

CAS LX 522

Syntax I

5

θ -roles, feature checking
(3.5-5.6)

Thematic relations

- The thematic relation that the argument has to the verb—the role it plays in the event—will prove useful in describing the behaviors of different classes of verb.
- One thematic relation is agent of an action, like *Bill* in:
1) Bill kicked the ball.

Common thematic relations

- Agent: initiator or doer in the event
- Theme/Patient: affected by the event, or undergoes the action
1) Sue kicked the ball.
- Experiencer: feel or perceive the event
3) Pat likes pizza.
- Proposition: a statement, can be true/false.
3) Bill said that he likes pizza.

Common thematic relations

- Goal:
1) Chris ran to Copley Square.
- Instrument:
4) Ed ate the burrito with a plastic spork.
- Benefactive:
5) Pat cooked dinner for Chris.
- Source:
3) Mary took a pencil from the pile.
- Location:
6) Betsy sits under the tree on Wednesdays.

Thematic relations

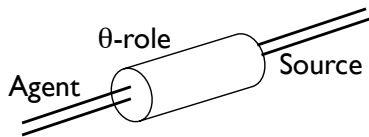
- Armed with these terms, we can describe the semantic connection between the verb and its arguments.
- Ray gave a grape to Bill.
 - Ray: Agent, Source, ...
 - A grape: Theme
 - Bill: Goal, Recipient, ...

Required vs. optional

- Things with certain thematic relations don't seem to be *needed* by a given verb, but can be there. E.g., location.
1) Pat screamed (in the library).
- Others, like theme/patient, goal, or agent, often do seem to be required. ("Required" means even if left out, there is something assumed)
2) Chris gave a book to Pat.

θ-roles

- An argument can participate in several thematic relations with the verb (e.g., Agent, Goal).
- In the syntax, we assign a special connection to the verb called a “θ-role”, which is a *collection* of thematic relations.
- For the purposes of syntax, the θ-role (the collection of relations) is much more central than the actual relations in the collection.



θ-roles

- We will often need to make reference to a particular θ-role, and we will often do this by referring to the most prominent relation in the collection.
- For example, in *Bill hit the ball*, we say that *Bill* has the “Agent θ-role”, meaning it has a θ-role containing the Agent relation, perhaps among others.

Unique θ Generalization

- Each θ-role must be assigned to a constituent, but a constituent cannot be assigned more than one θ-role.
- Historically, the “θ-criterion.”
- Verbs have a certain number of θ-roles to assign (e.g., *say* has two), and each of those must be assigned to a distinct argument.

Selection

- Verbs, as part of their meaning (that is, whatever is recorded in the lexicon), are often “selective” about what kinds of arguments, θ-roles they have.
- What verbs are said to do here is *select for* certain things.
- There are quite a number of things that verbs “care about.”

C(ategory)-selection (“subcategorization”)

- Verbs that take objects differ in what they allow the syntactic category those objects to be. Suppose *the ball* is category N (NP) and *that Bill left early* is category C (CP):
- 1) Sue saw/hit the ball.
 - 2) Sue saw/*hit that Bill left early.

Feelings

- The verb *feel* seems to have an Experiencer and a Theme/Source. But the Theme/Source can be any of several different syntactic categories. So: θ-role does not determine syntactic category; nor does syntactic category determine θ-role.
- 1) Pat felt a tremor.
 - 2) Pat felt uncomfortable.
 - 3) Pat felt that Chris had not performed well.

Kickings

- The verb *kick* seems to require a nominal (category N) argument.
- Verbs differ, so we need this to be recorded in the lexicon.
- *Kick* is a verb. It has a [V] feature.
- It “needs” a noun. Nouns have an [N] feature. But we need to distinguish between being and needing.

Interpretability

- The difference between “being” and “needing” will be referred to as a difference in *interpretability*.
- Being a verb, *kick* has an *interpretable* [V] feature.
- Needing a noun, *kick* has an *uninterpretable* [N] feature.
- The name gives a hint as to why the N is required. The uninterpretable [N] feature is dangerous. It must be gotten rid of. Otherwise, there will be something we can’t interpret.

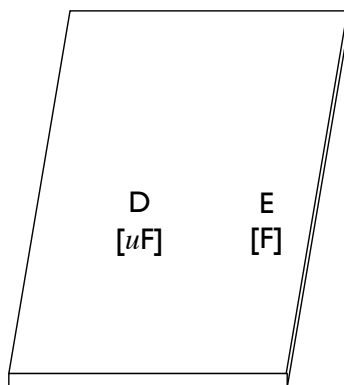
Feature checking

- For our model, we will say that if a syntactic object has an uninterpretable feature, it must Merge with a syntactic object that has a matching feature— and once it’s done, the requirement is met. The uninterpretable feature is checked.

Feature checking

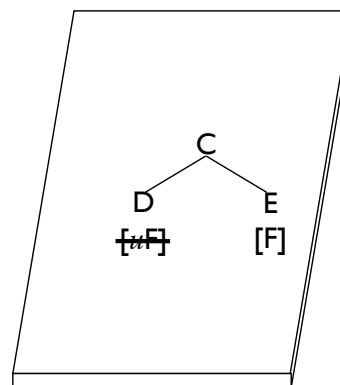
- Full Interpretation: The structure to which the semantic interface rules apply contains no uninterpretable features.
- Checking Requirement: Uninterpretable features must be checked (and once checked, they are deleted)
- Checking (under sisterhood): An uninterpretable feature F on a syntactic object Y is checked when Y is sister to another syntactic object Z which bears a matching feature F.

Feature checking



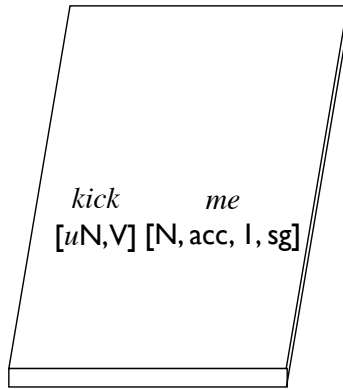
- To distinguish interpretable features from uninterpretable features, we will write uninterpretable features with a *U* in front of them.
- D has uninterpretable feature F
- E has interpretable feature F.
- If we Merge them, the uninterpretable feature can be checked (under sisterhood).

Feature checking



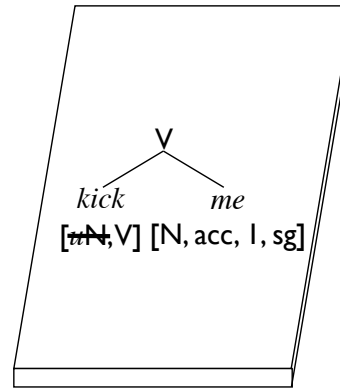
- To distinguish interpretable features from uninterpretable features, we will write uninterpretable features with a *U* in front of them.
- D has uninterpretable feature F
- E has interpretable feature F.
- If we Merge them, the uninterpretable feature can be checked (under sisterhood).

Feature checking



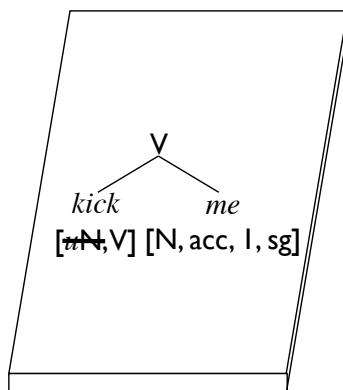
- Or, for a more concrete example
- *Kick* is a verb (has an interpretable V feature) and c-selects a noun (has an uninterpretable N feature).
- *me* is a noun (a pronoun in fact, has an interpretable N feature, and others like accusative case, first person, singular)

Feature checking



- Or, for a more concrete example
- *Kick* is a verb (has an interpretable V feature) and c-selects a noun (has an uninterpretable N feature).
- *me* is a noun (a pronoun in fact, has an interpretable N feature, and others like accusative case, first person, singular)

Feature checking



- The head is the “needy” one. The one that had the uninterpretable feature that was checked by Merge.
 - The combination has the features of the verb *kick* and so its distribution will be like a verb’s distribution would be.
- 1) Pat wants to kick me.
 - 2) Pat wants to drive.
 - 3) I like to draw elephants.
 - 4) *Pat wants to elephants.
 - 5) *I like to draw kick me.

Chris glanced at Pat

Pat [] Chris []
 at [] glanced []

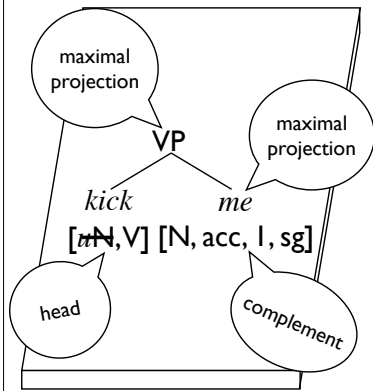
Syntactic operations

- **Merge** is a syntactic operation. It takes two syntactic objects and creates a new one out of them.
- The new syntactic object created by Merge inherits the features of one of the components (the head projects its features).
- Merge cannot “look inside” a syntactic object. Syntactic objects are only combined at the root.
- **The Extension Condition:** A syntactic derivation can only be continued by applying operations to the root projection of a tree.

Feature checking

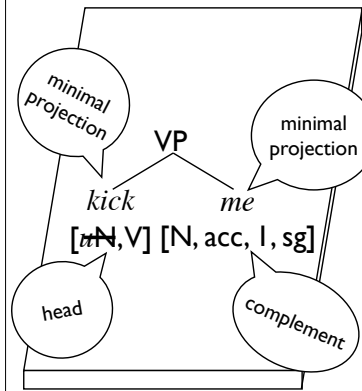
- Syntactic objects have features.
 - Lexical items (syntactic objects) are bundles of features.
- Some features are **interpretable**, others are **uninterpretable**.
- By the time the derivation is finished, there must be no uninterpretable features left (*Full Interpretation*).
- Uninterpretable features are eliminated by **checking** them against matching features. This happens as a result of Merge: Features of sisters can check against one another.
- Merge doesn’t just happen. It *has* to happen.

Heads and complements



- When Merge combines two syntactic objects, one projects its features, one does not.
- When a lexical item projects its features to the combined syntactic object, it is generally called the **head**, and the thing it combined with is generally called the **complement**.
- A syntactic object that projects no further is called a **maximal projection**.
 - Where X is the category, this is alternatively called X^{max} or XP.
 - The complement is necessarily a maximal projection.

Heads and complements



- A syntactic object that has not projected at all (that is, a lexical item) is sometimes called a **minimal projection**.
 - Where X is the category, this is alternatively called X^{min} or X.
 - The head is a minimal projection.
 - In traditional terminology, the complement of a verb is generally called the **object** (or "**direct object**").
 - So, often, is the complement of a preposition ("object of the preposition").

Linear order

- Merge takes two syntactic objects and combines them into a new syntactic object.
- Merge does not specify *linear order* (which of the two combined objects comes first in pronunciation).
- In the English VP, heads always precede complements. But languages differ on this.

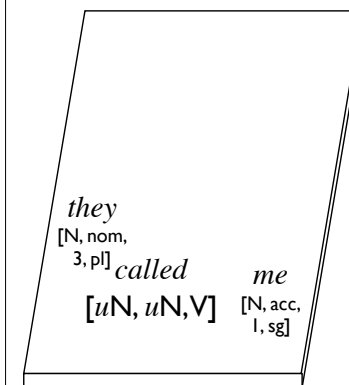
The head parameter

- Languages generally have something like a *basic word order*, an order in which words come in in "neutral" sentences.
- English: SVO
 - Akira ate an apple.
- Japanese: SOV
 - John wa ringo o tabeta.
John top apple acc ate
'John ate an apple.'
- In our terms, this amounts to a (generally language-wide choice) as to whether heads are pronounced before complements or vice-versa.
 - English: **head-initial** Japanese: **head-final**

Second Merge

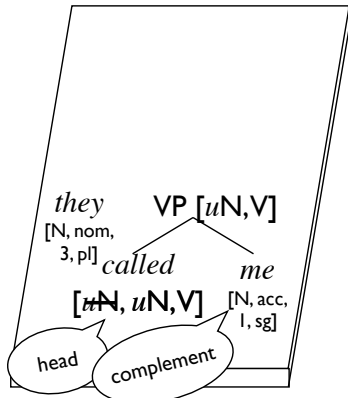
- Merge occurs when there is a selectional feature that needs to be satisfied.
 - If there is more than one such feature, Merge must happen more than once.
- As always, the node that projects is the one whose selectional feature was satisfied by the Merge.
 - The sister of the head (that projects) after the first Merge involving that head is called the **complement** (as above).
 - The nonprojecting sister of a syntactic object that has already projected once from a head is called the **specifier**.

Heads and complements



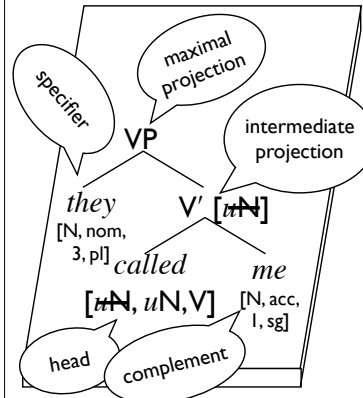
- A transitive verb like *called* needs two arguments (the caller and the callee).
- We encode this knowledge by hypothesizing two selectional features for N.
 - The first selectional feature will be checked by the callee.
 - The second selectional feature will be checked by the caller.
- So, *called* is Merged with *me*.

Heads and complements



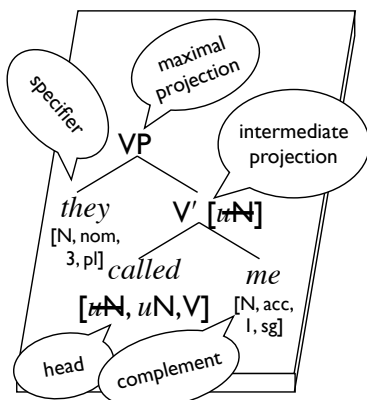
- So, *called* is Merged with *me*.
- One of the selectional features is checked off, the remaining features project to the new object.
- A selectional feature still remains.
- Merge applies again, Merging the new object with *they*.

Specifiers, XP, X-bar



- The second selectional feature has been eliminated.
- The sister to this second Merge is the **specifier**.
- A node that does not project further is a **maximal projection**.
- A node that has been projected and projects further is neither maximal nor minimal and is usually called an **intermediate projection**.

Specifiers, etc.

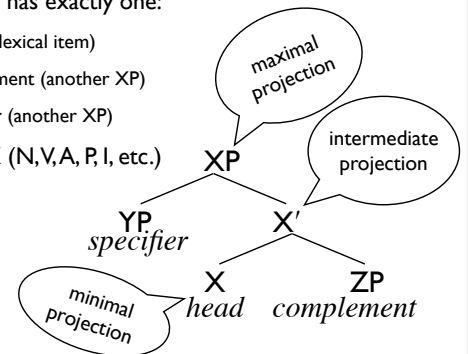


- In English, specifiers are on the left of the head, unlike complements.
- As with the head-complement order, languages (arguably) also differ in the linear order of their specifiers.
- However, Spec-initial order is overwhelmingly more common...
- VOS order (Malagasy)
Nahita ny mpianatra ny vehivavy.
saw the student the woman
'The woman saw the student.'

- In the '70s and '80s, these ideas went by the name "X'-theory":

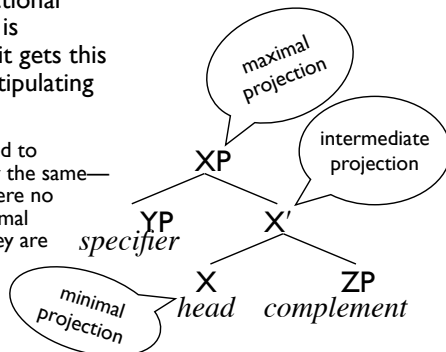
X'-theory

- Every XP has exactly one:
 - head (a lexical item)
 - complement (another XP)
 - specifier (another XP)
- for any X (N, V, A, P, I, etc.)



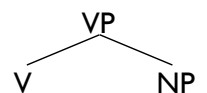
Merge v. X'-theory

- The system of selectional features and Merge is preferable because it gets this structure without stipulating the template.
- The structure assigned to sentences is generally the same—except that for us, there no intermediate or maximal projections unless they are needed.



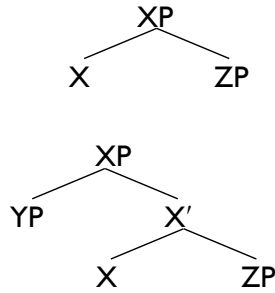
Node labeling conventions

- When we Merge two objects, the features of one of them projects to become the features of the new object.
- The label for new node comes in two pieces:
 - The category (projected from the head)
 - The projection "level":
 - P = maximal projection
 - ° or nothing = minimal projection
 - ' = intermediate projection
- An XP is any node that does not project its features up.
- An X° (or X) node comes from the lexicon.

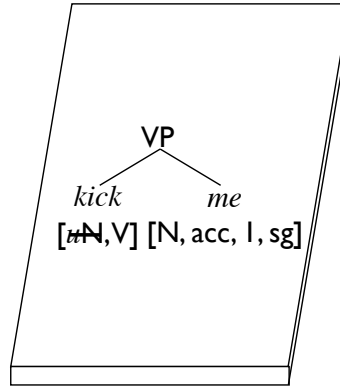


Maximal v. Minimal v. Intermediate

- Notice that whenever you Merge two things, the result is going to be a maximal projection. An “XP”.
- But if in the next step if projects when you Merge it with something, that same node is now an intermediate projection.



Features and checking



- When we combine two things with Merge and check an uninterpretable feature, we cross it out.
- For simplicity, we can simply write the features under the head, and cross them out there.
- This is as opposed to copying all but the checked feature and into a feature specification of the VP node.
- This is just about how we write it down, it is the same system either way.

Adjuncts

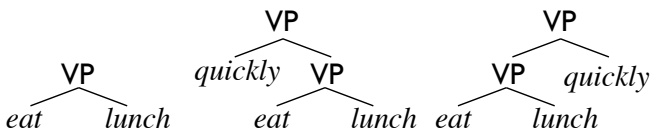
- *Pat put the book.
- Pat put the book on the shelf.
- Pat put the book on the shelf dramatically.
- Pat put the book on the shelf dramatically on Tuesday.
- Pat put the book on the shelf dramatically on Tuesday before several witnesses.
- Some things are required. Some things are not.
 - **Arguments** get θ -roles and are **required**.
 - **Adjuncts** are modificational and are **optional**.

Adjuncts and distribution

- Adjuncts are relatively “transparent”— having an adjunct does not seem to change the distributional characteristics.
 - Pat wants to eat lunch (quickly).
 - Pat wants to dine.
 - *I like to draw eat lunch (quickly).
 - I like to draw (happy) elephants.
 - *Pat wants to (happy) elephants.
- Idea: A verb (phrase) with an adjunct is still a verb (phrase), just as if it didn’t have an adjunct.

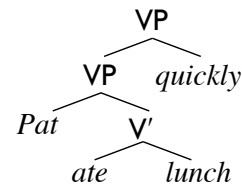
Adjoin

- The operations Merge and Adjoin are two different ways to combine two objects from the workbench.
- Merge takes two objects and creates a new object (with the label/features inherited from one of them).
- Adjoin attaches one object to the top of another one.
 - The linear order of adjuncts does not appear to be set parametrically, so they can either before or after the object they attach to.



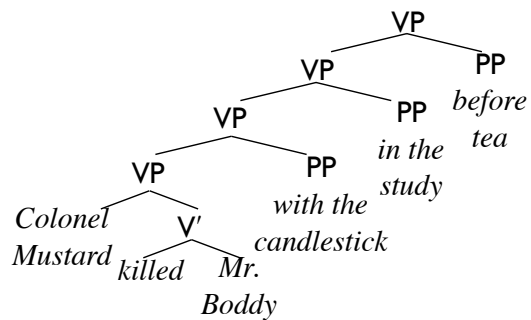
The luxury of adjunction

- We will also assume that Adjoin only applies to maximal projections.
- That is: If a syntactic object still has a selectional feature, Adjoin cannot attach something to it. Merge must happen first. Once all of the things that *need* to happen are taken care of, *then* you have the luxury of adjunction.



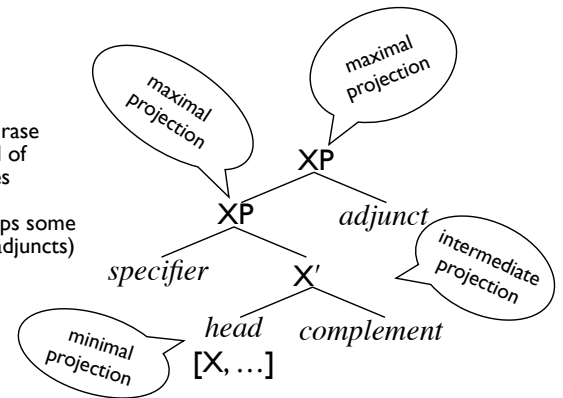
The luxury of adjunction

- Any number of adjuncts can be added, generally in any order. Adjuncts come in many different categories—“adjunct” is not a category, but rather a structural description.



A phrase

- So, a full phrase can have all of these pieces
(plus perhaps some additional adjuncts)



Complements vs. adjuncts

- PPs seem to be freely reorderable— when adjuncts.
 - I ate lunch on Tuesday at Subway with Pat
 - I ate lunch on Tuesday with Pat at Subway
 - I ate lunch with Pat on Tuesday at Subway
 - I ate lunch on Tuesday with Pat at Subway
- But consider *glance at Chris*.
 - I glanced at Chris on Tuesday
 - *I glanced on Tuesday at Chris
- Ok: Why?