Structural Relations

The mathematical properties of phrase structure trees
Important!
Important!

Even if you have trouble with the formal definitions, try to understand the INTUITIVE idea behind them. Don’t get lost in the details of the formalism.
Structural Relations

**Structural relations**: the formal relationships between items of a tree

**Why should we care?** We want to be able to talk about specific relationships in terms of structures.

**Structural relations are actually very simple!**

Don’t let the formalism scare you!
Some basic terms
Some basic terms

Branches

Diagram:

- M
  - N
    - D
    - E
    - F
  - O
    - G
    - H
    - J
Some basic terms

Branches

Labels: M,N,O,D,E,F,G,H,J
Some basic terms

Branches

Labels: M, N, O, D, E, F, G, H, J

Node: Any point with a label
Some basic terms

Labels: M, N, O, D, E, F, G, H, J
Node: Any point with a label
Some basic terms

- **Labels**: M, N, O, D, E, F, G, H, J
- **Node**: Any point with a label
- **Branches**: Non-terminal nodes
- **Root node**
Some basic terms

Labels: M, N, O, D, E, F, G, H, J

Node: Any point with a label

Branches

Root node

Non-terminal nodes

Terminal nodes
Domination
Domination

Intuitively: this is containment. If a node contains another, then it dominates it:
Domination

Intuitively: this is containment. If a node contains another, then it dominates it:

```
  A
 /|
B  C
 /|
D  E F G
```
Domination

Intuitively: this is *containment*. If a node contains another, then it dominates it:

A dominates B, C, D, E, F, G

D dominates E, F, G
Domination

Intuitively: this is containment. If a node contains another, then it dominates it:

A dominates B,C,D,E,F,G

D dominates E,F,G
Domination

Intuitively: this is containment. If a node contains another, then it dominates it:

A dominates B, C, D, E, F, G

D dominates E, F, G

Another way to think of it: “on top of”
Domination
A slightly more formal definition:

**Domination**: Node A dominates node B if and only if A is higher up in the tree than B and if you can trace a line from A to B going only downwards.
Immediate Domination

Node A **immediately dominates** node B if there is no intervening node G which is dominated by A, but dominates B. (in other words, A is the first node that dominates B)
Immediate Domination

Node A immediately dominates node B if there is no intervening node G which is dominated by A, but dominates B. (in other words, A is the first node that dominates B)
Immediate Domination

Node A immediately dominates node B if there is no intervening node G which is dominated by A, but dominates B. (in other words, A is the first node that dominates B)

A dominates B, C, D, E, F, G

but A immediately dominates only B, C, D
Exhaustive Domination

Node A exhaustively dominates a **SET** of **TERMINAL** nodes \{B, C, ..., D\},

- provided it dominates all the members of the set (so that there is no member of the set that is not dominated by A)
- AND there is no terminal node G dominated by A that is not a member of the set.
Exhaustive Domination
Exhaustive Domination

A exhaustively dominates the set \{B, C, D, E\}
Exhaustive Domination

A exhaustively dominates the set \{B,C,D,E\}

A does **NOT** exhaustively dominate the set \{B,C,D\}
Exhaustive Domination

A exhaustively dominates the set \{B,C,D,E\}
A does NOT exhaustively dominate the set \{B,C,D\}
A does NOT exhaustively dominate the set \{B,C,D,E,H\}
A formal definition of constituency
Constituent: The set of nodes exhaustively dominated by a single node
A formal definition of constituency

**Constituent**: The set of nodes exhaustively dominated by a single node
A formal definition of constituency

**Constituent:** The set of nodes exhaustively dominated by a single node
**A formal definition of constituency**

**Constituent:** The set of nodes exhaustively dominated by a single node
A formal definition of constituency

**Constituent**: The set of nodes exhaustively dominated by a single node.
A formal definition of constituency

**Constituent**: The set of nodes exhaustively dominated by a single node

\{E, H\} are NOT a constituent
Constituent vs Constituent of
Constituent vs Constituent of

Constituent of does NOT mean the same thing as constituent.
Constituent vs Constituent of

Constituent of does NOT mean the same thing as constituent.

Essentially ‘constituent of’ is the opposite of domination.
Constituent vs Constituent of

Constituent of does NOT mean the same thing as constituent.

Essentially ‘constituent of’ is the opposite of domination.

A dominates B, then we say B is a constituent of A.
Constituent vs Constituent of

Constituent of does NOT mean the same thing as constituent.

Essentially ‘constituent of’ is the opposite of domination.

A dominates B, then we say B is a constituent of A.

Immediate constituent of is the opposite of immediate domination.
Some Informal Terms
Some Informal Terms

**Mother**: the node that immediately dominates another.
Some Informal Terms

**Mother**: the node that immediately dominates another.

**Daughter**: the node that is immediately dominated by another (is an immediate constituent of another).
Some Informal Terms

- **Mother**: the node that immediately dominates another.
- **Daughter**: the node that is immediately dominated by another (is an immediate constituent of another).
- **Sisters**: two nodes that share the same mother.
Root and Terminal Nodes
Root and Terminal Nodes

**Root node:** A node with no mother
Root node: A node with no mother

Terminal node: A node with no daughters
Root and Terminal Nodes

**Root node**: A node with no mother

**Terminal node**: A node with no daughters

The diagram shows a parse tree with:

- **TP** (Terminal Phrase)
- **NP** (Noun Phrase)
- **VP** (Verb Phrase)

The sentence 'the platypus laughed' is parsed as follows:

- **D** (Determiner): the
- **N** (Noun): platypus
- **V** (Verb): laughed
**Root and Terminal Nodes**

- **Root node**: A node with no mother
- **Terminal node**: A node with no daughters

```
the platypus laughed
```

Diagram:

```
TP

NP
D the

N platypus

VP
V laughed
```

Root Node
**Root and Terminal Nodes**

- **Root node**: A node with no mother
- **Terminal node**: A node with no daughters

```
D       N     VP
the   platypus laughed
```

Diagram:
- **Root Node**
  - **NP**
  - **VP**
  - Terminal Nodes
  - **the platypus**
  - **laughed**
Precedence
**Precedence**: Node A precedes node B if A is to the left of B. (informal definition)
Precedence

Precedence: Node A precedes node B if A is to the left of B. (informal definition)

But this runs into problems with trees which are badly drawn
Precedence excludes domination
Precedence excludes domination

Note that if two nodes are in a domination relation they cannot be in a precedence relation.
Precendence excludes domination

Note that if two nodes are in a domination relation they cannot be in a precedence relation.
Precedence excludes domination

Note that if two nodes are in a domination relation they cannot be in a precedence relation.

Is the ball to the left or right of the box?
Precedence excludes domination

Note that if two nodes are in a domination relation they cannot be in a precedence relation.

Is the ball to the left or right of the box?
Neither! You can’t precede or follow something that dominates (contains) you or you dominate (contain).
Precedence
Consider this poorly drawn tree
Consider this poorly drawn tree:

```
TP
  NP          VP
    D       N
  the clown
V
  D       N
kissed the donkey
```
Consider this poorly drawn tree

Does kiss precede clown? Obviously not!
Consider this poorly drawn tree

Does kiss precede clown? Obviously not!

What is crucial here is that the dominator of clown precedes the dominator of kissed.
In order to define precedence we’re going to need a more local relation that refers to dominance. This is sister-precedence:

A sister-precedes B if and only if
- A and B are immediately dominated by the same node
- A appears to the left of B
Sister-Precedence

```
TP
   NP       VP
   D        V
   the      left
   man
```
Sister-Precedence

NP sister-precedes VP

\[
\text{the man left}
\]
Sister-Precedence

NP sister-precedes VP
D sister precedes N
Sister-Precedence

NP sister-precedes VP
D sister precedes N
N does NOT sister precede V (nor does D)
A Precedes B if and only iff

- A does not dominate B and B does not dominate A  AND

- Either:
  - A sister-precedes B OR
  - There is some node E that dominates A, and some node F that dominates B, and E sister-precedes F.
Sister-Precedence $\neq$ Immediate Precedence

But N does immediately precede V
Sister-Precedence ≠ Immediate Precedence

N does NOT sister-precede V
But N does immediately precede V
No Crossing Branches Constraint

If one node X precedes another node Y then X and all nodes dominated by X must precede Y and all nodes dominated by Y.
No Crossing Branches Constraint

If one node X precedes another node Y then X and all nodes dominated by X must precede Y and all nodes dominated by Y.
Immediate Precedence:

A immediately precedes B if there is no node G which follows A but precedes B.
Immediate Precedence:

A immediately precedes B if there is no node G which follows A but precedes B.
Immediate Precedence:

A immediately precedes B if there is no node G which follows A but precedes B.
Sister-Precedence ≠ Immediate Precedence

But N does immediately precede V
Sister-Precedence ≠ Immediate Precedence

N does NOT sister-precede V
But N does immediately precede V
C-command
C-command

Intuitively: The relationship between a node and its sister, and all the daughters of its sister
C-command

Intuitively: The relationship between a node and its sister, and all the daughters of its sister.
C-command

Intuitively: The relationship between a node and its sister, and all the daughters of its sister
C-command

Intuitively: The relationship between a node and its sister, and all the daughters of its sister

A c-commands C, D, E, F, G, H

Note: D does NOT c-command A
Node A c-commands node B if every node dominating A also dominates B, and A does not itself dominate B.
Node A **c-commands** node B if

- every node dominating A also dominates B,
- and A does not itself dominate B.
C-command

Node A \textbf{c-commands} node B if

every node dominating A also dominates B,

and A does not itself dominate B.

you can’t command something you dominate
Symmetric C-command
Symmetric C-command

A symmetrically c-commands B, if A c-commands B AND B c-commands A
Symmetric C-command

A symmetrically c-commands B, if A c-commands B AND B c-commands A

SAME THING AS SISTERHOOD
Symmetric C-command

A symmetrically c-commands B, if A c-commands B AND B c-commands A

SAME THING AS SISTERHOOD
Symmetric C-command

A symmetrically c-commands B, if A c-commands B AND B c-commands A

SAME THING AS SISTERHOOD

A & B symmetrically c-command one another
Symmetric C-command

A symmetrically c-commands B, if A c-commands B AND B c-commands A

SAME THING AS SISTERHOOD

A & B symmetrically c-command one another
Symmetric C-command

A symmetrically c-commands B, if A c-commands B AND B c-commands A

SAME THING AS SISTERHOOD

A & B symmetrically c-command one another
A does NOT symmetrically c-command D
Asymmetric C-command

A asymmetrically c-commands B, if A c-commands B but B does NOT c-command A.

(intuitively – A is B’s aunt)
Asymmetric C-command

A asymmetrically c-commands B, if A c-commands B but B does NOT c-command A.

(intuitively – A is B’s aunt)
Asymmetric C-command

A asymmetrically c-commands B, if A c-commands B but B does NOT c-command A.

(intuitively – A is B’s aunt)
Grammatical Relations

Subject: NP/CP daughter of TP

Object of a Preposition: NP daughter of PP

Direct Object:
- With verbs of type \( V_{\text{NP__NP}} \), \( V_{\text{NP__CP}} \) and \( V_{\text{NP__NP PP}} \), the NP or CP daughter of VP
- With verbs of type \( V_{\text{NP __ NP \{NP/CP\}}} \), an NP or CP daughter of VP that is preceded by another NP daughter of VP. (i.e., the second NP daughter of VP)
Grammatical Relations

**Indirect Object**: This is the 1st object indicating the goal of a verb of transfer (a ditransitive) or the PP of the same kind of verb:

- With verbs of type $V[\text{NP} \__ \text{NP PP}]$, the PP daughter of VP immediately preceded by an NP daughter of VP.
- With verbs of type $V[\text{NP} \__ \text{NP} \{\text{NP/CP}\}]$, the NP daughter of VP immediately preceded by $V$ (i.e. the first NP daughter of VP)

**Oblique**: any other NP/PP in the sentence.
Grammatical Relations
Grammatical Relations
Grammatical Relations

Subject

NP

TP

VP

N
Grammatical Relations

Subject  NP  N  V  NP  N  Object
Grammatical Relations

Subject → NP → N

TP → VP → V → NP → N

Object

PP → P → NP → N
Grammatical Relations

Subject → NP \[ \text{N} \] \[ \text{NP} \] → VP → V \[ \text{NP} \] \[ \text{NP} \] → Object

Object of a Preposition → PP → P \[ \text{NP} \] \[ \text{NP} \]
Grammatical Relations
Grammatical Relations

I gave Adam the book
Grammatical Relations

I gave Adam the book
Grammatical Relations

I gave Adam the book
Grammatical Relations

I gave Adam the book
Grammatical Relations

Indirect Object: I gave Adam the book

Direct Object: I gave the book to Adam
Grammatical Relations

I gave Adam the book

I gave the book to Adam
Grammatical Relations

I gave Adam the book

I gave the book to Adam
Grammatical Relations

I gave Adam the book

I gave the book to Adam
Summary

Structural Relations: relationships between nodes.

Dominance (=containment)
- immediate dominance (=motherhood)
- exhaustive dominance (=constituent)

Precedence (=to the left)
- immediate precedence (=adjacent & to the left)
Summary

C-command: sisters & nieces

Symmetric C-command: sisters

Asymmetric C-command: Aunt asymmetrically c-commands nieces

Grammatical Relations: Subject, Direct Object, Indirect Object, Object of a Preposition.