Structures of Words & Sentences

Morphology tips + practice

Example Together

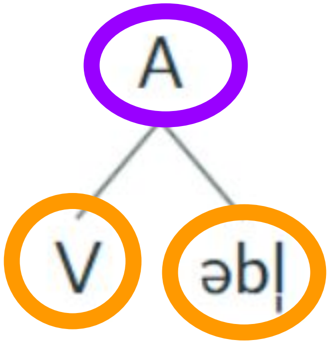
rerolling

Our Theory's Morpheme Rules

V + [əbl̩] = A (*readable*)	V + [ɨɹ] = N (*reader*)
V + [d] = V (*bored*)	A + [ɨɹ] = A (*happier*)
A + [nɛs] = N (*happiness*)	V + [ɪŋ] = A (*running*)
[ʌn] + V = V (*untie*)	[ɹij] + V =V (*retie*)
[ʌn] + A = A (*unhappy*)	N + [z] = N (*shoes*)

Trees: Just Visual Representation of Rules

V(erb) + [əbl̩] = A(djective)



Since these are our hypothesized morpheme rules…

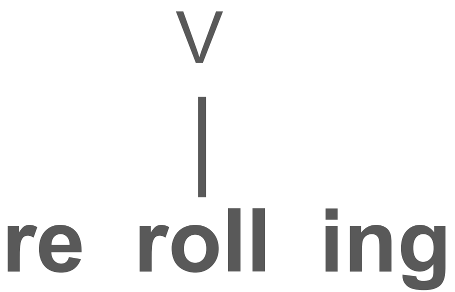
V + [əbl̩] = A (*readable*)	V + [ɨɹ] = N (*reader*)
V + [d] = V (*bored*)	A + [ɨɹ] = A (*happier*)
A + [nɛs] = N (*happiness*)	V + [ɪŋ] = A (*running*)
[ʌn] + V = V (*untie*)	[ɹij] + V =V (*retie*)
[ʌn] + A = A (*unhappy*)	N + [z] = N (*shoes*)

…we should be using one of these each time you attach a morpheme in our derivation trees!

V + [əbl̩] = A (*readable*)	V + [ɨɹ] = N (*reader*)
V + [d] = V (*bored*)	A + [ɨɹ] = A (*happier*)
A + [nɛs] = N (*happiness*)	V + [ɪŋ] = A (*running*)
[ʌn] + V = V (*untie*)	[ɹij] + V =V (*retie*)
[ʌn] + A = A (*unhappy*)	N + [z] = N (*shoes*)

re roll ing

V + [əbl̩] = A (*readable*)	V + [ɨɹ] = N (*reader*)
V + [d] = V (*bored*)	A + [ɨɹ] = A (*happier*)
A + [nɛs] = N (*happiness*)	V + [ɪŋ] = A (*running*)
[ʌn] + V = V (*untie*)	[ɹij] + V =V (*retie*)
[ʌn] + A = A (*unhappy*)	N + [z] = N (*shoes*)



V + [əbl̩] = A
(*readable*)

V + [ɨɹ] = N
(*reader*)

V + [d] = V
(*bored*)

A + [ɨɹ] = A
(*happier*)

A + [nɛs] = N
(*happiness*)

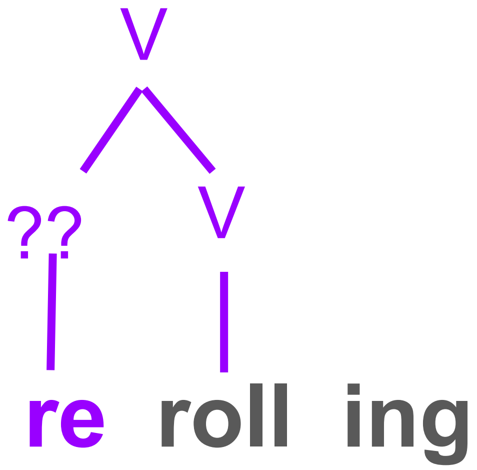
V + [ɪŋ] = A
(*running*)

[ʌn] + V = V
(*untie*)

[ɹij] + V =V
(*retie*)

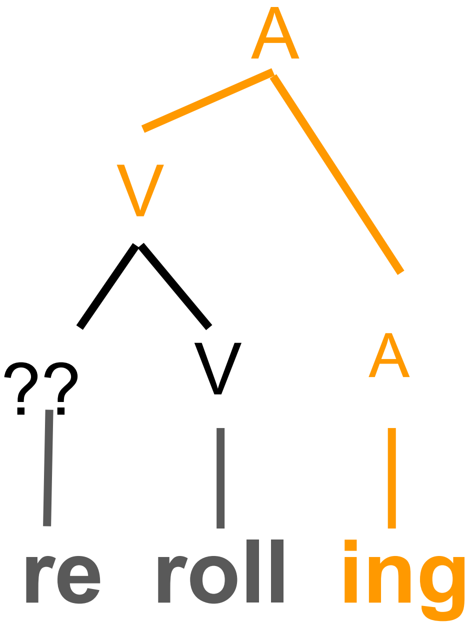
[ʌn] + A = A
(*unhappy*)

N + [z] = N
(*shoes*)



"reroll"

V + [əbl̩] = A (*readable*)	V + [ɨɹ] = N (*reader*)
V + [d] = V (*bored*)	A + [ɨɹ] = A (*happier*)
A + [nɛs] = N (*happiness*)	V + [ɪŋ] = A (*running*)
[ʌn] + V = V (*untie*)	[ɹij] + V =V (*retie*)
[ʌn] + A = A (*unhappy*)	N + [z] = N (*shoes*)



Right hand head rule

V + [əbl̩] = A (*readable*)	V + [ɨɹ] = N (*reader*)
V + [d] = V (*bored*)	A + [ɨɹ] = A (*happier*)
A + [nɛs] = N (*happiness*)	V + [ɪŋ] = A (*running*)
[ʌn] + V = V (*untie*)	[ɹij] + V =V (*retie*)
[ʌn] + A = A (*unhappy*)	N + [z] = N (*shoes*)

Key Differences between Inflectional and Derivational

1) inflectional affixes never change the category of the word (derivational affixes can **but don't have to**)

2) inflectional affixes can apply to any member of the category they attach to.

For example, every verb has a past tense(inflectional) form, but not every verb has an "-able" (derivational) form.

3) “Morpheme Ordering Constraint”: A derivational affix can't be added to a word that already has an inflectional affix attached.

For example, you can't get "bakeder", where you would have to add "er" (derivational, someone who does X) to "baked" (already has -ed past tense, inflectional).

Morphology Derivation Reference Sheet

- Each step in your tree should correspond to one of the morpheme rules
- Mark parts of speech
  - Right hand head rule
  - Not inflectional morphemes
- Each step should yield a real/possible English word by itself
  - e.g. not “unredo” in “unredoable”
- Morpheme ordering constraint
  - Can’t attach derivational after inflectional

V + [əbl̩] = A (*readable*)

V + [ɨɹ] = N (*reader*)

V + [d] = V (*bored*)

A + [ɨɹ] = A (*happier*)

A + [nɛs] = N (*happiness*)

V + [ɪŋ] = A (*running*)

[ʌn] + V = V (*untie*)

[ɹij] + V =V (*retie*)

[ʌn] + A = A (*unhappy*)

N + [z] = N (*shoes*)

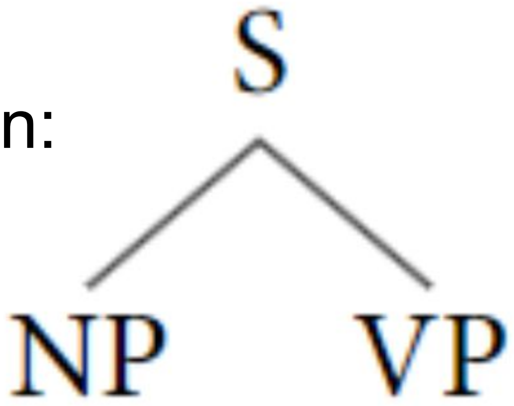
Syntax tips + practice

Deriving sentences instead of words

Rule:

$$S \rightarrow NP\ VP$$

Visual representation:



Our theory’s syntax (“phrase structure”) rules: https://umamherst.instructure.com/courses/9157/files/3405843?module_item_id=567195

N → car, her, Sam, children, discussion, elf…

V → eat, give, put, stand, scream, mix…

P → to, in, on, until, with…

A → blue, happy, small…

D → the, that, a, some, many…

Aux → have, be, must, may, can, should…

C → that, if, whether

$NP \rightarrow D\ \overline{N}$

$NP \rightarrow \overline{N}$

$\overline{N} \rightarrow A\ \overline{N}$

$\overline{N} \rightarrow \overline{N}\ PP$

$\overline{N} \rightarrow \overline{N}\ CP$

$\overline{N} \rightarrow N$

$CP \rightarrow C\ S$

$S \rightarrow NP\ VP$

$S \rightarrow CP\ VP$

$PP \rightarrow P\ NP$

$PP \rightarrow P$

$VP \rightarrow VP\ PP$

$VP \rightarrow VP\ CP$

$VP \rightarrow V\ NP$

$VP \rightarrow Aux\ VP$

$VP \rightarrow V$

α → α and α,

where α is a phrase: $\overline{N}$, NP, VP, S, CP, or PP

Example Together

They walked in the mud

Example Together

N V P D N

They walked in the mud

Example Together

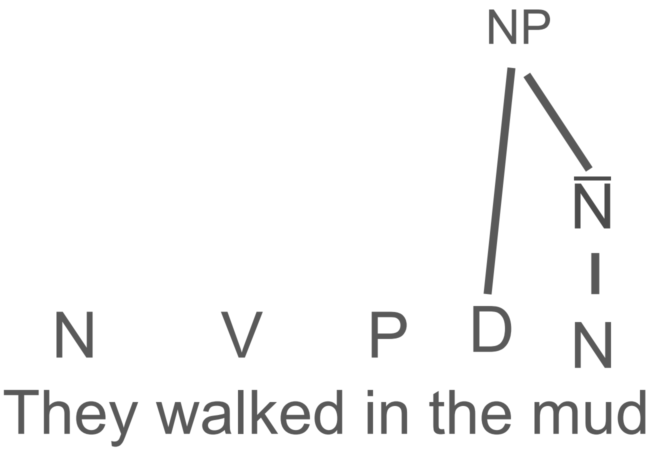
$\bar{N}$

N V P D N

They walked in the mud

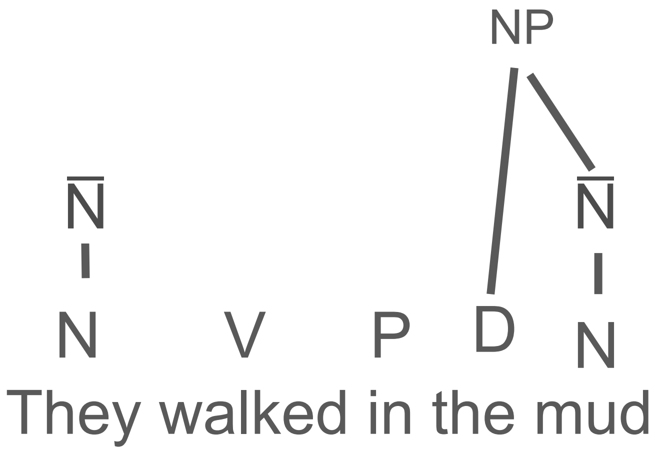
$$\bar{N} \rightarrow N$$

Example Together



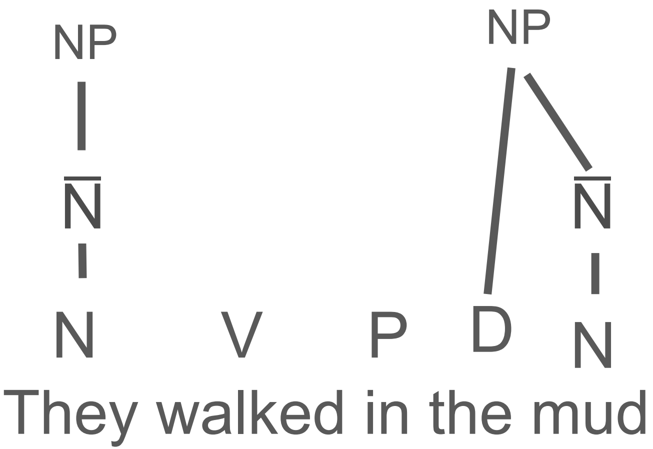
$$NP \rightarrow D\ \overline{N}$$

Example Together



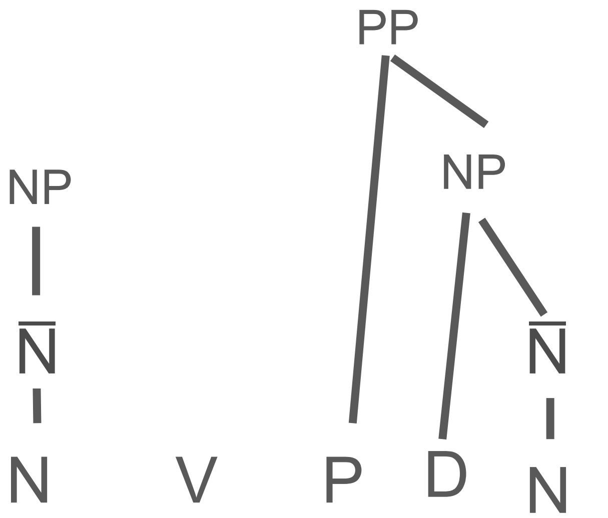
$$\bar{N} \rightarrow N$$

Example Together



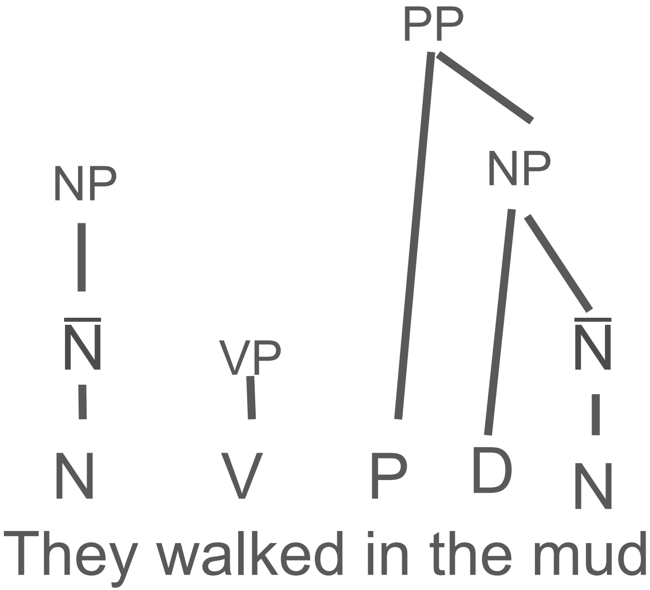
$$NP \rightarrow \overline{N}$$

Example Together

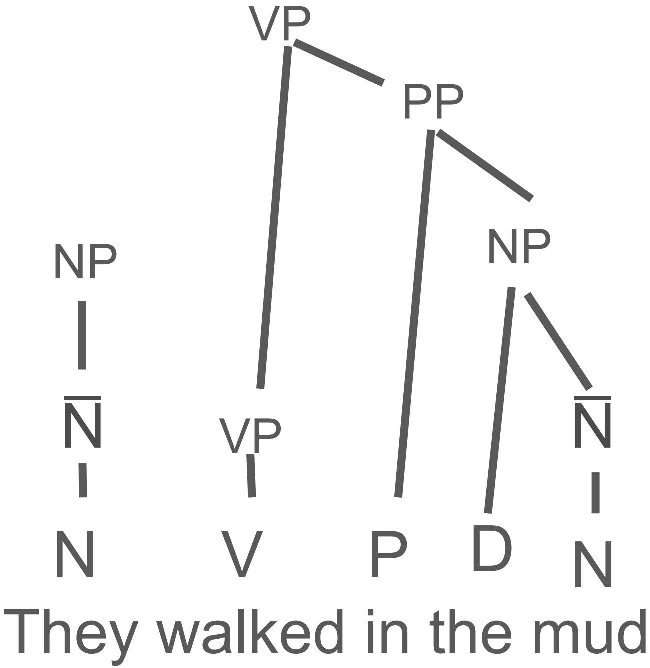


They walked in the mud

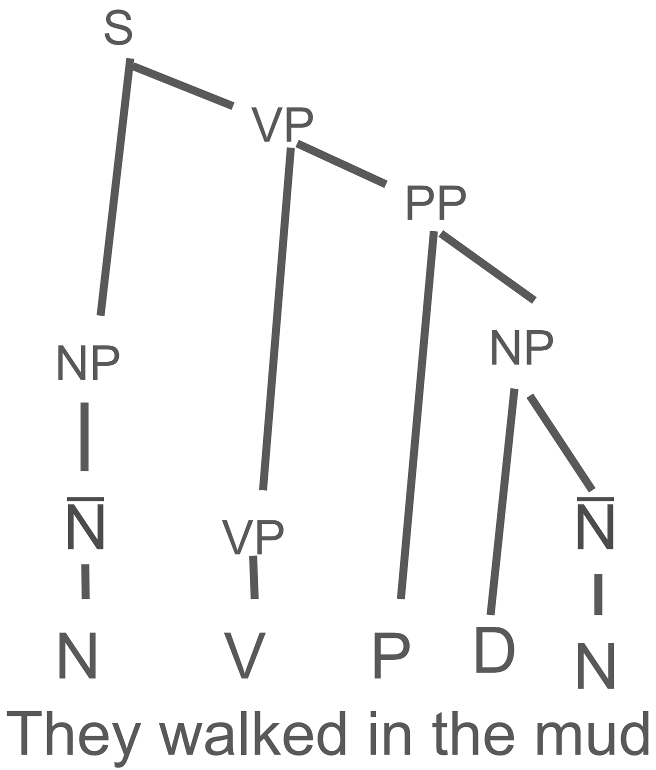
$$PP \rightarrow P\ NP$$



$$VP \rightarrow V$$



$$VP \rightarrow VP\ PP$$



$$S \rightarrow NP\ VP$$

Sam pets dogs on the bus

The children on the bus cried

Sam can pet dogs and cats

Sam can pet dogs on the bus

Dogs and cats like Sam

Sam left me on the bus

Assignment 6: Constraints vs Rules

A constraint is a generalization about the forms found in a whole language. It states that some combination of values isn't allowed in that language.

For example, in English, a constraint is that you can't have a voiced sound after a voiceless one in a coda.

Constraint: No voiced sounds after voiceless sounds in a coda

Not a constraint: voiced sounds turn voiceless after a voiceless sound in a coda

-This is a rule, not a constraint

-It tells us how English fixes violations of the constraint, not what the constraint is

-A language with that same constraint could fix violations some other way, like deleting the voiceless sound

Assignment 6: Morphology/Phonology interaction

Step 1: Brains add the morpheme they want, e.g. /z/ for past tense

e.g. [kæt] + [z] or [dɑg] + [z]

Step 2: Brains check if the result violates any of the constraints in their language

[kætz] violates "*[voiceless][voiced] in coda"

[dagz] is fine

Step 3: Brains apply the rule their language has for fixing that constraint violation

Change voicing of past tense: [kæts]