Natural Language Models and Interfaces

BSc Artificial Intelligence

Lecturer: Wilker Aziz
Institute for Logic, Language, and Computation

2018, week 5, lecture b

Context-Free Grammars

A **CFG** grammar G is denoted by

- a finite set of **nonterminal** symbols $\mathscr{V}$
- a finite set of **terminal** symbols Σ with $\Sigma \cap \mathscr{V} = \varnothing$
- a finite set $\mathscr{R}$ of **rules** of the form $X \rightarrow \beta$ where
  - $X \in \mathscr{V}$ and $\beta \in (\Sigma \cup \mathscr{V})^*$
- $S \in \mathscr{V}$ a distinguished **start** symbol

Let ε denote an **empty** string

Example CFG

S → NP VP	Vi → sleeps
VP → Vi	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Generative Device

Left-most derivation

- sequence of strings α_1 ... α_n
  - $\alpha_1 = \langle S \rangle$
  - $\alpha_n \in \Sigma^*$
  - $\alpha_{i \geq 2}$ derived from α_{i-1} by picking the left-most nonterminal X
    - and replacing it by some α such that $X \rightarrow \beta \in \mathscr{R}$

Example of Derivation

Example of Derivation

String	Substitution

Example of Derivation

	String	Substitution
$\alpha_1 =$	S	S → NP VP

Example of Derivation

	String	Substitution
α_1 =	S	S → NP VP
α_2 =	NP VP	NP → DT NN

Example of Derivation

	String	Substitution
$\alpha_1 =$	S	S → NP VP
$\alpha_2 =$	NP VP	NP → DT NN
$\alpha_3 =$	DT NN VP	DT → the

Example of Derivation

	String	Substitution
α_1 =	S	S → NP VP
α_2 =	NP VP	NP → DT NN
α_3 =	DT NN VP	DT → the
α_4 =	the NN VP	NN → man

Example of Derivation

	String	Substitution
$\alpha_1 =$	S	S → NP VP
$\alpha_2 =$	NP VP	NP → DT NN
$\alpha_3 =$	DT NN VP	DT → the
$\alpha_4 =$	the NN VP	NN → man
$\alpha_5 =$	the man VP	VP → Vi

Example of Derivation

	String	Substitution
α_1 =	S	S → NP VP
α_2 =	NP VP	NP → DT NN
α_3 =	DT NN VP	DT → the
α_4 =	the NN VP	NN → man
α_5 =	the man VP	VP → Vi
α_6 =	the man Vi	Vi → sleeps

Example of Derivation

	String	Substitution
$\alpha_1 =$	S	S → NP VP
$\alpha_2 =$	NP VP	NP → DT NN
$\alpha_3 =$	DT NN VP	DT → the
$\alpha_4 =$	the NN VP	NN → man
$\alpha_5 =$	the man VP	VP → Vi
$\alpha_6 =$	the man Vi	Vi → sleeps
$\alpha_7 =$	the man sleeps	

Example of Derivation

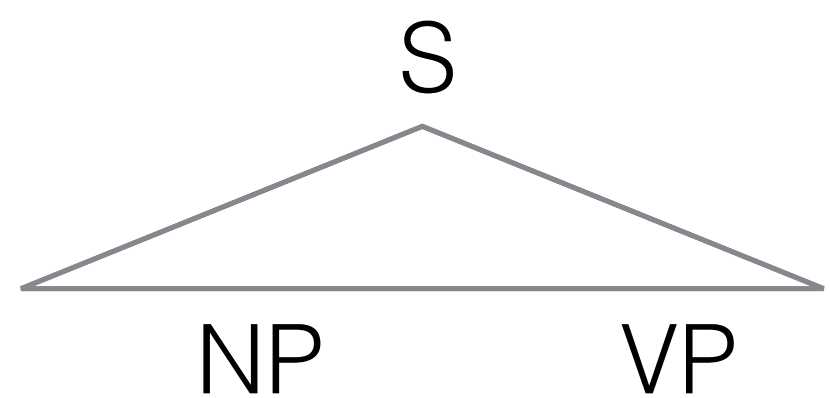
	String	Substitution
α_1 =	S	S → NP VP
α_2 =	NP VP	NP → DT NN
α_3 =	DT NN VP	DT → the
α_4 =	the NN VP	NN → man
α_5 =	the man VP	VP → Vi
α_6 =	the man Vi	Vi → sleeps
α_7 =	the man sleeps	
α_7 =	S $\Rightarrow^*$ the man sleeps	

Example of Generation

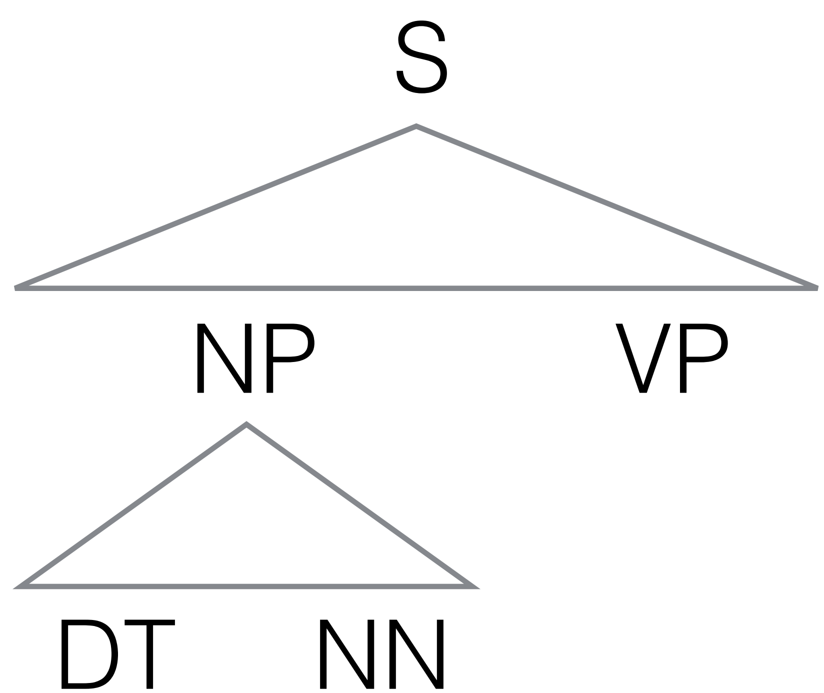
Example of Generation

S

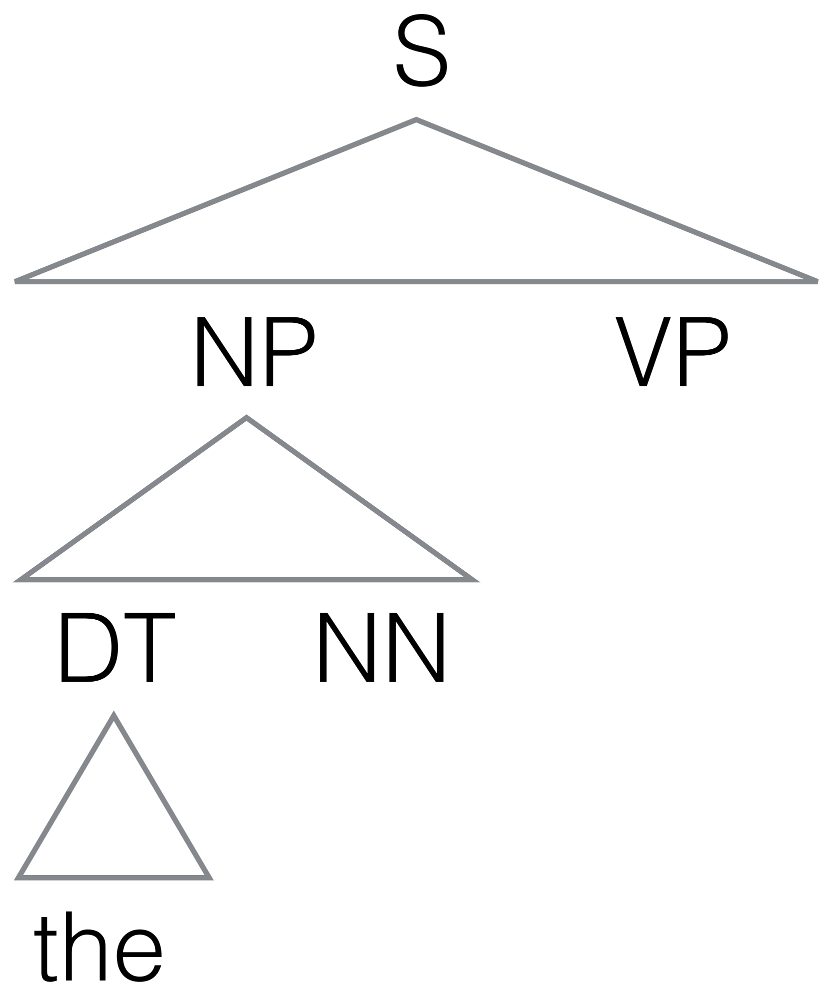
Example of Generation



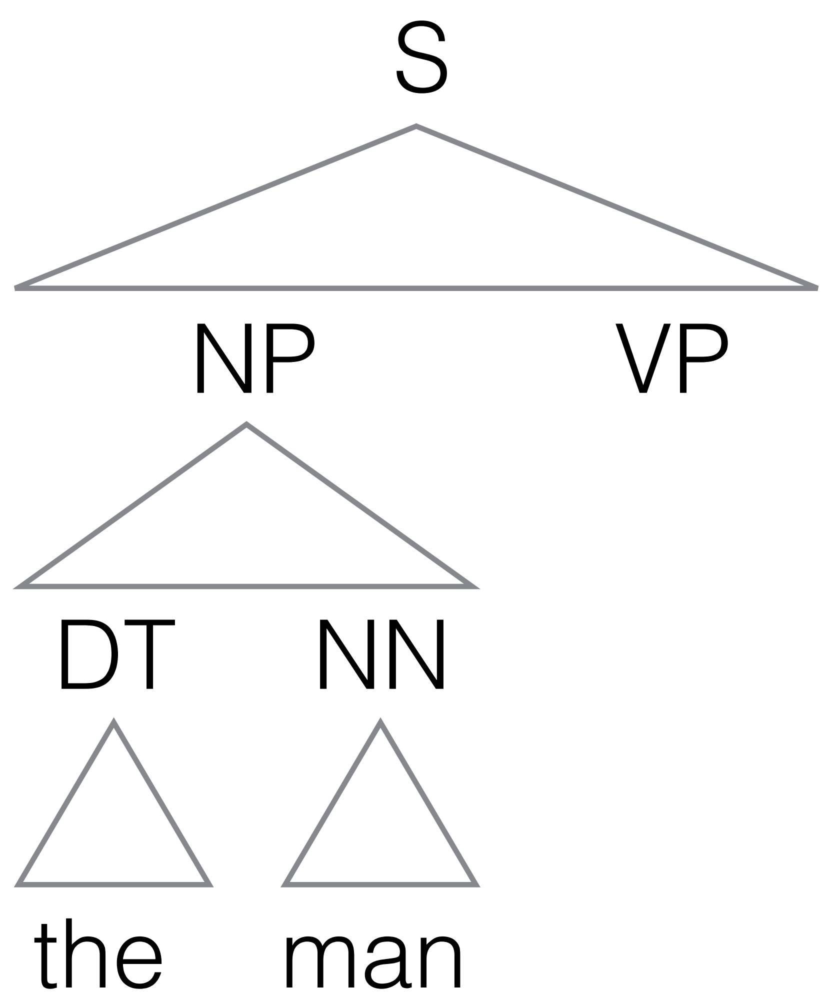
Example of Generation



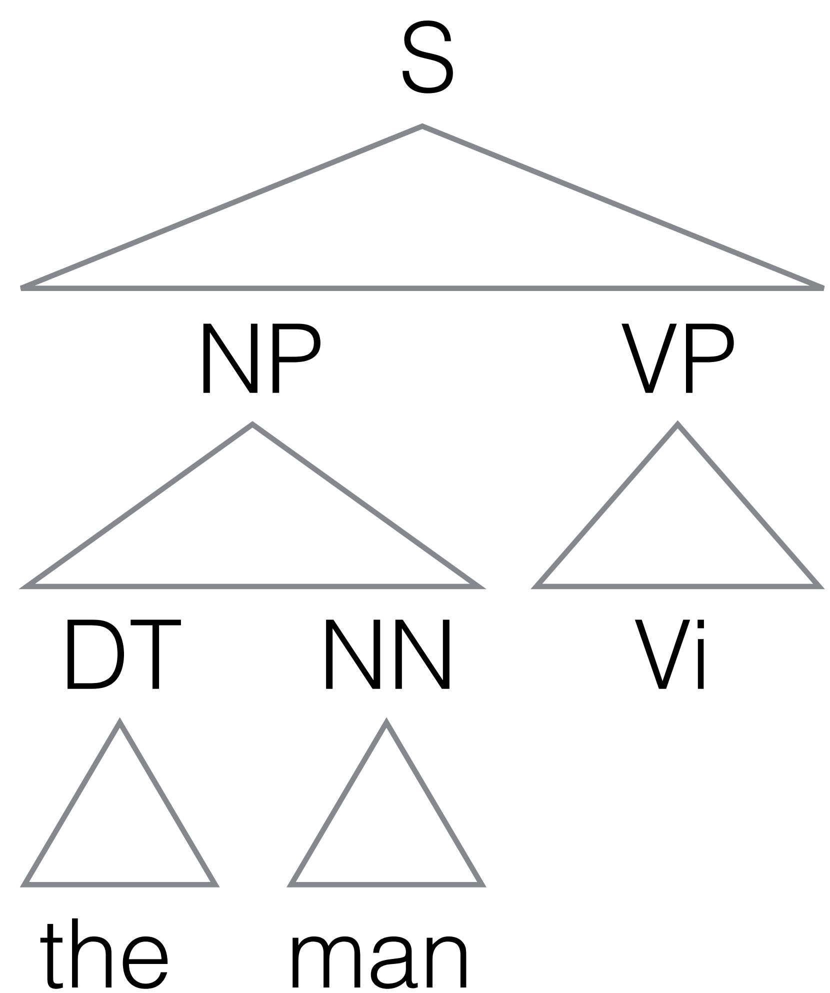
Example of Generation



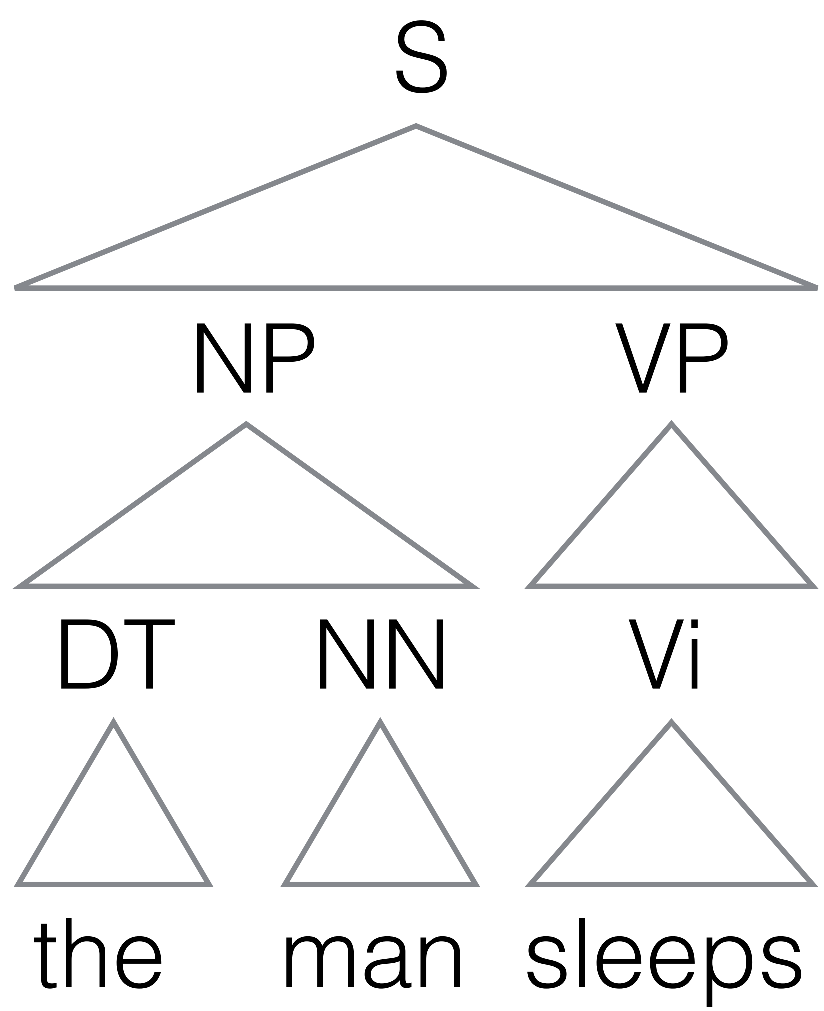
Example of Generation



Example of Generation



Example of Generation

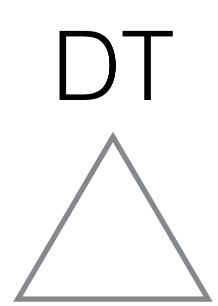


Example of Recognition

Example of Recognition

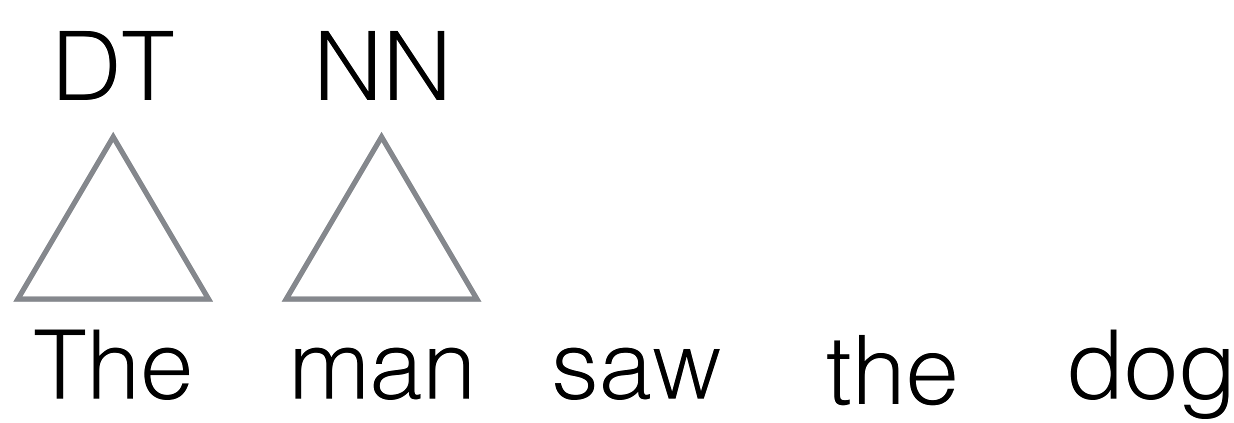
The man saw the dog

Example of Recognition

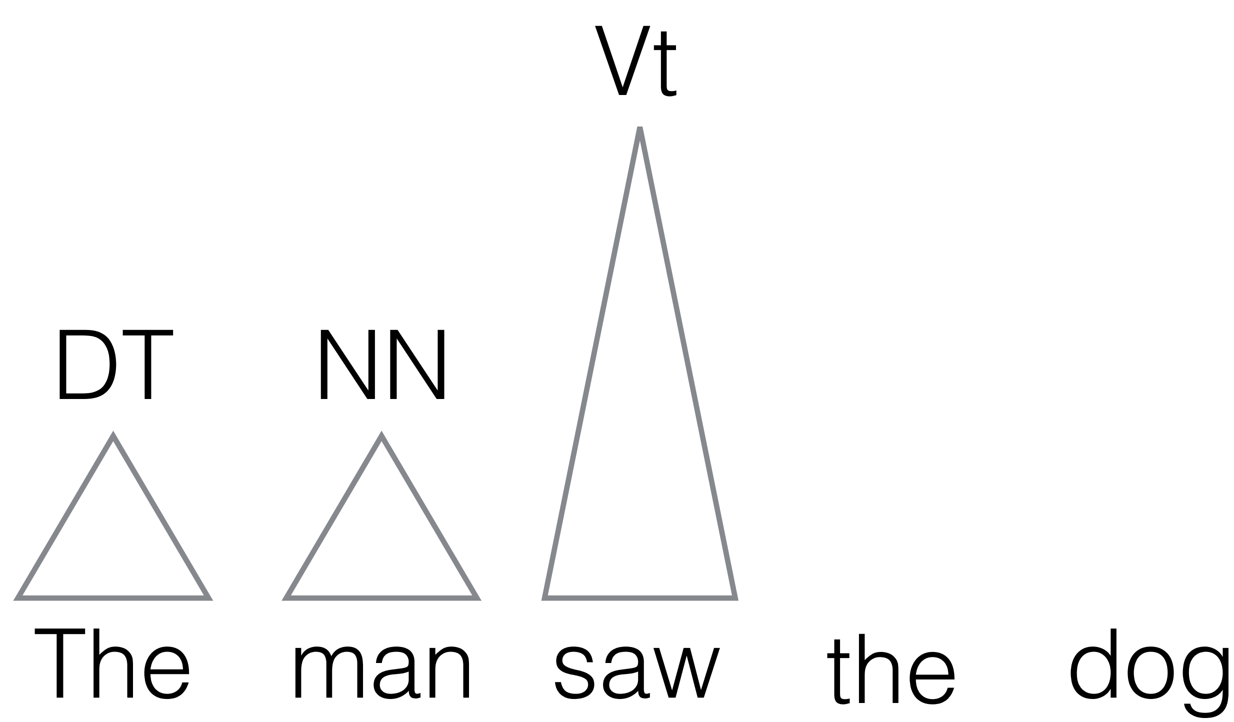


The man saw the dog

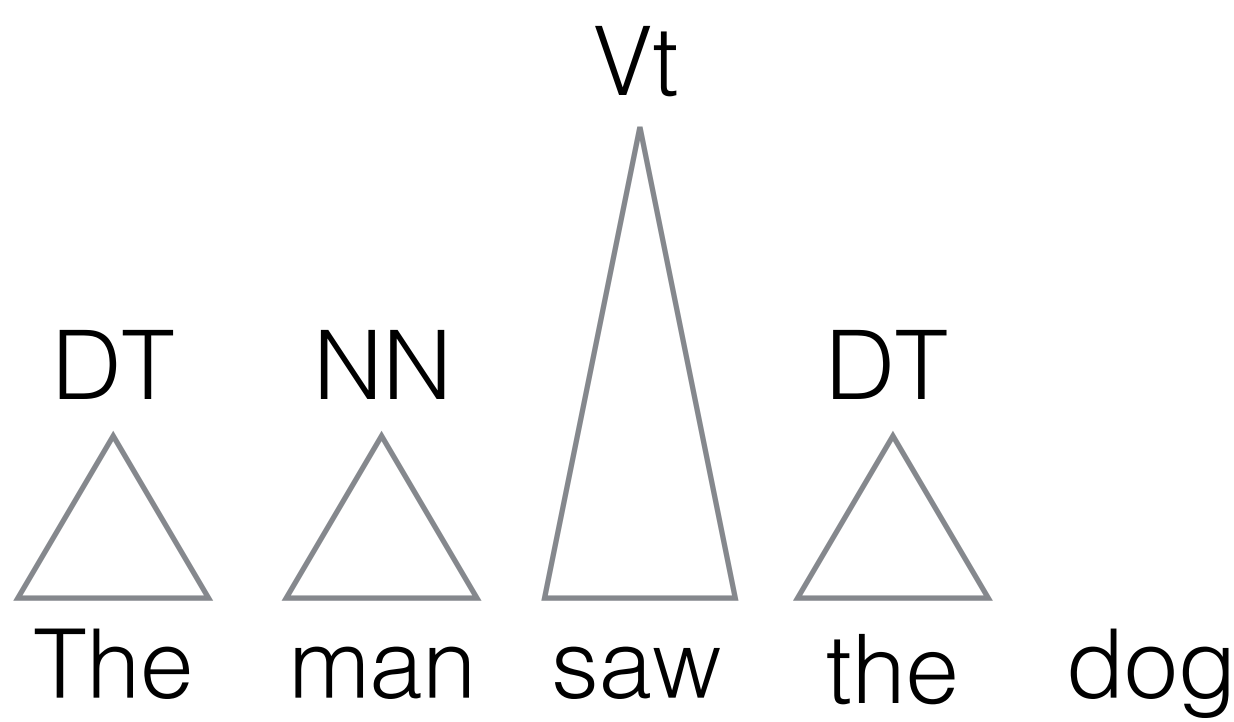
Example of Recognition



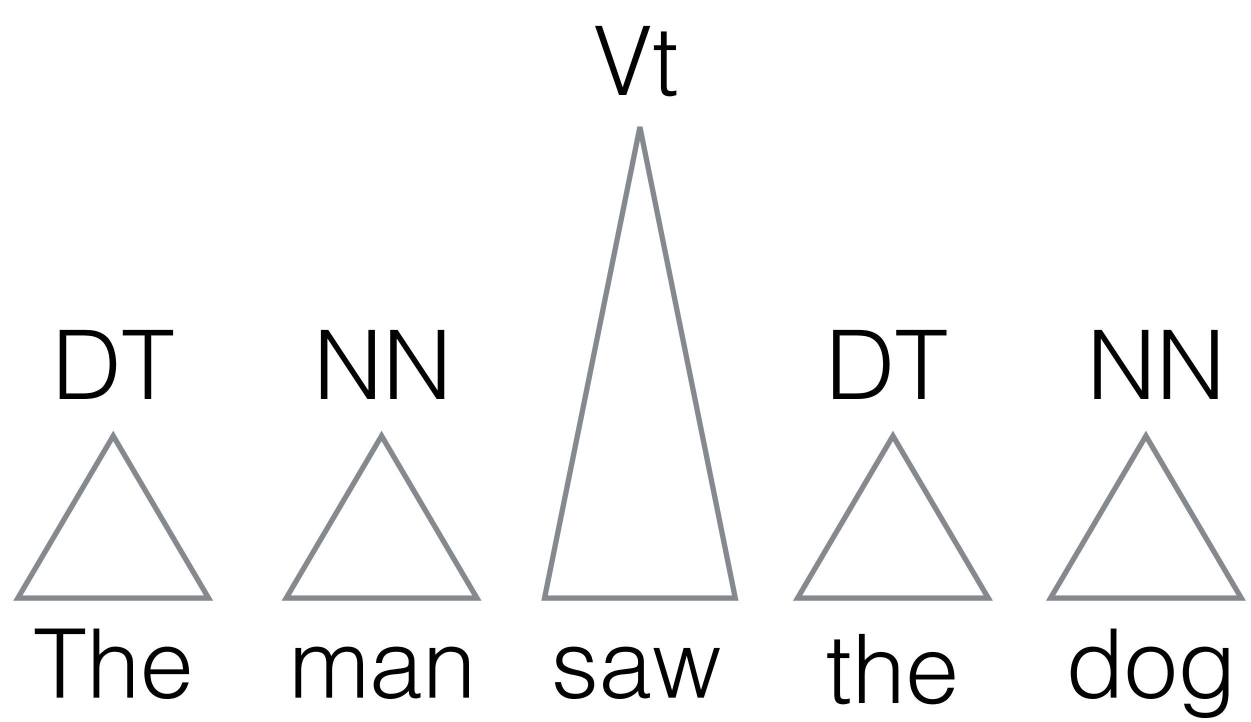
Example of Recognition



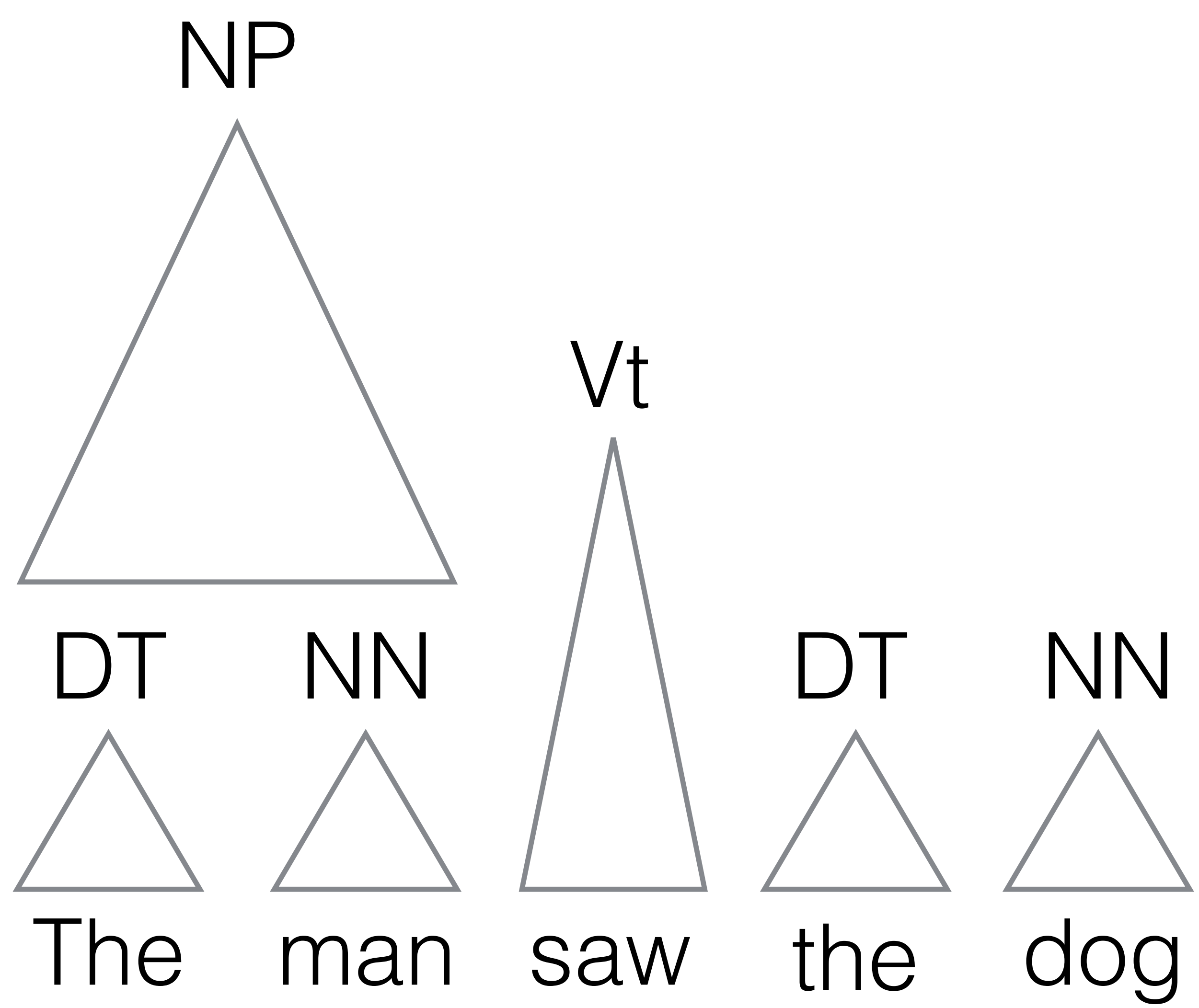
Example of Recognition



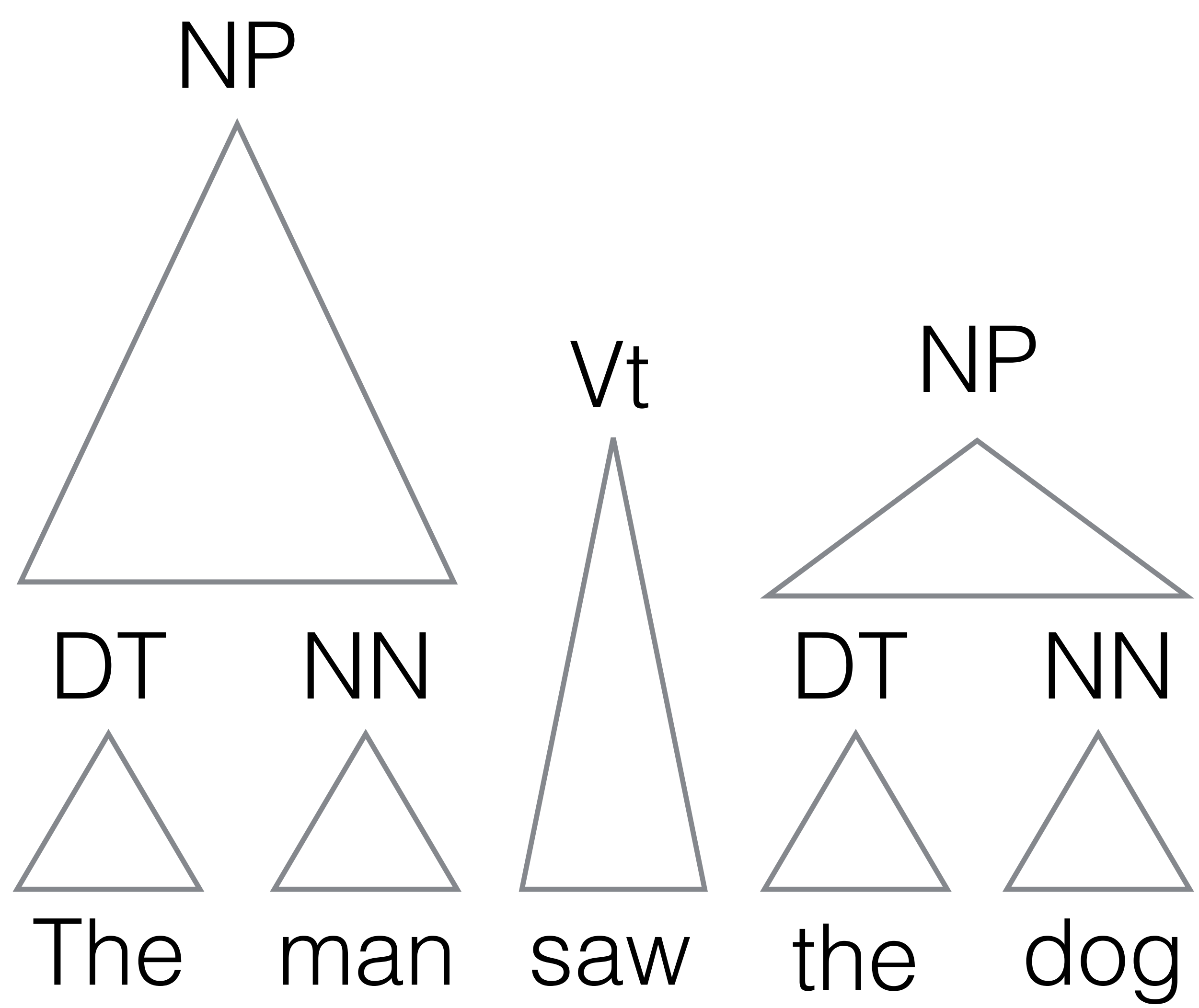
Example of Recognition



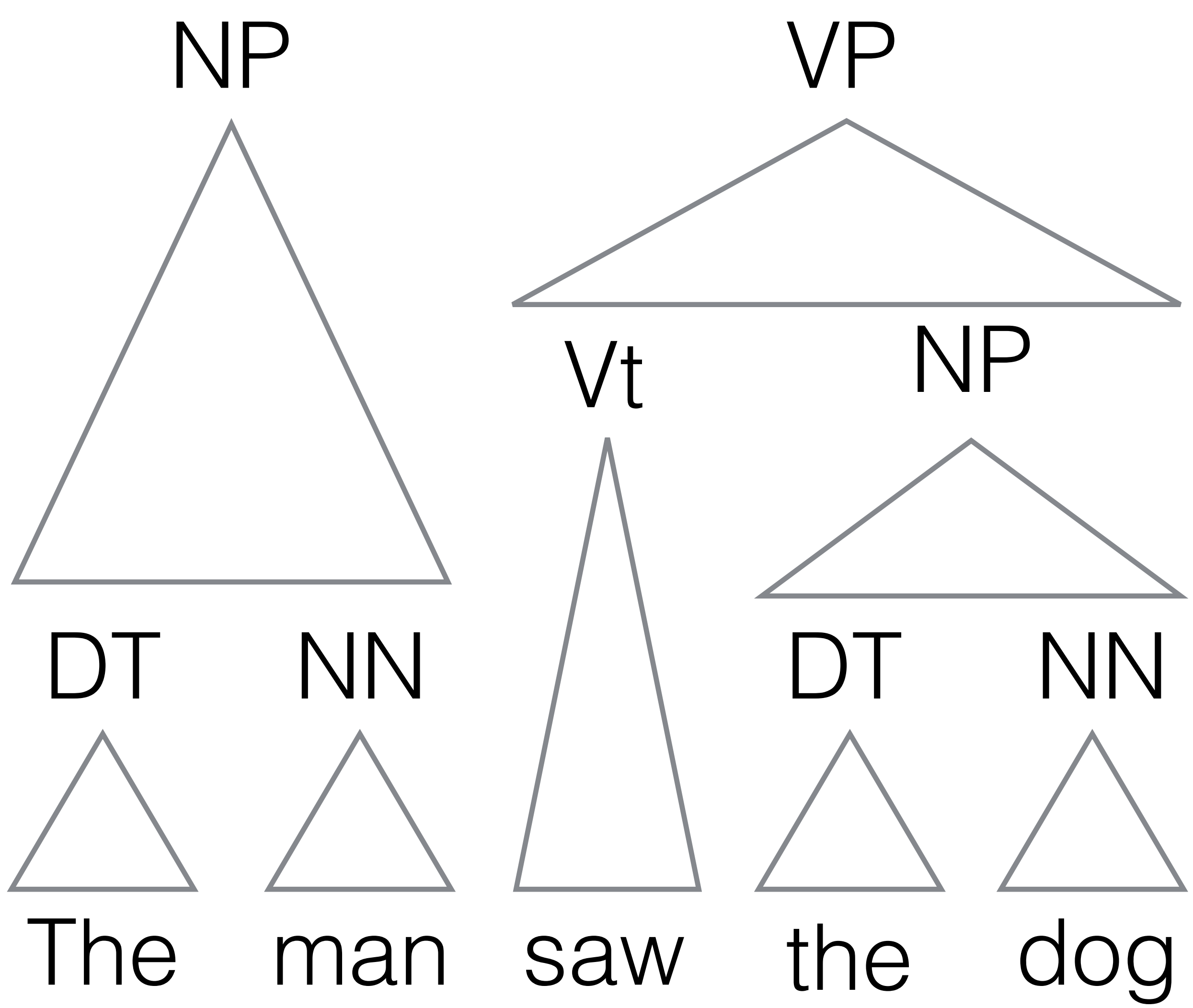
Example of Recognition



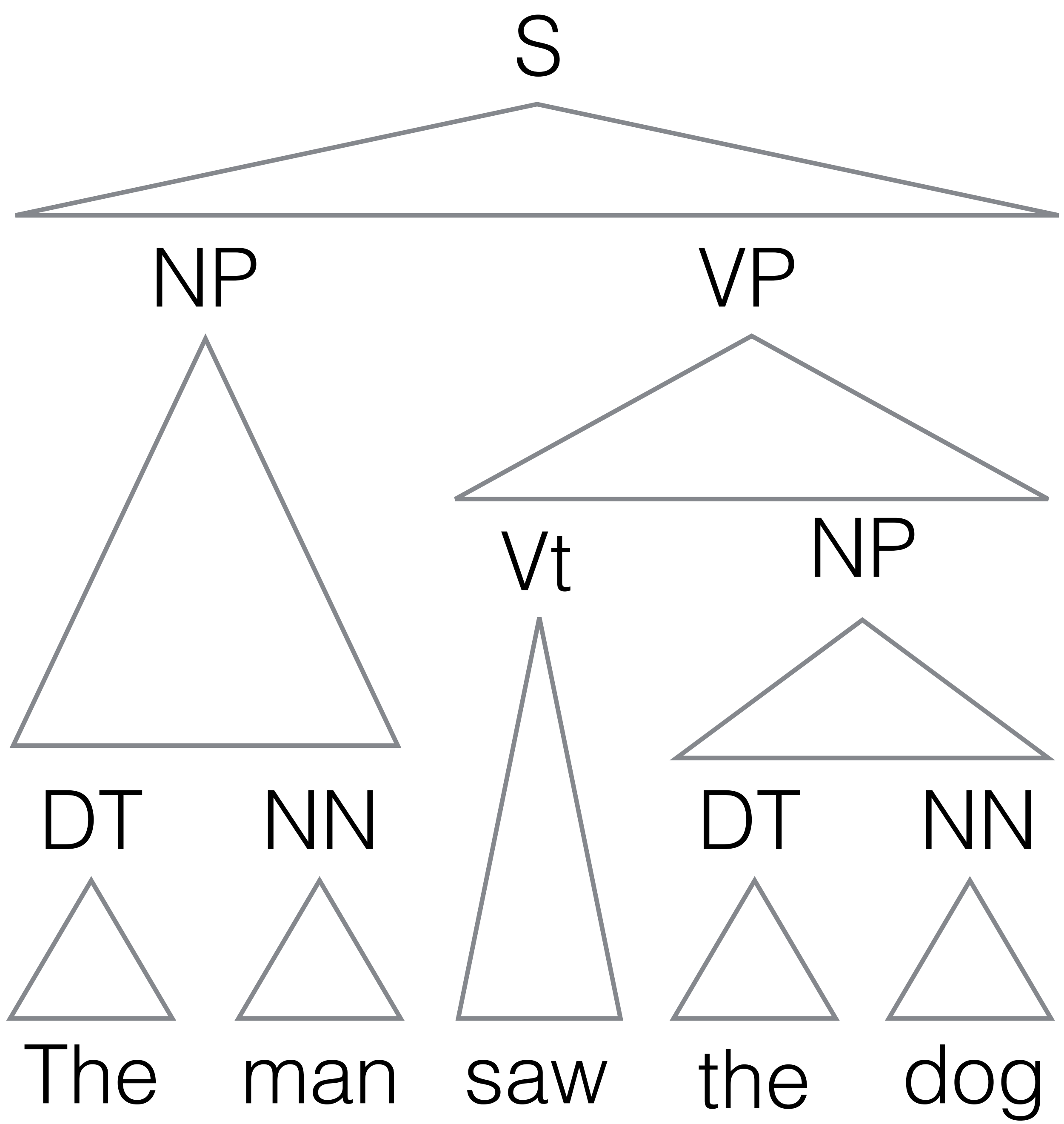
Example of Recognition



Example of Recognition



Example of Recognition



Language

A string $\omega \in \Sigma^*$ is generated/accepted by G if

$$S \Rightarrow^* \omega$$

$\Rightarrow^*$ denotes a sequence of rule applications

Language of G

$$L(G) = \{\omega: S \Rightarrow^* \omega\} \subseteq \Sigma^*$$

Chomsky Normal Form

Every CFG is weakly equivalent to another such that

- $X \rightarrow Y\ Z$ where $X, Y, Z \in \mathscr{V}$
- $X \rightarrow w$ where $w \in \Sigma$
- and possibly $S \rightarrow \varepsilon$

[Hopcroft and Ullman, 1979]

Parsing as Deduction

Parsing as Deduction

Deductive process to prove claims about grammaticality
[Shieber et al., 1995]

Parsing as Deduction

Deductive process to prove claims about grammaticality

[Shieber et al., 1995]

- focus on strategy rather than implementation

Parsing as Deduction

Deductive process to prove claims about grammaticality
[Shieber et al., 1995]

- focus on strategy rather than implementation
- soundness/completeness easier to prove

Parsing as Deduction

Deductive process to prove claims about grammaticality
[Shieber et al., 1995]

- focus on strategy rather than implementation
- soundness/completeness easier to prove
- complexity determined by inspection

Parsing as Deduction

Deductive process to prove claims about grammaticality

[Shieber et al., 1995]

- focus on strategy rather than implementation
- soundness/completeness easier to prove
- complexity determined by inspection
- dynamic program follows directly

Parsing as Deduction

Deductive process to prove claims about grammaticality

[Shieber et al., 1995]

- focus on strategy rather than implementation
- soundness/completeness easier to prove
- complexity determined by inspection
- dynamic program follows directly
- generality

Deductive systems

Item: a statement / intermediate sound result

- formula or schemata expressed with variables

Inference rule: statement derived from existing items

- $\dfrac{A_1 \ldots A_m}{B}$ (condition) where A_i and B are items
  - Ai are called antecedents
  - B is called consequent

Deductive program

Axioms: trivial items

- do not depend on previous statements

Goal: states that a proof exists

Proof:

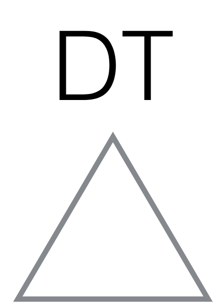
- start from axioms
- exhaustively deduce items
  - never twice under the same premises
- accept if goal is proven

Bottom-up: Shift-Reduce

Bottom-up: Shift-Reduce

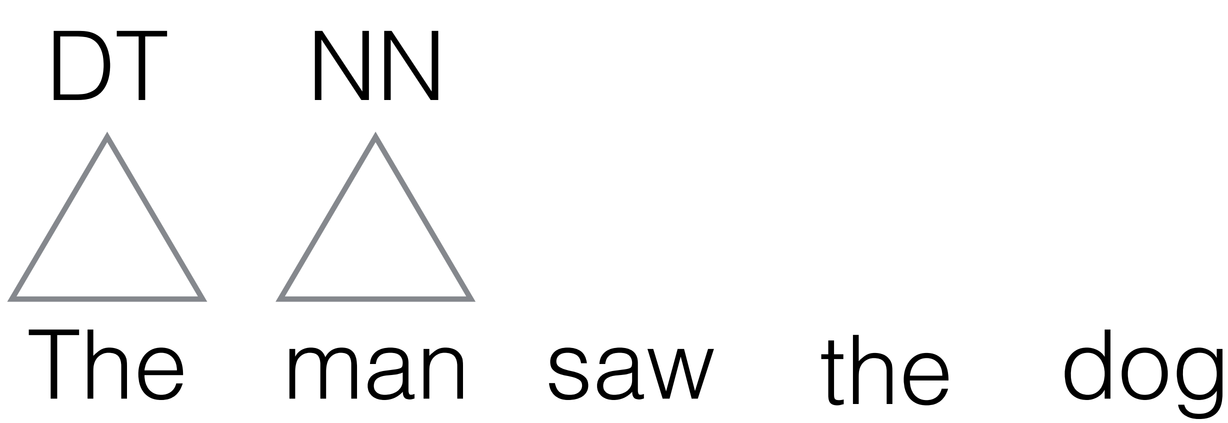
The man saw the dog

Bottom-up: Shift-Reduce

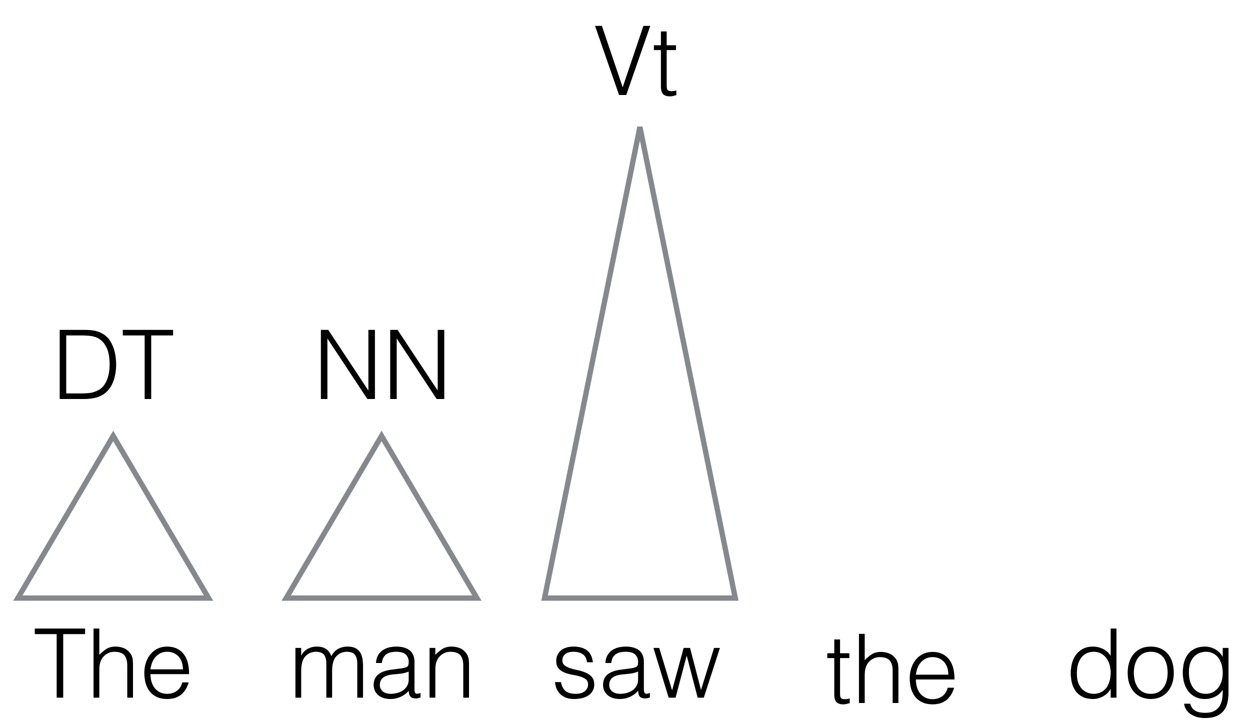


The man saw the dog

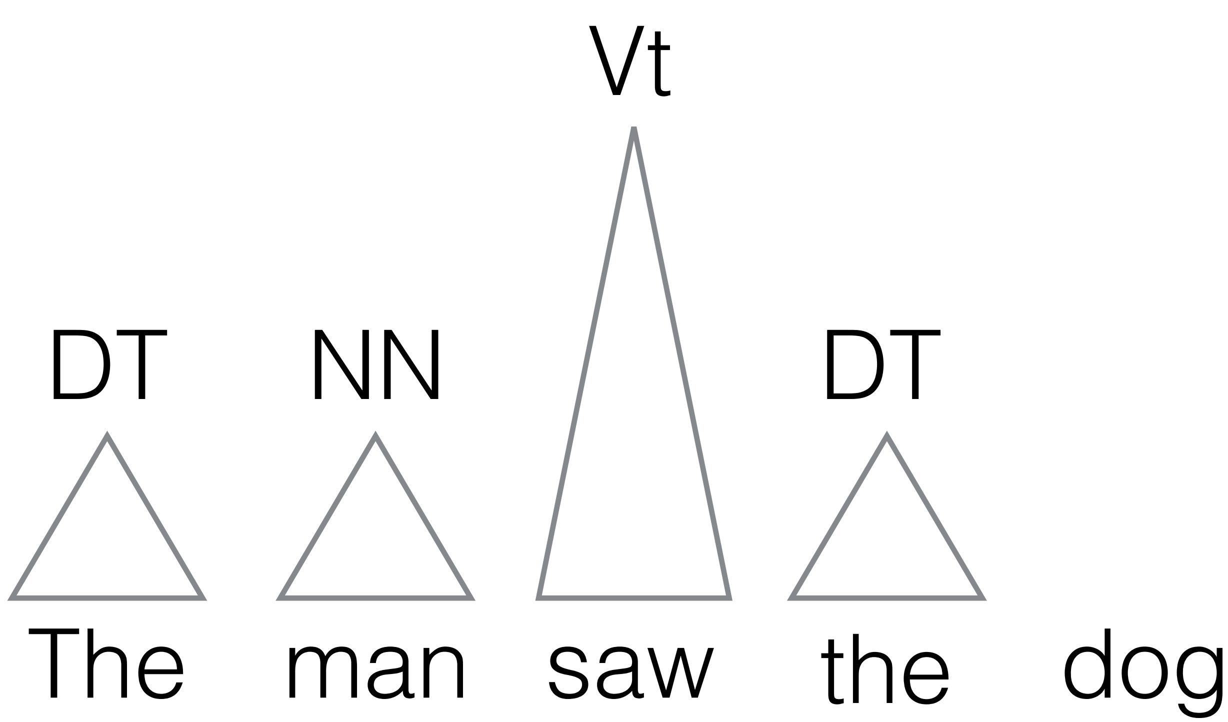
Bottom-up: Shift-Reduce



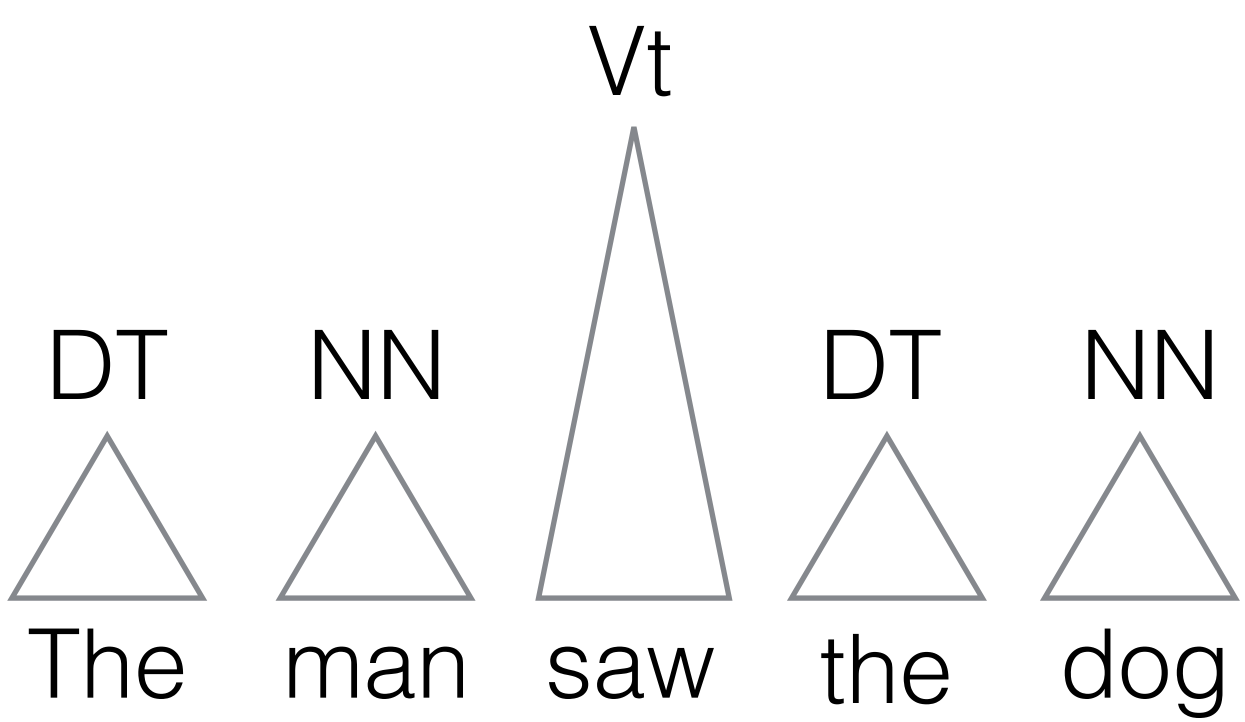
Bottom-up: Shift-Reduce



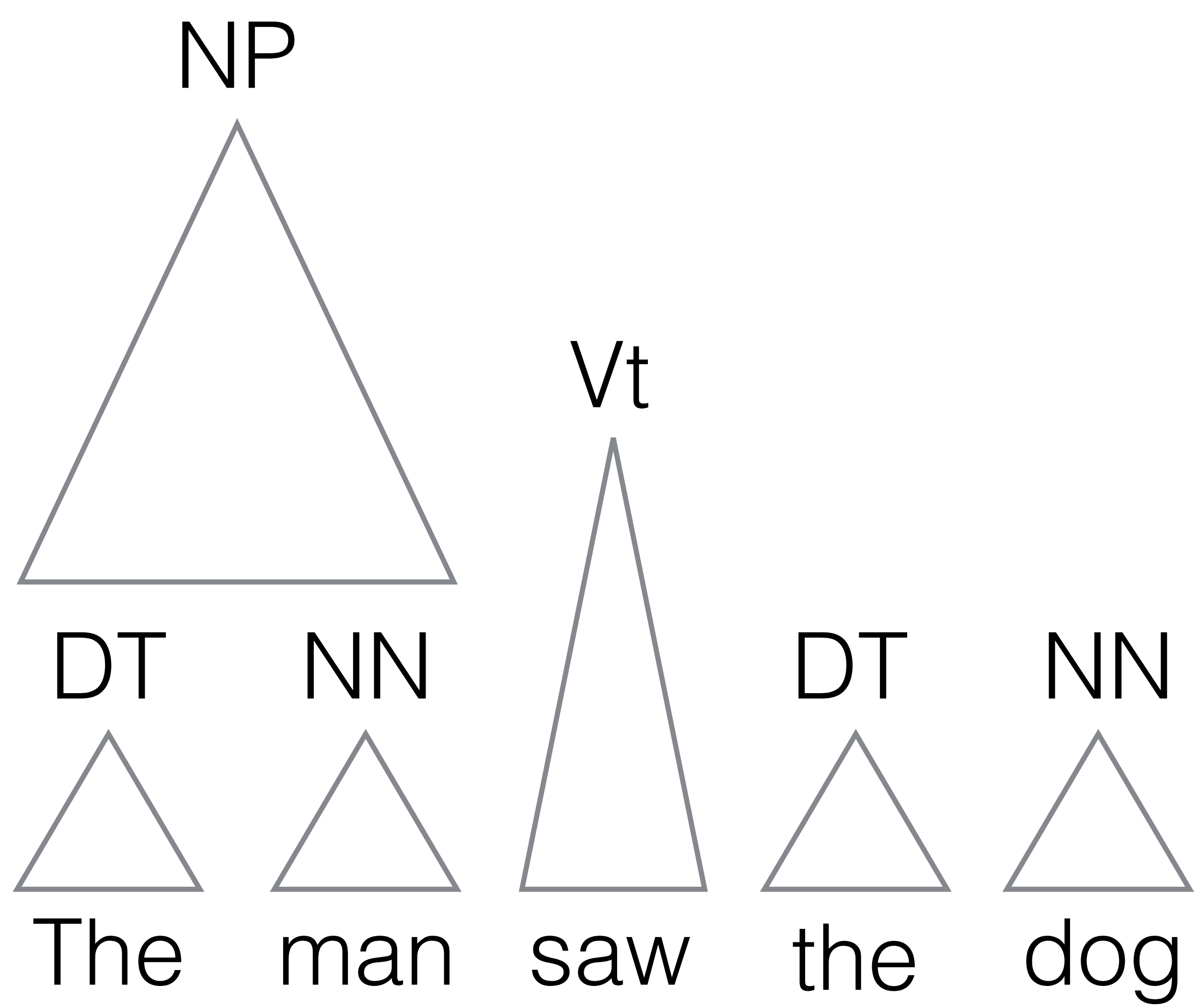
Bottom-up: Shift-Reduce



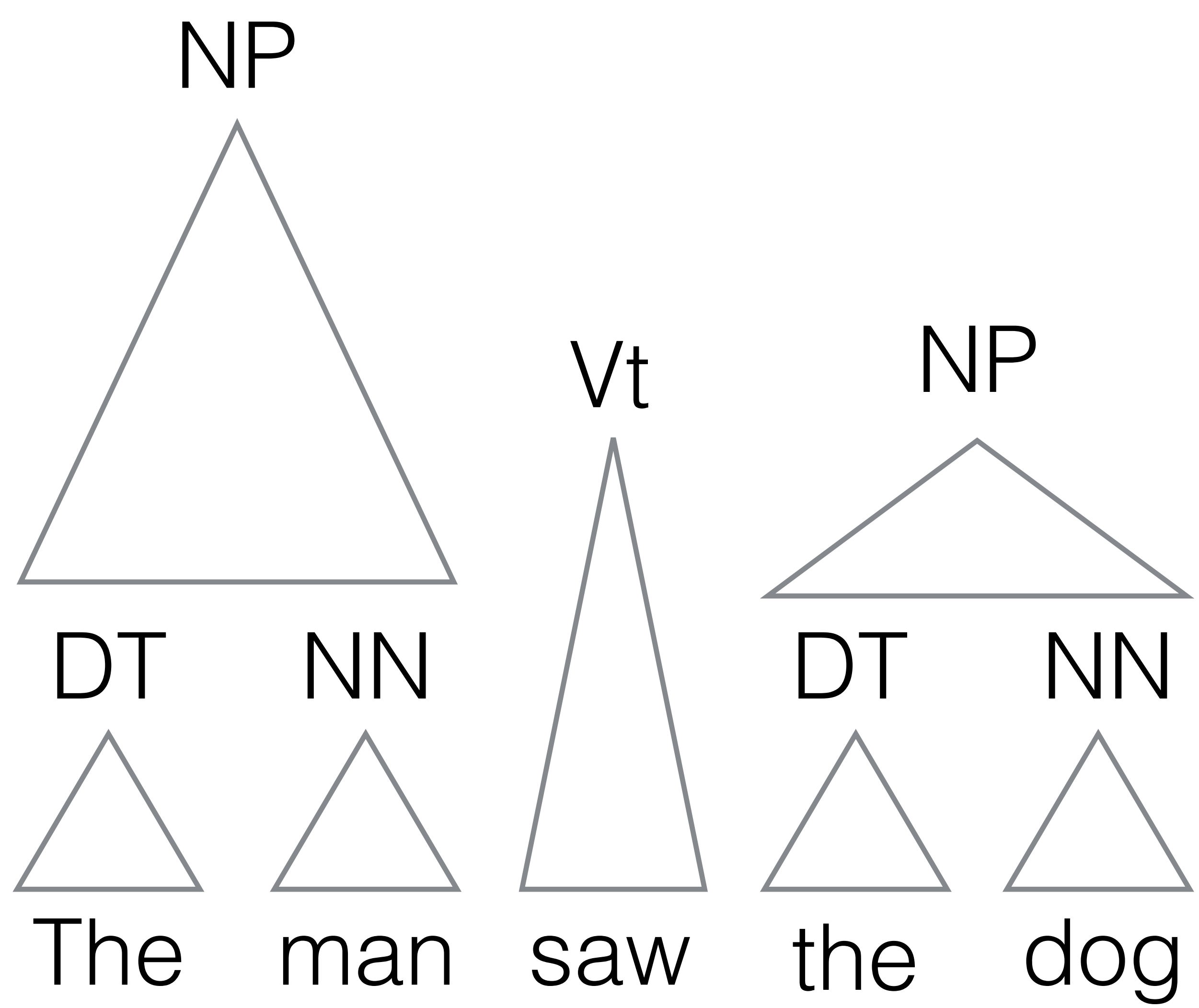
Bottom-up: Shift-Reduce



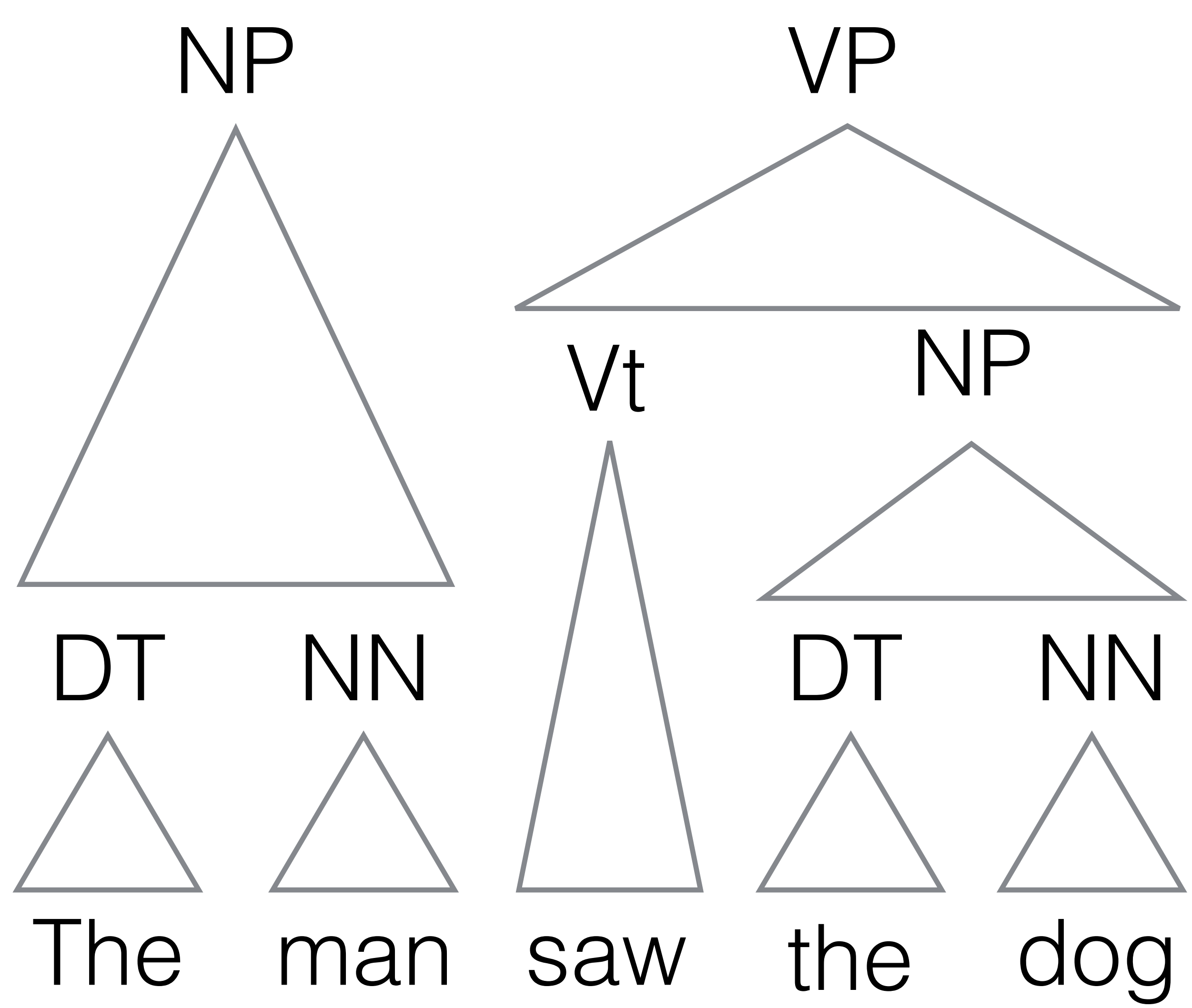
Bottom-up: Shift-Reduce



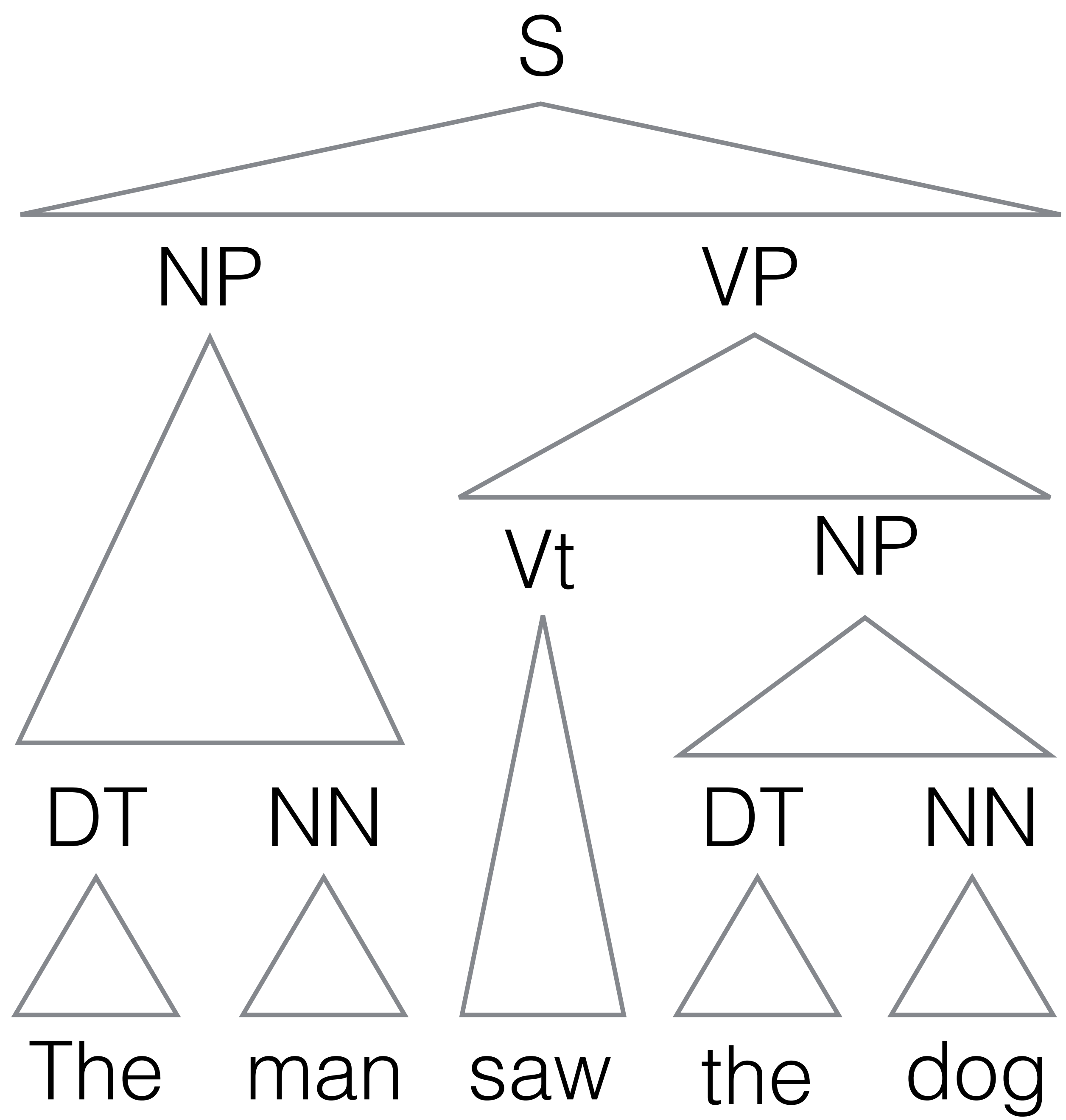
Bottom-up: Shift-Reduce



Bottom-up: Shift-Reduce



Bottom-up: Shift-Reduce



Shift-Reduce Example

Input: *the man sleeps*

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition	Statement	Queue

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition	Statement	Queue
Axiom	1	[•,0]	1

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition	Statement	Queue
Axiom	1	[•,0]	1

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition	Statement	Queue
Axiom	1	[•,0]	1
Shift: [1]	2	[the•,1]	2

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition	Statement	Queue
Axiom	1	[•,0]	1
Shift: [1]	2	[the•,1]	2

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3
Shift: [3]		4	[DT man •, 2]	4

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3
Shift: [3]		4	[DT man •, 2]	4

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3
Shift: [3]		4	[DT man •, 2]	4
Reduce: [4]	NN → man	5	[DT NN •, 2]	5

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3
Shift: [3]		4	[DT man •, 2]	4
Reduce: [4]	NN → man	5	[DT NN •, 2]	5

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3
Shift: [3]		4	[DT man •, 2]	4
Reduce: [4]	NN → man	5	[DT NN •, 2]	5
Reduce: [5]	NP → DT NN	6	[NP •, 2]	6

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3
Shift: [3]		4	[DT man •, 2]	4
Reduce: [4]	NN → man	5	[DT NN •, 2]	5
Reduce: [5]	NP → DT NN	6	[NP •, 2]	6

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3
Shift: [3]		4	[DT man •, 2]	4
Reduce: [4]	NN → man	5	[DT NN •, 2]	5
Reduce: [5]	NP → DT NN	6	[NP •, 2]	6
Shift: [6]		7	[NP sleeps •, 3]	7

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition	Statement		Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3
Shift: [3]		4	[DT man •, 2]	4
Reduce: [4]	NN → man	5	[DT NN •, 2]	5
Reduce: [5]	NP → DT NN	6	[NP •, 2]	6
Shift: [6]		7	[NP sleeps •, 3]	7

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3
Shift: [3]		4	[DT man •, 2]	4
Reduce: [4]	NN → man	5	[DT NN •, 2]	5
Reduce: [5]	NP → DT NN	6	[NP •, 2]	6
Shift: [6]		7	[NP sleeps •, 3]	7
Reduce: [7]	Vi → sleeps	8	[NP Vi •, 3]	8

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3
Shift: [3]		4	[DT man •, 2]	4
Reduce: [4]	NN → man	5	[DT NN •, 2]	5
Reduce: [5]	NP → DT NN	6	[NP •, 2]	6
Shift: [6]		7	[NP sleeps •, 3]	7
Reduce: [7]	Vi → sleeps	8	[NP Vi •, 3]	8

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3
Shift: [3]		4	[DT man •, 2]	4
Reduce: [4]	NN → man	5	[DT NN •, 2]	5
Reduce: [5]	NP → DT NN	6	[NP •, 2]	6
Shift: [6]		7	[NP sleeps •, 3]	7
Reduce: [7]	Vi → sleeps	8	[NP Vi •, 3]	8
Reduce: [8]	VP → Vi	9	[NP VP •, 3]	9

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3
Shift: [3]		4	[DT man •, 2]	4
Reduce: [4]	NN → man	5	[DT NN •, 2]	5
Reduce: [5]	NP → DT NN	6	[NP •, 2]	6
Shift: [6]		7	[NP sleeps •, 3]	7
Reduce: [7]	Vi → sleeps	8	[NP Vi •, 3]	8
Reduce: [8]	VP → Vi	9	[NP VP •, 3]	9

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3
Shift: [3]		4	[DT man •, 2]	4
Reduce: [4]	NN → man	5	[DT NN •, 2]	5
Reduce: [5]	NP → DT NN	6	[NP •, 2]	6
Shift: [6]		7	[NP sleeps •, 3]	7
Reduce: [7]	Vi → sleeps	8	[NP Vi •, 3]	8
Reduce: [8]	VP → Vi	9	[NP VP •, 3]	9
Reduce: [9]	S → NP VP	10	[S •, 3]	10

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[•,0]	1
Shift: [1]		2	[the•,1]	2
Reduce: [2]	DT → the	3	[DT•,1]	3
Shift: [3]		4	[DT man •, 2]	4
Reduce: [4]	NN → man	5	[DT NN •, 2]	5
Reduce: [5]	NP → DT NN	6	[NP •, 2]	6
Shift: [6]		7	[NP sleeps •, 3]	7
Reduce: [7]	Vi → sleeps	8	[NP Vi •, 3]	8
Reduce: [8]	VP → Vi	9	[NP VP •, 3]	9
Reduce: [9]	S → NP VP	10	[S •, 3]	10
GOAL: [10]				∅

S → NP VP
VP → Vi
VP → Vt NP
VP → VP PP
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Shift-Reduce

Input: G and $x_1 \ldots x_n$

Item form: $[\alpha\bullet, j]$
asserts that $\alpha \Rightarrow^* x_1 \ldots x_j$ or
that $\alpha\, x_{j+1} \ldots x_n \Rightarrow^* x_1 \ldots x_n$

Axiom: $[\bullet, 0]$

Goal: $[S\bullet, n]$

$$\textsc{Shift}\ \frac{[\alpha\bullet, j]}{[\alpha\, x_{j+1}, j+1]}$$

Scan (shift)
asserts that $\alpha\, x_{j+1} \Rightarrow^* x_1 \ldots x_j\, x_{j+1}$

$$\textsc{Reduce}\ \frac{[\alpha\,\beta\bullet, j]}{[\alpha\, B, j]}\ B \to \beta \in \mathcal{R}$$

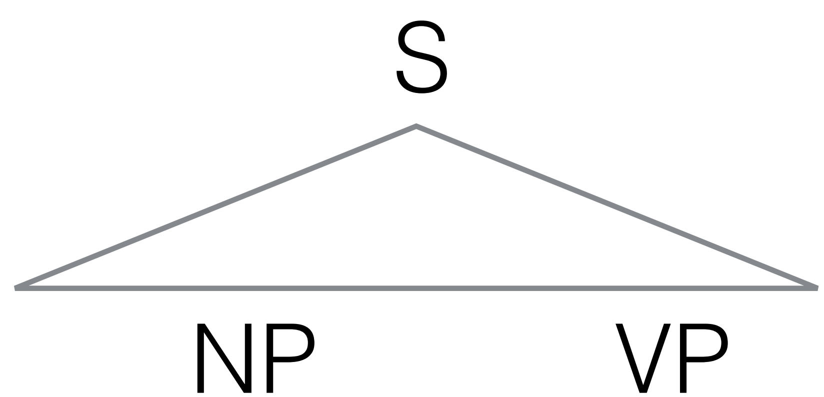
Complete (reduce)
asserts that $\alpha B \Rightarrow^* x_1 \ldots x_j$

Top-down: Predict-Scan

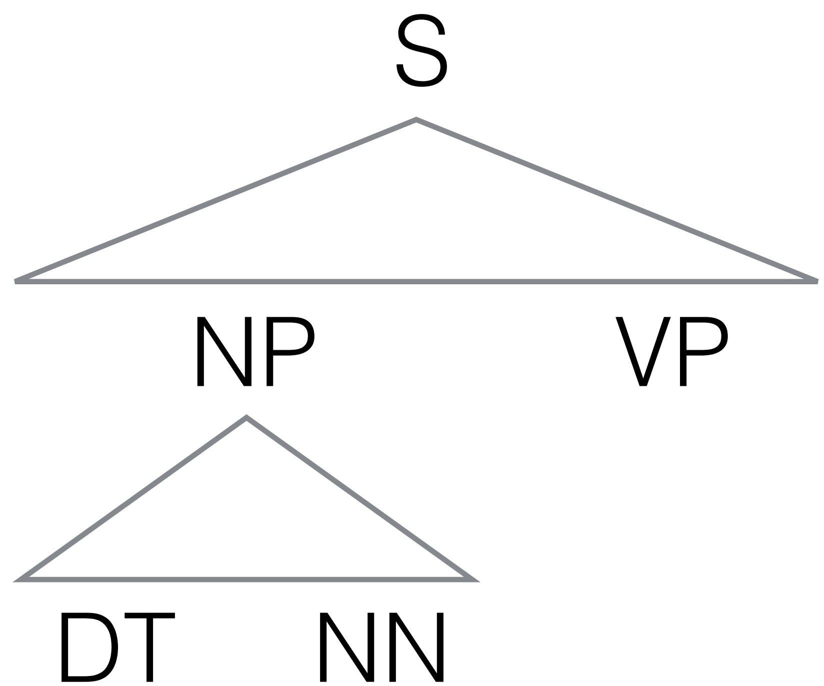
Top-down: Predict-Scan

S

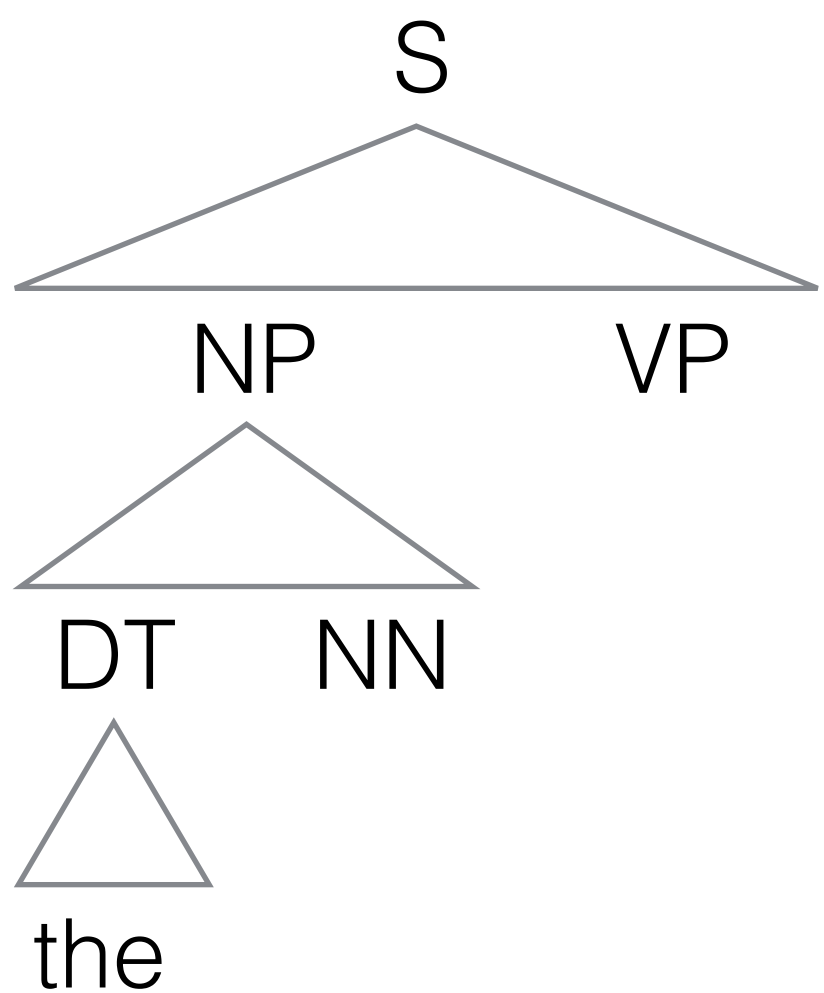
Top-down: Predict-Scan



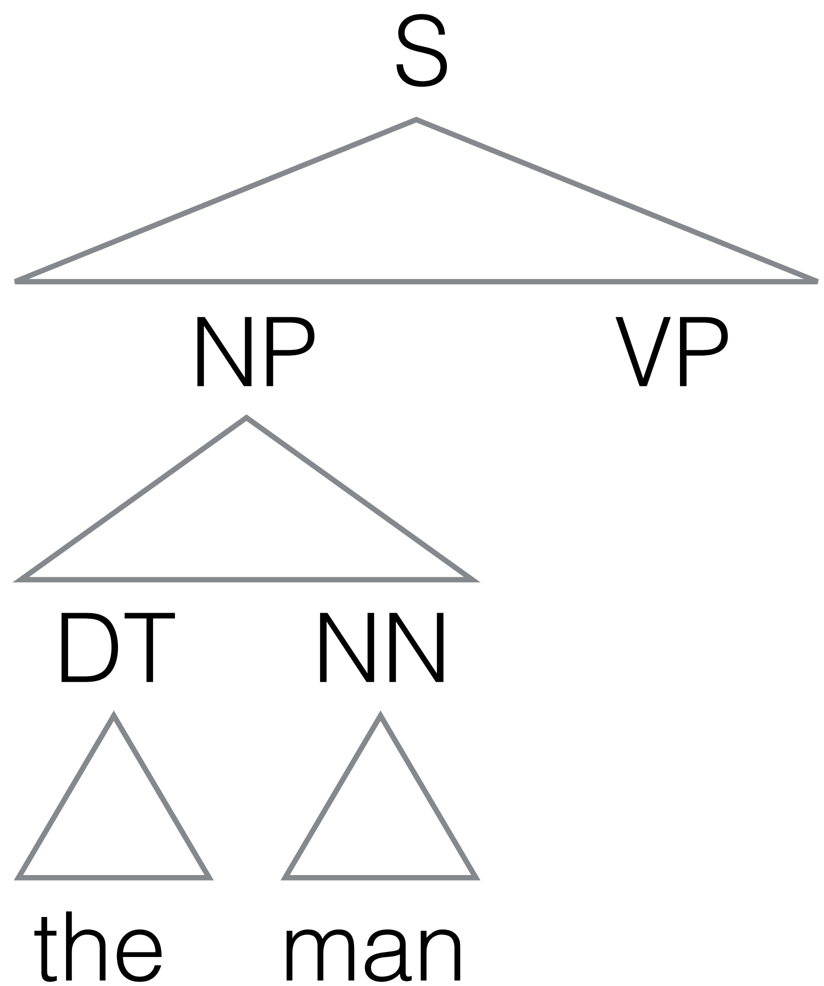
Top-down: Predict-Scan



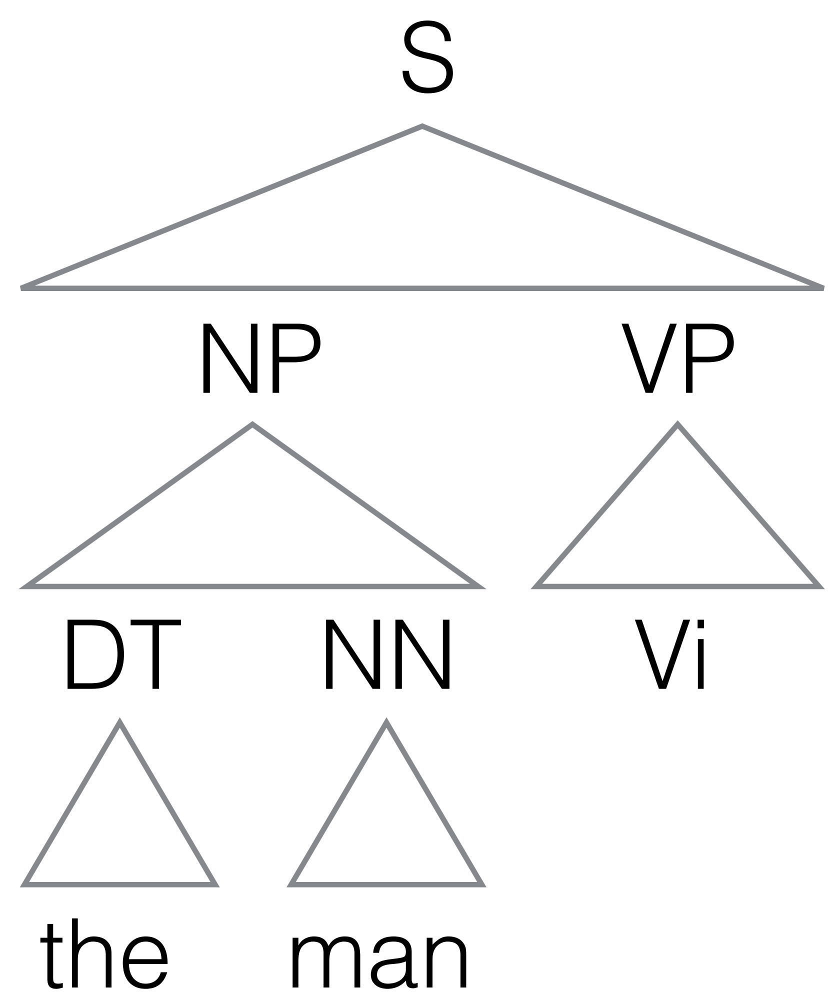
Top-down: Predict-Scan



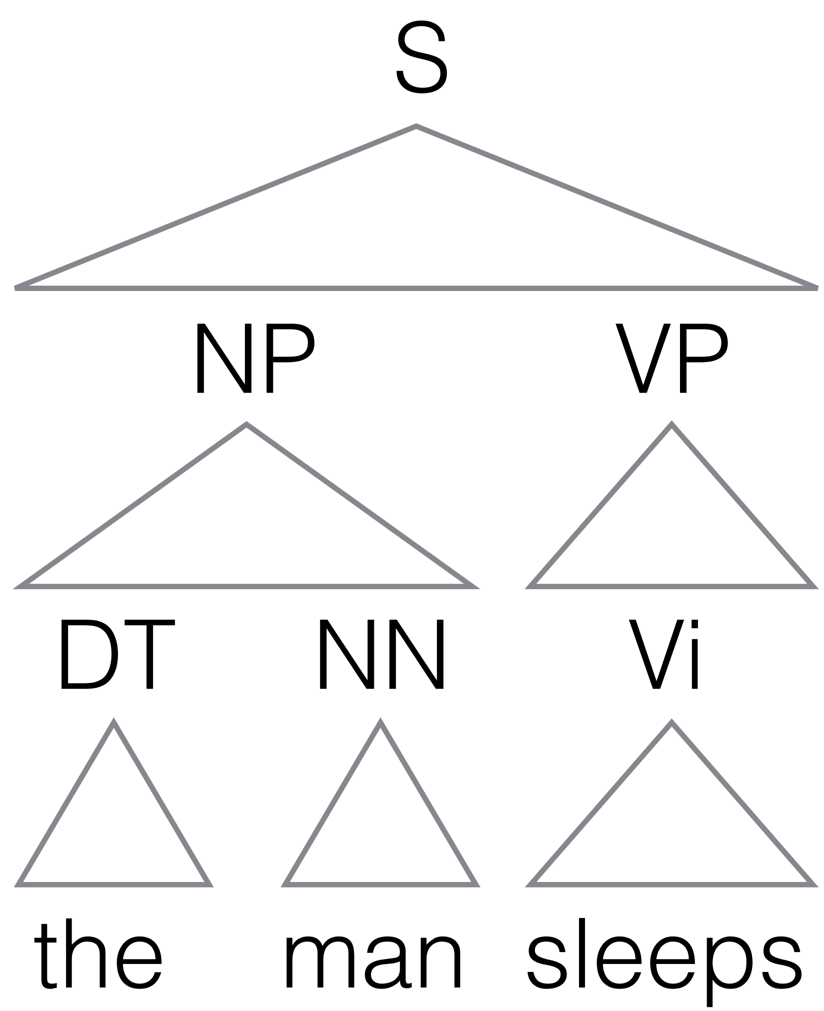
Top-down: Predict-Scan



Top-down: Predict-Scan



Top-down: Predict-Scan



Top-Down Example

Input: *the man sleeps*

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition	Statement	Queue

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition	Statement	Queue
Axiom	1	[• S, 0]	1

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition	Statement	Queue
Axiom	1	[• S, 0]	1

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5
Predict: [5]	NN → man	6	[• man VP, 1]	6

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5
Predict: [5]	NN → man	6	[• man VP, 1]	6

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5
Predict: [5]	NN → man	6	[• man VP, 1]	6
Scan: [6]		7	[• VP, 2]	7

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5
Predict: [5]	NN → man	6	[• man VP, 1]	6
Scan: [6]		7	[• VP, 2]	7

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5
Predict: [5]	NN → man	6	[• man VP, 1]	6
Scan: [6]		7	[• VP, 2]	7

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5
Predict: [5]	NN → man	6	[• man VP, 1]	6
Scan: [6]		7	[• VP, 2]	7
Predict: [7]	VP → Vi	8	[• Vi, 2]	8, 9

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5
Predict: [5]	NN → man	6	[• man VP, 1]	6
Scan: [6]		7	[• VP, 2]	7
Predict: [7]	VP → Vi	8	[• Vi, 2]	8, 9
	VP → Vt NP	9	[• Vt NP, 2]	

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5
Predict: [5]	NN → man	6	[• man VP, 1]	6
Scan: [6]		7	[• VP, 2]	7
Predict: [7]	VP → Vi	8	[• Vi, 2]	8, 9
	VP → Vt NP	9	[• Vt NP, 2]	

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5
Predict: [5]	NN → man	6	[• man VP, 1]	6
Scan: [6]		7	[• VP, 2]	7
Predict: [7]	VP → Vi	8	[• Vi, 2]	8, 9
	VP → Vt NP	9	[• Vt NP, 2]	
Predict: [8]	Vi → sleeps	10	[• sleeps, 2]	9, 10

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5
Predict: [5]	NN → man	6	[• man VP, 1]	6
Scan: [6]		7	[• VP, 2]	7
Predict: [7]	VP → Vi	8	[• Vi, 2]	8, 9
	VP → Vt NP	9	[• Vt NP, 2]	
Predict: [8]	Vi → sleeps	10	[• sleeps, 2]	9, 10

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5
Predict: [5]	NN → man	6	[• man VP, 1]	6
Scan: [6]		7	[• VP, 2]	7
Predict: [7]	VP → Vi	8	[• Vi, 2]	8, 9
	VP → Vt NP	9	[• Vt NP, 2]	
Predict: [8]	Vi → sleeps	10	[• sleeps, 2]	9, 10
[9]				10

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5
Predict: [5]	NN → man	6	[• man VP, 1]	6
Scan: [6]		7	[• VP, 2]	7
Predict: [7]	VP → Vi	8	[• Vi, 2]	8, 9
	VP → Vt NP	9	[• Vt NP, 2]	
Predict: [8]	Vi → sleeps	10	[• sleeps, 2]	9, 10
[9]				10

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5
Predict: [5]	NN → man	6	[• man VP, 1]	6
Scan: [6]		7	[• VP, 2]	7
Predict: [7]	VP → Vi	8	[• Vi, 2]	8, 9
	VP → Vt NP	9	[• Vt NP, 2]	
Predict: [8]	Vi → sleeps	10	[• sleeps, 2]	9, 10
[9]				10
Scan: [10]		11	[•, 3]	11

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down Example

Input: *the man sleeps*

Rule	Condition		Statement	Queue
Axiom		1	[• S, 0]	1
Predict: [1]	S → NP VP	2	[• NP VP, 0]	2
Predict: [2]	NP → DT NN	3	[• DT NN VP, 0]	3
Predict: [3]	DT → the	4	[• the NN VP,0]	4
Scan: [4]		5	[• NN VP,1]	5
Predict: [5]	NN → man	6	[• man VP, 1]	6
Scan: [6]		7	[• VP, 2]	7
Predict: [7]	VP → Vi	8	[• Vi, 2]	8, 9
	VP → Vt NP	9	[• Vt NP, 2]	
Predict: [8]	Vi → sleeps	10	[• sleeps, 2]	9, 10
[9]				10
Scan: [10]		11	[•, 3]	11
GOAL: [11]				∅

S → NP VP
VP → Vi
VP → Vt NP
~~VP → VP PP~~
NP → DT NN
NP → NP PP
PP → IN NP
Vi → sleeps
Vt → saw
NN → man
NN → dog
NN → telescope
DT → the
IN → with

Top-Down recognition

Input: G and x_1 ... x_n

Item form: [•β, j]
asserts that $S \Rightarrow^* x_1 \ldots x_j\ \beta$

Axiom: [•S,0]

Goal: [•,n]

$$\textsc{Scan}\ \frac{[\bullet x_{j+1}\,\beta, j]}{[\bullet\beta, j+1]}$$

Scan
asserts that $S \Rightarrow^* x_1 \ldots x_j\, x_{j+1}\ \beta$

$$\textsc{Predict}\ \frac{[\bullet A\,\beta, j]}{[\bullet\alpha\,\beta, j]}\ A \to \alpha \in \mathcal{R}$$

Predict
asserts that $S \Rightarrow^* x_1 \ldots x_j\, B\ \beta$

Bottom-Up for CNF: CKY

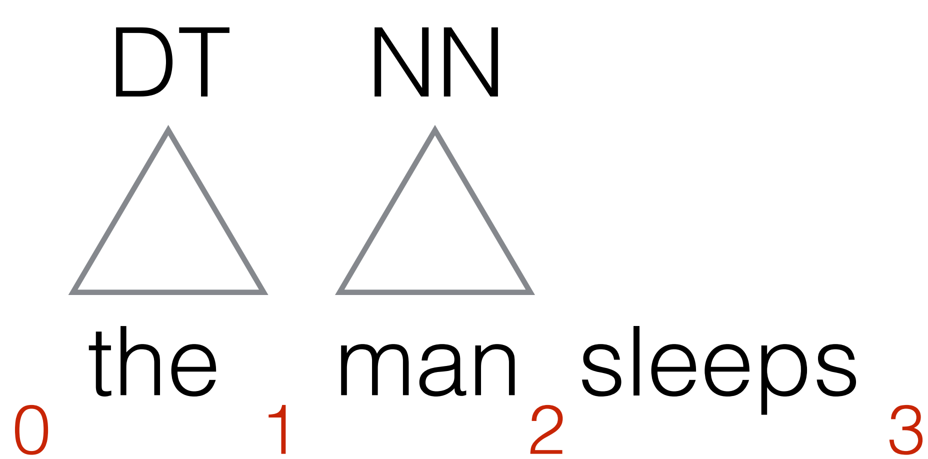
0 1 2 3

Bottom-Up for CNF: CKY

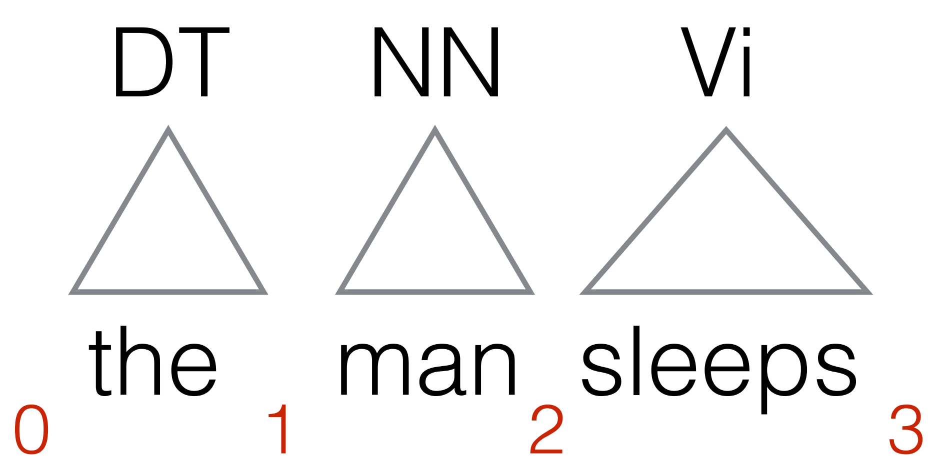
$_0$ the $_1$ man $_2$ sleeps $_3$

Bottom-Up for CNF: CKY

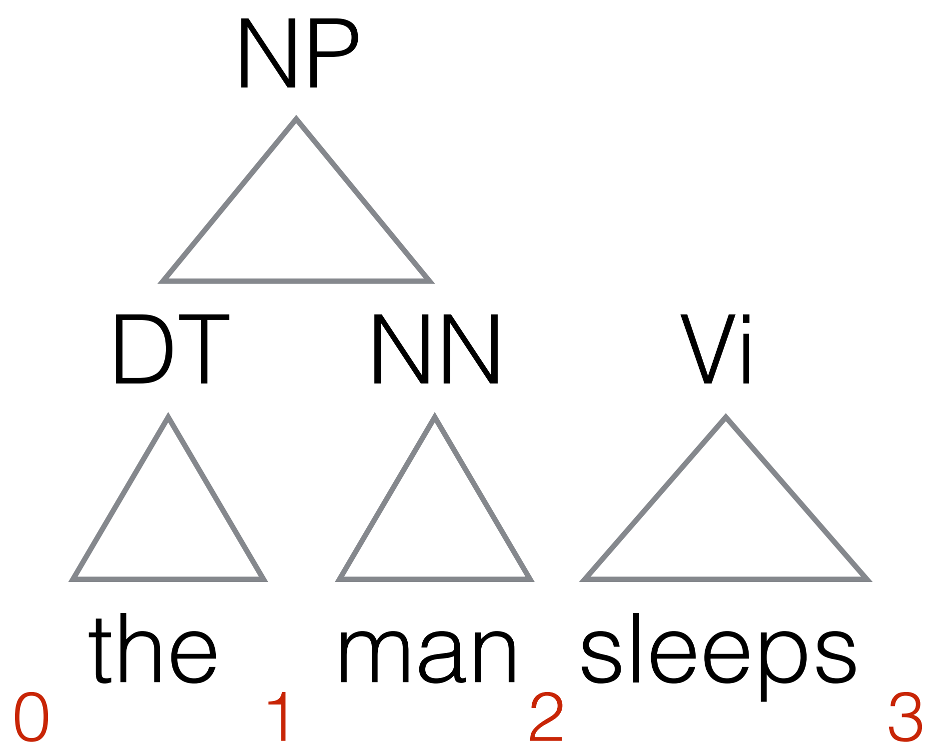
Bottom-Up for CNF: CKY



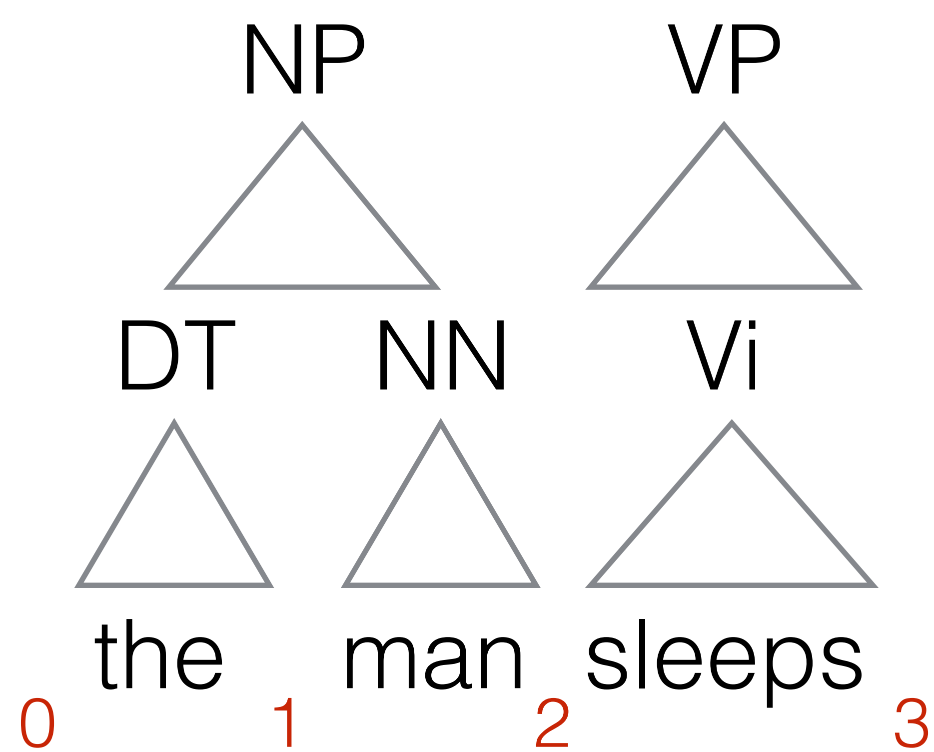
Bottom-Up for CNF: CKY



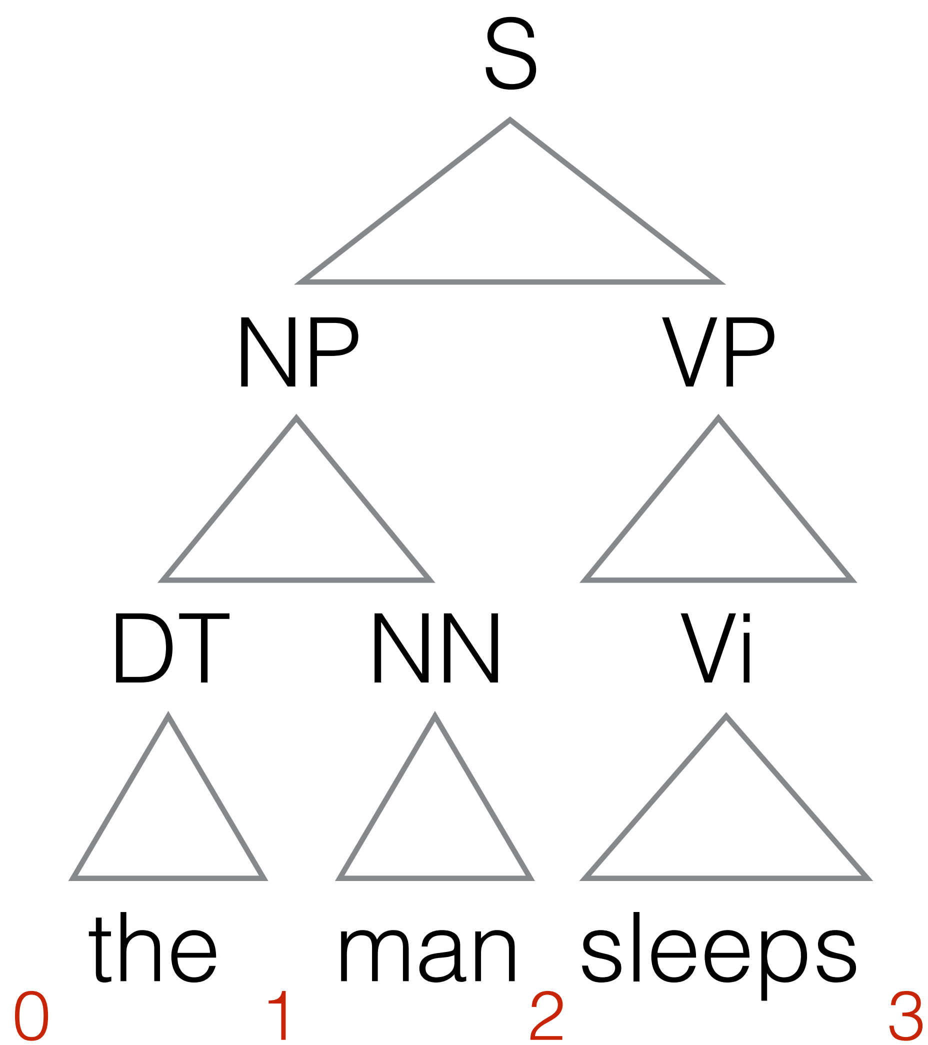
Bottom-Up for CNF: CKY



Bottom-Up for CNF: CKY



Bottom-Up for CNF: CKY



CKY - CNF only

Input: G and $s = x_1 \dots x_n$

Item form: [i, X, j]
asserts that $X \Rightarrow^* x_{i+1} \dots x_j$

Axioms: [i, X, i+1] $X \rightarrow x_i \in \mathscr{R}$

Goal: [0, S, n]

Merge:
asserts that
$x_{i+1} \dots x_k \, x_{k+1} \dots x_j \Rightarrow^* x_{i+1} \dots x_j$

$$\frac{[i, A, k][k, B, j]}{[i, C, j]} \; C \rightarrow A \, B \in \mathcal{R}$$

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition	Statement	Queue	Passive

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition	Statement	Queue	Passive
Axiom	DT → the	1 [0, DT, 1]	1	

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition	Statement		Queue	Passive
Axiom	DT → the	1	[0, DT, 1]	1	
	NN → man	2	[1, NN, 2]	1, 2	

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition	Statement		Queue	Passive
Axiom	DT → the	1	[0, DT, 1]	1	
	NN → man	2	[1, NN, 2]	1, 2	
	Vt → saw	3	[2, Vt, 3]	1, 2, 3	

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition		Statement	Queue	Passive
Axiom	DT → the	1	[0, DT, 1]	1	
	NN → man	2	[1, NN, 2]	1, 2	
	Vt → saw	3	[2, Vt, 3]	1, 2, 3	
	DT → the	4	[3, DT, 4]	1, 2, 3, 4	

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition	Statement		Queue	Passive
Axiom	DT → the	1	[0, DT, 1]	1	
	NN → man	2	[1, NN, 2]	1, 2	
	Vt → saw	3	[2, Vt, 3]	1, 2, 3	
	DT → the	4	[3, DT, 4]	1, 2, 3, 4	
	NN → dog	5	[4, NN, 5]	1, 2, 3, 4, 5	

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition		Statement	Queue	Passive
Axiom	DT → the	1	[0, DT, 1]	1	
	NN → man	2	[1, NN, 2]	1, 2	
	Vt → saw	3	[2, Vt, 3]	1, 2, 3	
	DT → the	4	[3, DT, 4]	1, 2, 3, 4	
	NN → dog	5	[4, NN, 5]	1, 2, 3, 4, 5	
				2, 3, 4, 5	1

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition		Statement	Queue	Passive
Axiom	DT → the	1	[0, DT, 1]	1	
	NN → man	2	[1, NN, 2]	1, 2	
	Vt → saw	3	[2, Vt, 3]	1, 2, 3	
	DT → the	4	[3, DT, 4]	1, 2, 3, 4	
	NN → dog	5	[4, NN, 5]	1, 2, 3, 4, 5	
				2, 3, 4, 5	1
Merge: [1][2]	NP → DT NN	6	[0, NP, 2]	3, 4, 5, 6	2

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition		Statement	Queue	Passive
Axiom	DT → the	1	[0, DT, 1]	1	
	NN → man	2	[1, NN, 2]	1, 2	
	Vt → saw	3	[2, Vt, 3]	1, 2, 3	
	DT → the	4	[3, DT, 4]	1, 2, 3, 4	
	NN → dog	5	[4, NN, 5]	1, 2, 3, 4, 5	
				2, 3, 4, 5	1
Merge: [1][2]	NP → DT NN	6	[0, NP, 2]	3, 4, 5, 6	2
				4, 5, 6	3

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition	Statement		Queue	Passive
Axiom	DT → the	1	[0, DT, 1]	1	
	NN → man	2	[1, NN, 2]	1, 2	
	Vt → saw	3	[2, Vt, 3]	1, 2, 3	
	DT → the	4	[3, DT, 4]	1, 2, 3, 4	
	NN → dog	5	[4, NN, 5]	1, 2, 3, 4, 5	
				2, 3, 4, 5	1
Merge: [1][2]	NP → DT NN	6	[0, NP, 2]	3, 4, 5, 6	2
				4, 5, 6	3
				5, 6	4

CKY Example

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Input: *the man saw the dog*

Rule	Condition	Statement		Queue	Passive
Axiom	DT → the	1	[0, DT, 1]	1	
	NN → man	2	[1, NN, 2]	1, 2	
	Vt → saw	3	[2, Vt, 3]	1, 2, 3	
	DT → the	4	[3, DT, 4]	1, 2, 3, 4	
	NN → dog	5	[4, NN, 5]	1, 2, 3, 4, 5	
				2, 3, 4, 5	1
Merge: [1][2]	NP → DT NN	6	[0, NP, 2]	3, 4, 5, 6	2
				4, 5, 6	3
				5, 6	4
Merge: [4][5]	NP → DT NN	7	[3, NP, 5]	6, 7	5

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition	Statement		Queue	Passive
Axiom	DT → the	1	[0, DT, 1]	1	
	NN → man	2	[1, NN, 2]	1, 2	
	Vt → saw	3	[2, Vt, 3]	1, 2, 3	
	DT → the	4	[3, DT, 4]	1, 2, 3, 4	
	NN → dog	5	[4, NN, 5]	1, 2, 3, 4, 5	
				2, 3, 4, 5	1
Merge: [1][2]	NP → DT NN	6	[0, NP, 2]	3, 4, 5, 6	2
				4, 5, 6	3
				5, 6	4
Merge: [4][5]	NP → DT NN	7	[3, NP, 5]	6, 7	5
				7	6

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition	Statement		Queue	Passive
Axiom	DT → the	1	[0, DT, 1]	1	
	NN → man	2	[1, NN, 2]	1, 2	
	Vt → saw	3	[2, Vt, 3]	1, 2, 3	
	DT → the	4	[3, DT, 4]	1, 2, 3, 4	
	NN → dog	5	[4, NN, 5]	1, 2, 3, 4, 5	
				2, 3, 4, 5	1
Merge: [1][2]	NP → DT NN	6	[0, NP, 2]	3, 4, 5, 6	2
				4, 5, 6	3
				5, 6	4
Merge: [4][5]	NP → DT NN	7	[3, NP, 5]	6, 7	5
				7	6
Merge: [3] [7]	VP → Vt NP	8	[2, VP, 5]	8	7

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition	Statement		Queue	Passive
Axiom	DT → the	1	[0, DT, 1]	1	
	NN → man	2	[1, NN, 2]	1, 2	
	Vt → saw	3	[2, Vt, 3]	1, 2, 3	
	DT → the	4	[3, DT, 4]	1, 2, 3, 4	
	NN → dog	5	[4, NN, 5]	1, 2, 3, 4, 5	
				2, 3, 4, 5	1
Merge: [1][2]	NP → DT NN	6	[0, NP, 2]	3, 4, 5, 6	2
				4, 5, 6	3
				5, 6	4
Merge: [4][5]	NP → DT NN	7	[3, NP, 5]	6, 7	5
				7	6
Merge: [3] [7]	VP → Vt NP	8	[2, VP, 5]	8	7
Merge: [6] [8]	S → NP VP	9	[0, S, 5]	9	8

CKY Example

Input: *the man saw the dog*

S → NP VP	Vi → sleeps
~~VP → Vi~~	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition	Statement		Queue	Passive
Axiom	DT → the	1	[0, DT, 1]	1	
	NN → man	2	[1, NN, 2]	1, 2	
	Vt → saw	3	[2, Vt, 3]	1, 2, 3	
	DT → the	4	[3, DT, 4]	1, 2, 3, 4	
	NN → dog	5	[4, NN, 5]	1, 2, 3, 4, 5	
				2, 3, 4, 5	1
Merge: [1][2]	NP → DT NN	6	[0, NP, 2]	3, 4, 5, 6	2
				4, 5, 6	3
				5, 6	4
Merge: [4][5]	NP → DT NN	7	[3, NP, 5]	6, 7	5
				7	6
Merge: [3] [7]	VP → Vt NP	8	[2, VP, 5]	8	7
Merge: [6] [8]	S → NP VP	9	[0, S, 5]	9	8
GOAL: [9]				∅	9

Rule Segmentation: "Split Points"

$_0S_3 \rightarrow {}_0NP_2\ {}_2VP_3$

$_0NP_2 \rightarrow {}_0DT_1\ {}_1NN_2$

$_2VP_3 \rightarrow {}_2Vi_3$

$_0DT_1 \rightarrow$ the

$_1NN_2 \rightarrow$ man

$_2Vi_3 \rightarrow$ sleeps

0 1 2 3

Rule Segmentation: "Split Points"

$_0$ the $_1$ man $_2$ sleeps $_3$

$_0S_3 \rightarrow {}_0NP_2\ {}_2VP_3$

$_0NP_2 \rightarrow {}_0DT_1\ {}_1NN_2$

$_2VP_3 \rightarrow {}_2Vi_3$

$_0DT_1 \rightarrow$ the

$_1NN_2 \rightarrow$ man

$_2Vi_3 \rightarrow$ sleeps

Rule Segmentation: "Split Points"

${}_0DT_1$

${}_0$ the ${}_1$ man ${}_2$ sleeps ${}_3$

$${}_0S_3 \rightarrow {}_0NP_2\ {}_2VP_3$$

$${}_0NP_2 \rightarrow {}_0DT_1\ {}_1NN_2$$

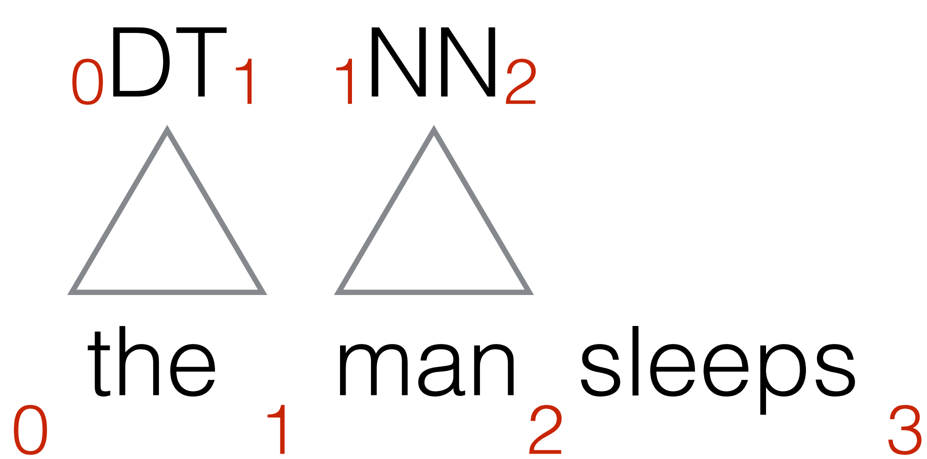
$${}_2VP_3 \rightarrow {}_2Vi_3$$

$${}_0DT_1 \rightarrow \text{the}$$

$${}_1NN_2 \rightarrow \text{man}$$

$${}_2Vi_3 \rightarrow \text{sleeps}$$

Rule Segmentation: "Split Points"



$_0S_3 \rightarrow {_0NP_2}\ {_2VP_3}$

$_0NP_2 \rightarrow {_0DT_1}\ {_1NN_2}$

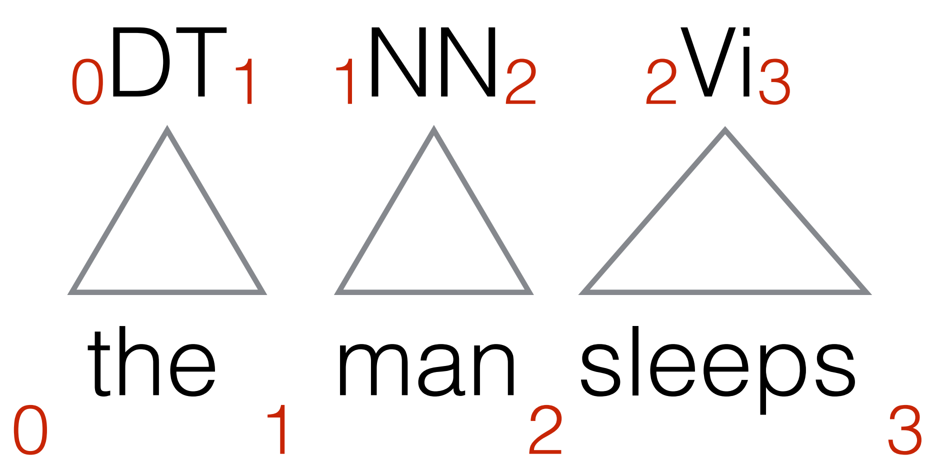
$_2VP_3 \rightarrow {_2Vi_3}$

$_0DT_1 \rightarrow$ the

$_1NN_2 \rightarrow$ man

$_2Vi_3 \rightarrow$ sleeps

Rule Segmentation: "Split Points"



$_0S_3 \rightarrow {}_0NP_2\ {}_2VP_3$

$_0NP_2 \rightarrow {}_0DT_1\ {}_1NN_2$

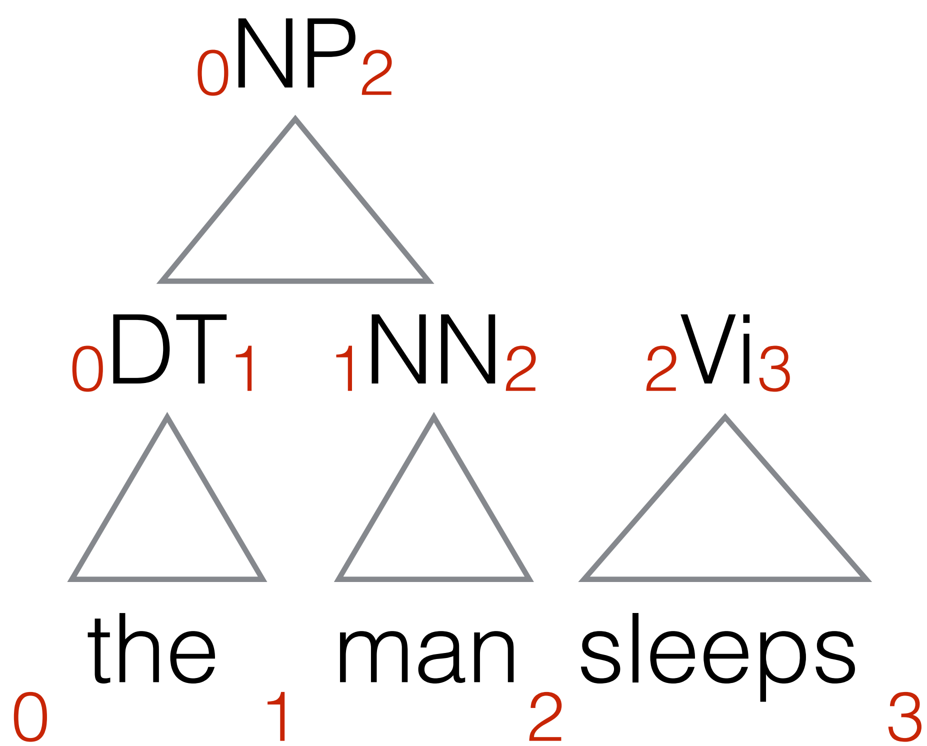
$_2VP_3 \rightarrow {}_2Vi_3$

$_0DT_1 \rightarrow$ the

$_1NN_2 \rightarrow$ man

$_2Vi_3 \rightarrow$ sleeps

Rule Segmentation: "Split Points"



$${}_0S_3 \rightarrow {}_0NP_2\ {}_2VP_3$$

$${}_0NP_2 \rightarrow {}_0DT_1\ {}_1NN_2$$

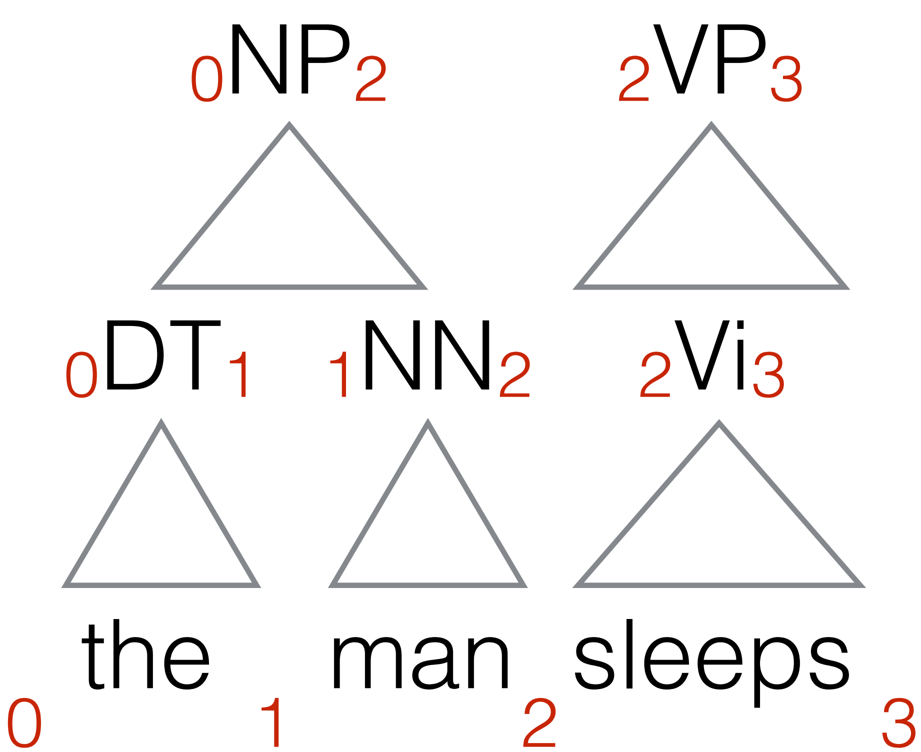
$${}_2VP_3 \rightarrow {}_2Vi_3$$

$${}_0DT_1 \rightarrow \text{the}$$

$${}_1NN_2 \rightarrow \text{man}$$

$${}_2Vi_3 \rightarrow \text{sleeps}$$

Rule Segmentation: "Split Points"



$_0\mathrm{S}_3 \rightarrow {_0\mathrm{NP}_2}\ {_2\mathrm{VP}_3}$

$_0\mathrm{NP}_2 \rightarrow {_0\mathrm{DT}_1}\ {_1\mathrm{NN}_2}$

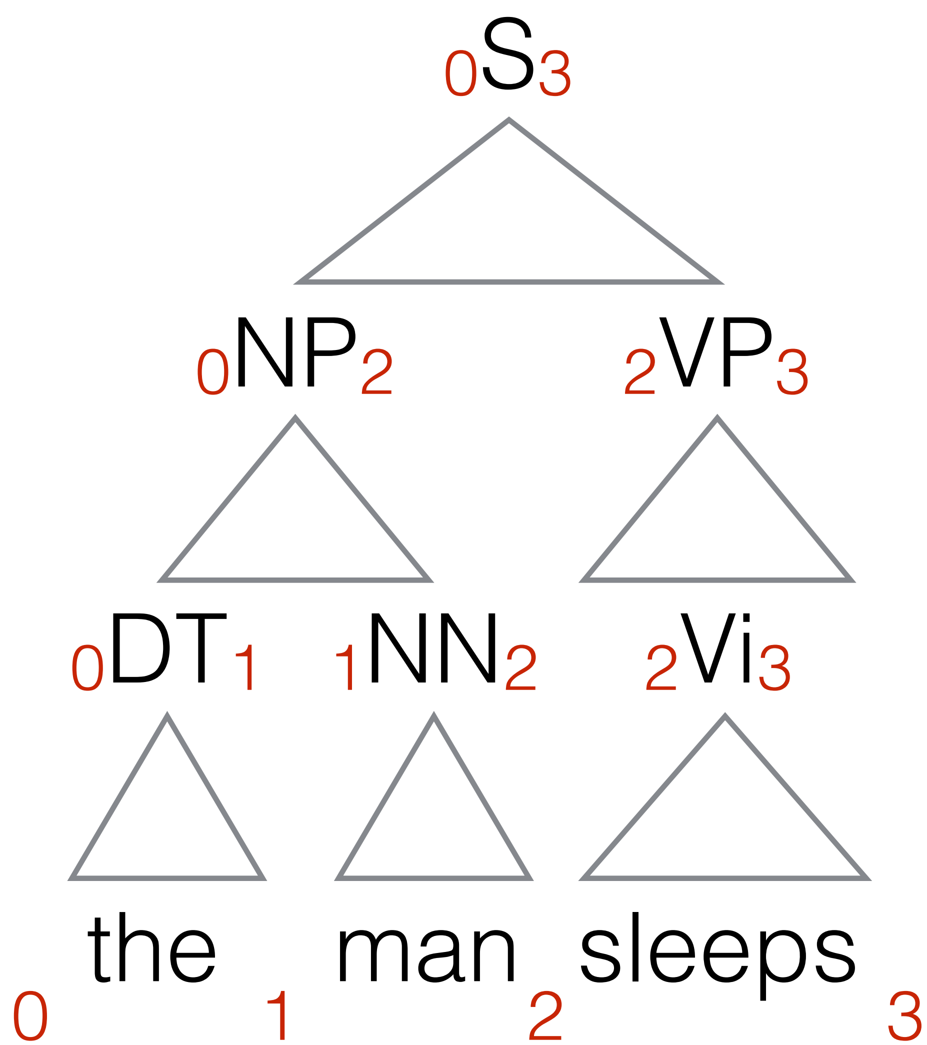
$_2\mathrm{VP}_3 \rightarrow {_2\mathrm{Vi}_3}$

$_0\mathrm{DT}_1 \rightarrow$ the

$_1\mathrm{NN}_2 \rightarrow$ man

$_2\mathrm{Vi}_3 \rightarrow$ sleeps

Rule Segmentation: "Split Points"



$${}_0\mathrm{S}_3 \rightarrow {}_0\mathrm{NP}_2\ {}_2\mathrm{VP}_3$$

$${}_0\mathrm{NP}_2 \rightarrow {}_0\mathrm{DT}_1\ {}_1\mathrm{NN}_2$$

$${}_2\mathrm{VP}_3 \rightarrow {}_2\mathrm{Vi}_3$$

$${}_0\mathrm{DT}_1 \rightarrow \text{the}$$

$${}_1\mathrm{NN}_2 \rightarrow \text{man}$$

$${}_2\mathrm{Vi}_3 \rightarrow \text{sleeps}$$

"Dotted items"

"Dotted items"

Parsing a CNF grammar is easy because we know the shape of rules

"Dotted items"

Parsing a CNF grammar is easy because we know the shape of rules

When that's not the case, we have to **scan rules symbol by symbol** using a general mechanism:

"Dotted items"

Parsing a CNF grammar is easy because we know the shape of rules

When that's not the case, we have to **scan rules symbol by symbol** using a general mechanism:

Item form: $[i, X \rightarrow \alpha_{\blacksquare} \bullet \beta_{\square}, j]$ where $X \rightarrow \alpha\ \beta \in \mathscr{R}$ is a rule

"Dotted items"

Parsing a CNF grammar is easy because we know the shape of rules

When that's not the case, we have to **scan rules symbol by symbol** using a general mechanism:

Item form: $[i, X \to \alpha_{\blacksquare} \bullet \beta_{\square}, j]$ where $X \to \alpha\ \beta \in \mathscr{R}$ is a rule

- In general, we segment rules with respect to the input $x_1 \ldots x_n$

"Dotted items"

Parsing a CNF grammar is easy because we know the shape of rules

When that's not the case, we have to **scan rules symbol by symbol** using a general mechanism:

Item form: $[i, X \rightarrow \alpha_{\blacksquare} \bullet \beta_{\square}, j]$ where $X \rightarrow \alpha\ \beta \in \mathscr{R}$ is a rule

- In general, we segment rules with respect to the input $x_1 \dots x_n$
- The dot represents progress through the rule's right-hand side (RHS)

"Dotted items"

Parsing a CNF grammar is easy because we know the shape of rules

When that's not the case, we have to **scan rules symbol by symbol** using a general mechanism:

Item form: $[i, X \rightarrow \alpha \blacksquare \bullet \beta \square, j]$ where $X \rightarrow \alpha\ \beta \in \mathscr{R}$ is a rule

- In general, we segment rules with respect to the input $x_1 \ldots x_n$
- The dot represents progress through the rule's right-hand side (RHS)
- The prefix α has already been parsed and we are waiting for β

"Dotted items"

Parsing a CNF grammar is easy because we know the shape of rules

When that's not the case, we have to **scan rules symbol by symbol** using a general mechanism:

Item form: $[i, X \rightarrow \alpha_{\blacksquare} \bullet \beta_{\square}, j]$ where $X \rightarrow \alpha\ \beta \in \mathscr{R}$ is a rule

- In general, we segment rules with respect to the input $x_1 \ldots x_n$
- The dot represents progress through the rule's right-hand side (RHS)
- The prefix α has already been parsed and we are waiting for β
- The filled box represents a segmentation of [0 .. j] into $|\alpha|$ adjacent parts

"Dotted items"

Parsing a CNF grammar is easy because we know the shape of rules

When that's not the case, we have to **scan rules symbol by symbol** using a general mechanism:

Item form: $[i, X \to \alpha_\blacksquare \bullet \beta_\square, j]$ where $X \to \alpha\ \beta \in \mathscr{R}$ is a rule

- In general, we segment rules with respect to the input $x_1 \ldots x_n$
- The dot represents progress through the rule's right-hand side (RHS)
- The prefix α has already been parsed and we are waiting for β
- The filled box represents a segmentation of $[0 .. j]$ into $|\alpha|$ adjacent parts
- The empty box has no actual role, it's just a reminder that the segmentation beyond j is unknown

CKY+

Input: G and $s = x_1 \ldots x_n$

Item form: $[i, X \to \alpha_\blacksquare \bullet \beta_\square, j]$
asserts that $X \Rightarrow^* x_{i+1} \ldots x_j \beta$

Axioms: $[i, X \to x_i \bullet \alpha_\square, i+1]$ $\quad X \to x_i\, \alpha \in \mathscr{R}$
$[i, X \to \varepsilon \bullet, i]$ $\quad X \to \varepsilon \in \mathscr{R}$

Goal: $[0, S \to \alpha_\blacksquare \bullet, n]$

Scan

$$\frac{[i, X \to \alpha_\blacksquare \bullet x_{j+1}\, \beta_\square, j]}{[i, X \to \alpha_\blacksquare\, x_{j+1} \bullet \beta_\square, j+1]}$$

Prefix

$$\frac{[i, Y \to \alpha_\blacksquare \bullet, j]}{[i, X \to Y_{i,j} \bullet \beta_\square, j]} X \to Y\, \beta \in \mathcal{R}$$

Complete

$$\frac{[i, X \to \alpha_\blacksquare \bullet Y\, \beta_\square, k][k, Y \to \gamma_\blacksquare \bullet, j]}{[i, X \to \alpha_\blacksquare\, Y_{k,j} \bullet \beta_\square, j]}$$

CKY+ Example

Input: *the man sleeps*

S → NP VP	Vi → sleeps
VP → Vi	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

CKY+ Example

Input: *the man sleeps*

S → NP VP	Vi → sleeps
VP → Vi	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition	Item	Active	Passive

CKY+ Example

Input: *the man sleeps*

S → NP VP	Vi → sleeps
VP → Vi	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition		Item	Active	Passive
Axiom	DT → the	1	[0, DT → the •,1]	1	

CKY+ Example

S → NP VP	Vi → sleeps
VP → Vi	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Input: *the man sleeps*

Rule	Condition		Item	Active	Passive
Axiom	DT → the	1	[0, DT → the •,1]	1	
	NN → man	2	[1, NN → man •, 2]	1, 2	

CKY+ Example

Input: *the man sleeps*

S → NP VP	Vi → sleeps
VP → Vi	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition		Item	Active	Passive
Axiom	DT → the	1	[0, DT → the •,1]	1	
	NN → man	2	[1, NN → man •, 2]	1, 2	
	Vi → sleeps	3	[2, Vi → sleeps •, 3]	1, 2, 3	

CKY+ Example

S → NP VP	Vi → sleeps
VP → Vi	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Input: *the man sleeps*

Rule	Condition		Item	Active	Passive
Axiom	DT → the	1	[0, DT → the •,1]	1	
	NN → man	2	[1, NN → man •, 2]	1, 2	
	Vi → sleeps	3	[2, Vi → sleeps •, 3]	1, 2, 3	
Prefix: [1]	NP → DT NN	4	[0, NP → $DT_{0,1}$ • NN, 2]	2, 3, 4	1

CKY+ Example

S → NP VP	Vi → sleeps
VP → Vi	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Input: *the man sleeps*

Rule	Condition		Item	Active	Passive
Axiom	DT → the	1	[0, DT → the •,1]	1	
	NN → man	2	[1, NN → man •, 2]	1, 2	
	Vi → sleeps	3	[2, Vi → sleeps •, 3]	1, 2, 3	
Prefix: [1]	NP → DT NN	4	[0, NP → $DT_{0,1}$ • NN, 2]	2, 3, 4	1
				3, 4	2

CKY+ Example

Input: *the man sleeps*

S → NP VP	Vi → sleeps
VP → Vi	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition		Item	Active	Passive
Axiom	DT → the	1	[0, DT → the •,1]	1	
	NN → man	2	[1, NN → man •, 2]	1, 2	
	Vi → sleeps	3	[2, Vi → sleeps •, 3]	1, 2, 3	
Prefix: [1]	NP → DT NN	4	[0, NP → $DT_{0,1}$ • NN, 2]	2, 3, 4	1
				3, 4	2
Prefix: [3]	VP → Vi	5	[2, VP → $Vi_{2,3}$ •, 3]	4, 5	3

CKY+ Example

S → NP VP	Vi → sleeps
VP → Vi	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Input: *the man sleeps*

Rule	Condition		Item	Active	Passive
Axiom	DT → the	1	[0, DT → the •,1]	1	
	NN → man	2	[1, NN → man •, 2]	1, 2	
	Vi → sleeps	3	[2, Vi → sleeps •, 3]	1, 2, 3	
Prefix: [1]	NP → DT NN	4	[0, NP → $\text{DT}_{0,1}$ • NN, 2]	2, 3, 4	1
				3, 4	2
Prefix: [3]	VP → Vi	5	[2, VP → $\text{Vi}_{2,3}$ •, 3]	4, 5	3
Complete: [4] [2]		6	[0, NP → $\text{DT}_{0,1}$ $\text{NN}_{1,2}$ •, 2]	5, 6	4

CKY+ Example

Input: *the man sleeps*

S → NP VP	Vi → sleeps
VP → Vi	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition		Item	Active	Passive
Axiom	DT → the	1	[0, DT → the •,1]	1	
	NN → man	2	[1, NN → man •, 2]	1, 2	
	Vi → sleeps	3	[2, Vi → sleeps •, 3]	1, 2, 3	
Prefix: [1]	NP → DT NN	4	[0, NP → $DT_{0,1}$ • NN, 2]	2, 3, 4	1
				3, 4	2
Prefix: [3]	VP → Vi	5	[2, VP → $Vi_{2,3}$ •, 3]	4, 5	3
Complete: [4] [2]		6	[0, NP → $DT_{0,1}$ $NN_{1,2}$ •, 2]	5, 6	4
				6	5

CKY+ Example

Input: *the man sleeps*

S → NP VP	Vi → sleeps
VP → Vi	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition		Item	Active	Passive
Axiom	DT → the	1	[0, DT → the •,1]	1	
	NN → man	2	[1, NN → man •, 2]	1, 2	
	Vi → sleeps	3	[2, Vi → sleeps •, 3]	1, 2, 3	
Prefix: [1]	NP → DT NN	4	[0, NP → $DT_{0,1}$ • NN, 2]	2, 3, 4	1
				3, 4	2
Prefix: [3]	VP → Vi	5	[2, VP → $Vi_{2,3}$ •, 3]	4, 5	3
Complete: [4] [2]		6	[0, NP → $DT_{0,1}$ $NN_{1,2}$ •, 2]	5, 6	4
				6	5
Prefix: [6]	S → NP VP	7	[0, S → $NP_{0,2}$ • VP, 2]	7	6

CKY+ Example

Input: *the man sleeps*

S → NP VP	Vi → sleeps
VP → Vi	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Rule	Condition		Item	Active	Passive
Axiom	DT → the	1	[0, DT → the •,1]	1	
	NN → man	2	[1, NN → man •, 2]	1, 2	
	Vi → sleeps	3	[2, Vi → sleeps •, 3]	1, 2, 3	
Prefix: [1]	NP → DT NN	4	[0, NP → $DT_{0,1}$ • NN, 2]	2, 3, 4	1
				3, 4	2
Prefix: [3]	VP → Vi	5	[2, VP → $Vi_{2,3}$ •, 3]	4, 5	3
Complete: [4] [2]		6	[0, NP → $DT_{0,1}$ $NN_{1,2}$ •, 2]	5, 6	4
				6	5
Prefix: [6]	S → NP VP	7	[0, S → $NP_{0,2}$ • VP, 2]	7	6
Complete: [7] [5]		8	[0, S → $NP_{0,2}$ $VP_{2,3}$ •, 3]	8	7

CKY+ Example

S → NP VP	Vi → sleeps
VP → Vi	Vt → saw
VP → Vt NP	NN → man
VP → VP PP	NN → dog
NP → DT NN	NN → telescope
NP → NP PP	DT → the
PP → IN NP	IN → with

Input: *the man sleeps*

Rule	Condition		Item	Active	Passive
Axiom	DT → the	1	[0, DT → the •,1]	1	
	NN → man	2	[1, NN → man •, 2]	1, 2	
	Vi → sleeps	3	[2, Vi → sleeps •, 3]	1, 2, 3	
Prefix: [1]	NP → DT NN	4	[0, NP → $DT_{0,1}$ • NN, 2]	2, 3, 4	1
				3, 4	2
Prefix: [3]	VP → Vi	5	[2, VP → $Vi_{2,3}$ •, 3]	4, 5	3
Complete: [4] [2]		6	[0, NP → $DT_{0,1}$ $NN_{1,2}$ •, 2]	5, 6	4
				6	5
Prefix: [6]	S → NP VP	7	[0, S → $NP_{0,2}$ • VP, 2]	7	6
Complete: [7] [5]		8	[0, S → $NP_{0,2}$ $VP_{2,3}$ •, 3]	8	7
GOAL: [8]				∅	

Correctness of Parsing Strategy

Soundness: if a goal item is proven for α

- then $\omega \in L(G)$

Completeness: if $\alpha \in L(G)$

- then a goal item can be proven for α

Parse Forest

Efficient representation of the whole space $T_G(\omega)$

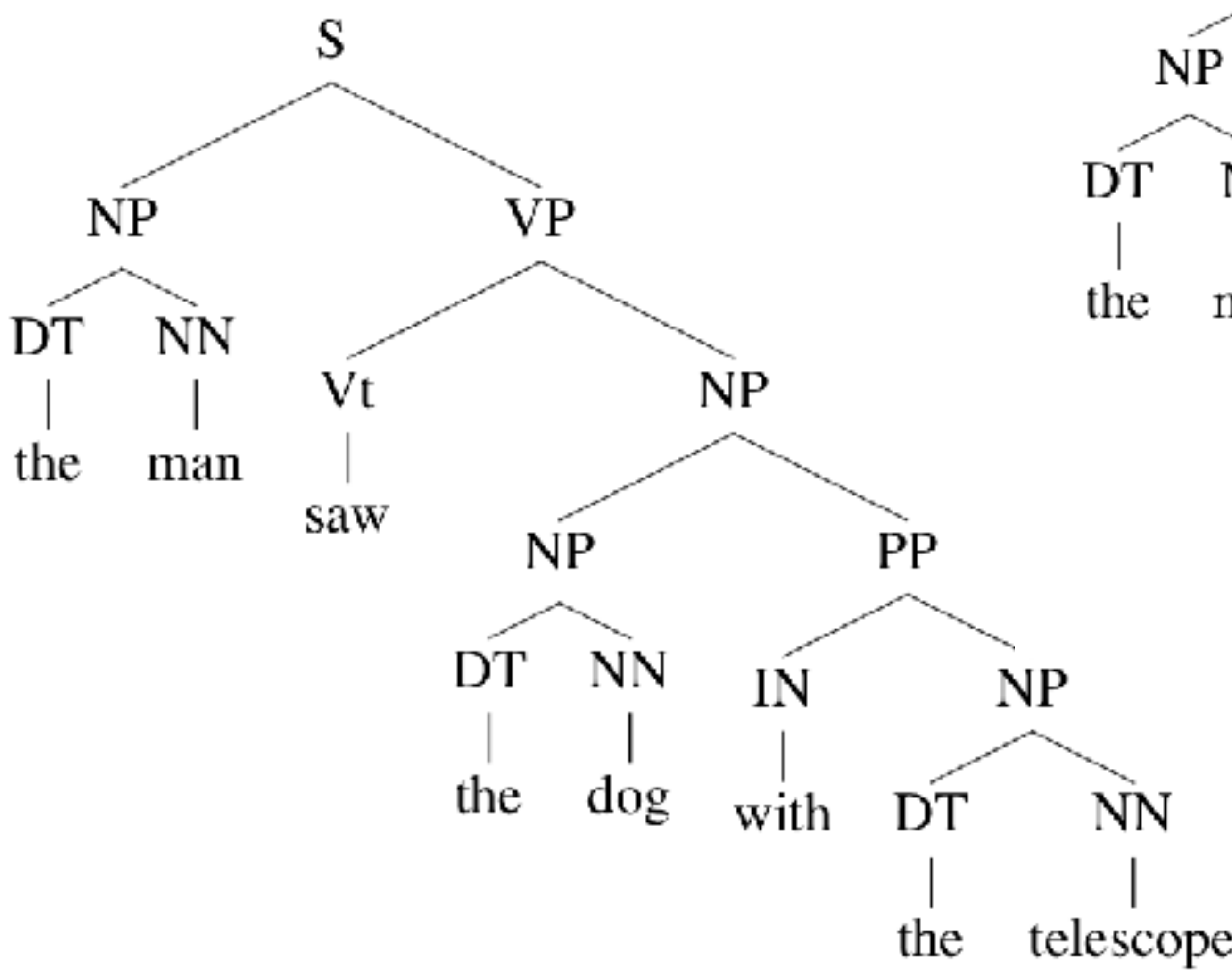
- each and every possible tree yielding ω

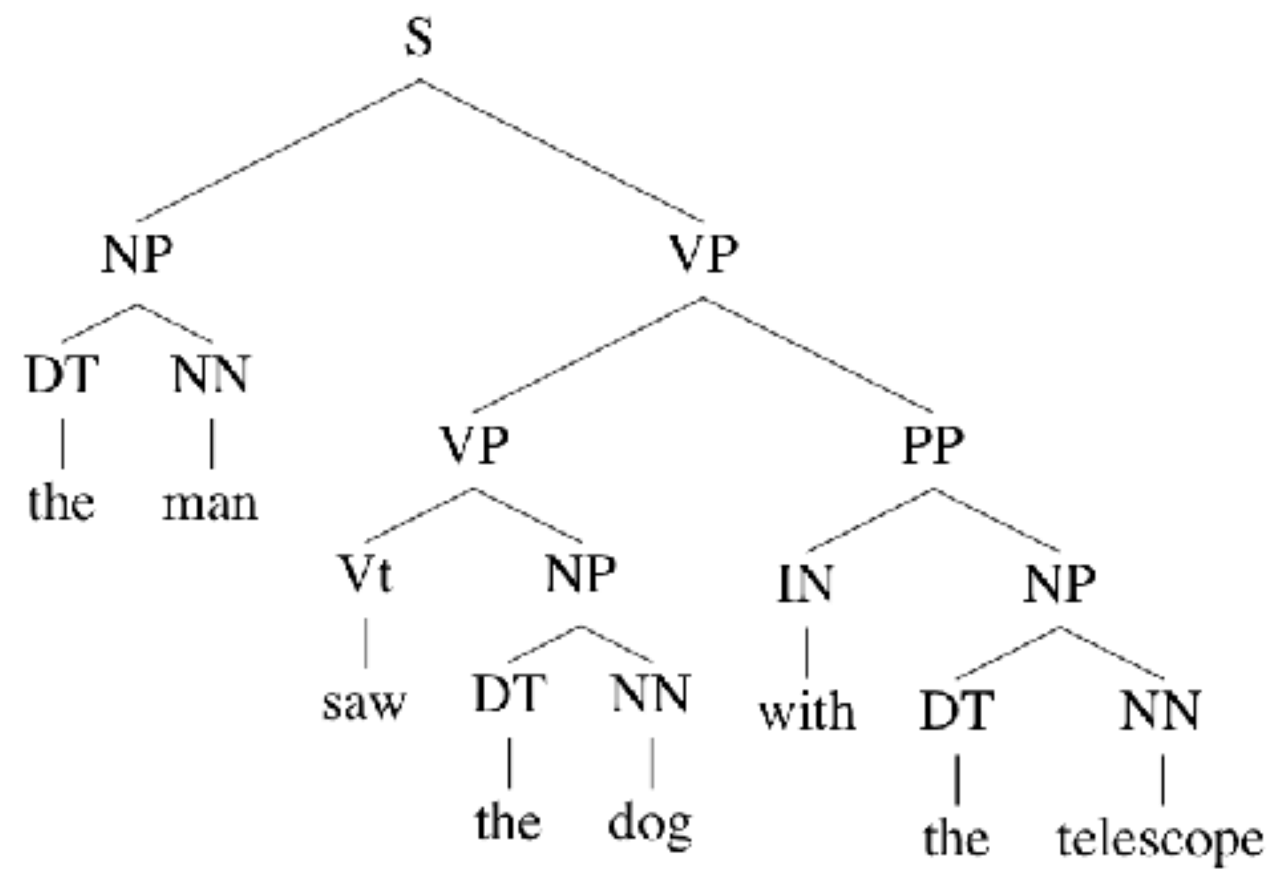
We must be able to represent partial derivations

- including alternative ones

Ambiguity

Some strings may have more than one derivation in G





Dealing with Ambiguity

Statistical model: weight steps in a derivation

- induces a partial ordering over derivations
- can be used to make a decision
  - e.g. best tree under the model

Probabilistic CFG

CFG extended with parameters $0 \leq \theta_r \leq 1$

- where $r \in \mathscr{R}$ and

$$\sum_{\alpha : X \to \alpha \in R} \theta_{X \to \alpha} = 1$$

Probabilistic CFG

Distribution over trees and their yields

$$P_{DS|NM}(R_1^m = r_1^m, X_1^n = \text{yield}(r_1^m)|n, m)$$
$$= \prod_{i=1}^{m} \theta_{r_i} = \prod_{i=1}^{m} \theta_{v_i \to \beta_i}$$
$$\text{where } r_i \text{ corresponds to } v_i \to \beta_i$$

Joint Distribution

${}_0S_3 \rightarrow {}_0NP_2\ {}_2VP_3$

${}_0NP_2 \rightarrow {}_0DT_1\ {}_1NN_2$

${}_2VP_3 \rightarrow {}_2Vi_3$

${}_0DT_1 \rightarrow$ the

${}_1NN_2 \rightarrow$ man

${}_2Vi_3 \rightarrow$ sleeps

Joint Distribution

${}_0S_3 \rightarrow {}_0NP_2\ {}_2VP_3$

${}_0NP_2 \rightarrow {}_0DT_1\ {}_1NN_2$

${}_2VP_3 \rightarrow {}_2Vi_3$

${}_0DT_1 \rightarrow$ the

${}_1NN_2 \rightarrow$ man

${}_2Vi_3 \rightarrow$ sleeps

the man sleeps

Joint Distribution

$_0S_3 \rightarrow {}_0NP_2\ {}_2VP_3$

$_0NP_2 \rightarrow {}_0DT_1\ {}_1NN_2$

$_2VP_3 \rightarrow {}_2Vi_3$

$_0DT_1 \rightarrow$ the

$_1NN_2 \rightarrow$ man

$_2Vi_3 \rightarrow$ sleeps

Joint Distribution

${}_0\mathrm{S}_3 \rightarrow {}_0\mathrm{NP}_2\ {}_2\mathrm{VP}_3$

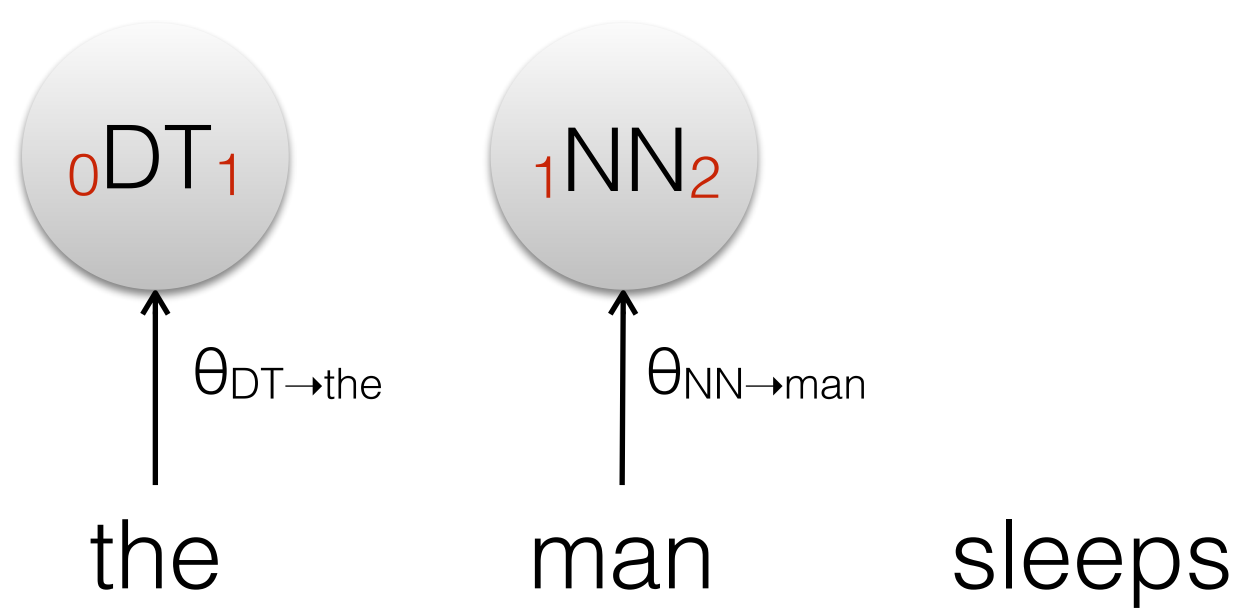
${}_0\mathrm{NP}_2 \rightarrow {}_0\mathrm{DT}_1\ {}_1\mathrm{NN}_2$

${}_2\mathrm{VP}_3 \rightarrow {}_2\mathrm{Vi}_3$

${}_0\mathrm{DT}_1 \rightarrow \mathrm{the}$

${}_1\mathrm{NN}_2 \rightarrow \mathrm{man}$

${}_2\mathrm{Vi}_3 \rightarrow \mathrm{sleeps}$



Joint Distribution

${}_0S_3 \rightarrow {}_0NP_2\ {}_2VP_3$

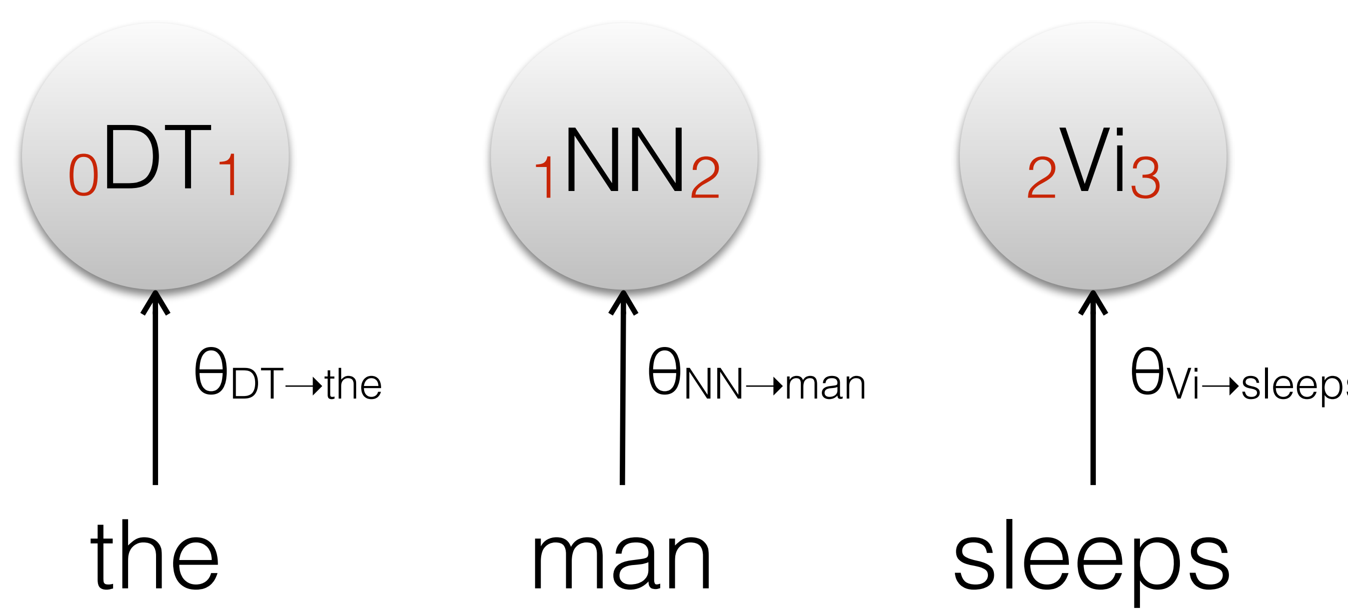
${}_0NP_2 \rightarrow {}_0DT_1\ {}_1NN_2$

${}_2VP_3 \rightarrow {}_2Vi_3$

${}_0DT_1 \rightarrow \text{the}$

${}_1NN_2 \rightarrow \text{man}$

${}_2Vi_3 \rightarrow \text{sleeps}$



Joint Distribution

$_0S_3 \rightarrow {}_0NP_2 \; {}_2VP_3$

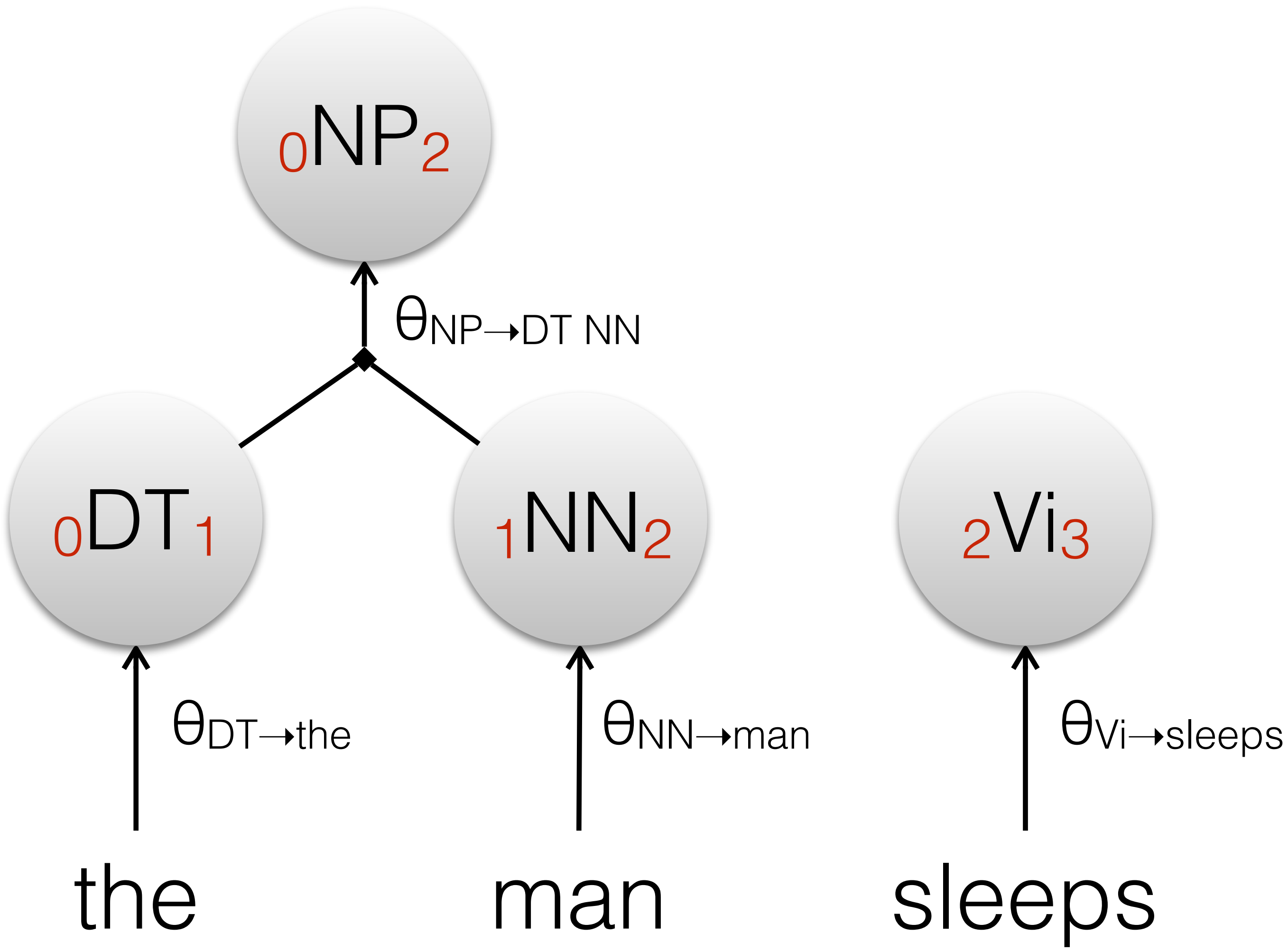
$_0NP_2 \rightarrow {}_0DT_1 \; {}_1NN_2$

$_2VP_3 \rightarrow {}_2Vi_3$

$_0DT_1 \rightarrow$ the

$_1NN_2 \rightarrow$ man

$_2Vi_3 \rightarrow$ sleeps



Joint Distribution

$_0S_3 \rightarrow {}_0NP_2\ {}_2VP_3$

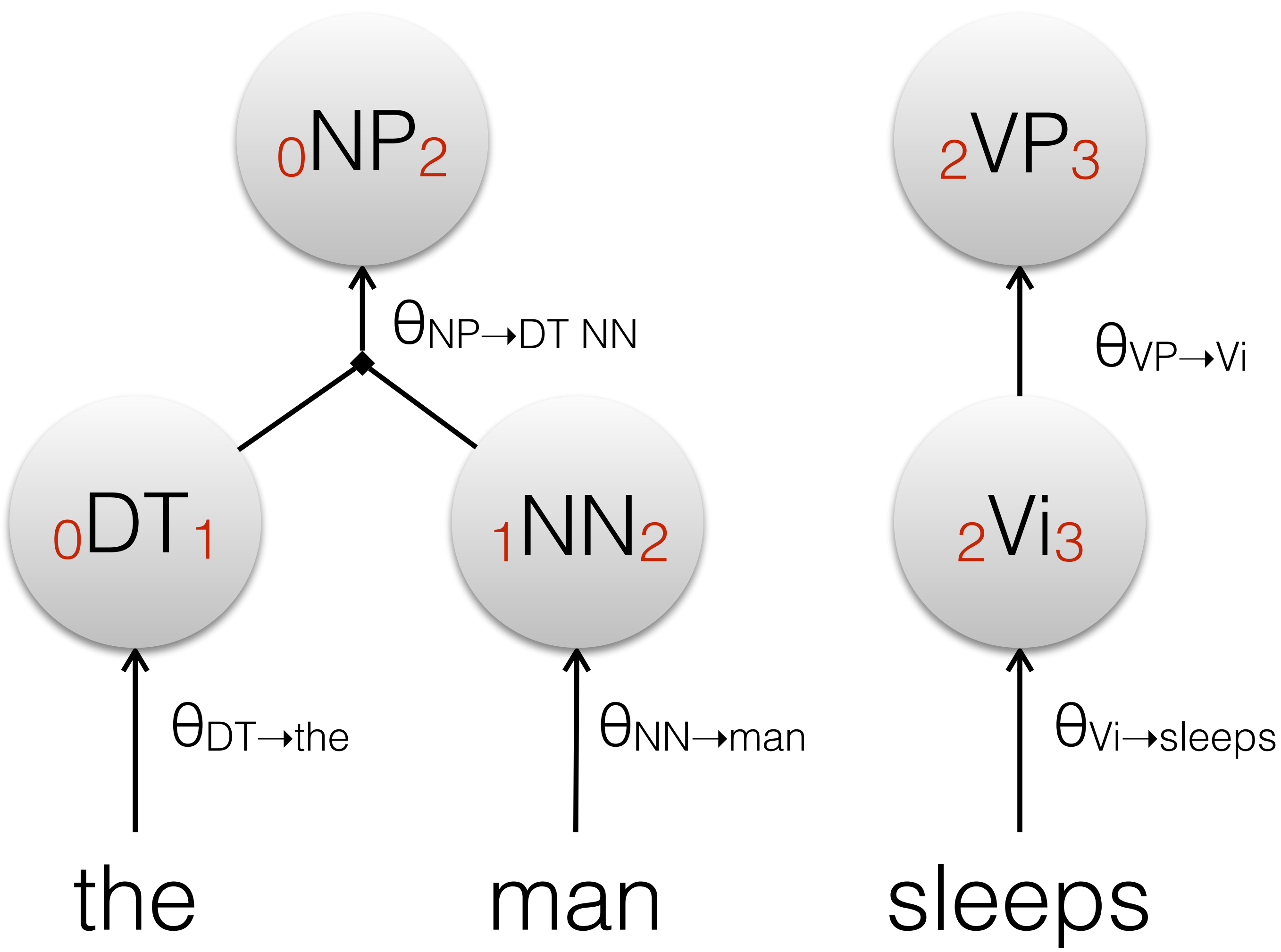
$_0NP_2 \rightarrow {}_0DT_1\ {}_1NN_2$

$_2VP_3 \rightarrow {}_2Vi_3$

$_0DT_1 \rightarrow$ the

$_1NN_2 \rightarrow$ man

$_2Vi_3 \rightarrow$ sleeps



Joint Distribution

$_0S_3 \rightarrow {_0NP_2}\ {_2VP_3}$

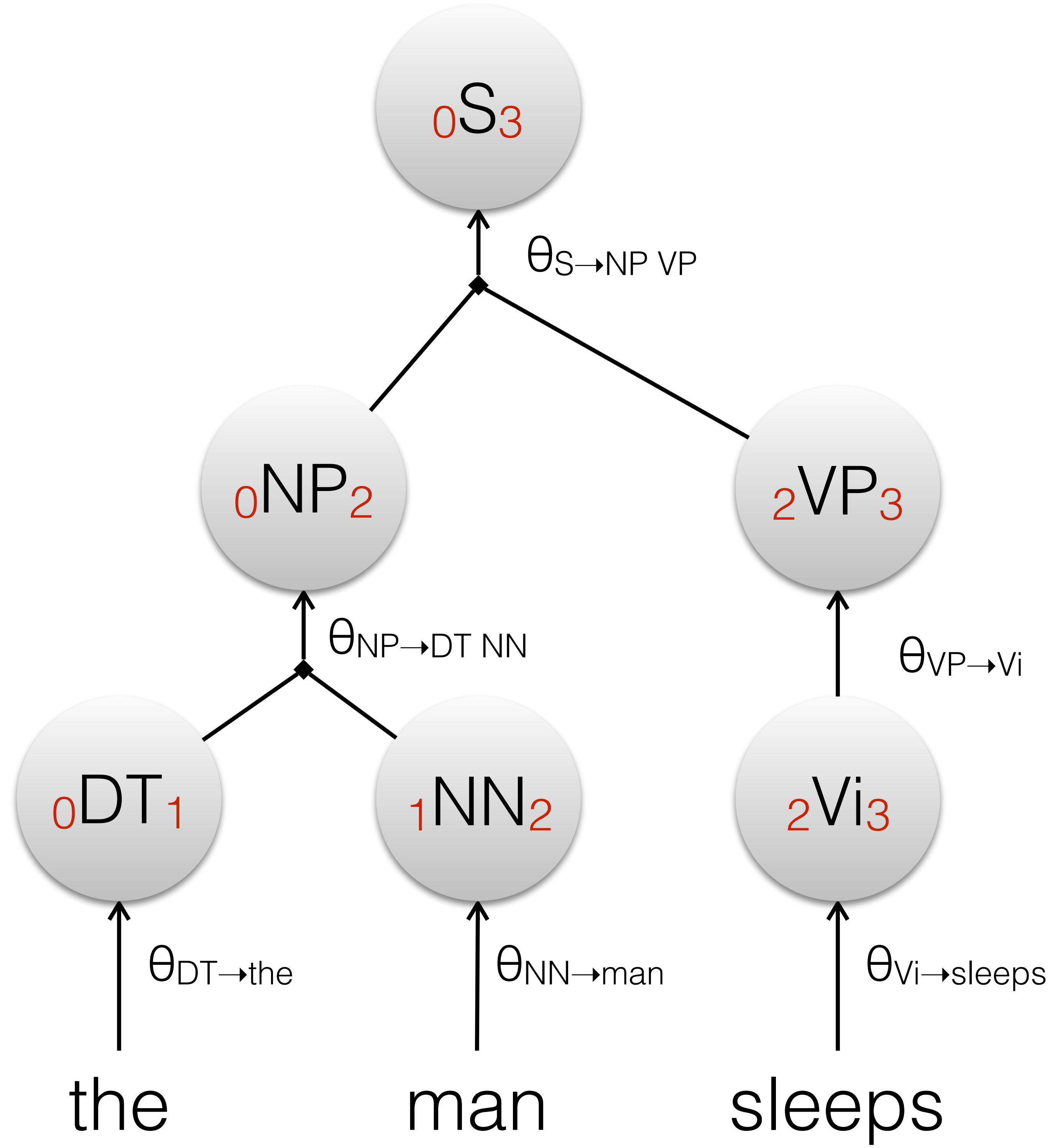
$_0NP_2 \rightarrow {_0DT_1}\ {_1NN_2}$

$_2VP_3 \rightarrow {_2Vi_3}$

$_0DT_1 \rightarrow \text{the}$

$_1NN_2 \rightarrow \text{man}$

$_2Vi_3 \rightarrow \text{sleeps}$



Ambiguity

Ambiguity

the man

saw

the dog

with the telescope

Ambiguity

${}_{0}\mathrm{DT}_{1}$

$\theta_{\mathrm{DT}\rightarrow\mathrm{the}}$

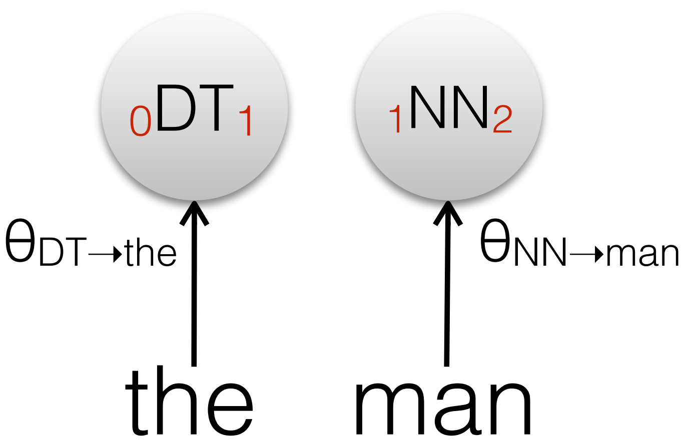
the man

saw

the dog

with the telescope

Ambiguity

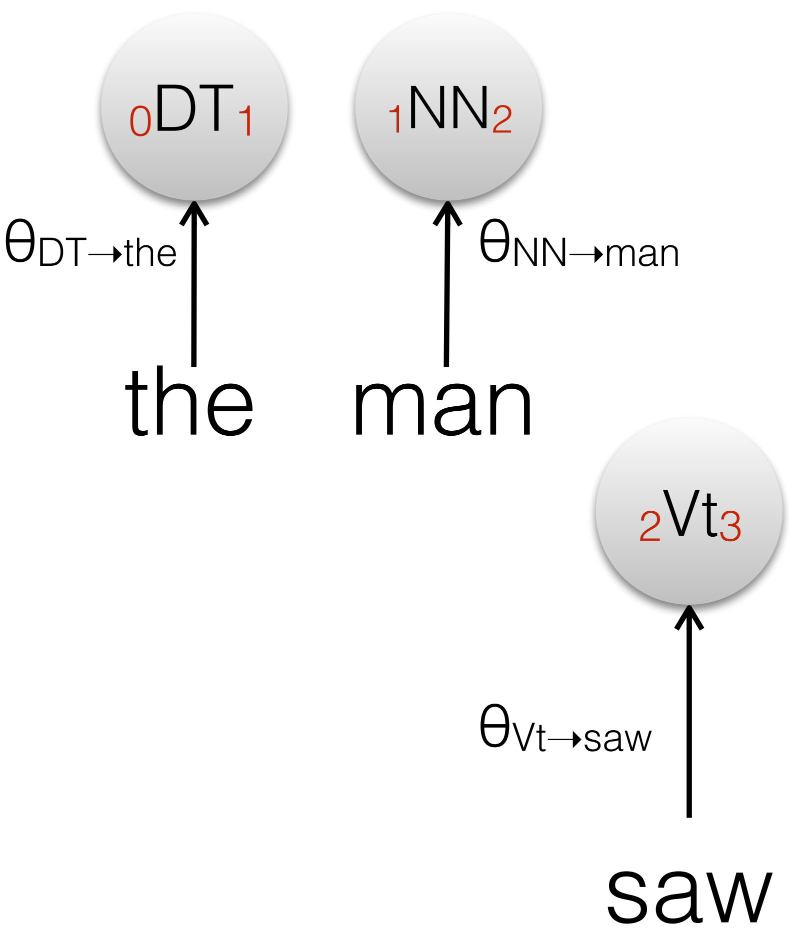


saw

the dog

with the telescope

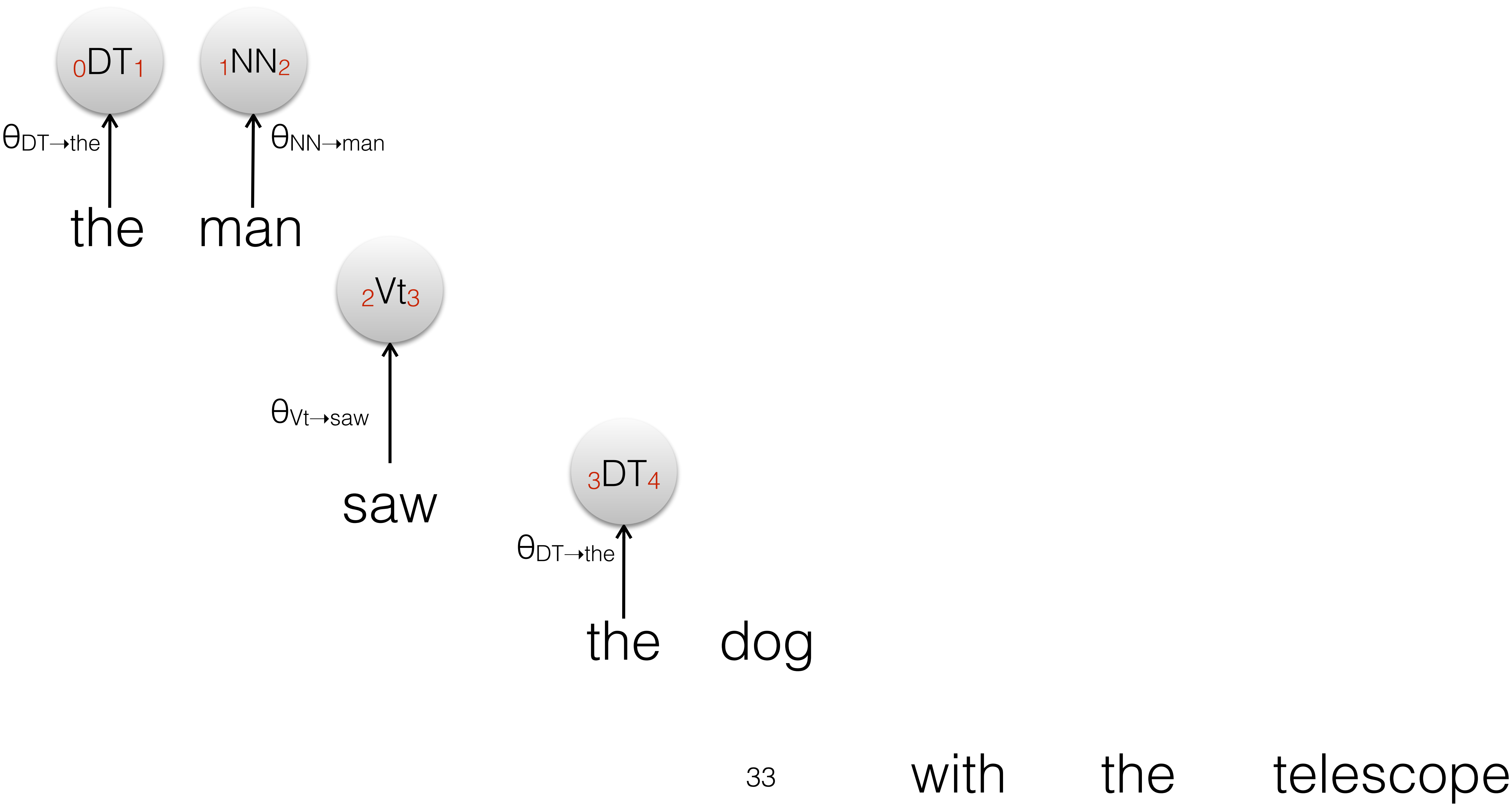
Ambiguity



the dog

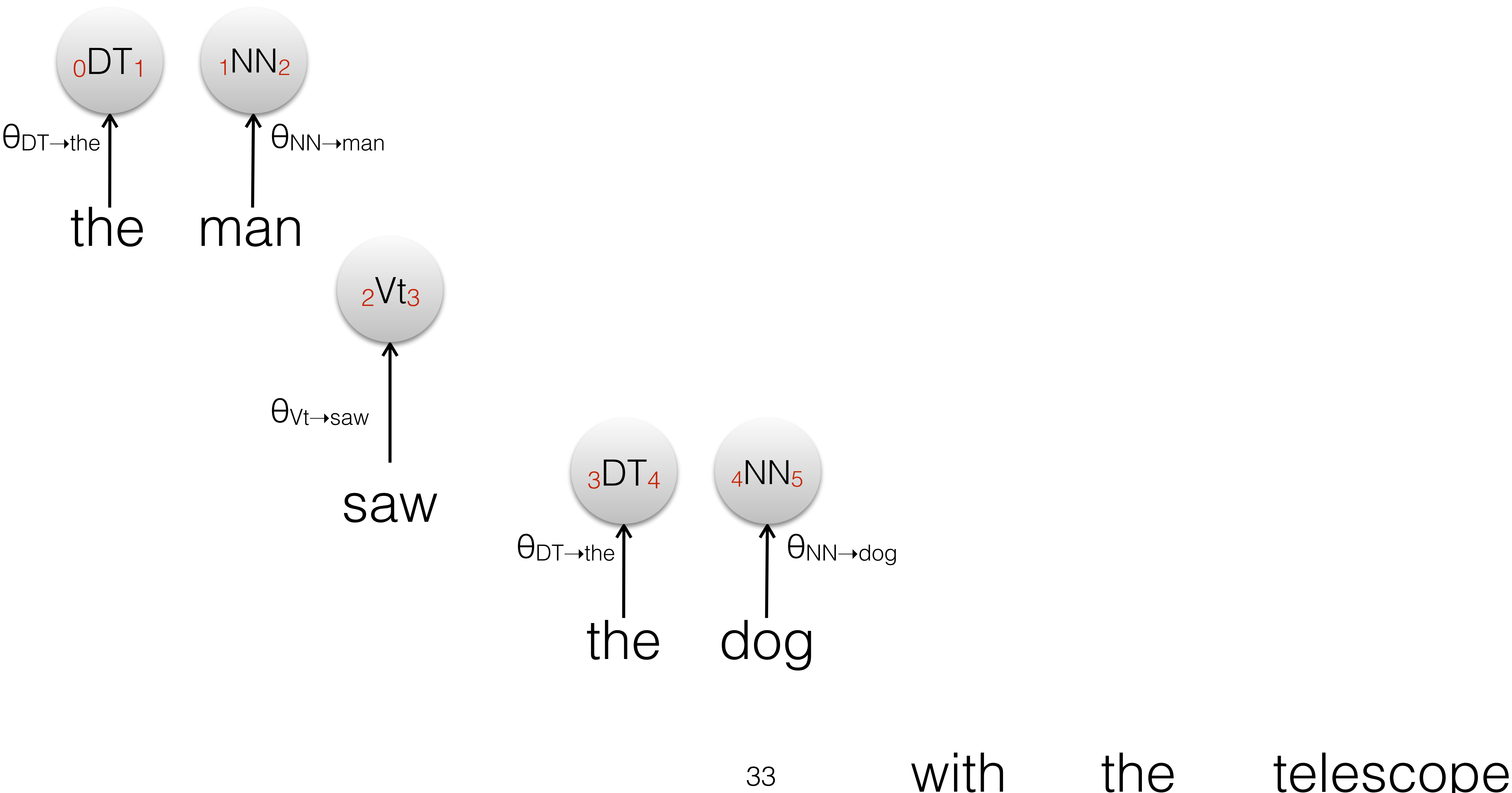
with the telescope

Ambiguity

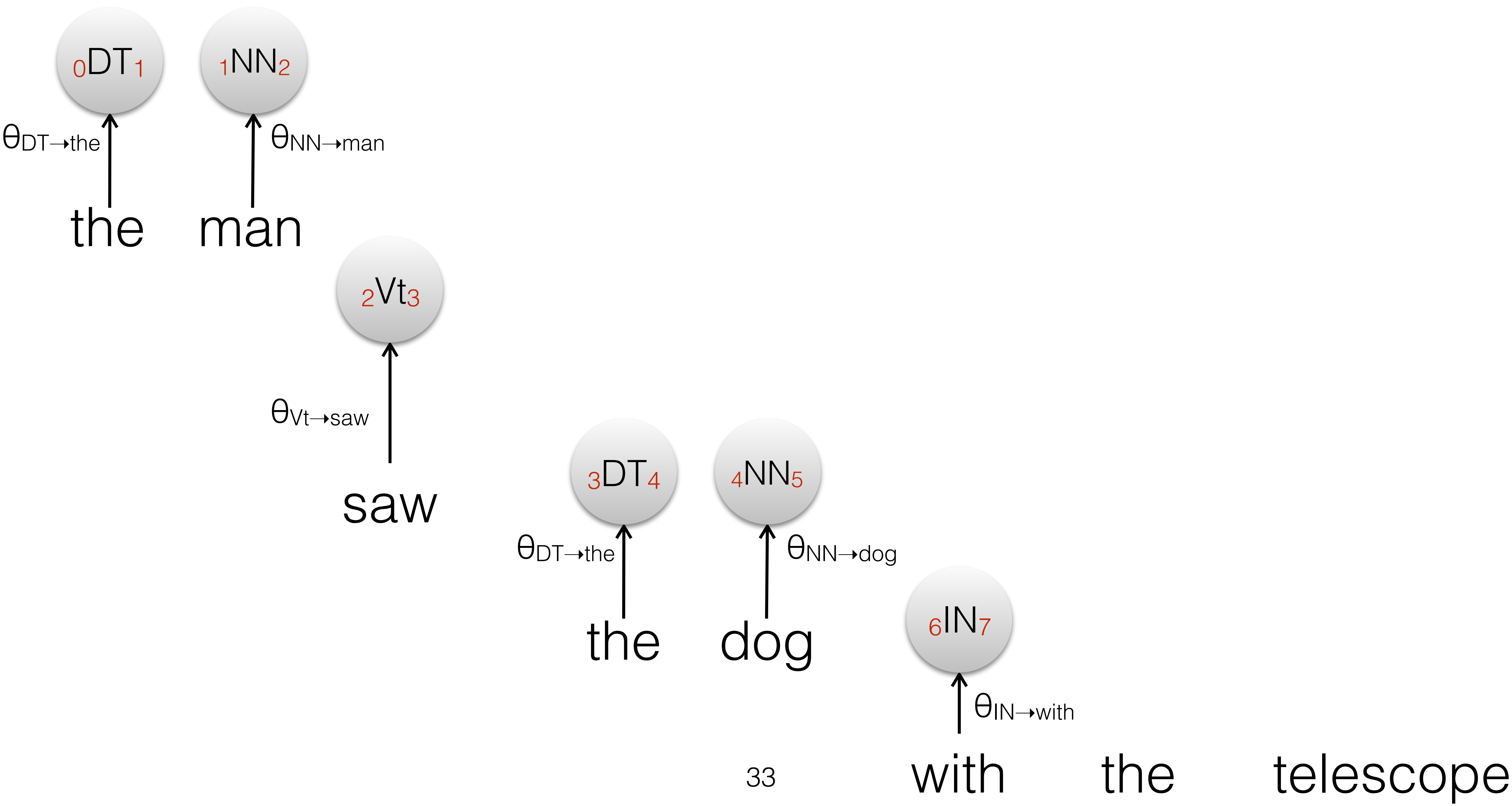


$_0DT_1$
$_1NN_2$
$\theta_{DT \to the}$
$\theta_{NN \to man}$
the
man
$_2Vt_3$
$\theta_{Vt \to saw}$
saw
$_3DT_4$
$\theta_{DT \to the}$
the
dog
with
the
telescope

Ambiguity

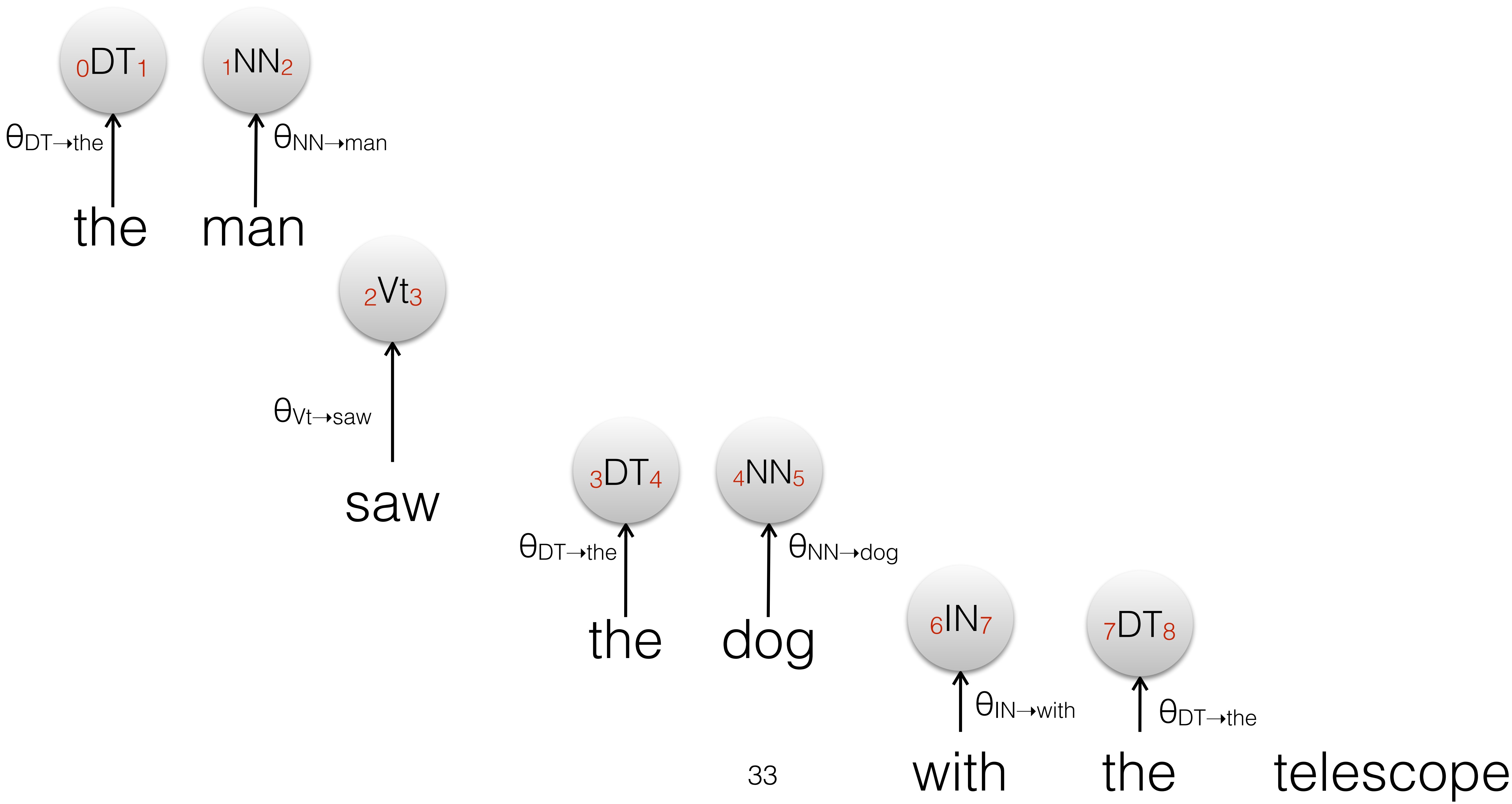

$_{0}DT_{1}$
$_{1}NN_{2}$
$\theta_{DT \to the}$
$\theta_{NN \to man}$
the
man
$_{2}Vt_{3}$
$\theta_{Vt \to saw}$
saw
$_{3}DT_{4}$
$_{4}NN_{5}$
$\theta_{DT \to the}$
$\theta_{NN \to dog}$
the
dog
with
the
telescope

Ambiguity



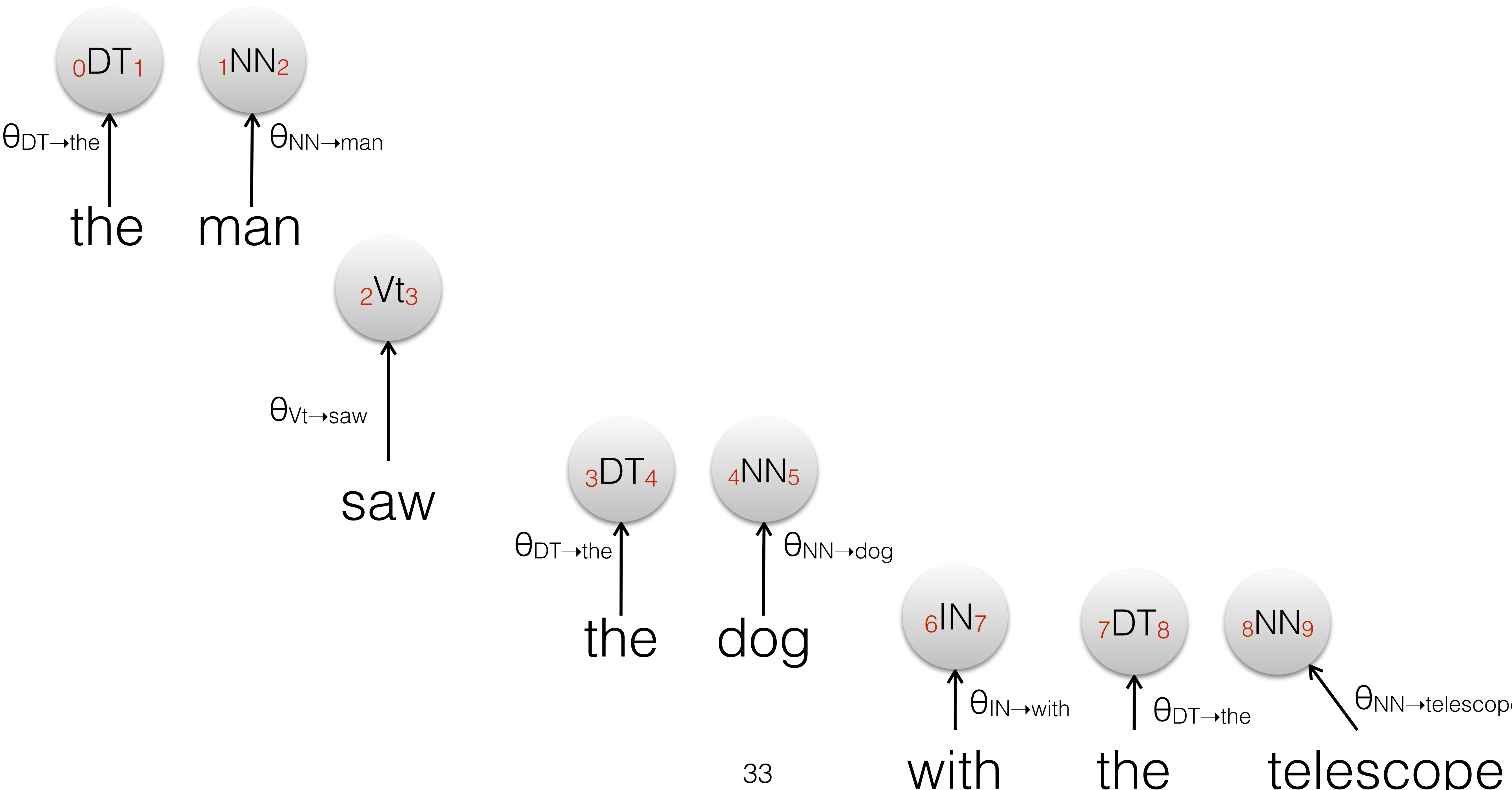
0DT1
1NN2
θDT→the
θNN→man
the
man
2Vt3
θVt→saw
saw
3DT4
4NN5
θDT→the
θNN→dog
the
dog
6IN7
θIN→with
with
the
telescope

Ambiguity



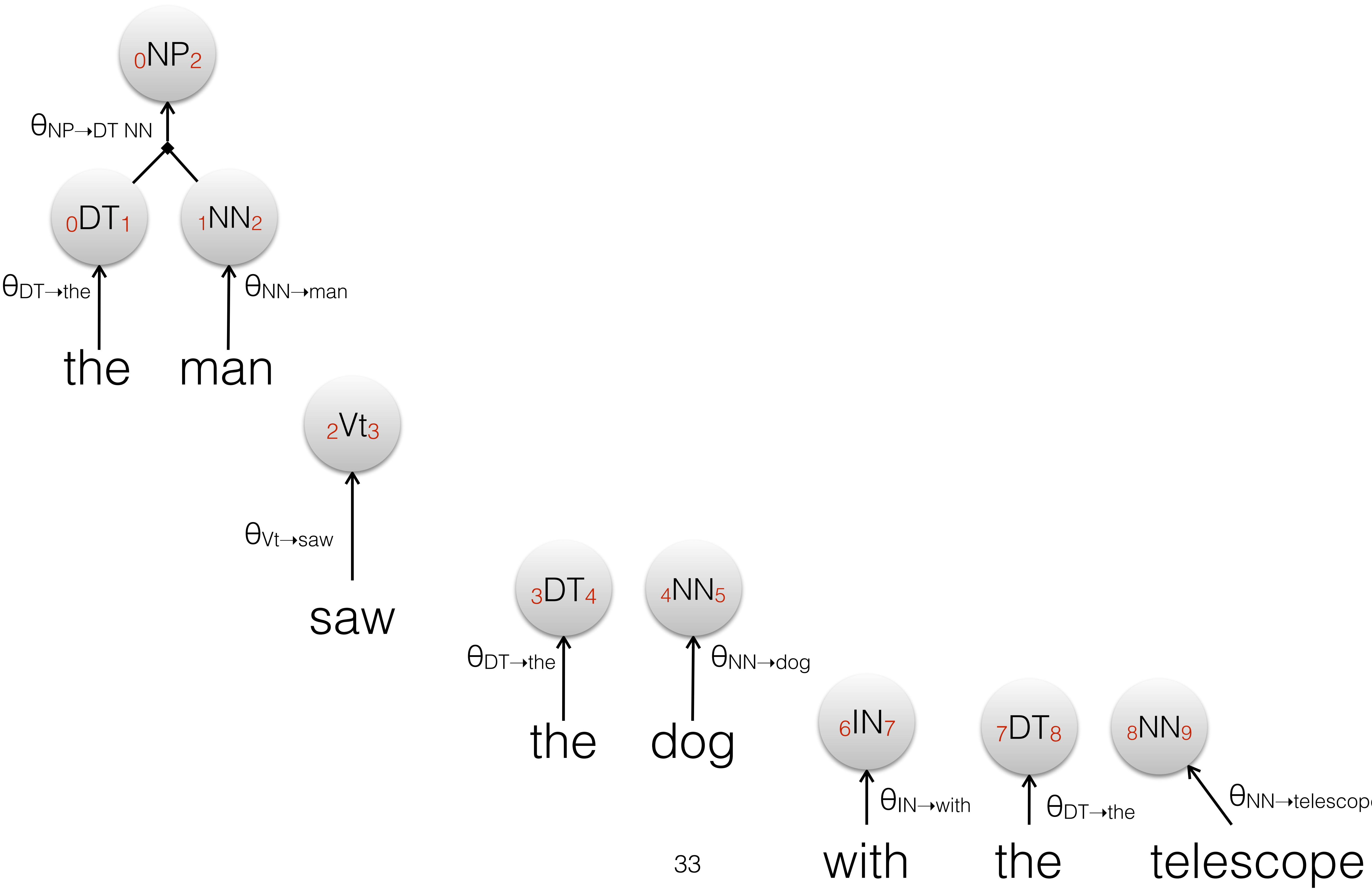
$_0DT_1$
$_1NN_2$
$\theta_{DT\to the}$
$\theta_{NN\to man}$
the
man
$_2Vt_3$
$\theta_{Vt\to saw}$
saw
$_3DT_4$
$_4NN_5$
$\theta_{DT\to the}$
$\theta_{NN\to dog}$
the
dog
$_6IN_7$
$_7DT_8$
$\theta_{IN\to with}$
$\theta_{DT\to the}$
with
the
telescope

Ambiguity



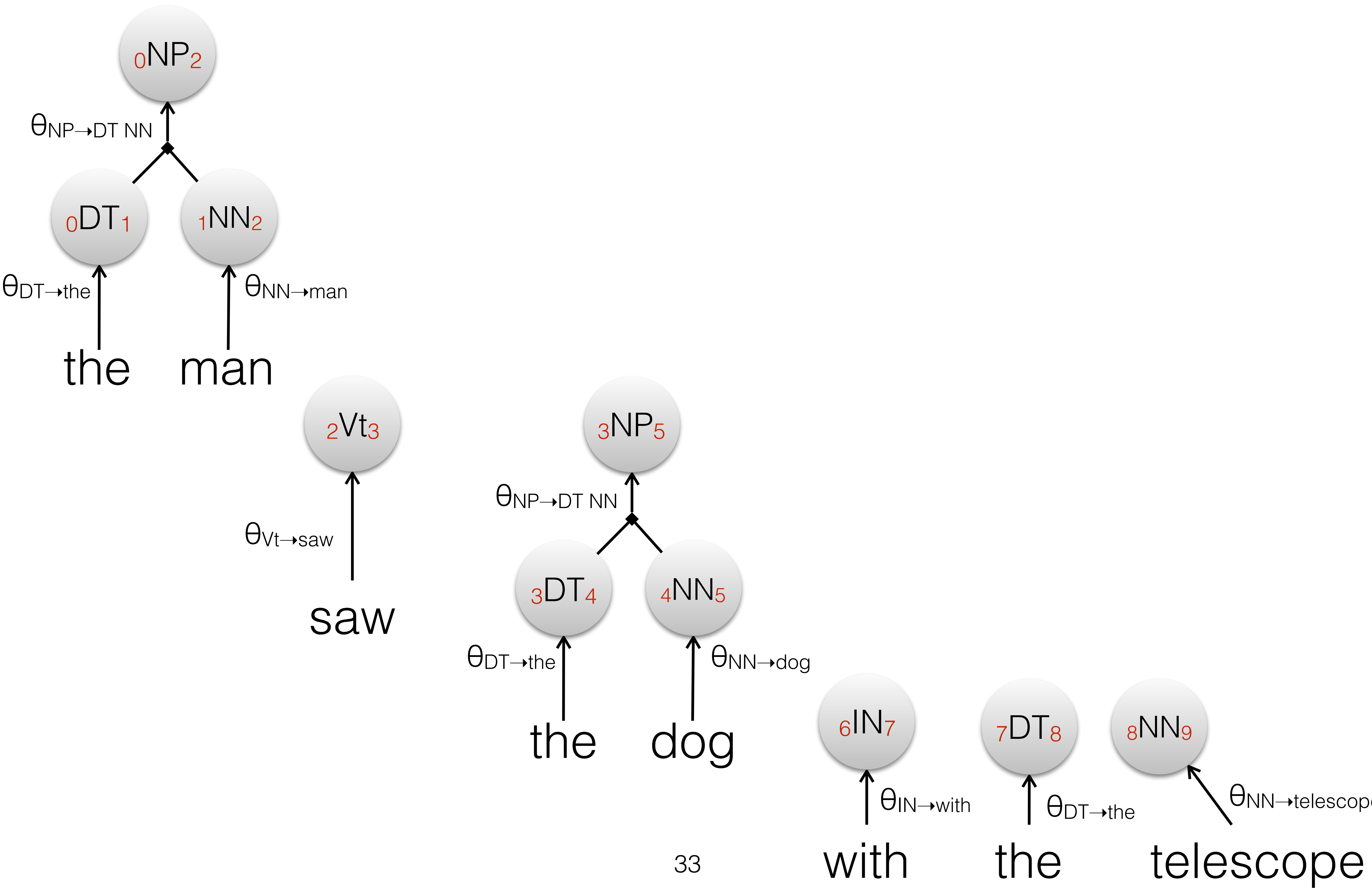
$_0DT_1$
$_1NN_2$
$\theta_{DT \to the}$
$\theta_{NN \to man}$
the
man
$_2Vt_3$
$\theta_{Vt \to saw}$
saw
$_3DT_4$
$_4NN_5$
$\theta_{DT \to the}$
$\theta_{NN \to dog}$
the
dog
$_6IN_7$
$_7DT_8$
$_8NN_9$
$\theta_{IN \to with}$
$\theta_{DT \to the}$
$\theta_{NN \to telescope}$
with
the
telescope

Ambiguity



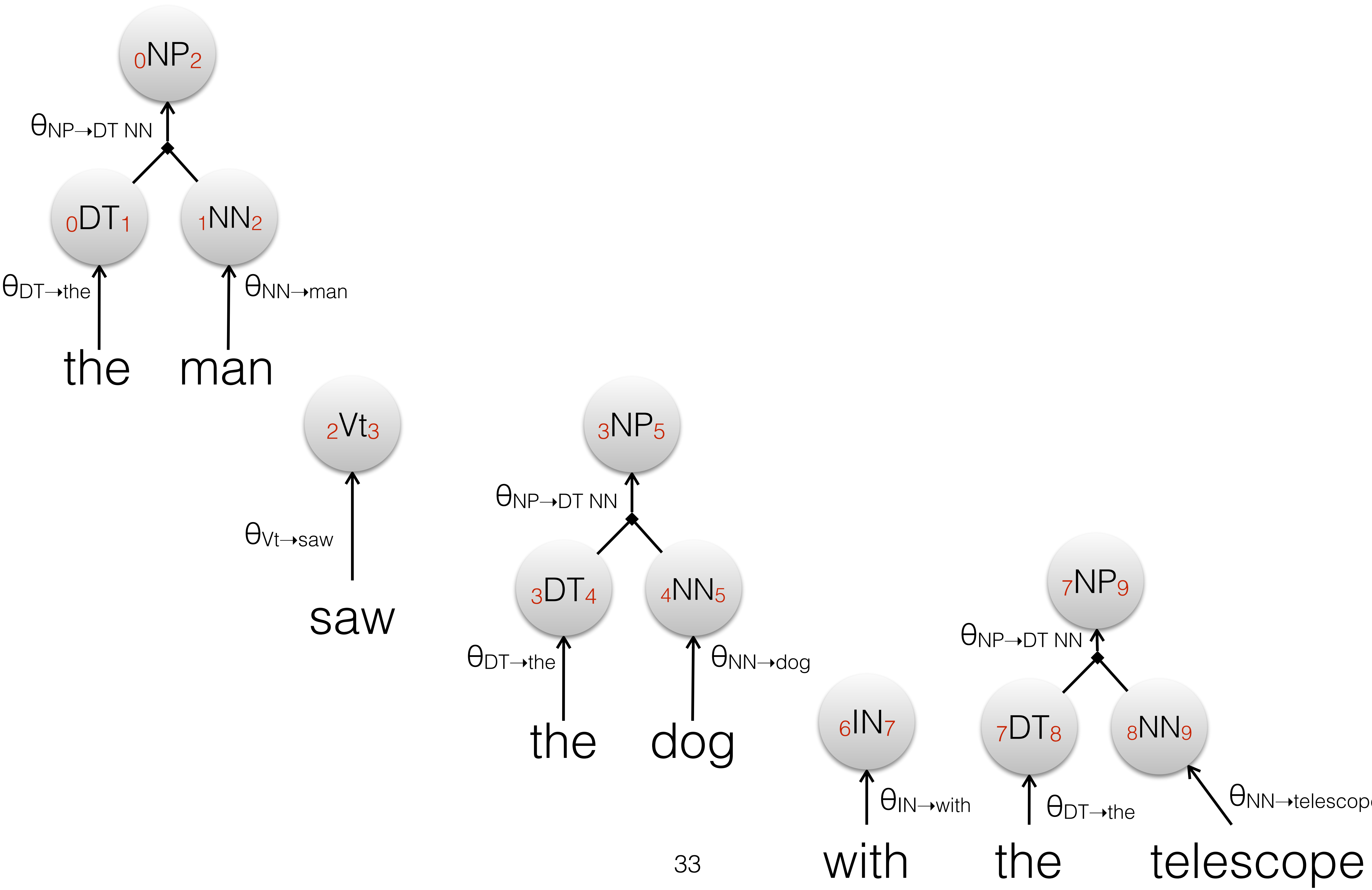
$_0NP_2$
$\theta_{NP \to DT\ NN}$
$_0DT_1$
$_1NN_2$
$\theta_{DT \to the}$
$\theta_{NN \to man}$
the
man
$_2Vt_3$
$\theta_{Vt \to saw}$
saw
$_3DT_4$
$_4NN_5$
$\theta_{DT \to the}$
$\theta_{NN \to dog}$
the
dog
$_6IN_7$
$_7DT_8$
$_8NN_9$
$\theta_{IN \to with}$
$\theta_{DT \to the}$
$\theta_{NN \to telescope}$
with
the
telescope

Ambiguity

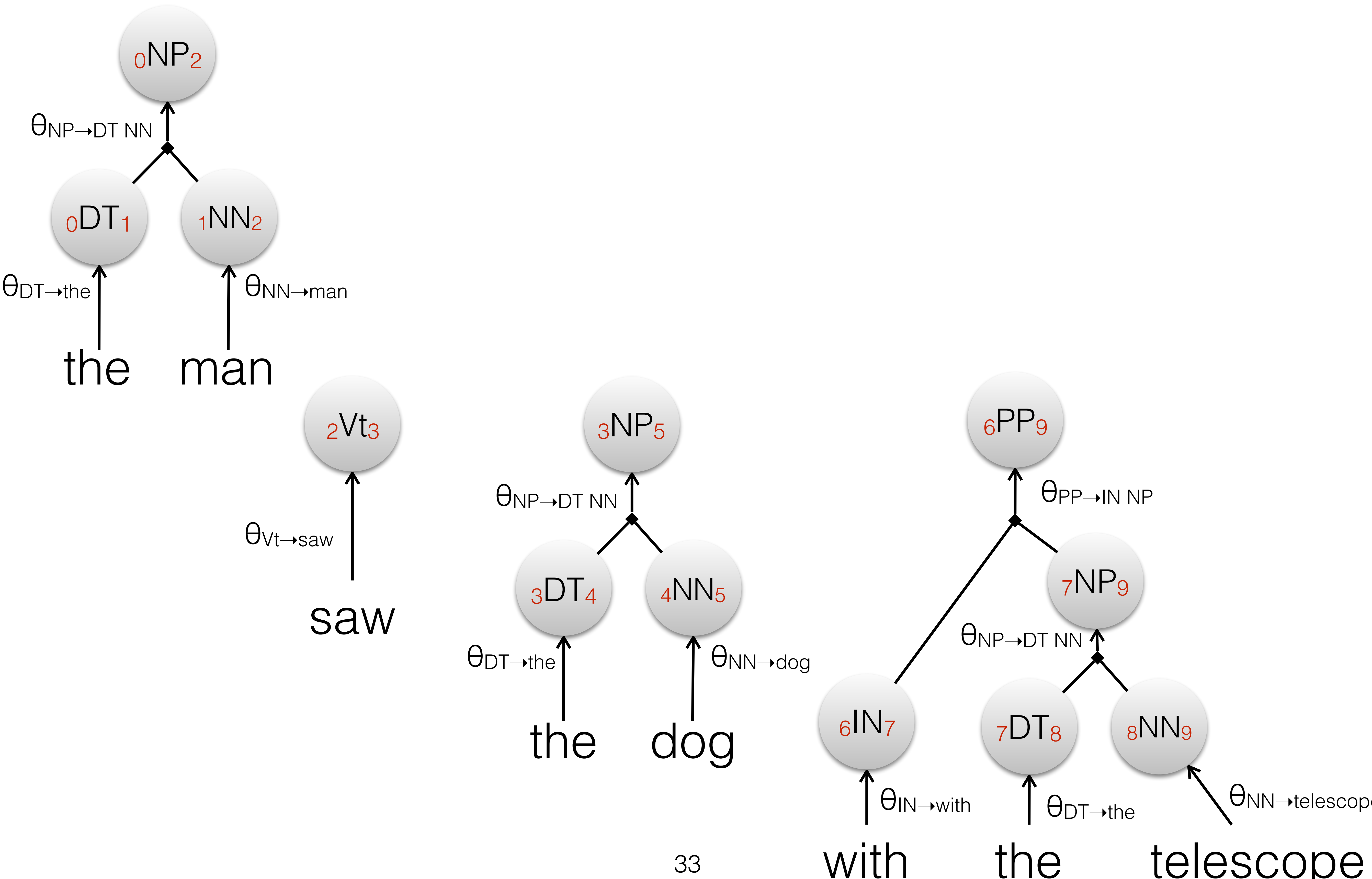


0NP2
θNP→DT NN
0DT1
1NN2
θDT→the
θNN→man
the
man
2Vt3
θVt→saw
saw
3NP5
θNP→DT NN
3DT4
4NN5
θDT→the
θNN→dog
the
dog
6IN7
7DT8
8NN9
θIN→with
θDT→the
θNN→telescope
with
the
telescope

Ambiguity

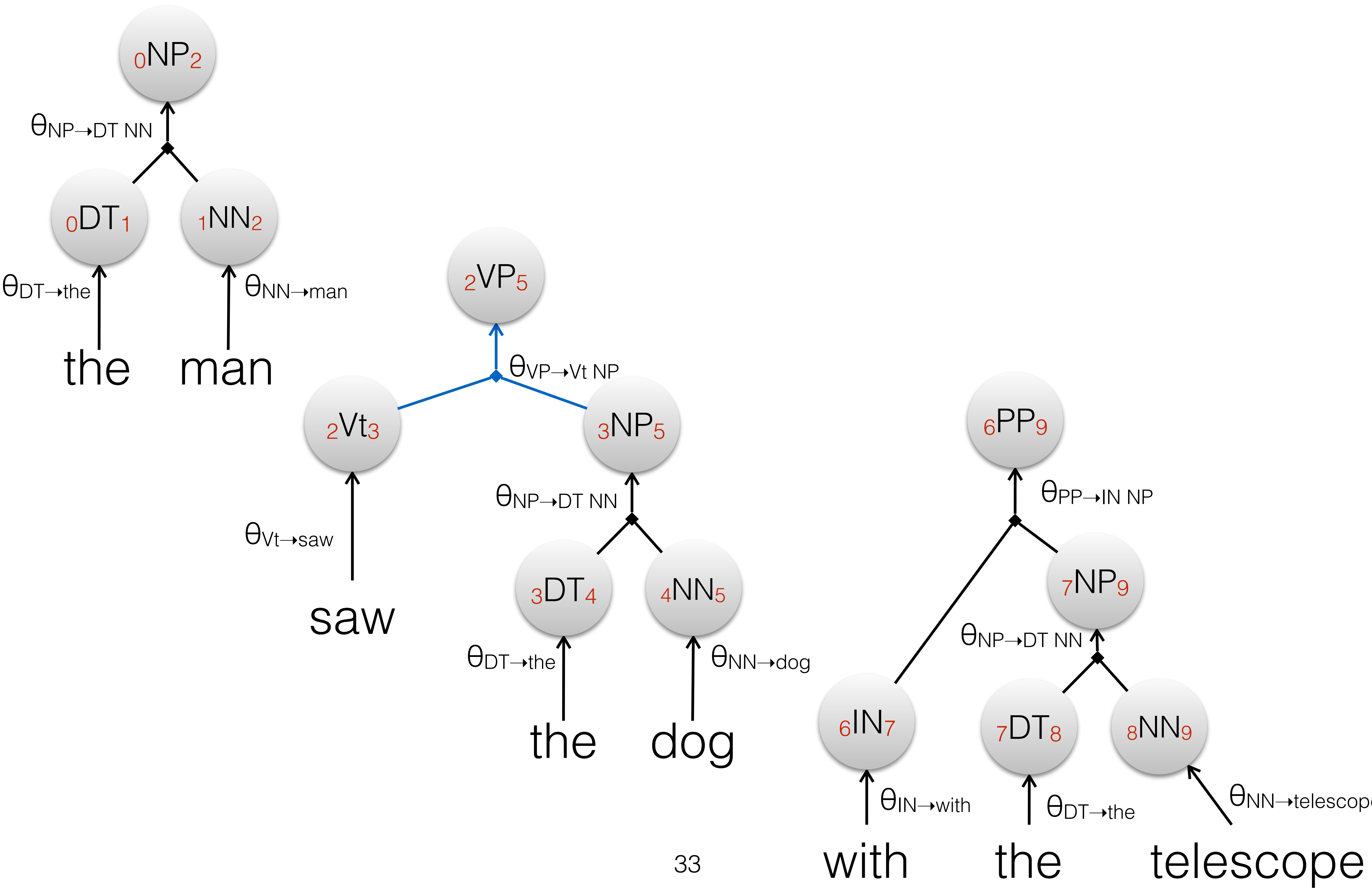

$_0NP_2$
$\theta_{NP \to DT\ NN}$
$_0DT_1$
$_1NN_2$
$\theta_{DT \to the}$
$\theta_{NN \to man}$
the
man
$_2Vt_3$
$\theta_{Vt \to saw}$
saw
$_3NP_5$
$\theta_{NP \to DT\ NN}$
$_3DT_4$
$_4NN_5$
$\theta_{DT \to the}$
$\theta_{NN \to dog}$
the
dog
$_6IN_7$
$\theta_{IN \to with}$
with
$_7NP_9$
$\theta_{NP \to DT\ NN}$
$_7DT_8$
$_8NN_9$
$\theta_{DT \to the}$
$\theta_{NN \to telescope}$
the
telescope

Ambiguity



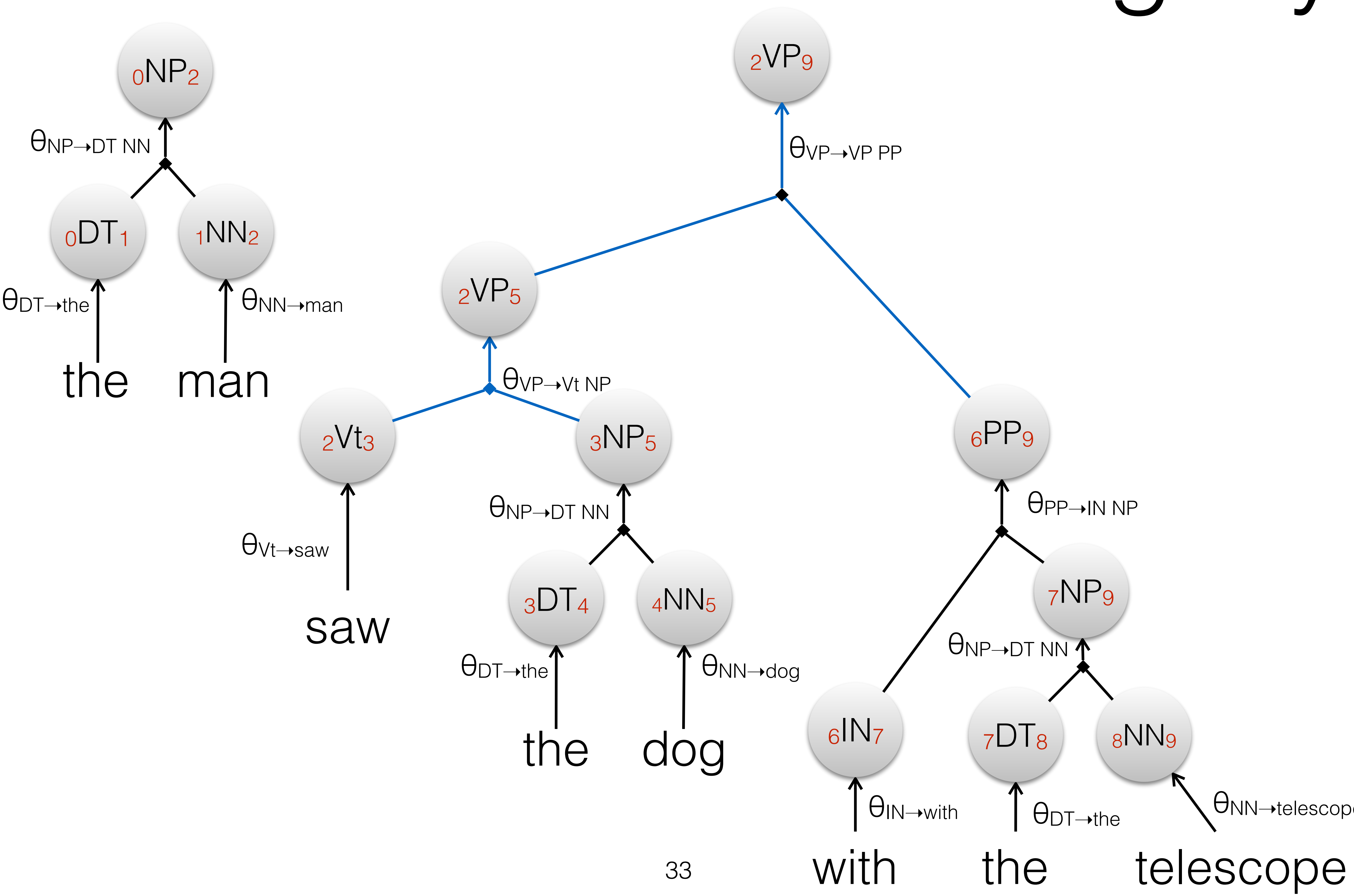
0NP2
θNP→DT NN
0DT1
1NN2
θDT→the
θNN→man
the
man
2Vt3
θVt→saw
saw
3NP5
θNP→DT NN
3DT4
4NN5
θDT→the
θNN→dog
the
dog
6PP9
θPP→IN NP
7NP9
θNP→DT NN
6IN7
7DT8
8NN9
θIN→with
θDT→the
θNN→telescope
with
the
telescope

Ambiguity



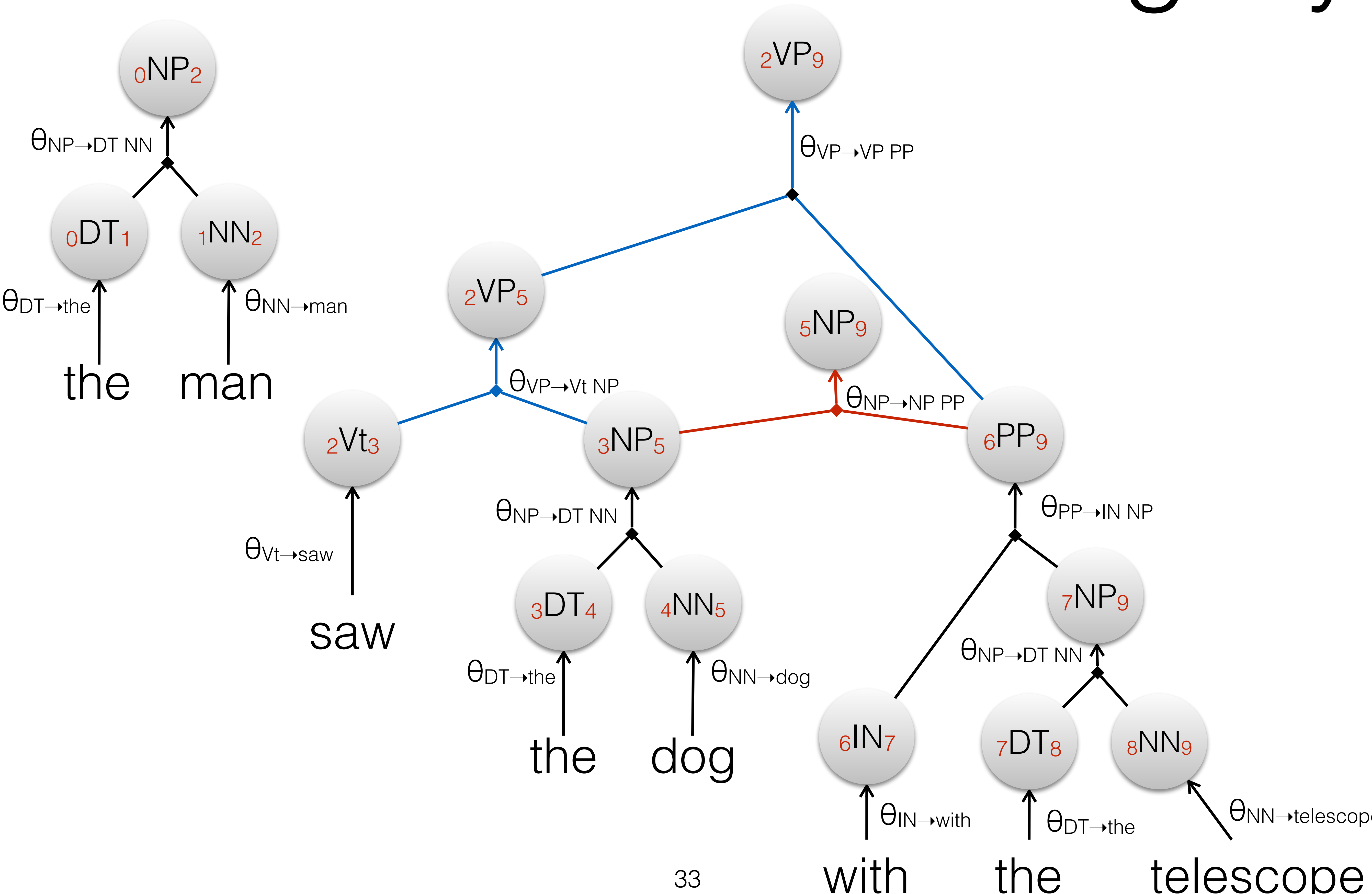
$_0$NP$_2$
θNP→DT NN
$_0$DT$_1$
$_1$NN$_2$
θDT→the
θNN→man
the
man
$_2$VP$_5$
θVP→Vt NP
$_2$Vt$_3$
$_3$NP$_5$
θNP→DT NN
θVt→saw
saw
$_3$DT$_4$
$_4$NN$_5$
θDT→the
θNN→dog
the
dog
$_6$PP$_9$
θPP→IN NP
$_7$NP$_9$
θNP→DT NN
$_6$IN$_7$
$_7$DT$_8$
$_8$NN$_9$
θIN→with
θDT→the
θNN→telescope
with
the
telescope

Ambiguity



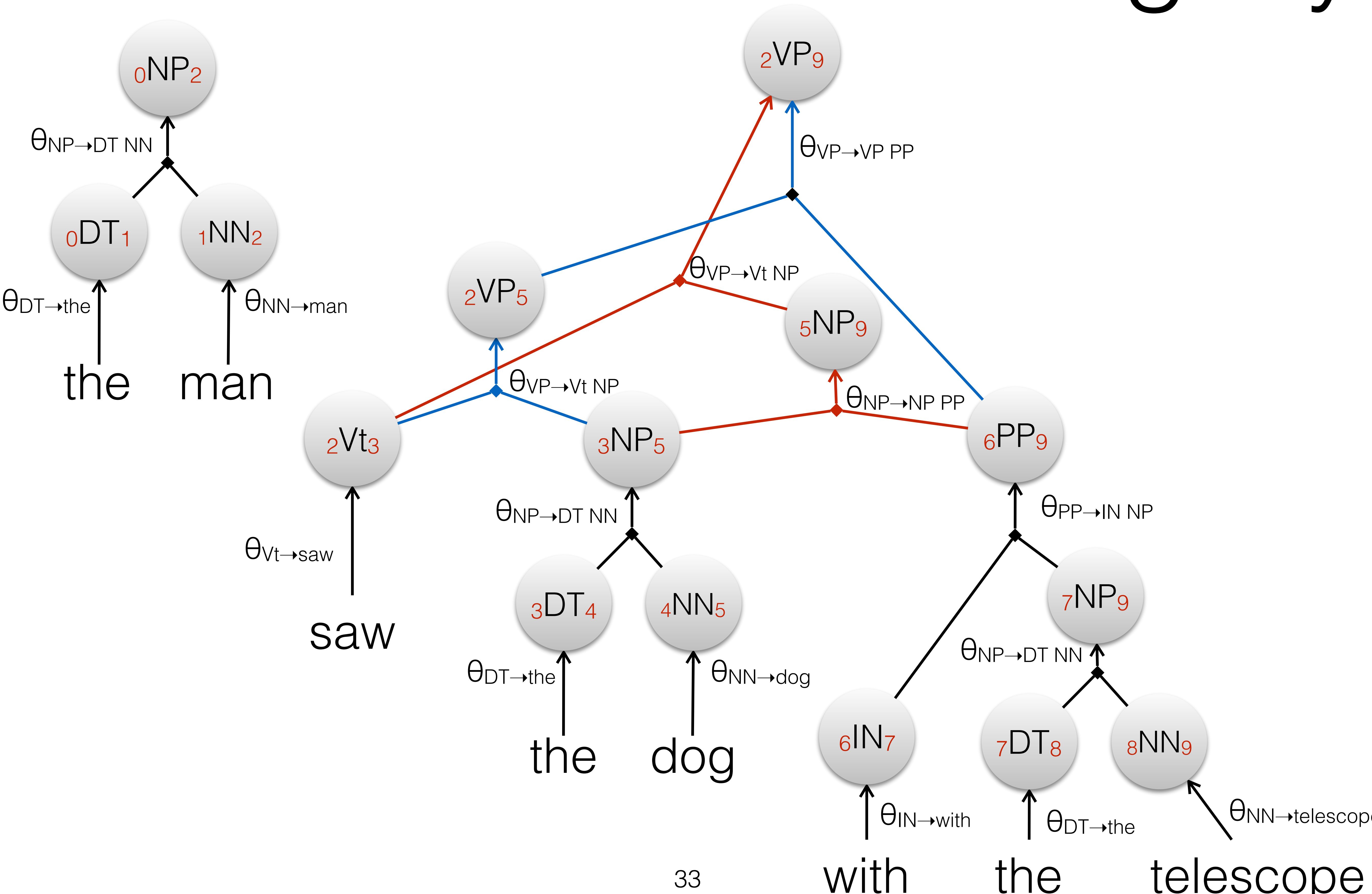
0NP2
θNP→DT NN
0DT1
1NN2
θDT→the
θNN→man
the
man
2VP9
θVP→VP PP
2VP5
θVP→Vt NP
2Vt3
3NP5
6PP9
θVt→saw
saw
θNP→DT NN
3DT4
4NN5
θDT→the
θNN→dog
the
dog
θPP→IN NP
7NP9
θNP→DT NN
6IN7
7DT8
8NN9
θIN→with
θDT→the
θNN→telescope
with
the
telescope

Ambiguity



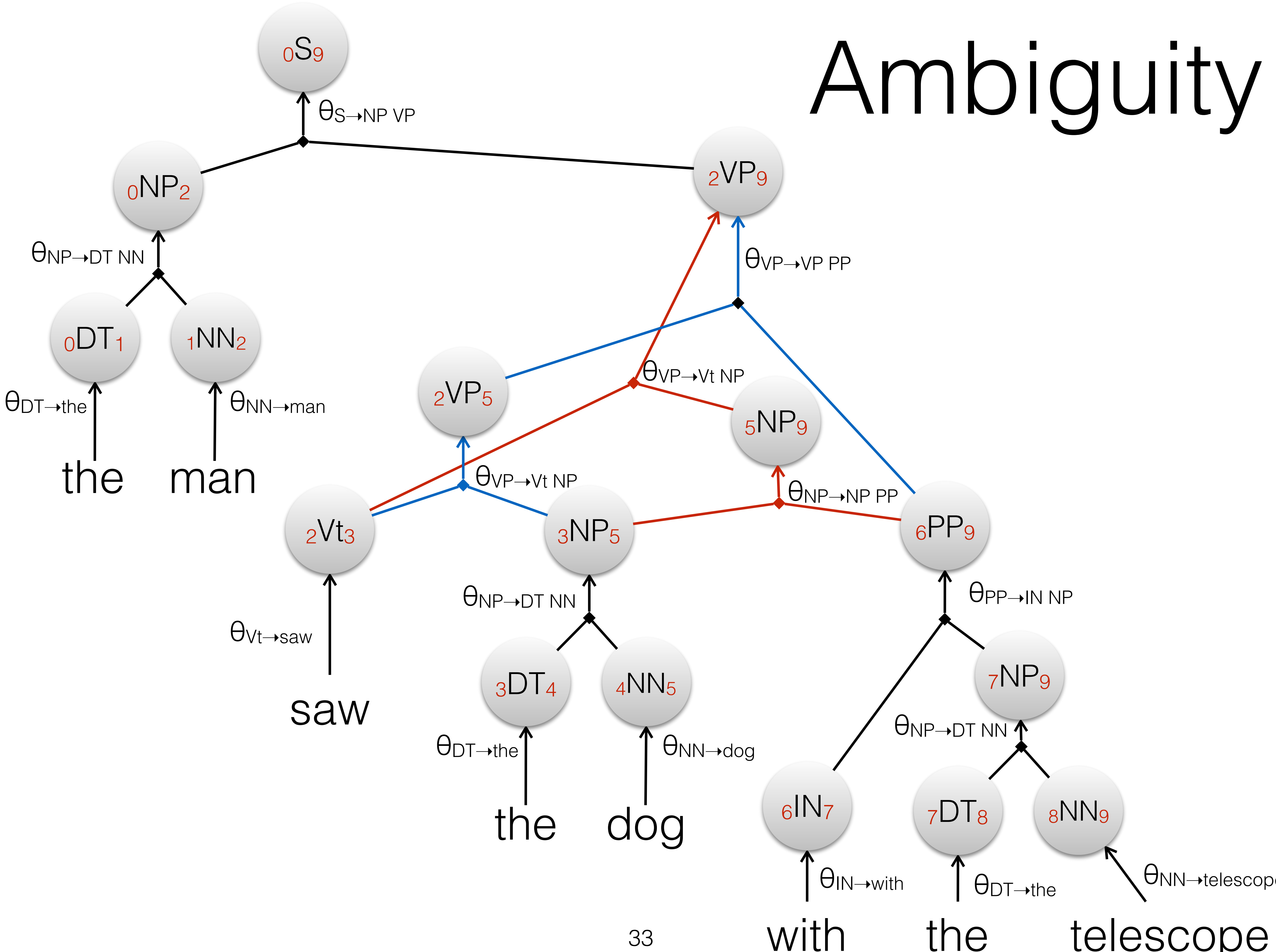
$_0NP_2$
$\theta_{NP \to DT\ NN}$
$_0DT_1$
$_1NN_2$
$\theta_{DT \to the}$
$\theta_{NN \to man}$
the
man
$_2VP_9$
$\theta_{VP \to VP\ PP}$
$_2VP_5$
$_5NP_9$
$\theta_{VP \to Vt\ NP}$
$\theta_{NP \to NP\ PP}$
$_2Vt_3$
$_3NP_5$
$_6PP_9$
$\theta_{NP \to DT\ NN}$
$\theta_{PP \to IN\ NP}$
$\theta_{Vt \to saw}$
saw
$_3DT_4$
$_4NN_5$
$_7NP_9$
$\theta_{DT \to the}$
$\theta_{NN \to dog}$
$\theta_{NP \to DT\ NN}$
the
dog
$_6IN_7$
$_7DT_8$
$_8NN_9$
$\theta_{IN \to with}$
$\theta_{DT \to the}$
$\theta_{NN \to telescope}$
with
the
telescope

Ambiguity



0NP2
θNP→DT NN
0DT1
1NN2
θDT→the
θNN→man
the
man
2VP9
θVP→VP PP
2VP5
θVP→Vt NP
5NP9
θVP→Vt NP
2Vt3
3NP5
θNP→NP PP
6PP9
θNP→DT NN
θPP→IN NP
θVt→saw
saw
3DT4
4NN5
7NP9
θDT→the
θNN→dog
θNP→DT NN
the
dog
6IN7
7DT8
8NN9
θIN→with
θDT→the
θNN→telescope
with
the
telescope

Ambiguity



0S9
θS→NP VP
0NP2
2VP9
θNP→DT NN
θVP→VP PP
0DT1
1NN2
θDT→the
θNN→man
the
man
2VP5
θVP→Vt NP
5NP9
θVP→Vt NP
θNP→NP PP
2Vt3
3NP5
6PP9
θNP→DT NN
θPP→IN NP
θVt→saw
saw
3DT4
4NN5
7NP9
θNP→DT NN
θDT→the
θNN→dog
the
dog
6IN7
7DT8
8NN9
θIN→with
θDT→the
θNN→telescope
with
the
telescope

Complexity

Complexity

Item form: $[i, X \rightarrow \alpha\blacksquare\bullet\beta\square, j]$

Complexity

Item form: $[i, X \rightarrow \alpha_{\blacksquare} \bullet \beta_{\square}, j]$

- Each rule segments the input $x_1 .. x_n$

Complexity

Item form: [i, X → α■•β□, j]

- Each rule segments the input x_1 .. x_n

Every CFG can be written in CNF (max arity = 2)

Complexity

Item form: $[i, X \rightarrow \alpha \blacksquare \bullet \beta \square, j]$

- Each rule segments the input x_1 .. x_n

Every CFG can be written in CNF (max arity = 2)

- In total we get up to 3 indices ranging from 0 .. n

Complexity

Item form: [i, X → α■•β□, j]

- Each rule segments the input x_1 .. x_n

Every CFG can be written in CNF (max arity = 2)

- In total we get up to 3 indices ranging from 0 .. n
- $O(n^3)$ annotated rules

Bibliography

- Hopcroft, John E. and Ullman, Jeffrey D. 1979. Introduction To Automata Theory, Languages, And Computation.

- Shieber, S. and Schabes, Y. and Pereira, F. 1995. Principles and implementation of deductive parsing. In *Journal of Logic Programming*

- Bar-Hillel, Y. and Perles, M. and Shamir, E. 1961. On formal properties of simple phrase structure grammars.

- Billot, S. and Lang, B. 1989. The Structure of Shared Forests in Ambiguous ParsingThe Structure of Shared Forests in Ambiguous Parsing. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*.