Computational Linguistics

CSC 2501 / 485
Fall 2015

4

4. Extending grammars with features

Frank Rudzicz
Toronto Rehabilitation Institute-UHN; and
Department of Computer Science, University of Toronto

Reading: Jurafsky & Martin: 12.3.4–6, 15.0–3;
*[*Allen: *4.1–5];*
Bird et al: 9.

Agreement and inflection

- Problem: **Agreement** phenomena.

  *Nadia {wash<u>es</u>/**<u>**wash**</u>**}** the dog.*

  *The boys {****<u>**washes**</u>**/wash} the dog.*

  *You {****<u>**washes**</u>**/wash} the dog.*

- **Morphological inflection** of verb 'must' match subject noun in person and number (in English).

Subject–verb agreement 1

	Present tense			
	Singular		**Plural**	
1	*I*	*wash*	*we*	*wash*
2	*you*	*wash*	*you*	*wash*
3	*he, she, it*	*washes*	*they*	*wash*

1	*I*	*am*	*we*	*are*
2	*you*	*are*	*you*	*are*
3	*he, she, it*	*is*	*they*	*are*

Subject–verb agreement 2

	Past tense			
	Singular		**Plural**	
1	*I*	*washed*	*we*	*washed*
2	*you*	*washed*	*you*	*washed*
3	*he, she, it*	*washed*	*they*	*washed*

1	*I*	*was*	*we*	*were*
2	*you*	*were*	*you*	*were*
3	*he, she, it*	*was*	*they*	*were*

Agreement features 1

- English agreement rules are fairly simple.
  - Subject :: verb w.r.t. **person** and **number**.
  - No agreement required between verb and object.
- Many languages have other agreements.
  - *E.g.*, German: Article and adjective ending depends on noun gender and case:

Agreement features 2

Nominative Case (Subject Case)

Masculine **der**	Feminine **die**	Neuter **das**	Plural **die**
der neu**e** Wagen the new car	die schön**e** Stadt the beautiful city	das alt**e** Auto the old car	die neu**en** Bücher the new books
Masculine **ein**	Feminine **eine**	Neuter **ein**	Plural **keine**
ein neu**er** Wagen a new car	eine schön**e** Stadt a beautiful city	ein alt**es** Auto an old car	keine neu**en** Bücher no new books

Accusative Case (Direct Object)

Masculine **den**	Feminine **die**	Neuter **das**	Plural **die**
den neu**en** Wagen the new car	die schön**e** Stadt the beautiful city	das alt**e** Auto the old car	die neu**en** Bücher the new books
Masculine **einen**	Feminine **eine**	Neuter **ein**	Plural **keine**
einen neu**en** Wagen a new car	eine schön**e** Stadt a beautiful city	ein alt**es** Auto an old car	keine neu**en** Bücher no new books

Ask about.com: German language: Adjective endings I and II.
http://german.about.com/library/weekly/aa033098.htm *and*
aa033098.htm

Agreement features 3

E.g., Chinese: Numeral classifiers, often based on shape, aggregation, … :

两条鱼 *liang tiao yu* ‘two CLASSIF-LONG-ROPELIKE fish’
两条河 *liang tiao he* ‘two CLASSIF-LONG-ROPELIKE rivers’
两条腿 *liang tiao tui* ‘two CLASSIF-LONG-ROPELIKE legs’
两条裤子 *liang tiao kuzi* ‘two CLASSIF-LONG-ROPELIKE pants’
两只胳膊 *liang zhi gebo* ‘two CLASSIF-GENERAL arms’
两件上衣 *liang jian shangyi* ‘two CLASSIF-CLOTHES-ABOVE-WAIST tops’
两套西装 *liang tao xizhuang* ‘two CLASSIF-SET suits’

Zhang, Hong (2007). Numeral classifiers in Mandarin Chinese. *Journal of East Asian Linguistics*, 16(1), 43–59. Thanks also to Tong Wang, Vanessa Wei Feng, and Helena Hong Gao.

Inflectional morphology

- Word may be **inflected** …
  - … to indicate some semantic properties:
    singular / plural, past / present, …
  - … to agree with inflection of other words.
- Each (open-class) word-type has a **base form / stem / lemma**.
- Each occurrence of a word includes inflection by a (possibly null) morphological change.

Rule proliferation 1

- **Problem:** How to account for this in grammar.
- **Possible solution:** Replace all NPs, Vs, and VPs throughout the grammar.

S → NP VP

NP → *you*, *dog*, *dogs*, *bear*, *bears*, …

VP → V NP

V → *washes*, *wash*, *washed*, *is*, *was*, …

S → NP_{3s} VP_{3s}
S → NP_{3p} VP_{3p}
S → NP_2 VP_2
S → NP_{1s} VP_{1s}
S → NP_{1p} VP_{1p}

NP_{3s} → *dog*, *bear*, …
NP_{3p} → *dogs*, *bears*, …

NP_2 → *you*
⋮
VP_{3s} → V_{3s} NP
⋮
V_{3s} → *is*, *was*, *washes*, *washed*, …
V_{3p} → *are*, *were*, *wash*, *washed*, …

V_{1s} → *am*, *was*, *wash*, *washed*, …
⋮

Rule proliferation 2

- *Drawback:* Losing the generalization:
  - All these Ss, NPs, VPs have the same structure.
  - Doesn't depend on particular verb, noun, and number.
- Context-free rules collapse together structural and 'featural' information.
- All information must be completely and directly specified.
  - *E.g.*, you can't specify agreement without **fully specifying** that agreement.

Feature structures 1

- *Solution:* Separate **feature** information from **syntactic**, **structural**, and **lexical** information.
- A **feature structure** is a list of pairs:

  [*feature-name feature-value*]

- Feature-values may be **atoms** or **feature structures**.
- You can consider **syntactic category** or **word** to be features too.
- You can represent syntactic structure.

Feature structures 2

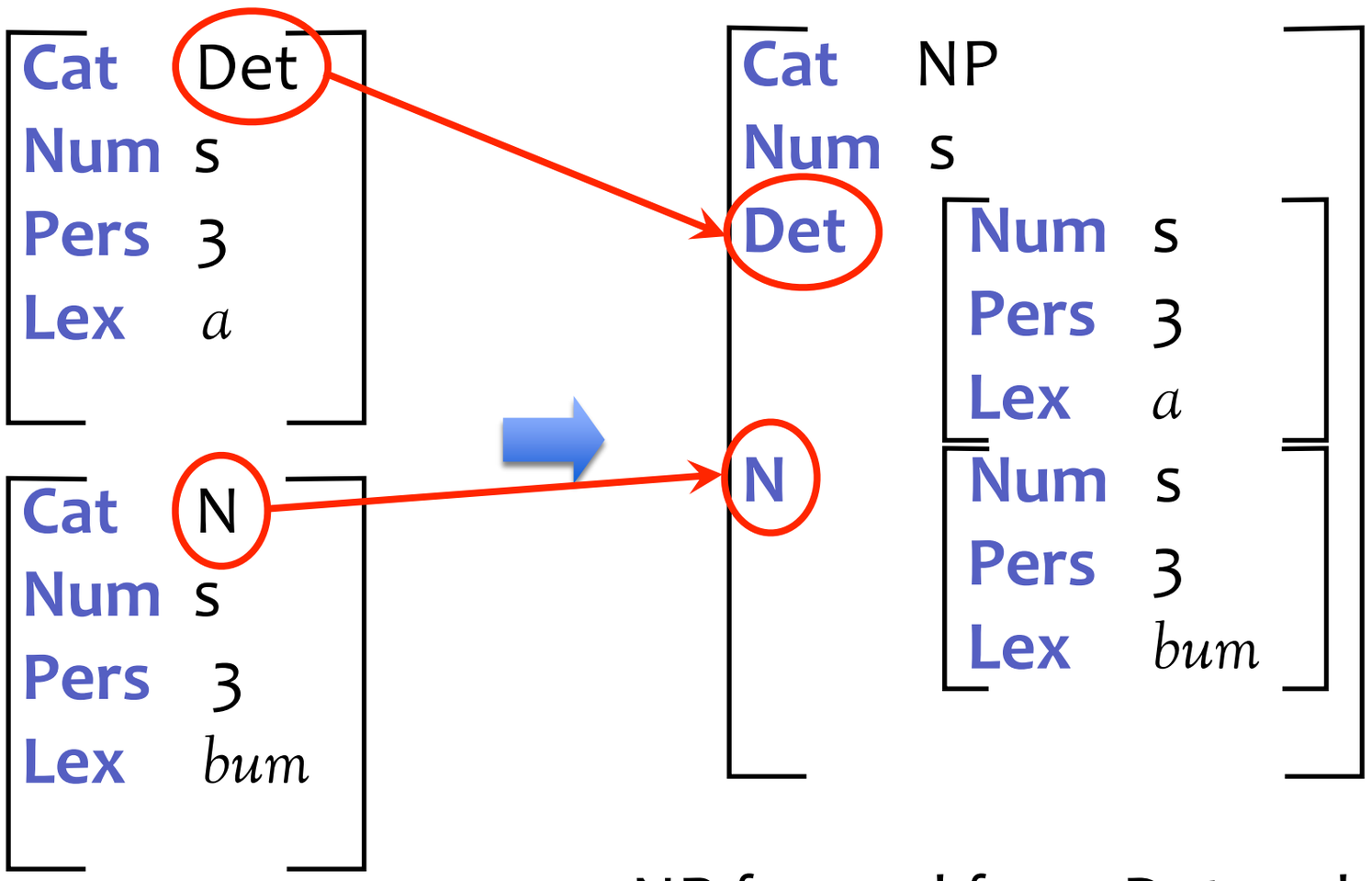
- Many equivalent notations.

N [**Num** s; **Pers** 3; **Lex** *dog*] = *dog* [**Cat** N; **Num** s; **Pers** 3] = N/*dog* [**Num** s; **Pers** 3]

[**Cat** N; **Num** s; **Pers** 3; **Lex** *dog*] = [**Cat** N; **Agr** [**Num** s; **Pers** 3]; **Lex** *dog*]

Feature paths: features of features; *e.g.*, (**Agr Pers** 3)

Feature structures 3



NP formed from Det and N.

Feature values in components become feature names in new constituent.

Components of feature use

- **1. Lexical specification:**

Description of ***properties*** of a word: morphological, syntactic, semantic, …

dog: [Cat N; Agr 3s]

dogs: [Cat N; Agr 3p]

sleeps: [Cat V; Agr 3s]

sleep: [Cat V; Agr {1s,2s,1p,2p,3p}]

^ = 'not'

Or: ^3s

Or: N → *dog* (N Agr) = 3s

N → *dogs* (N Agr) = 3p

V → *sleeps* (V Agr) = 3s

V → *sleep* (V Agr) = {1s,2s,1p,2p,3p}

Components of feature use

- **2. Agreement:**
  - ***Constraints*** on co-occurrence in a rule — within or across phrases.
  - Typically are equational constraints.

NP → Det N
(Det **Num**) = (N **Num**)

S → NP VP
(NP **Agr**) = (VP **Agr**)

Components of feature use

- **3. Projection:**
  - ***Sharing of features*** between the head of a phrase and the phrase itself.

VP → V …

(VP **Agr**) = (V **Agr**)

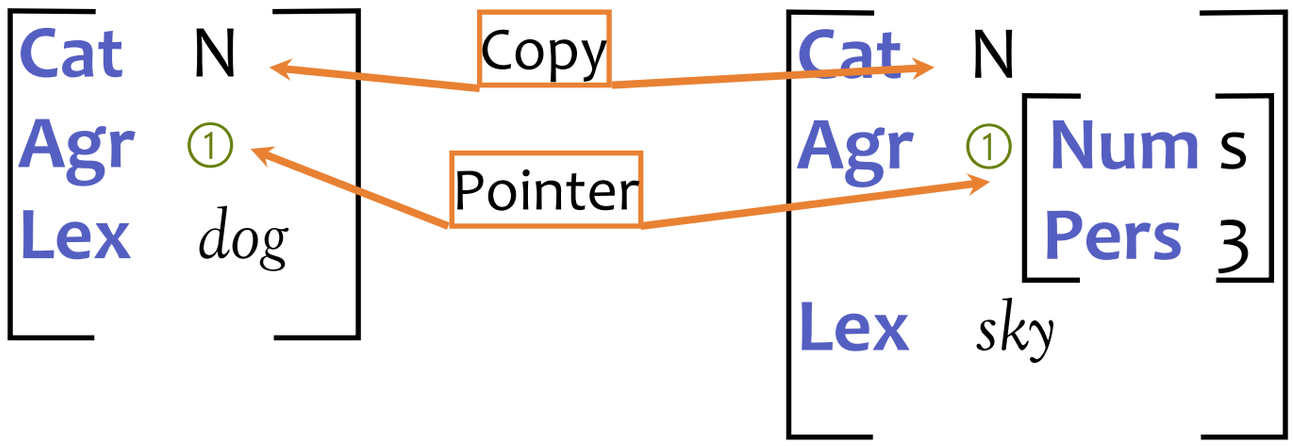
"head"

Usually the actual values are specified in the definition of the head.

- Head features:
  - **Agr** is typical, but the head word itself can be a feature.
    (Common enough that there's usually a mechanism for "declaring" head features and omitting them from rules.)

Feature pointers

- What does it mean for two features to be "equal"?
  - A ***copy*** of the value or feature structure, *or* a ***pointer*** to the same value or feature structure (re-entrancy, shared feature paths).



Constraints on feature values

- But: It may be sufficient that two features are not equal, just ***compatible*** — that they can be ***unified***.
  - *E.g.,* [Cat N, Pers 3, Num s] and [Cat N, Pers 3, Gndr F]

Subsumption of feature structures 1

- Feature structure X *subsumes* feature structure Y if Y is at least as specific as X.
  - We can also say that Y *extends* X.
    Y can add (non-contradictory) features to those in X.
- **Definition:** X *subsumes* Y (X ⊑ Y) iff:
  - For all f ∈ features of X:
    - **if** f is atomic, **then** X.f = Y.f
    - **else** X.f subsumes Y.f
  - For all paths p and q in X:
    - **if** X.p=X.q, **then** Y.p=Y.q

The value of feature f in structure X

For re-entrant structures

Subsumption of feature structures 2

- Examples:

$$\begin{bmatrix} \textbf{Cat} & \text{N} \\ \textbf{Pers} & 3 \end{bmatrix} \sqsubseteq \begin{bmatrix} \textbf{Cat} & \text{N} \\ \textbf{Pers} & 3 \\ \textbf{Gndr} & \text{F} \end{bmatrix} \quad \text{but} \quad \begin{bmatrix} \textbf{Cat} & \text{N} \\ \textbf{Pers} & 3 \\ \textbf{Num} & \text{s} \end{bmatrix} \not\sqsubseteq \begin{bmatrix} \textbf{Cat} & \text{N} \\ \textbf{Pers} & 3 \\ \textbf{Gndr} & \text{F} \end{bmatrix}$$

$$\begin{bmatrix} \textbf{Cat} & \text{VP} \\ \textbf{Agr} & \textcircled{1} \\ \textbf{Subj} & [\textbf{Agr} \quad \textcircled{1}] \end{bmatrix} \sqsubseteq \begin{bmatrix} \textbf{Cat} & \text{VP} \\ \textbf{Agr} & \textcircled{1} \\ \textbf{Subj} & \textbf{Agr} \quad \textcircled{1} \begin{bmatrix} \textbf{Pers} & 3 \\ \textbf{Num} & \text{s} \end{bmatrix} \end{bmatrix}$$

Third example from Jurafsky & Martin, p. 496

Unification 1

- The *unification* of X and Y (X ⊔ Y) is the most general feature structure Z that is subsumed by both X and Y.
  - Z is the smallest feature structure that extends both X and Y.

- Unification is a constructive operation.
  - If any features in X and Y are incompatible, it fails.
  - Else it produces a feature structure that includes all the features in X and all the features in Y.

Unification 2

$$\begin{bmatrix} \textbf{Cat} & \text{N} \\ \textbf{Pers} & 3 \\ \textbf{Num} & \text{s} \end{bmatrix} \sqcup \begin{bmatrix} \textbf{Cat} & \text{N} \\ \textbf{Pers} & 3 \\ \textbf{Gndr} & \text{F} \end{bmatrix} = \begin{bmatrix} \textbf{Cat} & \text{N} \\ \textbf{Pers} & 3 \\ \textbf{Num} & \text{s} \\ \textbf{Gndr} & \text{F} \end{bmatrix}$$

Features in chart parsing

- Each constituent has an associated feature structure.
  - Constituents with children have a feature structure *for each child*.
- **Arc addition**:
  - The feature structure of the new arc is initialized with all known constraints.
- **Arc extension**:
  - The feature structure of the *predicted* constituent must unify with that of the *completed* constituent extending the arc.

Sample grammar fragment

S → NP VP

(NP **Agr**) = (VP **Agr**)

NP → Det N

(NP **Agr**) = (N **Agr**)

(Det **Agr**) = (N **Agr**)

VP → V

(VP **Agr**) = (V **Agr**)

Det → *a*

[**Agr** 3s]

Det → *some*

[**Agr** 3p]

Det → *the*

[**Agr** {3s,3p}]

N → *dog*

[**Agr** 3s]

N → *dogs*

[**Agr** 3p]

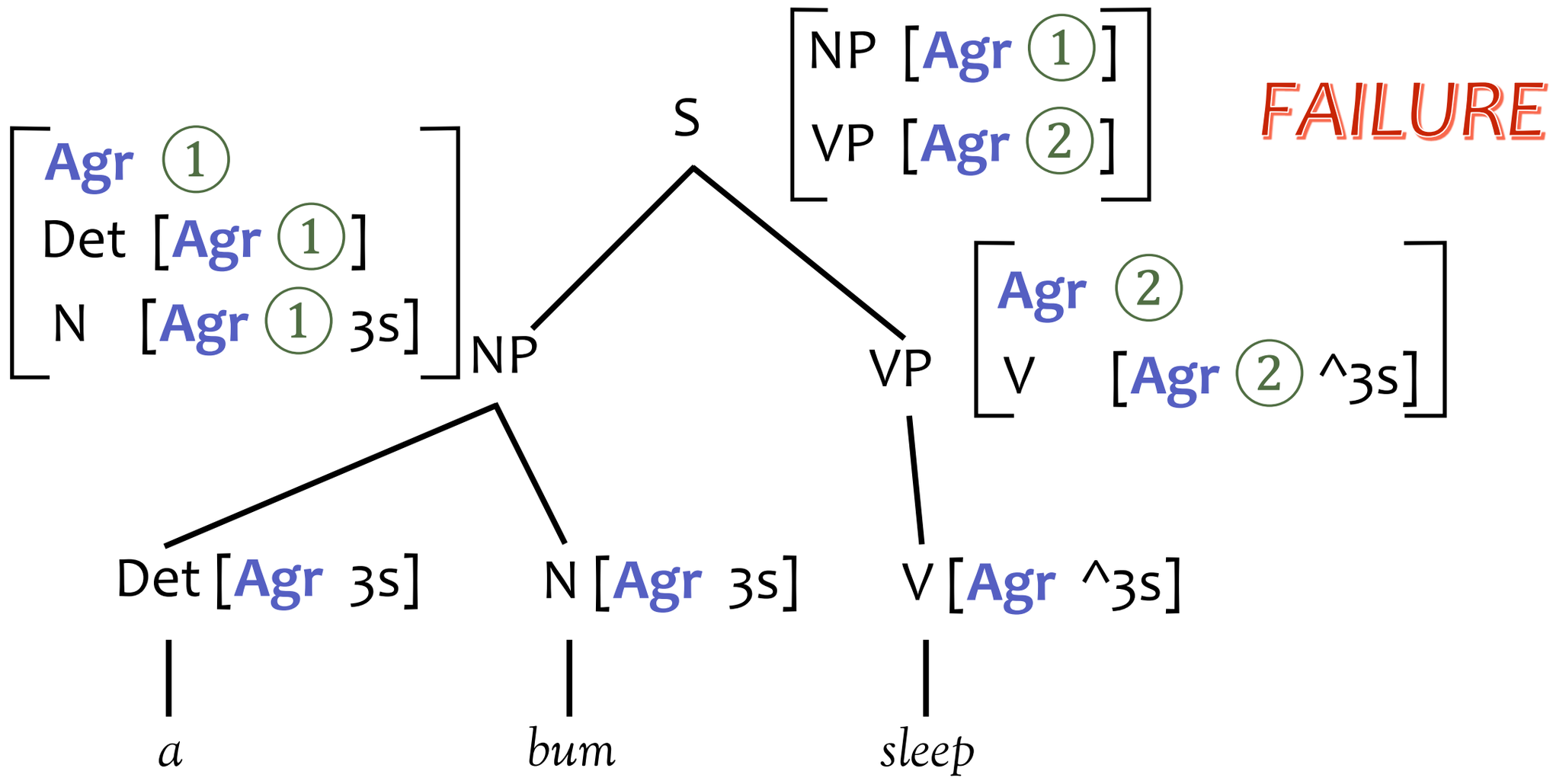
V → *sleep*

[**Agr** ^3s]

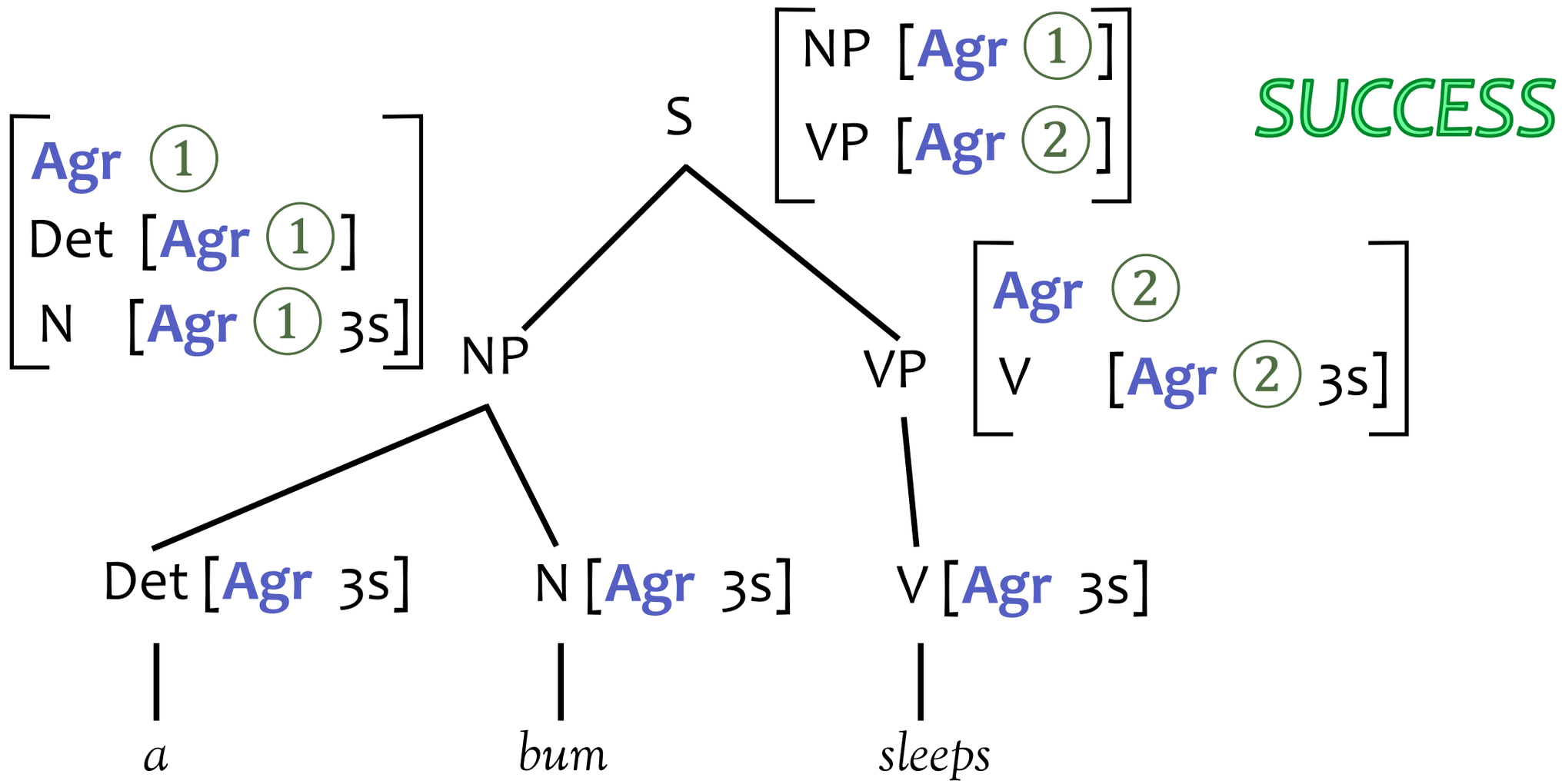
V → *sleeps*

[**Agr** 3s]

Mismatched features fail



Unifiable features succeed



Advantages

- Distinguishes structure from other info.
- Allows for economy of specification:
  - Equations in rules:

    S → NP VP

    (NP **Agr**) = (VP **Agr**)

    "Must unify with"
  - Sets of values in lexicon:

    N → *fish*

    (N **Agr** {3s, 3p})
- Allows for indirect specification and transfer of information, *e.g.*, head features.

Features and the lexicon

- Lexicon may contain each inflected form.
  - Feature values and base form listed.
- Lexicon may contain only base forms.
  - Process of *morphological analysis* maps inflected form to base form plus feature values.
  - Time–space trade-off, varies by language.
- Lexicon may contain *semantics* for each form.

Morphological analysis

- Morphological analysis is simple in English.
  - Reverse the rules for inflections, including spelling changes.

dogs → *dog* [**Agr** 3p]
dog → *dog* [**Agr** 3s]
berries → *berry* [**Agr** 3p]
buses → *bus* [**Agr** 3p]

eats → *eat* [**Agr** 3s, **Tns** pres]
ripped → *rip* [**Tns** past]
tarried → *tarry* [**Tns** past]
running → *run* [**Tns** pp]

- Irregular forms will always have to be explicitly listed in lexicon.

children → *child* [**Agr** 3p] *sang* → *sing* [**Tns** past]

Morphology in other languages

- Rules may be more complex in other languages.
- Languages with compounding (*e.g.*, German) or agglutination (*e.g.*, Finnish) require more-sophisticated methods.

Semantics as a lexical feature

- Add a **Sem** feature:

Cat	N
Num	s
Pers	3
Lex	*dog*
Sem	`dog`

`Typewriter` font for semantic objects

- The meaning of *dog* is `dog`.
  The meaning of *chien* and *Hund* are both `dog`.
  The meaning of *dog* is `G52790` (i.e., whatever).

Goal of parsing

- A representation of properties relevant to meaning and interpretation:
  - Things
  - Predicates (events)

  } **Entities** (e.g., in a knowledge base)

  - Roles } **Relations** between things and predicates.
- Syntactic structure helps in:
  - Determining **things** and **predicates**.
  - Determining mapping of **things** to **roles** of **predicates**.

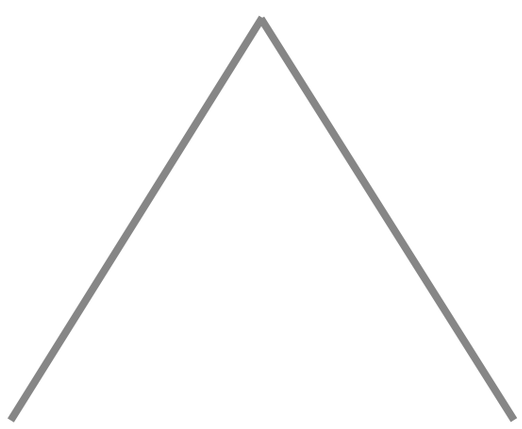
Example

The goalie kicked the bum.

Event: *kicked*

Role: Agent
(doer)

Role: Theme
(thing affected)



Thing: *The goalie*

Thing: *the bum*

```
kick (agent=goalie, theme=bum)
```

Syntax ↔ interpretation

- Mapping from structure to *objects of interpretation*
  - Things: NPs, Ss
  - Predicates: verbs, preps, APs
  - Roles: ??

- What are the roles in these examples?

Sara left.
Ben found the treasure in the garage.
Rosamund put the ball in the garage.
Tim cut the wire with a pair of scissors.
Ottawa was visited by Melissa and Nadia.
Andrew felt like a failure.

Syntax ↔ thematic roles

- Mapping is more or less regular:

Subject ≈ Agent / Experiencer
Object ≈ Theme
Object of preposition ≈ Goal/Location/
Recipient / Instrument

- This mapping is used to determine appropriate semantic representation.

Verb subcategorization 1

- **Problem:** Constraints on verbs and their complements.

*Nadia told / instructed / *said / *informed Ross to STFU.*
*Nadia *told / *instructed / said / *informed to STFU.*
*Nadia told / *instructed / *said / informed Ross of the requirement to STFU.*

Nadia gave / donated her painting to the museum.
*Nadia gave / *donated the museum her painting.*

Nadia put / ate the cake in the kitchen.
*Nadia *put / ate the cake.*

Verb subcategorization 2

- VPs are much more complex than just V with optional NP and/or PP.
  - Can include more than one NP.
  - Can include clauses of various types:

(knew) *that Ross caressed the marmoset*

(wanted) *to pay him the money*

- **Subcat**: A feature on a verb indicating the kinds of verb phrase it allows:

_np, _np_np, _inf, _np_inf, …

Write this way to distinguish from constituents.

Verb tense and aspect 1

- **Tense** and **aspect** markings on verb:
  - Locate the event in time (relative to another time).
  - Mark the event as complete/finished or in progress.

Nadia rides the horse. — In progress now.

Nadia rode the horse. — Completed before now.

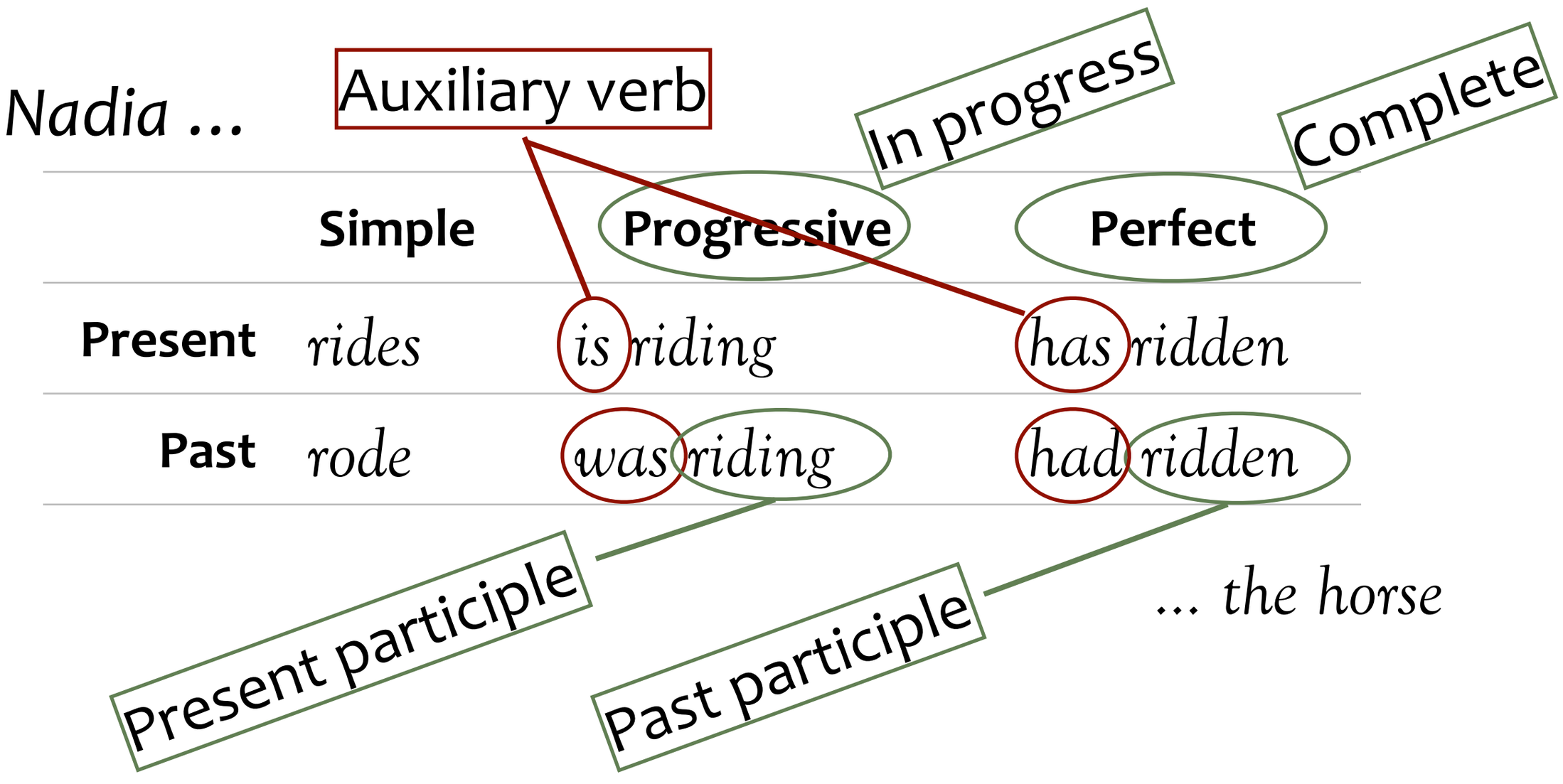
Nadia had ridden the horse. — Completed *before* before now.

Nadia was riding the horse. — In progress before now.

⋮

Verb tense and aspect 2

- **Tense**: past or present
- **Aspect**: simple, progressive, or perfect



Verb tense and aspect 3

- Tense: past or present
- Aspect: simple, progressive, or perfect

Nadia ...

Auxiliary verbs

	Simple	Perfect progressive (continuous)
Present	*rides*	*has been riding*
Past	*rode*	*had been riding*

... the horse

Modal verbs

- **Modal verbs:** Auxiliary verbs that express degrees of certainty, obligation, possibility, prediction, etc.

Nadia

{*could, should, must, ought to, might, will, ...*} ← Here they are

{*ride, be riding, have ridden, have been riding*}

the horse.

English auxiliary system

- Structure (so far):
  [MODAL] [HAVE] [BE] MAIN-VERB
- General pattern:

  VP → AUX VP

  AUX → MODAL | HAVE | BE
- Use features to capture necessary agreements.

Voice 1

- **Voice:** System of assigning thematic roles to syntactic positions.
  - English has **active** and **passive** voices**.**
- Passive expressed with *be*+past participle. Other auxiliaries may also apply, including progressive *be*.

Nadia was kissed.	*Nadia was being kissed.*
Nadia had been kissed.	*Nadia had been being kissed.*
Nadia could be kissed.	*Nadia could have been being kissed.*

- Structure:
  [MODAL] [HAVE] [BE_1] [BE_2] MAIN-VERB

Voice 2

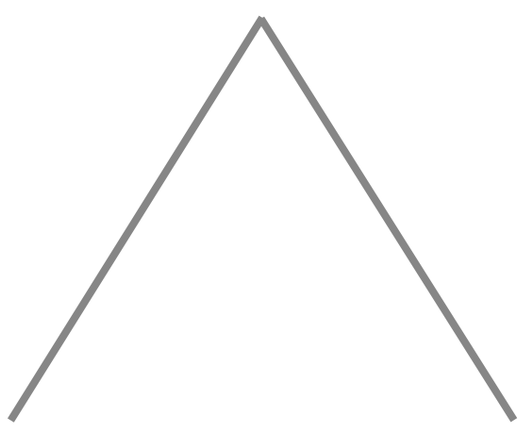
The goalie kicked the bum.

ACTIVE

Event: *kicked*

Role: Agent
(doer)

Role: Theme
(thing affected)



Thing: *the goalie*

Thing: *the bum*

```
kick (agent=goalie, theme=bum)
```

Voice 3

The bum was kicked.

PASSIVE

Event: *kicked*

Role: Theme
(thing affected)

Thing: *the bum*

```
kick (agent=?, theme=bum)
```

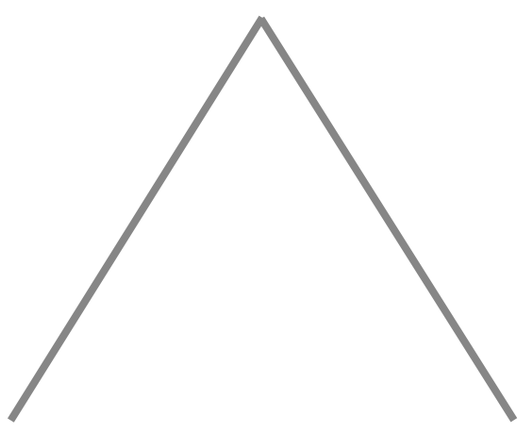
Voice 5

The bum was kicked by the goalie. **PASSIVE**

Event: *kicked*

Role: Theme
(thing affected)

Role: Agent
(doer)

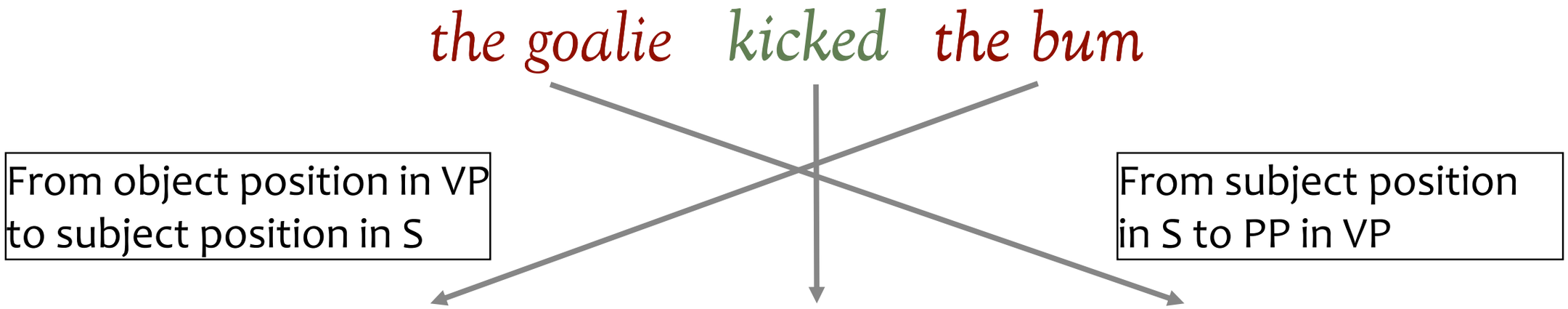


Thing: *the bum*

Thing: *the goalie*

```
kick (agent=goalie, theme=bum)
```

Passive as NP "movement"



The semantic representation doesn't change

Passive as creation of a gap

'the bum' goes from object position in VP to subject position in S

'the goalie' goes from the subject position in S to nowhere

Deletion from the semantic representation

Augmenting rules for passive voice

- For all rules of the form:

VP → V NP *X*
(V **Subcat**) = *_y*

ADD ➡

VP → V *X*
(V **Subcat**) = *_y*
(V **VForm**) = passive
(VP **VForm**) = passive

Metarule to ease grammar coding

- Augment Aux+VP rules:

VP → AUX VP
(AUX **Root**) = Be2
(AUX **CompForm**) = (VP_2 **VForm**)
(VP_2 **VForm**) = passive

The GAP feature for passive voice

S → NP VP

1 (NP **Agr**) = (VP **Agr**)
2 (VP **VForm**) = passive
3 (VP **Gap Cat**) = NP
4 (VP **Gap Agr**) = (NP **Agr**)
5 (VP **Gap Sem**) = (NP **Sem**)

VP → AUX VP

1 (VP_1 **Agr**) = (AUX **Agr**)
2 (VP_1 **VForm**) = (VP_2 **VForm**)
3 (VP_1 **Gap**) = (VP_2 **Gap**)
4 (AUX **Lex**) = be2
5 (VP_2 **VForm**) = passive

V → *kicked*

1 (V **VForm**) = {pastprt, passive}
2 (V **Subcat**) = _np
3 (V **Lex**) = *kick*
4 (V **Sem**) = `kick`

VP → V NP

1 (VP **VForm**) = (V **VForm**)
2 (VP **Gap**) = (NP **Gap**)
3 (V **Subcat**) = _np

NP → ε (Empty string)

1 (NP **Gap Cat**) = NP
2 (NP **Gap Agr**) = (NP **Agr**)
3 (NP **Gap Sem**) = (NP **Sem**)

NP → *cans*

1 (NP **Agr**) = 3p
2 (NP **Lex**) = *can*
3 (NP **Sem**) = `cans`

AUX → *were*

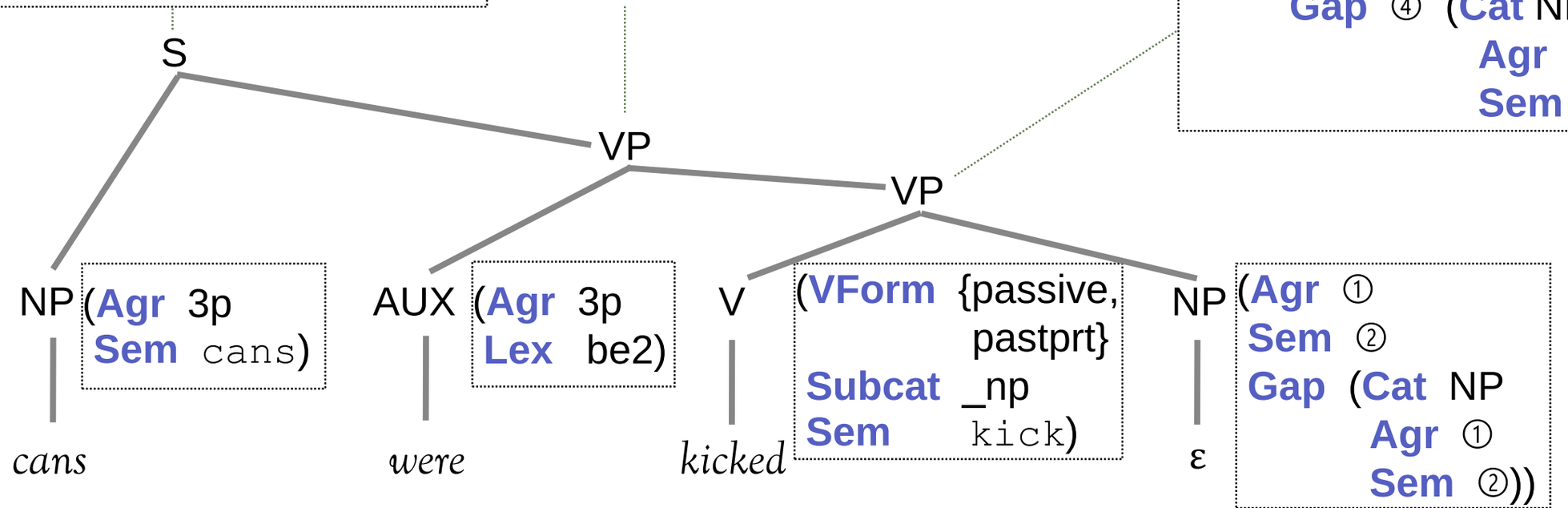
1 (AUX **Agr**) = 3p
2 (AUX **Lex**) = be2

(NP (Agr ❶ 3p
Sem ❷ cans)
VP (Agr ❶
VForm ③
Gap ④
AUX (Agr ❶ 3p
Lex be2)
VP (VForm ③
Gap ④
V (VForm ③ passive
Subcat _np
Sem kick)
NP (Agr ①
Sem ❷
Gap ④ (Cat NP
Agr ①
Sem ❷)))

(Agr ⑤
VForm ③
Gap ④
AUX (Agr ⑤ 3p
Lex be2)
VP (VForm ③
Gap ④
V (VForm ③ passive
Subcat _np
Sem kick)
NP (Agr ①
Sem ②
Gap ④ (Cat NP
Agr ①
Sem ②)))

Note: The green ❶'s of the S were ⑤'s until the 4th constraint of the rule S → NP VP. The 5th constraint fills in the **Sem** of the **Gap** ❷.

(VForm ③
Gap ④
V (VForm ③ {passive,
pastprt}
Subcat _np
Sem kick)
NP (Agr ①
Sem ②
Gap ④ (Cat NP
Agr ①
Sem ②)))



Other cases of NP "movement"

- Other constructions involve NPs in syntactic configurations where they would not get the right thematic roles using standard mapping.

Nadia seems to like Ross.
Nadia seems to be liked.
Nadia is easy to like.
Whom did Nadia like?
I fed the dog that Nadia likes to walk.

- We can use grammar rules with gap features to ensure correct structure/interpretation of these as well.

Summary

- Features help capture syntactic constructions in a general and elegant grammar.
- Features can encode the **compositional semantics** of a sentence as you parse it.
- Features can accomplish mapping functions between syntax and semantics that simplify the interpretation process.