



Ling 566  
Jan 16, 2019  
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 combining 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

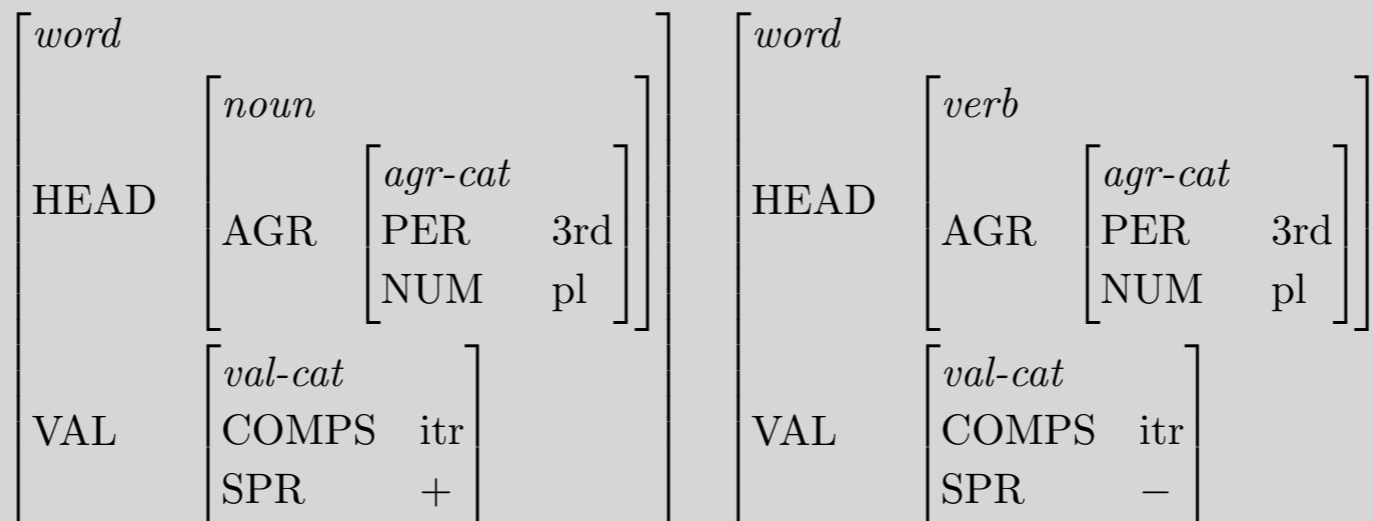
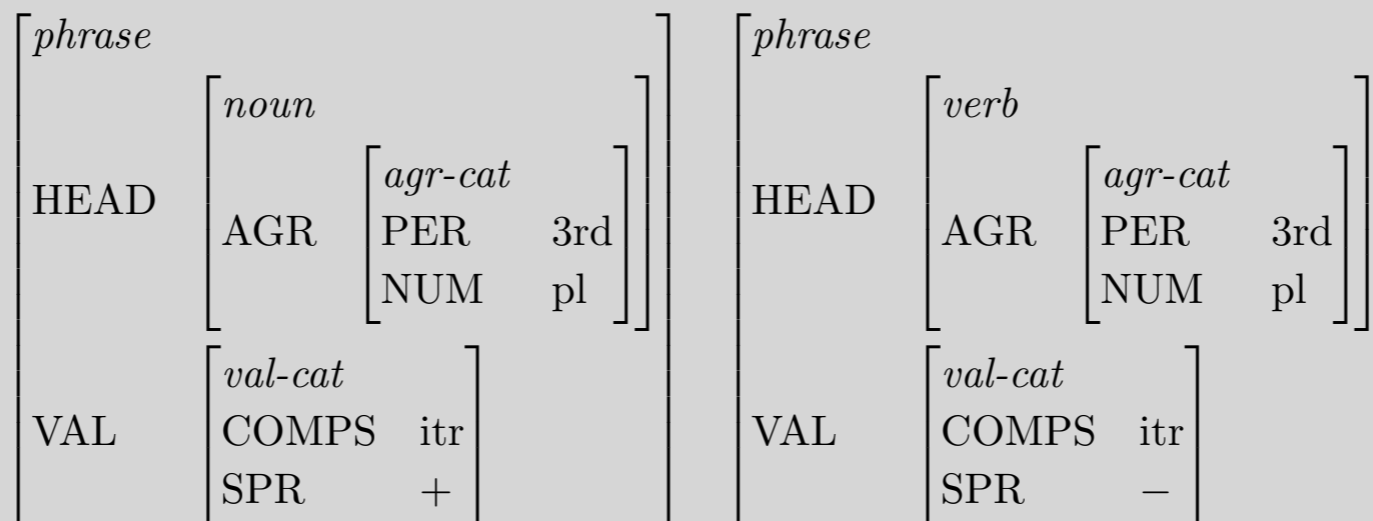
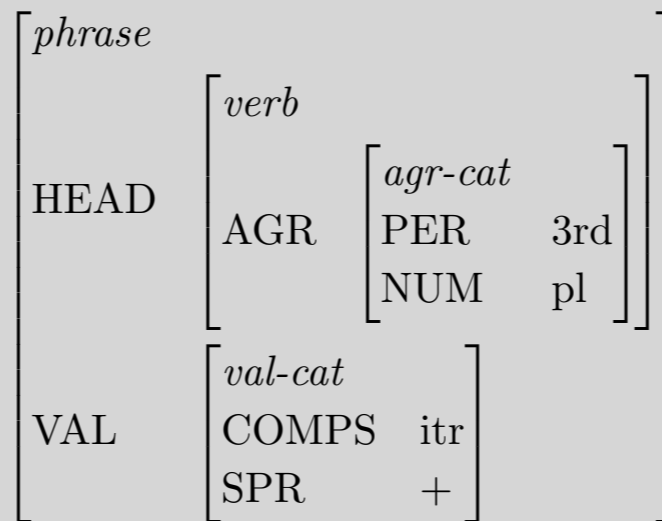
- Attempting to model English with CFG led to problems with the granularity of categories, e.g.
  - Need to distinguish various subtypes of verbs
  - Need to identify properties common to all verbs
- So we broke categories down into feature structures and began constructing a hierarchy of types of feature 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



they

swim

# 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:

$$\left[ \begin{array}{l} \textit{phrase} \\ \text{VAL} \end{array} \left[ \begin{array}{ll} \text{COMPS} & \textit{itr} \\ \text{SPR} & - \end{array} \right] \right] \rightarrow \mathbf{H} \left[ \begin{array}{l} \textit{word} \\ \text{VAL} \end{array} \left[ \begin{array}{ll} \text{COMPS} & \textit{itr} \\ \text{SPR} & - \end{array} \right] \right]$$

Head Complement Rule 2:

$$\left[ \begin{array}{l} \textit{phrase} \\ \text{VAL} \end{array} \left[ \begin{array}{ll} \text{COMPS} & \textit{itr} \\ \text{SPR} & - \end{array} \right] \right] \rightarrow \mathbf{H} \left[ \begin{array}{l} \textit{word} \\ \text{VAL} \end{array} \left[ \begin{array}{ll} \text{COMPS} & \textit{str} \\ \text{SPR} & - \end{array} \right] \right] \text{ NP}$$

Head Complement Rule 3:

$$\left[ \begin{array}{l} \textit{phrase} \\ \text{VAL} \end{array} \left[ \begin{array}{ll} \text{COMPS} & \textit{itr} \\ \text{SPR} & - \end{array} \right] \right] \rightarrow \mathbf{H} \left[ \begin{array}{l} \textit{word} \\ \text{VAL} \end{array} \left[ \begin{array}{ll} \text{COMPS} & \textit{dtr} \\ \text{SPR} & - \end{array} \right] \right] \text{ NP NP}$$



# 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:

$$\left[ \begin{array}{l} \textit{phrase} \\ \text{VAL} \left[ \text{COMPS} \langle \rangle \right] \end{array} \right] \rightarrow \mathbf{H} \left[ \begin{array}{l} \textit{word} \\ \text{VAL} \left[ \text{COMPS} \langle \boxed{1}, \dots, \boxed{n} \rangle \right] \end{array} \right] \boxed{1}, \dots, \boxed{n}$$

- 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

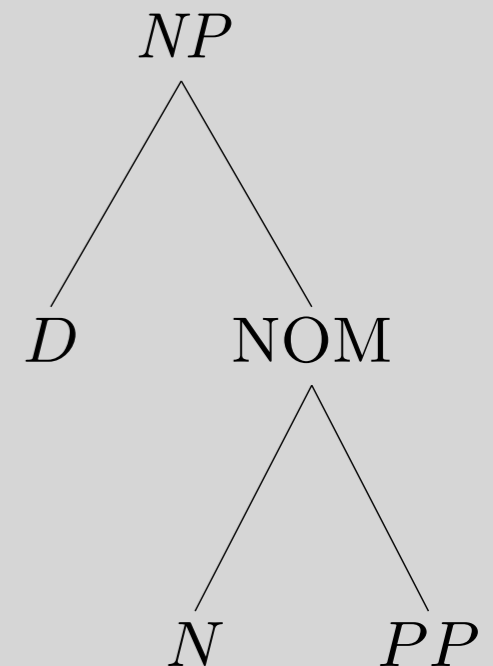
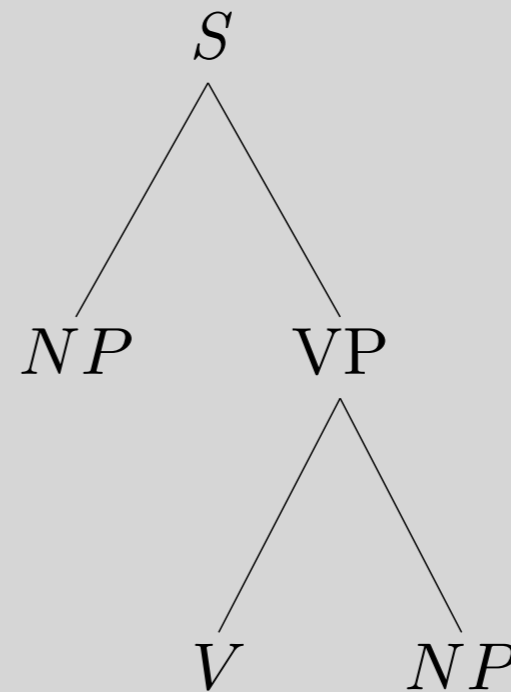
## Head-Specifier Rule (Version I)

$$\left[ \begin{array}{l} \textit{phrase} \\ \text{VAL} \left[ \begin{array}{l} \text{COMPS} \langle \rangle \\ \text{SPR} \langle \rangle \end{array} \right] \end{array} \right] \rightarrow \boxed{2} \mathbf{H} \left[ \begin{array}{l} \text{VAL} \left[ \begin{array}{l} \text{COMPS} \langle \rangle \\ \text{SPR} \langle \boxed{2} \rangle \end{array} \right] \end{array} \right]$$

- 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 carry up what's not “canceled”.

# The Parallelism between S and NP

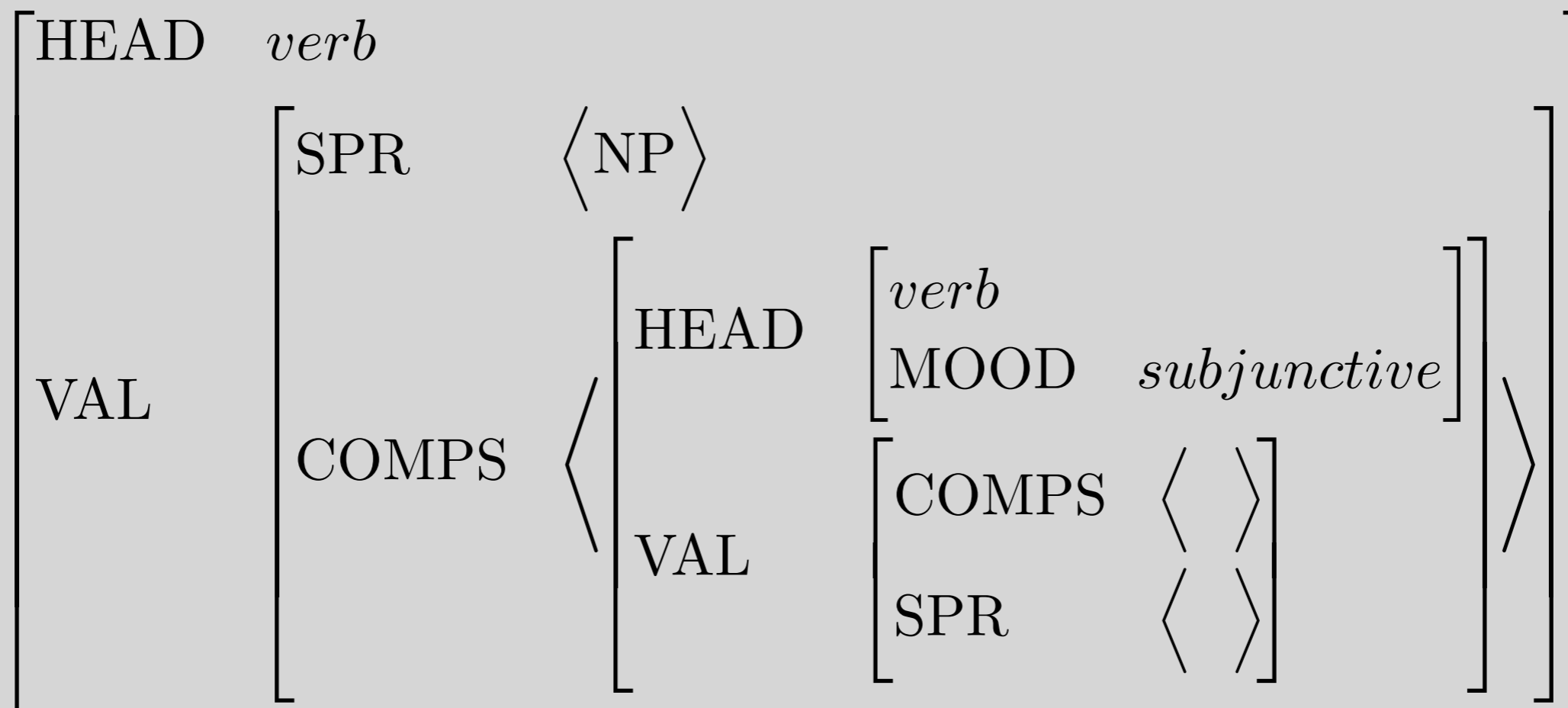
- Motivation:
  - pairs like *Chris lectured about syntax* and *Chris's lecture about syntax*.
  - both S and NP exhibit agreement
    - The bird sings/\*sing* vs. *The birds sing/\*sings*
    - this/\*these bird* vs. *these/\*this birds*
- So we treat NP as the saturated category of type *noun* and S as the saturated category of type *verb*.



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

- In mainstream American English, sentences must have verbs. (How about other varieties of 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*



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.

# 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*

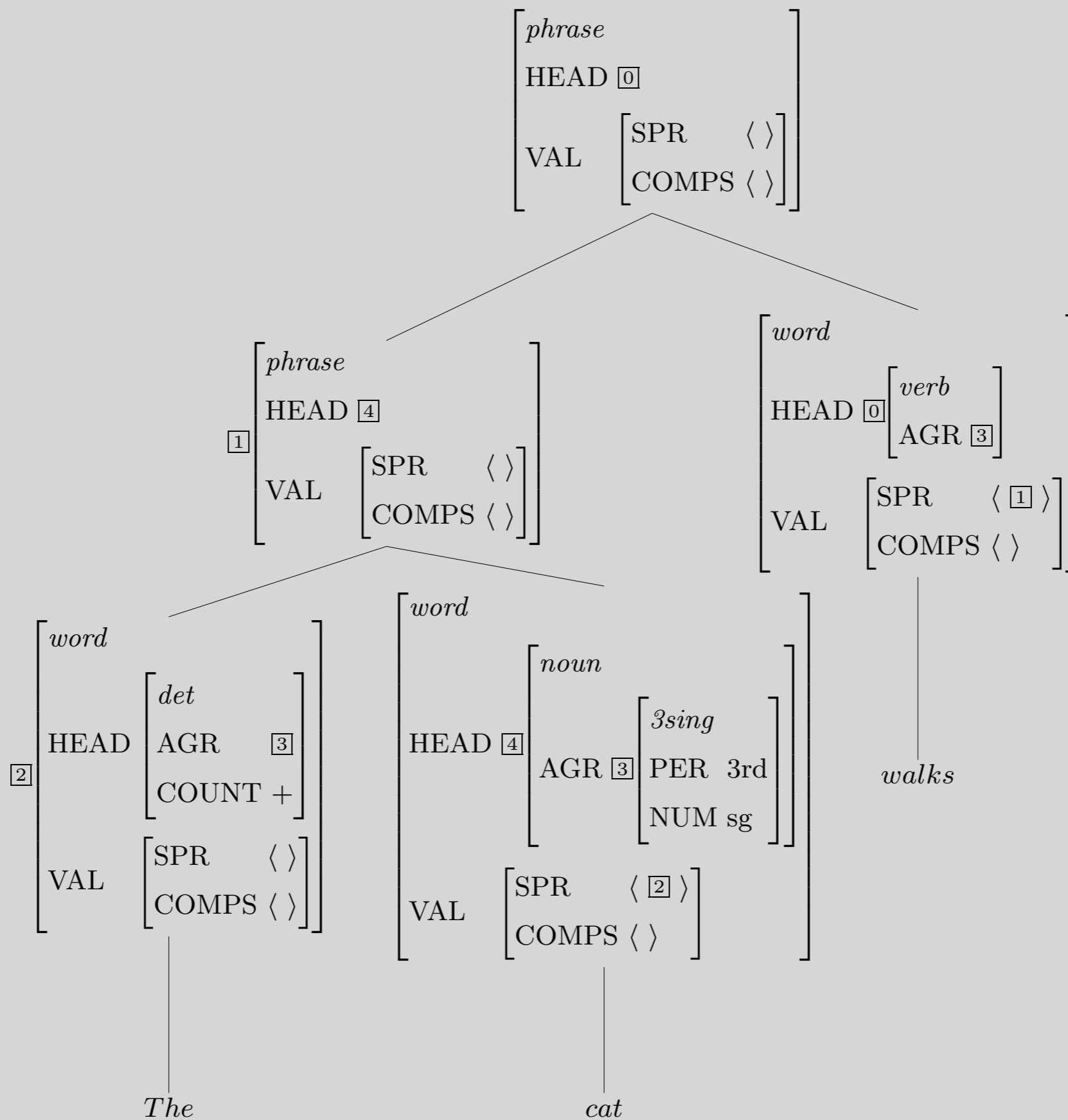
# Agreement

- 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:

$$\left[ \begin{array}{l} \text{HEAD} \left[ \text{AGR} \quad \boxed{1} \right] \\ \text{VAL} \left[ \text{SPR} \left\langle \left[ \text{AGR} \quad \boxed{1} \right] \right\rangle \right] \end{array} \right]$$



# 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.

## RQ: Conversion between count and mass

- Universal packager: a milk, a chocolate
- Universal grinder: cat all over the driveway
  
- Name to common noun: a *Newsweek* about crime, the Kim we met yesterday

# 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

- p. 113: "*\_he* will be unmarked for this feature [COUNT]". What does it mean for a lexical entry to be "unmarked" for a certain feature? Does this have any counterpart in our beloved pizza example?
- As (3)e&f on Page 97 show, some elements listed in COMPS should be optional. How could we show the optionality in rules?

# Reading Questions

- What is [2] doing in the Head-Specifier Rule? What is a scenario where a head-daughter and non-head daughter would have to agree?

$$\left[ \begin{array}{l} \textit{phrase} \\ \text{VAL} \left[ \begin{array}{l} \text{COMPS} \langle \rangle \\ \text{SPR} \langle \rangle \end{array} \right] \end{array} \right] \rightarrow \boxed{2} \mathbf{H} \left[ \begin{array}{l} \text{VAL} \left[ \begin{array}{l} \text{COMPS} \langle \rangle \\ \text{SPR} \langle \boxed{2} \rangle \end{array} \right] \end{array} \right]$$

# Reading Questions

- Why do we posit that D & N must have the same value for their entire AGR feature? Why can't it just be required that they only have the same value for their NUM feature? The NP *my daughter* is made up of a D that is 1st person and singular but a noun that is 3rd person and singular.
- How does the SHAC work with nouns that don't have to take specifiers? What about imperative verbs?

# Reading Questions

- I am wondering why we would treat *rely on* as a verb *rely* that requires a prepositional phrase with *on* as a complement, rather than as a single transitive phrasal verb.

# Reading Questions

- Why should SPR be a separate list from COMPS?
- Why use SPR for both subjects of verbs and specifiers of nouns?
- Why make SPR list valued?
- I'm wondering could the model in ch4 handle languages that allow multiple specifiers (model in ch3 certainly doesn't)? I'm thinking no, and would that be fixable if we want to make it possible to handle?



# Reading Questions

- I understand why we would consider words with variable/optional complements to be a single lexical entry (like *I ate* vs. *I ate the sandwich*); it is both more concise and intuitively feels more "correct" by native speaker intuitions. But for practical applications, e.g. in some sort of NLU program, would it be useful to segment these two options into separate lexical entries? Additionally, at what point of semantic distance, even in this framework, do we consider words with different valences to be lexically distinct (e.g. *I walked* vs. *I walked the dog* feels like it has a more significant semantic distance)?

# Reading Questions

- The change in this chapter of encoding the complements words can take not in the grammar rules but rather in the specific lexical entities makes me think that the set of acceptable sentences our eventual grammar will license will be quite different from the set that the average speaker would license. Speakers frequently use and successfully parse constructions that a by-the-dictionary evaluation wouldn't allow, and different speakers disagree with each other about acceptability, and dictionaries are constantly revised to keep up with changing usage anyway, etc. It seems like an impossible task to completely describe the set of complements each word can take. So my question is, are there ways to build a HPSG lexicon automatically, by looking at distributions in corpora, to get better coverage and maybe reflect evolving usage more easily?

# Reading Questions

- If *Alex* can be NP or N, is there ever any reason to use one in preference to the other?