Lecture 28: Theory Overview.

Andrei Antonenko

LIN 311: Syntax

December 6, 2018

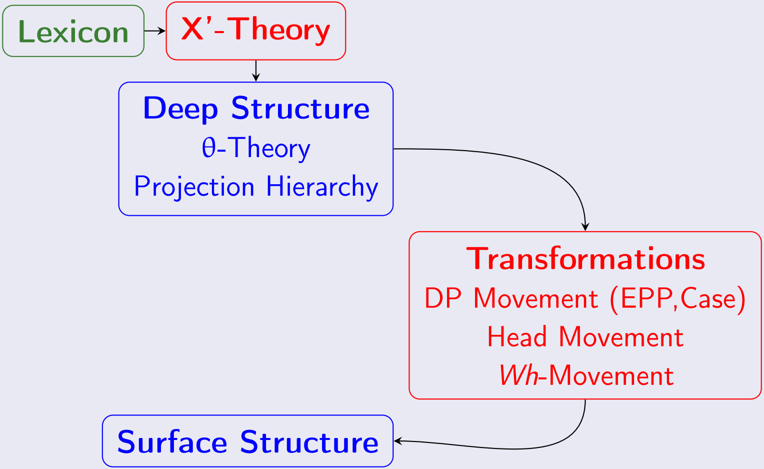
Outline

1. Structure Building
   - Classical View
   - Minimalist Program
2. Covert Movement
   - Scope Ambiguity
   - T-Model of Grammar
3. Ellipsis
   - Ellipsis Resolution
4. Summary

Structure Building

Old view of structure building

Old view of structure building



Old view of structure building

- **Lexicon**: Lexical items (words) are pulled out from the lexicon.
- **X'-Theory**: X'-Theory defines a template on how to put these words together. Each word must be a head of some phrase: VP, DP, NP, PP, etc.
- **Deep Structure**: X'-Bar structures should conform to θ-theory: each verb should have a proper number of arguments, all projections should be in the right order, etc.
- **Transformations**: Elements need to move to get Case, satisfy EPP principle, *wh*-phrases must move to the Spec,CP positions, etc.
- **Surface Structure**: The result of these transformations gives us a surface structure, i.e. the sentence with the correct order and all features checked.

Minimalist Program

Main syntactic operations

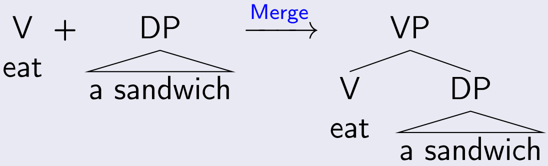
The modern view on syntactic structures is formulated within the framework called Minimalism Program (Chomsky 1995).
This framework relies on three major operations:

- **Merge**: Put stuff together.
- **Agree**: Relate some elements in the structure.
- **Move**: Move some elements to different positions in the structure.

Minimalist Program

Merge

- The structure is built bottom up, and we use operation **Merge** to put things together.
- **Merge** is an operation that takes two elements, combines them, and assigns a label to the resulting structure.



Minimalist Program

Agree

- Each lexical item (word) contains a number of features.
- Features can be interpretable/uninterpretable; and valued/unvalued.
- Unvalued features are probes: they need to find a value by probing other element (Goal) with a matching valued feature: **Agree**.

Minimalist Program

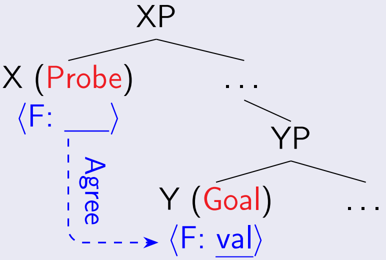
Move

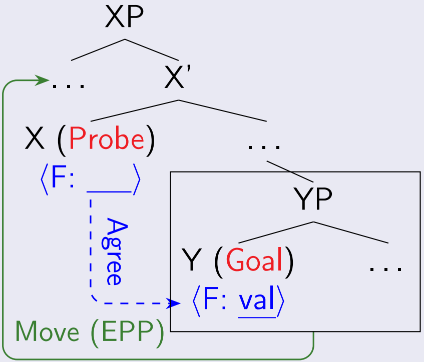
- If the probe has an EPP-feature, the goal may undergo **Movement** close to the probe.
  - **Head-Movement**: only a head moves — V-to-T, T-to-C.
  - **Phrase Movement**: entire phrase moves — *wh*-movement, subject raising from Spec,VP to Spec,TP.
  - Note that on this view we don't need empty Specifier positions. Specifiers are built only when we need them.
- Alternatively, EPP can be satisfied by another **Merge** operation: put something in the specifier (for example, an expletive).

Minimalist Program

Agree and Move

- **Agree**: Probing process triggered by the need to value a feature.
- **Move**: Agree may trigger movement, if a head has EPP.





Minimalist Program

Example of Agree - 1

- **T-Subject**: T agrees with the subject (subject-verb agreement), so it means that T must probe the subject. The feature which triggers that is a φ-feature (person/gender/number). If T has an EPP feature, the subject raises to Spec,TP.

(1) a. The cat is in the garden.
b. The cats are in the garden

- It is also possible that this agreement is not triggering movement, and the EPP is satisfied by inserting an expletive (*there*) to the Spec,TP.

(2) a. There is a cat in the garden.
b. There are cats in the garden.

Minimalist Program

Example of Agree - 2

- **C-*Wh***: C$_{[+\text{wh}]}$ probes to find a phrase with a *wh*-feature and agree with it. In English, *wh*-phrase then undergoes movement to Spec,CP (EPP on C), but in Japanese and Chinese they don't (lack of EPP on C).
- In English, we don't see agreement like in the case of Subject-Verb agreement. But in some languages (Kinande, Bantu) we do:

(3) a. IyondI yO Kambale alangIra.
who.1 that.1 Kambale saw

b. EkIhI kyO Kambale alangIra.
what.7 that.7 Kambale saw

A- vs. A'-movement

There are two possible types of phrasal movement:

1. **A-movement**: Movement to an A-position (argument position), for a fixed grammatical function.
   - Movement of a subject from Spec,VP to Spec,TP;
   - Movement of object to subject position in passives;
   - etc.
2. **A'-movement**: Movement to an A'-position, a position where a fixed grammatical function is not assigned.
   - *wh*-movement;
   - Movement to Spec,CP in German V2-constructions;
   - etc.

This distinction is not always clear. Some types of movement have properties of both A and A'-movements.

Covert Movement

Scope ambiguity

Ambiguity

(4) Everyone ate a pie.

The sentence (4) is ambiguous:

- **Meaning 1:** Each person ate their own pie.
- **Meaning 2:** There was a single pie that everyone had a piece of.

Scope ambiguity

Quantifiers

(5) Everyone ate a pie.

The difference in meaning can be expressed using the formal logic apparatus — quantifiers:

- **Universal Quantifier** ∀: *every*, *all*, *each*.
- **Existential Quantifier** ∃: *some*, *a*, *one*.

Scope ambiguity

(6) Everyone ate a pie.

Two different meanings of (6):

- **Meaning 1:** Each person ate their own pie.
  For every person x, there is some pie y, such that x ate y
  (Universal quantifier has "wide scope"):

  $$\forall x, x \text{ is a person}, (\exists y, y \text{ is a pie}, [x \text{ ate } y])$$

- **Meaning 2:** There was some pie that everyone had a piece of.
  For some (fixed) pie y, and every person x, x ate y
  (Universal quantifier has "narrow scope").

  $$\exists y, y \text{ is a pie}, (\forall x, x \text{ is a person}, [x \text{ ate } y])$$

Scope ambiguity

(7) Everyone ate a pie.

How can this ambiguity be accounted for?

- Every instance of ambiguity must be reflected in the structure!
- How do we put *some* before *every*?

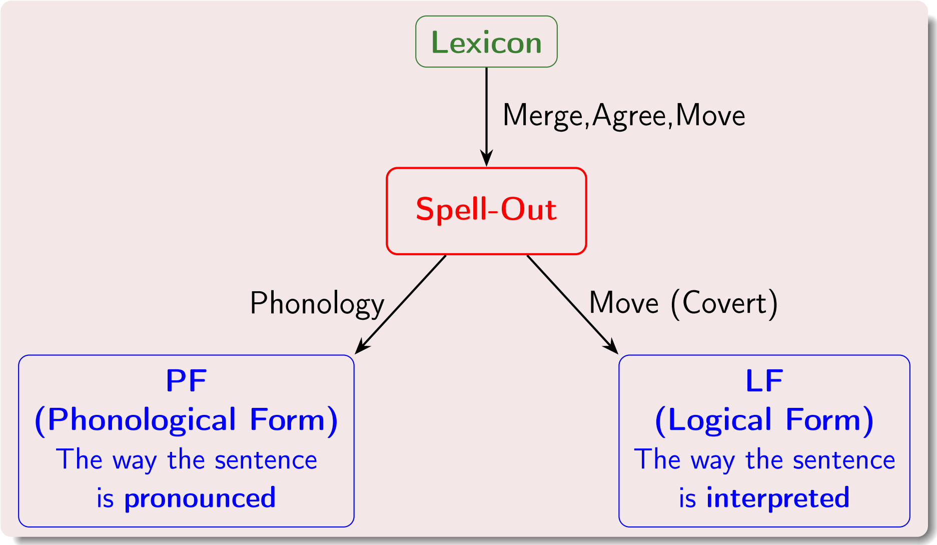
Covert Movement

We can use **Movement**:

(8) A pie everyone ate ___.

- But this gives us a wrong word order (not to mention a potentially ungrammatical sentence)!
- **Solution:** A type of movement that we cannot hear: **Covert Movement**.

T-Model of grammar



Lexicon

Merge,Agree,Move

Spell-Out

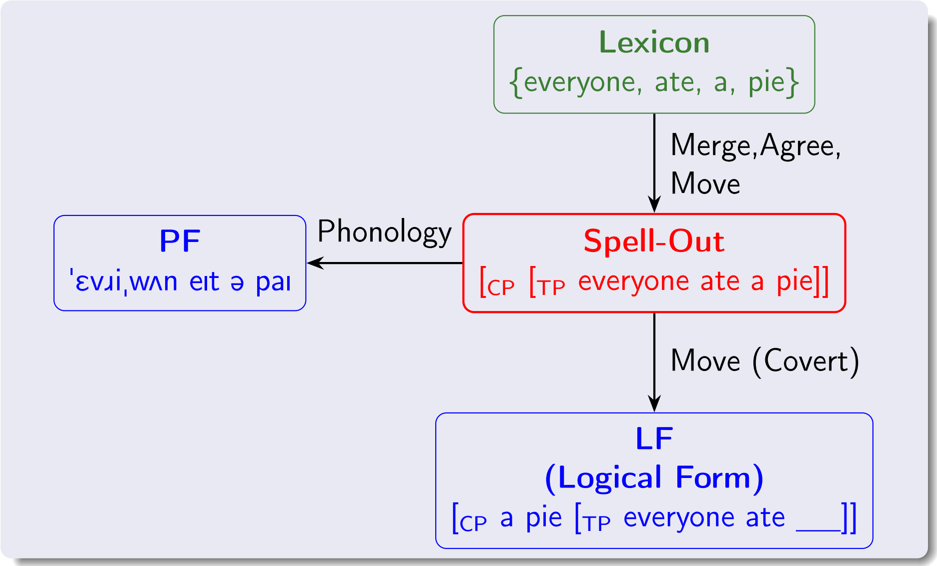
Phonology

Move (Covert)

PF
(Phonological Form)
The way the sentence
is **pronounced**

LF
(Logical Form)
The way the sentence
is **interpreted**

Example 1: Scope in English



Lexicon
{everyone, ate, a, pie}

Merge,Agree,
Move

Spell-Out
[$_{CP}$ [$_{TP}$ everyone ate a pie]]

Phonology

PF
ˈɛvɹiˌwʌn eɪt ə paɪ

Move (Covert)

LF
(Logical Form)
[$_{CP}$ a pie [$_{TP}$ everyone ate ___]]

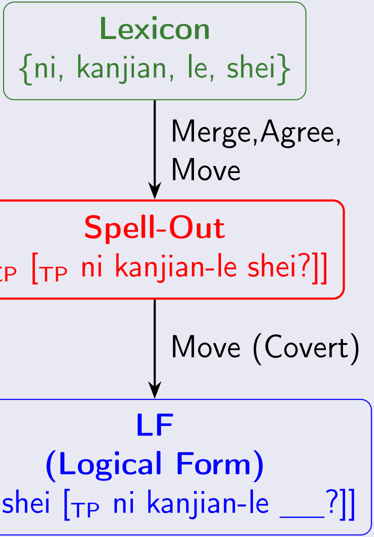
Example 2: Chinese *wh*-in-situ

***Wh*-in-situ in Chinese:**

(9) a. Ni kanjian-le shei?
you saw who
"Who did you see?"

b. *Shei ni kanjian-le ___?
who you saw
"Who did you see?"

- *Wh*-movement exists in Chinese, but unlike in English, it is covert.



Lexicon
{ni, kanjian, le, shei}

Merge,Agree, Move

Spell-Out
$[_{\text{CP}}$ $[_{\text{TP}}$ ni kanjian-le shei?]]

Move (Covert)

LF
(Logical Form)
$[_{\text{CP}}$ shei $[_{\text{TP}}$ ni kanjian-le ___?]]

Crosslinguistic Variation

Crosslinguistically it is possible to say that movements are the same in all languages, but in some languages they are overt, and in some languages they are covert:

- **V-to-T:** French: overt, English: covert
- ***wh*-movement:** English:overt, Chinese: covert
- **Subject movement to Spec,TP:** English: overt, Irish: covert
- ***wh*-movement of the 2nd and further *wh*-phrases:** Slavic: overt, English: covert

Ellipsis

Ellipsis

Ellipsis: Deletion of a part of the sentence.

- **Simple VP ellipsis:**

  (10) John will [eat a sandwich] but Sally won't ~~[eat a sandwich]~~.

- **Pseudogapping:**

  (11) Megan has eaten more ice-cream than Bill has ~~[eaten]~~ octopus.

- **Sluicing (TP ellipsis):**

  (12) $[_{\text{CP}}$ $[_{\text{TP}}$ John could bake something$]_{i}$] but I'm not sure $[_{\text{CP}}$ what ~~$[_{\text{TP}}$ John could bake$]_{i}$~~]

Other types: Gapping, N-ellipsis, Stripping, Comparative deletion, Antecedent Contained Deletion.

Ellipsis

(13) Frank will $[_{VP}$ eat an apple$]_i$ and Morgan will $[_{VP}\ \varnothing]_i$ too.

Question: How are such sentences derived and interpreted?

- There must be something happening: we need to resolve what $\varnothing$ stand for.
- PF (Phonological form, the way the sentence is pronounced) doesn't tell us that.

Two options:

1. **LF Copying**: Second VP $[_{VP}\ \varnothing]_i$ was empty from the beginning, but got filled up with the content at LF.
2. **PF Deletion**: Second VP $[_{VP}\ \varnothing]_i$ was originally a full VP, but got deleted at PF.

Ellipsis resolution: LF copying

LF Copying

(14) Frank will $[_{VP}$ eat an apple$]_i$ and Morgan will $[_{VP}$ $\varnothing]_i$ too.

1. **LF Copying**: Ellipsis is really a null verb without any structure. Its meaning gets resolved the same way anaphors resolve their meanings: $\varnothing$ is changed to a full verb phrase at LF.

(15) a. **Spellout**: Sally likes herself. ⇒
b. **LF**: Sally likes Sally.

(16) a. **Spellout**: Frank will $[_{VP}$ eat an apple$]_i$ and Morgan will $[_{VP}$ $\varnothing]_i$ too. ⇒
b. **LF**: Frank will $[_{VP}$ eat an apple$]_i$ and Morgan will $[_{VP}$ eat an apple$]_i$ too.

Ellipsis resolution: PF deletion

PF Deletion

(17) Frank will $[_{\text{VP}}$ eat an apple$]_i$ and Morgan will $[_{\text{VP}}\ \varnothing]_i$ too.

2. **PF Deletion**: The entire phrase $[_{\text{VP}}$ eat an apple$]$ was there from the very beginning, but got deleted after Spell Out, at PF *(don't want to pronounce the same thing twice)*.

(18) a. **Spellout**: Frank will $[_{\text{VP}}$ eat an apple$]_i$ and Morgan will $[_{\text{VP}}$ eat an apple$]_i$ too. ⇒

b. **PF**: Frank will $[_{\text{VP}}$ eat an apple$]_i$ and Morgan will $[_{\text{VP}}\ \varnothing]_i$ too.

An argument for LF Copying

(19) Sally will [$_{VP}$ praise herself] and Alice will [$_{VP}$ ∅] too.

Two possible readings:

Strict: Sally praises Sally; Alice praises Sally.

Sloppy: Sally praises Sally; Alice praises Alice.

What does PF Deletion predict?

(20) a. **Spellout:** [Sally will [$_{VP}$ praise herself]] and [Alice will [$_{VP}$ praise herself]] too.

b. **PF:** [Sally will [$_{VP}$ praise herself]] and [Alice will [$_{VP}$ ∅]] too.

c. **LF:** [Sally will [$_{VP}$ praise Sally]] and [Alice will [$_{VP}$ praise Alice]] too.

- **PF Deletion predicts that only sloppy reading is possible!**

An argument for LF Copying

LF Copying on the other hand has no problem with both readings:

(21) a. **Spellout:** [Sally will [$_{VP}$ praise herself]] and [Alice will [$_{VP}$ $\varnothing$]] too.

b. **LF-1 (Resolve *herself*):** [Sally will [$_{VP}$ praise Sally]] and [Alice will [$_{VP}$ $\varnothing$]] too.

c. **LF-2 (Copy VP):** [Sally will [$_{VP}$ praise Sally]] and [Alice will [$_{VP}$ praise Sally]] too. **(Strict)**

(22) a. **Spellout:** [Sally will [$_{VP}$ praise herself]] and [Alice will [$_{VP}$ $\varnothing$]] too.

b. **LF-2 (Copy VP):** [Sally will [$_{VP}$ praise herself]] and [Alice will [$_{VP}$ praise herself]] too.

c. **LF-1 (Resolve *herself*):** [Sally will [$_{VP}$ praise Sally]] and [Alice will [$_{VP}$ praise Alice]] too. **(Sloppy)**

An argument for PF Deletion

(23) a. *Which language do you want to hire $[_{DP}$ someone who speaks ___]?

b. *They want to hire someone who speaks a Balkan language, but I don't know which language they do ~~$[_{VP}$ want to hire someone who speaks ___]~~.

- Example (23-a) is a violation of Complex NP Constraint.
- If (23-b) starts out with $[_{VP}\ \varnothing]$ instead of a full VP, as LF Copying hypothesis predicts, then:
  - It is unclear where the *wh*-phrase is coming from;
  - It is unclear why this sentence is a violation of an island constraint (if there is only $\varnothing$, there should not be any islands!).

Ellipsis resolution

- There are good arguments both
  1. in favor of PF Deletion; and
  2. in favor of LF Copying.
- We will not be able to resolve this debate now.
- Syntacticians are still working on questions of this sort!

Summary

Summary

Overt vs. covert movements

- There are some movements which we can observe; they happen before Spellout, and are visible at PF (Phonological Form).
- There must be some movements which are not reflected in pronunciation of the sentence. They are necessary for semantic reasons, for recovering the meaning after ellipsis, etc. Such movements are called covert, and they happen after Spellout on the way to LF.
- LF (Logical Form) and PF (Phonological Form) split is necessary to be able to account for a lot of facts: scope, ellipsis, anaphors, etc.
- Figuring out properties of covert movement is not so easy: we need to know a fair bit of semantics, and some more syntax.