LING 364: Introduction to Formal Semantics

Lecture 9

February 9th

Administrivia

- Reminder
  - Homework 2 due next Tuesday
  - *need help getting started*?

Administrivia

- **today**
  - (3:30pm – 4:40pm)
    - lecture here in Comm 214
  - (4:45pm – 5:45pm) (**EXTRA)**
    - lab practice in Social Sciences Lab 224
    - *we'll begin doing the homework exercises in the lab*

Today's Topics

- Grammar Rule Recursion
  - Prolog behavior
- Handout (from Tuesday)
  - Chapter 3: *More about Predicates*
    - Short Quiz #3 on Thursday

Grammar Rule Recursion

- **Recursion**:
  - A phrase may contain embedded inside another instance of the same phrase
- Example:
  - sentence with a **relative clause**
  - $[_{Sbar}\ [_{S}$ I saw $[_{NP}$ the man $[_{Sbar}$ who $[_{S}$ attacked me]]]]]
  - $[_{Sbar}\ [_{S}$ I saw $[_{NP}$ the man $[_{Sbar}$ who $[_{S}$ attacked $[_{NP}$ the dog $[_{Sbar}$ who $[_{S}$ attacked me]]]]]]]

Grammar Rule Recursion

- Example:
  - assuming NP (not DP analysis) for simplicity...
  - $[_{NP}\ [_{NP}$ John] 's mother]
  - $[_{NP}\ [_{NP}\ [_{NP}$ John]'s mother]'s cat]
- DCG rules:
  - `np --> np, ['''s'], n.`
  - `n --> [mother].`
  - `n --> [cat].`
  - `np --> [john].`

```
'''s' = 's
```

Grammar Rule Recursion

- Prolog Computation Rule:
  - select “first” matching grammar rule each time we call a non-terminal
  - “first” = first line that matches
- DCG rules:
  - `np --> np, [```s`], n.`
  - `n --> [mother].`
  - `n --> [cat].`
  - `np --> [john].`
  - *Leads to infinite loop here...*

Grammar Rule Recursion

- **General Rule for writing recursive rules:**
  - put recursive case **last**
  - i.e. *place non-recursive rules for a non-terminal ahead of the recursive ones*
- DCG rules:
  - `np --> [john].`
  - `np --> np, ['''s'], n.`
  - `n --> [mother].`
  - `n --> [cat].`
  - *no looping here...*

Grammar Rule Recursion

- You'll need it for homework 2...
- Examples:
  - $[_{Sbar}\ [_{NP}$ Who$]\ [_{S}\ [_{VP}\ [_{V}$ is$]\ [_{NP}[_{NP}\ [_{DET}$ a$][_{N}$ student$]]]]][_{CONJ}$ and$][_{NP}\ [_{DET}$ a$][_{N}$ baseball fan$]]]]]]$
  - $[_{Sbar}\ [_{NP}$ Who$]\ [_{S}\ [_{VP}\ [_{V}$ is$]\ [_{NP}[_{NP}\ [_{DET}$ a$][_{N}$ student$]]]]][_{CONJ}$ and$][_{NP}\ [_{NEG}$ not$][_{NP}[_{DET}$ a$][_{N}$ baseball fan$]]]]]]]$

- Consider a possible NP rule for conjoining two NPs:
  - `np --> np, conj, np.`
  - `conj --> [and].`

More about Predicates

- 3.1 Other Types of Predicates: Adjectives, Predicate Nominals
  - (1) Shelby is small
  - (2) Shelby is a dog
- Semantics of *is* and *a.*
- Possibilities:
  - Meaningless
  - Non-interfering meaning - trivial meaning

More about Predicates

- 3.1 Other Types of Predicates: Adjectives, Predicate Nominals
  - (1) Shelby is small
  - (2) Shelby is a dog
- Semantics of (indefinite determiner) *a.*
  - (3) a dog bit me
  - (4) the/one/every dog bit me
  - quantifier?

More about Predicates

- Semantics of (indefinite determiner) *a.*
  - (3) a dog bit me
  - (4) the/one/every dog bit me

  - quantifier?

  - (3') there exists a dog x such that bit(x,me)
  - (4') *every*: for each dog x, bit(x,me)

More about Predicates

- Semantics of (indefinite determiner) *a.*
  - (3) a dog bit me
  - (3') there exists a dog *x* such that bit(*x*,me)

  - (2) Shelby is a dog
  - semantics involving "there exist a dog *x*"

More about Predicates

- Semantics of (indefinite determiner) *a.*
  - (3) a dog bit me
  - (3') there exists a dog *x* such that bit(*x*,me)

  - (2) Shelby is a dog
  - semantics involving "there exist a dog *x*"
  - No...

More about Predicates

- 3.2 Transitive Verbs
  - (5) Shelby saw Hannibal
- 3.3 Relative Clauses
  - (7) Hannibal is [who Shelby saw]
  - semantics of [who Shelby saw]

More about Predicates

- 3.3 Relative Clauses
  - (7) Hannibal is [who Shelby saw]
  - semantics of [who Shelby saw]
  - Shelby saw who — saw(shelby,who).
  - (with logic variable) — saw(shelby,X).

More about Predicates

- 3.4 Topicalization
  - (9) Shelby, Mary saw

- Semantics?
- Paraphrase (9) as:
  - (10) Shelby is who$_1$ Mary saw e_1

More about Predicates

- 3.5 Sub-atomic Semantics
- Event semantics
  - (11) Sylvia petted Shelby
- introduce an **event variable**, call it e
- Prolog-style, we can say:
  - event(e), agent(e,sylvia), patient(e,shelby).
- Notions like:
  - agent, patient, instrument etc.

  are called **thematic roles**

More about Predicates

- **lambda calculus:**
  - *easy to introduce now...*
- Example:
- *barks*: λx.x barks barks(X).

  - Shelby barks
  - [λx.x barks](Shelby)
  - barks(X), X = shelby

- Generalization:
  - [λx.[λy.y saw x]

Quiz 3

- (3pts)
- Give lambda calculus semantics for:
  - likes
  - likes Mary
  - John likes Mary