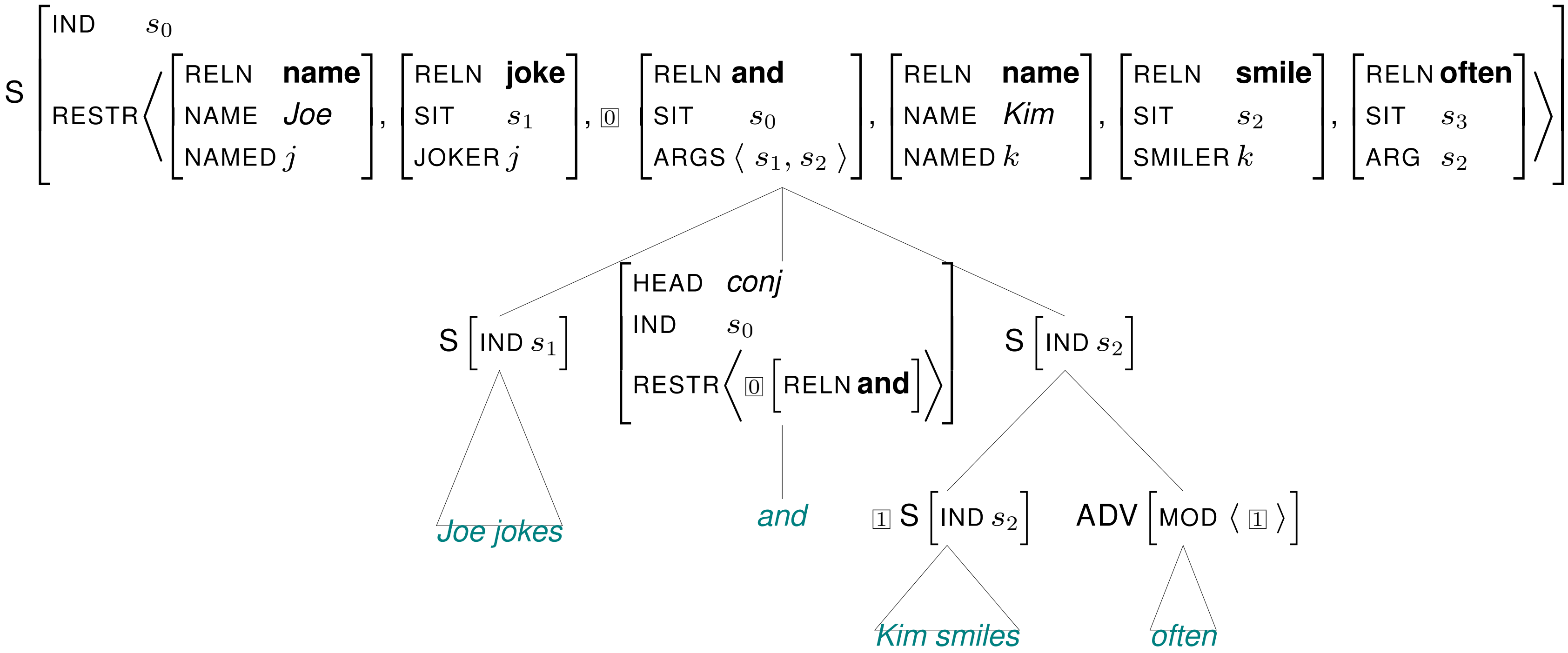


# Ambiguity



S
IND $s_0$
RESTR
RELN name
NAME Joe
NAMED $j$
RELN joke
SIT $s_1$
JOKER $j$
0
RELN and
SIT $s_0$
ARGS ⟨ $s_1$, $s_2$ ⟩
RELN name
NAME Kim
NAMED $k$
RELN smile
SIT $s_2$
SMILER $k$
RELN often
SIT $s_3$
ARG $s_0$
1 S [IND $s_0$]
ADV [MOD ⟨ 1 ⟩]
S [IND $s_1$]
HEAD conj
IND $s_0$
RESTR ⟨ 0 [RELN and] ⟩
S [IND $s_2$]
often
Joe jokes
and
Kim smiles

# Ambiguity


S
IND $s_0$
RESTR
RELN name
NAME Joe
NAMED $j$
RELN joke
SIT $s_1$
JOKER $j$
0
RELN and
SIT $s_0$
ARGS ⟨ $s_1$, $s_2$ ⟩
RELN name
NAME Kim
NAMED $k$
RELN smile
SIT $s_2$
SMILER $k$
RELN often
SIT $s_3$
ARG $s_2$
S IND $s_1$
HEAD conj
IND $s_0$
RESTR ⟨ 0 RELN and ⟩
S IND $s_2$
Joe jokes
and
1 S IND $s_2$
ADV MOD ⟨ 1 ⟩
Kim smiles
often

# Question About Structural Ambiguity

Why isnt this a possible semantic representation for the string *Joe jokes and Kim smiles often*?

$$\begin{bmatrix} \text{IND} & s_0 \\ \text{MODE} & \textit{prop} \\ \text{RESTR} & \left\langle \begin{bmatrix} \text{RELN} & \textbf{name} \\ \text{NAME} & \textit{Joe} \\ \text{NAMED} & j \end{bmatrix}, \begin{bmatrix} \text{RELN} & \textbf{joke} \\ \text{SIT} & s_1 \\ \text{JOKER} & j \end{bmatrix}, \boxed{0} \begin{bmatrix} \text{RELN} & \textbf{and} \\ \text{SIT} & s_0 \\ \text{ARGS} & \langle s_1, s_2 \rangle \end{bmatrix}, \begin{bmatrix} \text{RELN} & \textbf{name} \\ \text{NAME} & \textit{Kim} \\ \text{NAMED} & k \end{bmatrix}, \begin{bmatrix} \text{RELN} & \textbf{smile} \\ \text{SIT} & s_2 \\ \text{SMILER} & k \end{bmatrix}, \begin{bmatrix} \text{RELN} & \textbf{often} \\ \text{SIT} & s_3 \\ \text{ARG} & s_1 \end{bmatrix} \right\rangle \end{bmatrix}$$

# Some Standard Extensions

- Quantification
  - typically expressed as restrictions on scope
  - Minimal Recursion Semantics goes further
- Pragmatics
  - typically expressed as another feature: CONTEXT
  - contains things like *speaker, hearer, audience*
  - used for pronominal reference, politeness

# Problem: Two Kinds of Modifiers in English

In English, modifiers of nouns can appear either before or after the noun, although any given modifier is usually restricted to one position or the other.

(i) *The red dog on the roof*

(ii) **The on the roof dog*

(iii) **The dog red*

Our current Head-Modifier Rule only licenses post-head modifiers (like *on the roof* in (i)).

A. Write a second Head-Modifier Rule that licenses pre-head modifiers (e.g., *red* in (i)).

B. Modify the Head-Modifier 1 and Head-Modifier 2 Rules so that they are sensitive to which kind of modifier is present and don't generate (ii) or (iii). [*Hint: Use a feature* [*POST-HEAD* {*+,−*}] *to distinguish red and on the roof.*]

C. Is POST-HEAD a HEAD feature? Why or why not?

D. Give lexical entries for *red* and *on* that show the value of POST-HEAD. (You may omit the SEM features in these entries.)

E. Is (i) ambiguous according to your grammar (i.e. the Chapter 5 grammar modified to include the two Head-Modifier Rules, instead of just one)? Explain your answer.

This problem assumed that we don't want to make the two Head-Modifier Rules sensitive to the part of speech of the modifier. One reason for this is that modifiers of the same part of speech can occur before and after the head, even though individual modifiers might be restricted to one position or the other.

F. Provide three examples of English NPs with adjectives or APs after the noun.

G. Provide three examples of adverbs that can come before the verbs they modify.

H. Provide three examples of adverbs that can come after the verbs they modify.

# Problem: Semantics of Number Names

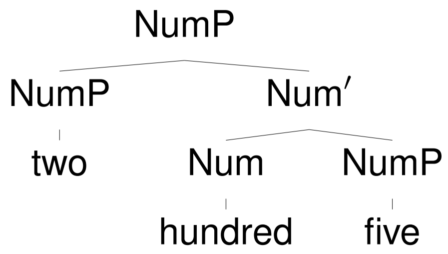
In Chapter 3, we considered the syntax of English number names, and in particular how to find the head of a number name expression. Based on the results of that problem, the lexical entry for *hundred* in a number name like *two hundred five* should include the constraints in (i): (Here we are assuming a new subtype of *pos*, *number*, which is appropriate for number name words.)

(i)

$$\left\langle \textit{hundred}\ ,\ \begin{bmatrix} \textit{SYN} & \begin{bmatrix} \textit{HEAD} & \textit{number} \\ \textit{VAL} & \begin{bmatrix} \textit{SPR} & \left\langle \begin{bmatrix} \textit{HEAD} & \textit{number} \end{bmatrix} \right\rangle \\ \textit{COMPS} & \left\langle \begin{bmatrix} \textit{HEAD} & \textit{number} \end{bmatrix} \right\rangle \end{bmatrix} \end{bmatrix} \end{bmatrix} \right\rangle$$

This lexical entry interacts with our ordinary Head-Complement and Head-Specifier Rules to give us the phrase structure shown in (ii):

(ii)




Smith (1999) provides a compositional semantics of number names. The semantics of this NP should be (iii):

(iii)

$$\begin{bmatrix} \textit{INDEX} & i \\ \textit{MODE} & \textit{ref} \\ \textit{RESTR} & \left\langle \begin{bmatrix} \textit{RELN} & \textbf{constant} \\ \textit{INST} & l \\ \textit{VALUE} & \textit{2} \end{bmatrix}, \begin{bmatrix} \textit{RELN} & \textbf{times} \\ \textit{RESULT} & k \\ \textit{FACTOR1} & l \\ \textit{FACTOR2} & m \end{bmatrix}, \begin{bmatrix} \textit{RELN} & \textbf{constant} \\ \textit{INST} & m \\ \textit{VALUE} & \textit{100} \end{bmatrix}, \begin{bmatrix} \textit{RELN} & \textbf{plus} \\ \textit{RESULT} & i \\ \textit{TERM1} & j \\ \textit{TERM2} & k \end{bmatrix}, \begin{bmatrix} \textit{RELN} & \textbf{constant} \\ \textit{INST} & j \\ \textit{VALUE} & \textit{5} \end{bmatrix} \right\rangle \end{bmatrix}$$

This expresses “(two times one hundred) plus five” (i.e. 205) as a FS.

A. Assume that the two constant predications with the values 2 and 5 are contributed by the lexical entries for *two* and *five*. What predications must be on the RESTR list of the lexical entry for *hundred* in order to build (iii) as the SEM value of *two hundred five*?

B. The lexical entry for *hundred* will identify the indices of its specifier and complement with the value of some feature of a predication on its RESTR list. Which feature of which predication is the index of the specifier identified with? What about the index of the complement?

C. The lexical entry for *hundred* will identify its own INDEX with the value of some feature of some predication on its RESTR list. Which feature of which predication must this be, in order for the grammar to build (iii) as the SEM value of *two hundred five*?

D. Based on your answers in parts (A)–(C), give a lexical entry for *hundred* that includes the constraints in (i) and a fully specified SEM value. [*Note: Your lexical entry need only account for hundred as it is used in two hundred five. Don't worry about other valence possibilities, such as two hundred, two hundred and five, or a hundred.*]

E. The syntax and semantics of number names do not line up neatly: In the syntax, *hundred* forms a constituent with *five*, and *two* combines with *hundred five* to give a larger constituent. In the semantics, the constant predications with the values 2 and 100 are related via the times predication. The result of that is related to the constant predication with the value 5, via the plus predication Why is this mismatch not a problem for the grammar?

# Overview

- Some notes on the linguist's stance
- Which aspects of semantics well tackle
- Our formalization; Semantics Principles
- Building semantics of phrases
- Modification, coordination
- Structural ambiguity

# Acknowledgments and References

- Course design and slides borrow heavily from Emily Bender's course: *Linguistics 566: Introduction to Syntax for Computational Linguistics* `http://courses.washington.edu/ling566`

# Questions

**C**: Please send to `bond@ieee.org>` with subject HG4041

**Q**: The value of INDEX is an index corresponding to the situation or individual referred to." Is the purpose of the index is to help assimilate the roles in RESTR with the syn-cat part of the lexical entry? Because in the examples it just seems like a reference point for the role of the word in the situation given.

**A**: It does both: allows us to link bits together and gives a place to link to the real world.

**Q**: Is the list of restrictions infinite? (If I read the book right, yes) if so, how will we know what to specify under it for each lexical entry?

**A**: Each lexical entry only introduces a finite number of indices (typically one), but we can refer to an infinite number of things (*the red cat and the blue cat and the pink cat, . . .* so the grammar must allow for any number.

**Q**: How does the content in section 6.3 link with feature structures?

**A**: This is a more formal description of features structures in terms of set theory.

**Q**: I don't really know how to read and understand these parts (with the mathematical formula type things).

**C**: As it says in the book, we can skip over this if we are not interested in formal foundations.

**Q**: [. . . ] on the Satisfaction portion in Chapter 6 - How the Grammar Works. In page 193-195, I do not understand a single thing from this section despite reading it over and over again. How do I make sense of this portion?

**Q**: How do we specify/represent (with the use of coding symbols) the pronunciation of Japanese homographs in a lexical entry? For example with the use of the kanji ''? Do we represent it like '/mizu/' and '/sui/' beside the kanji?

**A**: If you want it to be part of the grammar, then add a new feature HATSUON (this forces the two to be two different entries)

**A**: If it is just for the reader (a gloss) then you can write it next to it. In this case it should be in italics.

**Q**: I hope you have had a good weekend. My question with regards to the reading is: Most of the examples given were about situations that may or may not be true (for example, Kim is running). What about cases such as metaphors (for example, Life is a rollercoaster) where the literal meaning of it is almost always never true but the listener would still understand it to be true?

**A**: We need to have some kind of semantic type coercion: “life is [a thing like] a rollercoaster”. This is very hard to constrain. For very common metaphors “It happened before” the coerced meaning “previous in time” may also be lexicalized (so we don’t think of it as being coerced).

**Q**: In Chapter 5 Semantics, there is introduction of a BV value, or Bound Variable value. There isn't much on it but it is repeated quite a fair bit in Chapter 6. I've tried looking it up and I wonder if it means that the noun cannot stand on its own, for example ”Family eats together” but it has to be ”This family/ Every family” etc?

**A**: This says that the set of individuals introduced by this predicate is bound in scope by this quantifier (recall HG2002). We don't go into this further.

**Q**: From the example in (46) and the subsequent explanation of the use of Quantifier Scope, is it only used to solve ambiguities in sentences; are there any other instances where the scope can be applied to other than quantifiers?

**A**: It is also used for negation and some modals (like *maybe*)

**Q**: For e.g. for an ambiguous sentence like that, could it be possible to use the function of a scope to support understanding of the two meanings on the semantic level?

*Look at the dog with one eye*

Or would it be that this ambiguity could just be/ needs to be solved at the syntactic level?

**A**: Yes

**C**: I am a bit confused with Chapter 5, Page 139 and Page 140. In those two pages there were explaining about INST(ANCE), SIT(UATION), REST(RICTION) and predictions.

**Q**: However, when do we input INST(ANCE)s and SIT(UATION)s. I am guessing they are not interchangeable but what is the difference between the two?

**A**: Referring expressions (things) are instances, everything else is a situation (verbs, adjectives, . . . )

**Q**: Are SIT(UATION)s just for REST(RICTION)s? **A**: they are used to lnik

2. In example (17), why is the INDEX and the SIT indexed as the same letter? Is it**Q**: because they are referring to the same occurence? Then, whats the difference between the two? Cant we change INDEX to SIT?

**A**: SIT is within the predicate, INDEX is in the grammar. In (17), the sentence as a whole refers to the situation of **save** .

**Q**: Are predictions only contained to RESTs? Could other values or entitities be in the REST as well? Other than predictions.

**A**: No, just predications

**C**: I found how grammar analysis was presented in Chapter 5 to be similar to logic in Mathematics (eg.   x: P(x) with the notion of predicates) to be interesting in that I did not expect maths and linguistics to have so much in common.

**Q**: However, I do not fully understand the section about QSCOPE. Why was the QSCOPE in (51) for the second interpretation of the sentence ”a dog saved every family.” tagged to ”save” and not ”family”?

**A**: The restriction of *every* is only to those families who were saved by dogs (the sentence will still be true if there are families not saved by dogs who are outside of our interest).

The Semantics of Questions:

**Q**: short version: Is the way presented in this book (MODE/SF) the proper / state-of-the-art way of representing questions when considering the many types of questions that exist?

**A**: Not at all. Ginzburg and Sag have a much fuller representation.
Jonathan Ginzburg and Ivan Sag (2000) *Interrogative Investigations: The Form, Meaning and Use of English Interrogatives*. Stanford: CSLI.

long version: I felt that the semantics of questions elaborated by simple stating MODE (SentenceForce) = question to be slightly unsatisfying. It seems there there are different types of questions that postulate slightly different things:

Proposition: John left.

```
Question type1: Did John leave?
Question type2: John left?
Question type3: Did someone leave?
Question type4: Who left?

Question type5: Who was it that left?
Question type6: Was it John that left?
```

It seems that both type1 and type2 have the same semantics of the proposition with MODE (SentenceForce) of question. Type3 is consistent with the above, transforming someone into some+person (which seems fair enough). For type4, who is transformed into which-person, and MODE (SentenceForce) is also marked as question for leave. For type 5, a similar thing happens to who, but the SF is marked on the verb be (I can't quite follow the ERG semantics, I'm not sure if everything is correctly linked for this one.) Type6 also seems to give complex semantics which I am not sure how to evaluate if they are correct (i.e. be_v_itcleft)

There seems to be some semantic value that goes beyond pragmatic usage of the above sentences, namely the degree of certainty about related propositions or the kind of question that is being asked about said proposition.

type1 seems to ascertain nothing about the related proposition. (a true unbiased question if not marked pragmatically: did John leave OR did John not leave?). type2 seems to lean on the likelihood that John has actually already left (asking for a confirmation of some sort: is it true that John leave? or maybe TRUE OR FALSE: JOHN LEFT).

type3 and type4 behave similarly to the example above, but for generic pronouns.

type5 and type6 assume something altogether, basically postulates that someone left and it either asks who, or wants to confirm if it was John (again, probably with a focus on John?). type5: SOMEONE LEFT. WHICH PERSON LEFT? type6: SOMEONE LEFT. TRUE OR FALSE: JOHN LEFT.

Returning to the topic of a more formulaic question, have questions been analysed by the entailment they might offer? Can a yes/no question be analysed as a choice of disjunctive meanings?

**A**: Yes There are two kinds of questions Yes/No (polar) and WH (open-ended) (and also echo: *You paid how much?*). They are both SF *ques*, but only one has an interrogative *which*.

p.s. I left out negative counterparts of the above questions, but they would provide other fun experiments. e.g. ”Did John not leave?” [I assumed John had left but I now have evidence that contradicts this belief. Did John leave?] These might be a bit more pragmatic.