# Ling 566
# Oct 11, 2016

## Valence, Agreement

# Overview

- Review: pizza, feature structures, well-formed trees, HFP
- A problem with the Chapter 3 grammar
- Generalize COMPS and SPR
- The Valence Principle
- Agreement
- The SHAC
- Reading Questions

# Pizza review

- Unification is an operation for combing constraints from different sources.
- What are those sources in the pizza example?
- Why do we need to combine information from different sources in our grammars?

# Reminder: Where We Are

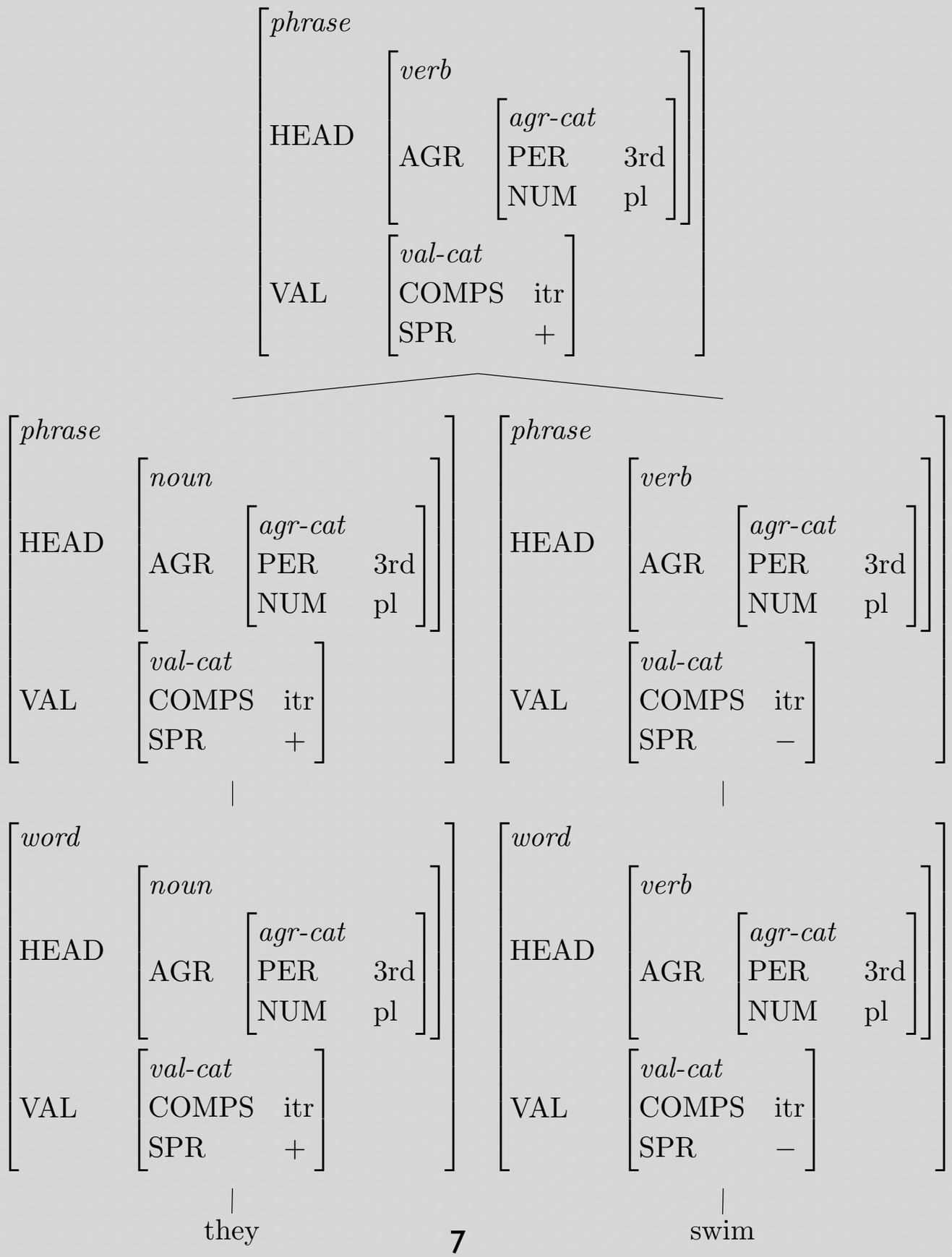
- Attempting to model English with CFG led to problems with the granularity of categories, e.g.
  - Need to distinguish various subtypes of verbs
  - Need to identify properties common to all verbs
- So we broke categories down into feature structures and began constructing a hierarchy of types of feature structures.
- This allows us to schematize rules and state cross-categorial generalizations, while still making fine distinctions.

# A Tree is Well-Formed if …

- It and each subtree are licensed by a grammar rule or lexical entry
- All general principles (like the HFP) are satisfied.
- NB: Trees are part of our model of the language, so all their features have values (even though we will often be lazy and leave out the values irrelevant to our current point).

# The Head Feature Principle

- Intuitive idea:  Key properties of phrases are shared with their heads
- The HFP:  In any headed phrase, the HEAD value of the mother and the head daughter must be identical.
- Sometimes described in terms of properties "percolating up" or "filtering down", but this is just metaphorical talk



# But it's still not quite right…

- There's still too much redundancy in the rules.
- The rules and features encode the same information in different ways.

Head-Complement Rule 1:

$$\begin{bmatrix} \textit{phrase} \\ \text{VAL} \quad \begin{bmatrix} \text{COMPS} & \text{itr} \\ \text{SPR} & - \end{bmatrix} \end{bmatrix} \rightarrow \mathbf{H} \begin{bmatrix} \textit{word} \\ \text{VAL} \quad \begin{bmatrix} \text{COMPS} & \text{itr} \\ \text{SPR} & - \end{bmatrix} \end{bmatrix}$$

Head Complement Rule 2:

$$\begin{bmatrix} \textit{phrase} \\ \text{VAL} \quad \begin{bmatrix} \text{COMPS} & \text{itr} \\ \text{SPR} & - \end{bmatrix} \end{bmatrix} \rightarrow \mathbf{H} \begin{bmatrix} \textit{word} \\ \text{VAL} \quad \begin{bmatrix} \text{COMPS} & \text{str} \\ \text{SPR} & - \end{bmatrix} \end{bmatrix} \text{ NP}$$

Head Complement Rule 3:

$$\begin{bmatrix} \textit{phrase} \\ \text{VAL} \quad \begin{bmatrix} \text{COMPS} & \text{itr} \\ \text{SPR} & - \end{bmatrix} \end{bmatrix} \rightarrow \mathbf{H} \begin{bmatrix} \textit{word} \\ \text{VAL} \quad \begin{bmatrix} \text{COMPS} & \text{dtr} \\ \text{SPR} & - \end{bmatrix} \end{bmatrix} \text{ NP NP}$$

# Solution:
# More Elaborate Valence Feature Values

- The rules just say that heads combine with whatever their lexical entries say they can (or must) combine with.
- The information about what a word can or must combine with is encoded in list-valued valence features.
  - The elements of the lists are themselves feature structures
  - The elements are "cancelled" off the lists once heads combine with their complements and specifiers.

# Complements

Head-Complement Rule:

$$\begin{bmatrix} phrase \\ \text{VAL} \quad \left[\text{COMPS} \quad \langle\ \rangle\right] \end{bmatrix} \rightarrow \mathbf{H}\begin{bmatrix} word \\ \text{VAL} \quad \left[\text{COMPS} \quad \langle \boxed{1}, \ldots, \boxed{n} \rangle\right] \end{bmatrix} \boxed{1}, \ldots, \boxed{n}$$

- This allows for arbitrary numbers of complements, but only applies when there is at least one.
  - Heads in English probably never have more than 3 or 4 complements
  - This doesn't apply where Head-Complement Rule 1 would. (Why?)
- This covers lots of cases not covered by the old Head-Complement Rules 1-3. (Examples?)

# Specifiers

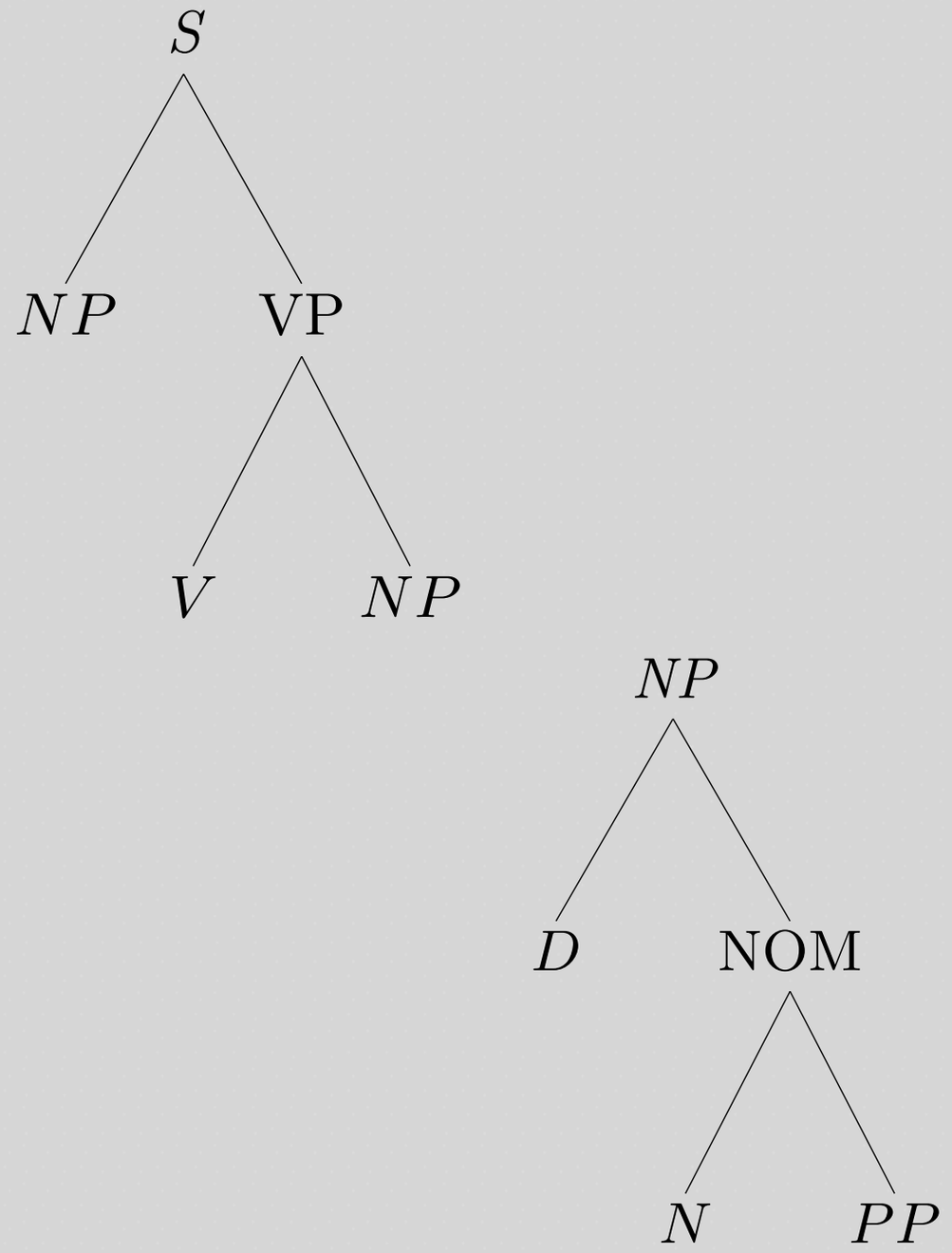
## Head-Specifier Rule (Version I)

$$\begin{bmatrix} phrase \\ \text{VAL} \begin{bmatrix} \text{COMPS} & \langle\,\rangle \\ \text{SPR} & \langle\,\rangle \end{bmatrix} \end{bmatrix} \rightarrow \boxed{2}\ \mathbf{H}\begin{bmatrix} \text{VAL} \begin{bmatrix} \text{COMPS} & \langle\,\rangle \\ \text{SPR} & \langle\,\boxed{2}\,\rangle \end{bmatrix} \end{bmatrix}$$

- Combines the rules expanding S and NP.
- In principle also generalizes to other categories.
- Question: Why is SPR list-valued?

# Question:

Why are these right-branching? That is, what formal property of our grammar forces the COMPS to be lower in the tree than the SPR?



# Another Question…

What determines the VAL value of phrasal nodes?

ANSWER: The Valence Principle

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

# More on the Valence Principle

- Intuitively, the VAL features list the contextual requirements that haven't yet been found.
- This way of thinking about it (like talk of "cancellation") is bottom-up and procedural.
- But formally, the Valence Principle (like the rest of our grammar) is just a well-formedness constraint on trees, without inherent directionality.

# So far, we have:

- Replaced atomic-valued VAL features with list-valued ones.
- Generalized Head-Complement and Head-Specifier rules, to say that heads combine with whatever their lexical entries say they should combine with.
- Introduced the Valence Principle to "cancel" things off the COMPS and SPR lists.

# The Parallelism between S and NP

- Motivation:
  - pairs like *Chris lectured about syntax* and *Chris's lecture about syntax*.
  - both S and NP exhibit agreement

    *The bird sings/*sing* vs. *The birds sing/ *sings*

    *this/*these bird* vs. *these/*this birds*
- So we treat NP as the saturated category of type *noun* and S as the saturated category of type *verb*.

# Question: Is there any other reason to treat V as the head of S?

- In standard English, sentences must have verbs. (How about non-standard English or other languages?)
- Verbs taking S complements can influence the form of the verb in the complement:

  *I insist/*recall (that) you be here on time*.
- Making V the head of S helps us state such restrictions formally

# A possible formalization of the restriction on *insist*

$$\left[\begin{array}{ll}\text{HEAD} & \textit{verb} \\ \text{VAL} & \left[\begin{array}{ll}\text{SPR} & \left\langle \text{NP} \right\rangle \\ \text{COMPS} & \left\langle \left[\begin{array}{ll}\text{HEAD} & \left[\begin{array}{ll}\textit{verb} \\ \text{MOOD} & \textit{subjunctive}\end{array}\right] \\ \text{VAL} & \left[\begin{array}{ll}\text{COMPS} & \langle\ \rangle \\ \text{SPR} & \langle\ \rangle\end{array}\right]\end{array}\right] \right\rangle\end{array}\right]\end{array}\right]$$

Note that this requires that the verb be the head of the complement. We don't have access to the features of the other constituents of the complement.

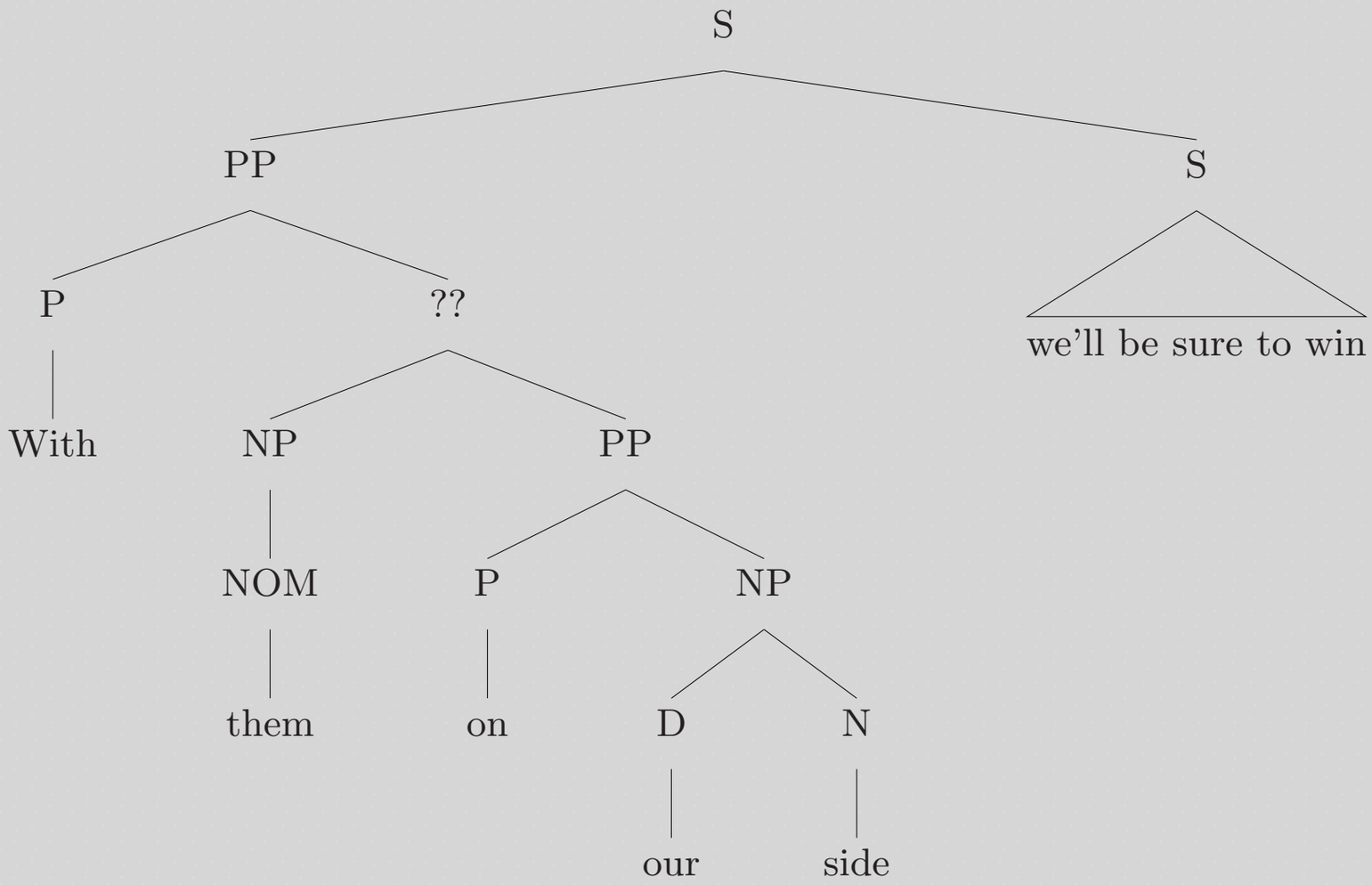
# Reading Questions

- On figure(51) page 113, why does SPR include another VAL feature?

$$\left\langle \text{dog} , \begin{bmatrix} word \\ \text{HEAD} & \begin{bmatrix} noun \\ \text{AGR} & \boxed{1} \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \left\langle \begin{bmatrix} \text{HEAD} & \begin{bmatrix} det \\ \text{AGR} & \boxed{1}3sing \\ \text{COUNT} & + \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SPR} & \langle\,\rangle \\ \text{COMPS} & \langle\,\rangle \end{bmatrix} \end{bmatrix} \right\rangle \\ \text{COMPS} & \langle\,\rangle \end{bmatrix} \end{bmatrix} \right\rangle$$

# Reading Questions

- What would a tree for *With them on our team, we'll be sure to win* look like?



# An Overlooked Topic: Complements *vs*. Modifiers

- Intuitive idea: Complements introduce essential participants in the situation denoted; modifiers refine the description.
- Generally accepted distinction, but disputes over individual cases.
- Linguists rely on heuristics to decide how to analyze questionable cases (usually PPs).

# Heuristics for Complements vs. Modifiers

- Obligatory PPs are usually complements.
- Temporal & locative PPs are usually modifiers.
- An entailment test: If **X Ved (NP) PP** does not entail **X did something PP**, then the PP is a complement.

Examples

- *Pat relied on Chris* **does not entail** *Pat did something on Chris*
- *Pat put nuts in a cup* **does not entail** *Pat did something in a cup*
- *Pat slept until noon* **does entail** *Pat did something until noon*
- *Pat ate lunch at Bytes* **does entail** *Pat did something at Bytes*

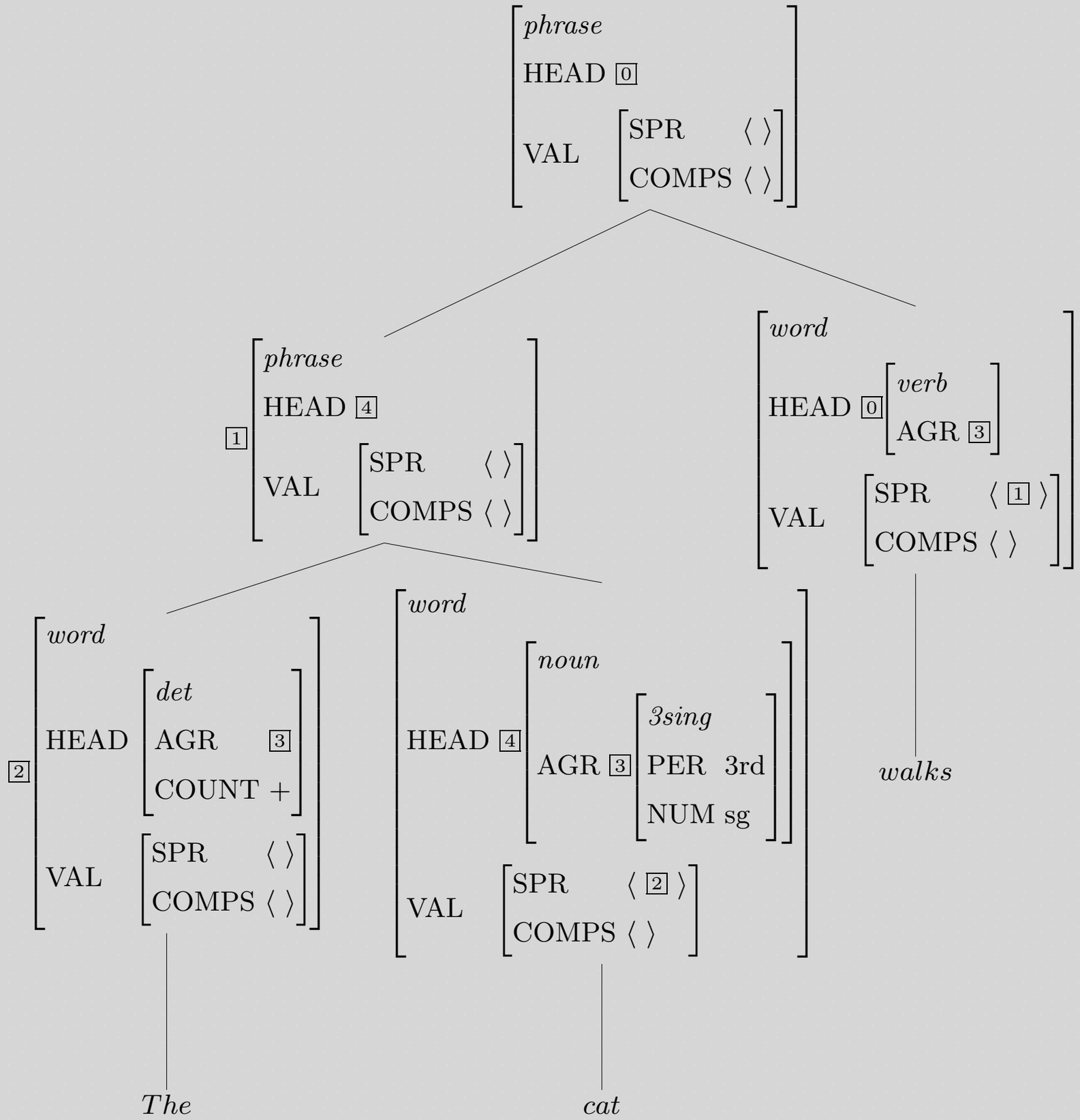
# Agreement

- Two kinds so far (namely?)
- Both initially handled via stipulation in the Head-Specifier Rule
- But if we want to use this rule for categories that don't have the AGR feature (such as PPs and APs, in English), we can't build it into the rule.

# The Specifier-Head Agreement Constraint (SHAC)

Verbs and nouns must be specified as:

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



# The Count/Mass Distinction

- Partially semantically motivated
  - mass terms tend to refer to undifferentiated substances (*air, butter, courtesy, information*)
  - count nouns tend to refer to individuatable entities (*bird, cookie, insult, fact*)
- But there are exceptions:
  - *succotash* (mass) denotes a mix of corn & lima beans, so it's not undifferentiated.
  - *furniture, footwear, cutlery*, etc. refer to individuatable artifacts with mass terms
  - *cabbage* can be either count or mass, but many speakers get *lettuce* only as mass.
  - borderline case: *data*

# Our Formalization of the Count/Mass Distinction

- Determiners are:
  - [COUNT –] (*much* and, in some dialects, *less*),
  - [COUNT +] (*a, six, many*, etc.), or
  - lexically underspecified (*the, all, some*, *no*, etc.)
- Nouns select appropriate determiners
  - "count nouns" say SPR <[COUNT +]>
  - "mass nouns" say SPR <[COUNT –]>
- Nouns themselves aren't marked for the feature COUNT
- So the SHAC plays no role in count/mass marking.

# Overview

- Review: pizza, feature structures, well-formed trees, HFP
- A problem with the Chapter 3 grammar
- Generalize COMPS and SPR
- The Valence Principle
- Agreement
- The SHAC
- Reading Questions

# Reading Questions

- If the HCR gives the head exactly the complements it's expecting, how do we get the right P in examples like depend on/ depend with?
- How would we handle flexible word order languages?
- Do all phrases have to be COMPS < >? Why?

# Reading Questions

- Are the Ch 4 trees flatter than the Ch 3 trees? Why?
- Why is eliminating non-branching nodes an improvement/desirable?
- How do we handle cases like *Stop the music* where the VP doesn't take a specifier? Or cases like *I like opera* where the N doesn’t?

# Reading Questions

- Is there an order of operation that needs to be followed while applying the rules (HSR, HCR, and HMR, etc.) or are the rules already accounting for any discrepancies between them that it isn't an issue the order they are applied?
- I would like to know what is meant by the phrase "left edge" here: "lexical head daughters in English uniformly occur at the left edge of their phrases."

# Reading Questions

- In the valence principle, it says unless otherwise specified, VAL features are shared between mother and head daughter. What is an example that contains an "otherwise specified" rule?
- What is "saturation" and how does the empty list encode it?

# Reading Questions

- Are the updated rules in (30) in 4.5 underspecified where there is <> or are they required to have an empty list there?
- Is this grammar still order-independent (top-down same as bottom-up)?

# Reading Questions

- Why would we want the two-branching structure for head-complement phrases (fn 5, p.97) in grammar engineering?
- If most verbs only do 3sing v. non-3sing, why do we need the rest of that hierarchy? Isn't it redundant?

# Reading Questions

- Why are we ignoring generalizations about e.g. walks and walk sharing lots of info?
- Do we handle lexical ambiguity with underspecification or multiple lexical entries for the same form?