Quantifier Raising and Scope Ambiguity

1 Transformations in grammar

• The Government and Binding theory (Chomsky 1973) involves four levels of representation:

```
  DS
   ↓
  SS
```

- **Deep Structure (DS):** Where the derivation begins.
- **Surface Structure (SS):** Where the word order matches what we see (after e.g. *wh*-movement).
- **Phonological Form (PF):** Where the words are realized as sounds (after e.g. ellipsis).
- **Logical Form (LF):** The input to semantic interpretation (after e.g. quantifier raising).

• Overt movement takes place at the transformation from DS to SS:

  (1) **Wh-movement**
  a. **Deep structure**
     
     I know [Mary saw whom].
  b. **Surface structure**
     
     I know [whom, Mary saw t_i].

  (2) **Subject-auxiliary inversion**
  a. **Deep structure**
     
     you did come?
  b. **Surface structure**
     
     Did, you t_i come?

  **Discussion:** What movements are involved in the formation of the following matrix question?

  (3) Whom did Andy invite?

• Ellipsis takes place from SS to PF:

  (4) a. Andy invited Mary, but Jenny didn’t invite Mary.
       **Surface structure**
  b. Andy invited Mary, but Jenny didn’t invite Mary.
       **Phonological form**

  **Discussion:** What ellipsis is involved in the following sentence?

  (5) Andy invited Mary, but not Jenny.

• For the rest of today: what operations are involved from the transformation from SS to LF?
2 Quantifier raising

2.1 Generalized quantifiers and quantificational determiners

- We treat quantificational DPs as second-order functions of type $\langle et, t \rangle$, called generalized quantifiers. In (6), *meows* is an argument of *every cat*.

\[
\text{(6)} \quad S_t \quad \Downarrow
\]
\[
\text{DP}_\langle et, t \rangle \quad \text{VP}_\langle et, t \rangle
\]
\[
\text{D} \quad \text{NP} \quad \text{meows}_\langle et, t \rangle
\]
\[
\text{every} \quad \text{cat}_\langle e, t \rangle
\]

\[
\text{(7)} \quad \begin{align*}
\text{a. } [\text{every cat}] &= \lambda P_{\langle e, t \rangle}, \forall x[\text{cat}(x) \rightarrow P(x)] \\
\text{b. } [\text{every cat meows}] &= [\text{every cat}](\text{[meows]}) \\
&= (\lambda P_{\langle e, t \rangle}, \forall x[\text{cat}(x) \rightarrow P(x)])(\lambda y_e. \text{meows}(y)) \\
&= \forall x[\text{cat}(x) \rightarrow \text{meows}(x)]
\end{align*}
\]

- The determiner *every* combines with a common noun of type $\langle e, t \rangle$ to return a generalized quantifier of type $\langle et, \langle et, t \rangle \rangle$. Therefore, its type is quite complex: $\langle et, \langle et, t \rangle \rangle$.

\[
\text{(8)} \quad \begin{align*}
\text{a. } [\text{every}] &= \lambda Q_{\langle e, t \rangle}, \lambda P_{\langle e, t \rangle}, \forall x[Q(x) \rightarrow P(x)] \\
\text{b. } [\text{some}] &= \lambda Q_{\langle e, t \rangle}, \lambda P_{\langle e, t \rangle}, \exists x[Q(x) \land P(x)] \\
\text{c. } [\text{no}] &= \lambda Q_{\langle e, t \rangle}, \lambda P_{\langle e, t \rangle}, \neg \exists x[Q(x) \land P(x)]
\end{align*}
\]

- A quantificational determiner takes two arguments (both of which are of type $\langle e, t \rangle$). The first argument is its restrictor, and the second argument is its scope.

\[
\text{(9)} \quad S \quad \Downarrow
\]
\[
\text{DP} \quad \text{VP}
\]
\[
\text{D} \quad \text{NP} \quad \text{meows}_\langle e, t \rangle
\]
\[
\text{every} \quad \text{cat}_\langle e, t \rangle
\]

**Discussion:** Identify the restrictor and scope of *every* in the following sentences.

- (10)  
  a. Every student who read chapter 5 passed the exam.
  b. Everyone passed the exam.
  c. John read every chapter.
2.2 Quantifier raising

- A type-mismatch arises when a generalized quantifier appears at a non-subject position.

\[(11) \quad \text{Andy loves every cat.}\]

- This problem can be resolved by a covert movement of the generalized quantifier, called **Quantifier Raising** (QR). This movement takes place at the transition from SS to LF.

\[(12) \quad \text{The generalized quantifier every cat is moved to the left edge of the sentence, leaving a trace.}\]

In semantics, we interpret this trace as a variable of a matching type, namely \(x_e\), and then abstract over this variable by inserting \(\lambda x_e\) immediately below *every cat*. This abstraction operation is called **Predicate Abstraction**.

- **Exercise**: Compose the meaning of (12) and then identify the restrictor and scope of *every*.
• **Exercise:** Draw a tree to illustrate the LF of the following sentence.

(13) We require no student to come to office hour.

2.3 **Scope ambiguity**

• A generalized quantifier can be raised to adjoin to any sentential node. Which sentential node it adjoins to determines its semantic scope.

(14) Andy didn’t read a book.

– *a book > not:* ‘There is a book *x* such that Andy didn’t read *x*’

\[
\begin{align*}
S & \rightarrow DP \cdot \lambda x \{ e \} \cdot S \\
& \quad \rightarrow \text{not} \cdot S \\
& \quad \rightarrow DP \cdot VP \\
& \quad \rightarrow \text{Andy} \cdot V_{tr} \cdot x_e
\end{align*}
\]

– *not > a book:* ‘There isn’t any book *x* such that Andy read *x*’

\[
\begin{align*}
S & \rightarrow \text{not} \cdot S \\
& \quad \rightarrow DP \cdot \lambda x \{ e \} \cdot S \\
& \quad \rightarrow DP \cdot VP \\
& \quad \rightarrow \text{Andy} \cdot V_{tr} \cdot x_e
\end{align*}
\]
2.4 (Other) phrasal movement

- In generative grammar, phrasal movement (overt or covert) is uniformly formalized as follows:

  (15) **Movement of a phrase α from position A to position B:**
  a. α is moved to position B;
  b. α in A is replaced with a trace, interpreted as a variable;
  c. we abstract over this trace variable by inserting a matching lambda node immediately below B, forming a λ-abstract.

  The sister node of the λ-abstract is the moved phrase. The variable bound by the λ-operator is the trace.

Example: using predicate abstraction to represent subject movement:

(16) Mary will leave.

```
TP
  Mary
    λx
  T'
    T₀
      will
      x
      V
      leave
```

Example: using predicate abstraction to represent *wh*-movement


```
CP
  ANS
    λp
  whoₑᵗₑᵗ
    λxₑ
      ID
      p
      IP
      xₑ left
```

What elements are moved in the above structure?

- **If a generalized quantifier occurs at a subject position, does it move at all?**
  The modern syntactic theory says “Yes”. But this movement is not driven by type-mismatch, but by syntactic reasons. It occurs at the transformation from DS to SS.

(18) Mary left.

```
S
  Mary
    λx
  VP
    x left
```

(19) Everyone left.

```
S
  everyone
    λx
  VP
    x left
```
Discussion: The following (simplified) structures are problematic. Identify the problems:

(20)  a. Andy read every chapter.

\[
\text{DP} \quad \text{every chapter} \quad \lambda x \quad \text{read} \quad x
\]

b. Every boy met some girl.

\[
\text{DP} \quad \text{every boy} \quad \lambda y \quad \text{DP} \quad \lambda x \quad \text{VP} \quad x \quad \text{met} \quad y
\]

c. Which book did every student buy?
   (Intended reading: ‘For every student x, which book did x buy?’)

\[
\text{DP} \quad \text{every kid} \quad \lambda x \quad \text{CP}_{(st,t)} \quad \text{which book did } x \quad \text{buy}
\]