Assignment 3 Tutorial 2

Jinyue Feng

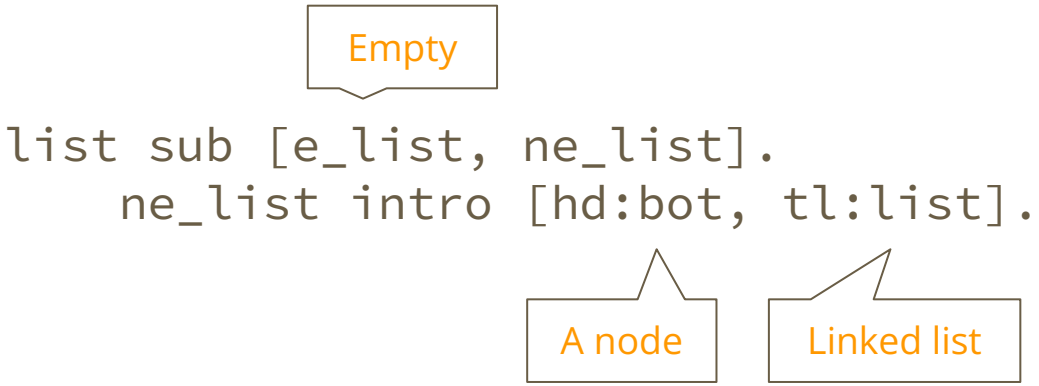
Overview

- TRALE Basics
- Grammatical Gender Agreement
- **Subcategorization**
- **Passive Voice (Gap Constuct)**
- **Semantic Head**
- **Goals and Macros**
- Beta Reduction
- Quantifier Storage

Don't forget to check out the tutorial online!

Lists

- Recursive data structure: linked list
  - A linked list is either empty or a node pointing to a linked list



List Notations

- Empty list: `[]`
- List with element(s): `[X]` `[X1, X2, X3]`
- List with head and tail: `[Head|Rest]`
- List with at least one element: `[_|_]`

Subcategorization

- Subcategorization refers to the ability of lexical items (typically verbs) to require or allow co-occurring syntactic arguments.
  - A transitive verb requires one object (I notice books).
  - A ditransitive verb requires two objects (I give him books).

- Lists are useful for subcategorization (`subcat`)

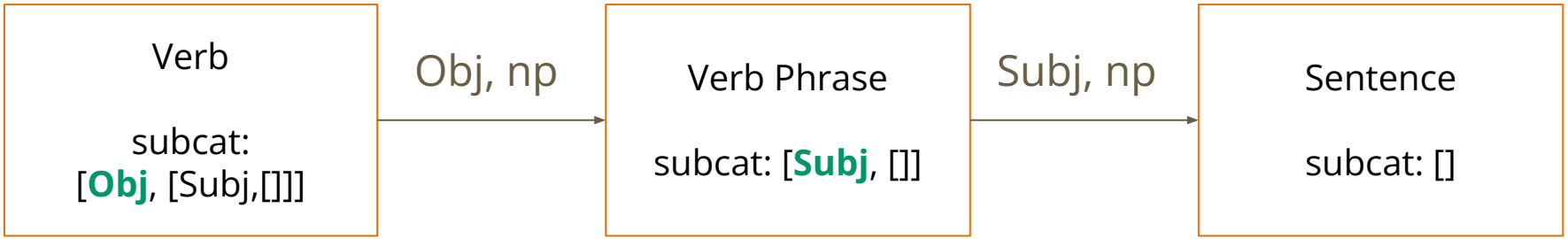
Subcat Example

```
notice  --->  (v,  sem:notice,  agr:person:first,

    subcat:  [(Obj,  np,  agr:case:objective),

        (Subj,  np,  agr:case:subjective)]).

vp rule  (vp,  sem:Sem,  agr:Agr,  subcat:(Rest,  [_|_]))  ===>

cat>  (verbal,  sem:Sem,  agr:Agr,  subcat:  [Obj|Rest]),

cat>  Obj.
```

List the items in the reverse order of which they are consumed in the construction process!

(Objects before the subject)

How are lists processed



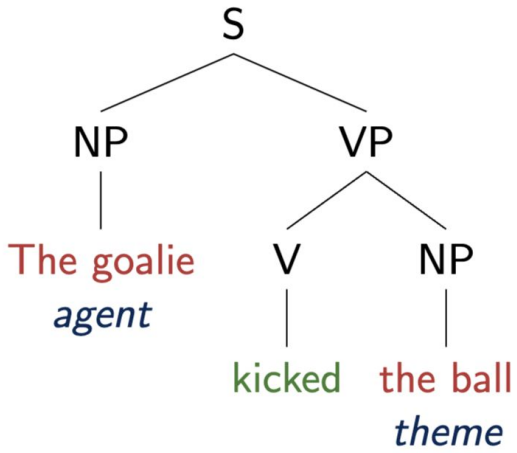
Gaps

- Goal of Gaps: Passive Voice
  - In active voice, the agent is the subject and the theme is the object.
  - In passive voice, the theme becomes the grammatical subject.
  - **The gap stores the moved noun phrase.**

```
gap_struc sub [np, none].
```

The Active Voice

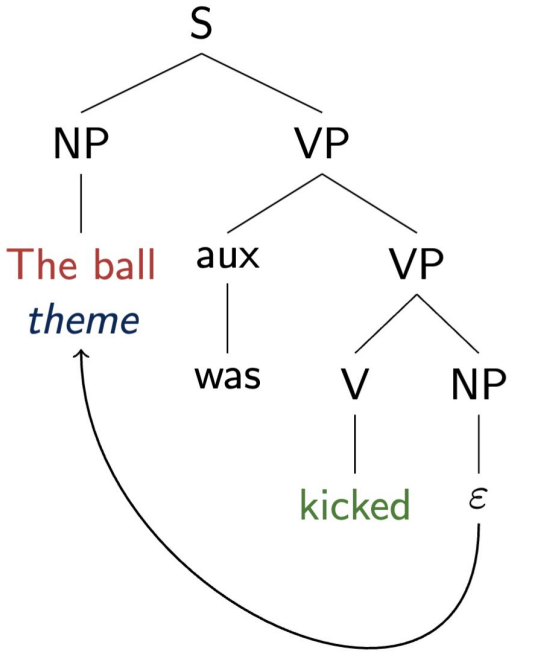
The goalie kicked the ball.




```
sem:(kick, agent:goalie, theme:ball)
```

The Passive Voice

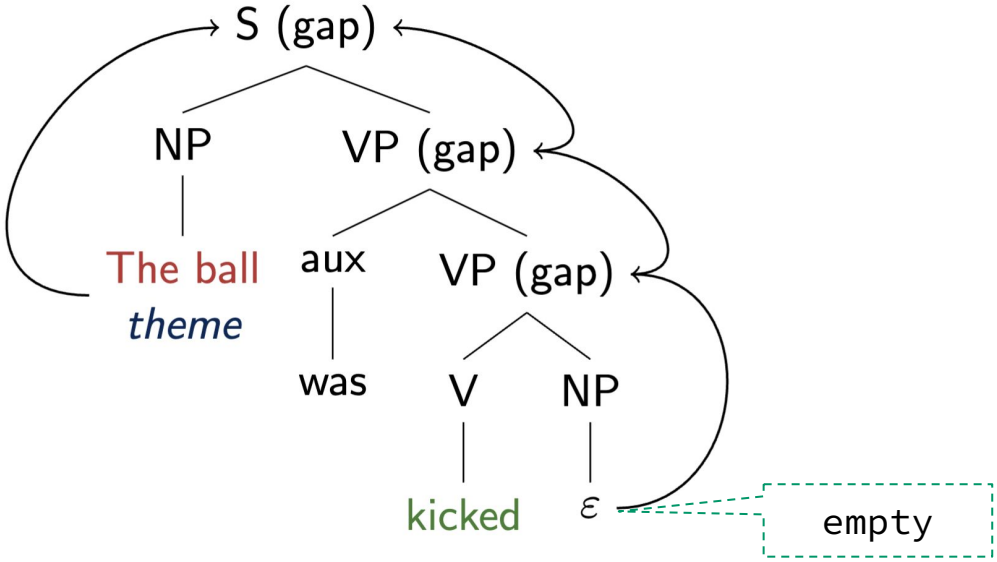
The ball was kicked.



`sem:(kick, agent:?, theme:ball)`

Passing the Gap in the Passive Voice

The ball was kicked.



`sem:(kick, agent:?, theme:ball)`

Example: g3.pl

- Subcat
  - List declaration
  - Lexical entry: define subcat for verbs
  - How are the lists recursively processed
- Gap
  - `Gap_construct` and the `empty` category
  - How gap is passed
  - How to define the passive voice rules

Semantic Head

- The semantic head determines the semantic category of a product type
  - N → NP
  - V → VP
  - CL → CLP
  - VP → S
- **Replace `cat>` with `sem head>` on all the semantic heads**

Semantic-Head-Driven Generation

- Translation: parsing and generation
  - You may test your grammar's generation without loading up the other grammar with `gen!`

```
gen((s, sem:(chase, subj:student, obj:sheep)).
```

- TRALE uses semantic-head-drive generation to speed up the generation process
  - It would take too long to traverse all of the possible rules
  - Semantic-Head-Driven generation utilizes the fact that the semantics of the semantic head and the product are exactly the same

To read more on this: see Chapter 6 of TRALE User's Manual

Goals and Macros: Functions in TRALE

- Goal: A logical function that returns true or false

```
goal_name(Arg1, Arg2, ...) if true.

goal_name(Arg1, Arg2, ...) if
    condition1(...),
    condition2(...).
```

- Macro: Wrapper for complex expressions

```
macro_name(Arg1, Arg2, ...) macro (type, arg1:Arg1, ...).

@macro_name(Arg1, Arg2, ...)
```

Goals in Rules

- Use the `goal>` keyword
- Enforce constraints: the parser only accepts a sentence (or phrase) if the goal evaluates to true

- We can also use goals and Prolog functions to debug our grammars
  - Example debug.pl

Questions?